*I. J. Computer Network and Information Security,* **2019, 4, 35-42**
Published Online April 2019 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2019.04.05

# A DOS and Network Probe Attack Detection based on HMM using Fuzzy Inference

**Mohsen Salehi**
Imam Reza International University, Mashhad, Iran
E-mail: Mohsen.salehi@Imamreza.ac.ir

**Jamal Karimian**
Imam Reza International University, Mashhad, Iran
E-mail: jamal.karimian@Imamreza.ac.ir

**Majid Vafaei Jahan**
Islamic Azad University, Mashhad Branch, Iran
E-mail: VafaeiJahan@mshdiau.ac.ir

Received: 12 August 2018; Accepted: 05 September 2018; Published: 08 April 2019

***Abstract*—**This paper aims to provide an intrusion detection system for network traffic that achieves to the low false positive rate with having high attack detection rate. This system will identify anomalies by monitoring network traffic. So, Features extracted from the network traffic by the number of HMM, are modeled as a Classifier ensemble. Then by integrating the outputs of the HMM within a group, probability value is generated. In this system each feature receives a weight and rather than a threshold value, using the fuzzy inference to decide between normal and abnormal network traffic. So at first, the fuzzy rules of decide module are formed manually and based on the value of the security of extraction feature. Then probability output of each HMM groups converted to fuzzy values according to fuzzy rules. These values are applied by a fuzzy inference engine and converted to an output indicating the being normal or abnormal of network traffic. Experiments show that the proposed system in detecting attacks that are the main candidate error is working well. Also, measures recall, precision and F1-measure respectively with 100%, 99.38% and 99.69% will pass. Finally, attack detection rate close to 100% and false positive rate of 0.62%, showing that the proposed system is improved compared to previous systems.

***Index Terms*—**DOS, Probe, HMM, Fuzzy inferences, Attack detection.

## I. INTRODUCTION

Nowadays Internet has a big part in our daily life and has been developed in most of the social, scientific and other aspects, but is not without trouble. The main issue that we face is information leak and unauthorized access to sources. Having such a problem forced security and protection as the main area discussed in computer systems and programs.[1, 2]. The intrusion detection is a process of monitoring the events that occurring in a computer system or a computer network. Which that's aims is to detect signs of intrusion and unauthorized intrusion detection system to implement this process. Intrusion detection systems are operationally divided into two classes: signature-based and anomaly-based [3].

Signature-based systems: This system is aimed to detect attacks They have occurred in the past and although Their error rate is better than anomaly-based in intrusion detection systems, But they act week to discovering Zero day attacks, that this weakness have been greatly relieved in the anomaly-based systems. The main goal of this work is to develop an intrusion detection system with high accuracy and minimum error that discover attacks with Monitoring network traffic and anomaly detection in them. Anomaly-based systems: Unfortunately, numbers of Zero day attacks that appear every day are is steadily increasing. Traditional signature-based systems that done detect attacks process by comparing the existing database attacks, will be helpless in the face of new attacks. A proposal to this problem would be to always have the latest attacks are immediately recorded in the database that would be very costly to operate. A possible solution to solve this is use as anomaly-based systems is proposed. Anomaly-based system makes the Model of normal behavior of a system that is supposed to protect it and compared attack pattern with normal model and In case of any inconsistency it detects abnormal [4]. The proposed system is configured according to structure of the intrusion detection systems based on hidden Markov models and uses Hidden Markov model and fuzzy inference to solve the problem of detecting anomalies.

The important tasks in anomaly diagnosing are modeling and decision which if they are done well, better performance could be observed in anomaly detecting. The proposed system has two approaches that cause

mentioned tasks to be performed well and consequently the rate of error to be decreased and attack detection rate to be increased. First approach, In order to make the error due to the limited number of categories in the decision of the board, the proposed system uses a simple majority. Thus, the use of multiple categories for each feature extracted from a network traffic, we produce a model that the obtained outputs from each category will eventually merge and thus, some sort of error will be ignored. Second approach, In most previous works that they separate normal and abnormal network Traffic by using threshold value that is when the similarity of normal or abnormal packets are abundant, system has been impaired.to be able to separate the normal or abnormal network traffic packets when the packets have abundance similarity, The Fuzzy inference is used. According to the two mentioned approaches, the proposed intrusion detection system reaches a high detection rate while offers the lowest false positive rate. First, in Section 2, some of the works that have been done in this area will be studied. Section 3 deals with methods to reduce the error rate and increasing the accuracy of attack detection. InSection4, the proposed system is described and in Section 5, it will be evaluated and tested. Finally conclusion is presented in Section 6.

