International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

COPY RIGHT

ELSEVIER SSRN

2019IJIEMR.Personal use of this material is permitted. Permission from IJIEMR must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. No Reprint should be done to this paper, all copy right is authenticated to Paper Authors

IJIEMR Transactions, online available on 20th Nov 2019. Link :http://www.ijiemr.org/downloads.php?vol=Volume-08&issue=ISSUE-11

Title **MEASURING THE PERFORMANCE OF THE SNORT INTRUSION DETECTION USING CLONAL SELECTION**

Volume 08, Issue 11, Pages: 87–96.

Paper Authors

GOGINENI MAMATHA, K.RAMESH

Chintalapudi Engineering College, Ponnur, A.P, India-522124

USE THIS BARCODE TO ACCESS YOUR ONLINE PAPER

To Secure Your Paper As Per UGC Guidelines We Are Providing A Electronic Bar Code

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

MEASURING THE PERFORMANCE OF THE SNORT INTRUSION DETECTION USING CLONAL SELECTION

[1]GOGINENI MAMATHA,[2]K.RAMESH
[1](M.Tech), Dept of CSE,Chintalapudi Engineering College, Ponnur, A.P, India-522124
[2] Asst.Professor, Dept of CSE, Chintalapudi Engineering College, Ponnur, A.P, India-522124.

Abstract
Network intrusion detection system (NIDS) monitors network traffic to detect an unauthorized activity in computer networks. The NIDS is classified according to detection technique into signature and anomaly based. Each of them has its own advantage and disadvantage. The signature-based is more effective in detecting known attacks but it is unable to detect new attacks. The anomaly-based is better in detecting new attacks but it may produce many false alarms. NIDS which use both of them try to exploit the strengths of them. In this paper we propose an improvement of the well known Snort NIDS using clonal selection algorithm (CSA). The proposed approach is evaluated using the 1999 DARPA Intrusion Detection Evaluation Data Sets of MIT (Massachusetts Institute of Technology) as a testbed. The conducted experiments compare the recall, precision, and F-score of Snort NIDS on its own, Snort NIDS improved by negative selection algorithm (NSA), and the proposed approach. Theobtained results show that the proposed approach is morepowerful than the others.
Keywords: Intrusion, NIDS, CSA, NSA

1. Introduction

Recent surveys and studies have shown that cyber-attacks have caused a lot of damage to organizations, governments, and individuals around the world. Although developments are constantly occurring in the computer security field, cyber-attacks still cause damage as they are developed and evolved by hackers. This research looked at some industrial challenges in the intrusion detection area. The research identified two main challenges; the first one is that signature-based intrusion detection systems such as SNORT lack the capability of detecting attacks with new signatures without human intervention. The other challenge is related to multi-stage attack detection; it has been found that signature-based is not efficient in this area. The novelty in this research is presented through developing methodologies tackling the mentioned challenges. The first challenge was handled by developing a multi-layer classification methodology. The first layer is based on decision tree, while the second layer is a hybrid module that uses two data mining techniques; neural network, and fuzzy logic. The second layer will try to detect new attacks in case the first one fails to detect. This system detects attacks with new signatures, and then updates the

SNORT signature holder automatically, without any human intervention. The obtained results have shown that a high detection rate has been obtained with attacks having new signatures. However, it has been found that the false positive rate needs to be lowered. The second challenge was approached by evaluating IP information using fuzzy logic. This approach looks at the identity of participants in the traffic, rather than the sequence and contents of the traffic. The results have shown that this approach can help in predicting attacks at very early stages in some scenarios. However, it has been found that combining this approach with a different approach that looks at the sequence and contents of the traffic, such as event- correlation, will achieve a better performance than each approach individually.

The illegal access through computer networks to obtain sensitive data information is increasing in spite of the presence of several security methods such as access control, data encryption and the use of a firewall. [1] As a result of growing popularity and fast growth of the Internet, many networks have been created for commercial, social media and governments. The technical team may be unable to administrate their network security because of the varied and complex skills required for this. So, the networks are vulnerable to be attacked and the concept of NIDS has been introduced.

