Mitigating DDoS using Threshold-based Filtering in Collaboration with Capability Mechanisms

Shubha Mishra
PG Scholar
Maulana Azad National Institute of Technology
Bhopal, India

R. K. Pateriya, Ph.D
Associate Professor
Maulana Azad National Institute of Technology,
Bhopal, India

ABSTRACT
Capability based approaches have been a major area of work since long time. They are robust against address spoofing attacks. However, they are vulnerable to a new type of attack called Denial-of-Capability attack. Also, bandwidth flooding is another serious issue. This article proposed a novel approach for collaboration of capability with a filtering mechanism. Dynamic threshold for traffic monitoring, implemented over underlying basic capability approach is an effective attempt to mitigate these two major vulnerabilities. A detailed framework is discussed in this research work along with estimation of the expected latency. Essential algorithms are provided for implementation of the approach. The approach is an effective key to handle loopholes in capability techniques. Since, no standalone solution exists for DDoS mitigation; this work provides a collaborative defense, thereby, enhancing robustness against them.

General Terms
Threshold-based filtering, Regular and Request traffic, Packet analysis, Attacker, Colluder and legitimate traffic.

Keywords
Capability and Filtering based mechanisms, Bandwidth flooding attack.

1. INTRODUCTION
Internet was originally developed with an aim to provide an open, effortless and timely communication and services to all. However, with its growth, security grew as a challenging factor in offering the services unhindered and in a smooth way. Denial of Service (DoS) attacks had been a very serious concern since their origin, rendering several networks and communication mediums unsafe. Distributed Denial of Service (DDoS) attacks grants the attackers some additional privileges by offering a shield of innocent users behind which the real Their motives vary from minor revenge or thrill to a critical extent and so do their impacts. Some of the real scenarios where this attack is most frequently performed involve political rivalries, business competitors aiming to block trustworthy customers of each other, banking and terrorism. Where in some cases their impact is negligible, in some others the chaos caused may be dangerously alarming. For example on February 9, 2000, Yahoo, eBay, Amazon.com, E*Trade, ZDnet, Buy.com, the FBI, and several other websites faced DDoS attacks resulting in substantial damage and inconvenience [1]. From December 2005 to January 2006, 1,500 separate IP addresses were victims of DDoS attacks, with some attacks using traffic rates as high as 10 gigabits per second [2][3].

As the problem is not new, researchers have already been trying to tackle the problem in the best effective way possible. Efforts have been made not only in developing strong detection techniques but also, in the deployment of effective attack avoidance and prevention mechanisms. But, it is high time to realize that none of the works have been highly robust against this attack environment. Multiple tools and varied test bed environments are now freely available to support real-world testing conditions. Still, the prime challenge is to differentiate between attack traffic and legitimate data with accuracy. DDoS attacks are not restricted to misusing network weaknesses but, could target vulnerable protocols and applications as well. Deployment of security mechanisms may vary in their positional and functional behavior but, at last, their ultimate goal remains the same.

In this article, a framework is proposed which would be an effective step in mitigating DDoS attacks. The prime objective of this proposed approach is to effectively handle Denial-of-Capability attacks and bandwidth flooding attacks which largely exploit the vulnerabilities of capability based mechanisms. The collaboration of threshold-based filtering with the existing capability techniques provides this approach as a possible effective solution to tackle the Denial-of-Capability (DoC) and bandwidth flooding attacks. The efficient use of packet analyzer software installed on the filter offer better opportunity to analyze the packets and detect any suspected source. The proposed approach is also an easy, standalone employment solution for SYN flooding attack and under implementation of any other connection-oriented protocols.

In Section-2, a brief overview on the related research work in this field is provided. Section-3 discusses the problem statement and essential assumptions for our framework. Followed by this, Section-4 provides an overview of capability-based approach and a detailed layout and functionality of the proposed architecture along with supporting algorithms. Section-5 provides a discussion on issues for this approach followed by comparative analysis on filtering and capability mechanisms and the need for such a collaborative approach. Lastly, Section-6 concludes the article following the future scope of this approach.

