Software based Low Rate DoS Attack Detection Mechanism

Rejo Mathew
Department of Information Technology
NMIMS University,
Mumbai, India

Vijay Katkar
Department of Computer Engineering
NMIMS University,
Mumbai, India

ABSTRACT
Existing DoS attack detection tools are unable to detect Low rate DoS (LDoS) attacks. Many researchers have proposed mechanisms to detect LdoS attack. But they require modifications to the existing infrastructure or protocols which is not practical. There should be a lightweight mechanism which could be integrated with existing Intrusion Detection Systems. This paper proposes a lightweight software-based approach for LdoS detection which could be integrated with existing Intrusion detection system and does not require any change in existing infrastructure and protocol.Experimental results are provided to support the effectiveness and efficiency of proposed mechanism.

Keywords
DoS Attacks, Shrew Attacks, LDoS Attacks, RTO Exploitation Attack, Lighweight software.

1. INTRODUCTION
Most DoS detection tools anticipate large number of incoming packets to identify them as DoS attack.So the attackers have started exploiting various factors and vulnerabilities that vary from iterative servers [11] to fixed minimum RTO property of TCP. In this new kind of attack called shrew attack or low rate TCP attack, attacker sends packets at a low rate such that the standard tools cannot detect them.TCP is widely used, so a solution requiring changes to TCP and widespread modifications of users' software may not be practical. This motivates us to consider a software based approach that can be implemented in a resilient routing infrastructure benefitting a large community of standard TCP users. The attacker ensures periodic overflow of a router's buffer, a basic signature of attack traffic will be intermittent short bursts of high rate traffic in between periods of little or no activity. In practice, however, attack traffic can deviate from the basic signature.Moreover, in a distributed attack, the traffic from individual attack sources may not have the expected traffic characteristics, but the aggregation of such traffic does. Therefore, it is essential to develop detection algorithms using time based network traffic features that are both robust to practical traffic distortions and efficient to carry out even at a busy router. [1]

Section 2 of this paper describes the current LDoS attack detection schemes .Section 3 is the proposed mechanism. Section 4 describes the experimental setup and the results obtained. Section 5 concludes the paper.

2. RELATED WORK
The researchers have proposed detection methods which require modification to the existing systems or protocol. Amey and Ansari [8] added an extra module which monitors the flow and filters the malicious objects based on comparison with standard objects stored in memory. [9] Suggests RTO randomisation to confuse the attacker and avoid a low rate DoS attack. Some researchers have focussed on the buffer of the router and queue management algorithms to detect low rate DoS attacks.Sandeep and Terz [7] suggested increase in the buffer size so that the attacker has to send packets at higher rates to fill up the target buffer which would then cease to be a low rate DoS Attack. In [6] two parameters are considered, one is packet percentage at the cache queue of a target router, and another is the threshold percentage which is calculated on the basis of the number of packets of client as well as the number of packets of the attacker. [5] Proposes a mechanism called HAWK (halting anomaly with weighted choking) which focusses is on dropping algorithms for detection of DoS flows and achieving fairness among adaptive and non adaptive flows. Jin and Shin [2] use an IP to hop-count (IP2HC) mapping table to detect and discard spoofed IP packets. Rodriguez, Briones and Nolazco [3] have applied fuzzy logic on the Hop count mechanism to make it more accurate in terms of finding out the packet arrival time-series data; Hubei [10] discovered that most of the attacks are of self-similar in nature so if we study the patterns then large scale attacks could be mitigated to a certain extent. But John and David propose a Dynamic Detection mechanism [1] not based on the traffic pattern but gets back to the source.Here the attack port is identified and the path of attack is backtracked to detect the source of the attack. PAD [4] which relies on the near-periodic nature of the traffic is slower than MAD which operates on the sampled time-series. We require mechanism that would be integrated to the intrusion detection systems and should be easily deployable.

3. PROPOSED MECHANISM
LDoS attacker periodically sends data bursts to overflow the buffer of Router. Thus the Time-based network traffic features can be used to detect LdoS attacks effectively. Parameters used in proposed mechanism are listed in Table 4.1. All parameters listed in table except α, β, γ, $Th_{Discard}$, $TS_{duration}$ are time-based parameters.Based on the formula to determine the retransmission timeout for TCP connections [12] we derive the formulas to calculate Average Traffic, Average Packet inter-arrival time which are listed below:

Table 1. List of parameters for LdoS Detection Mechanism

Symbol	Meaning
$TS_{duration}$	Duration of Time Slot (in seconds)
$TS_{Traffic}$	Traffic in curent Time Slot
$TS_{Atraffic}$	Average traffic per Time Slot
TS_{IA}	Packet Inter-arrival time in current Time Slot
TS_{AIA}	Average Packet Inter-arrival time
TS_{To}	Number of Time-outs in current Time Slot
TS_{Tn}	Number of dicarded packets in current Time Slot
TS_{Con}	Number of connections to the server in current Time Slot
$Th_{Discard}$	Threshold value for number of packets discarded in a particular Time Slot