The main rule of the NIDS systems is to protect networks from being attacked and to prevent or detect suspicious behavior produced by attackers. [2] Intrusion detection systems (IDS) classified according to detection technique to signature-based and anomaly-based intrusion detection. The signature-based IDS have signatures database of known attacks. It monitors the network activities. If any activity matches with existing signatures of the database, the intrusion detection system produces an attack alarm to report the network security administrator. [3] The anomaly-based IDS uses the network historical traffic to predict a profile of normal behavior and, if the current behavior differs significantly from the normal profile, it will be considered as an attack. [4]

2. Related Works

A well-known example of a commercial low interaction honeypot is Honeyd (Provos, 2003). Honeyd (Provos, 2003) is a daemon that can 13 used to simulate a large network on a single network host. It is a framework for creating virtual honeypots using unused IP addresses of a network, which simulates various operating systems and services. Other lowinteraction honeypot include Specter (Netsec, 2012) and KFSensor (KFSensor, 2012). Specter can monitor a total of 14 TCP ports and of these 14 monitored ports, 7 ports are called traps, and the other 7 are called services. Traps are port listeners: when the attacker makes a connection, the attempt is terminated, and then logged. Services are more advanced where there is interaction with the attacker, emulating the application (Netsec, 2012). The level of emulation depends on each service. For example, the HTTP service emulates a simple Web server with default static Web pages. Figure 2.1, shows the control panel for low-interaction honeypot tool – Specter and KFSensor simulates

system services at the application layer (Kuwatly et al., 2004). Reference (KFSensor, 2012) explains the methods in which KFSensor can be used to setup new firewall rules.

High interaction honeypots are complex solutions, which include deploying of a real operating systems and applications (Saini et al., 2011). As it involves real operating system, the level of risk is increased by many folds, but it is a trade-off in order to capture extensive amounts of information by allowing the attackers to interact with real systems (Singh and Joshi, 2011). This facilitates capturing / logging of full extent of attacker's behaviour that can be analysed at later stage. According to (Singh and Joshi, 2011), as the attacker has more resources to exploit at his disposal, a high interaction honeypot should be regularly monitored to ensure that it does not become a security issue. Example of high interaction honeypots include Honeynets (Project, 2012), Sebek (Huang et al., 2009), Argos (Portokalidis et al., 2006). Argos offer a full operating system to the attacker and when the attackers tries to 15 do something malicious the honeypot will shut down and makes dumps of memory and disk to get information about what the attacker was trying to do (Meijerink and Spellen, 2006).

Intrusion is described as an attack or attempt to sidestep security mechanisms of computer or networks, or compromising the confidentiality, integrity, or availability (CIA) (Bace and Mell, 2001). Weber (1998) categorized attacks into 5 classes: Denial-of-Service (DoS), Probing attacks, User to Root (U2R) attacks, Remote to Local (R2L) attacks, and Data attacks. In the first class, computing resources are overwhelmed by attackers in order to handle legitimate users' requests (Labib, Vemuri, 2008).

U2R attack is defined as an attempt by hackers exploiting weaknesses in the system in order to obtain root user privileges. R2L attack is an attempt from a remote machine to get unauthorized local access. In the last class, malicious code is injected in data looking normal that passes firewalls to attack and destroy systems. According to SANS Institute (2001), the intrusion detection process is involved in observing and analysing user and system activity, reviewing both system configurations and vulnerabilities, evaluating the stability of critical system and data file, reporting abnormal activities, and carrying out system audit. This process is carried out by a software application or hardware system. There are three components for intrusion detection systems; Network Intrusion Detection system, Network Node Intrusion detection system, and Host Intrusion Detection System. The first component (NIDS) is in charge of scanning traffic from and to all machines over the network (Bradley, 2014). The second component (NNIDS) is responsible for examining and analysing traffic directed from the network to a specific host (SANS, 2001). The last component (HIDS) checks incoming and outgoing packets from a host only and notifies users or administrator of suspicious activities (Bradley, 2014).