2. RELATED WORK
DDoS attacks focus mainly on some of the weaknesses either at the network level or at the application level so as to target the victim. In short, the victim of these attacks may be a single host, a group of hosts or an entire network and the attack medium may be the bandwidth depletion for the victim, protocol exploitation or applications on which a user is most frequently dependent. Solutions found for this problem may be categorized broadly into three categories: *a.) Proactive Mechanisms, b.) Reactive Mechanisms and c.) Post-attack*

Analysis [1]. The wide range of techniques developed may belong to one or more of these categories.

Ingress and Egress Filtering [4] [5], maintains a set of permissible IP addresses from where packet is accepted into the network and out of the network. This could be implemented using Access Control Lists (ACLs) or by the information from Routing Information Base (RIB). This approach demands consistency in updating the list of permitted set, which may need to be done manually at times. Also, this mechanism is not secure against intra-network malicious hosts.

John Ioannidis and Bellovin [6], proposed a rate-limiting based filtering mechanism called Pushback, which starts close to the victim and gradually shifts upwards. The downstream router informs upstream links about suspected attack flow which is then dropped for a specific time period depending upon the rate-limiter by the upper level routers, gradually moving close to the attack source. Though the mechanism is effective on a large-scale but, it relies on destination to differentiate between attack and legitimate traffic. STOPIT [7] and AITF [8] are other filtering techniques that install filters for a limited period to block a suspected host. These methods demand support from destination networks and does not offer any economic incentive, thereby restricting their large-scale deployment.

Another class of defense mechanisms includes Capability-based approach. PORTCULLIS [9], SIFF [10] and TVA [11] are some of these where capabilities are used for effective identification of legitimate user from attacker. Their major drawback is the difficulty of their deployment on a large-scale. Also, secure transfer of capabilities is another challenging issue. However, these methods have been highly effective in mitigation of flooding attacks. SOS [13] and MAYDAY [14] are techniques based on use of secure overlay networks for effective protection of hosts. Besides these, numerous other methods exist some of which offer greater economic incentives while, others offer a more robust environment for secure communication.

Some methods rely on packet marking to perform IP traceback [13], intended to trace the attack target accurately. Strong hashing and cryptographic algorithms have been added to defense mechanisms with the aim to achieve secure message exchanges during attack scenarios. This, however, considerably increased the computational requirement and complexity of approaches supported by these algorithms. Till date, none of the techniques could be titled as a panacea to all of the above discussed DDoS related issues. The next Section provides the necessary assumptions for our proposal along with the underlying problem.

3. PROBLEM STATEMENT AND ASSUMPTIONS

Our proposal aims at offering a more robust technique to handle DDoS attacks and provide a more secure communication environment for legitimate users. This approach is mainly focused towards bandwidth flooding attack and the attempt is overcome the drawbacks of capability techniques. The terms capability and privileges are inter-changeably used throughout the article. Following two attacks are the serious issue for concern in capability techniques:

i.) ***Denial-of-Capability (DoC) attacks:*** *Capability techniques aimed at mitigating DDoS attacks by allowing destination to decide on the set of permitted senders. This, however, led to the restriction of legitimate user from obtaining timely privileges leading to origin of DoC attack.*

While some techniques require users to solve puzzles as per their resource availability, others demand hosts to request the privilege. In either case, a genuine host with limited resources may be easily out-numbered by the attackers. Our approach handles this issue by fixing the threshold value for request traffic type based on the restricted bandwidth allocation for this type of traffic. Actions are performed whenever this value is crossed by the total number of request packets (from all the possible hosts requesting capabilities). This grants equal opportunity to all the hosts for gaining access to capability and drags the impact of DoC attacks on legitimate user to a negligible level.

ii.) ***Bandwidth flooding attacks:*** *Most of the capability-based techniques are vulnerable to this attack type. When colluders (hosts that help attackers by granting privileges intentionally) share common bottleneck bandwidth with the attacker, two-way flooding attack (a type of bandwidth flooding) could be easily performed to target the victim. The colluders could continuously grant capabilities with larger timestamps to the attackers. This may lead to flooding of the bottleneck bandwidth, thereby denying access to the victim.* Figure-1 provides a network topology overview for above discussed attacks.

