Secure Gateway Defender- a Network Intrusion Detection System

Vikram Kothari
I.T. Department
Sardar Patel Institute of Technology
Mumbai, India
vikramkothari6495@yahoo.com

Manali Raut
I.T. Department
Sardar Patel Institute of Technology
Mumbai, India
rautmanali04@gmail.com

Sairaj Samant
I.T. Department
Sardar Patel Institute of Technology
Mumbai, India
sairajsamant0795@gmail.com

Prof. Renuka Pawar
I.T. Department
Sardar Patel Institute of Technology
Mumbai,India
renuka_pawar@spit.ac.in

Abstract—A secure defender strives to meet accuracy and active responses. Traditional approaches towards network security largely rely on the statistical and algorithm-based designs for detecting intrusion, but have largely left response up to the network operator. We will build a NIDS which will use Apriori algorithm. The basic idea of the algorithm is to find all frequency sets in first, the frequent item sets occur at least a predefined minimum support and confidence. The system permits an early classification of the data packets by making them pass through a central server where they are processed.This early classification improves the performance. This paper focuses on building a system for network administrators to securely manage the network along with providing systems for logging and intrusion detection.

Keywords—Apriori algorithm; Network Intrusion Detection System(NIDS); support; confidence

I. Introduction

With the increase in number of Internet access methods and tremendous growth of the usage of computers over network captures the attention towards network security. This epitome exploits the security on all computer systems, which makes the task hard, and therefore attacks can be identified only after it happens.

Several actions attempt to bypass security mechanisms of computer systems. To overcome this situation, frequent updating of patterns is needed.Thus securing internal resources becomes critical. Adding to it, put forth a strategy to designing a secure Local Area Network to intellectually differentiate both intrusive and non intrusive records.

A network attached to the Internet can be secured with a firewall, however the software needs to be carefully configured, failure in doing so leads to the failure of network security. The intrusion detection system inspects incoming, outgoing, and local traffic. NIDS analysis the packets on the network as they pass by filter. A continuous, real-time analysis is performed by acquiring information about the actions immediately after they happen.

Naive approaches to the network security do not consider threats that are recent and therefore fail to provide protection against latest malware attacks. Considering the above failure of traditional approaches we have taken into account the rapid growth in the nature of attacks and therefore we are using an algorithm that takes care of the frequent analysis of the packets entering the network.

II. Literature Survey

Certain proposals have been made that can efficiently identify the data of user interest and also predicts the results that can be utilized in the future [1]. Another proposal describes a Knowledge Discovery and Data Mining (KDD) such that each and every instance in the dataset have several features or attributes necessary for mining for frequent pattern analysis [2]. KDD Cup 99 data set contains several features to determine the attack such as attributes of TCP/IP connection, same host feature and same service feature[8]. It converts data into a feature space that usually has a large dimension. It involves converting the data from monitored system (computer network, host machine) into data parameters that will be used in data mining models. The data set highly affects the performance of evaluated systems, and results in a very poor evaluation of detection approaches[10].

A proposal states that the procedures of applying Apriori algorithm is divided into two steps. The first step finds out iteratively all frequent itemsets. The second step establishes association rules that satisfy the threshold values[11].The mining process is an integral and continuous part of an intrusion detection system because the rule sets used by the detection module may not be static over a long period of time[12]. E. Saboori, S. Parsazad, and Y. Sanatkhani. stated that association rules provide information in the form of if then statements. An association rules has two numbers which refer to antecedent and consequent. Antecedent refers to the ”if part of rule and consequent refers to the ”then” part of rule. The first number is called the support for the rule. The support is simply the number of transactions that include all items in the antecedent and consequent parts of the rule[15].

Our proposed system uses the protocol, IP address and port number features of the packet. Nithya and Jayakumar put forth an idea of counting the frequently occurring items and the generation of automatic rules[3]. Also a proposal states that data mining techniques can be implemented in detecting the attacks [5]. With the maliciousness of a flow lies the probability of the packet being from an attacker [4]. The features among the rules to include or exclude a specific

978-1-5090-4890-8/17/$31.00 ©2017 IEEE

pattern of packet also can be set by the network administrator [7]. It is formulated that a model must begin by analysing the data provided, to look for a particular trends and patterns. Further, the results of analysis can be applied for defining optimal parameters. The parameters are applied across the dataset [6]. The main tasks of NIDS is to help distinguish between malevolent and innocent intrusions. It involves event monitoring and generating responses to anomalous behaviour[9]. KDD Cup 99 data set contains several features to determine the attack such as attributes of TCP/IP connection, same host feature and same service feature[8].

