*IOSR Journal of Computer Engineering (IOSRJCE)*
*ISSN: 2278-0661 Volume 2, Issue 2 (July-Aug. 2012), PP 30-36*
*www.iosrjournals.org*

# Detection of Intruders and Flooding In Voip Using IDS, Jacobson Fast And Hellinger Distance Algorithms

[1] A. Rahul, [2]S.K.Prashanth, [3]B.Suresh kumar, [4]G.Arun

*[1] Assistant Professor[2]AssociateProfessor,[3]M.Tech (C.S.E) ,[4]Assistant Professor*
*[1,2,3] Department of C.S.E,Vardhaman College of Engineering, [4] Department of C.S.E,Vignana Bharathi Institute of Technology Hyderabad-501218, A.P., India*

***Abstract:-*** *VoIP services are becoming increasingly a big competition to existing telephony services (PSTN). Hence, the need arises to protect VoIP services from all kinds of attacks that target network bandwidth, server capacity or server architectural constrains. SIP Protocol is used for VoIP connection establishment. It works based on either TCP or UDP Protocols. This protocol structure is almost as same as HTTP Protocol, i.e. for every request there will be some response, even though the request is invalid. HTTP Protocol is prone to flooding attacks, like SYN-Flood attack. Because of Session Initiation Protocol (SIP) is same as HTTP, SIP is also prone to Flooding attacks. The proposed Intrusion Detection System (IDS) is used to detect the intruders in telephony system. Genetic algorithm is used to recognize the authorized user. VoIP Flood Detection System (VFDS) is aimed to detect TCP Flooding attacks and SIP Flooding attacks on SIP devices using Jacobian Fast and Hellinger distance algorithms. The Jacobian Fast Algorithm fixes the threshold limit and Hellinger distance calculation is a statistical anomaly based algorithm uses to detect deviation in traffic*

***Key words:*** *VoIP, attacks in VoIP, flood attacks, IP telephony, Jacobian Fast, Hellinger Distance, Intrusion Detection System, Genetic algorithm.*

## I. INTRODUCTION

At its simplest, Voice over Internet Protocol (VoIP) is the transport of voice using the Internet Protocol (IP), however this broad term hides a multitude of deployments and functionality and it is useful to look in more detail at what VoIP is being used for today. Currently the following types of VoIP applications are in use:

Private users who are using voice over IP for end to end phone calls over the public internet. These users typically trade quality, features and reliability for the fact that the service is very low cost and are generally happy with the service.

Business users on private networks provided by telecom and datacom providers. These services offer relatively high quality and reliability and are feature rich but come at a price.

IP trunking solutions used by long haul voice providers. Typically these offerings use private IP networks to connect islands of the PSTN together, e.g. a low cost way of calling the USA from the UK. Customers access these services using traditional black phones but the voice is carried over an IP network.

The actual problem lying in the VoIP scenario is that the VoIP servers are vulnerable to Intrusion, DoS attacks. We construct the IDS to avoid intruders based on Genetic Algorithm and VFDS(VoIP Flooding Detection System) that uses the Hellinger Distance algorithm to conclude whether there is any flooding or not. Detection of flood is based on the threshold level which is calculated using jacobsons' fast algorithm.

## II. INTRODUCTION TO IDS

Intrusion is the formal term describing the act of compromising a system. Intruder is an unauthorized user who is going to steal the information. Intruder may be insider or outsider .80% of security breaches are committed by insider. Outsiders attempt to go around firewall to attack machines on internet network. He may come from internet, dial-up lines, physical break-ins. IDS [1] is used to detect the intruders which reduce the unauthorized traffic in the network based on Genetic Algorithm [1]. Chromosome conversion [1] is used to convert packets in to chroms and it performs two basic operations as crossover (rotation) and mutation (interchange). Genetic algorithm identifies the authorized user based on the Data Set. Data Set is a Set just acts like a Database which contains the record of authorized users. The below Fig: 1 illustrates the above information. IDS is followed by VFDS is to detect the flooding is discussed in the further topics.

