International Journal of Digital Information and Wireless Communications (IJDIWC) 9(1): 8-21.
The Society of Digital Information and Wireless Communications, 2019 ISSN: 2225-658X (Online); ISSN 2412-6551 (Print)

Improvement of Mitigation Techniques against SYN Flood Attack for Free and Open Source Firewalls

[1]Kosmas Kapis , [2]Davis Nico Kyando,
[1,2]*University of Dar es Salaam, Dar es Salaam, Tanzania*
[1]**E-mail**: kkapis@gmail.com
[2]***E-mail:***davisky09@gmail.com

ABSTRACT

Firewalls are very potential solutions to network security. Cyber-attacks have been seriously impacting technology operations and security applications which support many business processes and protect valuable information assets. Denial of service attacks specifically SYN flood attacks are highly rated to jeopardize network security by tampering with network applications like firewalls. Existing firewalls solutions are not secure enough to protect the network against SYN flood attacks. This paper reassesses Firewall's SYN Flood mitigation techniques performance and presents an improvement model for minimizing the severity of the SYN flood attacks. The proposed model explores dynamic tweaking of TCP open timer, adjustment of firewall state table size and management of data size when the attack is detected. Experiment results of the model shows that CPU usage during SYN flood attacks hits 100% but when the improvement model is applied the CPU usage decreases by 24.6 % which is an improvement of CPU usage which is one of firewall's key resources.

KEYWORDS

SYN flood attack, CPU usage, half open connections.

1. INTRODUCTION

Business processes play an important role in meeting the organization objectives. In this era of science of technology most of the businesses have adopted technology into business processes to enhance productivity, efficiency, competitive advantage and scalability. The rate of adopting technology into businesses is growing at the high rate from small to large scale businesses. Businesses have been adopting technologies in different aspects of business units for instance communication, transaction processing and data storage. Despite of the valuable contribution of technology to the growth of businesses, there are serious challenges posed by security threats in every aspect of information flow.

In the last nine years more, than 7.1 billion identities have been exposed in data breaches according to report [1] . Security threats challenges are also growing in size and rate of occurrence which poses risk to the survivability of businesses according to [2] [3].

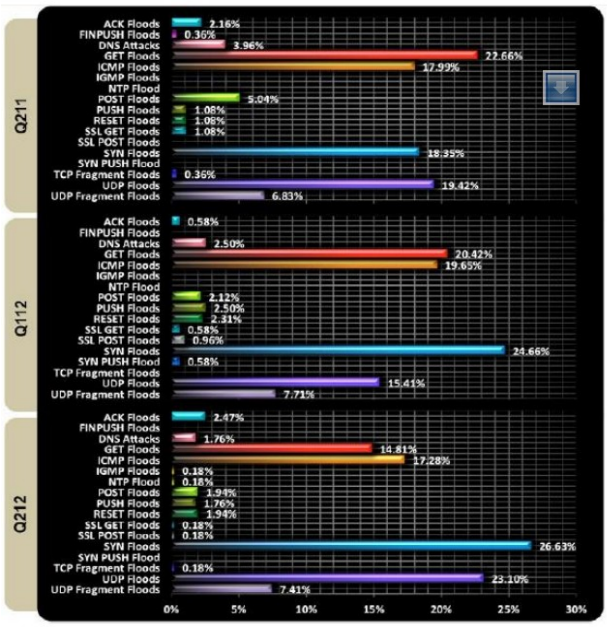

Figure 1: DDoS Attacks statics **Source:** [4]

Most of external security threats emerge from the network level. Any network with vulnerabilities is at very high risk of facing serious attacks. There is a lot of initiatives and researches to enhance security at the network level. Following that, different firewalls technologies have been emerging as hardware based and software based to offer protection. Firewall solutions are categorized as open sources distribution or commercial and

proprietary distribution [5]. An approach to use firewall as defense mechanism against SYN flood attacks is an intermediate defense mechanism. The main problems in source-end and victim-end defense involve challenge in accurately detecting the source and preventing high percentage of bandwidth consumption by the attack. To overcome this problem, the intermediate network defense mechanism optimizes both of these problem features and makes careful trade offs [6].

Firewalls themselves can be subjected into attacks from external and untrusted networks. One of the serious attacks through the firewall is a form of DDOS attacks called SYN Flood Attacks [7] and [8]. SYN floods remain one of the most powerful and popular flooding method as shown in figure 1 and figure 2. Firewall SYN flood attacks mitigation techniques are still questionable. There is still a need to improve firewalls mechanisms to handle SYN flood attacks.

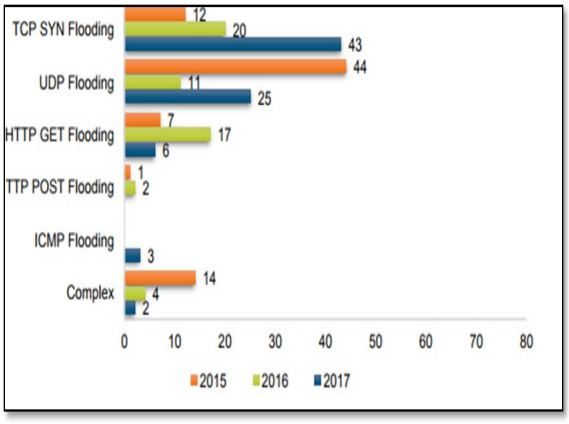

Figure 2: Comparison by attack type in Q2 of each year **Source:[9]**

SYN Flood Attack is a failure in completing a three way handshake as far as the generic Transmission Control Protocol (TCP) mechanism is concerned. When one side of the TCP connection is disabled can results into one or more of endpoint nodes becoming unable to accept connections, nodes may crash and authorization between nodes may be impaired. The SYN flood attack exploits a vulnerability of the TCP three-way handshake, namely, that a server needs to allocate a large data structure for any incoming SYN packet regardless of its authenticity [10]. This involves the stream of SYN flood attack packets directed to a specific port which runs a specific service. Mechanisms to defend against SYN flood attack [11] need to be included during the firewall design.

In recent years, there has been a stronger demand for analyzing the network firewalls performance when subjected to attacks through Distributed Denial of Service [12]. There is a high demand in firewall enhancement particularly to entirely rely on analysis, modeling and network firewalls performance simulation for the sake of predicting how efficient the firewall in network is, in terms of the effectiveness and efficiency under the attacks of DDoS [13].

Existing firewall technologies have their own weaknesses in design of architecture, configuration management, and traffic control that affect to firewall performance especially when the firewall is subjected to attacks. Researchers have proposed several aspects of the firewall that need to be considered for improvement specifically, on the firewall's ability to withstand attacks while reducing high rate of packets loss. It is known that, when the firewall encounters attacks, specifically SYN flood attacks, its performance and availability decreases and also its quality of packet filtering deteriorates. The existing SYN flood mitigation methods perform detection and prevention tasks at either the expense of significant resource usage or at the expense of sacrificing some connections. Firewalls are instrumental in defense due to high speed increasing perimeter of the networked devices at the global level. Unfortunately the technology and applications revolutions challenge the effectiveness of firewalls in case of attacks and finally firewalls fall into failures due in adequate attacks mitigations.