A threshold value is also decided for regular (packets sent after obtaining privileges) traffic based on the greater share of bandwidth (about 95-98 percent in most approaches) allocated to this traffic type. The necessary actions performed whenever threshold value is reached under each case are discussed later.

A very simple, but essential assumption for this framework is as follows:

It is assumed, that all the traffic passes through the edge routers which embed the proposed filtering algorithms and an efficient packet analyzer tool. Some already existing packet analyzer tools are suggested later along with the purpose of these tools in this approach. It is not mandatory to implement this mechanism on the border routers. Installation may be done on a separate network device such as, a router or even a switch to reduce the overhead of border routers.

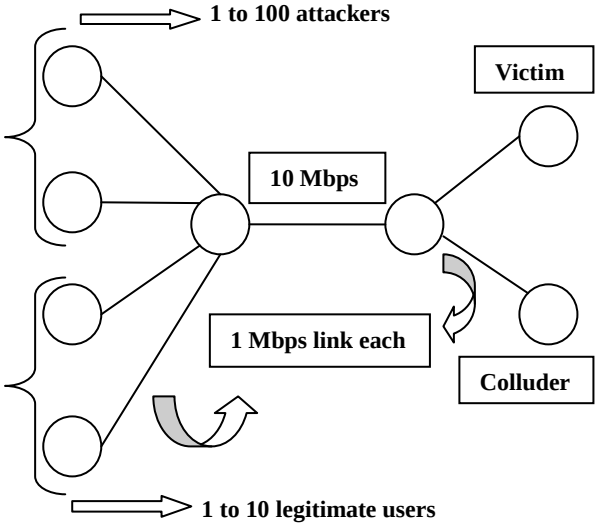

Figure-1: A type of network topology for DDoS attacks [11].

4. PROPOSED FRAMEWORK

4.1 Basic Underlying Capability-based Architecture

Capability based techniques are based on the concept of granting privileges to the sender, which act as the primary element in distinguishing between legitimate and attack traffic. These capabilities in some techniques could be directly requested by the hosts which desire to communicate to a particular destination, using the piggybacking concept (for example accompanying request packet with SYN request and receiving response with the SYN ACK packets). However, in some cases, hosts require to solve puzzles, where their level of difficulty decides the priority of their request. Though the working approach of each of these techniques may vary, there lies considerable resemblance in their capability types and their use.

Time-stamp (8 bits)	hash (src IP, dest IP, T, secret) (56 bits)

Pre-capability appended by routers

Time-stamp (8 bits)	hash (pre-capability, N, T) (56 bits)

Capability appended by hosts

Figure2: Format of capabilities in TVA [11] and capability- based mechanisms.

Figure-2 describes the packets appended by the routers, with the pre-capabilities and capabilities. These specify the definite timestamp and the amount of data a host is privileged to transmit to that specific destination. The major limitation of capability-based mechanism is the maximum area of their coverage. These mechanisms are employable only up to a confined range for their maximum efficiency due to their complexity. Also, in most of the techniques, DoC is a major issue of concern. This approach acts as a solution to mitigate these problems to some extent. With the use of filtering mechanism along with capabilities, equal opportunities are provided to all the hosts for communication establishment.

4.2 Design Overview

Figure-1 provides a topological view of the underlying architecture, which however, is only one type of the multiple topologies possible, on which this approach could be effectively employed. The basic functionality of capability-based approach has been discussed previously. A filter is installed on the edge router or even separately with a packet analyzer tool (such as, *Wireshark, Netflow, tcpdump and dSniff*) [12]. This filter continuously monitors the border router traffic at regular intervals (say, every one second). Every packet passes through this filter based router. The threshold value calculated for each class is then compared with the total traffic of its respective type. If the traffic passing is greater than threshold value, packets are dropped linearly and sent to packet analyzer tool. This process is repeated until; the total traffic for each class is below its respective threshold value. The thresholds are also dynamically changing values and they depend on the amount of traffic of their respective types per unit time. Therefore, they must be recalculated and updated at regular interval *'T'*.

