INTRUSION DETECTION SYSTEM BASED SNORT USING HIERARCHICAL CLUSTERING

Moch. Zen Samsono Hadi, Entin M. K., Aries Pratiarso, Ellysabeth J. C.
Telecommunication Department
Electronic Engineering Polytechnic Institute of Surabaya
Institut Teknologi Sepuluh Nopember (ITS) Surabaya
Telp : +62+031+5947280; Fax. +62+031+5946011
Email : zenhadi@eepis-its.edu

Abstract

One effort to protect the network from the threats of hackers, crackers and security experts is to build the Intrusion Detection System (IDS) on the network. The problem arises when new attacks emerge in a relatively fast, so a network administrator must create their own signature and keep updated on new types of attacks that appear.

In this paper, it will be made an Intelligence Intrusion Detection System (IIDS) where the Hierarchical Clustering algorithm as an artificial intelligence is used as pattern recognition and implemented on the Snort IDS. Hierarchical clustering applied to the training data to determine the number of desired clusters. Labeling cluster is then performed; there are three labels of cluster, namely Normal, High Risk and Critical. Centroid Linkage Method used for the test data of new attacks. Output system is used to update the Snort rule database.

This research is expected to help the Network Administrator to monitor and learn some new types of attacks. From the result, this system is already quite good to recognize certain types of attacks like exploit, buffer overflow, DoS and IP Spoofing. Accuracy performance of this system for the mentioned above type of attacks above is 90%.

Keywords: IDS, IIDS, SNORT, Hierarchical Clustering.

1. Introduction

No computer is one hundred percent safe in this world unless we buried 100 meters below the ground and turn it off. Oracle claims "impenetrable" but in just a couple of time has been compromised by hackers. One effort to protect the network from the threats of hackers, crackers and security experts is to build the Intrusion Detection System (IDS) on the network. Periodically IDS vendors will release a signature for new attacks and become a task of the Network Administrator to deploy it to the IDS of their network. The problem arises when new attacks emerge in a relatively fast, the network administrator cannot fully expect to IDS vendors to create a new signature in a short period, so that a network administrator must make their own signature and stay updated on the types of new attack.

Nowadays, workload of network administrators is extensively high, they did not always update new attacks and briefly create new signature for that attack. Then it comes the idea of how to create a new intrusion detection that can recognize a new pattern of attacks and automatically create a signature for the attack as well as add to an existing rule of IDS. This system became known as the Intelligence Intrusion Detection System (IIDS) which put an artificial intelligence (Artificial Intelligence) into the Intrusion Detection System (IDS).

Various machine learning techniques have been applied to the design of IDS, such as k-nearest neighbor [1][2] to improve the accuracy of the intrusion detection. And in this paper, Hierarchical Clustering is used in building an intrusion detection model based snort subsequently used for classifying network traffic either as normal or attack.

2. SNORT IDS

Snort is one example of NIDS program i.e. a program that can detect intrusions on a computer network. Snort is open source so that the software is free to be used to secure systems without having a license server.

Basic types of IDS are:

- Rule-based system: it is based on a database of signs of intrusion or attack has been known. If the IDS records the traffic in accordance with the existing database, then it is immediately categorized as an intrusion.
- Adaptive systems: it use more sophisticated methods. Not only it is based on existing databases but it also opens the possibility for the detection of a new form of infiltration.

The often used form for network security is rule-based system. The used approach in a rule based system are two, namely preemptory and reactionary. The difference is only in time. In the preemptory approach intrusion-detection system will pay attention to all the network traffic. If a suspicious package was found, then the system will perform the necessary actions. In the reactionary approach, intrusion detection system only observe the log file. If a suspicious package was found, the system also will perform the necessary actions.

Snort can be operated in three modes that is:

1. Sniffer mode, to see the package through the network.
2. Logger mode, to record all packets on the network for later analysis.
3. Intrusion Detection Mode, in this mode snort will have function to detect the attacks over computer networks. To use the IDS mode is required setup from different files or rules that would distinguish a normal and attack package.