## II. Related Works

Valuable work has been done about intrusion detection. Among these activities that is common to use powerful modeling tool called Hidden Markov Models, we will briefly use HMM instead of Hidden Markov models. HMM is a suitable tool to model normal behaviors that can be identified the noises and abnormal behaviors. This tool in other activities, such as motion detection and speech recognition has also been used. Corona et al [5] presented a new framework in which the submitted queries to web applications, is analyzed using HMM and pay special attention to noise in the training data. Yong zhong Lee [6] has used fuzzy approaches for the HMM where Flexibility, system has Adaptability capabilities to changing patterns. As a result it improved in the recognition of Zero Days Attacks.

Hmong Zhang [10] defined Fuzzy scheme for intrusion detection system to identify the disordering's in a system program system calls, Where used fuzzy logic is instead of using classical fixed terms. Ajyth Abraham al [13] used soft computing techniques, Particularly, Fuzzy logic and neural networks in order to build a powerful and flexible intrusion detection system. This system is based on multiple modeling with different measurements and finally fuzzy logic has been used to make the final decision. Kruegel [9] has introduced a multi-model framework to detect attacks against web applications that analyzed Received queries in terms of both spatial and temporal characteristics .Among various modeling that is presented, HMM left better results. Estevez Tapiader [11] has proposed a method based on the monitoring "incoming HTTP requests" through a Markov model that includes a set of states and transitions between them, tries to detect attacks against web servers and decides about being Normal or attack state of a HTTP request, due to the HTTP protocol specification.

## III. Reduce False Positive Rate

The main goal of this study is to reduce the error rate. Error may include attack that intrusion detection system recognizes it as a normal network traffic (false negatives) or normal network traffic which system detects an attack (false positives). Usually in an intrusion detection system, for network traffic or any extracted feature, you see just one trained HMM and only use the same model to classify network traffic. This leads to the desired result is affected by errors in the model. These errors may be due to inappropriate training data set or the use of unsuitable training algorithms or improperly configured HMM.

Also threshold value that obtained in the training phase and Trial and error, for network traffic that are main error candidate does not work well and increase the system error. As you can see in Figure1 shown Points of two Normal or abnormal network traffic respectively are marked with blue points and red points, separated by a line. In the meantime points that are located near the line are the main candidate error, So that even the smallest change in the normal patterns causes the increasing of false positive rate. Since we involved a large volume of data, so situation has a high-risk. Thus it is possible that similarity of packets to each other cause a large error.

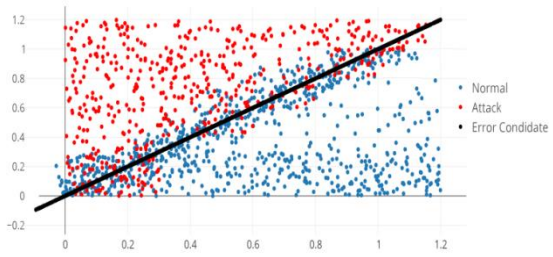



Fig.1. Points of Normal or Abnormal Network Traffic Respectively are Marked with Blue Points and Red Points, These Points are Separated by a Line Which Points Are Located near This Line are the Main Error Candidates.

In order to cross this problem that occurred and also improve the accuracy and reduce detected anomaly errors, we use the two solutions which include Multiple Classifier Systems and Fuzzy.

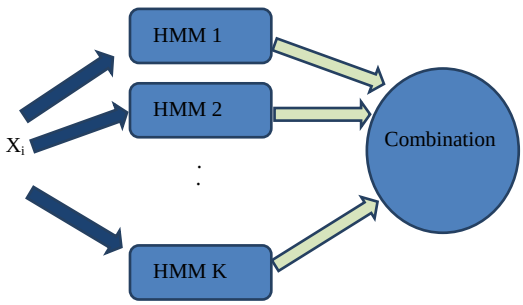



Fig.2. Input xi is sent to Every HMM, and their Output is converted by Combiner to a Produce Final Output.

### A. Multiple Calssiffier System

Multiple Classifier Systems (MCS) has been used widely in pattern recognition problems, So that shown more accuracy rather than single Classifier systems. The reasons can be found in [3, 4]. Figure 2 shows an example of the Multiple Classifier Systems.

