Volume 2, No. 3, May-June 2011
ISSN No. 0976-5697
International Journal of Advanced Research in Computer Science
RESEARCH PAPER
Available Online at www.ijarcs.info

Exit Control Based Cooperative Defense Mechanism to Minimize DDoS Attacks that Mimic Flash Crowds

Rashpinder Pal*
M. Tech. Scholar,
Department of Computer Science & Engineering,
Bhai Maha Singh College of Engineering,
Sri Muktsar Sahib, Punjab, India
er.rashpinder@gmail.com

Sunil Kumar
M. Tech.,
Department of Computer Science & Engineering,
Bhai Maha Singh College of Engineering,
Sri Muktsar Sahib, Punjab, India
Sunil,budhlada@gmail.com

Mandeep Singh
Assistant Professor
Department of Computer Science & Enginwwring,
Bhai Maha Singh College of Engineering,
Sri Muktsar Sahib, Punjab, India
Write_mandeep@yahoo.co.in

Abstract-- The Internet is considered as main infrastructure of the global information society. Therefore, the availability of Internet is very critical. Distributed Denial-of-Service (DDoS) attacks tend to degrade internet services severely. In order to effectively reduce the influence of DDoS attacks and its severity on the entire internet, we need Cooperative defense technique that can block the attackers' requests on the edge routers of ISP boundary. In this paper, we have proposed a ISPs' Cooperation based DDoS attack mitigation approach, which makes use of good characteristics of existing defense scenarios such as D-WARD and co-operation among the ISP's to make the defense distributed. The suggested topology contains gateway on every edge router of the ISP and validate every client's request by Exit Control Mechanism on every first request made from a new IP address. The puzzles can only be passed by humans and not by the bots, and thus it blocks all the non-genuine packets inside the ISP boundary & saves the entire internet's bandwidth from the attack traffic.

Keywords: DDoS, Defense, Cooperative, Puzzle, Edge router.

I. INTRODUCTION

Internet security includes aspects such as confidentiality, authentication, message integrity and non repudiation [1], [2]. One area that has been neglected thus far has been that of service availability in the presence of denial of service (DoS) attacks, and their distributed variants (DDoS). *Denial of Service (DoS)* attacks attempt to make a computer resource unavailable to its intended users. The attacks in DDOS Scenario become coordinated and come from multiple sources at the same time thus are even more devastating [3].

The bandwidth congestion attacks are identified as "Bulls Eye" in the communications substrate and attackers flood them with large volumes of traffic in case of web services [4].

To circumvent detection, attackers are increasingly moving away from pure bandwidth floods to stealthy DDoS attacks that mimic flash crowds. They profile the victim server and mimic legitimate Web browsing behaviour of a large number of clients. These attacks target higher layer server resources like sockets, disk bandwidth, database bandwidth and processes. Many DDoS attacks hide the true origin of the attacker by using spoofed source addresses. DDoS attacks are particularly attractive because their nature makes attribution even harder. Unlike traditional single-source attacks, DDoS attacks are virtually impossible to trace due to the numerous attack paths and the multiple levels of indirection [5]. Moreover, attack tools are constantly evolving and some already incorporate defenses like encryption and "decoy" packets to sidetrack their detection and traceback.

Insecure machines are used by DDoS attackers as their army to launch attack [6]. An attacker or hacker gradually implants attack programs on these insecure machines. These compromised machines are called Masters / Handlers or Zombies and are collectively called bots and the attack network is called botnet. Hackers send control instructions to masters, which in turn communicate it to zombies for launching attack. The zombie machines under control of masters/handlers as shown in Figure 1 transmit attack packets, which converge at victim or its network to exhaust either its communication or computational resources [7].

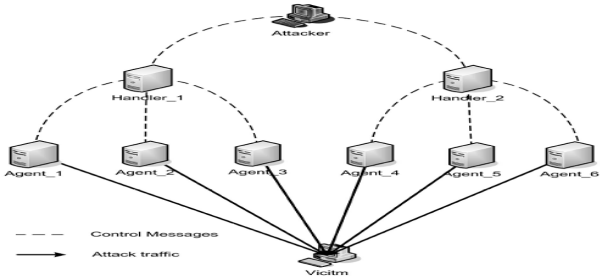

Figure 1: Typical architecture of a DDoS attack

There are two types of DDoS attacks. The first type of DDoS attack has the aim of attacking the victim to force it

out of service for legitimate users by exploiting software and protocol vulnerabilities of the system. The second type of DDoS attack is based on a huge volume of attack traffic, which is known as a flooding-based DDoS attack. Flooding DDoS is basically a resource overloading problem, Mirkovic et al. [7] and Peng et al. [8], the resource can be bandwidth, memory, CPU cycles, and buffers etc. The congestion and flow control signals [9], [10] force legitimate clients to decrease their rate of sending requests, whereas attack packets keep coming. Finally, a stage comes when only attack traffic is reaching at the server. Thus, service is denied to legitimate clients due to limited bottleneck bandwidth.

