Defense Techniques of SYN Flood Attack Characterization and Comparisons

Shaila Ghanti, G. M. Naik
(Corresponding author: Shaila Ghanti)
Department of Electronics, Goa University
Taleigao Plateau, Goa 403206, India
(Email: shailaghanti@yahoo.com)
(Received Mar. 26, 2017; revised and accepted June 26, 2017)

Abstract

Automation systems are being used widely for performing different tasks at homes, offices, and industries. Remote clients access these web based automation system services on the Internet. Web services are also prone to attacks due to SYN flood that deny genuine clients a right to use the services. Therefore such services are required to be secured. The primary purpose of this manuscript is to characterize the defense mechanisms and compare the technical details involved in defense mechanisms of attacks due to SYN flood. This will help the researchers to propose improved and more efficient defense mechanisms. The manuscript furthermore includes the experimental study of the Victim Side SYN Flood (VSSF) attack protection system implemented using a general purpose processor during the attack. Subsequently the study compares the performance of VSSF attack protection system implemented using general purpose processor, with the VSSF attack protection system implemented using NIOS core processor. It is found that the NIOS core processor based protection system performance is better than the general purpose processor based systems.

Keywords: Attacks; Defense Mechanisms; FPGA; NIOS Processor; SYN Flood Attack

1 Introduction

Today a vast number of online services are used to transfer crucial data. These online services are widely used by the users to transfer data. Meanwhile the basic principles like integrity, confidentiality, and availability of resources have to be applied to secure these services.

Distributed Denial of Service (DDoS) attack is generated on the network to deny access to the services by utilizing the resources of the network or system [2, 4, 14, 18]. DDoS attacks are classified into many types of attacks, such as UDP flood, SYN flood, Ping flood attacks, etc. [3, 7, 25] . In this study we have concentrated on SYN flood attacks. SYN flood attack is a type of DDoS attack that will target the specific resources of victims [29]. These attacks generated on the server will disrupt the services and genuine users are not able to access these services. Therefore, there is a need to protect such services from attacks. Although there are many types of defense systems against the SYN flood attacks, contributed by the researchers are available, designed using software [5] and hardware [10], still the issue remains challenging.

This manuscript deals with the characterization and classification of defense mechanisms used only for SYN flood attack. This will help the researchers to propose improved and more efficient defense mechanisms. Furthermore it proposes the Victim Side SYN Flood (VSSF) attack protection system implemented using a general purpose processor.

Subsequently the performance of VSSF attack protection system implemented using general purpose processor is compared, with the VSSF attack protection system implemented using NIOS core processor [10]. This comparison will help to choose and design the appropriate security system.

Contributions of this manuscript are:

1) Identify the characteristics of various defense mechanisms against attacks due to SYN flood.

2) Compare the various defense systems, which will help researchers to invent a comprehensive and efficient defense system so that the genuine user's access to server is not denied.

3) Experimental study of proposed VSSF attack protection system implemented on a PC using a general purpose processor.

4) Compare the performance of the general purpose processor based system, with the NIOS core processor based VSSF protection system.

2 SYN Flood Attack

To communicate over the network between any server and client using TCP, the client has to first set up a connection with the server. Then they can receive and transmit the data. After the data transfer is done, the connection has to be disconnected. The process of setting up of the connection between client and server involves exchange of three way handshake signals [23]. The client first sends the SYN packet to the server, the server in turn sends the SYN-ACK back to the client by setting the TCP half open connection on the server. Then the ACK is sent from the client side to the server and the connection is set up between the client and server in normal situation.

In order to disrupt the Internet services available on the server the malicious attacker can generate the SYN flood attack. During this attack the attacker attacks the server by sending many spoofed SYN packets. Due to these packets the TCP half open connections are set up on the server and then the server sends back SYN-ACK to the clients and waits for the corresponding ACK [15, 22]. As IP addresses used are spoofed there may not be a client with that IP address and subsequently the server will not receive corresponding ACK packet. Each TCP half open connections reserve certain resources of server. Large number of SYN flood attack leads to the consumption of server resources and genuine users will not have access to the server.