III. System Overview

A. Architecture

The system comprises of Internet, NIDS (acts as a secure defender) and LAN. The proposed system is placed at the entry point of the LAN to monitor traffic to and from all devices on the network. The basic intrusion detection model consist of five modules such as Capture Module, Decode Module, Detection Module, Known Attack Pattern Module and Action Module[13]. The packets that come from the Internet into the LAN are first traced and captured by the NIDS packet tracer. These packets are then stored in the buffer for further analysis and processing[4]. The stored packets are then processed by the packet decoder i.e. separating protocol(e.g. TCP, UDP) and payload (the message data). Further, the intrusion detection algorithm is applied on the processed packets to filter the incoming traffic and allow legitimate traffic to enter the LAN[7].

1) *Packet Tracing:* The packets that arrive from the Internet first go through the NIDS application and are captured in this phase.

2) *Packet storage:* Inside the system the packets are captured and stored in the buffer, as the packet arrival time is not equal to the processing time.

3) *Packet decoder:* Then the stored packets are burst, thereby separating the protocol and payload of the packet.

4) *Detection and prevention:* The count of the packets with respect to their protocol, IP address and port is maintained in a global list. The criteria such as support which is a counter, can be set as the blocking mechanism[5]. The initial packets may be tagged as safe packets.

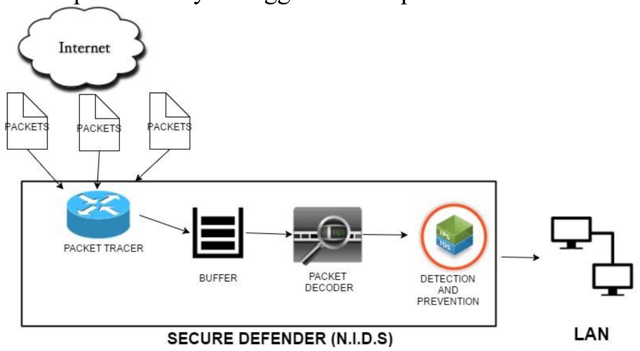


Fig. 1. architecture diagram

B. System flow

We will track all the Internet access points, and thus the request from them need to be traced[12].The packets that arrive from the Internet first go through the NIDS application. A continuous arrival data in multiple, rapid, time-varying, possibly unpredictable and unbounded streams. Inside the intrusion detection system the packets are captured and stored in the buffer as the packet arrival time is not equal to the processing time. Then the stored packets are burst ; thereby separating the protocol and payload of the packet. Further using the detection algorithm i.e Apriori algorithm , the intrusion is detected and prevented. The algorithm uses a threshold counter to block the attack. A threshold is a value that sets the limit between normal and abnormal behavior [14]. If number of requests is greater than threshold counter , the attack gets blocked at a particular port,protocol or IP address.

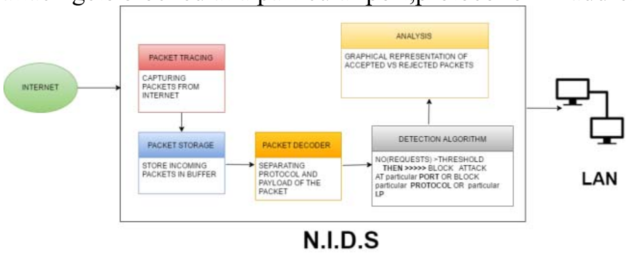

INTERNET
PACKET TRACING
CAPTURING PACKETS FROM INTERNET
PACKET STORAGE
STORE INCOMING PACKETS IN BUFFER
PACKET DECODER
SEPARATING PROTOCOL AND PAYLOAD OF THE PACKET
DETECTION ALGORITHM
NO(REQUESTS) >THRESHOLD THEN >>>>> BLOCK ATTACK AT particular PORT OR BLOCK particular PROTOCOL OR particular I.P
ANALYSIS
GRAPHICAL REPRESENTATION OF ACCEPTED VS REJECTED PACKETS
LAN
N.I.D.S

Fig. 2. system flow diagram

C. Sequence diagram

A LAN user sends request to the Internet. The packets arriving from the net are firstly traced by the NIDS , then stored in the buffer. Secondly, they are decoded by separating protocol and payload and intrusion detection algorithm is applied to the processed packets i.e . Apriori algorithm. It generates an alert signal to the computer user or network administrator for hostile activity on the opening session by inspecting hazardous network activities. Finally, generate rules for blocking the harmful packets and allow pure traffic to be directed to the LAN user.