Snort logger mode generally uses file to write its log, and it also provides a log into the database mysql, freebsd and oracle. With supporting of the database, it will make easy the process of reading the log data by the program because the database is very well known and understood. To make logging into the database, it needs additional settings so that Snort will automatically log into the database.

a. Hierarchical Clustering

Hierarchical clustering is a cluster analysis method that aims to build a cluster hierarchy. In hierarchical clustering, each data must include into a specific cluster, and a data in a process step, it can not move to another cluster at a later stage. Hierarchical clustering algorithms can be done with the following steps:

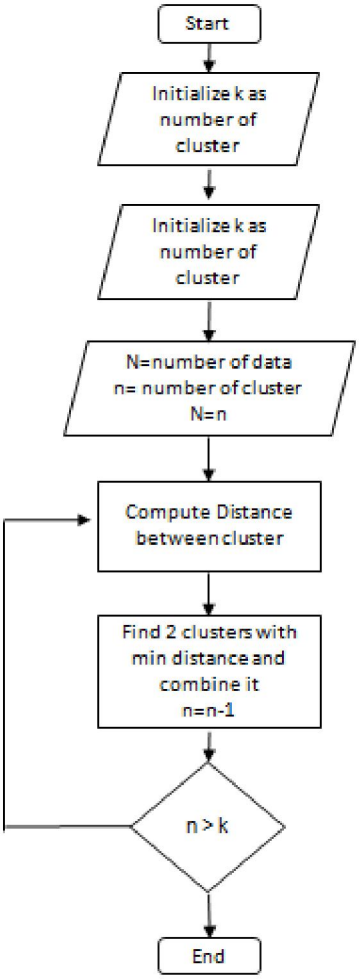

Fig. 1. Flowchart of Hierarchical Clustering

1. Define k as the number of clusters to be formed.
2. Each data is considered as a cluster.
   If N = number of data and n = number of clusters, there is n = N.
3. Calculate the distance between clusters.
4. Find the two clusters that have a distance between the cluster and combine the most minimal (mean n = n-1).
5. If n> k, go back to step 3.

b. **Compute Distance using Centroid Linkage**

Input of centroid linkage algorithm is the distance between clusters. Groups are formed by entities of cluster with combining the midpoint distance between the clusters. To determine the distance between the clusters is by calculating the midpoint of each cluster i.e. the average value data of each cluster. The distance between clusters is calculated by calculating the distance (Euclidian distance) between the midpoint of each cluster.

In this research, centroid linkage method was used as a method for testing new data. Training data that has been established in several clusters are sought its centroid point by finding the average value. In this case, the average value of each parameter is used in this project. The new data are calculated the distance to each centroid point using Euclidian.

Centroid linkage method is used because of the best and easy method for such case in this research.

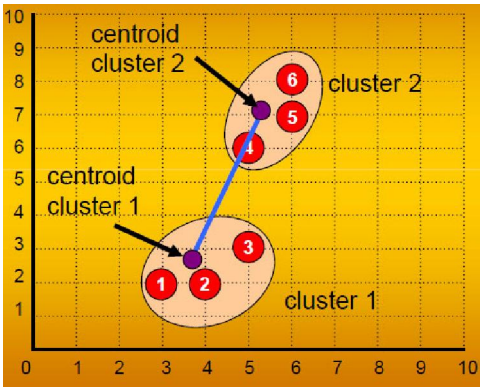

Fig. 2. Centroid Linkage

c. BASE (Base Analysis And Security Engine)

BASE can search and process a database containing the security events recorded by a variety of network monitoring tools such as firewalls and IDS programs. BASE is written in the PHP programming language and displays information from the database user interface. When it is used with Snort, it read both kinds of log formats and the tcpdump binary format as well as the Snort alerts. After the data is entered and processed, BASE has the capability to display information packet as graphical. The query from database is based on information alerts such as sensors, log alerts, signature, classification, and detection time, as well as data packets such as source / destination address, port, packet payload, or flags package. BASE also supports other databases and it can display information through any web server that supports PHP.

3. SYSTEM DESIGN

Data captured by Snort will be stored on Snort log file, then it will be filtered any selected parameters and inserted into a database. In the next step, it will be analyzed and clustered on the pre-processor with hierarchical clustering. The results will be displayed on the user interface using PHP and Ajax in real-time web-based, the output from the web interface will be constantly updating Snort rules if there are new attacks.

