ISSN 2047-3338

Design and Development of a Framework to Mitigate DoS/DDoS Attacks Using IPtables Firewall

Koushik Chatterjee
Sir Padampat Singhania University, Udaipur, India

Abstract— **A DoS (Denial of service) attack is characterized by an explicit attempt to prevent the legitimate use of a service. These attacks overwhelm the processing or link capacity of the target sites by saturating them with bogus packets. Such attacks can seriously disrupt legitimate communications. These attacks can disrupt the availability of Internet services completely, by eating either computational or communication resources through sheer volume of packets sent from distributed locations in a coordinated manner or graceful degradation of network performance by sending attack traffic at low rate. Distributed Denial of Service (DDoS) Attacks, when an attacker attacks from multiple source systems, it is called a Distributed Denial of Service (DDoS) attack. The actual owners are usually not aware of their system being used in a DDoS attack. DDoS (Distributed Denial of Service) attacks are amplified form of DoS attacks where attackers direct hundred or even more zombie (Slave) machines against a single target. DDoS attacks have two phases: Deployment and Attack phase. DDoS program must be deployed on one or more compromised hosts before attacks are possible. The several mechanisms are available to mitigate DoS/DDoS attacks. In this paper, it is proposed a design of a framework or a mechanism for defending against Denial of Service attacks, have become one of the major threats to the operation of the Internet today. The IPtables is a Linux kernel based packet filter firewall. IPtables modules are present in the kernel itself, there is no separate daemon for it. This makes it very fast and effective firewall. The IPtables rules control the incoming and outgoing traffic on a network device. This design will be based on firewall for detecting and preventing the most harmful and difficult to detect DoS attacks. The firewall that can distinguish the attack packets (containing source addresses) from the packets sent by legitimate users, and thus filters out most of the attack packets before they reach the victim. The firewall scripts are written using command-line tool IPtables in Linux to deny the suspicious traffic. Packet sniffer tool will be used to display the effectiveness and performance of the scripts in mitigating the various kinds of DoS attacks.**

Index Terms— **DoS/DDoS, Firewall, Iptables and Packet Sniffer Tool**

I. INTRODUCTION

ACCORDING to the CIAC (Computer Incident Advisory Capability), the first DoS attacks occurred in the summer of 1999 [4]. In February 2000, one of the first major DDoS attacks was waged against Yahoo.com. This attack kept Yahoo off the Internet for about 2 hours and cost Yahoo a significant loss in advertising revenue.

Types of Attack and its impact on Network [20]:

A. *Attacks with direct communication*

During attacks with direct communication, the agent and handler machines need to know each other's identity in order to communicate. This is achieved by hard-coding the IP address of the handler machines in the attack code that is later installed on the agent. Each agent then reports its readiness to the handlers, who store its IP address in a file for later communication. The obvious drawback of this approach is that discovery of one compromised machine can expose the whole DDoS network. Also, since agents and handlers listen to network connections, they are identifiable by network scanners.

B. *Attacks with indirect communication*

Attacks with indirect communication deploy a level of indirection to increase the survivability of a DDoS network. Recent attacks provide the example of using IRC channels for agent/handler communication. The use of IRC services replaces the function of a handler, since the IRC channel offers sufficient anonymity to the attacker. Since DDoS agents establish outbound connections to a standard service port used by a legitimate network service, agent machine open, enabling easy future access and modification of the attack code. Both semi-automatic and automatic attacks recruit the agent machines by deploying automatic scanning and propagation techniques. Based on the scanning strategy, we differentiate between attacks that deploy random scanning, hit list scanning, topological scanning, permutation scanning and local subnet scanning.

C. *Protocol Attacks*

Protocol attacks exploit a specific feature or implementation bug of some protocol installed at the victim in order to consume excess amounts of its resources. Examples include the TCP SYN attack, the CGI request attack and the authentication server attack. In the TCP SYN attack, the exploited feature is the allocation of substantial space in a connection queue immediately upon receipt of a TCP SYN request. The attacker initiates multiple connections that are never completed, thus filling up the connection queue indefinitely. In the CGI request attack, the attacker consumes the CPU time of the victim by issuing multiple CGI requests. In the authentication server attack, the attacker exploits the fact that the signature verification process consumes significantly more resources than bogus signature generation. He sends numerous bogus authentication requests to the server, tying up its resources.