II. CAUSES OF DDOS ATTACKS

The rapid expansion of the Internet and the proliferation of low-cost PCs are two important factors that have made DDoS feasible. In addition, the following recent trends have contributed to the rise in DDoS attacks:

A. The increase in the number of new software and the (inevitable) security vulnerabilities that accompany them, present many opportunities to hijack computers.
B. The number of computers with broadband connections has been rapidly increasing. Not only do these computers pose a danger (if hijacked) due to their high-speed connections, but their "always on" nature makes them far more susceptible to compromise.
C. The lack of automated security update of software vulnerabilities means that the user is responsible for carrying out this task manually. Since many users either lacks the time, knowledge or motivations to do so, many systems remain running software with known insecurities.
D. The availability of attack tools (along with instructions on how to use them) on several web sites, drastically expands the number of potential attackers, who no longer need to understand the operation of the tools in order to use them. Termed "script kiddies", attacker can use attack tools without understanding them.

Efforts to stop DDoS attacks by manually securing systems or by tracing back the attack although commendable are difficult to achieve. The lack of attribution, impossibility of securing every machine on the Internet, and difficulty of performing intrusion detection, mean that host-based or highly localized solutions to neutralize DDoS attacks will not work. What is needed is a solution which offers the right incentives and low administration overhead, so that it can be willingly adopted by ISPs and network administrators in the entire Internet [5].

III. CURRENT SCENARIO OF DDOS DEFENSES

For the most part, they still require a high degree of manual intervention. Individuals highly trained in network operations and security, pour over audit data and form convincing hypotheses consistent with the audit trails. They then contact other ISPs in the Internet to confirm suspicious traffic patterns and coordinate a collective response to the attack.

Attempts are being made to develop tools to automate the analysis of audit data using Intrusion Detection Systems (IDS) that perform high-speed pattern matching against a database of known attack signatures. Studies of the effectiveness of IDS systems have so far shown that they are incapable of reasonably detecting previous unknown attacks. They only perform well when presented with attacks which are represented in their signature databases.

Ongoing research efforts have been focused on traceback techniques for attribution. Many believe that if one could trace back to the origin of the attack, it will be possible to effectively counter the attack by automated means. It is unclear whether it is useful to expend vast amounts of resources to traceback and identify individual attacker when the generals at the top of the hierarchical command and control continue to operate unnoticed and uninhibited [5].

Three innovative Mechanisms can be used to provide DDoS defenses in a better way:

An Exit Control Defense that can detect attacks that have not been seen before, this defense can block the malicious request packets inside the ISP's boundary by Exit Control Puzzle based Mechanism.

The First technique is to be cooperatively employed by all the ISP's in order to limit the attacks originated from their network.

Make use of Cookie Table to store the IP addresses of those legitimate users', who have passed the graphical test. After storing the cookie information in Cookie Table that update must be conveyed to the other gateways which are connected to other edge router of same ISP. So that the legitimate users may not suffer from the overhead of another graphical puzzle by other gateways at boundaries.

Our proposed Cooperative system leverages these innovative technologies to develop an automated system for DDoS attack mitigation. It requires no manual intervention, will be attack signature independent, and will be largely complementary to ongoing research in traceback.

IV. RELATED WORK

The D-Ward system [6] monitors outgoing traffic from a given source network and attempts to determine outgoing attack traffic. Attack traffic is identified by comparing the traffic patterns against models of reasonable congestion control behavior. For example, TCP traffic is monitored and compared to an equation approximation of the TCP congestion control model. TCP streams that are observed violating the behavior of the model is marked as an attack and is subsequently throttled back by the edge network's egress router. The amount of throttling is proportional to the flows deviation from its expected behavior. In a similar fashion, the same approach can be applied to other transport protocols. The health of destination hosts can be gleaned using ICMP echo/reply probes or other techniques that generate the necessary 2-way traffic needed to analysis the compliance of a given flow to reasonable congestion control behavior.