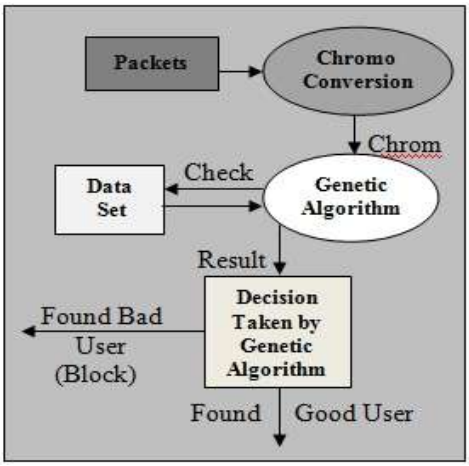


Fig 1: IDS Design Architecture
Fig.1. Intrusion Detection System Architecture

## III. INTRODUCTION TO SIP NETWORKS

The SIP [4] has established itself as the de-facto standard for VoIP services. Several providers are already offering Internet Tele- phony services based on SIP. The popularity of SIP will likely rise even more with the advent of the IP Multimedia System (IMS), the next-generation telephony network which is also based on SIP.

SIP is deployed in a client-server infrastructure where SIP clients (User Agent clients, UAC) contact a central SIP Proxy (i.e. a server) to manage ongoing sessions. It is a text based protocol designed to establish or terminate a session among two or more partners. The message format is derived from the HTTP protocol, with message headers and corresponding values, e.g. "From: user@sip.org" to denote the sender of a message.

As it is generally deployed in the open internet, SIP infrastructures can easily become a target for different attacks from the outside world, including Denial-of-Service attacks, Message Tampering, Call Hijacking and other threats.

Here it is presented a *VFDS* [2], an open security architecture that is designed to monitor the traffic flow between SIP servers and external users and proxies. The goal is to detect attacks directed at the protected SIP servers and provide a framework for attack prevention / mitigation.

Our focus lies especially on high traffic flooding attacks that can easily overwhelm a proxy's resources in terms of CPU processing power, memory requirements or bandwidth capacity. Hence, VFDS was designed with scalability in mind. As such, a layered solution with dedicated tasks (traffic monitoring, analysis, decision) with each layer optimized for scalability also presented here. The SIP traffic is forwarded to individual intelligent extension modules which provide monitoring, attack detection.

The proposed architecture is described in section I, II, IV. The approach is discussed in Section V, VI and results are shown in Section VII. Conclusion and Future work is presented in section VIII followed by References in Section IX.

The rapid evolution of voice and data technology is significantly changing the business environment with such services such as instant messaging, integrated voice and e-mail, and follow-me services—all offering an environment where people can communicate much more efficiently. To meet the demands of the changing business environment businesses are beginning to deploy converged voice-and-data networks based on SIP.

SIP was originally defined in 1999, by the Internet Engineering Task Force (IETF) in RFC 2543. The definition was the culmination of years of work in the IETF MMUSIC Working Group to provide a mechanism to allow voice, video, and data to be integrated over the same network. SIP provides the foundation for building converged networks that support seamless integration with traditional voice networks, e-mail, the World Wide Web, and next-generation technologies such as instant messaging.

As businesses continue to increase their use and reliance on converged services, reliability and availability becomes increasingly important. This paper introduces the key elements of a SIP network, further define the concept of high availability in SIP networks, and explore various techniques to increase the availability of SIP-based VoIP networks.