3 SYN Flood Attack Defense Mechanism Classifications and Their Characterizations

Literature review shows that many types of defense mechanisms are contributed by the researchers. Here we characterize and compare the different mechanisms.

3.1 Characterization

Defense mechanisms of SYN flood systems are classified based on the location of deployment. The defense mechanisms can be deployed at the source side, victim side and intermediate on the network side.

Source side defense systems: These are the defense systems implemented at the attacker side. These defense systems are more effective as the attacks are blocked at the source side itself. However these defense systems need to be deployed at all the source/client side.

Victim side defense systems: Victim side defense systems can detect and defend the attack easily as all the attack requests are available at the victim side. It is sufficient only to deploy the defense system near the victim.

Intermediate defense system on the network: Generally the defense systems are deployed in between the source and the victim. Here detecting the attack is difficult as all the packets of attack are not available at one place.

Defense mechanism systems are classified as Distributed or Autonomous. The autonomous systems independently take decision to protect the servers. Whereas distributed systems needs coordination amongst different systems to protect the servers.

3.2 Comparisons

The comparison of the following different defense mechanisms of attack due to SYN flood is shown in Table 1.

SYN Cookies: In this technique no memory is reserved to store the initial request information on server. Instead a code is generated using the received initial request information and cryptographic techniques. This code is used as "sequence number" in the SYN-ACK packet and sent back. When the respective ACK is received it extracts the initial request information and is used to set up the connection [5].

SYNkill: Detects the attack due to SYN flood and then responds to lessen the effect of attack. SYNkill tool generates RST and ACK packets depending on the type of client request so that the resources of the server are not wasted [24].

Ingress Filtering: It is a source side defense system. It will only forward the packets that have the address of source(Prefix) same as the source network address(prefix). The exit or gateway routers are configured in such a way that they block all packets that are not the component of the input network address [9]. It is more effective provided, it is implemented for all clients.

SYN Cache: In this method the resource allocation strategy is changed. Minimum initial information of the request is stored and then during the connection all the required information is stored [16].

SYNMON: Network processor is used as a processing unit to detect the SYN attack. An embedded system is designed to detect the attack using the CUSUM method [17].

D-WARD: This technique is implemented near the source, but also supports near the victim and within the core. However the source end deployment supports good response as compared to victim end and core side. The D-WARD scheme consists of 3 units called observation unit, traffic policing unit and rate-limiting unit. It maintains the information of all the traffic and then the statistics of aggregate flow is periodically compared with the models of traffic to detect the attack. Attack Mitigation is performed by the method called rate limiting [19].

Throttling Source Side SYN Flooding Attack: It is a defense mechanism that needs to be deployed at the attacker/source side. Bloom filter is used to store the information of traffic as it provides storage

efficient data structure. CUSUM method detects the malfunctions in the traffic. Thus identifies the attack and then the mitigation of attack is performed using the ingress filter and rate limiting scheme [6].

A Router-based Novel Scheme: This method detects and mitigates the attack due to SYN flood. It is a router based mechanism that uses Bloom filter (counting) to keep track of number of SYN and FIN/RST packets. During the attack,if the count of SYN is more than number of FINs [26], then the attack is detected. During mitigation every client's first request is dropped. The client retransmits the request and only such requests are forwarded to server. Thus the effect of attack is mitigated.

Active Probing Method: Attack detection is done using the probing technique. Based on the values of TTL the Rate-limiting filter is used to mitigate the attack [28].

Detection of IP Header Threats Using Anomaly Detection: Anomaly detection methods use the information from headers of TCP and IP to detect the attack. Traffic capturing tools are used to capture the TCP and IP information on the network. Anomaly detection supports three types of detection methods. The first type is the protocol detection wherein any violations in the protocol pattern is detected. The second type is based on the rate that detects the attack by comparing it with the normal traffic rate and the third type is based on behavioural changes that compares with the normal behaviour of clients with the servers. In this rate based detection is used. Once the attack is detected attentive messages are sent to the administrator [13].