$TS_{ATraffic} = \alpha * TS_{ATraffic} + (1- \alpha)\ TS_{Traffic}$

$TS_{AIA} = \alpha * TS_{AIA} + (1- \alpha)\ TS_{IA}$

Two threads are used to process network traffic data (Say, Thread 1 and Thread 2). Thread 1 is used to collect network traffic data from network interface card and calculate following parameters:

i. $TS_{Traffic}$
ii. TS_{To}
iii. TS_{Tn}

Thread 2 sleeps for $TS_{duration}$ and after that it calculates following parameters using data generated by Thread 1:

i. TS_{IA}
ii. TS_{AIA}
iii. $TS_{Atraffic}$

Using the above calculated parameters, Thread 2 uses the following algorithm to detect an LdoS attack.

3.1 Algorithm

***Input*:** Network traffic Data

***Output*:** Detection of LdoS attack

Step 1: Compare $TS_{Traffic}$ and $TS_{ATraffic}$;

If ($TS_{Traffic}$ > ((1+β) * $TS_{ATraffic}$)) Then
Goto step 2
Else
Goto step 4

Step 2: Compare TS_{To} with T_{Con} and TS_{Tn} with $Th_{Discard}$;
If ((TS_{To} >= 2*TS_{Con}) and (TS_{Tn} > $Th_{Discard}$)) Then
Goto step 3
Else
Goto step 4

Step 3: Compare T_{IA} with T_{AIA};

If ((T_{IA} <= γ* T_{IA})) Then
Conclude low rate DoS attack and not congestion
Else
Goto step 4

Step 4: Suspend thread processing till the end of current time slot interval

After calculating these parameters the average traffic per timeslot is compared with traffic in the current time slot. When the traffic in current time slot $TS_{Traffic}$ exceeds (1+ β) times the average traffic per time slot ($TS_{ATraffic}$) then we proceed further.Next we check the number of timeouts in that timeslot and the number of discarded packets.When the timeouts are twice the total number of connections and the number of discarded packets exceed the set threshold level we proceed further.Next we compare the average inter-arrival time along with the inter-arrival time in current timeslot.When we confirm that the interarrival time has reduced we conclude that it is not congestion but we are under an LdoS attack.If any of the steps fail then the thread processing is suspended for that time slot.

4. EXPERIMENTAL SETUP & RESULTS

Experimental setup is as shown in figure 2. Configuration of Server machine is; Intel Core2 Duo 1.6 GHz, 1GB RAM, OS: MS Windows Server2003 SP2, Softwares installed on machine are: JDK 1.6, Netbeans 6.9. Configuration of Client machine and Attack Machine is; Intel Core2 Duo 1.6 GHz, 1GB RAM, OS: MS Windows XP SP3, Softwares installed on machines are: JDK 1.6, Netbeans 6.9.

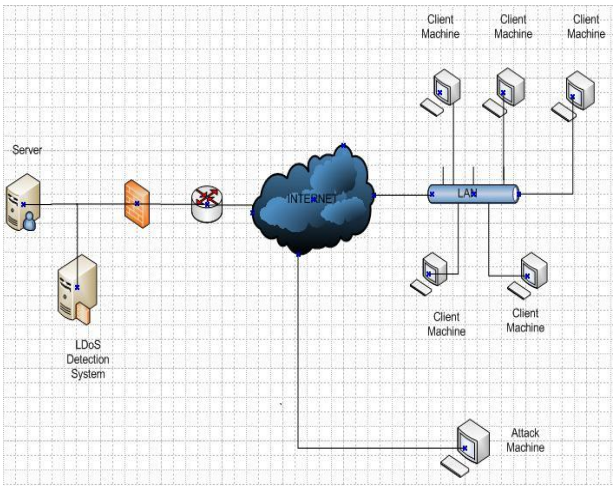

Fig 1: Architecture of our LDoS Detection Mechanism

Table 2. Comparison of LDoS Detection Mechanisms

Methods Vs Features	**Dynamic Detection**	**PAD and MAD Models**	**HAWK**	**At EDGE Router**	**RTO Randomisation**	**Self-Similarity**	**Software Based**
Effective	For Non-Distributive Type of LDoS Attack	Depends on Signature database	Depends on Threshold value set	Depends on Signature database	Not Effective	Depends on Signature database	Can detect Distributed attacks, Real Time Detection
Modification to the existing infrastructure	No Extra Memory is required at router	Monitoring mechanism needed	Monitoring mechanism Needed	Extra Memory Needed	Congestion control mechanism needs to be modified	Extra Memory Needed	Can integrate it to the existing Network
Overhead	Processing and memory overhead	Processing and memory overhead	Processing overhead	Processing overhead	Changes needed to Congestion Control Method	Processing overhead	Less overhead
Accuracy	Fails against Distributed LDOS Attack	Depends on accuracy of fuzzy controller designer	Depends on Threshold value set	Depends on known patterns matched	Fails for Distributive LDoS Attack	Depends on known patterns matched	Highly Accurate

Values of α, β, γ and $Th_{Discard}$ used for experiment are 0.75, 0.30, 0.70, 1000 respectively. Experimental test is divided into two phases.