As shown in Figure 2, in this work used several HMM as a Classifier, and Input xi will be sent to each of the HMM within the group. The output of each HMM is combined by a combiner, Si as a final output is produced [2]. According to [12] for computational and statistical reasons, Classifier ensembles are used Instead of a Classifier. There are many integrated functions that each has its own characteristics. Here are the rules, largest, smallest, mean and the geometric mean is shown [2]:

$$S_i^* = \max \{S_{ij}\} \quad (1)$$

$$S_i^* = \min \{S_{ij}\} \quad (2)$$

$$S_i^* = \frac{1}{N} \sum_{j=1}^{N} S_{ij} \quad (3)$$

$$S_i^* = \left[\prod_{j=1}^{N} s_{ij}\right]^{\frac{1}{N}} \quad (4)$$

### B. Fuzzy

One of the soft computing techniques for anomaly detection is fuzzy logic that is discussed in [10].anomaly detection based on fuzzy sets and rules leads to better results for following reasons:

a) Since the being normal or abnormal is not absolute concept, the definition of a certain concept causes Sharp distinction between normality and abnormality. So it is natural to use fuzzy sets for packets that are the main candidate error [15, 16].
b) Anomaly detection system based on fuzzy inference can combine received inputs from several sources, which this will improve the diagnosis efficiency [15].

## IV. The Proposed Intrusion Setection System

Generally intrusion detection systems based on machine learning algorithms are developed in two phases: training and testing. In the training phase, the system models and modules are configured with respect to the training data. In Figure 3, you can see the overview of system. In the testing phase, the system is organized according to the data assessed by normal and abnormal operations. Probable value resulted from HMMs is combined in an HMM ensemble and final considered output is obtained. Then with fuzzifying the output of each HMM ensemble, Inference process becomes ready to be performed. In the inference process, by applying fuzzy sets and rules, inference engine specifies normal and abnormal network traffic.

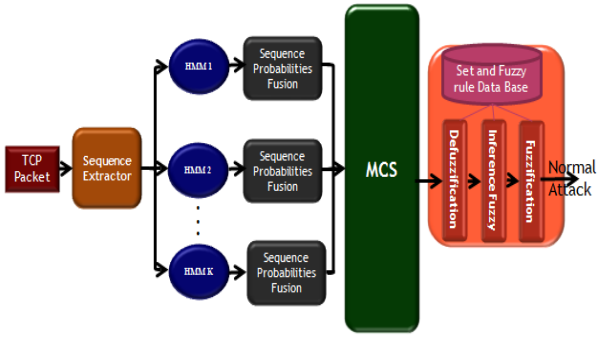



Fig.3. A Simplified Scheme of Proposed System.

The sequence extractor extracts five features from the network traffic. Then, the value of each feature is sent to the processor and the bands corresponding HMM. Then HMMs of each group, according to the past learning itself produce probability that shown Normal rate of each features. Then probability values are merged to be formed output of each HMM. Finally, fuzzy inference system according to fuzzy rules uses the outputs of each group to produce a final vote on the issue of the network traffic is normal or abnormal.

### A. Data Preprocessing and Feature Extraction

This study focuses on five important features of network traffic KDD Cup 1999 Centralization, which are shown in Table1.

Table 1. Features Extracted from Network Traffic KDD Cup 1999

| Feature | Type | named |
|---|---|---|
| src_bytes | continuous | Src |
| dst_bytes | continuous | Dst |
| logged_in | continuous | Log |
| dst_host_count | continuous | Hst |
| dst_host_srv_count | continuous | srv |

The main reasons selecting these features is listed as follows:

a) Numerical Variation of selected features is more than usual; so their training is more detailed, and they are easier to diagnose.
b) Other features are labeled as floating-point numbers, which have a lot of computational complexity; therefore, we are considering Normal Numbers.
c) One of the possible scenarios in training and recognition is to use these five features together.

Achieving this purpose HMM is applied as a modeling tool, and it is applied to model extracted features of network traffic. According to the point that all features extracted from values domain are selected continuously in the training phase to determine the magnitude of observed variables, algorithm1 and algorithm2 are used for preprocessing of Training and test data. In fact, pre-processing training and preprocessing test algorithms are applied to selected features and their values in order to improve system performance and consequently to reduce the size of data sets.