This paper attempts to solve the challenge on the effectiveness of firewalls by coming up with improved SYN flood attacks mitigation techniques of the firewall that focuses on free and open source solutions. The paper addresses three questions. The first question had to do with analyzing performance of mitigation techniques that are applied in existing firewalls against SYN Flood attacks. The second question looks at how to design improved mitigation techniques against SYN Flood attacks, and the last question is about evaluating performance the of newly proposed mitigation technique relatively to the existing.

2.0. LITERATURE REVIEW

2.1 Analysis of SYN flood attacks

SYN flood attacks normally explore the weaknesses of the TCP three-way handshake mechanism. The TCP three way handshake anatomy involves creation and tear-down of the connection in a transmission control protocol. This requires the client and the server to exchange the SYN and ACK (acknowledge) prior to data transfer process as illustrated in figure 3.

The TCP, a connection oriented protocol, uses three way handshake in the following fashion, When the client wants to connect to server node to transfer data, it does this by sending a packet to server with the SYN (or synchronization) flag set, implying asking for a connection server converse back to client. The client responds with SYN and ACK flags set, indicating its readiness to take part in conversation. Finally, client returns the third part of the handshake, a packet with only the ACK flag set, confirming to take part in conversation too and immediately data transfer commences as [14]. Figure 3 illustrates the processes.

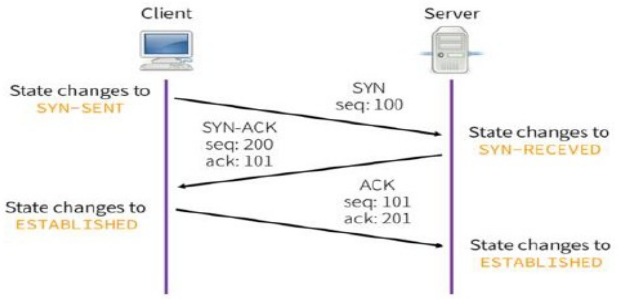

Figure 3. Complete and successful TCP three way handshake.

SYN flood attack is caused by a failure to complete TPC three way handshakes. Figure 4 illustrates the scenario in which a 3-way handshake is violated where the SYN request is sent to establish connection and the target responds by sending SYN/ACK flag. To complete the 3-way handshake the source should send ACK to initiate data transfer, but ACK is not sent, instead the source keeps sending many SYN to generate a lot of half open connections which may lead to the crash of the target for the attack to be successfully. The TCP SYN flood attack depends on the SYN request packets and the weakness of three-way handshake mechanism that allows half open connections [15].

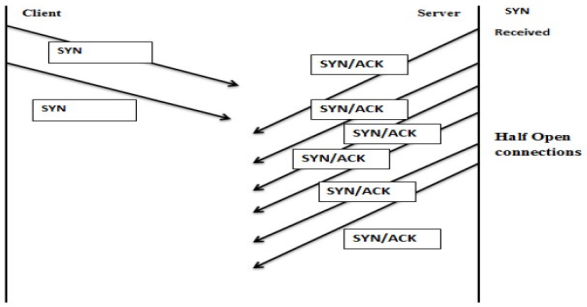

Figure 4. Three way Handshake violations resulting SYN flood Attacks.

2.1 Software Based Approach in Firewall Development

Firewalls are security appliances used to enforce security from the edge of the network. Firewall uses filtering methods to isolate the trusted traffic and untrusted traffic. Various methods of filtering can be used on implementing firewalls based on seven OSI layers ,commonly used methods operates on application, transport, and network, and data-link levels [16].

Most of the firewall systems are hardware based. Hardware platform has got its limitations compared to software based approach of firewall implementation. Shortcomings of hardware based firewalls includes being very expensive and most of the hardware based firewall requires extensive configuration procedures. Network administrators should be comprehensively trained to handle configuration and maintenance procedures. Moreover configuration from each specific vendor is different and hence the knowledge of one type of firewall may not be applicable to another type of firewall. With all these limitations it is reasonable to adopt software based firewall.

Microsoft has introduced a new platform named windows filtering platform as an architecture implemented from windows vista and windows server 2008. This platform has special features such as filtering and modifying TCP/IP packets which provides unprecedented access to the TCP/IP packet processing path where one can examine or modify outgoing and incoming packets before additional processing occurs. One can more easily create firewalls, antivirus software, diagnostic software, and other types of applications and services by accessing the TCP/IP processing path at different layers [17]. Windows Filtering Platform (set of API and system services which enables to create filtering applications) provide higher flexibility in performance and diagnostics support. However, this is not an open-source firewall.

2.2 Software based Firewall design logic

The logic behind firewall design is a series of actions that manage the flow of inbound and outbound data traffic. The series of actions that has to manage the flow of data traffic are represented as an algorithm as shown in figure 5 which is the basis of firewall software development. Generally the flow pattern of firewall algorithm will contain at least six steps.

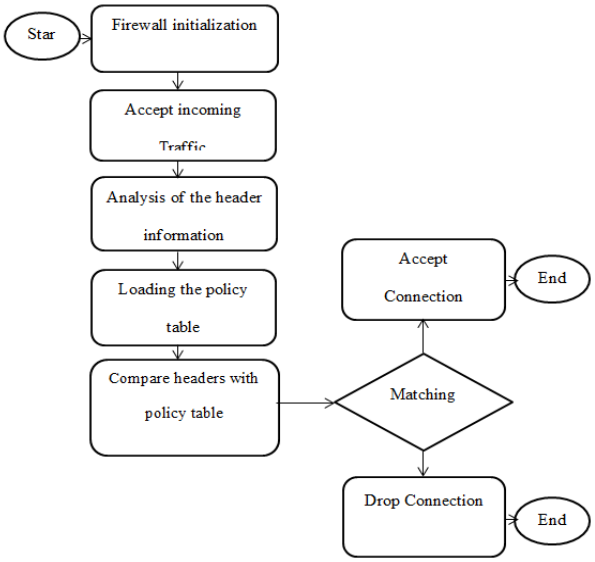

Figure 5. Firewall Logic Flow.