STONE: This method presents a mechanism with expert system functions that protects the server from DDoS attack. STONE detects the attack and mitigates the attack. The network traffic is captured and it aggregates all the addresses into common prefix of IP addresses. The attack is identified when these aggregated data deviates from the regular traffic flow. Following the attack discovery the STONE allows traffic from known sources to access the services where as the other traffic is blocked [12].

Secured Ethernet Interface System: This method detects and mitigates the attack. It classifies the incoming requests as good by authenticating. Every incoming request with the good IP is forwarded and other requests are blocked. During spoofed SYN flood attack it allows only good requests to be passed to the server and blocks all other attacks. However it is likely that the attack requests with the spoofed good IPs are forwarded to the server. Once the number of half open connections becomes greater than the threshold value then the good registry is cleared. Since the good registry is cleared every incoming request is authenticated and then only it forwards. Thus it blocks the attacks [10].

3.3 Parameters To Be Measured

SYN flood attack defense systems performance can be analysed based on the below mentioned parameters that need to be measured by the researchers.

1) Response time: It gives the total time period needed by the server(web server) to react to client request. In linux the tool called time curl, ab tool (linux/windows) can be used [21].

2) Connections and half-open connections: Total half-open and complete connections setup on the server is an important parameter. The connections information indicates the number of genuine client requests served by the server. To measure these parameters Netstat tool is used.

3) Processor utilization: It is used to measure the quantity of processing achieved by the server processor. Top tool can be used to measure the utilization of processor.

4) Network bandwidth: The bandwidth of the network that is protected is to be measured. This indicates the traffic details.

The response time, number of TCP connections, half open connections, processor utilization and network bandwidth parameters can be measured in three different scenarios while genuine requests are sent to the server.

1) Without generating attack and no protection system;

2) With attack but no protection system;

3) With attack as well as protection system.

4 VSSF Protection System

In Secured Ethernet Interface System [10] the protection against attack is performed using the NIOS II soft core processor. It is interesting to compare the performance of the SYN flood attack protection system using PC based pure software and FPGA based NIOS processor. This information is very useful to decide upon what type of security system must be used for protecting the services.

The VSSF protection system is proposed and implemented using the general purpose processor on a PC. This system is implemented using the same algorithm as proposed in Secured Ethernet Interface System and the performance is analysed experimentally.

4.1 VSSF Protection Method

Figure 1 indicates the VSSF protection method used for protecting the SYN flood attack protection system as used

Table 1: Comparison of SYN flood attack defense methods

Year	Method name	Deployment	Method used	Spoofing	Testing (Simulations/ Real time)	Distributed/ Autonomous	Testing data	Software/ hardware	Results/ parameters measured
1996	SYN cookies	Victim	No need of memory to store initial request information	Yes	——	Autonomous	—	Software	Implemented in Linux
1997	SYNkill	Victim	Classification of client requests	Yes	Real time	Autonomous	—-	Software	Delay in setting up of connections and number of connections
2000	Ingress Filtering	Source	Filtering	Yes	—-	Autonomous	—	Software	—
2002	SYN Cache	Victim	Initial minimum information is stored	—	Real time	Autonomous	—-	Software	Percentage of connections set up and time taken during the attack
2005	SYNMON	Victim	CUSUM	yes	Realtime	Autonomous (Detection)	Packt, hping	Hardware/ Network Processor	Attack detection
2005	D-WARD	source	Aggregate flow statistics and compare with models of traffic	Yes	Emulab	Autonomous/ distributed	Cleo tool	Software	Number of connection, connection delay, failed connections
2006	Throttling SYN flooding attack	Source	Bloom filter, CUSUM method, ingress filter	yes	Simulations (ns2 simulator)	Autonomous	DARPA dataset	Software	Number of connections set up and detection rate
2007	Router based Novel scheme	Victim	Counting Bloom filter to keep track of SYN and FIN/RST. Persistence of client property	Yes	Simulations (Trace driven)	Autonomous	—	Software	Keeps track of number of SYNs and FIN/RSTs
2008	Active Probing technique	Victim	Active probing and Ratelimiting method	Yes	Simulations (NS2-simulator)	Autonomous	Simulations	Software	Bandwidth consumption of server by legitimate traffic
2011	Anomaly Detection	Vctim	Rate based Anomaly Detection	—-	Real time	Autonomous (Detect)	tcpdump	software	Packets are tested
2015	STONE	Distributed	Streaming processing paradigm	—	Real time	Distributed	—	Software	Attack detection time and Mitigation precision
2016	Secured Ethernet Interface System	Victim	Keeping track of the genuine requests	yes	Real time	Autonomous	Ostinato and ab tools are used	FPGA, NIOS (Hardware)	Measured half open connections, connections set up