Intrusion detection methodologies are categorized into three types; signature based detection (SD), Anomaly-based detection (AD), and stateful protocol analysis (SPA). The first methodology (SD) defines a pattern

that matches a particular attack. This methodology is very effective to find known attacks or threats. However, it is not easy to keep patterns up to date. In addition to this, this methodology is not effective in detecting unknown threats or attacks (Liao et al. 2012). On the other hand, the second type (AD) is very effective in finding new vulnerabilities. AD works on the basis of defining the network behaviour (profile). Then, the defined profile is compared with monitored events and activities to detect significant attacks. The main disadvantage of this methodology is its high dependency on profile definition, not well-defined profiles can lead to weak accuracy in detecting attacks or threats (Jyothsna et al. 2011). The third category (SPA) works similarly to AD but with generic profiles defined by vendors. Those preloaded profiles are related to specific protocols. Therefore, the system will be able to find unexpected sequences of commands like issuing a command repeatedly (Scarfone and Mell, 2007). However, it will not be able to cease attacks behaving as benign protocol (Liao et al. 2012).

SNORT tool is a widely used source network intrusion prevention and detection system built by Source fire. It is an example of signature based IDS. Numerous researchers have evaluated the performance of this tool. One example of such a study carried out by Rani and Singh (2014) concluded that SNORT managed to find 12 signatures one of them is ICMP PING attack having the max numbers of alerts reported in figure 1.1.

by SNORT. Another study (Salah et al. 2011) carried out to evaluate the performance of SNORT when using Windows 7 and Windows 10.According to (Lazarevic et al., 2003, Zanero and Savaresi, 2004), anomaly detection rate and false alarm rate are the best measures that can be used for evaluation of IDSs. Clearly, the detection rate is equivalent to efficiency and the false alarm rate refers to effectiveness of IDSs. The anomaly detection rate is the ratio of number of detected intrusions to the total number of intrusions that were introduced into the network traffic as show in the equation below (Ertoz et al., 2004).

3. Proposed Architecture

The Proposed design IIDSS (Intelligent Intrusion Detection Security System has the accompanying key modules. They are, Preparation-Pre-handled information is accessible promptly for the execution, and Attack related information is gathered from KDD Cup dataset. Data around 23 distinct assaults is utilized for interruption identification. Pre-preparing Data pre-processing is finished utilizing WEKA instrument by utilizing RemoveUseless() capacity to kill excess traits, out of 41 qualities few characteristics can be dispensed with for the execution improvement in the framework. Arrangement Using Simple K Nearest Neighbors Classification, the kind of assault can be ordered adequately. Interruption location Module: The modules sent as appeared