2.3 Strength of Open Source Firewall Solutions

An open source firewall is the one which available freely and can be used by anyone and anyone can modify the source code and even find bugs and report them [18]. Open sources are rich in feature because of its software development methodology. All open-source applications are licensed by an open-source license, which gives the user the right to use the software, access and modify the source-code, and redistribute the software for free. This type of software is very popular due to its many advantages, since it promises to accelerate the diffusion of Information Technologies (IT) solutions in healthcare. Thereby, it can contribute to reduced development costs based on study by [19],[20]. Generally speaking, the strength of open source lies in its: no license costs, interoperability, easier integration and customization, compliance with open technology and data standards and freedom from vendor lock in. Studies have shown that the benefits of open source generally materialize in the medium to long term. Furthermore, because open source software is free, there is greater flexibility in selecting the level of services or support that a customer wants to pay for, if at all [21].There is growing demand of the open source solutions, the trend is shown in the **figure 6.**

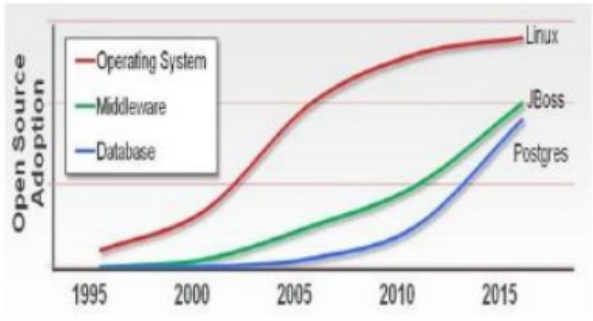

Figure 6: The expected growth of OSS from 1995-2015. Source: (Red hat summit 2009).

2.4 Stateful Firewall Systems

Stateful firewall always keeps track closely to every connection which passes through it from end to end. The firewalls are always of all connection paths and are normally capable of implementing the IPSEC components like tunneling and encryption. With stateful firewall it is possible to know the status of the connection at any state whether it is (open, synchronized, open sent, synchronization acknowledge or established). Stateful firewalls faces the challenge of being slower compared to stateless firewalls. Stateful firewall maintains the table of states for every connection passing through it, the size of connection state table has an impact to the performance of the firewall when it is subject to voluminous amount of connections at once.

2.5 Firewall Performance Metrics

Performance metrics are benchmarks or instruments used for gauging the performance of the firewall and according to literature review, firewall performance metrics are proposed based on the specific scenario. Generally performance metrics proposed in previous researches include throughput rate, CPU usage, and number of concurrent connections per second, HTTP transfer rate according to IETF. In this case, the study focused on two metric, namely, the CPU usage and number of half open connections which closely explain impact of SYN flood attacks. The two metrics were chose based on their importance, suitability and the nature of the study.

2.5.1 Impact of Firewall CPU Usage During the Attacks

CPU is one of important resources in a computing device. The CPU processes the instructions it

receive when data from TCP traffic is decoded. During the SYN flood attacks the firewall is overwhelmed by SYN packet at very significant voluminous rate. Hence CPU experiences extremely many instructions to process per second. Automatically the CPU usage will shoot up at the expense of handling high volume of incoming SYN flood traffic. A study by [22] identified the impact of DDoS attacks on Usage and proposed a solution to reduce. **Figure 7** illustrate the impact of SYN flood attacks on CPU usage.

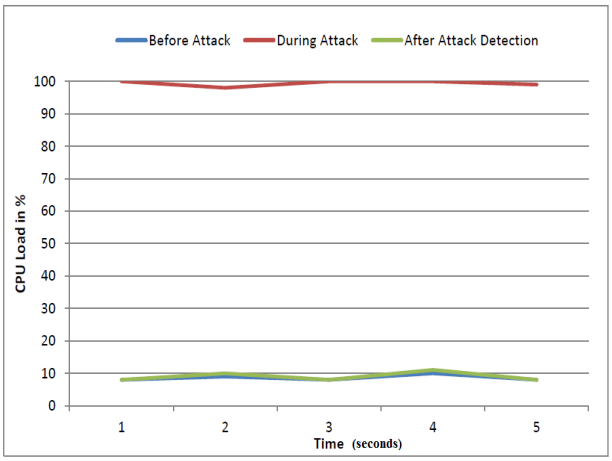

Figure 7: CPU load for TCP SYN flood attacks. **Source: ([22])**

2.5.2 Impact of SYN Flood Attacks on Connection State Table

Any stateful firewall keeps track of all the connections which are established within the firewall. If an attacker floods a server with SYN packets, that server may reach its TCP connection limit and begin refusing legitimate connections. The attacker simply sends SYN packets, but never acknowledges the server's SYN-ACK packets. The server waits for up to 60 seconds. During this period, the attacker can send SYNs much faster than the server can time them out [23].

3 .0 Approach

A quantitative research design approach has been used while experimental and explorative method was used to collect data. The data was collected through experiment on the existing SYN mitigation solutions against SYN flood attacks on the firewall and later on from the enhanced version of the mitigation techniques.

3.1 Research Design

Following the identified SYN flood mitigation improvement factors the proposed model has been developed. Data were collected quantitatively using experimental method using series of performance tests through experiment against SYN Flood Attacks on target firewall system. Finally explorative method was be used to comprehend and analyze the current free and open source system protection implementations against SYN flood attacks. This method was also used to understand various approaches on how to mitigate SYN flood attacks using Free and open Source firewalls is implemented. This method helped to generate qualitative information.

3.2 Data Collection Methods

The methodology involved exploring different literature review and identifies the improvement factors which were used to design the improvement model. Based on the identified improvement factors an experiment environment was prepared and series experiments were conducted to study the behavior of the firewall when it is subjected to SYN flood attacks.

The targeted firewall solution was subjected to simulated SYN flood attacks, where the hping3 contained in kali linux tool was used to simulate SYN flood attacks and results were documented and analyzed. The performance parameters against SYN flood attacks were number of half-closed states and CPU usage behaviors which are important firewall resources which are potentially get affected.

The proposed improvement factors were varied while the SYN flood was in action. This included variation of TCP open timer, data size and state table. Data were collected by using different values of TCP open timer which is one of important factor to improve firewall ability against SYN flood attacks. TCP open timer was set to 30s, 20s, 10s and finally 1s then the value for CPU usage and number of half open connections were recorded at every 30 seconds using stop watch.

Moreover state table size, one of vital resource, value was set to 10%, 20%, 40% and 90% of the system RAM size CPU usage and Number of half open connections were recorded for every 150 seconds. Finally the data size of the packets

were altered starting from the small value of 64 bytes up to almost 19,000 bytes and the results of CPU usage and number of half open connections were recorded.

3.3 Data Analysis Method

Data were collected from SYN flood attack tests performance results, sorted and well arranged in a tabular format and also data has been presented in graphs. These data were analyzed using quantitative data analysis methods between the existing free and open firewall systems and the improved version of the free and open firewall system as per proposed improvement model. The proposed model approach is expected to perform better against SYN flood attacks.