In first phase clients sets up the connection with server. Server has a pool of files and it transfers these files to clients on demand. Network traffic Data and server performance data is collected during this phase and graphical representation of this data is shown in Fig 2. RTO is plotted versus the number of discarded packets, in an attackfree environment, from the figure we can see that the number of timeouts and the number of discarded packets are quite less (almost zero). The average traffic does not increase and the Interarrival time is evenly spread across that particular timeslot.It means that the file requested is sent and the overall bandwidth utilisation does not increase rapidly.

In second phase; attack machines are used to launch low rate DoS attack against server. The process followed by attack machines is described below:

Step 1: Send 100 spoofed packets to server
Step 2: Wait for 5 seconds
Step 3: Goto back to step 1

Network traffic data and server performance data collected during this phase is analyzed graphically to support efficiency and effectiveness and proposed mechanism.

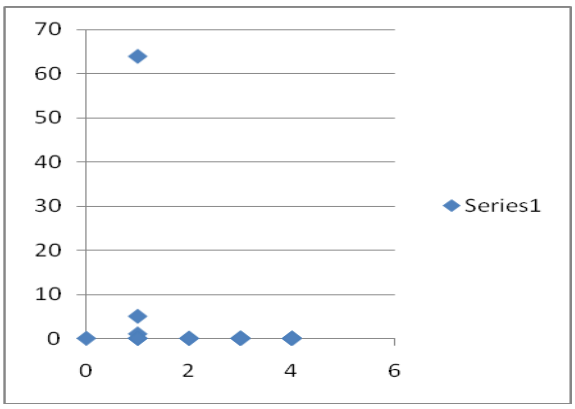

Fig 2: RTO versus Discarded packets

From Fig 3: we can observe that as soon as the attack starts the traffic per time slot increases and the average traffic exceeds the current traffic in timeslot.Then watchout for the timeouts. Fig 4: clearly shows that the number of timeouts it is twice the number of connections to the server (connections made were 3 here timeouts are 6).Also the number of packets discarded has exceeded the Threshold limit (Here above 600).Now we have to observe the Interarrival time of the incoming packets.

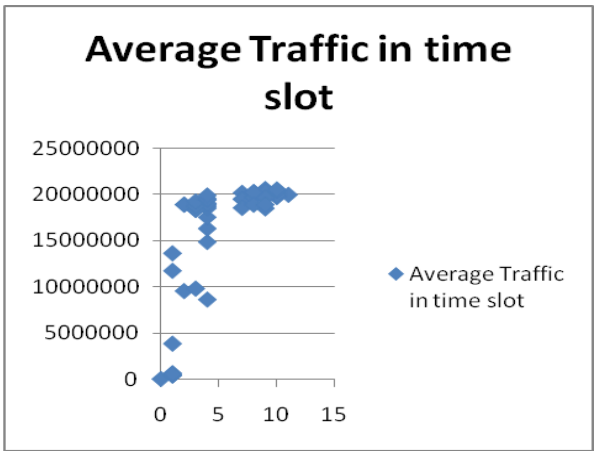

Fig 3: Average traffic in time slot versus RTO

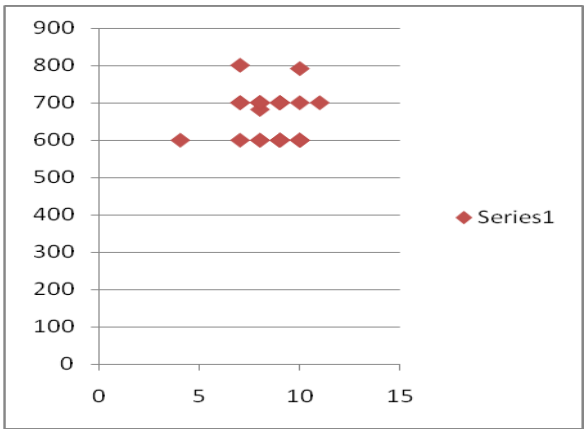

Fig 4: RTO versus Discarded packets

From Fig 5: we can observe that the average interarrival time has reduced drastically and is much less than the packet interarrival time in that time slot.hence we can conclude that it is a low rate DoS attack.