D. Brute-force Attacks

Brute-force attacks are performed by initiating a vast amount of seemingly legitimate transactions. Since an upstream network can usually deliver higher traffic volume than the victim network can handle, this exhausts the victim's resources. We further divide brute-force attacks based on the relation of packet contents with victim services into filterable and non-filterable attacks.

E. Variable Rate Attacks

Variable rate attacks are more cautious in their engagement, and they vary the attack rate to avoid detection and response. Based on the rate change mechanism, we differentiate between attacks with increasing rate and fluctuating rate and resume it at a later time. If this behavior is simultaneous for all agents, the victim experiences periodic service disruptions. If, however, agents are divided into groups who coordinate so that one group is always active, then the victim experiences continuous denial of service.

F. Classification by Impact

Depending on the impact of a DDoS attack on the victim, we differentiate between disruptive and degrading attacks.

Disruptive Attacks:

The goal of disruptive attacks is to completely deny the victim's service to its clients. All currently known attacks belong to this category.

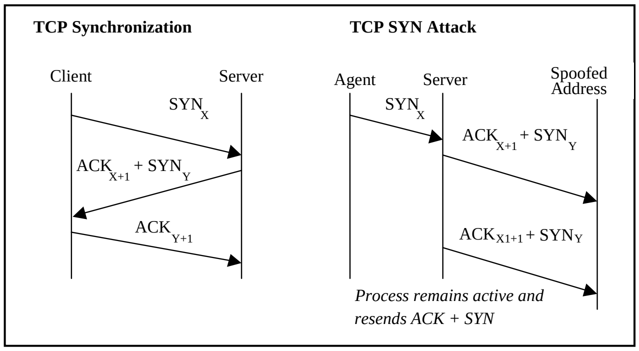
Degrading Attacks:

The goal of degrading attacks would be to consume some (presumably constant) portion of a victim's resources. Since these attacks do not lead to total service disruption, they could remain undetected for a significant time period. On the other hand, damage inflicted on the victim could be immense.

II. COMMON ATTACKS AND TYPES OF PREVENTION MECHANISM

A. Common Attacks

TCP SYN Flood Attacks: The Transfer Control Protocol (TCP) includes a full handshake between sender and receiver, before data packets are sent. The initiating system sends a SYN (Synchronize) request (see figure a). The receiving system sends an ACK (acknowledgement) with its own SYN request. The sending system then sends back its own ACK and communication can begin between the two systems.



TCP Synchronization
TCP SYN Attack
Client
Server
Agent
Server
Spoofed Address
SYN_X
$\text{ACK}_{X+1} + \text{SYN}_Y$
ACK_{Y+1}
SYN_X
$\text{ACK}_{X+1} + \text{SYN}_Y$
$\text{ACK}_{X1+1} + \text{SYN}_Y$
Process remains active and resends ACK + SYN

Fig. 1. TCP Synchronization and TCP SYN Attack [4]

If the receiving system is sent a SYN$_X$ packet but does not receive an ACK$_{Y+1}$ to the SYN$_Y$ it sends back to the sender, the receiver will resend a new ACK + SYN$_Y$ after some time has passed (see figure b). The processor and memory resources at the receiving system are reserved for this TCP SYN request until a timeout occurs [4].

The TCP SYN attack exploits the three-way handshake between the sending system and the receiving system by sending large volumes of TCP SYN packets to the victim system with spoofed source IP addresses, so the victim system responds to a non-requesting system with the ACK+SYN [5]. When a large volume of SYN requests are being processed by a server and none of the ACK+SYN responses are returned, the server begins to run out of processor and memory resources. Eventually, if the volume of TCP SYN attack requests is large and they continue over time, the victim system will run out of resources and be unable to respond to any legitimate users.