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

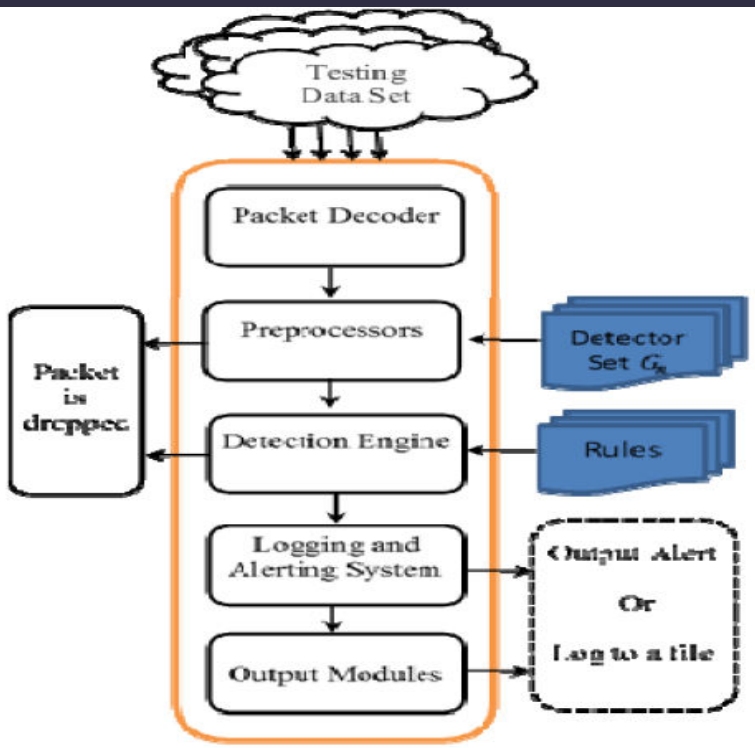

Figure 3.1: Proposed IIDS Architecture

3.1. Data Preparation

Input Dataset: KDDCUP99 dataset Output Dataset: Attacks data

Step 1: Generate an Array by using training records

Step 2: Choose model training records for data preparation, no_of_attacks, no_of_normal

Step 3: Produce training files Anomaly and Misuse

Step 4: Produce files with each attack type

Step 5: Generate String array Attacks

Step 6: Scan input training data records and generate a file list

Repeat Step 6 (for anomaly dataset)

Step 7: Read inputs data records and create a list

Repeat Step 7 (for misuse dataset)

Step 8: Generate output trained files list

The KDD CUP99 information with the aggregate of 42 qualities is utilized in this examination. The KDD has diverse kinds of information variants, in this 10% of preparing information has been utilized in the information planning stage and has an aggregate of 6222 examples. Each occurrence has been determined with included qualities of 42. In this paper, just 41 traits are isolated into 4 classes as referenced beneath.

• Basic features (B) is the component for TCP associations which has singularity.

• Content based (C) means space learning.

• Traffic based (T) are figured highlights of a two-second time window

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

• Host_based (H) highlights are utilized to survey the assaults from the most recent two seconds.

This period of information arrangement is structured and created to identify an abnormality and abuse based interruptions utilizing proposed strategy. Get ready datasets to choose the quantity of records arbitrarily for every single sort of assault. Data preprocessing is finished by utilizing "remove useless ()" work. Information Preparation steps are as per the following

- In the underlying advance, KDD container 10% information has been administered assault savvy and put away each record with a naming tradition Optimized_"name_of_the_attack".
- DataPreparation.java is utilized to separate records arbitrarily from every one of the documents named as Optimized_'name_of_attack' made in step1.
- Dataset Anomaly and Dataset Misuse are created in the wake of executing the "DataPreparation.java" code, which is later utilized for arrangement.

4. Feature Extraction

Feature Extraction is done using RFE (Recursive Feature Elimination) method and to reduce the feature set Random Forest approach is for minimizing the feature subset. The proposed feature selection process consists of four phases. Preprocessing, Result preprocessing, Exploring the result Analysis, Elimination and Prediction of Feature set

4.1. Preprocessing the Dataset

1. Load the classified dataset in to R environment, In our proposed system dataset is named as "Dataset_Misuse.csv"
2. Summarize the data, in order to identify number of rows and columns
3. Neglect or omit the rows (values) with "NA"
4. Read all the Features into a vector
5. Convert the data type of feature set values, by using coercion function in r

Ex: data duration = as. numeric (as character (data duration)), for duration attribute all the values are converted to numeric

6. For categorical data like protocol feature, use the following conversion

Ex: data$protocol_type = factor (data$protocol_type)

4.2. Grouping of Preprocessed Results

In this result preprocessing phase, The result is classified into 4 groups and Sub setting the result variable into 4 groups.Ex: DoS, Probe, U2R, R2L and Normal

- data$result[data$result == "ipsweep."] = "probe"
- data$result[data$result == "buffer_overflow."] = "u2r"
- data$result[data$result == "back."] = "dos"

4.3. Exploring the result Analysis

The preprocessed result is analyzed by using the following rules

1. For "dst_host_same_src_port_rate" value greater than equal to 1 it can be "probe" and "r2l"
2. For flag= "REG" and "S0" it is "dos"
3. For duration Greater than 30000 we can see it's 'probe'

