RESEARCH ARTICLE OPEN ACCESS

Performance Analysis of RED & Robust RED

Arsh Arora[1], Lekha Bhambhu[2]
Research Scholar, HOD,
Department of Computer Science and Engineering
Jan Nayak Ch. Devilal College of Engineering, Sirsa
Haryana - India

ABSTRACT
Active Queue Management is a way to control Congestion. The existing RED (Random Early Detection) Active Queue Management algorithm and its variants are found vulnerable to emerging LDoS attacks. The RRED algorithm was proposed to improve TCP throughput under LDoS attacks by detecting and filtering out attack packet. We conduct a set of simulation to compare and evaluate the performance of RED and RRED under normal TCP traffic without LDoS attacks and With LDoS attacks. The results show that, RRED algorithm nearly fully preserves the TCP throughput in the presence of LDoS attacks. Simulation is done by using Network Simulator (NS2)

Keywords:- AQM, RED, RRED, DoS, LDoS, NS2

I. INTRODUCTION

An enhancement of the router based queue management is known as Active Queue management (AQM) [1]. Generally, In AQM schemes congestion is measured and control actions are taken. There are two approaches to measure congestion in AQM. (1) In **Flow based** congestion is observed and action is taken based on the packet arrival rate. (2) In **Queue based** congestion is measured by queue size and action is taken by maintaining a set of queues by Internet routers. The basic goal of all AQM techniques is to keep the average queue size in routers small to Controls average queue size to avoid global synchronization of TCP and absorbs bursts without dropping packets to prevent bias against bursty connections [7]. AQM also reduces the number of timeouts in TCP and take actions against misbehaving flows.

In past decades a few active queue management (AQM) algorithms such as Random Early Detection (RED) [2] and its variants have been proposed to improve the TCP performance in congestion. AQM algorithms are highly robust to diverse network conditions; most of them were designed without considering their robustness against network attacks, such as the Denial-of-Service (DoS) [10] attacks that have been considered as a major threat to Internet services now days. DoS attacks such as TCP SYN flood attacks, Internet Control Message Protocol (ICMP) flood attacks, ping attacks, smurf attacks, DNS flood attacks [11]. These attacks can be detected and recovered because they generate high rate broadcast of packets toward the target node. Recently low-rate DoS attack has been proposed in [5] that manipulate the TCP's retransmission timeout mechanism to bring down TCP throughput without being detected. RRED is proposed in [3] to thwart these LDoS attacks by detecting and filtering out LDoS attack packets.

In this paper we conduct simulation to evaluate the performances of RED and RRED active queue management algorithm under TCP traffic in absence of LDoS attacks and in presence of LDoS attacks in NS2[6].

II. LDOS ATTACKS

An LDoS attack is a kind of DoS attack, in which more number of packets are sent in a short period of time and it is repeated for several intervals. The packet sending rate is so high, so that it crosses the link capacity and hence congestion will occur in network. It is very difficult to identify LDoS attack, because of low average R_b rate is maintained during network congestion [5].

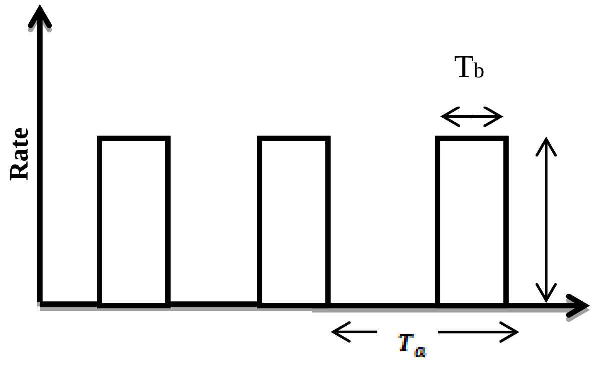

Fig.1: LDoS attack stream

In the above figure regarding LDoS attacks, where T_a the attacking period is, T_b is the attacking burst width and R_b is the attacking burst rate. The LDoS attack catches TCP's slow-time-scale dynamics of retransmission time out (RTO) mechanism to decrease TCP throughput [9].