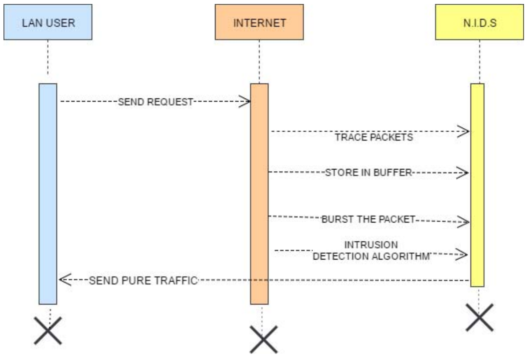


Fig. 3. sequence diagram

D. Implementation

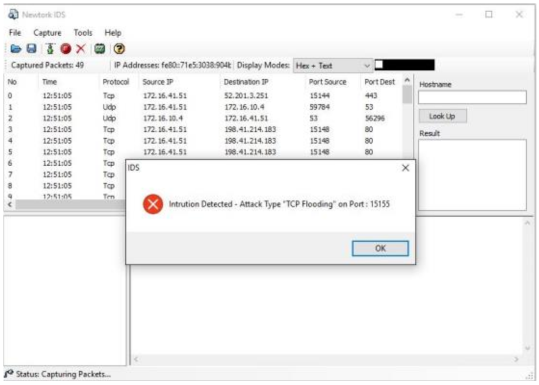
We are using Apriori algorithm to analyse the packets and prevent the attacks on the networks like Ping of death,TCP flooding, UDP flooding and ICMP flooding. Every packet is handled individually. The packets entering the system are traced, stored in the buffer(as the arrival rate is greater than

978-1-5090-4890-8/17/$31.00 ©2017 IEEE

processing speed) and decoded(separating the payload and the protocol) before applying the mining process (data mining algorithm i.e. Apriori algorithm). The attacks are prevented in the following manner

1) *Ping of death:* Initially the source address of the incoming packet is checked, if it is not the same as the user address, then the intrusion prevention process begins. Further the time stamp of the incoming packets is checked; if the time interval between the incoming packets is less than two minutes then the packets are added in the packet filter and the count is incremented for subsequent pings in the list. Finally, if the number of requests or pings exceeds the threshold value i.e. 5 pings or 20 packets in our case, a pop up will be displayed indicating Ping of death attack.

2) *Flooding attacks(IGMP,TCP and UDP):* Attacks like TCP and UDP flooding can be triggered using multiple IP addresses, therefore the prevention is not restricted to a particular IP address in figure 4. We consider the threshold as the number of packets of a particular protocol i.e. TCP,UDP. In our case we have considered 40 packets as threshold.



Captured Packets: 49 | IP Addresses: fe80::71e5:3038:904t | Display Modes: Hex + Text [redacted]

No	Time	Protocol	Source IP	Destination IP	Port Source	Port Dest
0	12:51:05	Tcp	172.16.41.51	52.201.3.251	15144	443
1	12:51:05	Udp	172.16.41.51	172.16.10.4	59784	53
2	12:51:05	Udp	172.16.10.4	172.16.41.51	53	56296
3	12:51:05	Tcp	172.16.41.51	198.41.214.183	15148	80
4	12:51:05	Tcp	172.16.41.51	198.41.214.183	15148	80
5	12:51:05	Tcp	172.16.41.51	198.41.214.183	15148	80
6	12:51:05	Tcp				
7	12:51:05	Tcp				
8	12:51:05	Tcp				
9	12:51:05	Tcp				

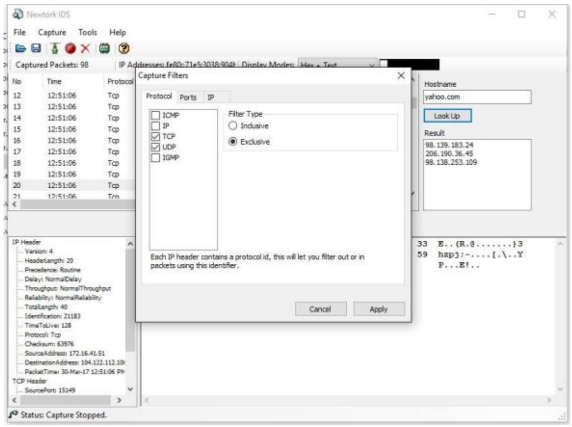
IDS

Intrution Detected - Attack Type "TCP Flooding" on Port : 15155

Status: Capturing Packets...