Another coordination approach that has been explored is traceback [11]. In SPIE [12], state is stored in the network for a short period of time that enables edge networks to traceback the origin of a given packet. A query mechanism traces back into the network looking for evidence of a

packet traversing particular routers. A probabilistic match algorithm follows back a small number of possible paths until the correct path is determined. Recent efforts on neutralizing DDoS attacks have focused on attribution via IP traceback. The immediate goal is to locate the hosts the attack originates [11] [13] [14] [15]. Traceback also offers the hope of locating the attacker through the instruments of the attack. Traceback schemes can be divided in two categories: (a) probabilistic packet marking (PPM), and (b) tunneling techniques. While PPM techniques work well for single-source attacks, they are woefully inadequate for large DDoS attacks.

The main reason is that there exists a trade-off between localization and marking probability, path length and traffic volume [16] [17]. Tunneling techniques [18] require the ability to dynamically set up tunnels between any access points and thus require substantial support from the network. They also suffer from the same limitations as PPM techniques.

Pushback [19] and Aggregate-Based Congestion Control (ACC) are project at AT&T Center for Internet Research. The routers in the system assume that the congestion of local packet queue is the sign of DDoS attack and take action to rate limit the identified aggregates which are responsible of queue congestion according to local policy. If the congested router cannot control the aggregate itself, it issues a rate limit request to its immediate upstream neighbors who carry the aggregates traffic to apply rate limiting to specified excessive flows. These requests will be propagated upstream as far as the identified aggregates have been effectively controlled. This approach request all the routers on the path of aggregate traffic be augmented with the pushback capability.

In [20] a collaborative DDoS defense system is proposed in which routers act as gateways, detecting DDoS attacks locally and identifying and dropping packets from misbehaving flows. Gateways are installed and communicate only within the source and the victim domains, thus providing cooperative defense of a limited scope.

Proactive defense mechanism [21], the motivation for these approaches is based on the observation that it is hard to detect DDoS attacks. So instead of detecting the attacks by using signatures (attack pattern) or anomaly behaviour, these approaches try to improve the reliability of the global Internet infrastructure by adding extra functionality to Internet components to prevent attacks and vulnerability exploitation. The primary goal is to make the infrastructure immune to the attacks and to continue to provide service to normal users under extreme conditions.

Post attack analysis [22], the purpose of post attack analysis is to either look for attack patterns that will be used by IDS or identify attackers using packet tracing. The goal of packet tracing is to trace Internet traffic back to the true source (not spoofed IP address). As attackers change their strategy frequently, analyzing huge amount of traffic logs is time consuming and useless in detecting new attacks. Trace back mechanism can help to identify zombies in some situations; however, it is impractical to defend against DDoS attacks for the following reasons. First, during a DDoS attack, the attacker will control thousands of zombies (numbers will increase in the future) to launch an attack. As a result, identifying these zombies is expensive and infeasible. Second, since different network administrators control different section of the global Internet, it would be difficult to determine who would be responsible for providing trace back information.

Similarly, COSSACK [5] forms a multicast group of defense nodes which are deployed at source and victim networks. Each defense node can autonomously detect the attack and issue an attack alert to the group. Sources involved in the attack cooperate with the victim to suppress it.

V. PRESENT WORK

This paper proposes a co-operative Exit Control defense technique, which implements a topology containing a gateway on every edge router of the ISP boundary and validate every client's outgoing request by implementing Exit Control Mechanism on every gateway connected to every edge router, which consists of graphical puzzle [23] for every first request made from a new IP address. This Mechanism provides authentication using graphical tests and is different from other systems that use graphical tests. It uses a counting stage to identify the IP addresses that ignore the test, and attack the server with requests despite repeated failures at solving the tests. These machines are bots because their intent is to congest the bandwidth or server. Once these machines are identified by their failure threshold count, Exit Control Mechanism blocks their requests, turns the graphical tests off, and allows access to legitimate users who are unable or unwilling to solve graphical tests.

Authenticating Gateway sends a test and checks the client's answer without allowing unauthenticated clients access to sockets, buffers, ports and processes [23] [24] [25]. Thus, it protects the Exit Control Mechanism itself from being attacked.

This Technique makes use of SYN cookies to prevent spoofing of IP addresses and a counter to count how many times an IP address failed to solve a test. It discards requests from a client if unsolved tests' count exceeds a given threshold.

It would be inconvenient if legitimate users had to solve a puzzle for every HTTP request or every TCP connection. The Exit Control Gateway at edge router gives an HTTP cookie to a user who solves the test correctly. This cookie allows the user to re-enter the system. If a new HTTP request is accompanied by a cryptographically valid HTTP cookie, the Exit Control Gateway allows the request to go outside the ISP's Boundary without serving a new graphical test.