An attacker can make a TCP flow to regularly enter a RTO state by pushing high-rate (R_b), but short duration burst (T_b), and repeating at slower RTO time-scales (T_a). A successful LDoS attack will have rate large enough to establish loss and duration of attack must be long enough to set up congestion but short enough to keep away from detection. LDoS attacks will decrease the performance of TCP traffic and web traffic. Low the RTT of packets, high the effect of attack. LDoS attack triggers unbalance in network load and also interrupts in internet routing. The LDoS attack will burn up recourses of the assigned server with only low-rate traffic; therefore server security schemes are violated.[5]

III. RANDOM EARLY DETECTION

Random Early Detection (RED) [2] was proposed by Floyd and Jacobson as an efficient congestion avoidance mechanism in the network routers/gateways. It also helps to prevent the global synchronization in the TCP connections sharing a congested router and to decrease the bias against bursty connections. It is assumed to solve the traditional problems of queue management techniques. It was an improvement over the previous techniques such as Random Drop and Early Random Drop [10]. RED use probabilistic discard methodology of queue fill before overflow conditions are reached. By detecting incipient congestion early and to convey congestion notification to the end-hosts, allowing them to decrease their transmission rates before queues in the network overflow and packets are dropped.

The RED gateway computes the average queue size by using a low pass filter along with an exponential weighted moving average. The average queue size is compared with two thresholds: a *minimum* and a *maximum* threshold. When the size of average queue is less than the minimum threshold, no packets are marked. When the size of average queue is greater than the maximum threshold, every arriving packet from gateway is marked. If marked packets are, in fact, dropped or if all source nodes are collaborative, this assures that the average queue size does not significantly exceed the maximum threshold.

When the average queue size is varying in between the minimum and maximum thresholds, each arriving packet is marked with a probability p_a, where p_a, is a function of the average queue size avg. Each time a packet is marked, the probability that a packet is marked from a particular link is roughly relative to that connection's share of the bandwidth at the gateway. The general RED algorithm is given below:

```
For each packet arrival
calculate the average queue size $avg$

if   $min_{th}$  $\leq avg < max_{th}$
calculate probability $p_a$
        with probability $p_a$:
                mark the arriving packet
else if $max_{th} \geq avg$

        mark the arriving packet
```

Fig. 2. General algorithm for RED gateways [2]

Thus, the RED gateway has two separate algorithms. One of those computes the average queue size determines the degree of burstiness that will be allowed in the gateway queue. And the other one calculates the packet-marking probability that determines how often the gateway marks packets; give the current level of congestion. The goal of gateway is to mark the packets fairly, in order to avoid biases and global synchronization, and also to mark packets sufficiently frequently to control the average queue size.

IV. ROBUST RANDOM EARLY DETECTION

RED can detect and respond to attacks those has high rate transmission of packets toward the targeted node, but it cannot detect congestion caused by short-term traffic load changes. In addition, it is well known that an appropriate tuning of RED parameters is not an easy task and may result in a non-stabilizing controls scheme. Robust random early detection (RRED) [3] is a queuing discipline for a network scheduler. The existing random early detection (RED) algorithm and its variants are found vulnerable to emerging attacks, especially the Low-rate Denial-of-Service attacks (LDoS). Experiments have confirmed that the existing RED-like algorithms are notably vulnerable under LDoS attacks due to the oscillating TCP queue size caused by the attacks. The Robust RED (RRED) algorithm was proposed to increase the efficiency of TCP throughput against LDoS attacks. The RRED algorithm consists of a new detection & filtering algorithm and a traditional RED algorithm. the RRED detects and filter out LDoS attack packets from incoming flows before they feed to the RED algorithm. RRED algorithm can significantly improve the performance of TCP under Low-rate denial-of-service attacks [5].