4. Firewall SYN Flood Mitigation Model

The model was developed to cater enhancement and re-powering the firewall ability to withstand during the SYN flood attacks. This is because most of the existing methods still need improvement and many firewall solutions have shown weakness in handling the SYN flood attacks. In this paper, factors which can contribute to the improvement of .firewalls mitigation technique against SYN flood attacks have been identified. The proposed model namely dynamic Adjustment of TCP open timer value, detection scheme using SYN-ACK pairs counter, dynamic adjustment the firewall state table and tweaking the data size is depicted in **figure 9**.

4.1.1 Dynamic Adjustment of TCP Open Timer Value

TCP open timer is an important parameter that is hard coded into the firewall system source code which determines how long it takes to establish a TCP connection. set when a SYN is transmitted, and aborts the connection if no response is received within specified seconds[24].Whenever this value is lowered it show significant improvement on SYN flood defense but in other side it may lead to failure in keeping states. TCP open timer management re-powers the firewall to change this value dynamically when the SYN flood attacks have been detected. In this regard the value for the timer is not remaining fixed as hard-coded.

4.1.2 Detection Scheme Using SYN-ACK Pairs Counter

Detection of SYN flood attacks starts prior to defensive actions. The detection of SYN flood attacks has been implemented in different ways. This study proposed detection mechanism based on the SYN-ACK pairs [25]. This scheme regards SYN and SYN/ACK as valid SYN packets while FIN and RST as valid FIN packets. In case there are more valid SYN packets that valid FIN packets then the firewall are in the state of SYN flood Attacks. The detection using this concept is more efficient against SYN flood attacks. The statistical correlation of SYN and ACK confirm the presence of SYN flood attack when it strongly positive.

To capture TCP SYN flagged packet tcpdump utility was used by running the command tcpdump -i <interface=NIC 2> "tcp[tcpflags] & (tcp-syn) != 0" and to capture the TCP ACK packets the command tcpdump -i <interface=NIC 2> "tcp[tcpflags] & (tcp-ack) != 0".During the SYN flood attacks the ratio of TCP SYN packets /TCP ACK packets is greater than 1 and when there is no SYN flood attacks the ratio is equal to 1.In this study detection is effective using this technique where any ratio violation will signal a SYN flood attack presence. This technique does spend less of CPU resources and hence ensures efficiency in detection.

4.1.3 Dynamic Adjustment the Firewall State Table

State table is one of the very precious resource to the firewall and its careful management results into improved firewall performance, a study by [26] extended the firewall session table to accelerate NAT, QoS classification, and routing processing time while providing the same level of security protection.

This study explored a technique of automatically adjust/resize the firewall state table during the SYN flood attacks. Upon detecting the attack system state table is resized and optimized to handle more connections. This ultimately capacitate a firewall ability to widen the room to handle more connections in the event of flood

attacks and relieved out of any struggling efforts which normally will make the firewall CPU higher and higher which will finally affect the performance of the firewall

4.1.4 Tweaking the Data Size

Data size of the packet sent during the SYN flood attacks has proved to be one of factor that impact of performance of the firewall during the SYN flood attacks. When the SYN flood is generated with the small sized data size of impact of the SYN flood is expected to be very high in such a way that the state table as well as CPU utilization goes extremely high. In reverse when the data size of large in size the impact of the SYN flood attacks is lessen ,that is, the state table utilization is so much minimized while also the CPU usage is getting significantly lowered.

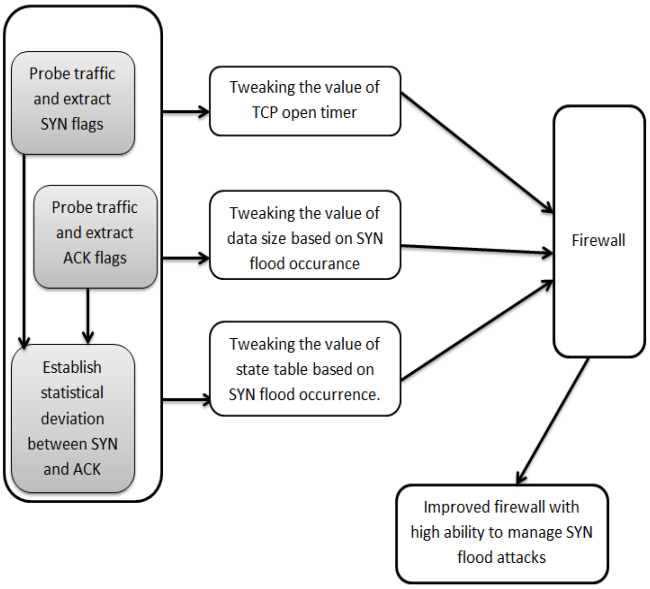

Figure 9: Proposed firewall improvement model.

Algorithm for mitigation of SYN flood attack was designed based on detection and preventive techniques. The detection technique was designed based on statistical deviation of SYN and ACK contained in the traffic. Signaling of SYN flood attacks is raised when there is SYN which has not been acknowledged. The preventive techniques capitalizes adjustment of TCP open timer which has proved to reduce the effect of SYN flood attack when its value is set to be small, though it has an adverse impact which results to connection losses. This design aim to dynamically optimize the TCP open timer value and reduce connection losses reduce, CPU load of the firewall, lowering number of half open connections hence improve firewall performance. Moreover the proposed solution capitalizes the use of state table management adjustment to counter effect of the SYN flood attacks to overwhelm the state table within a short time. Mitigation of SYN flood attack in this study has also capitalized the use of data size management of the SYN flood attack. Enlargement of data size potentially reduced the effect of the SYN flood attack which lowered the CPU usage and number of half open connection that will be generated. Few numbers of half open connections will delay the state table to its full capacity during the attack. The algorithm is combining all the proposed factors of improvement that can be taken into consideration during the firewall design. The algorithm is divided into sequential processes as shown in **figure 10** the process flow depicts detection and defense mechanism. Detection is followed by actions to defend against the SYN attacks. The actions to defend are dynamic in nature which makes an intelligent firewall by design and the improved firewall will be adaptive and efficient against SYN flood attacks.