Fig. 4. alert generated to display TCP flooding

3) *Prevention:* In the prevention step, a list of packets is created in detection step which are to be blocked. The rules are automatically generated which will be executed on the next incoming packet. The packet is filtered and then allowed to enter the LAN if it satisfies the rules. The user can manually enter the rules to include or exclude the rules.At any instance, each rule inspects only one packet.



Captured Packets: 98 | IP Addresses: fe80::71e5:3038:904t | Display Modes: Hex + Text [redacted]

No	Time	Protocol
12	12:51:06	Tcp
13	12:51:06	Tcp
14	12:51:06	Tcp
15	12:51:06	Tcp
16	12:51:06	Tcp
17	12:51:06	Tcp
18	12:51:06	Tcp
19	12:51:06	Tcp
20	12:51:06	Tcp
21	12:51:06	Tcp

Capture Filters

☐ ICMP
☐ IP
☑ TCP
☑ UDP
☐ IGMP

Filter Type
○ Inclusive
● Exclusive

Each IP header contains a protocol id, this will let you filter out or in packets using this identifier.

Hostname: yahoo.com

Result:
98.139.183.24
206.190.36.45
98.138.253.109

```
IP Header
  Version: 4
  HeaderLength: 20
  Precedence: Routine
  Delay: NormalDelay
  Throughput: NormalThroughput
  Reliability: NormalReliability
  TotalLength: 40
  Identification: 21183
  TimeToLive: 128
  Protocol: Tcp
  Checksum: 63976
  SourceAddress: 172.16.41.51
  DestinationAddress: 104.122.112.10
  PacketTime: 30-Mar-17 12:51:06 PM
TCP Header
  SourcePort: 15149
```

```
33   E..(R.@.......)3
59   hzpj;-....[.\..Y
     P...E!..
```

Status: Capture Stopped.

Fig. 5. Rule generator to include or exclude rules

IV. OPERATIONS

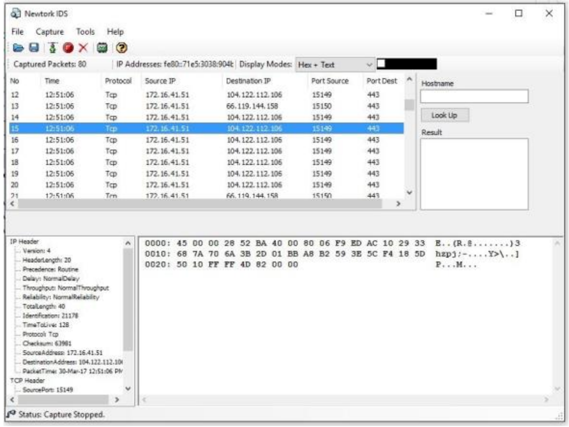
This section deals with some of the functions and operations of the software. They are discussed in depth below.

A. Starting the packet capture

This function allows the software to start capturing the packets arriving from the Internet.

B. Stopping the packet capture

This functions stops capturing the packets displayed in figure 6.



Captured Packets: 80 | IP Addresses: fe80::71e5:3038:904t | Display Modes: Hex + Text [redacted]