When the Exit Control Gateway issues a puzzle, it creates a packet as shown in Figure 2. The packet consists of a puzzle ID, a Random Number, Packet creation time, an Encrypted Key of puzzle ID & Random Number and an Authentication Bit (Set if Key Matches & Reset if No-Match) . The packet is embedded in the HTML form as the puzzle and sent to the client.

Puzzle ID	Random No.	Packet Time	Key	Authentication Bit

Figure: 2 Gateways' Packet

the answer to the server along with the Gateway Packet. The Gateway verifies the packet by checking the

key. Then the Exit Control Gateway checks the Gateway Packet to ensure the packet was created in this session only. Then, the Gateway checks if the answer to the puzzle is correct. If all checks are successful, the Gateway creates a HTTP cookie and gives it to the user. The cookie is created from the packet by updating the packet creation time and recording the packet in the table of valid HTTP cookies. Subsequently, when a user issues a new TCP connection with an existing HTTP cookie, the server validates the cookie by matching the Key and ensuring that the cookie has not expired. The Exit Control Gateway uses the cookie table to keep track of the number of simultaneous HTTP requests that belong to each cookie.

There can be a severe problem if the attacker would solve a single graphical test manually and distribute the HTTP cookie to a large number of bots. Exit Control Gateway doesn't ignore this issue & the client can execute a Limited number of simultaneous HTTP requests [26] after solving a graphical test. The cookie will not work if distributed among multiple zombies.

The defenses based only on graphical puzzle have two disadvantages. First, the attacker attack with SYN cookies by ignoring graphical tests, imposing an unnecessary overhead on the Gateway. Second, and more important, humans who are unable to solve Graphical Puzzles may be denied service. To deal with this problem, our strategy distinguishes legitimate users from bots by their reaction to the graphical test rather than their ability to solve it. Once the zombies are identified, they are blocked inside the ISP boundary and they can not generate the non-genuine HTTP Request packets. Thus it saves the entire internet's bandwidth from the attack traffic [27] [13].

VI. SIMULATION SETUP

A. Simulation Technique

The Desired Network topology is created using topology generator tools for NS-2, so a compatible script e.g. .tcl or .ns has to be generated by the topology generator. GT-ITM topology generator generates tcl script of desired topology as output.

We have chosen TCP applications to be run on our simulation topology. So we have used web cache model available in NS-2 for generating legitimate web traffic. Attack traffic is generated using CBR model in our simulations. Now whole of traffic is monitored and is logged for off-line analysis. The logged file is then used for measuring performance of the Defense Mechanism.

B. Topology

Figure 3 shows our simulation topology. Six ISP's networks, out of which one ISP's network contains zombies and two ISPs' networks contains Servers. All the Six ISPs' networks contain legitimate users that generate legitimate traffic. All the networks are connected with each other via edge routers, Access routers and core routers. In the simulated topology each ISP network has 1500 client nodes they all are connected to the core via an access router. The two ISPs with malicious nodes contain 700 zombies each. Links between the access router and the core have 100 Mbps bandwidth.