The packet analyzer software monitors each incoming and outgoing packet on a specified port and examines them on the basis of some metrics such as, sender and destination ports, sender and destination addresses and also monitors bandwidth utilization and network usage. This analysis provides a valuable opportunity to detect malicious hosts and attack scenarios beforehand. The suspected hosts may be subsequently blocked for some pre-defined time period. This grants immense space for re-evaluation of network performance within the next interval, thereby easing the detection of actual attack source (or at the least, direction of attack flow).

Packets are linearly dropped to avoid significant dropping of legitimate traffic, as this may not be permissible in some networks. It may not be effective when the traffic outreaches the threshold value considerably. However, since this depends mostly on the interval for monitoring, it is not a serious issue. The reason behind, is that the interval need not be static; it may be varied with the dynamic nature of the attack or as per the network requirements.

The approach offers robustness against Denial-of-Capability attack and bandwidth flooding (especially two-way flooding attack). Negligible impact would be on the legitimate host even under attack environment, irrespective of whether it is on the sender side or the receiver side.

4.3 Dynamic Threshold based Filtering

The entire traffic is broadly divided into two main categories: *a.) Request traffic*, and *b.) Regular traffic.* However, in some techniques such as TVA [11], there is another lowest priority data called *Legacy traffic.* Regular packets gain maximum priority followed by request packets. Since our approach is a work on existing capability-based mechanisms in general, the threshold values have been considered only for these two prime classes of traffic. However, the approach is fairly adjustable to bandwidth allocation for any number of traffic types.

Efficiency of this approach depends, on the accuracy of threshold value for each traffic type and the decisive actions performed once the respective threshold values are crossed, as its major part. The threshold value depends on the amount of bandwidth allocated as per the priority of data as a prime factor. Table-1 provides the names of parameters used along with the symbols used for their representation.

The threshold and bandwidth are represented as 'Th_c' and 'B_c' for their respective traffic classes 'c'. Also, the average packet size is taken as *'*P_c*'*. These details are necessary in cases where the number of traffic classes is more than two. We may assume that packet size is usually almost equal. However, this may not be true always and so, to increase the accuracy average packet size is considered as a metric. Similarly, parameter *'N',* describing the number of traffic classes is also included for the aforementioned reason. Their exists processing delay 'D' for all traffic types, which is the delay induced due to classification of packets to identify the class it belongs to, by examining its header and then placing it on the queue. Though, this value may be very small, it is considered for accurate value of expected latency due to queuing.

Table-1: Parameters used for determining threshold values

PARAMETERS USED	SYMBOLS
Total outbound link capacity	C_t
Number of classes or traffic types	N (here only 2 are considered)
Latency induced by queuing	L
Processing delay per packet (equal for all packets)	D
Total number of packets passed through the router	N_{tot}
Threshold for request traffic	Th_r
Threshold for regular (normal) traffic	Th_n
Bandwidth allocated for request traffic	B_r
Bandwidth allocated for regular traffic	B_n
Number of connected hosts at time 't'	N^*
Number of active valid connections at time 't'	N'
Average packet size for request packets	P_r
Average packet size for regular packets	P_n

B_r = Maximum percentage of bandwidth (C_t) allocated to request traffic.
B_n = Maximum percentage of bandwidth (C_t) allocated to regular traffic.
The threshold values for request and regular traffic are based on the share of the bandwidth allocated to that traffic type. Following equations determine the dynamic threshold value for request and regular traffic. These values are recalculated at every 'T' interval as they depend on network connections and hosts, which may or may not be constant.