No	Time	Protocol	Source IP	Destination IP	Port Source	Port Dest
12	12:51:06	Tcp	172.16.41.51	104.122.112.106	15149	443
13	12:51:06	Tcp	172.16.41.51	66.119.144.158	15150	443
14	12:51:06	Tcp	172.16.41.51	104.122.112.106	15149	443
15	12:51:06	Tcp	172.16.41.51	104.122.112.106	15149	443
16	12:51:06	Tcp	172.16.41.51	104.122.112.106	15149	443
17	12:51:06	Tcp	172.16.41.51	104.122.112.106	15149	443
18	12:51:06	Tcp	172.16.41.51	104.122.112.106	15149	443
19	12:51:06	Tcp	172.16.41.51	104.122.112.106	15149	443
20	12:51:06	Tcp	172.16.41.51	104.122.112.106	15149	443
21	12:51:06	Tcp	172.16.41.51	66.119.144.158	15150	443

```
IP Header
  Version: 4
  HeaderLength: 20
  Precedence: Routine
  Delay: NormalDelay
  Throughput: NormalThroughput
  Reliability: NormalReliability
  TotalLength: 40
  Identification: 21178
  TimeToLive: 128
  Protocol: Tcp
  Checksum: 63981
  SourceAddress: 172.16.41.51
  DestinationAddress: 104.122.112.10
  PacketTime: 30-Mar-17 12:51:06 PM
TCP Header
  SourcePort: 15149
```

```
0000: 45 00 00 28 52 BA 40 00 80 06 F9 ED AC 10 29 33   E..(R.@.......)3
0010: 68 7A 70 6A 3B 2D 01 BB A8 B2 59 3E 5C F4 18 5D   hzpj;-....Y>\..]
0020: 50 10 FF FF 4D 82 00 00                           P...M...
```

Status: Capture Stopped.

Fig. 6. Stop packet capture

C. Saving the packet dump

This operation is used to store the packet dump in binary or text format in repository. Captured packet data can be reviewed via interface and in tcpdump format[13].

D. Binary Converter

It classifies the bytes as int, short and long and then convert the hexadecimal values into bytes.

E. Displaying packets content

The packet is split dynamically as header and other options, where the header comprises of key value pairs of source address, destination address, port numbers of source and destination and time stamp. Different protocol has different payload.

F. Adapter settings

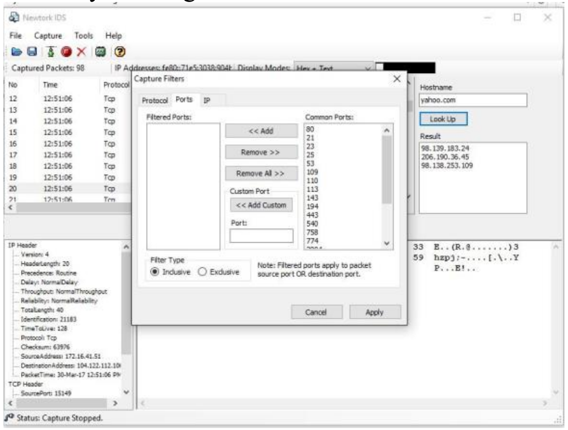
Version of the network adapter gives us the assembly company, copyrights, download speed in kbps, configuration information about the file name with path and several details. Using socket information, the local IP address can be found.

G. Filters

This is the operation having major importance in the software. The filters that are applied to block or accept the packets are included in the operation. A packet triggers the first rule that matches and does not examine the remaining rules.

1) Protocol Filters: Each IP header contains a protocol identifier. This filter uses a particular protocol i.e. TCP , UDP , ICMP etc to block the network traffic. A threshold value is used for the same which maintains the system in safe state. A protocol to be blocked can be added from the list provided. Also multiple protocols can be blocked.

2) Port Filters: This filter uses a particular port number in order to block flooding attacks on a specific port. Also the software allows the network administrator to manually add or remove ports to filter the packets. Filtered ports apply to packet source port or destination port. The port is added by manually entering the value.



Captured Packets: 98 | IP Addresses: fe80::71e5:3038:904b | Display Modes: Hex + Text [redacted]

No	Time	Protocol
12	12:51:06	Tcp
13	12:51:06	Tcp
14	12:51:06	Tcp
15	12:51:06	Tcp
16	12:51:06	Tcp
17	12:51:06	Tcp
18	12:51:06	Tcp
19	12:51:06	Tcp
20	12:51:06	Tcp

Capture Filters — Protocol | Ports | IP

Filtered Ports:

Common Ports: 80 21 23 25 53 109 110 113 143 194 443 540 758 774

Filter Type: ● Inclusive ○ Exclusive

Note: Filtered ports apply to packet source port OR destination port.