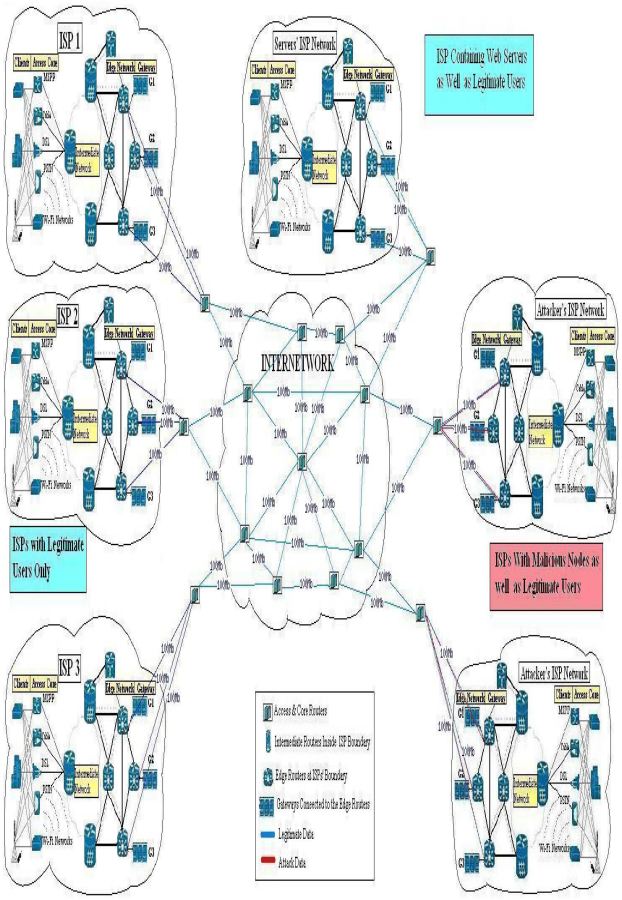

Figure 3: Proposed Topology for Simulation

Table 1: Basic Parameters for Simulation

Parameters	Values
Legitimate Traffic Type	HTTP
Legitimate Packet Size	584 bytes
Attack Request Size	75 Bytes
IP Spoofing for Attack	Enabled
Legitimate Clients	9000 For Total 6 ISPs
Attackers	700 from 2 ISPs
Attack Traffic Type	TCP
Access Bandwidth	100 Mbps
Access Link Delay	3ms
Attack Period	20 to 60 Seconds
Defense Period	40 to 60 Seconds

C. Legitimate Connections & Traffic

We create HTTP traffic which is a typical traffic in the current Internet network using the Web cache model. So

HTTP traffic is created as legitimate traffic on our simulation network. Among several Web traffic models of NS2, the Web caching model matches real traffic produced by Web application very well.

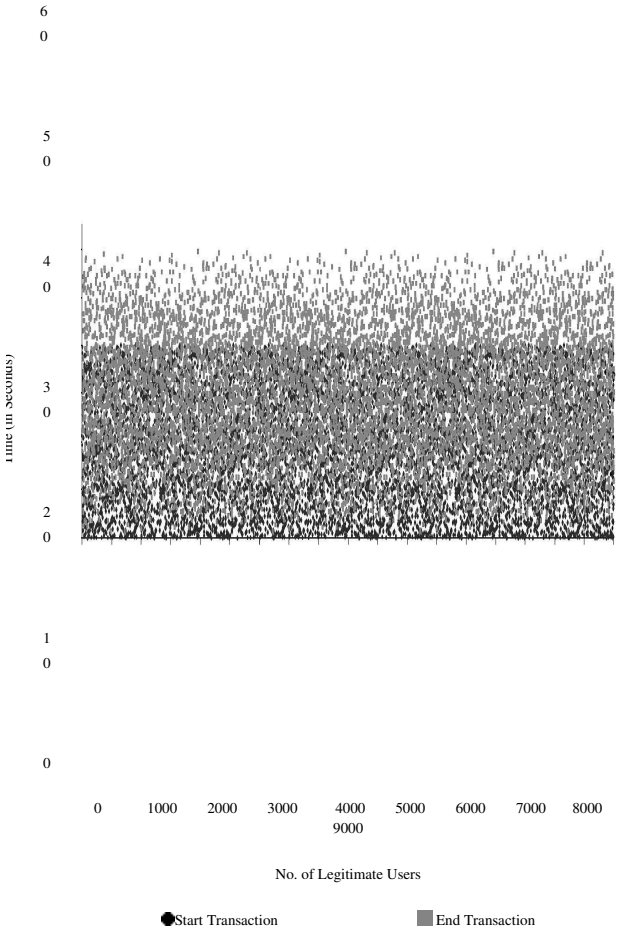

Figure 4: The Legitimate SYN & FIN Packets

D. Attack Traffic

We have used TCP traffic for generating DDoS flood. Flooding attacks can deny service in two ways:

a. By generating a huge volume of traffic that exhausts bandwidth on the backbone links.
b. By generating a high packet rate that exhausts the CPU at an intermediate router or the target host. In this simulation, we have generated TCP bandwidth flood with FLAT, PULSE and RAMP distributions to achieve attacks in different scenarios.

VII. RESULTS AND DISCUSSIONS

A. Performance Metrics

To evaluate performance of the proposed defense mechanism, we adopt the following measurement:

a. Measure the GoodPut, Badput and Throughput in All Scenarios.
b. Measure the Legitimate Traffic Drop Rate under the different pattern of DDoS attacks & Defenses respectively.
c. We will analyze the communication overhead also, which is introduced by this cooperative mechanism.

B. Goodput, Badput, and Throughput

During a DDoS attack, attack traffic consumes the entire bandwidth in order to force the edge router at the ISP of victim end to drop most legitimate packets. In the following figures, we just concentrate on the attack period which is started at 20s and defense period which is started at 40s.