Algorithm1: preprocessing train data

```
Function pretrain (Normal Folder)
  For Trace File in Normal Folder do
   */extract five feature of normal file /*
  End for
    For count to end do
     */allocation unique number to all normal items /*
     End for
End function
```

Algorithm2: preprocessing test data

```
Function pretest (test folder)
   For Trace File in Test Folder do
     If test items==normal items then
       Test items =preprocess normal
     Else
      */allocation unique number to test items /*
     End if
    End for
End function
```

The training data is shown in Table 2; therefore the preprocessing is shaped in Table 3. Now, we can use tables 2 and 3 for preprocessing of test data, and the test data output will be processed in Table 5.

Table 2. Training Data

| Train data | | | | |
|---|---|---|---|---|
| 10 | 271 | 0 | 235 | 4 |
| 1252 | 410 | 1 | 239 | 148 |
| 702 | 332 | 1 | 240 | 149 |
| 233 | 2032 | 1 | 3 | 3 |
| 667 | 332 | 1 | 242 | 150 |

Table 3. Pre-processing of Training Data

| Preprocess-train | | | | |
|---|---|---|---|---|
| 1 | 6 | 10 | 12 | 16 |
| 2 | 7 | 11 | 13 | 17 |
| 3 | 8 | 11 | 14 | 18 |
| 4 | 9 | 11 | 15 | 15 |
| 5 | 8 | 11 | 16 | 19 |

Table 4. Test Data

| Test Data | | | | |
|---|---|---|---|---|
| 10 | 39 | 1 | 8 | 23 |
| 246 | 515 | 1 | 255 | 255 |
| 251 | 4137 | 1 | 255 | 255 |
| 252 | 7886 | 1 | 255 | 255 |
| 233 | 2032 | 1 | 3 | 3 |

Table 5. Pre-processing of Test Data

| Preprocess-test | | | | |
|---|---|---|---|---|
| 1 | 23 | 11 | 27 | 29 |
| 20 | 24 | 11 | 28 | 28 |
| 21 | 25 | 11 | 28 | 28 |
| 22 | 26 | 11 | 28 | 28 |
| 4 | 9 | 11 | 15 | 15 |

As seen here, the data processing must be maintained by these three conditions in order to have a fair look at the size of datasets:

ها باید سه شرط اصلی زیر برقرار باشند تا نگاهی عادلانه به کم مجموعه داده ها داشته باشیم:

a) Similar rows would receive 5 similar assigned digits in advance (Red).
b) Cells are assigned the same numbers on their previous experience (Blue).
c) Since it is possible to see similar numbers in pre-processing each column, previous allocated amounts would be assigned (Green).

## B. Train Markov Model

As you would expect, here we learn and asses the proposed model. The proposed system will use Baum-Welch algorithm written in [8], to teach HMM method. Since HMM design is an empirical matter and depends on trial and error, you can get the best educational condition by changing the parameters of mode numbers, initial state, symbol Distribution matrix and mode transition matrix. Figure 4 proves this claim. As seen here, when less state numbers are used, the accuracy of HMM decreases and consequently less training time is spent; but when more state numbers are used, the accuracy of each HMM Increases, therefore more learning time is spent. This process continues until the system becomes almost stable. Regarding that learning accuracy have direct correlation with learning time, we have used 15 states to maintain a good balance between time and accuracy.

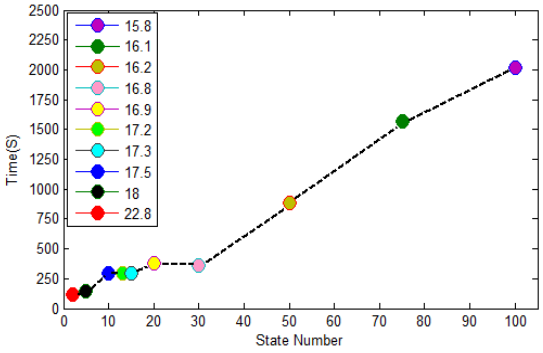



Fig.4. Effect of the Number of States on Learning Outcomes. Horizontal Axis Is The Number Of States And Vertical Axis Is Time Learning. Dialog Box In The Diagram Represents The Amount Of Learning.