UDP Flood Attack: In UDP Flood attack attacker sends large number of UDP packets to a victim system, due to which there is saturation of the network and the depletion of available bandwidth for legitimate service requests to the victim system. A UDP Flood attack is possible when an attacker sends a UDP packet to a random port on the victim system. When the victim system receives a UDP packet, it will determine what application is waiting on the destination port. When it realizes that there is no application that is waiting on the port, it will generate an ICMP packet of destination unreachable [23] to the forged source address. If enough UDP packets are delivered to ports of the victim, the system will go down. UDP flood attacks may also fill the bandwidth of connections located around the victim system (depending on the network architecture and line-speed). This can sometimes cause systems connected to a network near a victim system to experience problems with their connectivity [4].

ICMP Attacks: ICMP Flood attacks exploit the Internet Control Message Protocol (ICMP) [4], which enables users to send an echo packet to are mote host to check whether it’s alive. More specifically during a DDoS ICMP flood attack the agents send large volumes of ICMP_ECHO_ REPLY packets

(“ping”) to the victim. These packets request reply from the victim and this results in saturation of the bandwidth

of the victim’s network connection. During an ICMP flood attack the source IP address may be spoofed.

Land Attacks: Land attacks have been found in services like Simple Network Management Protocol (SNMP) and

Windows which were caused by design flaws where the devices accepted requests on the wire appearing to be from themselves and causing replies repeatedly.

B. DoS attack Prevention Mechanisms [20]

The prevention mechanisms are: pattern detection, anomaly detection, hybrid detection, and third-party detection. Mechanisms with Pattern Attack Detection: Mechanisms that deploy pattern detection store the signatures of known attacks in a database. Each communication is monitored and compared with database entries to discover occurrences of DDoS attacks. Occasionally, the database is updated with new attack signatures. The obvious drawback of this detection mechanism is that it can only detect known attacks, and it is usually helpless against new attacks or even slight variations of old attacks that cannot be matched to the store d signature. On the other hand, known attacks are easily and reliably detected, and no false positives are encountered.

Mechanisms with Anomaly Attack Detection: Mechanisms that deploy anomaly detection have a model of normal system behavior, such as a model of normal traffic dynamics or expected system performance. The current state of the system is periodically compared with the models to detect anomalies. Approaches presented in provide examples of mechanisms that use anomaly detection. The advantage of anomaly detection over pattern detection is that unknown attacks can be discovered. However, anomaly-based detection has to address two issues:

- *Threshold setting:* Anomalies are detected when the current system state differs from the model by a certain threshold. The setting of a low threshold leads to many false positives, while a high threshold reduces the sensitivity of the detection mechanism.
- *Model update:* Systems and communication patterns evolve with time, and models need to be updated to reflect this change. Anomaly based systems usually perform automatic model update using statistics gathered at a time when no attack was detected. This approach makes the detection mechanism vulnerable to increasing rate attacks that can mistrial models and delay or even avoid attack detection.

Mechanisms with Hybrid Attack Detection: Mechanisms that deploy hybrid detection combine the pattern-based and anomaly-based attack stream. The disadvantage is that they allow some attack traffic through, so extremely high scale attacks might still be effective even if all traffic streams are rate-limited.

Filtering Mechanisms: Filtering mechanisms use the characterization provided by a detection mechanism to filter out the attack stream completely. Examples include dynamically deployed firewalls, and also a commercial system Traffic Master. Unless detection strategy is very reliable, filtering mechanisms run the risk of accidentally denying service to legitimate traffic. Worse, clever attackers might leverage them as denial-of-Service tools.

Reconfiguration Mechanisms: Reconfiguration mechanisms change the topology of the victim or the intermediate network to either add more resources to the victim or to isolate the attack machines. Examples include reconfigurable overlay networks, resource replication services, attack isolation strategies etc.