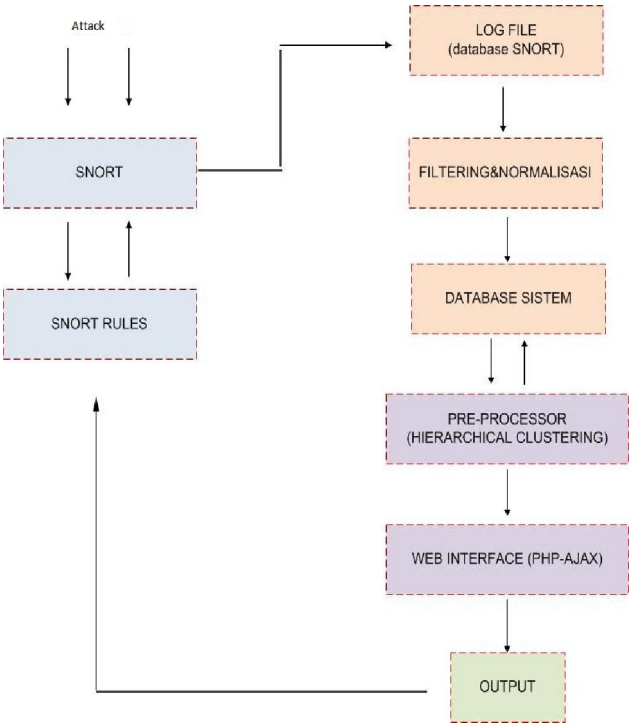

Fig. 3. System Design

4. EXPERIMENTS AND ANALYSIS

In this chapter, it will discuss the experiments and analysis of Hierarchical Intrusion Detection Engine (HIDE) for hacking activities as follows:

1. Port Scanning
2. Vulnerable Nessus Scanning
3. Exploit
4. Buffer Over Flow
5. Ping of Death (DoS)
6. Land Attack and SYN Flooding (IP spoofing)
7. Normal access (HTTP, FTP, Telnet, SSH)

The purpose of the above experiment is to find out how good HIDE detect network packets as normal or attack data. In this research, system experiments are performed by the host acting as an attacker by using several

types of attacks against multiple hosts that act as targets of attack, and then observed and analyzed the accuracy of the system in recognizing this type of attack. That will be classified as normal categories, high risk and critical to use varying the amount of training data i.e. 30, 60, 90, 120 and 257 training data in the data clustering. Analysis of system reliability will be compared with systems using fuzzy algorithms in FIRE (Fuzzy Intrusion Recognition Engine).

Table 1. Experiment with TCP Connect Scan

Training Data	Normal (%)	High Risk (%)	Critical (%)
30	50	50	0
60	100	0	0
90	0	100	0
120	84	0	16
257	25	0	75

The above result is for TCP Scanning. The best of training data number for the type of scanning is 60, and the second is 257. But for the overall results, the best average data training to detect various types of attack is 257.

The results of HIDE system with 257 training data are compared with the FIRE (Fuzzy Intrusion Recognition Engine) that uses an Fuzzy algorithm with 30 Fuzzy rules. Below are the results of comparison:

Table 2. HIDE vs FIRE

	Data Packet	HIDE (%)			FIRE(%)		
		N	H	C	N	H	C
Scanning	TCP Connect Scan	25	0	75	0	25	75
	Windows Scan	33	0	67	0	33	67
	FIN Scan	100	0	0	87	13	0
	XMAS Scan	0	0	100	0	95	5
	NULL Scan	100	0	0	80	20	0
	Average	51,6	0	48,4	33,4	37,2	29,4
Another Attacks	Nessus	22	0	78	24	1	75
	Exploit	86	0	14	86	0	14
	Buffer Over Flow	60	0	40	60	1	40
	Ping of Death	100	0	0	100	0	0
	Land Attack	99	1	0	99	0	1
	SYN Flooding	0	10	90	0	0	100
	Average	61,1	1,8	37	61,5	0,3	38,3
Normal Packet	HTTP	100	0	0	100	0	0
	FTP	100	0	0	100	0	0
	Telnet	100	0	0	100	0	0
	SSH	100	0	0	100	0	0
	Average	100	0	0	100	0	0

Note:
N = Normal Packet
H = High Risk Packet
C = Critical Packet

From the above table, FIRE system is better than HIDE system when it is used for scanning (HIDE recognize scanning packet as attack as much as 48.4%, but FIRE recognize it as much as 66.6%). If it is used to detect another attack, HIDE system is a little bit better (38.8%) than FIRE system (38.6%). For normal packet, both of methods can recognize 100% as normal data.