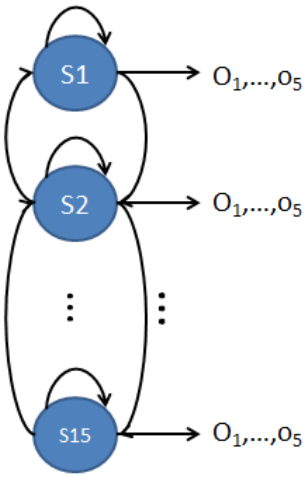



Fig.5. Hidden Markov Model of the Proposed Intrusion Detection System. The Number Of State Considered Is 15 and Observations Created According to Extracted Features of Each Packet.

The proposed system uses 15 states to train and as a respect to the items listed in pre-processing step, the amount of training data observations and diagnoses will be variable. The Symbol Distribution matrix and mode transition matrix are randomly initialized in the first symbol. The built system consists of a large number of HMMs which vary according to security's sensitivity. We have used 20 HMMs, in which every HMM contains 10 million records, and each record includes 5 features. Made HMMs are shown in Figure 5.

In order to combine all outputs produced by each HMM for an input sequence, the sequence is called HMMi which is expressed as follows:

$$P(\text{Sequence} \mid Hmm_i) = P(Hmm_i \mid \text{Sequence})\, P(\text{Sequence}) / p(Hmm_i) \quad (5)$$

So we're in the training phase:

$$\text{Train} = \max \{P(\text{Sequence} \mid \text{Hmm}_i)\},\ i \in [1, N] \quad (6)$$

And in detection phase, we have:

$$\text{Detection} = \text{Average} \{P(\textit{Sequence} \mid Hmm_i)\},\ i \in [1, N] \quad (7)$$

Here, N is the symbol of Total HMM Within an ensemble. Therefore, using the Largest in the training phase and Median in the detection phase, the system can achieve the highest detection rate and number of HMM's errors can be omitted by detection or averaging. This is actually the main reason of using HMM group.

### *C. Making Fuzzy Rules*

Actions of Defuzzification rules are different in facing sensitivity and impact of each selected feature, hence it should be expected that every single input received by HMM's output, would make different decision. Thus, there are three sets low, medium and high for each extracted feature which shows the possibility amount of each HMM groups. In other word, fuzzy rules can be created by Fuzzification sets. If the following fuzzy rules are known as follows:

$$R = \Sigma_{i=1}^{n}\ R_i \quad (8)$$

And five selected features are named as follows:

$$F = \Sigma_{i=1}^{5}\ f_i \quad (9)$$

he membership degree of each feature in Fuzzification set is graphically expressed in Figure6:

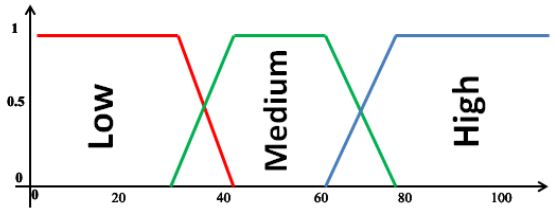
$$\mu_R(F_i),\ \mu_R(F_i) \in \{low, medium, high\} \quad (10)$$

For example, if appropriate probability for DST, Log, HST and srv achieved but the results of Src was inappropriate, the Packet is normal. As well as if a proper probability is obtained for DST and HST, but the obtained probability for srv, log and hast is improper, the Packet is abnormal. As shown in Table 6, an example of Fuzzification rules has been formed for each HMM output, can be expressed as follows:

Table 6. Defined Rules for Fuzzy Inference System

| | Rules |
|---|---|
| 1 | (F1& F2==low) AND (F3&F4&F5==high) => (detection= normal) |
| 2 | (F1==low) AND (F2&F4==medium) AND (F3&F5==high) =>detection=normal) |
| 3 | (F2&F5== high) AND (F1&F3&F4==low) => (detection= attack) |
| 4 | (F1==high) AND (F2&F3&F4== medium) AND (F5==low) => (detection= attack) |

Hence, according to the rules and membership functions built in Fuzzification stage, classical probability values are converted to Fuzzy input values which are usually fuzzy sets.




During Defuzzification stage, Fuzzy values will turn to needed real values which can be either normal or abnormal. In this task, the maximum Defuzzification is applied.

## V. Experimental Result

To evaluate the system the KDD Cup 1999 dataset was used [19]. The proposed system on a server with processors Cori7, 6 GB of RAM and 4 MB of cache memory is tested. The proposed intrusion detection system based on features extracted from the network traffic produced HMM ensemble which each HMM ensemble has a specific number of HMM. HMM training data are normalized to 200,000've used. Intrusion detection systems approaches in dealing with incoming packets into 4 categories are dived that shown in Table 7.