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

4. For serror_rate and srv_serror_rate=0 or 1 its "dos" and
5. For serror_rate between 0.25 to 0.5 its "probe“
6. Flag value is a strong predictor for "dos" type intrusion

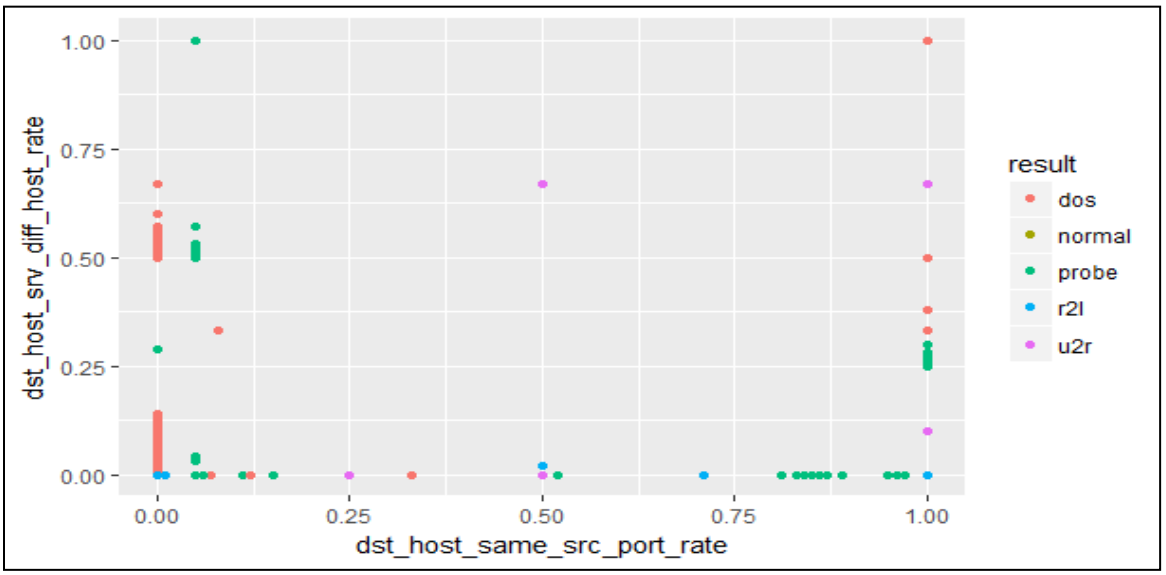

Figure 4.1: dst_host_srv_diff_host_rate Vs dst_home_same_src_port_rate

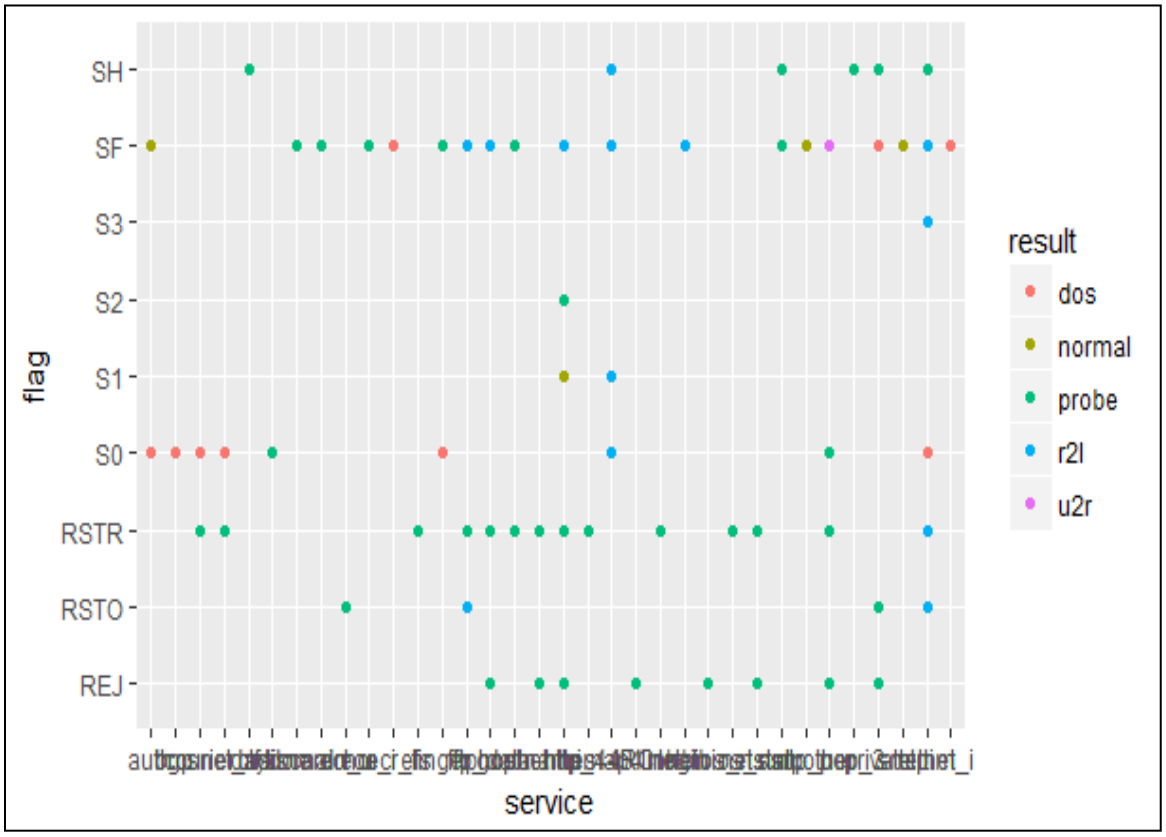

Figure 4.2: Service Vs flag

5. CONCLUSION & FUTURE WORK

The proposed Intrusion Detection display has indicated better outcomes contrasted and existing strategies. The procedure has indicated broad productivity in limiting the bogus alert rate and will diminish the manager remaining burden. This model has created 13.24% higher affectability when contrasted and C4.5, 10.55% contrasted and C4.5+ACO, and 2.95% when contrasted and EDADT. The exploratory outcome indicates better precision contrasted and the current framework. The Future work is meant to prepare the IDS to distinguish number of assaults, and the tally can be expanded from 23 to 40. The intrusion detection systems are very efficient for monitoring and detecting network traffic data packets. This research paper has proven that alerts are generated when there is a deviation in the behavioral patterns of the packets. The patterns are matched and compared with the proposed snort rules signature base. The proposed