As shown in Fig.2 a SIP-based network consists of:

1.) ***SIP User Agent***—any network endpoint that can originate or terminate a SIP session. This may include a SIP-enabled telephone, a SIP PC client (known as a "soft phone"), or a SIP-enabled gateway.
2.) ***SIP Proxy Server***— A call-control device, such as the Cisco SIP Proxy Server, that provides many services such as routing of SIP messages between SIP user agents
3.) ***SIP Redirect Server***—A call-control device that provides routing information to user agents when requested, giving the user agent an alternate uniform resource identifier (URI) or destination user agent server (UAS).
4.) ***SIP Registrar or Location Server***—A device that stores the logical location of user agents within that domain or sub-domain. A SIP registrar server stores the location of user agents and dynamically updates its data via REGISTER messages.

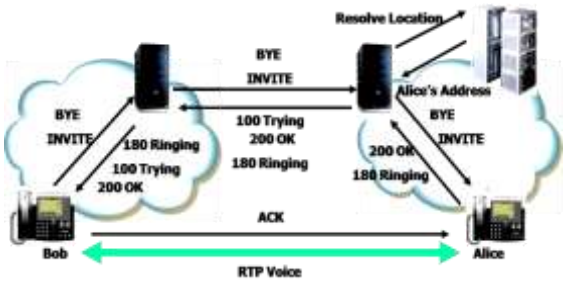


Fig.2. SIP Message flow during Connection Establishment

The recent attacks on popular web sites like Yahoo, eBay and E*Trade, and their consequent disruption of services have exposed the vulnerability of the Internet to Distributed Denial of Service (DDoS) attacks [3]. It has been shown that more than 90% of the DoS attacks use TCP. The TCP SYN flooding is the most commonly-used attack. It consists of a stream of spoofed TCP SYN packets directed to a listening TCP port of the victim. Not only the Web servers but also any system connected to the Internet providing. TCP-based network services, such as FTP servers or Mail servers, are susceptible to the TCP SYN flooding attacks. The SYN flooding attacks exploit the TCP's three-way handshake [5] mechanism and its limitation in maintaining half-open connections. When a server receives a SYN request, it returns a SYN/ACK packet to the client. Until the SYN/ACK packet is acknowledged by the client, the connection remains in halfopen state for a period of up to the TCP connection timeout, which is typically set to 75 seconds. The server has built in its system memory a backlog queue to maintain all half-open connections. Since this backlog queue is of finite size, once the backlog queue limit is reached, all connection requests will be dropped.

If a SYN request is spoofed, the victim server will never receive the final ACK packet to complete the three-way handshake. Flooding spoofed SYN requests can easily exhaust the victim server's backlog queue, causing all the incoming SYN requests to be dropped. The stateless and destination-based nature of Internet routing infrastructure cannot differentiate a legitimate SYN from a spoofed one, and TCP does not offer strong authentication on SYN packets. Therefore, under SYN flooding attacks, the victim server cannot single out, and respond only to, legitimate connection requests while ignoring the spoofed.

To counter SYN flooding attacks, several defense mechanisms have been proposed. All of these defense mechanisms are installed at the firewall of the victim server or inside the victim server, thereby providing no hints about the sources of the SYN flooding. They have to rely on the expensive IP trace back to locate the flooding sources. Because the defense line is at, or close to, the victim, the network resources are also wasted by transmitting the flooding packets.

Therefore, a simple stateless mechanism to detect SYN flooding attacks, which is immune to the SYN flooding attacks. Also, it is preferred to detect an attack early near its source, so that one can easily trace the flooding source without resorting to expensive IP trace back.

## IV. THE VFDS DESIGN

The VFDS consists of training phase and testing phase where packets are captured and HD is calculated as shown in Fig: 3. Generally VFDS detects anomalies in collections of packet streams, going through a cyclic behavior consisting of two phases: the training and testing phases.