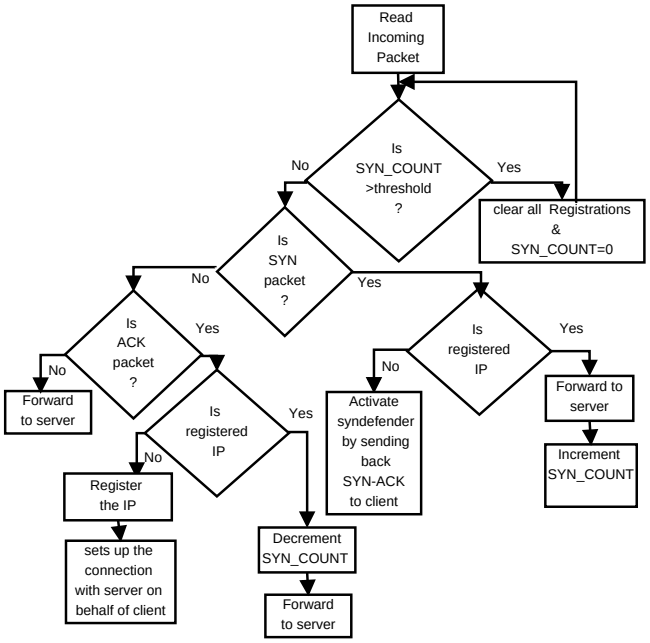

Figure 1: VSSF protection method [10]

by S.Ghanti and G.M.Naik [10], and is based on [11, 24, 27].

This method detects the spoofed attack and also it blocks such attacks. In this method a registry is used that maintains the information about the IP addresses of clients that have already accessed and set the connection with the server. Every incoming client request needs to be identified by the VSSF protection system as a genuine client request or from an attacker and is explained below.

The IP address of the incoming client request is first compared with the contents of the registry. If the entry is not traced in the registry then these are termed as new clients or non registered clients. These new clients are not forwarded to the server. Instead the SYN-ACK packet is sent to the client using the syndefender. If this request had come from the genuine client then corresponding reply is received by the VSSF protection system and will set up the connection between the client and the server. Also entry in the registry is added with such request information. If the request had come from 'non registered attack clients' then the VSSF protection system will not receive the ACK thus they are treated as attack requests and are not forwarded instead they are blocked.

An incoming client request if found in the registry, then it is treated as registered client request. The registered client requests are forwarded to the server and the server communication continues in a normal way.

The VSSF protection system uses the below explained method to detect the spoofed client requests attack. The spoofed client requests generated can be either from registered or non registered IP addresses. The non registered spoofed requests are treated as new client requests and taken care by not forwarding to the server and activating the syndefender. Thus non registered spoofed requests are taken care by the VSSF protection system.

In case of registered spoofed request attacks, the VSSF protection system uses an innovative method to detect and block the spoofed attacks. On receiving the spoofed registered request the VSSF protection method forwards such packets to the server as they are registered. For such packets SYN_Count is incremented and the SYN_ACK is sent back. Since it is a spoofed registered request, ACK is not received thus the SYN_Count will not be decremented. Thus for every registered spoofed request the SYN_Count goes on increasing. The spoofed attack is detected once the count reaches the threshold value. Once it detects the attack further attack is blocked by clearing the registry. Thus the VSSF protection system detects the attack and it also blocks the attack.