III. LINUX OS AND IPTABLES

Linux OS: The Linux kernel is the operating system (OS) kernel used by the Linux family of Unix-like operating systems. It is a prominent example of free and open source software. The Linux kernel is released under the GNU General Public License version 2 (GPLv2) plus some firmware images with various non-free licenses, and is developed by contributors worldwide. The Linux kernel was initially conceived and created by Finnish computer science student Linus Torvalds [21] in 1991. Linux rapidly accumulated developers and users who adapted code from other free software projects for use with the new operating system. The Linux kernel has received contributions from thousands of programmers. Many Linux distributions have been released based upon the Linux kernel.

IPtables: IPtables is a software solution which is available on most Linux OS with a kernel version 2.4 or newer [17]. To be honest we have to say that IPtables is not the firewall itself. The IPtables program is a frontend which can be called from the command line to alter filter tables in the kernel. The real firewall is present in the kernel. Because most people will only use the IPtables program, it is often referred to as the Linux firewall and we will do it here also for convenience. There are a number of ways to solve potential performance problems in the firewall. The first thing that should be done is order the rules in such a way that rules which have the highest chance to match should be in the beginning of the tables. Furthermore we should try to organize the tables for the different types of packets we have defined are processed in such a way that packets which are accepted are accepted in an early stage, while only packets which will probably blocked anyway should traverse the whole chain of tables and rules.

We will divide packets in different groups. Each group of packets deserves its own treatment.

- The enemies are packets coming from sources, or going to destinations which are prohibited.
- Friends are packets which are coming from a trusted source. Friends have more privileges than other packets.
- One method hackers use to attack networked computers is to send them packets which are invalid, i.e. Bogus packets.
- Allowed packets are packets where it is absolutely sure that no blocking firewall rule will match them.

IPtables management: More elaborate rules can be created that control access to specific subnets, or even specific nodes, within a LAN. You can also restrict certain dubious applications or programs such as trojans, worms, and other client/server no legitimate services that communicate via these non-standard ports, blocking them can effectively diminish the chances that potentially infected nodes on your network independently communicate with their remote master servers. We can also block outside connections that attempt to spoof private IP address ranges to infiltrate your LAN.

Structure of IPtables:

IPtables commands have the following structure:

IPtables [**-t** <***table-name***>] <***command***> <***chain-name***>
<***parameter-1***> <***option-1***> \ <***parameter-n***> <***option-n***>

<***table-name***> — Specifies which table the rule applies to

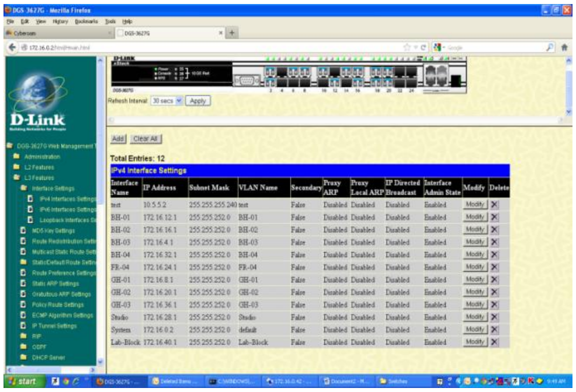
<***command***> — Specifies the action to perform, such as appending or deleting a rule.

<***chain-name***> — Specifies the chain to edit, create, or delete.

<***parameter***>-<***option***> pairs — Parameters and associated options that specify how to process a packet that matches the rule. The length and complexity of an IPtables command can change significantly, based on its purpose.

IV. BACKGROUND OF THIS RESEARCH WORK

I have done an extensive survey on exiting network infrastructure. The network's IP pool 172.16.0.0/22, is divided into twelve different VLANs (Virtual Local Area Network) according to the building, as per requirement. The diagram shows VLANs details:



Total Entries: 12

IPv4 Interface Settings

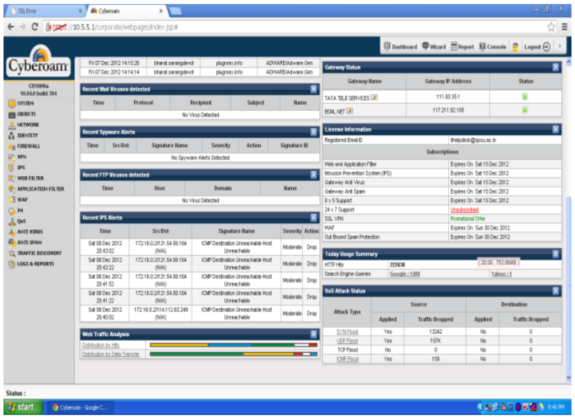
Interface Name	IP Address	Subnet Mask	VLAN Name	Secondary	Proxy ARP	Proxy Local ARP	IP Directed Broadcast	Interface Admin State	Modify	Delete
test	10.5.5.2	255.255.255.240	test	False	Disabled	Disabled	Disabled	Enabled	Modify	✕
BH-01	172.16.12.1	255.255.252.0	BH-01	False	Disabled	Disabled	Disabled	Enabled	Modify	✕
BH-02	172.16.16.1	255.255.252.0	BH-02	False	Disabled	Disabled	Disabled	Enabled	Modify	✕
BH-03	172.16.4.1	255.255.252.0	BH-03	False	Disabled	Disabled	Disabled	Enabled	Modify	✕
BH-04	172.16.32.1	255.255.252.0	BH-04	False	Disabled	Disabled	Disabled	Enabled	Modify	✕
FR-04	172.16.24.1	255.255.252.0	FR-04	False	Disabled	Disabled	Disabled	Enabled	Modify	✕
GH-01	172.16.8.1	255.255.252.0	GH-01	False	Disabled	Disabled	Disabled	Enabled	Modify	✕
GH-02	172.16.20.1	255.255.252.0	GH-02	False	Disabled	Disabled	Disabled	Enabled	Modify	✕
GH-03	172.16.36.1	255.255.252.0	GH-03	False	Disabled	Disabled	Disabled	Enabled	Modify	✕
Studio	172.16.28.1	255.255.252.0	Studio	False	Disabled	Disabled	Disabled	Enabled	Modify	✕
System	172.16.0.2	255.255.252.0	default	False	Disabled	Disabled	Disabled	Enabled	Modify	✕
Lab-Block	172.16.40.1	255.255.252.0	Lab-Block	False	Disabled	Disabled	Disabled	Enabled	Modify	✕

Fig. 2. VLAN configured on L3 Switch

It is observed during survey that, the UTM (Unified Threat Management) which installed for internet security in exiting network is rebooting frequently due to heavy Dos/DDos attack. The fact is observed on the Dashed Board of the UTM (Cyberoam 1000ia). It is also observed that, some specific VLANs are the origin of these attacks. The Dos/DDos attacks mainly shown on the UTM Dash Board are TCP SYN Flood attack, UDP Flood attack and ICMP Flood attack.

The main concern of this research is mitigation/prevention of the DoS attacks, so that, the existing network will be optimize. The configuration of VLANs on Layer Three (L3) switch and DHCP scope which are configured on DHCP server through which the DoS/DDoS attacks are able to identify from where the DoS/DDoS attacks are coming. We can also indicate the most affected VLANs in the existing network.

Following figures (Fig. 3 and Fig. 4) shows the UTM-Dash Board and DHCP scope on DHCP server:



https://10.5.5.1/corporate/webpages/index.jsp#

Fri 07 Dec 2012 14:15:26	bharat.sarangdevot	plugnrex.info	ADWARE/Adware.Gen
Fri 07 Dec 2012 14:14:14	bharat.sarangdevot	plugnrex.info	ADWARE/Adware.Gen

Recent Mail Viruses detected

Time	Protocol	Recipient	Subject	Name
No Virus Detected				

Recent Spyware Alerts

Time	Src:Dst	Signature Name	Severity	Action	Signature ID
No Spyware Alerts Detected					