Table 7. Intrusion Detection System Approaches in Deal with the Malicious and Non-malicious Package.

| Alarm | Intrusion attack | No intrusion attack |
|---|---|---|
| Alarm sounded | TP(true positive) | FP(False positive) |
| No Alarm sounded | FN(False negative) | TN(true negative) |

Therefore, to obtain the values of the following would work [14]:

$$\text{False positive rate} = \frac{\text{normal packet which system detects an attack}}{\text{total number of normal}}$$

$$\text{rate}=\frac{\text{False negative}}{\text{attack packet that system detects as normal}}{\text{total number of attack}}$$

Detection rate =1 - false negative rate

Table 8. Evaluate the Proposed System.

| IDS | results | | | |
|---|---|---|---|---|
| Number of detection attacks | ipsweep | portsweep | neptune | smurf |
| | 10000 | 10000 | 50000 | 100000 |
| Number of normal packets, known attack | 620 | | | |
| Number of attack packets, known normal | 0 | | | |
| Detection rate | %100 | | | |
| False positive rate | %0.62 | | | |
| False negative rate | %0 | | | |

To evaluate the proposed system from 100,000 to 1,700,000 normal and abnormal data that are randomly extracted from the data set have been used. The results of the condition matrix appears in Table 8 shows that the proposed system has been successful in detecting attacks and was able to detect all attacks But has a low error in False negative.

The main idea of the proposed system is using HMM ensemble to extract the features of network traffic. Figure 6 shows the difference in the case of single HMM system with the use of all network traffic features and HMM ensemble with a 5 extracted features. As you can see in the case of system using HMM ensemble with five features have a very high precision As far as a difference between malicious data type Portsweep probe, Smurf dos, Ipsweep probe and Neptune dos can be seen with normal data.

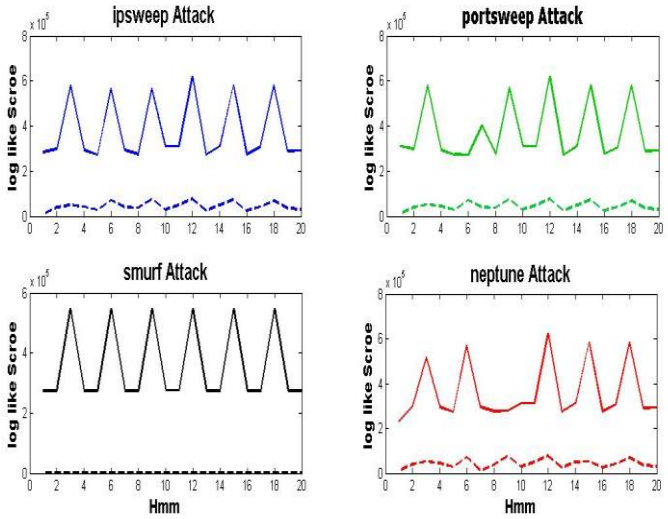



Fig.6. Evaluation Results of the Proposed System to Similarity of Portsweep Probe, Smurf Dos, Ipsweep Probe and Neptune Dos Attacks with Trained HMMs.

In the case of system used five feature extraction with single HMM or HMM ensemble, a constant state was obtained When use a single HMM and There are no difference between malicious and non-malicious, but when HMM ensemble used differences found between malicious data to the data stored in the database, Insofar as the proposed system to detect these attacks achieve to higher accuracy.

### *A. Making Fuzzy Rules*

The parameters evaluated the proposed system can accurately recall and F1-measure named. The evaluation of these parameters with regard to malicious class as the positive class is shown in Table 9 [7]. Precision measure indicates the probability that malicious packets by the system to correctly identify how much is malicious.

Recall measure is Criteria for measurement True positive predicted by the system, the percentage of attack data sets which correctly by the system as attack have been identified. Since both criteria in evaluating a machine learning method recall and Precision are important, the criterion F1-measure which combines the two criteria used. This criterion specifies how much the system predicts the correct answer and sustainable in terms of Precision has been successful. In calculating F1-measure method calculating Precision and recall are involved. Evaluation measures listed can be seen in Table 9. As you can see, the system has good Precision and F1-measure on a measure acceptable result is achieved.