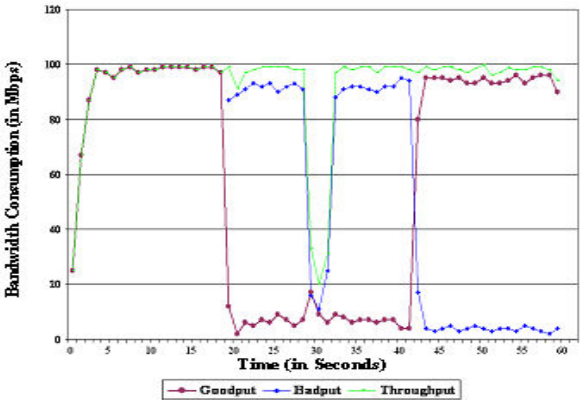

Figure 5: Measurement of Goodput, Badput and Throughput Without Attack, With TCP Flat Attack and With Defense Applied

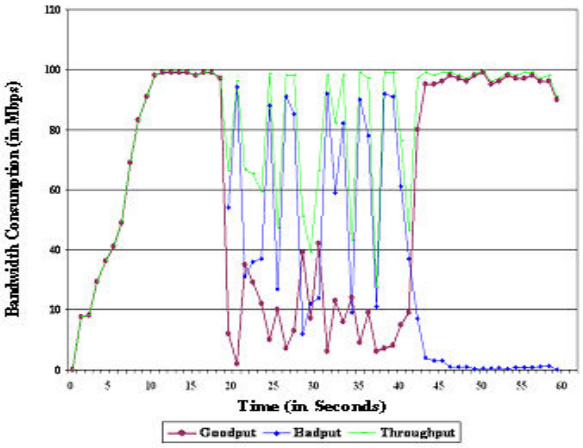

Figure 6: Measurement of Goodput, Badput and Throughput Without Attack, With TCP Pulse Attack and With Defense Applied

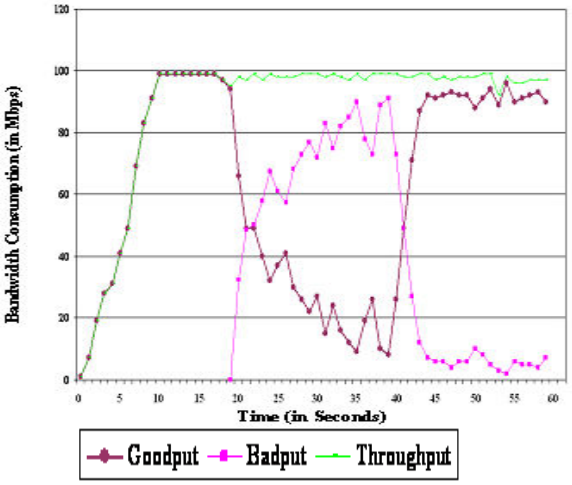

Figure 7: Measurement of Goodput, Badput and Throughput Without Attack, With TCP Ramp Attack and With Defense Applied

C. No. of Legitimate Packets Dropped

From 0s to 20s Flash Event has been shown to show little drop of packets in normal case. The attack is launched from 20s to 60s. During this time, due to congestion at bandwidth, lots of legitimate packets are dropped as shown in figure 8. Hence service is denied to the legitimate clients. After the defense is applied at 40s the drop rate reduces to a great extent. Thus, we can measure the performance as no. of packets of clients being dropped due to congestion on network bandwidth.

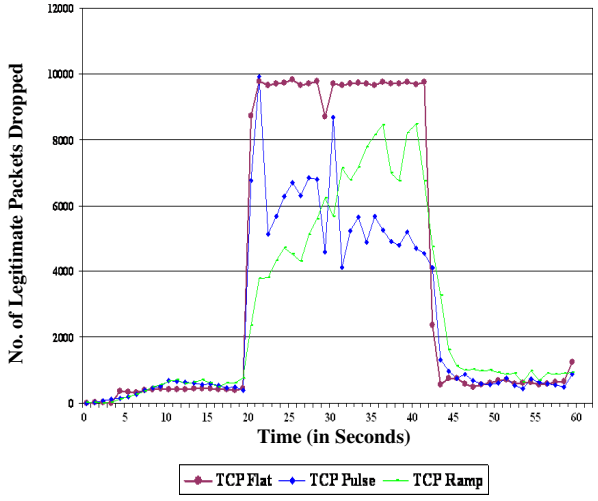

Figure 8: No. of Legitimate Packets dropped due to Flash Event and Due to different attacks

D. Communication overhead

The time spent on communicating with gateway and authenticating the legitimate user is termed as communication overhead. We have tried our best to keep it as much minimum as possible, The Gateways' CPU takes approximate 0.7µSeconds to check the failure threshold, 10.8µSeconds to check the SYN cookie, 8.2µSeconds to process the Gateway Packet and 32.1µSeconds to serve the graphical puzzle to clients. The communication overhead increases shortly in case of flash crowd or in case of attack as shown in figure 9.