Hostname: yahoo.com

Result:
98.139.183.24
206.190.36.45
98.138.253.109

```
IP Header
  Version: 4
  HeaderLength: 20
  Precedence: Routine
  Delay: NormalDelay
  Throughput: NormalThroughput
  Reliability: NormalReliability
  TotalLength: 40
  Identification: 21183
  TimeToLive: 128
  Protocol: Tcp
  Checksum: 63976
  SourceAddress: 172.16.41.51
  DestinationAddress: 104.122.112.10
  PacketTime: 30-Mar-17 12:51:06 PM
TCP Header
  SourcePort: 15149
```

```
33  E..(R.@.......)3
59  hzpj:-....[.\..Y
    P...E!..
```

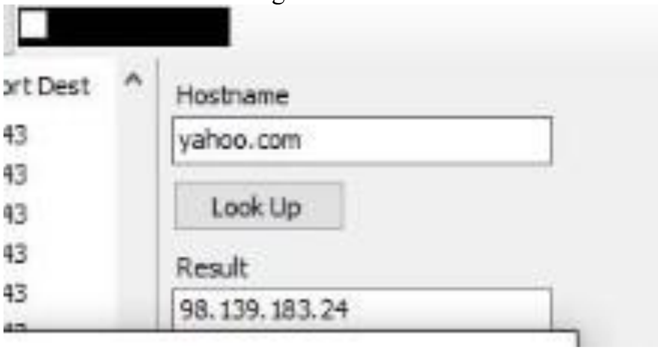
Status: Capture Stopped.

Fig. 7. Port filtering

3) IP Address Filters: This filter blocks a particular IP address in order to prevent any further traffic from that particular IP. This filter plays a very crucial role in the prevention of Ping of Death attack. The software allows the network admin to manually insert the IP. address of the user he wants to block.

H. D.N.S Lookup

This is an additional feature that allows the user to search or look up for the host names of the servers of a particular domain that are in the nearby proximity of the network. On searching the domain the software returns the IP addresses of the servers as shown in figure 8.



[redacted]

Hostname: yahoo.com

Result: 98.139.183.24

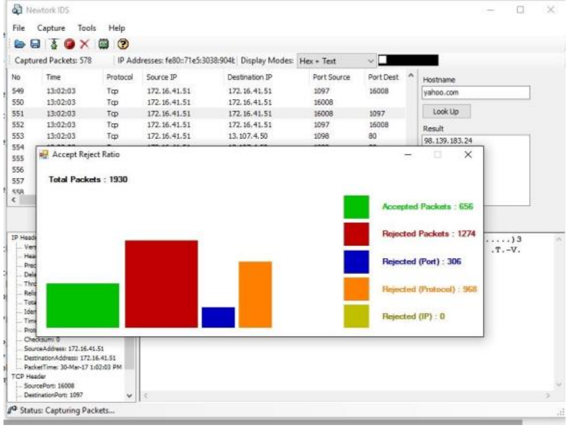
Fig. 8. DNS lookup query

I. Graphical Analysis

Finally in order to ease the process of analysis, the software provides graphical representation of the number of captured packets and the ratio of accepted versus rejected packets. Each instance in a data set is labeled as normal or intrusion. The rejected packets are further classified so as to understand how many packets are blocked by a particular IP address, Protocol and Port. The representation is in the form of Bar graphs.

V. RESULTS

The frequency of the rejected packets is incremented every time a packet is blocked, followed by the criteria of blocking. The total number of packets is determined through the sum of accepted and rejected packets. The value of the accepted and rejected packets is then scaled as per the maximum coordinates of the graph. A graphical representation of accepted packets versus blocked packets is shown in the Figure 9.



Captured Packets: 578 | IP Addresses: fe80::71e5:3038:904b | Display Modes: Hex + Text [redacted]

No	Time	Protocol	Source IP	Destination IP	Port Source	Port Dest
549	13:02:03	Tcp	172.16.41.51	172.16.41.51	1097	16008
550	13:02:03	Tcp	172.16.41.51	172.16.41.51	16008	
551	13:02:03	Tcp	172.16.41.51	172.16.41.51	16008	1097
552	13:02:03	Tcp	172.16.41.51	172.16.41.51	1097	16008
553	13:02:03	Tcp	172.16.41.51	13.107.4.50	1098	80

Hostname: yahoo.com

Result: 98.139.183.24

Accept Reject Ratio

Total Packets : 1930

- 🟩 Accepted Packets : 656
- 🟥 Rejected Packets : 1274
- 🟦 Rejected (Port) : 306
- 🟧 Rejected (Protocol) : 968
- 🟨 Rejected (IP) : 0

```
  Checksum: 0
  SourceAddress: 172.16.41.51
  DestinationAddress: 172.16.41.51
  PacketTime: 30-Mar-17 1:02:03 PM
TCP Header
  SourcePort: 16008
  DestinationPort: 1097
```

Status: Capturing Packets...