Table 9. Evaluation of Precision, Recall and F1-Measure Proposed System

| F1-measure | | recall | | precision |
|---|---|---|---|---|
| negative | positive | negative | positive | |
| $\frac{TN}{TN+Fp}$ | $\frac{Tp}{Tp+FN}$ | $\frac{TN}{TN+FN}$ | $\frac{Tp}{Tp+Fp}$ | $\frac{2*\text{precision}*\text{recall}}{\text{precision}+\text{recall}}$ |

### *B. Fuzzy Inference*

In the Defuzzification stage, the fuzzy values are converted to real values are needed which can be either normal or abnormal. The maximum Defuzzification is used in proposed system. As you can see in Table 10, can be easily understand that as fuzzy rules, detection rate and false positive rate is in good condition.

Table 10. Results of Evaluation the Proposed System with Fuzzy Rules

| Fuzzy rule | Detection rate | False positive rate |
|---|---|---|
| 10 | %98.20 | %0.20 |
| 12 | %99.26 | %0.32 |
| 25 | %99.78 | %0.46 |
| 33 | %100 | %0.62 |

Although, by increasing fuzzy rules, system complexity increases and systems act slower but it must be a balance between the complexity and the false positive rate in order to bring the best [17]. Results to display ultimate performance system, the Area under the Curve are used. ROC curve Results to display a balance between detection rate and false-positive rates on a committee classification method used. An important characteristic area under the curve of the AUC is to say [18]. AUC value equal to 1 is a good classification. AUC for the proposed system is equal to 0.99 as shown in Figure 7.

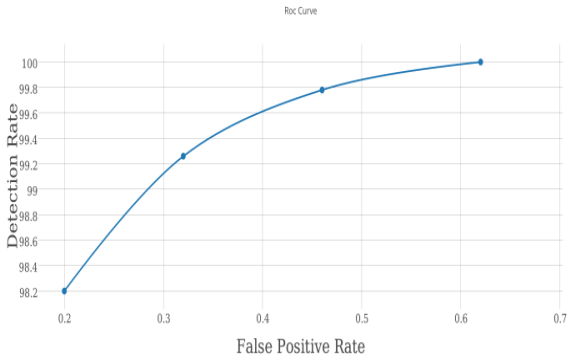



Fig.7. Graph AUC When the System is in Addition to the High Detection Rate, Low False Positive Rate has reached. Points Marked on the Graph Represent a Different Threshold Values with Continuously Varying Quantity, Attack Detection Rate and False Positive Rate of the System Also Increases

## VI. Conclusion and Future Work

In this paper, an intrusion-detection system was introduced to detect malicious attacks on network traffic that instead of using one HMM, multiple HMMs were used as an HMM ensemble to apply MCS technique for modeling extracted Features. On the other hand, proposed system has reached to the lowest false-positive rate by generating fuzzy sets and rules and applying fuzzy inference to decide about normality or being attack instead of using a threshold. By using these strategies, the proposed system will achieve to good performance in detecting new attacks, it behaves more flexibly with the packages that are main candidates error and decreases false-positive rate to an acceptable level. Experimental Result shows that proposed system with acceptable false-positive rate 0.62% and detection rate %100 has improved rather than an intrusion-detection system which has used the threshold for deciding. as well as The system detects attacks portsweep probe, smurf dos, ipsweep probe and neptune dos works pretty well and Evaluation Criteria recall, precision and F1-measure respectively By 100%, 99.38% and 99.69% will pass.

For future work, Machine learning systems can be designed however more accurate operate and Fuzzy inference system can be configured so that a set of fuzzy rules to be affected system performance and achieve better results.

## References

[1] Aljawarneh, S., Aldwairi, M. and Yassein, M.B., Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model. *Journal of Computational Science*, *25*, pp.152-160, 2018.

[2] Salehi, M. and Karimian, J., A Trust-based Security Approach in Hierarchical Wireless Sensor Networks. *Ad Hoc Netw*, *7*(6), pp.58-67, 2017.

[3] Viegas, E., Santin, A.O., Franca, A., Jasinski, R., Pedroni, V.A. and Oliveira, L.S., Towards an energy-efficient anomaly-based intrusion detection engine for embedded systems. *IEEE Transactions on Computers*, *66*(1), pp.163-177, 2017.

[4] Li, Q., Tan, Z., Jamdagni, A., Nanda, P., He, X. and Han, W., An Intrusion Detection System Based on Polynomial Feature Correlation Technique. *2017 IEEE Trustcom/BigDataSE/ICESS*, 2017.