4.2 Software-based VSSF Protection System

To implement the software based VSSF protection system a PC with two Network Interface Cards (NIC) is used. This system needs to be connected between the server that needs to be secured and the clients. To start with, this system is configured using iptables as router so that it forwards the packets. The VSSF protection system should be able to capture every incoming packet and analyse it using the same algorithm as used in FPGA based Secured Ethernet Interface System shown in Figure 1 to detect and block the attack [10]. The VSSF protection system software is implemented using **libipq** so that it analyses every incoming packet and accordingly the packet is forwarded or blocked [8]. **Libipq** is an iptables packet queing development library at the user space. The libipq provides different APIs to create handle, to set mode, to read packets from queue, to issue verdict on packet like drop, forward, to destroy the handle etc. It allows packets to pass to the user space where packet details can be analysed. Then the packets can be passed to the kernel indicating whether packet can be dropped or forwarded to the server. The packet contents can also be modified if required and then the packets can be passed to the kernel from user space.

Libipq APIs are used to monitor the incoming requests from the clients/attackers and identify the SYN flood attacks and accordingly block the SYN flood attacks is shown in Figure 1 so that server resources are not consumed by the attacks, and also genuine users will have access to the server.

The set up used for the software based experiment is shown in Figure 2. One computer is configured as a web server, while another is used as a client to generate the attacks and the genuine requests. The third machine is the VSSF protection system implemented using software. We have used apache bench ab tool to generate genuine client requests to the server [1] and Ostinato tool to generate SYN flood attack to the server [20]. Experiments were conducted to study the response of VSSF protection system with and without attack.

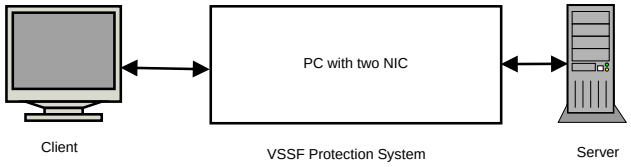

Figure 2: The software based VSSF protection system experimental set up

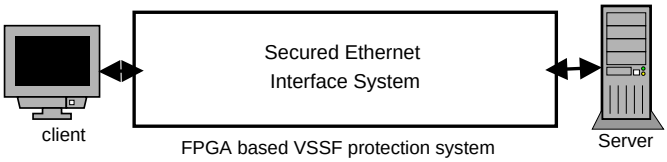

Figure 3: FPGA based protection system

4.3 FPGA-based Protection System

The FPGA based Secured Ethernet Interface System (referred here as FPGA-based VSSF protection system) was implemented on a hardware DE-4 FPGA board as a System on Chip and was reported in [10] by Ghanti and Naik. It uses Stratix IV device supported by the industry standard peripherals. Different IP cores used are Triple-speed Ethernet core IP, receive and transmit SGDMA, NIOS processor, on chip memory, etc. [10]. SYN flood attack protection system flowchart is as shown in Figure 1 and was implemented using Quartus II.

The experimental setup used in FPGA based Secured Ethernet Interface System [10] is shown in Figure 3. One computer was configured as a web server, while another was used as a client to generate the attacks and the genuine requests.

The FPGA based protection system was connected just before the server. Experiments were performed to study the response of VSSF protection system with and without attack. The results were discussed in [10] by S.Ghanti and G.K.Naik.

5 Results and Discussions

5.1 Results of Software-based VSSF Protection System

To find out the performance of software based VSSF protection system, studies are conducted initially by generating genuine requests from the client using ab tool (without generating attack) to the server and the response is noted. In this experiment ab tool is used to generate 500 genuine requests to the server.

Later the client generates genuine requests with the attacks to the server . The attacks are generated using ostinato tool to the server. Then the response of the server is recorded and is shown in Figure 4 and Table 2.