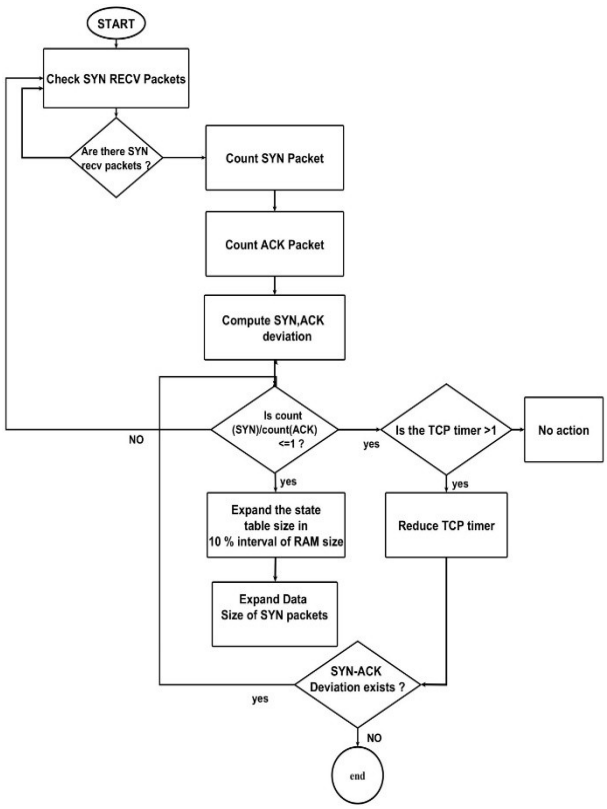

Figure 10. Detection and defence process proposed mitigation technique.

4.2.0 The Experiment

i. Experiment Setup

Environment for the experiment was a given firewall system against security tests as shown in figure 11. A dedicated server will initiate SYN Flood Attack from untrusted network and the reaction of each firewall system will be observed, measurement of CPU utilization, number of half-closed states metrics will be recorded. The attack of specific size will be for a considerable amount of time.

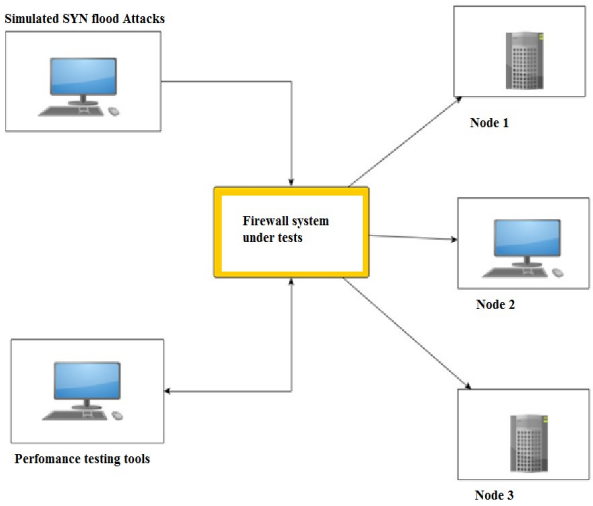

Figure 11: SYN flood Attack Setup Environment.

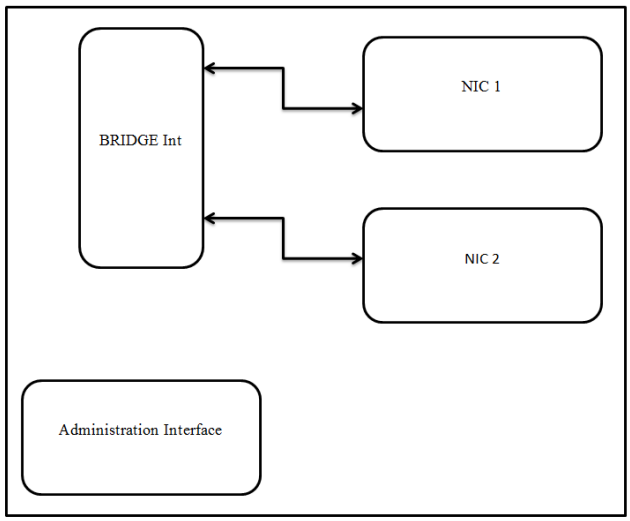

Figure 12. Computer box layout for firewall design.

Figure 12 represents the computer designed as a firewall appliance using free and open source pfsense software solution. The computer box should contain at least two network interface card. One network card is used as a LAN gateway interface as named NIC 1. Second network card is used as WAN interface as named NIC 2 which the external traffic can pass through into the firewall before landing into the LAN network. Also NIC 1 is used as a gateway interface for the traffic leaving the firewall. Bridge interface is used to create the logical link between two or more Ethernet interfaces.

ii. Experiment Summary

With respect of the experiment setup above the firewall under test was fired by voluminous amount of SYN flood packets as simulated attacks. Hping tool was used to generate the SYN flood packets toward the firewall to be tested. During the moment of attacks the firewall behavior was being recorded based on the CPU utilization, number of half-closed states metrics with the default values of tcp open timer, state table and data size.

The experiment was repeated with different values of the tcp open timer, state table and data size and the behavior of the firewall was recorded based on the factors discussed above.

4.2.1 Experiment Procedures

Experiment procedures were summarized as shown in the figure 13. To generate the attacks SYN flood attacks traffic were simulated using hping3 towards the firewall and observe its impact on CPU usage, number of half open connections and record results in a table using command *hping3 -V -c 1000000 -d 64 -S -w 64 -p 445 -s 445 --flood --rand-sources (Victim IP)*. And this command generated voluminous amount of SYN flagged packets to the firewall's Victim IP.

Below are the experiment procedures which show step by step how to conduct tests, figure 13 show the summary of the steps.

1. Setting up experiment as shown in figure 11
2. Execute wireshark on both machine to monitor traffic inbound and outbound traffic
3. Observe the inbound and outbound traffic in the firewall when firewall is not under attacks and record.
4. Simulate the SYN flood attack traffic using Hping3 towards the firewall and observe its impact on CPU usage, number of half open connections and record results in a table using command hping3 -V -c 1000000 -d 64

-S -w 64 -p 445 -s 445 --flood --rand-source (Victim IP).

5. Repeat step 4 when the TCP open timer is set to 30s, 20s, 10s,1s.

6. Record in tabular form CPU usage, number of half open connections when connection state table size to 20%,40%, 60% and 90% a of the size of the RAM.

7. Maintain the state table at 10% of RAM size and keep varying the data size from small values say 64 bytes to 15,000 bytes of the SYN flood attacks packets and record the CPU usage and number of half open connections and records the results in a table.

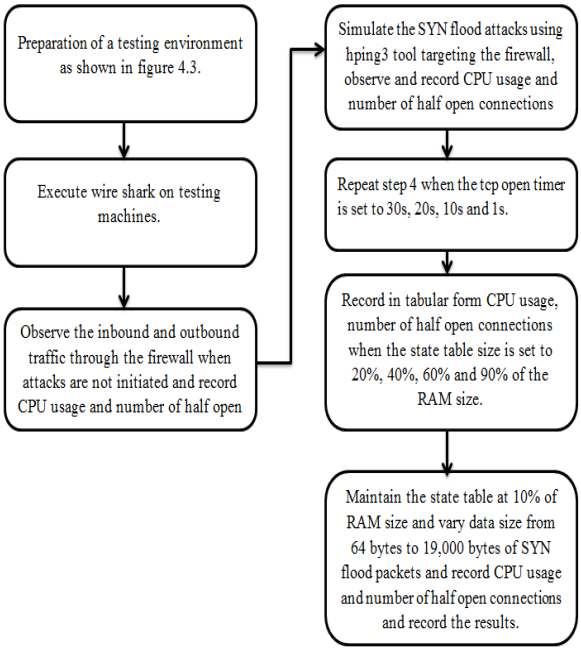

Figure 13.Summary of the testing procedures.