[5] Yong zhong Li, Yang Ge, Xu Jing, and Zhao Bo, "A New Intrusion Detection Method Based on Fuzzy HMM," ICIEA, IEEE Conference on, 3rd, pp. 36-39, 2008.

[6] Ruchi Jain, Nasser S. Abouzakhar. "A Comparative Study of Hidden Markov Model and Support Vector Machine in Anomaly Intrusion Detection." Journal of Internet Technology and Secured Transactions (JITST), Volume 2, Issues 1/2/3/4, 2013.

[7] Annachhatre, C., Austin, T.H. and Stamp, M., Hidden Markov models for malware classification. *Journal of Computer Virology and Hacking Techniques*, *11*(2), pp.59-73, 2015.

[8] Cahyanto, T.A., 2015. BAUM-WELCH Algorithm Implementation For Knowing Data Characteristics Related Attacks On Web Server Log. *PROCEEDING IC-ITECHS 2014*.

[9] C. Kruegel, G. Vigna, and W. Robertson, "A multi-model approach to the detection of web-based attacks," Computer Networks, Vol. 48, Issue 5, pp. 717–738, 2005.

[10] Dau Xuan Hoang, and Minh Ngoc Nguyen, "A program-based anomaly intrusion detection scheme using multiple detection engines and fuzzy inference," Journal of Network and Computer Applications, Vol. 32, Issue 6, November 2009.

[11] Estevez Tapiador, Garcia Teodoro, and Diaz Verdejo, "Detection of Web-based Attacks through Markovian Protocol Parsing", 10th IEEE Symposium on Computers and Communications, pp. 457-462, 2005.

[12] R.O. Duda, P.E. Hart, and D.G. Stork, "Pattern Classification," Wiley, pp. 10-40, 2000.

[13] Ajith Abraham, Ravi Jain, "Soft Computing Models for Network Intrusion Detection Systems", Classification and Clustering for Knowledge Discovery Studies in Computational Intelligence, Vol. 4, pp. 191-207, 2005.

[14] Ji-yao An, G. Y.-f. Intrusion Detection Based on Fuzzy Neural Networks. In Z. Y.-L. Jun Wang, Advances in Neural Networks - ISNN 2006 (pp. 231-239). Berlin, Heidelberg: Springer Berlin Heidelberg. 2006

[15] J.E. Dickerson, J. Juslin, O. Koukousoula, and J.A.

Dickerson, "Fuzzy Intrusion Detection," IFSA World Congress and 20th NAFIPS International Conference on, Vol. 3, pp. 1506-1510, Vancouver, Canada, 2001.
[16] J. Gomez, F. Gonzalez, and D. Dasgupta, "An Immuno-Fuzzy Approach to Anomaly Detection," Fuzzy Systems, 12th IEEE International Conference on, Vol. 2, pp. 1219-1224, 2003.
[17] Bazara I. A. Barry, H. Anthony Chan. "Intrusion Detection Systems." In Handbook of Information and Communication Security, 193-205. Springer Berlin Heidelberg, 2010.
[18] Tang, T.A., Mhamdi, L., McLernon, D., Zaidi, S.A.R. and Ghogho, M., October. Deep learning approach for network intrusion detection in software defined networking. In *Wireless Networks and Mobile Communications (WINCOM), International Conference on* (pp. 258-263). IEEE, 2016.
[19] KDD Cup 1999 Data. 1999. http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

## Authors' Profiles

**Mohsen Salehi** received MSc Degree in Computer Engineering from Imam Reza International University, Iran, in 2014. He received the B.S. degree in computer engineering from Shahrood University. He has published several researches Paper in Computer science Field. His research interest includes Data Mining, Network, Security, image processing and artificial intelligence.

**Jamal Karimian** (born September 17, 1989) is an Iranian Software Engineer. He received his Master degree from Imam Reza University in Mashhad in 2014 and has been researching in various fields of computer science such as data mining, artificial intelligence and Network. He has a lower academic level He spent his time at the Montazeri Shahid University and Jahad Daneshgahi, and became one of his most distinguished students.

**How to cite this paper:** Mohsen Salehi, Jamal Karimian, Majid Vafaei Jahan,"A DOS and Network Probe Attack Detection based on HMM using Fuzzy Inference", International Journal of Computer Network and Information Security(IJCNIS), Vol.11, No.4, pp.35-42, 2019.DOI: 10.5815/ijcnis.2019.04.05