From Figure 4 it is clear that 90 % of the requests from client are served within less time when no attack is

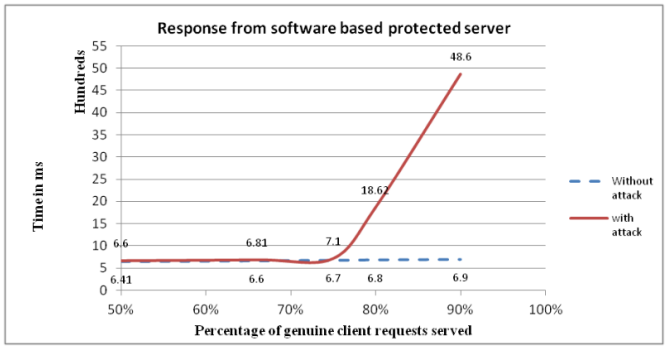

Figure 4: Response from software based VSSF protected server

Table 2: Response of software based VSSF protection system with and without attack (when client sends 500 genuine requests).

	Using PC based protection system without attack	Using PC based protection system with attack
Transfer Rate (Kilobytes/sec)	0.49	0.19
Time taken for tests (in seconds)	315	825

generated. During the attack, genuine clients could access the server but only the delay is more. It may be seen in Figure 4 that there is a sharp change at around 75% of the client requests i.e. response time of server quite significant when there is an attack. Thus, the software based VSSF protection system protects the server from the attack and also genuine clients can access the server during attack.

5.2 Comparison of Software(PC)-based With The FPGA-based Protection System

The experimental results of the PC based protection system are shown in Figure 4. The results of the experiments conducted by generating attacks and genuine clients' requests to the server that is protected by FPGA based SYN flood attack protection systems as reported by S.Ghanti and G.K.Naik in [10] is used here for comparison. The comparison of results of software based VSSF protection system, with the FPGA based protection system are tabled in Table 3 and are shown in Figure 5.

In Figure 5 there is a sharp change at 75% of client requests when the server is protected using software based system, but the response of hardware i.e. FPGA based VSSF protection system shows much better response as

Table 3: Comparison of FPGA based protection system and software based VSSF protection system with attack (500 Genuine Requests with attack are sent from client to the server).

	Using PC (software) based protection System	Using FPGA based protection system [10]
Transfer Rate (Kilobytes/sec)	0.19	2.25
Time taken for tests(in seconds)	825	64.179

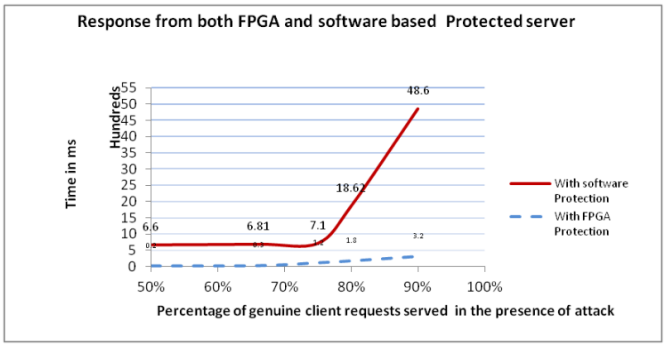

Figure 5: Comparison of FPGA based and software based protection system

compared to software based in spite of an attack. Figure 6 indicates that the FPGA (NIOS) based VSSF protection system supports higher transfer rate as compared to software based VSSF protection system.

6 Conclusions

SYN flood attacks generated on the servers cause the services to be disrupted. Defense systems reported in the literature are characterized and compared extensively. Different parameters of defense system that should be measured experimentally are suggested. This will help researchers to develop better efficient systems.

This article also demonstrates the implementation and working of the VSSF protection system on PC experimentally. It then compares the performance of PC based with the FPGA based System on Chip protection system. It is found that VSSF protection system implemented using software efficiently blocks the attack and allows genuine requests to access the server. The FPGA based SYN flood attack protection system supports the faster data transfer than the software based system. Thus the protection system designed using FPGA is more efficient than the

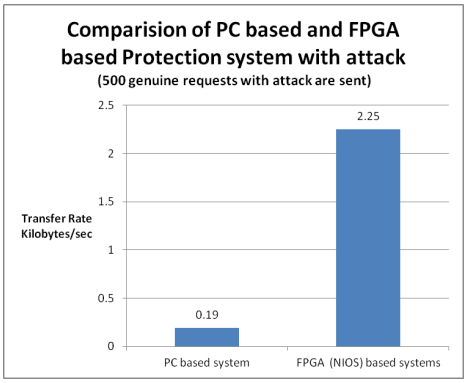

Figure 6: Comparision of PC (software) based and FPGA based VSSF protection system with attack

software based protection system.