Recent FTP Viruses detected

Time	User	Domain	Name
No Virus Detected			

Recent IPS Alerts

Time	Src:Dst	Signature Name	Severity	Action
Sat 08 Dec 2012 20:43:02	172.16.0.2/121.54.58.164 (N/A)	ICMP Destination Unreachable Host Unreachable	Moderate	Drop
Sat 08 Dec 2012 20:42:22	172.16.0.2/121.54.58.164 (N/A)	ICMP Destination Unreachable Host Unreachable	Moderate	Drop
Sat 08 Dec 2012 20:41:52	172.16.0.2/121.54.58.164 (N/A)	ICMP Destination Unreachable Host Unreachable	Moderate	Drop
Sat 08 Dec 2012 20:41:22	172.16.0.2/121.54.58.164 (N/A)	ICMP Destination Unreachable Host Unreachable	Moderate	Drop
Sat 08 Dec 2012 20:40:02	172.16.0.2/114.112.63.249 (N/A)	ICMP Destination Unreachable Host Unreachable	Moderate	Drop

Web Traffic Analysis

Distribution by Hits

Distribution by Data Transfer

Gateway Status

Gateway Name	Gateway IP Address	Status
TATA TELE SERVICES	111.93.35.1	●
BSNL.NET	117.211.92.105	●

License Information

Registered Email ID	thelpdesk@spsu.ac.in
Subscriptions	
Web and Application Filter	Expires On Sat 15 Dec 2012
Intrusion Prevention System (IPS)	Expires On Sat 15 Dec 2012
Gateway Anti Virus	Expires On Sat 15 Dec 2012
Gateway Anti Spam	Expires On Sat 15 Dec 2012
8 x 5 Support	Expires On Sat 15 Dec 2012
24 x 7 Support	Unsubscribed
SSL VPN	Promotional Offer
WAF	Expires On Sun 30 Dec 2012
Out Bound Spam Protection	Expires On Sun 30 Dec 2012

Today Usage Summary

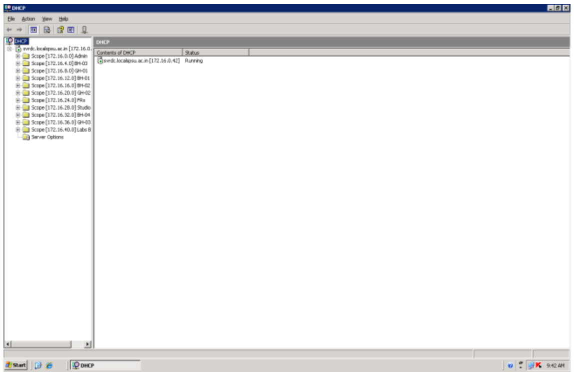
HTTP Hits	222638	(20:00 , 753.66MB)
Search Engine Queries	Google : 1498	Yahoo : 1

DoS Attack Status

Attack Type	Source		Destination	
	Applied	Traffic Dropped	Applied	Traffic Dropped
SYN Flood	Yes	13242	No	0
UDP Flood	Yes	157K	No	0
TCP Flood	No	0	No	0
ICMP Flood	Yes	159	No	0

Status :

Fig. 3. UTM Dash-Board



DHCP
- svrdc.localspsu.ac.in [172.16.0.
  - Scope [172.16.0.0] Admin
  - Scope [172.16.4.0] BH-03
  - Scope [172.16.8.0] GH-01
  - Scope [172.16.12.0] BH-01
  - Scope [172.16.16.0] BH-02
  - Scope [172.16.20.0] GH-02
  - Scope [172.16.24.0] FRs
  - Scope [172.16.28.0] Studio
  - Scope [172.16.32.0] BH-04
  - Scope [172.16.36.0] GH-03
  - Scope [172.16.40.0] Labs B
  - Server Options

Contents of DHCP	Status
svrdc.localspsu.ac.in [172.16.0.42]	Running

Fig. 4. DHCP scope for different VLANs on DHCP Server

VLANs gives a virtualized network environment in which IP addresses are virtually segregated for the whole network. It prevent unwanted broadcast between different VLAN and preventing unwanted congestion in the network.

V. DESIGN OF FRAMEWORK

The aim of this research work to design the below framework to mitigate the DoS/DDoS attacks inside the network or most affected VLANs. The proposed model will be deployed in the heavy affected VLANs and filter the unwanted traffic and the performance of live network traffic will be analyzed.