$$Th_r = \frac{B_r}{N^*} + 1 \quad \text{- (1.)}$$

$$Th_n = \frac{B_n}{N'} + 1 \quad \text{- (2.)}$$

The total outgoing link capacity is the sum of all the bandwidths allocated to each traffic type. In our approach only two traffic types are considered.

$$C_t = \sum_{c=1}^{N} (B_c) \quad \text{- (3.)}$$

$$C_t = B_r + B_n \quad \text{- (4.)}$$

4.3.1 Latency

Latency refers to the time taken for a packet to traverse from its source to destination. In a network, this value depends on several factors which may introduce a finite amount of delay. These factors include speed of the communication medium, delay due to intermediate routing devices such as, packet processing delays, buffering and queuing delays [16]. However, these delays could be reduced using some of the enhancing techniques for efficient buffer and queue management [15] [17] [18].
In this network environment, since all the packets share the same queue, the latency will be the same for all traffic classes. *The expected latency specifies the maximum permissible limit on the latency introduced due to queuing.* Therefore, this value gives the worst-case latency, which will be reached when both the traffic types reach their maximum permissible threshold for queue occupancy. This latency is given as a function of maximum number of bytes per traffic class enqueued and the outbound bandwidth plus the overall processing delay introduced, in an average worst-case scenario.

$$L = \{[(\sum_{c=1}^{N} Th_c * P_c)/C_t] + (D * N_{tot})\} \quad \text{- (5.)}$$

Considering only request and regular traffic, following equation is derived to determine maximum expected permissible latency.

$$L = \{\{[(Th_r * P_r) + (Th_n P_n)]C_t\} + (D * N_{tot})\} \quad \text{- (6.)}$$

4.4 Algorithms

This section provides the set of algorithms as follows: a.) Threshold setting and update. b.) Flow monitoring and decision on packet handling. c.) Action performed on dropped packets.

4.4.1 Algorithm for setting threshold values.

Input: Time period for threshold update (T), Bandwidth share allocated, N*, N' (table-1.)

```
DynamicThrshldCalc_Algo(Br, Bn,
                        N*, N',T)
 {
    while(T)
    do
          Thr = Br/N* +1;
  //Request Threshold
          Thn = Bn/N' +1;
  //Regular Threshold
           T--;
  }
```

Figure 3: Algorithm for threshold calculation.

4.4.2 Algorithm for traffic monitoring

Input: Packet, Threshold values, Total number of packets of each type.

```
FlowMonitoring_Algo(pckt, Thr, Thn,
                    Ptotrqst, Ptotreg )
{
  /* Ptotrqst, is total no. of request packets
received.  Ptotreg is total no. of regular
packets received. Thr and Thn are
threshold values for each.*/

if (pckt == rqstpckt)
      if (Ptotrqst < Thr) then
         enqueue(pckt);
      else
         drop(pckt);      /* send pckt to
                          packet  analyzer*/

   else if (pckt == regpckt)
       if (Ptotreg < Thn) then
         enqueue(pckt);
       else
         drop(pckt);
}
```

Figure 4: Algorithm for traffic monitoring.

4.4.3 Algorithm for handling dropped packets and analyzing them.

The packets are analyzed by a packet analyzer tool based on various parameters. The term suspected packet in the algorithm refers to packet in which parameters are identified as malicious by the packet analyzer.

Input: Dropped packets.

```
DroppedPcktAnalysis_Algo(pckt)
For all dropped pckts:
do
{
  analyze(pckt); //based on some metrics
  chk(src_id, dest_id,flow_info);
   {
    if(isSuspected(pckt))
     {
        report pckt info;     // to the router
        block(pcktsrc or pcktdest or both);
      }
    }
  }
```

Figure 5: Algorithm for handling dropped packets and analyzing them

5. DISCUSSION

This proposed work offers a feasible solution to mitigate Distributed Denial of Service attacks further. Capability mechanisms are robust against these attacks. However, they are prone to two major attacks and are found very less effective in handling them. These are Denial-of-Capability attack and two-way bandwidth flooding attacks which are already discussed in Section-3.