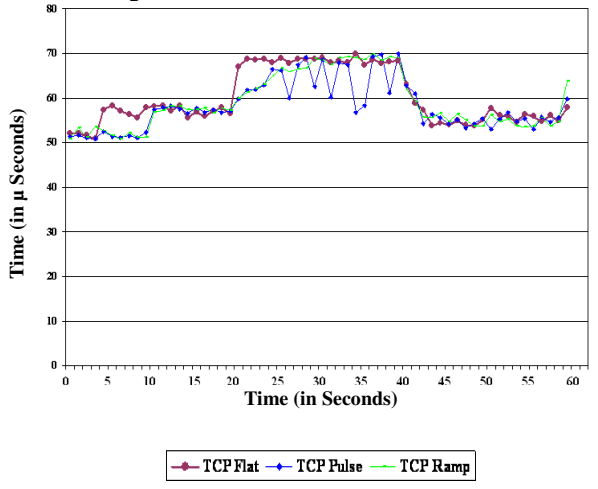

Figure 9: Communication Overhead of Legitimate Users while communicating with Gateway at edge router.

VIII. CONCLUSION AND FUTURE WORK

There are various defense mechanisms available in related work for measuring impact of DDoS Attacks, but existing defense are implemented at Source or Destination network. Those techniques have several limitations due to openness and vulnerabilities in the architecture of internet and they are unable to defend the bandwidth floods [27] [28] [29] [21] that mimic flash crowds. Few defense mechanisms have been employed at intermediate networks also, but those mechanisms increase the job of routers, because of their complex implementation, which is not acceptable due to high QoS requirements. This topology overcomes the Limitations of those mechanisms by implementing the defense mechanism only at edge routers, also the job of access and core routers remains to receiving and forwarding the packets to fulfil the QoS requirements . We evaluate our simulations on NS-2.

A Little Overhead for the Web Servers Still remains, if Attackers Intelligently Make the Vulnerable Legitimate Machines work as Zombies, Because in this case Legitimate users will pass the test for Attackers and they unknowingly attack the Bandwidth and the Victim Servers. This Case is very much crucial and it is very difficult to recognize a legitimate user which is unknowingly sending attack packets.

Our Future work will address this problem in more detail and we plan to extend this defense technique to solve the above said problem.

Another threat to this technique is the increasing load on the ISPs' Boundaries from which Attack is being launched, Although we have emphasized on keeping the QoS above the threshold by limiting the No. of Unsuccessful requests by using counters but still this may little slow down the performance due to which ISPs' don't agree to install Exit Control Gateways at their Edge Routers.

Future work will fold in more topology information and vulnerability information also. We are investigating several important questions that still need to be addressed. These include the authentication period and communication overhead among gateways, in case of Flash Crowd. We also plan to validate this scheme by running them on real attack data sets.

IX. REFERENCES

[1] J. McCumber, "Information System Security: A Comprehensive Model", "Proceedings of the 14th National Computer Security Conference, Baltimore", 1991, MD, USA.

[2] J. Kurose, and K. W. Ross, "A Top-Down Approach Featuring the Internet", Computer Networking pp 605-607. Second Edition, Addison Wesley, 2002

[3] P.G Neumann, "Denial-of-Service Attacks", Communications of the ACM, Volume 43, no. 4, pp. 136-136.

[4] D.L Cook, W. G. Morein, A.D. Keromytis, V. Misra, and D. Rubenstein, "WebSOS: protecting web servers from DDoS attacks". 11th IEEE International Conference on networks (ICON), pp. 461 – 466, 2003.

[5] C. Papadopoulos, R. Lindell, J. Mehringer, A. Hussain, R. Govindan, "COSSACK: Coordinated Suppression of Simultaneous Attacks".