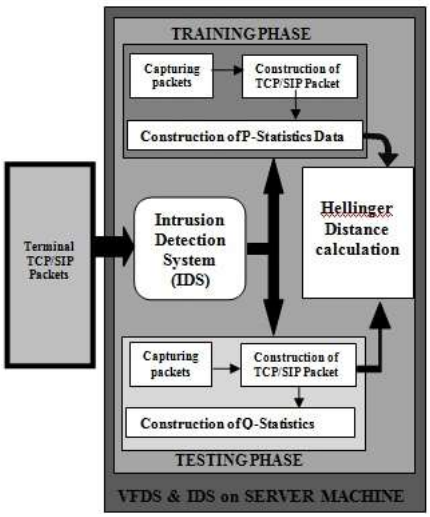


Fig.3. VFDS Design Architecture

As shown in Fig. 4, during the training phase, the training data set consisting of the attribute set is collected over n sampling periods of duration Δt over a normal traffic stream. This initial training data set is assumed to be devoid of any attacks and acts as a base for comparing with the next (n+1) th periods of the testing data set. Using the soon-tobe-described HD, we measure the distance between these two data sets. If the measured distance exceeds a threshold, an alarm is raised; otherwise, the testing data set is included

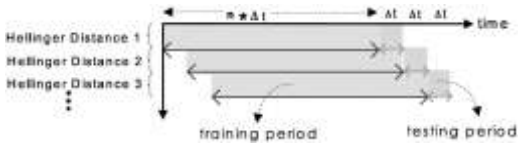


Fig.4. Relationship between training and testing periods

In the immediately (n-1) sampled traffic data to derive a new training data set. This moving window mechanism helps the training data set to adapt with the dynamics of network traffic. In order for this design to work, the following three parameters are computed online:

Using HD (soon to be described) we calculate the following:

1. The probabilistic distribution for training data. This is computed as the ratio of packets that satisfy the feature to the total number of packets received during the training phase.
2. The probabilistic distribution for testing data. This is computed as averages during the time window immediately following the training period, again as a ratio of packets satisfying the chosen feature to the total number of packets, whereas the deviation of the two probability distributions are computed using the (soon-to-be-described) HD.
3. The threshold of deviation to distinguish normal behavior from the abnormal behavior. This is used to compute a dynamic threshold as the computation progresses through cycles of training and testing phases, using Jacobson's fast deviation computing algorithm [5].

## V. HELLINGER DISTANCE

Hellinger distance presents an intrinsic way to estimate the distances between probability measures independent of the parameters. It is closely related to the total variation distance [6] but with several advantages. To explain this, let P and Q be two probability distributions on a finite sample space Ω, where P and Q on Ω are N-tuples (p1,p2, p3, …, $p_N$ ) and (q1,q2, q3, …, $q_N$) respectively, satisfying (in)equalities $p_\alpha >= 0$, $q_\alpha >= 0$, $\sum_\alpha P_{\alpha=}=1$. Then, the HD between P and Q is defined as

$$d_H^2(P,Q) = \frac{1}{2}\sum_{\alpha=1}^{N}\left(\sqrt{P}\alpha + \sqrt{Q}\alpha\right)^2 \qquad (1)$$

The HD satisfies the inequality $0 \leq d_H^2 \leq 1$,and $d_H^2 = 0$ when P=Q. Disjoint P and Q shows the maximum distance of one. Sometimes, the factor ½ is not used in the above equation. A related notion is the affinity between probability measures, which is defined as

$$A(P,Q) = 1 - d_H^2(P,Q) = \sum_{\alpha=1}^{N} \left(\sqrt{P\alpha Q\alpha}\right) \quad (2)$$

The affinity between two probability measures P and Q is one (that is, A = 1) if they are equal and zero if the measures are totally different. Further details on HD can be found in (5).

*A. Measuring Protocol Deviations Using the Hellinger Distance*

In order to detect protocol violations, depending upon the protocol to be observed and a collection of potential attacks that can launched against it, we select and track the distribution of a (small) set of attributes. Suppose we choose N attributes of a protocol, which satisfy $p_\alpha$ , $q_\alpha >= 0$, $\sum_\alpha p_\alpha = 1$, $\sum_\alpha q_\alpha = 1$ and Here, α represents an attribute in the chosen set of N attributes. Probability measure P is defined over the training data set, whereas probability measure Q is defined over the testing data set. Both P and Q are hypothesized to be an array of the normalized frequencies of all N attributes.