The below diagram describe that, the access level switches are connected with Desktop (having Dual NIC and IPtables configured in LINUX) through one NIC (Network Interface Card) in the same VLAN and then the Desktop is connected with Fiber Switch (Layer Two, L2) with another NIC of the desktop. The Fiber switch further connected to Server Room for Internet access. The desktop is the gateway of all the access level switches in that particular VLAN. The operating system (OS) Linux has been chosen because the Linux is open source and in that, IPtables are more powerful firewall than other software firewall.

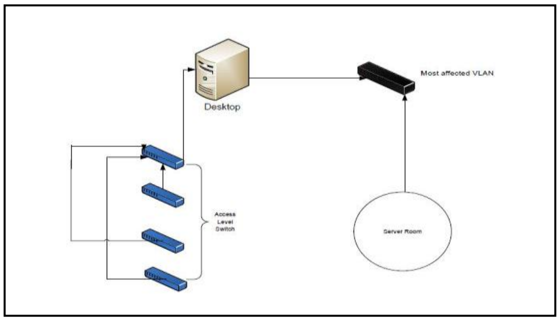


Fig. 5. Proposed model for mitigating DoS/DDoS attack in heavily affected VLANs

A. Basic functions of the Iptables Firewall

The first thing that should be done is order the rules in such a way that rules which have the highest chance to match should be in the beginning of the tables. To do this, it is proposed the following flow of packets through the firewall. Five chains [17] of rules are predefined in the kernel. The PREROUTING is the first chain for incoming packets. From the PREROUTING chain, packets can be either forwarded to the INPUT chain or the FORWARD chain. The INPUT chain is for packets which should be delivered locally. The FORWARD chain is only used on computers where routing is enabled. It causes packets to be forwarded to another destination than the local computer. The OUTPUT chain is used to postprocess packets which originate from the local computer and the postrouting chain brings the packets to the networking hardware for remote delivery.

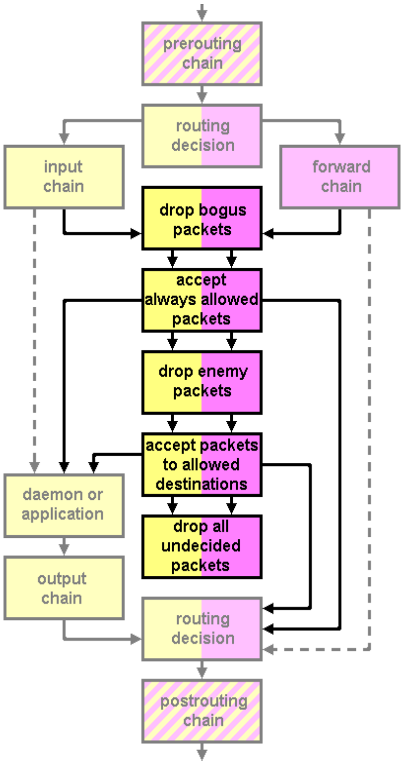


Fig. 6. Packet flow through IPtables firewall [17]

B. Expected Results

It is very costly to replace all of fixed infrastructure of any existing networks to provide to give better efficiency and performance. There are main four parameters which affects the performance of any networked system that are packet processing speed, bandwidth, throughput and uninterrupted power supply to the network equipments. This proposed framework mitigate (minimize) DoS/DDoS attacks and analyzed the optimization for better performance of the major affected VLANs of the existing network. The following are the impacts of proposed framework:

- The efficiency will be increased of the systems.
- The performance will also increase.
- If the error decrease than throughput will as usually increase.
- The IT infrastructure cost will be saved.

VI. CONCLUSION

Mitigation of DoS/DDoS attacks is a part of an overall risk management strategy for an organization. Each organization must identify the most important DoS/DDoS risks. In this paper, the proposed framework is for mitigation of various DoS/DDoS attacks from different VLANs of existing network and it also optimize the efficiency, throughput and reduce IT infrastructure cost. The Iptables firewall will filer bogus traffic on live network. This enables us to protect our system from a wide variety of hazards, including service attacks and hack attempts. The performance of the firewall will analyzed with help of packet sniffer tool for further improvement of the IPtables scripting. To determine whether the network traffic is legitimate or not, a iptables relies on a set of rules it contains that are predefined by a network or system administrator. The framework or model is easily deployable and it is compatible with existing network.