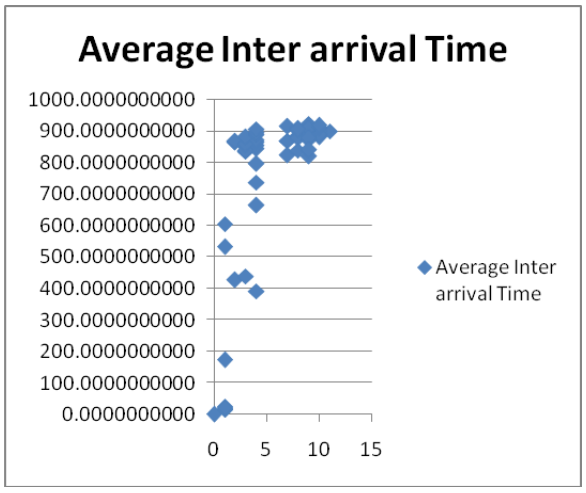

Fig 5: Average InterArrival Time versus RTO

5. CONCLUSION

This paper proposes a lightweight and efficient software-based detection scheme against low rate TCP DoS attacks. This mechanism has been able to address and overcome some of the challenges and issues faced by other detection mechanisms which are mentioned below:

- *Can be integrated to intrusion detection system*: No Modification to the existing routers.
- *Can be deployed without any extra cost*: the delay and cost factor poses serious problems to implement other mechanisms in a global perspective.
- *Robust to changes in the transport-layer headers:* Most of the detection approaches perform analysis based on packet contents rather than packet inter-arrival times.
- *Based on time based network traffic:* If any attacker can imitate even one packet of the genuine TCP flow then the whole mechanism fails.
- *No Flow Separation*: It acts upon aggregate traffic without flow separation, enabling analysis of encrypted traffic even in a passive monitoring framework.
- *Based on real time traffic*: most detection mechanisms assume the attack parameters a priori which is not preferable in today's real world scenario.
- *Works on Application layer:* Most researchers have not explored the LDoS attacks at the application layer.

6. REFERENCES

[1] Haibin Sun, John C.S. Lui, David K.Y. Yau, ,"Defending Against Low-rate TCP Attacks: Dynamic Detection and Protection",Proceedings of the 12th IEEE International Conference on Network Protocols (2004)

[2] C. Jin, H. Wang and K. Shin: Hop-Count Filtering, "An Effective Defense against Spoofed DoS Traffic", *ACM CCS* (2003)

[3] J.C.C. Rodriguez, A.P. Briones and J.A. Nolazco, "Dynamic DDoS Mitigation based on TTL field using fuzzy logic",*CONIELECOMP '07*, Mexico (2007)

[4] Gautam Thatte, Urbashi Mitra and John Heidemann, "Detection of Low-Rate Attacks in Computer Networks", University of Southern California IEEE (2005)

[5] Zenghui Liu , Liguo Gua, "Attack simulation and signature extraction of low-rate DoS.", 3rd International Symposium on Intelligent Information Technology and Security Informatics IEEE 2010 Computer Society (2010)

[6] Y.K. Kwok, R .Tripathi, Y. Chen and H. K. HAWK, "Halting anomalies with weighted choking to rescue well-behaved TCP sessions from shrew DDoS attacks", Proc. of the 3rd Int'l Conf. on Networking and MobileComputing (ICCNMC 2005). New York:Springer-Verlag, pp: 423-432 (2005)

[7] Sandeep Sarat and Andreas Terz, "On the Effect of Router Buffer Sizes on Low-Rate Denial of Service Attacks", IEEE Computer Society (2005)

[8] Amey Shevtekar, Karunakar Anantharam and Nirwan Ansari, "Low Rate TCP Denial-of- Service Attack

Detection at Edge Routers", IEEE COMMUNICATIONS LETTERS,VOL. 9, NO. 4 (2005)

[9] G. Yang, M. Gerla, and M. Y. Sanadidi, "Defense against low rate tcp-targeted denialof-service attacks", ISCC '04 Proceedings of the Ninth International Symposium on Computers and Communications 2004 Volume2 (ISCC"04), pages 345–350, Washington, DC, USA. IEEE Computer Society (2004)

[10] Wuhan, Hubei, "Detection of Low-rate DDoS Attack Based on Self-Similarity", China in 2010 Second International Workshop on Education Technology and Computer Science (March 06-March 07)

[11] Gabriel Macia-Fernandez, Jesus E. Diaz-Verdejo and Pedro García- Teodoro : Evaluation of a low-rate DoS attack against iterative servers : Department of Signal Theory, University of Granada, c/Daniel Saucedo Aranda, s/n, 18071 Granada, Spain (2006)

[12] RFC 793 Transmission Control Protocol.