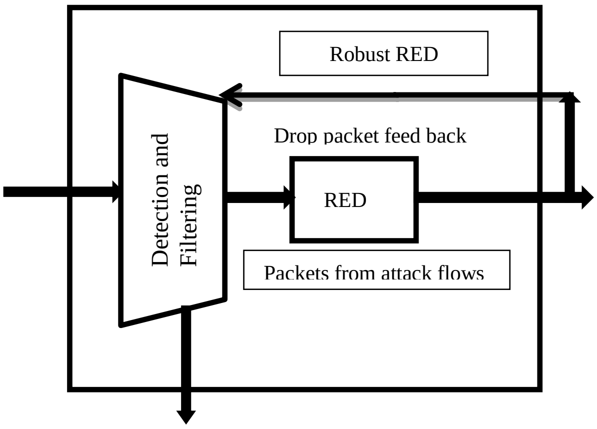

Fig.3 - Architecture of Robust RED

As shown in the fig. a detection and filter block is added in front of a regular RED block on a router. All the incoming packets are filtering out the LDoS attacks before they feed to RED. How to distinguish an attacking packet from normal TCP packets is critical in the RRED design.

Within a benign TCP flow, the sender will delay sending new packets if loss is detected (e.g., a packet is dropped). Consequently, a packet is suspected to be an attacking packet if it is sent within a short-range after a packet is dropped. This is the basic idea of the detection algorithm of Robust RED (RRED).

V. SIMULATION MODEL

A. *Simulation:*

We use the Network Simulator (NS_2). TheNS_2 is a discrete event simulator developed by the University of California at Berkeley and the Virtual Internetwork Tested (VINT) project. The NS_2 support two languages, system programming languages C++ for detail implementation and scripting languages TCL for configuring and experimenting with different parameters quickly. The NS_2 has all the essential features like abstraction, visualization, emulation, traffic and scenario generation [6].

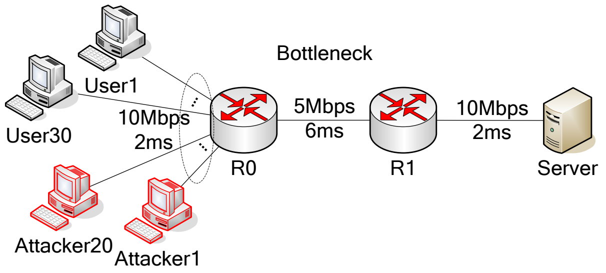

Fig. 4 Simulation topology

B. *Simulation model:*

A Simple network topology is chosen to make it easier to understand the congestion network environment. As shown in Fig. 4 The queue size of the bottleneck link is 50 packets. AQM algorithms are used on the bottleneck queue. A TCP (*Newreno*) based FTP flow with packet size of 1000 bytes is generated from each user (User 1 to User 30). LDoS traffic is generated from Attacker 1 to Attacker 20 by sending UDP packets with packet size of 50 bytes.

VI. SIMULATION RESULTS

Case 1: The average throughput versus attack peroid (T_a) with constant attack burst width (T_b) = 1ms and attack burst rate (R_b) = 0.25Mbps. Attack peroid (T_a) varies from 0.2s to 2s.

TABLE I
AVERAGE THROUGHPUT OF NORMAL TCP TRAFFIC THROUGH BOTTLENECK LINK WHEN THERE IS LDOS ATTACKS WITH ATTACK PERIOD

Time of T_a	RED	RRED
0.2	91.27	598.70
0.5	222.82	588.81
1.0	370.28	597.22
1.5	519.07	595.02
2	530.75	581.51

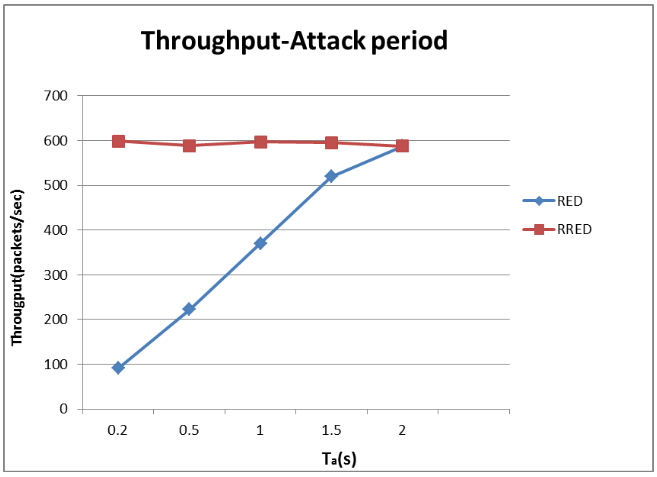

Fig. 5 Throughput-Varying Attack period Ta (s)