5.Results

Results are based on the set of data collected using series of experiments and a detailed analysis of system performance against SYN flood attacks from the obtained data, before and after the implementation of factors of improvement. These factors are taken into account in the proposed model.

To obtain the data, the SYN flood attacks were simulated targeting the firewall under test in randomized fashion through a WAN port of the firewall appliance. Timer was set and half open connection and CPU usage were observed during the SYN flood attack in an interval of 30 seconds. During this testing the value of TCP open timer were set to 30s and SYN flood with random source were generated using Hping3 and the observation showed that the firewall state table went completely filled with half open connections and CPU usage went closely to 100% over time as show in table 4 and figure 14. In the case where the TCP open timer has been reduced to 1s the number of half open connections and CPU usage went down as shown in figure 17.

Another factor for improvement that was tested was data size and table 5 is shows results collected when the data size of packet is varied. The CPU usage and number of half open connections were collected at different values of data size starting with 64 bytes to 19000 bytes. Data size values from 64 bytes to 960 bytes the impact of SYN flood attacks is high due to the generated number of half open connections and CPU usage values. As shown on table 5 where the data size values start from 5120 bytes to around 19000 bytes as collected, the number of half open connections and CPU usage shows to shoot down significantly, the impact is also illustrated figure 17.

State table size was another factor of improvement which this study considered. When the value of state table size was expanded it proved to add firewall's capability to withstand efficiently during the SYN flood attacks. As shown in the tables 2 and 3 the capacity of firewall to hold more connection states is possible when the state table size is adjusted close to the value of RAM size. The data were collected when the state table size is at 10% size of the RAM later at 20%,50% and 90% .The improvement against SYN flood attacks were significant and it can be observed that the room for storing more connection is expanded and the CPU usage significantly appears to shoot down as supported by the results.

Table 1. The record of CPU usage, number of half open connections generated and data size.

State Table size 394000		
data size(bytes)	CPU usage (%)	No of half Open Connections
64	69	394,008
128	72	394,008
192	81	394,008
256	81	394,008
320	79	370,068
384	77	366,163
448	86	376,769
512	70	366,163
576	81	365,269
640	79	376,161
704	83	362,262
768	70	377,063
832	68	366,163
896	61	301,595
960	61	310,296
1024	17	301,295

Table 2: Table size in relation to CPU usage, half open connections at 10% and 20% size of the RAM.

State Table size as % of RAM size					
10%			**20%**		
Time (s)	CPU usage (%)	Number of Half Open connections	Time(s)	CPU usage (%)	Number of Half Open connections
0	4	246	0	5	321
150	70	390,001	150	79	764,123
300	110	402,231	300	106	783,983

Table 3. Record of table size, CPU usage, half open connections at 50% and 90% size of the RAM.

State Table size as % of RAM size					
50%			**90%**		
Time (s)	CPU usage (%)	Number of Half Open connections	Time(s)	CPU usage (%)	Number of Half Open connections
0	7	356	2	10	196
150	80	1,970,000	150	78	1,994,003
300	77	1,981,389	300	76	1,997,113

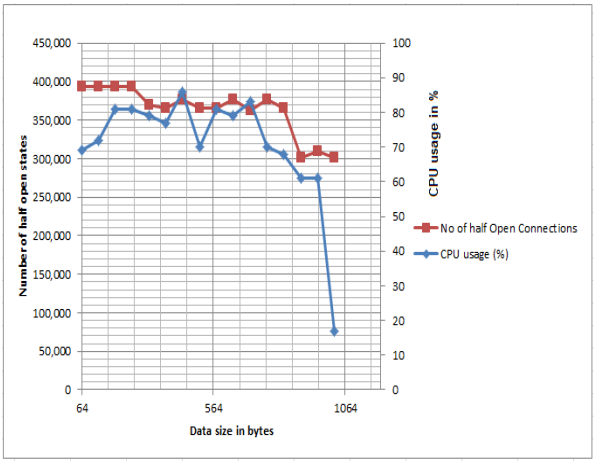

Figure 14. Variations of CPU usage and number of half Open connections.

5.1 Comparative Analysis

From the data collected in first scenario during the SYN flood attacks, it showed that the firewall was getting overwhelmed with voluminous amount of half open connections and high utilization of the CPU. This was possible within a short time after the attacks is launched targeting the firewall system.

Table 1 represents tabular results during SYN flood attack on a firewall system and figure 1 is indicates impact of SYN flood attacks to CPU usage where it hikes close to 100% and the state table is completely exhausted to maximum capacity which is 400k connection states. This signals for a negative impact to the firewall as far as its performance is concerned. At this moment is firewall is unable to handle it normal operations.

Next scenario is when data were collected after taking into effect the factors of improvement which includes data size management, state table management, tcp open timer. In this case when the tcp open timer was reduced toward 1 the number of half open connections appeared to decrease in number and also CPU usage too and this is shown from Table 4 and based on this experiment the number of half open connections generated has gone down to around 300k.

The number of half connections and CPU usage appeared to lower down when the factor of data size was varied to higher values. Table 2 indicates the variation of data size with respect to the generated number of half open connections and CPU usage. CPU usage with large data size has shown to lower down to an average of 10-25% during the attack.

The identified factors indicate that when the firewall is improved to properly manage these resources then, its ability to defend against SYN flood attacks is improved. This clearly shown significant difference with the first scenario where the firewall is operating with its default resource management.

Table 4. Performance Parameters results during SYN flood Attack for various parameters set.