*B. Detection Threshold*

Normal attribute behaviors also change with time, although the strong attribute correlation makes the fluctuation of its dynamics much less than that of traffic behaviors. To accurately keep track of the normal attribute behaviors, we use a dynamic threshold for detection. Such a dynamic setting of threshold will make an attack harder to evade. We employ the stochastic gradient algorithm to compute the dynamic threshold based on the HD observed during the previous training period. Our threshold is an instance of Jacobson's Fast algorithm for RTT mean and variation [6]. Fast estimators for average a and mean deviation ν, given measurement m, can be computed as

$$Err = m_n - a_{n-1}, \quad (3)$$

$$a_n \leftarrow a_{n-1} + g.Err, \quad (4)$$

$$v_n \leftarrow v_{n-1} + h.(|Err| - v_{n-1}), \quad (5)$$

Where mn is the current sample of the HD, $a_{n-1}$ and an are the previous and current smoothened Hellinger distances, respectively, and $v_{n-1}$ and $v_n$ represent the previous and current mean deviations. To make the computation efficient, g and h are chosen to be negative exponents of two. Here, we use the values $g=1/2^3$ and $h=1/2^2$, as previous research suggested [7&8]. Although the original g and h are used in the context of RTT measurement, the underlying principles of both cases are the same: based on the past and present values, we attempt to predict the future values. The smoothened HD an is based on the observed HD m, which is measured between the probability measures P and Q. During the testing periods, we derive the estimated threshold HD ($HD^{thres}$) using the smoothened HD (2) and the mean deviation (3):

$$HD_{n+1}^{thres} = X * a_n + \eta * v_n \quad (6)$$

The purpose of the multiplication factors X and $\eta$ is to get a safe margin for the setting of the threshold value, so that VFDS avoids any false alarms without degrading its detection sensitivity. The first factor in (4), which largely depends upon the observed HDs, should be large enough to make the first part of (4) higher than the maximum observed HD, whereas the second factor is tied with the variations of these observed Hellinger values. These two factors are adjustable parameters and can be properly tuned during the training period.

## VI. DEPICTING PROTOCOL BEHAVIORS

We used SIP Packet generating tools which is capable of generating SIP, TCP, UDP packets, to experimentally profile normal protocol behaviors.

*A. Observation of TCP Packets*

In order to study the attribute behaviors of VoIP traffic, we build a testbed. The testbed consists of a PC(Intel P-IV 2.4GHz, 1GB RAM) in which a virtual LAN is configured with VMWARE in which Linux operating system is installed out of which one System is acting as Proxy Server and the other one is acting as SIP clients. Enterprise networks A and B are simulated within same PCs equipped with SIP traffic generators in client machine and VFDS is installed in Proxy Server playing the role of multiple systems.

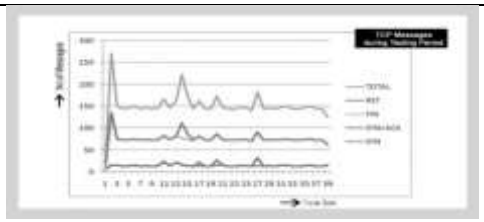

Fig.5. TCP attribute behavior during training

On the server we install our VFDS in which we will capture the incoming packets generated by sip tool and find out how many no of packets arrived in testing period. During experiment for the duration of one minute a total of 2225 SYN and 12050 FIN packets were arrived in testing period. The statistics during training period for the duration of one minute a total of 519 SYN Packets and 2326 FIN packets were arrived as shown in fig 5 and fig 6 shows the statistics during testing period.