system was methodically tested and compared with existing snort rules, the proposed rules proved to be more accurate and efficient. In future work, advanced data mining techniques and machine learning techniques used for detecting new suspicious attacks on a huge amount of data.

6. REFERENCES

1. Abbes, T., Bouhoula, A., and Rusinowitch, M. (2004) Protocol analysis in intrusion detection using decision tree ITCC 2004. International Conference 1(April 2004), pp. 404-408 2. Abraham, T., IDDM: Intrusion detection using data mining techniques. 2001.

3. Alberto, P., Sala, A. and Olivares, M. “Fuzzy Logic Controllers. Methodology. Advantages and Drawbacks” [Online] Available from: http://www.softcomputing.es/estylf08/es/2000-X%20Congreso/01%20SESION%20INAUGURAL.pdf [Accessed: 19th April 2015]

4. Albin, E. (2011) A COMPARATIVE ANALYSIS OF THE SNORT AND SURICATA INTRUSION-DETECTION SYSTEMS, NAVAL POSTGRADUATE SCHOOL.

5. Allen, W. H. 2007. Mixing wheat with the chaff: Creating useful test data for IDS evaluation. IEEE Security & Privacy, 65-67.

6. Al-Mamory, S. and Zhang, H. (2008) New data mining technique to enhance IDS alarms quality. Journal in Computer Virology , 6(1), pp 43-55

7. Alserhani, F. et al. (2010): "Multi-Tier Evaluation of Network Intrusion Detection Systems" Journal for Information Assurance and Security (JIAS), 5 (4): 301-310.