TCP TIMER=30s			TCP TIMER=20s			TCP TIMER=10s			TCP TIMER=1s		
Time (s)	Half Open connections (X1000)	CPU (%)	Time (s)	Half Open connections (X1000)	CPU (%)	Time(s)	Half Open connections (X1000)	CPU %	Time (s)	half Open connections (X1000)	CPU (%)
0	65	12	0	110	7	0	102	21	0	26	19
30	390	55	30	390	17	30	390	36	30	370	39
60	390	61	60	390	60	60	390	59	60	365	62
90	390	64	90	391	62	90	390	71	90	356	63
120	390	69	120	390	62	120	390	79	120	370	64
180	392	80	180	392	64	180	390	81	180	330	81
210	399	90	210	399	68	210	390	72	210	358	81
240	400	100	240	400	70	240	390	75	240	360	76
270	399	97	270	400	83	270	390	74	270	374	75
300	399	98	300	392	79	300	390	74	300	378	78

Table 5. The CPU usage, number of half open connections for different data size

State Table size 394000		
Data size(bytes)	CPU Usage (%)	No of Half Open Connections
5120	15	70,664
6144	11	66,586
7168	8	63,434
8192	13	63,040
9216	8	51,858
10240	8	35,840
11264	9	35,532
12288	10	34,832
13312	6	34,438
14336	5	34,936
15360	6	33,336
16384	7	33,636
17408	7	32,946
19456	11	32,811

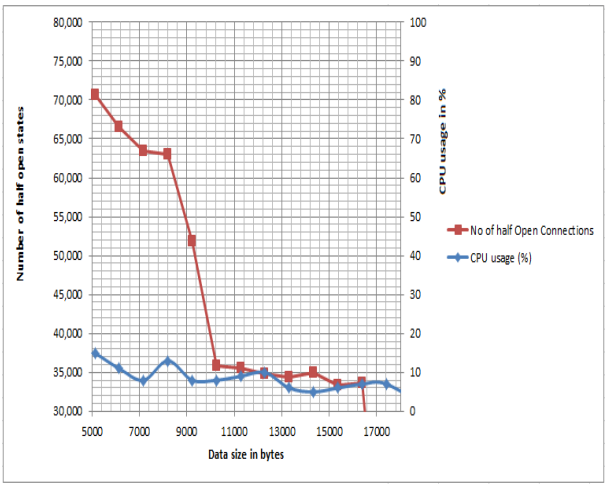

Figure 15:Data size, CPU usage and half open connections

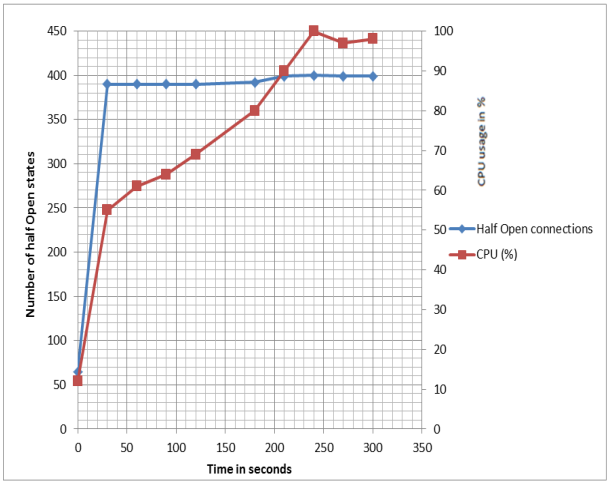

Figure 16. Number of half open connections generated against time when TCP Timer=30s during the SYN flood attacks.

A tabular result from table 2 and table 3 shows the impact of the data size towards the firewall during the attacks. The study observed that when the data size sent is small the effect of the attack is becoming serious and we can observe that the depletion is state table is up to 100% shown in appendix G while the CPU usage is very high close to 100% as shown in figure 15, in this situation the firewall is not able to receive new connections. The CPU usage is normally not recommended for the observed records. Moreover table 5 results show that, as data size increases the impact of the SYN flood attack to the firewall is getting minimized as shown in figure 16 which represents this trend. Appendix H is showing the impact of the large data size on firewall performance.

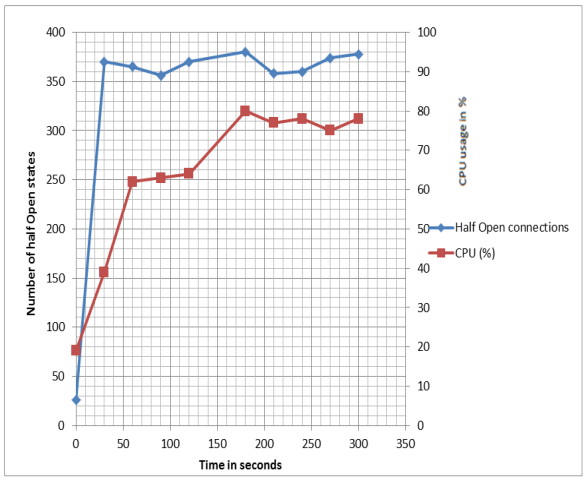

Figure 17. CPU utilization of the firewall against the time when TCP Timer=1s during SYN flood attacks.

6.Discussion

In this study a firewall mitigation technique improvement model has been proposed as shown in figure 9. Factors of improvement were identified from literature review. The factors identified were; adjustment of TCP open timer, state table management and data size management. The proposed improvement model was developed based on these factors. These improvement model factors were analyzed and was found that they have significant effect to the protection against SYN flood attacks. The proposed management of these factors potentially resulted in low CPU usage and reduced number of half open connections generated during SYN flood attacks.

Specifically, this study proposed the dynamic adjustment of the TCP open timer to automatically reduce the impact SYN flood attacks when the attack is detected. The automatic adjustment of TCP open timer has proved to reduce the impact of SYN flood attack by rescuing significant losses of legitimate connection since only the required value is set when there is an attack, and when there is no attack the required default value is set. This is not the case when TCP open timer is manually adjusted.

Furthermore, in this study, it has been found that the size of state table is a very important factor in reducing the impact of the SYN flood attacks. When size of the state table is small the table gets filled up within a very short time and firewall fails to take up more new connections. By default, the optimal value set is always 10% of system RAM based on the case study firewall solution presented in this paper. In this case, since the firewall is set to have 4GB then default value of state table is 384,000 states.

The study proposes immediate automatic resizing of the state table when the SYN flood attack has been detected. This paralyzes the immediate impact of the flood attacks and lengthens its performing working condition rather that proving failure in a short time due to limited state table size. Moreover results confirmed that when the data size of the SYN flood attack packets decreases in size the impact of the SYN flood attacks is becomes very serious in such a way that the firewall gets devastated and fails within a very short time. The larger the data size of the SYN flood packet the less the impact of SYN flood attack. This finding can be used as one of improvement factor on the firewall design. The management of data size after the attack is detected can be incorporated during the design by including other model components as explained by the proposed model.

This study has contributed another improvement to a SYN flood mitigation mechanisms of a firewall by incorporating it with the ability to defend against SYN flood attacks. During the experiment on firewalls with its existing SYN flood mitigation implementations it has been observed that a firewall ability to defend against the attacks is not adequate as compared to its importance being situated at the gateway of the network. At any moment the firewall performs poorly, the whole network performance is seriously affected. Inbound as well as Outbound Traffic will not cross the gateway when the firewall is under SYN flood attacks.

Attacks were simulated using Hping3 tool running on Kali Linux Platform towards the firewall. Table 4 presents the values of CPU usage, Number of state generated, and time. The data on graph in figure 15 explains the impact of SYN flood attacks on the firewall. This study shows that there is significant increase of the firewall CPU usage and Number of Half open connections generated when SYN flood attack is in action. In duration of 1-2 minutes the firewall connection table gets completely filled and the CPU usage approaches 100%. When The Firewall is under this situation its capability as a gateway device to secure the whole network is completely distorted and finally becomes unable to respond on the request within a short time.