Fig. 9. Accepted versus rejected ratio

VI. CONCLUSION AND FUTURE WORK

The motive is to detect suspicious activity. It provide entry for any arbitrary data type where any random characters is not a malware. It allows generating rules to allow only safe packets to enter the LAN. It provides verbose, detailed logging. It makes intelligent use of all data which not only generate alerts, but also for correlating the data.The programmability provides the users to manually design their own security rules. The proposed system when inspected in real time can produce reliable results and leads to the development of new security mechanisms.

This paper presents a basic implementation of NIDS rules using frequent pattern analysis, whereas in future other data mining algorithms can be included to test other attacks. Various other attributes of the packet can be exploited to shield against the threats.

REFERENCES

[1] Sally Lin, Xiong Huang , The IDS technology base on association rules in Advances in Computer Science, Environment, Ecoinformatics, and Education , Part IV , 2011, pp 36-38

[2] S. Nithya, C. Jayakumar, Automatic Firewall Rule Generator for Network Intrusion Detection System based on Multiple Minimum Support, in Indian Journal of Science and Technology, India, 2016

[3] Lalli, Palanisamy, Modernized Intrusion Detection Using Enhanced Apriori Algorithm in International Journal of Wireless & Mobile Networks (IJWMN) Vol. 5, No. 2, April 2013

[4] M. H. Bhuyan, D. K. Bhattacharyya, J. K. Kalita, Network Anomaly Detection: Methods, Systems and Tools, IEEE Communications Surveys & Tutorials, 2013

[5] O. Bilalovi , D. Donko , Usage of Data Mining Techniques for Analyzing Network Intrusions in Telecommunications (BIHTEL), 2014 X International Symposium, 2014

[6] D. Gupta, S. Singhal, S. Malik, A. Singh, Network Intrusion Detection System Using various data mining techniques, in 2016 International Conference on Research Advances in Integrated Navigation Systems (RAINS - 2016), India, 2016

[7] M. A. Sayeed, M. A. Sayeedt and S. Saxena, Intrusion Detection System based on Software Defined Network Firewall, in 2015 1st International Conference on Next Generation Computing Technologies, NGCT-2015, India, 2015

[8] G.V. Nadiammai , M. Hemalatha, (2016), Effective approach toward Intrusion Detection System using data mining techniques [online]. Available:http://www.sciencedirect.com/science/article/pii/S11108665 13000418

[9] Alexis Cort, SANS Institute InfoSec Reading Room , Algorithmbased Approaches to Intrusion Detection and Response in March 16, 2004 [online]. Available:https://www.sans.org/readingroom/whitepapers/detection/algorithm-based-approaches-intrusiondetection-response-1413

[10] M. Tavallaee, E. Bagheri, Wei Lu, and Ali A. Ghorbani, "A Detailed Analysis of the KDD CUP 99 Data Set", in Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009, Canada

[11] Nour Moustafa, Jill Slay, "The significant features of the UNSW-NB15 and the KDD99 data sets for Network Intrusion Detection Systems",in Building Analysis Datasets and Gathering Experience Returns for Security (BADGERS), 2015,Japan

[12] Lekhraj Mehra, Mukesh Kumar Gupta, Harpreet Singh Gill, "An Effectual & Secure Approach for the Detection and Efficient Searching of Network Intrusion Detection System (NIDS)", in Computer, Communication and Control (IC4), 2015 International Conference, 2015, India

[13] D. Kshirsagar, S. Sale, D. Tagad, G Khandagale, "Network Intrusion Detection based on Attack Pattern", in Electronics Computer Technology (ICECT), 2011 3rd International Conference, 2011, India.

[14] K. Scarfone, P. Mell, "Guide to Intrusion Detection and Prevention Systems (IDPS)",in NIST Special Publication 800-94, Feb. 2007.

[15] E. Saboori, S. Parsazad, Y. Sanatkhani , "Automatic firewall rules generator for anomaly detection systems with Apriori algorithm", in Advanced Computer Theory and Engineering (ICACTE), 2010 3rd International Conference on, vol.6, no., pp.V6-57,V6-60, China