5.1.1 Issues.

There are some major issues which may affect the deployment of this proposed architecture. *First,* timely update of threshold values must be performed so as to obtain optimal performance. This parameter is kept variable to handle this issue effectively. Since network conditions may alter from time to time, the period after which the threshold and analysis of dropped packets will be carried is dynamically variable. *Second,* installation of filter at proper location is another challenge for this approach. For bandwidth flooding attacks, defense must be as close to the source as possible to reduce collateral damage and attack impact to minimum. Hence, deploying this at the edge routers offers maximum robustness. DoC can also be best restricted when nearest to the attack source.

5.1.2 Analysis

Filtering and Capability techniques have been compared under various scenarios by many researchers as discussed in previous sections. Table 2 summarizes the comparative study on Capability and Filtering mechanisms based on several parameters. While for some parameters capabilities are better performers, for others Filtering are superior ones. However, neither of them is a standalone defense mechanism, capable of completely outnumbering the other when all the parameters are considered.

Table 2: Summary of comparison between Capability and Filtering mechanisms

Parameters for comparison	**Capability mechanisms**	**Filtering mechanisms**
Deployment position	Hybrid (source to destination)	Source or destination/Hybrid
Scalability	Small-Medium	Upto Large scale
Performance	High if capabilities are secure.	High only upto limited attack intensity
Complexity	High	Low - Medium
Dependability	On path routers	On routers, ISPs
Cost effectiveness	Low (much costly)	Medium

The graph shown in figure 6 depicts the difference in the effectiveness of Capability and Filtering mechanisms under different attack intensities. Attack power (also called attack intensity in this article), in general, is considered as a measurement parameter based on number of attackers and size of packet. The effectiveness is a secondary term based on the number successful (completed) TCP transfers by legitimate users. Based on these parameters, the above graph proves that

initially, when attack intensity is low, both the mechanisms are effective against the attack.

However, filters are slightly more effective then capabilities. With the rise in the attacker's power, the effectiveness of both the approaches declines drastically. It is clearly shown that filters are visibly ineffective due to the delay or inability of their timely installation. The scenario when the attacker is able to gain unhindered long-term access to capabilities, this approach also becomes ineffective for defense. Capability mechanisms show a constant performance for larger range of attack power as compared to its counterpart. However, with the continuous increase in the attack intensity, both the mechanisms fail in defending against the attack. It is hence, clear that a fail-safe mechanism is the need of the hour. An approach which could effectively handle high attack intensities, thereby, offering a more robust environment for secure communication is vitally essential.

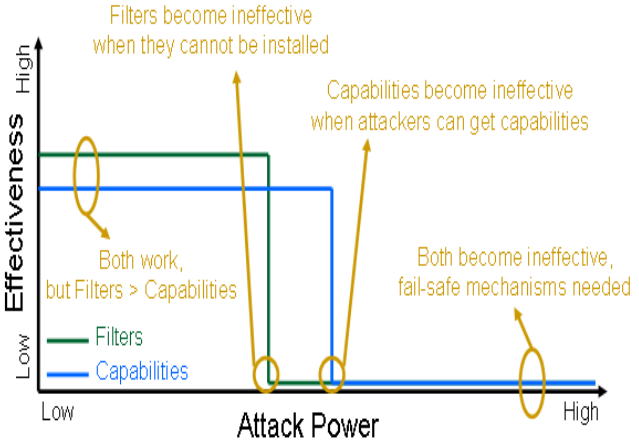

Figure 6: Effectiveness of Capability and Filtering mechanisms under different attack intensities [19].

6. CONCLUSION AND FUTURE WORK

Denial of service had always been the most dangerous attack forms among the bulk of other existing ones. Their distributed nature had well added strength thereby, amplifying their impact and rendering the task of their mitigation more tedious. Out of the variety of available mechanisms aimed at tackling the problem, none could be stated as a stand-alone solution. This article aimed at offering another stepping stone in mitigation of the problem further, to some extent.