7.Conclusion

The study proposed, developed and evaluated the improvement model through proposed improvement factors which build up the model. The important improvement factors were identified and used in this model and after the thorough analysis all the factors found to have

significant contribution in improving the firewall ability to defend against the SYN flood attack.

In this study results showed that CPU usage during SYN flood attacks hits 100% but when the improvement model is applied the CPU usage shoots down by 24.6 % which is an improvement results focusing on CPU usage which is as one of firewall ‘s key resources. Moreover results showed number of half open connections generated without application of the model hiked to maximum of state table size which is 394k based on our experiment conditions during attacks. With the proposed improvement model the maximum number of half open connections were significantly reduced when TCP open timer is between 1s-10s and the data size is between 64 bytes and 1024 bytes to the average of 345k which is 12 % improvement gained. In this regard the results suggest that improvement model have shown an advanced fashion on handling of SYN flood attacks than previous ones.

The designed model is useful by developers who design or redesign the firewall architectures. The study emphasizes the use of improvement model and the proposed algorithm to manage the improvement factors in an intelligent fashion. The analysis of these model attributes helps to provide information to firewall designers so that these factors can be well managed to design high efficient firewalls when it comes to SYN flood attacks management.

REFERENCES

[1] Symantec, “Internet Security Threat Report,” 2017.

[2] W. Fuertes, P. Zambrano, M. Sánchez, M. Santillán, C. Villacís, and E. Torres, “Repowering an Open Source Firewall Based on a Quantitative Evaluation,” vol. 14, no. 11, pp. 118–125, 2014.

[3] D. Emm, M. Garnaerva, D. Makrushin, and A. Ivanov, “IT Threat Evolution: Q3 2015,” 2015.

[4] M. Bogdanoski, S. Tomislav, and R. Aleksandar, “Analysis of the SYN Flood DoS Attack,” *I. J. Comput. Netw. Inf. Secur.*, vol. 8, no. June, pp. 1–11, 2013.

[5] O. R. Ehimen and I. Oyakhilome, “Development of a Software Based Firewall System for Computer Network Traffic Control,” *Leonardo Electron. J. Pract. Technol.*, no. 15, pp. 75–80, 2009.

[6] T. M. Anwar, K. Saira, and M. Wani, “IJSER-DDoS SYN Flooding; Mitigation and Prevention,” *Int. J. Sci. Eng. Res.*, vol. 5, no. 12, pp. 484–490, 2014.

[7] K. Salah, K. Elbadawi, and R. Boutaba, “Performance Modeling and Analysis of,” *IEEE Trans. Netw. Serv. Manag.*, vol. 9, no. 1, pp. 12–21, 2012.

[8] D. Moore, G. M. Voelker, and S. Savage, “Inferring Internet Denial-of-Service Activity,” in *Proceedings of USENIX Security Symposium*, 2000.

[9] CDNetworks, “Q2 2017 DDoS Attack Trends Report,” 2017.

[10] S. Jaydip, “A robust mechanism for defending distributed denial of s service attacks on,” *Int. J. Netw. Secur. Its Appl.*, vol. 3, no. 2, pp. 162–179, 2011.

[11] T. ao Peng, L. Christopher, and R. Kotagiri, “Survey of Network-Based Defense Mechanisms Countering the DoS and DDoS Problems,” *ACM Comput. Surv.*, vol. 39, no. 1, pp. 22–34, 2007.

[12] L. Zhou and O. Alina, “Operational security log analytics for enterprise breach detection,” in *2016 IEEE Cybersecurity Development (SecDev)*, 2016, no. 1, pp. 1–8.

[13] J. R. Rao, S. N. Chari, D. Pendarakis, R. Sailer, W. Teiken, and A. Wespi, “Security 360 ° : Enterprise security for the cognitive era,” *IBM J. Res. Dev.*, vol. 60, no. 4, pp. 1–13, 2016.

[14] U Rani J, M. M. Nayak, and G. Anandhi, “Visualization of Three Way Handshake Mechanism of TCP / IP,” *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 5, no. V, pp. 165–169, 2017.

[15] D. M. Divakaran, H. A. Murthy, and T. A. Gonsalves, “Detection of SYN flooding attacks using linear prediction analysis,” in *Networks, 2006. ICON ’06. 14th IEEE International Conference on*, 2006, vol. 1, pp. 1–6.

[16] E. A. Anaya, M. Nakano-Miyatake, and H. M. P. Meana, “A History and Survey of Network Firewalls,” *Midwest Symp. Circuits Syst.*, vol. V, pp. 848–852, 2009.

[17] P. Poudel, “Network Access Control using Software Based Firewall System,” in *9th International Conference on Software, Knowledge, Information Management & Applications*, 2015, pp. 1–7.

[18] R. Sharma and P. Chandresh, “Firewalls : A Study and Its Classification,” *Int. J. Adv. Res. Comput. Sci.*, vol. 8, no. 4, pp. 1979–1983, 2017.

[19] B. Janamanchi, E. Katsamakas, W. Raghupathi, and W. Gao, “The State and Profile of Open Source Software Projects in health and medical informatics,” *Int. J. Med. Inform.*, vol. 78, pp. 457–472, 2009.

[20] T. Waring and P. Maddocks, “Open Source Software implementation in the UK public sector : Evidence from the field and implications for the future,” *Int. J. Inf. Manage.*, vol. 25, pp. 411–413, 2005.

[21] B. Brian, “Factors influencing open source software adoption in public sector national and

international statistical organisations Review of Current Thinking," Dublin, Ireland and Manila, Philippines, 2014.

[22] D. Kshirsagar, S. Sawant, A. Rathod, and S. Wathore, "CPU Load Analysis & Minimization for TCP SYN Flood Detection," *Procedia Comput. Sci.*, vol. 85, no. 2016, pp. 626–633, 2016.

[23] C. Roeckl, "Stateful Inspection Firewalls." Juniper Networks, Inc., Sunnyvale, pp. 1–14, 2000.

[24] K. Mansley, "Tweaking TCP's Timers," in *Engineering*, 2004, pp. 1–20.

[25] S. Gavaskar, R. Surendiran, and E. Ramaraj, "Three Counter Defense Mechanism for TCP SYN Flooding Attacks," *Int. J. Comput. Appl. (0975 – 8887)*, vol. 6, no. 6, pp. 12–15, 2010.

[26] M. Mostafa, A. Abou, E. L. Kalam, and C. Fraboul, "Extending Firewall Session Table to Accelerate NAT , QoS Classification and Routing," in *19th International Conference on Computer Theory and Applications*, 2009, pp. 40–43.