Though the FPGA based SYN flood attack protection system shows better results as compared to PC based systems, it may be noted that FPGA solutions are not easy as one requires access to hardware, whereas PC based system provides a solution which can be implemented and administered locally very easily. Further the response of software based VSSF protection system can be easily improved by incorporating parallel processing.

Acknowledgments

The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] Apache, *ab - Apache http Server Benchmarking Tool*, Jan. 12, 2018. (`https://httpd.apache.org/docs/2.4/programs/ab.html`)

[2] R. C. Baishya, N. Hoque, and D. K. Bhattacharyya, "DDoS attack detection using unique source IP deviation," *International Journal of Network Security*, vol. 19, no. 6, pp. 929–939, 2017.

[3] S. Behal and K. Kumar, "Characterization and comparison of DDoS attack tools and traffic generators: A review," *International Journal of Network Security*, vol. 19, no. 3, pp. 383–393, 2017.

[4] S. Behal, K. Kumar, M. Sachdeva, "Discriminating flash events from DDoS attacks: A comprehensive review," *International Journal of Network Security*, vol. 19, no. 5, pp. 734-741, 2017.

[5] D. J. Bernstein, *SYN Cookies*, Jan. 12, 2018. (`http://cr.yp.to/syncookies.html`)

[6] W. Chen and D.-Y. Yeung, "Throttling spoofed SYN flooding traffic at the source," *Telecommunication Systems*, vol. 33, no. 1, pp. 47–65, 2006.

[7] S. Deore and A. Patil, "Survey denial of service classification and attack with protect mechanism for TCP SYN flooding attacks," *International Research*

Journal of Engineering and Technology, vol. 3, no. 5, pp. 1736–1739, 2016.

[8] die.net, *libipq(3) - Linux Man Page*, Jan. 12, 2018. (URLhttps://linux.die.net/man/3/libipq)

[9] P. Ferguson, D. Senie, *Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing*, RFC 2827, 2000.

[10] S. R. Ghanti and G. Naik, "Efficient data transfer rate and speed of secured ethernet interface system," *International Scholarly Research Notices*, vol. 2016, 2016.

[11] S. R. Ghanti and G. Naik, "Protection of server from SYN flood attack," *International Journal of Electronics and Communication Engineering & Technology*, vol. 5, no. 11, pp. 37–46, 2014.

[12] V. Gulisano, M. Callau-Zori, Z. Fu, R. Jiménez-Peris, M. Papatriantafilou, and M. Patiño-Martínez, "Stone: A streaming DDoS defense framework," *Expert Systems with Applications*, vol. 42, no. 24, pp. 9620–9633, 2015.

[13] S. Haris, G. M. W. Al-Saadoon, A. P. D. R. Ahmad, and M. Ghani, "Anomaly detection of IP header threats," *International Journal of Computer Science and Security*, vol. 4, no. 6, p. 497, 2011.

[14] N. Hoque, D. K. Bhattacharyya, and J. K. Kalita, "FFSC: A novel measure for low-rate and high-rate DDoS attack detection using multivariate data analysis," *Security and Communication Networks*, vol. 9, no. 13, pp. 2032–2041, 2016.

[15] D. Kshirsagar, S. Sawant, A. Rathod, and S. Wathore, "CPU load analysis & minimization for TCP SYN flood detection," *Procedia Computer Science*, vol. 85, pp. 626–633, 2016.

[16] J. Lemon, "Resisting SYN flood dos attacks with a SYN cache," in *Proceedings of the BSD Conference (BSDC'02)*, pp. 89–97, 2002.