As we can see from the fig. 5, In case of attack period increases in the of normal TCP traffic through the bottleneck link when there is LDoS attack, as the attack period increases the throughput varies. So in this scenario, the performance of RRED is better than RED and is best at 0.2 seconds.

Case 2: The average throughput versus attack burst width (T_b) with constant attack period (T_a)= 1s and attack burst rate (R_b) = 0.25Mbps. Attack burst width (T_b) varies from 100ms to 600ms.

TABLE II
AVERAGE THROUGHPUT OF NORMAL TCP TRAFFIC THROUGH BOTTLENECK LINK WHEN THERE IS LDOS ATTACKS WITH ATTACK BURST WIDTH

Time of T_b	RED	RRED
100	536	499
200	597	513
300	424	499
400	394	499
500	324	499
600	293	499

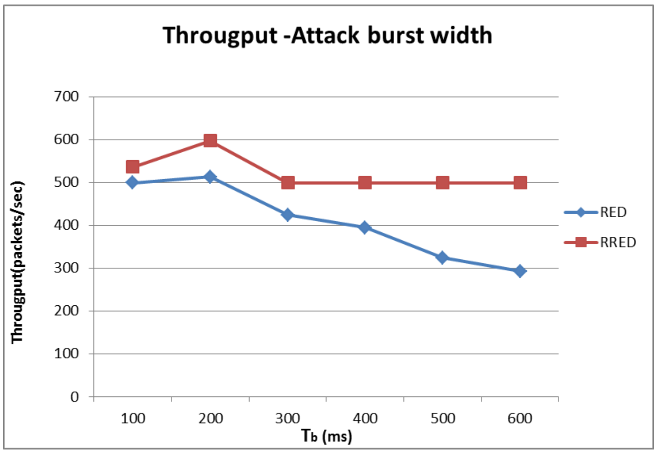

Fig. 6 Throughput-Varying Attack burst width Tb (ms)

As we can see from the fig. 6, In case of burst width increases in the of normal TCP traffic through the bottleneck link when there is LDoS attack, as the burst width increases the throughput varies. So in this scenario, the performance of RRED is better than RED and is best at 200 ms.

Case3 : The average throughput versus attack burst rate (R_b) with constant attack period (T_a)= 1s and attack burst rate (T_b) = 0.2ms. Attack burst rate (R_b) varies from 0.1 to 0.5 Mbps.

TABLE III
AVERAGE THROUGHPUT OF NORMAL TCP TRAFFIC THROUGH BOTTLENECK LINK WHEN THERE IS LDOS ATTACKS WITH ATTACK PERIOD

Time of Ta	RED	RRED
0.1	537	538
0.2	516	491
0.3	498	491
0.4	458	489
0.5	448	488

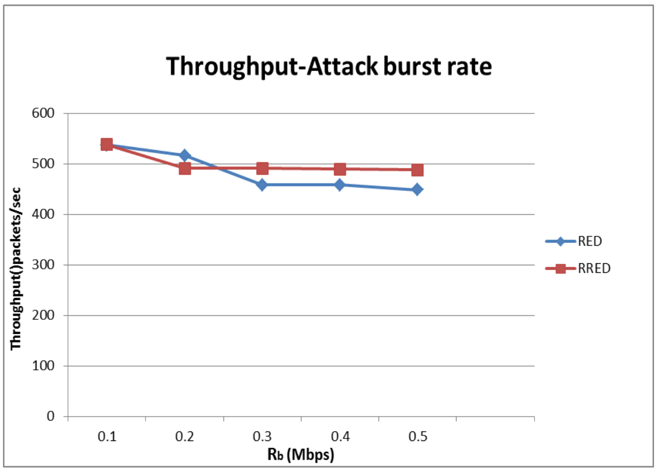

Fig. 7 Throughput-Varying Attack burst rate Rb (Mbps)

As we can see from the fig. 7, In case of attack burst rate increases in the of normal TCP traffic through the bottleneck link when there is LDoS attack, as the attack burst rate increases the throughput varies. So in this scenario, the performance of RRED is better than RED and is best at 0.1 sec.