8. Altwaijry, H., Bayesian based intrusion detection system, in IAENG Transactions on Engineering Technologies2013, Springer. p. 29-44.

9. Amiri, F., et al., Mutual information-based feature selection for intrusion detection systems. Journal of Network and Computer Applications, 2011. 34(4): p. 1184-1199 148

10. Amor, N., Benferhat, S. and Elouedi, Z. (2004) Proceeding SAC'04 Proceedings of the 2004 ACM symposium on Applied computing, PP. 420-424.

11. Amro, S. et al. 2012. Evolutionary Computation in Computer Security and

Forensics: An Overview. Computational Intelligence for Privacy and Security, 25-34.
12. Anthony, C. (2014) ARTIFICIAL NEURAL NETWORKS IN PROTEIN SECONDARY STRUCTURE PREDICTION: A CRITICAL REVIEW OF PRESENT AND FUTURE APPLICATIONS, Stanford University
13. Anthony, M. and Bartlett, P. "Neural Network Learning: Theoretical Foundations" 2009, Campridge University.
14. Antichi, G. et al. Enabling open-source high speed network monitoring. In Proceedings of NetFPGA IEEE 2012 Network Operations and Management Symposium, pp.1029-1035, April 2012
15. Axelsson, S. The base-rate fallacy and its implications for the difficulty of intrusion detection. in Proceedings of the 6th ACM Conference on Computer and Communications Security. 1999. ACM.
16. Bace, R. and Mell, P. (2001) Intrusion Detection System. National Institute of Standards and Technology, Available from http://www.cryptome.org/
17. Baecher, P. et al. The nepenthes platform: An efficient approach to collect malware. Recent Advances in Intrusion Detection, 2006. Springer, 165-184.
18. Barbara, D., et al., ADAM: a testbed for exploring the use of data mining in intrusion detection. ACM Sigmod Record, 2001. 30(4): p. 15-24.
19. Batal, I. (2014) Dimensionality Reduction techniques, University of Pittsburgh. Available From http://people.cs.pitt.edu/~iyad/ 149
20. Bellard, F. Qemu, a fast and portable dynamic translator. 2005. USENIX.
21. Bhargava, N., et al., Decision Tree Analysis on J48 Algorithm for Data Mining. International Journal, 2013. 3(6).
22. Bidgoli, B., Analoui, M., Rezvani, M. ,and Shahhosein, H. (2008) Performance Evaluation of Decision Tree for Intrusion Detection Using Reduced Feature Spaces. Iran University of Science and Technology
23. Borah, S. and A. Chakraborty, Towards the Development of an Efficient Intrusion Detection System. International Journal of Computer Applications, 2014. 90.
24. Bouzida, Y. and Cuppens, F. Neural networks vs. decision trees for intrusion detection. 25. Bradely, T. Introduction to Intrusion Detection Systems (IDS). About Technology, Available from http://netsecurity.about.com/
26. Bridges, S. and Vaughn, R. (2000). FUZZY DATA MINING AND GENETIC ALGORITHMS APPLIED TO INTRUSION DETECTION. Department of Computer Science Mississippi State University
27. Castro, L. and Zuben, F. (2002) Learning and Optimization Using the Clonal Selection Principle. IEEE Transactions on Evolutionary Computation, 6 (3), pp. 239–251
28. Chae, H. and Choi, S. (2014) Feature Selection for efficient Intrusion Detection using Attribute Ratio. INTERNATIONAL JOURNAL OF COMPUTERS AND COMMUNICATIONS, 8(2014)
29. Chang, R. et al., "Intrusion detection by backpropagation neural networks with sample-query and attribute-query," International Journal of Computational Intelligence Research, 2007, vol.3, n.1, pp. 6-10. 150

30. Chang, R.-I., et al., Intrusion detection by backpropagation neural networks with sample-query and attribute-query. International Journal of Computational Intelligence Research, 2007.