The following table describes the HIDE system performance and the results of Snort alerts in recognizing the type of attack in the Network Packet.

Table 3. HIDE vs SNORT alert

	Data Packet	HIDE (%)			SNORT (%)	
		Normal	H	C	Normal	Attack Packet
			Attack Packet			
Scanning	TCP Connect Scan	25	0	75	91	1
	Windows Scan	33	0	67	70	30
	FIN Scan	100	0	0	20	80
	XMAS Scan	0	0	100	20	80
	NULL Scan	100	0	0	40	60
	Average	51,6	0	48,4	48,2	62,75
Another Attack	Nessus	22	0	78	24	76
	Exploit	86	0	14	6	94
	Buffer Over Flow	60	0	40	60	40
	Ping of Death	100	0	0	80	20
	Land Attack	99	1	0	50	50
	SYN Flooding	0	10	90	55	45
	Average	61,1	1,8	37	45,8	54,1
Normal Packet	HTTP	100	0	0	100	0
	FTP	100	0	0	100	0
	Telnet	100	0	0	100	0
	SSH	100	0	0	100	0
	Average	100	0	0	100	0

From the above table,

From the above table, average of snort system is better than average of HIDE system for scanning and another attack experiments. But for experiments i.e. Scanning (TCP Connect Scan, Windows Scan, XMAS Scan), Another Attack (Nessus and SYN Flooding), HIDE system is better than snort system. For normal packet, both of methods can recognize 100% as normal data.

5. CONCLUSIONS

From the experiment results of Hierarchical clustering System (Hierarchical Intrusion Detection Engine/HIDE) in recognizing network packet, it can be made the following conclusion:

1. Hierarchical clustering system is able to distinguish the type of network attack packet in High Risk and Critical Network data.
2. The best training data for hierarchical clustering is 275.
3. If HIDE (Hierarchical Clustering) system is compared with FIRE (Fuzzy), performance of HIDE system is better when it detects DoS attack, Buffer Overflow and IP Spoofing.
4. If HIDE system is compared with snort system, the overall results, snort is better. But for experiments i.e. Scanning (TCP Connect Scan, Windows Scan, XMAS Scan), DoS (SYN Flooding), HIDE system is better than snort system.
5. Hierarchical clustering is quite good in recognizing network packet for certain types of attacks, like Exploit, Buffer Over Flow, DoS and IP Spoofing. Accuracy of HIDE system performance for the type of attack is 90%.

6. REFERENCES

[1] Adebayo O. Adetunmbi, Samuel O. Falaki, Olumide S. Adwale, Boniface K. Alese, "Network Intrusion Detection Based on Rough Set and K-Nearest Neighbour", International Journal of Computing and ICT Research, Vol. 2 No. 1, June 2008.

[2] Yihua Liao, V. Rao Vemuri, "Using K-Nearest Neighbor Classifier for Intrusion Detection", Department of Computer Science. University of California, Davis, CA, wwwcsif.cs.ucdavis.edu/~vemuri/papers/knn-ss02.pdf

[3] Bambang , Wijanarko and Entin , Martiana and Idris , Winarno (2009) "*Algoritma Fuzzy Sebagai Metode Pendeteksi Pola Serangan Pada Jaringan Berbasis SNORT IDS",* EEPIS Final Project., http://repo.eepis-its.edu

[4] Charlie Scott, Paul Wolfe, Bert Hayes, "SNORT for Dummies", Wiley Publishing, Inc, 2006.

[5] Toby Kohlenberg, "SNORT IDS and IPS Toolkit", Syngress, September 2007.