For the three parameters of the LDoS attack, we choose Ta=1s since [5] reported that LDoS attacks with $Ta \approx 1s$ are most effective. Tb is set to 200ms and Rb is set as 0.15Mbps so that the aggregate Rb of 20 attackers is equal to the bottleneck bandwidth of the network (5Mbps). With these three parameters, we conduct three sets of experiments to evaluate and compare the performance of the AQM algorithms. For each set, we fix two values and vary the other value. For example, for set one, we vary Ta from 0.2 to 2 while fixing Tb and Rb. Varying these three parameters aims to investigate the robustness of the RRED algorithm if an attacker changes its resending behavior during an attack.

It is clear from the simulation that RRED performs better than RED.

VII. CONCLUSIONS

We have simulate both RED and Robust RED (RRED) in NS2. Results show that the RRED algorithm is (i) highly robust (ii) can improve the performance of normal TCP traffic through bottleneck link under LDoS attacks and (iii) obviously it performs better than RED

ACKNOWLEDGMENT

Special thank you goes to those who contributed to this paper:
Mrs. Lekha Bhambhu for his valuable comments and sharing her knowledge. Mr. Deepak Mehta & Mr. Gopal Sharma for their keen interest, expert guidance, valuable suggestions and persistent encouragement.

REFERENCES

[1] S. B. Braden, et al, Recommendations on Queue Management and Congestion Avoidance in the Internet, RFC 2309, April, 1998.

[2] Floyd, S., and Jacobson, V. (1993), Random Early Detection gateways for Congestion Avoidance V.1 N.4, August 1993, pp. 397-413..

[3] Changwang Z., Jianping Y., Zhiping C., and Weifeng C (2010), RRED: Robust RED Algorithm to Counter Low-Rate Denial-of-Service Attacks IEEE COMMUNICATIONS LETTERS, VOL. 14, NO. 5, MAY 2010

[4] H. V. Shashidhara, Dr. S. Balaji, "Low Rate Denial of Service (LDoS) attack – A Survey", IJETAE, Volume 4, Issue 6, June 2014

[5] A. Kuzmanovic and E. W. Knightly, "Low-rate TCP-targeted denial of service attacks and counter strategies," *IEEE/ACM Trans. Netw.*, vol. 14, no. 4, pp. 683–696, 2006.

[6] Paul Meeneghan and Declan Delaney, "An Introduction to NS, Nam and OTcl Scripting", National University of Ireland, Maynooth, 2004

[7] Serhat ÖZEKES, "EVALUATION OF ACTIVE QUEUE MANAGEMENT ALGORITHMS", _stanbul Ticaret Üniversitesi Fen Bilimleri Dergisi Yıl:4 Sayı:7 Bahar 2005/1 s.123-140

[8] V. Sharma, J.Virtamo and P. lassila, "PERFORMANCE ANALYSIS OF RANDOM EARLT DETECTION ALGORITHM", Networking Laboratory, Helsinki University of Technology

[9] Lija Mohan, Bijesh M. G. and Jyothish K. John, "Survey of Low rate Denial of Service (LDoS) attack on RED And its Counter Strategies", IEEE International Conference on Computational Intelligence and Computing Research 2012

[10] Floyd, S., "TCP and Explicit Congestion Notification. ACM Computer Communication Review, V. 24 N. 5, October 1994, pp. 10-23.

[11] Sílvia Farraposo, Laurent Gallon and Philippe Owezarski, "Network Security and DoS Attacks", 2005.