The collaboration of the capability and filtering technique is a far more robust approach than each of these alone. This article aimed at providing one such collaborative mitigation mechanism. Dynamic threshold based filtering enhances DDoS defense by securing the vulnerabilities of capability mechanisms. Therefore, this approach might prove as a more secure strategy for protecting legitimate users even under attack environments.

The future scope is to test the framework in a real-world environment and simulation tools such as, NS-2. Currently, we are testing the approach under different attack scenarios to estimate its effectiveness and reliability, which might prove as its incentives for real-world implementation. Collaboration of two or more attack mechanisms might offer the best possible security and fastest recovery from attacks.

7. REFERENCES

[1] Garber, L. "Denial-of-service attacks rip the Internet", IEEE Journal on Computer, vol. 33 (4), pp. 12-17, 2000.

[2] Scalzo, F. 2006. "Recent dns reactor attacks", VeriSign, [online] http://www.nanog.org/mtg-0606/pdf/frank-scalzo.pdf.

[3] Vaughn, R. and Evron, G. 2006. "DNS amplification attacks", [online] http: // www.isotf.org/news/DNS-Amplification-Attacks.pdf.

[4] P. Ferguson, and D. Senie, Network Ingress Filtering: Defeating Denial of Service Attacks that employ IP source address spoofing, Internet RFC 2827, 2000.

[5] Egress Filtering, [online] http: // en.wikipedia.org/wiki/Egress_Filtering.

[6] John Ioannidis and Steven M. Bellovin, "Implementing Pushback: Router-Based Defense Against DDoS Attacks", in Proc. of Network and Distributed System Security Symposium, 2002.

[7] Xin Liu, Xiaowei Yang and Yanbin Lu, "To Filter or to Authorize: Network-Layer DoS Defense against Multimillion-node Botnets", ACM SIGCOMM'08, August 17–22, 2008, Seattle, Washington, USA.

[8] K. Argyraki and D. R. Cheriton, Scalable network-layer defense against internet bandwidth-flooding attacks, IEEE/ACM Transaction Netw., 17(4), pp. 1284-1297, August 2009.

[9] B. Parno, D. Wendlandt, E. Shi, A. Perrig, B. Maggs, and Y.-C. Hu, "Portcullis:Protecting Connection Setup from Denial-of-Capability Attacks." , ACM SIGCOMM, 2007.

[10] A. Yaar, A. Perrig, and D. Song, "SIFF: a Stateless Internet Flow Filter to Mitigate DDoS Flooding Attacks", in Proc. 2004 IEEE Symposium on Security and Privacy, pp. 130-143, May 2004.

[11] X.Yang, D.Wetherall, and T.Anderson, "TVA: a DoS-limiting network architecture", IEEE/ACM Trans. Netw., vol. 16, no. 6, pp. 1267-1280, 2008.

[12] "Packet analyzer", [online] http://en.wikipedia.org/wiki/Packet_analyzer.

[13] A. John, and T. Sivakumar, "DDoS: Survey of Traceback Methods", International Journal of Recent Trends in Engineering ACEEE, Association of Computer Electronics & Electrical Engineers, vol. 1, no. 2, May 2009.

[14] P. Gupta and N. McKeown, Packet classification on multiple fields, in the Proceedings of ACM SIGCOMM'99, ACM, August 1999.

[15] R. Guerin, S. Kamat, V. Peris, and R. Rajan, Scalable QoS Provision Through Buffer Management, Proceedings of SIGCOMM'98,

[16] "Latency (engineering)", [online] http://en.wikipedia.org/wiki/Latency_(engineering).

[17] V. Firoiu, M. Borden, "A Study of Active Queue Management for Congestion Control", INFOCOMM 2000.

[18] R. Pan, B. Prabhakar, K. Psounis," CHOKe: A Stateless Active Queue Management Scheme for Approximating Fair Bandwidth Allocation", INFOCOMM'00.

[19] X. Yang, A DoS Limiting Network Architecture, [online] http://www.cs.duke.edu/nds/ddos/