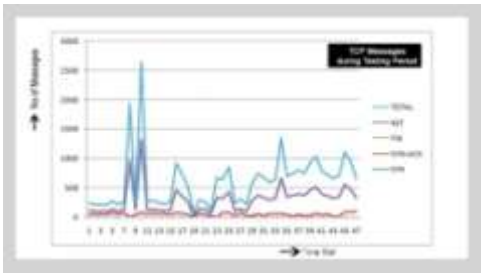

Fig.6. TCP attributes behavior during testing periods.

*B. Observation of SIP Packets*

To study the traffic of the SIP packets we have selected Sip packet generator tool. During the training phase we collected a data of 2323 INVITE packets and 4650 no of 200-OK packets in testing period a total number of 27583 INVITE packets and 32925 no of 200-OK packets were arrived. The SIP traffic during training and testing periods are shown the graphs in fig 7 and fig 8.

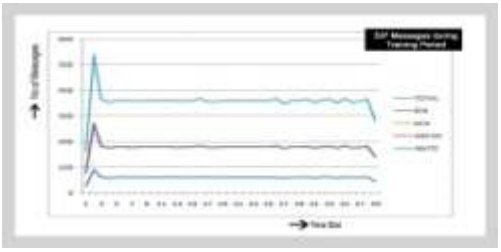

Fig. 7. SIP-attribute behavior during Training Period

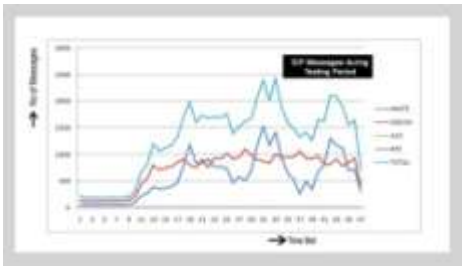

Fig.8. SIP-attribute behavior during Training Period

## VII. COMPUTING THE HELLINGER DISTANCE FOR TCP

In this experiment, we choose four attributes SYN, SYN-ACK, FIN for the calculation of the TCP HD values as shown in fig 8.

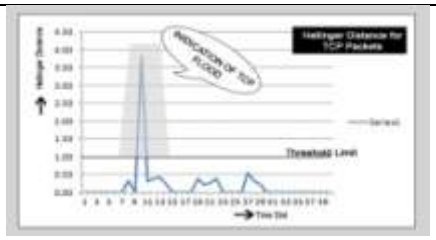


Fig.9. HD of TCP attributes.

Now, suppose that there are NSYN, NSYN_ACK, NFIN, and NRST packets during the training period (that is, in n*Δt time). P is an array of the normalized frequencies of pSYN, pSYN_ACK, pFIN, and pRST over the training data set, and Q is an array of the normalized frequencies of qSYN, qSYN_ACK, qFIN, and qRST of the same attributes observed over the testing period (that is, at the (n + 1)th sampling duration), defined as follows:

$p_\alpha = N_\alpha / N_{Total}$ (7)

Where $\alpha \in \{SYN, SYN-ACK, FIN, RST\}$, and

NTotal = (NSYN+NSYN-ACK+NFIN+NRST) (8)

$p_\alpha = N^I_\alpha / N^I_{Total}$ (9)

Where $\alpha \in \{SYN, SYN-ACK, FIN, RST\}$, and $N^I Total = (N^I SYN+N^I SYN\text{-}ACK+N^I FIN+N^I RST)$ (10)

The HD between IP and QQ at the end of on þ 1Þth sampling period is computed as follows:

$$HD = \left(\sqrt{P_{SYN}} - \sqrt{Q_{SYN}}\right)^2 + \left(\sqrt{P_{SYN-ACK}} - \sqrt{Q_{SYN-ACK}}\right)^2 + \left(\sqrt{P_{FIN}} - \sqrt{Q_{FIN}}\right)^2 + \left(\sqrt{P_{RST}} - \sqrt{Q_{RST}}\right)^2 \quad (11)$$