ACKNOWLEDGEMENT

I would like to thank Sir Padampat Singhania University, Udaipur, India for providing me the platform for this research work.

REFERENCES

[1] SOS: An Architecture For Mitigating DDoS Attacks by Angelos D. Keromytis, Member, IEEE, Vishal Misra, Member, IEEE, Dan Rubenstein, Member, IEEE, JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 21.

[2] An ISP Level Solution to Combat DDoS Attacks using Combined Statistical Based Approach, By B. B. Gupta, Manoj Misra and R. C. JoshiDepartment of Electronics And Computer Engineering Indian Institute of Technology, Roorkee, Journal of Information Assurance and Security 2 (2008) 102-110.

[3] Taxonomies of Distributed Denial of Service Networks, Attacks, Tools, and Countermeasures by Stephen Speeht and Ruby Lee, PALMS, Department of Electrical Engineering, Princeton University.

[4] A Taxonomy of DDoS Attack and DDoS Defense Mechanisms by Jelena Mirkovic, 449 Smith Hall, Computer and Information Sciences Department, University of Delaware Newark, DE 19716, Peter Reiher, 3564 Boelter Hall, Computer Science Department UCLA Los Angeles, CA 90095, Copyright 2004 ACM.

[5] Design and Development of Proactive Models for Mitigating Denial-of-Service and Distributed Denial-of-Service Attacks, by Nagesh H.R, K. Chandra Sekaran, IJCSNS International Journal of Computer Science and Network Security,VOL.7 No.7, July 2007.

[6] http://www.expertslogin.com/linux-administration/block-common-attacks-iptables/

[7] Detecting and Preventing IP-spoofed Distributed DoS Attacks by Yao Chen, Shantanu Das, Pulak Dhar, Abdulmotaleb El Saddik1, and Amiya Nayak, School of Information Technology and Engineering, University of Ottawa, International Journal of Network Security, Vol.7, No.1, PP.70–81, July 2008.

[8] http://www.techrepublic.com/article/linux-101-configuring-and-managing-iptables-to-improve-network-security/5997057

[9] http://security.stackexchange.com/questions/4603/tips-for-a-secure-iptables-config-to-defend-from-attacks-client-side

[10] https://access.redhat.com/knowledge/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Security_Guide/

[11] http://www.linuxforu.com/2011/10/syn-flooding-using-scapy-and-prevention-using-iptables/

[12] http://www.zoobey.com/index.php/resources/all-articles-list/495-25-most-frequently-used-linux-iptables-rules-examples

[13] http://agix.com.au/blog/?p=166

[14] http://www.expertslogin.com/linux-administration/block-common-attacks-iptables/

[15] http://www.slideshare.net/stom123/iptables-14068296#btnNext

[16] http://www.liquidcomm.net/news/tech-tips/apache/How-to-manage-a-DDOS-or-DOS-attempt-directed-at-your-linux-server.html

[17] http://www.lammertbies.nl/comm/info/iptables.html

[18] K.J. Houle, G.M. Weaver, Trends in Denial of Service attack technology, CERT and CERT coordination center, Carnegie Mellon University, October 2001.

[19] P.J. Criscuolo, Distributed Denial of Service Trin00, Tribe Flood Network, Tribe Flood Network.2000, and Stacheldraht CIAC-2319, Department of Energy Computer Incident Advisory (CIAC), UCRL-ID-136939, Rev. 1, Lawrence Livermore National Laboratory, February 14, 2000.

[20] A Taxonomy of DDoS Attacks and DDoS Defense Mechanisms Jelena Mirkovic, Janice Martin and Peter Reiher Computer Science Department University of California, Los Angeles Technical report #020018

[21] http://en.wikipedia.org/wiki/Linux_kernel