[6] J. Mirkovic, "D-WARD: Source-End Defense Against Distributed Denial-of-service Attacks", Ph.D. Thesis, University of California, Los Angeles, 2003.
[7] J. Mirkovic and P. Reiher, "A Taxonomy of DDoS Attack and DDoS Defense Mechanisms", ACM SIGCOMM Computer Communications Review, Volume 34, Issue 2, pp. 39-53, April, 2004.
[8] T. Peng, C. Leckie, and K. Ramamohanarao, "Survey of Network-Based Defense Mechanisms Countering the DoS and DDoS Problems", ACM Computing Surveys, Vol. 39, No. 1, Article 3, April 2007.
[9] M. Kisimoto, "Studies on Congestion Control Mechanisms in the Internet – AIMD-based Window Flow Control Mechanism and Active Queue Management Mechanism", Master Thesis, Osaka University, 2003
[10] S. Floyd and K. Fall, "Router Mechanisms to Support End-to-End Congestion Control," Lawrence Berkeley Laboratories Technical Report, 1997.
[11] S. Bellovin, "ICMP traceback messages", Internet Drafts: draft-bellovinitrace-00.txt.
[12] Alex C. Snoeren, C. Partridge, Luis A. Sanchez, Christine E. Jones, F. Tchakountio, Stephen T. Kent, W. T. Strayer, "Hash-Based IP Traceback", Proceedings of ACM Sigcomm 2001. San Diego, CA 2001.
[13] Savage, S., Weatherall, D., Karlin, A., Anderson, T., "Practical Network Support for IP Traceback", Proceedings of Sigcomm 2000.
[14] Song, D., Adrian, P., "Advanced and Authenticated Marking Schemes for IP Traceback", Technical Report No. UCB/CSD-00-1107, University of California at Berkeley, June 2000.
[15] http://www.darpa.mil/ito/psum2000/J910-.html
[16] Park, K., Lee, H., "On the effectiveness of probabilistic packet marking for IP traceback under denial of service attack", Technical Report CSD-TR 00-013, Purdue University, June 2000.
[17] http://www.nanog.org/mtg-0006/savage.html
[18] Stone, R., "CenterTrack: An IP Overlay Network for Tracking DoS Floods", October 1999. http://www.nanog.org/mtg-9910/robert.html
[19] Ioannidis, J. & Bellovin, S. M. (2002), "Implementing pushback: Router-based defense against DDoS attacks", in 'Proceedings of Network and Distributed System Security Symposium, NDSS '02', Reston, VA, USA, pp. 100–108.
[20] Xuan, D., Bettati, R. & Zhao, W. (June 2001), "A gateway-based defense system for distributed dos attacks in high-speed networks", in 'in Proceedings of 2001 IEEE Workshop on Information Assurance and Security'.
[21] A. Keromytis, V. Misra, and D. Rubenstein, "SOS: Secure Overlay Services", In ACM SIGCOMM, 2002.
[22] Song, D. X. & Perrig, A. (2001), "Advanced and authenticated marking schemes for IP traceback", in"Proceedings of IEEE Infocomm", Vol. 2, Anchorage, Alaska, USA, pp. 878–886.
[23] L. von Ahn et al, "Captcha: Using Hard AI Problems for Security", In EUROCRYPT, 2003.
[24] CERT Incident Note IN-2004-01W32/Novarg, a Virus, 2004.
[25] V. Paxson, "An Analysis of Using Reflectors for Distributed Denial-of-Service Attacks", ACM CCR, 2001.
[26] J. Leyden, East European Gangs in Online Protection Racket 2003. ww.theregister.co.uk/2003/11/12/east_european_gangs_in_online
[27] H. Jamjoom, and K. G. Shin, "Persistent Dropping: An Efficient Control of Traffic", In ACM SIGCOMM, 2003.
[28] T. Anderson. T. Roscoe, and D. Wetherall, "Preventing Internet Denial-of-Service with Capabilities", In HotNets, 2003.
[29] T. Gil and M. Poletto, "MULTOPS: A Data-Structure for Bandwidth Attack Detection", In USENIX Security, 2001.
[30] R. Mahajan et al., "Controlling High Bandwidth Aggregates in the Network", CCR, 2002.

AUTHOR'S PROFILE

Rashpinder Pal is an M. Tech student in Department of Computer Science & Engineering at Bhai Maha Singh College of Engineering, Sri Muktsar Sahib Punjab, India. He has done his B. Tech. Computer Science and Engineering from Giani Zail Singh College of Engineering and Technology, Bathinda in 2008. His research interests include Network Security, Digital Image Processing and Mobile Databases.

Sunil Kumar is an M. Tech in Computer Science & Engineering from Bhai Maha Singh College of Engineering, Sri Muktsar Sahib Punjab, India. He has done his B.E. Computer Science and Engineering from Career Institute of Technology and Management, Faridabad in 2008. His research interests include DDoS Defenses, Information Security, Performance of Web Servers, and Design and Analysis of Algorithms.

Mandeep Singh is an M. Tech in Computer Science & Engineering from Guru Nanak Engineering College, Ludhiana Punjab, India.. Currently he is working as an Assistant Professor in Department of Computer Science & Engineering at Bhai Maha Singh College of Engineering, Sri Muktsar Sahib Punjab, India. His research interests include Digital Image Processing, DDoS Attack Impact Measurement and Defenses and MATLAB.