*A. Computing the Hellinger Distance for SIP*

We choose to experiment with SIP attributes INVITE, 200 OK, ACK, and BYE. Here, the probability measure P is an array of the normalized frequencies of PINVITE, P200 OK, PACK, and PBYE over the training data set. Similarly, Q is an array of QINVITE, Q200 OK, QACK, and QBYE during the chosen testing period. All other details are similar to the previous example. To calculate the HD between P and Q, we use

$$HD = \left(\sqrt{P_{INVITE}} - \sqrt{Q_{INVITE}}\right)^2 + \left(\sqrt{P_{200\ OK}} - \sqrt{Q_{200\ OK}}\right)^2 + \left(\sqrt{P_{ACK}} - \sqrt{Q_{ACK}}\right)^2 + \left(\sqrt{P_{BYE}} - \sqrt{Q_{BYE}}\right)^2 \quad (12)$$

Fig. 10 shows the HD for the SIP attribute set of {INVITE, 200 OK, ACK, BYE}.

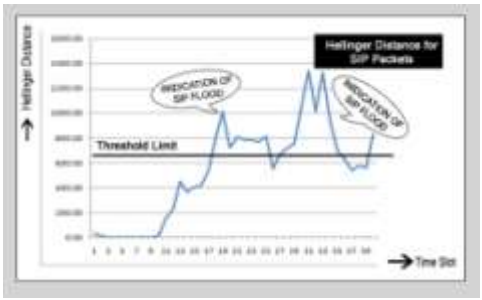


Fig.10. HD of SIP attributes.

Fig. 10 illustrates the dynamics of the SIP data estimated threshold HD and the observed HD. Because the spikes of the observed HD are much higher than those of the estimated threshold distance, no false alarm is raised.

## VIII. CONCLUSIONS AND FUTURE WORK

SYN, INVITE, and TCP packet floods pose a serious threat to the IP telephony infrastructure. The multiprotocol-based VoIP service needs a fast and generic detection mechanism working across different protocol layers. Here it is investigated, the protocol attribute behaviors and characterize the network traffic with respect to the intrinsic correlation among protocol attributes. Utilizing HD, it is presented an online statistical flooding detection mechanism, called VFDS, in which we measure the similarity (or dissimilarity) of the correlation among protocol attributes at different times. The rationale behind our approach is that a deviation from normal protocol behaviors can be measured and quantified.

In this paper it was reviewed the concepts of the reference [2]. A trial to simulate and develop a new frame work to prove that, this mechanism will also work for low traffic and highly congested traffic and to avoid false alarm.

## References

[1] Mohammad Sazzadul hoque , Md. Abdul mukit2 and Md. Abu Naser bikas "An implementation of intrusion detection system using genetic algorithm " International Journal of Network Security & Its Applications (IJNSA), Vol.4, No.2, March 2012
[2] Hemant Sengar, Haining Wang, Duminda Wijesekera, and Sushil Jajodia, "Detecting VoIP Floods Using the Hellinger Distance", IEEE Transactions On Parallel And Distributed Systems, Vol. 19, No. 6, June 2008.
[3] R Jones, J Cruz, "Carrier Class Voice over IP", August 1999, 9 pages,.
[4] M. Handley, ACIRIH .Schulzrinne Columbia U E. Schooler Cal Tech J. Rosenberg ,Bell Labs ,March 1999 "SIP: Session Initiation Protocol", March 1999, 132 pages, ftp://ftp.isi.edu/in-notes/rfc2543.txt
[5] Rakesh Arora, Voice Over IP: Protocols and Standards http://www.cis.ohio-state.edu/~jain/cis788-99/voip_protocols/index.html ,
[6] V. Jacobson and M.J. Karels, "Congestion Avoidance and Control," Proc. ACM SIGCOMM '88, pp. 314-329, Aug. 1988.
[7] W. Stevens, TCP/IP Illustrated Volume-1, first ed. Addison-Wesley, 1994.
[8] D. Pollard, Asymptopia, first ed., book in progress, http://www.stat.yale.edu/pollard/, 2000.