[17] B. Lim and M. S. Uddin, "Statistical-based syn-flooding detection using programmable network processor," in *Third International Conference on Information Technology and Applications*, vol. 2, pp. 465–470, 2005.

[18] J. Mirkovic and P. Reiher, "A taxonomy of DDoS attack and DDoS defense mechanisms," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 2, pp. 39–53, 2004.

[19] J. Mirkovic and P. Reiher, "D-ward: A source-end defense against flooding denial-of-service attacks," *IEEE Transactions on Dependable and Secure Computing*, vol. 2, no. 3, pp. 216–232, 2005.

[20] Ostinato, *Ostinato Network Traffic Generator Network Traffic Generator and Analyzer*, Jan. 12, 2018. (http://ostinato.org/)

[21] S. Rao and S. Rao, "Denial of service attacks and mitigation techniques: Real time implementation with detailed analysis," *SANS Institute Reading Room*, 2011.

[22] H. Safa, M. Chouman, H. Artail, and M. Karam, "A collaborative defense mechanism against SYN flooding attacks in IP networks," *Journal of Network and Computer Applications*, vol. 31, no. 4, pp. 509–534, 2008.

[23] S. Sathwara, C. Parekh, "Distributed denial of service attacks – TCP SYN flooding attack mitigation," *International Journal of Advanced Research in Computer Science*, vol. 8, no. 5, pp. 2392–2396, 2017.

[24] C. L. Schuba, I. V. Krsul, M. G. Kuhn, E. H. Spafford, A. Sundaram, and D. Zamboni, "Analysis of a denial of service attack on TCP," in *Proceedings of IEEE Symposium on Security and Privacy*, pp. 208–223, 1997.

[25] A. Srivastava, B. Gupta, A. Tyagi, A. Sharma, and A. Mishra, "A recent survey on DDoS attacks and defense mechanisms," in *Advances in Parallel Distributed Computing*, pp. 570–580, 2011.

[26] C. Sun, J. Fan, L. Shi, and B. Liu, "A novel router-based scheme to mitigate SYN flooding DDoS attacks," *IEEE INFOCOM (Student Poster)*, 2007.

[27] C. Sys, *Check Point Software Techs., Inc. v. SRI Int'l, Inc.*, Jan. 12, 2018. (https://www.casemine.com/judgement/us/5914e6c3add7b04934911071)

[28] B. Xiao, W. Chen, and Y. He, "An autonomous defense against SYN flooding attacks: Detect and throttle attacks at the victim side independently," *Journal of Parallel and Distributed Computing*, vol. 68, no. 4, pp. 456–470, 2008.

[29] S. T. Zargar, J. Joshi, and D. Tipper, "A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 2046–2069, 2013.

Biography

Shaila Ghanti is a Ph.D. student in the Department of Electronics Goa University, Goa. She is teaching in the Department of Computer Science in Chowgule College, Margao Goa, India. She is an author for few research paper and completed two minor research project. Her research area is networking, network security and embedded Systems.

Prof. Gourish Naik obtained his Ph.D from Indian Institute of Science, Bangalore (1987) and served the institute as research associate in the areas of Optoelectronics and Communication till 1993. He was also senior research Fellow of BARC for the period 1985-1987. For the last 15 years, he is associated with Goa University Electronics Program. He is the founding head of University Instrumentation Center of Goa University. He is also coordinator of DEITI (an educational broadcast studio supported by Indian Space Research). His other commitments are regulating digitization Center at Goa University to support the various Digital repository projects like DIGITAP (Digital Repository for Fighter Aircrafts of Indian Navy) Million Book project of Ministry of Information Technology, New Delhi and Antarctica Study Center (NCAOR), Govt.of India. He has to his credit around 50 odd research

papers published in National and International Journals and has presented research works at various National and International Forums. He has delivered several keynote addresses and invited talks at various institutes and also authored two books on Embedded Systems published by Springer (Holland). He is a member of Goa State Rural Development Authority, member of advisory board, Goa Police Cyber Crimes and also advisor for Directorate of Technical Education. He was the chairman of Goa University Technical Advisory Committee. Presently he is head of Electronics department at Goa University.