Proceedings of the 5th WSEAS International Conference on Applied Computer Science, Hangzhou, China, April 16-18, 2006 (pp670-675)

A New Mechanism for Improving Robustness of TCP against Pulsing Denial-of-Service Attacks

HIROSHI TSUNODA, KENJIROU ARAI, NEI KATO, and YOSHIAKI NEMOTO
Graduate School of Information Sciences
Tohoku University
Aramaki Aza Aoba 6-6-05, Aoba-ku, Sendai, Miyagi, 980–8579
JAPAN
http://www.nemoto.ecei.tohoku.ac.jp/%7Etsuno/

Abstract: In this paper, we propose a new mechanism to combat pulsing Denial-of-Service (DoS) attacks. Pulsing DoS attacks can seriously degrade the throughput of legitimate TCP flows in a stealthy manner. The attacker send periodic short bursts of traffic (i.e. pulses) to cause packet losses of TCP flows. For improving robustness of TCP against the attacks, we propose to use adaptive bandwidth estimation mechanism in TCP congestion control process. The performance of the proposed method is evaluated through simulations, and is compared with the other TCP variants. From the simulation results, we verified that the proposed method can effectively mitigate the effect of pulsing DoS attacks.

Key-Words: Pulsing DoS Attack, Robustness, Transmission Control Protocol (TCP), Congestion Control, Available Bandwidth Estimation, Adaptive Estimation Mechanisms

1 Introduction

Flooding Denial-of-Service (DoS) attacks and Distributed DoS (DDoS) attacks are serious threats in the current Internet. A DoS attacker send a huge number of packets to a victim using many *zombies*. As a result of attacks, resources around the victim such as network bandwidth and computational power are consumed and legitimate users cannot access to services provided by the victim. Today, various countermeasures to conventional flooding type of DoS attacks have been proposed. Especially, statistical analysis in traffic rate is useful to detect the massive packet flooding traffic [1].

In recent years, however, a new class of DDoS attack, which is hard to detect by conventional approaches, have been reported. In the literature, this kind of DDoS attack is called low-rate TCP-targeted DoS attack, shrew attack [2] [3], or pulsing DoS attack [4]. The attacker of pulsing DoS attacks send short bursts of traffic periodically, instead of continuous packet flood like conventional DDoS attacks. Such bursts fill up the buffer of intermediate routers and cause packet losses of legitimate TCP flows. Since a TCP sender interprets packet losses as occurrence of congestion, the throughput of legitimate flows is reduced as a result of congestion control. Note that the objective of pulsing DoS attacks is not to prevent provision of service, but to degrade quality of service. Therefore, pulsing DoS is also known as Degradation-of-Service attacks [5]. The difficulty in detecting pulsing DoS attacks is that the attacker has low average rate compared with traditional DoS attacks. Moreover, the attacker can control the level of damage caused by the attacks, by adjusting burstiness and interval between each burst. Hence, it can be difficult for the victim itself to find the attacks. Thus, if the attacker targets an e-commerce web site and TCP connections between the site and its customers, the site may miss out on the potential profit and new customers.

To detect the attacks and mitigate the damage, various countermeasures have been proposed. Major approaches at the present moment try to detect and filter the attacks at an intermediate router. However, all of router-based approaches have deployment issues. Thus, an another approach, end-point TCP improvement, needs for more research. Existing methods of TCP improvement focus on only parameter setting, such as initial congestion window and minimum value of retransmission timeout. In this paper,

we propose a novel approach using available bandwidth estimation mechanisms for improving robustness of TCP to pulsing DoS attacks. The proposed method uses two bandwidth estimation mechanisms, and switches them adaptively depending on the situation. The performance of several TCP variants and the proposed method under pulsing DoS attacks, are evaluated through simulations.

The rest of the paper is organized as follows. Section 2 explains the characteristics of pulsing DoS attacks, then describes about the overview of existing countermeasures. In section 3, we present the proposed method using adaptive bandwidth estimation mechanisms, in addition to the brief explanation of existing TCP modification using bandwidth estimation mechanisms. Section 4 gives performance evaluation results. Concluding remarks are in section 5.

2 Related Works

2.1 Overview of Pulsing DoS Attacks

The attack traffic of pulsing DoS is composed by periodic sequence of short bursts as shown in Fig. 1. The detailed mechanism of pulsing DoS attacks and related parameters are discussed in [2]. The effect of pulsing DoS attacks is to give an illusion of a continuous congested networks to legitimate TCP flow senders. Due to periodic packet losses caused by the pulses, the legitimate TCP flows decrease their sending rate according to congestion control algorithm. Since the congestion control algorithm of TCP is a *additive-increase, multiplicative-decrease (AIMD)* algorithm, it needs a long time until the sending rate returns to the original level.

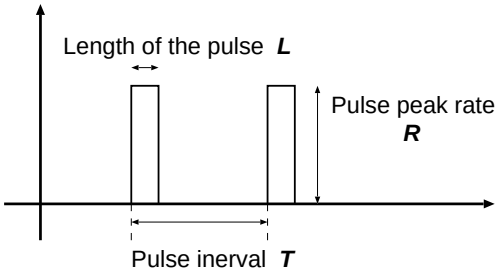

Fig. 1: Square-wave Pulse

Pulsing DoS attacks can be more effective by exploiting the deficiencies in the Retransmission Time-Out (RTO) mechanism of TCP [2]. In most of TCP implementation, in order to maximize throughput, a minimum RTO (minRTO) is set to 1 sec as recommended in [6]. Therefore, many TCP flows tend to time out at the same timing after multiple packet losses. If the attacker synchronizes pulse sending interval to minRTO, the attacker can easily force the retransmitted packets of legitimate flows to be lost. In this case, the throughput of the flows decrease significantly.

Moreover, X. Luo *et al.* points out the new class of Pulsing DoS attacks, AIMD-based attacks [4]. A basic pulsing DoS attack is termed as a timeout-based attack by the authors. The AIMD-based attacker aims to make target TCP flows frequently enter the fast recovery state instead of timeout. Although the pulse sending frequency of AIMD-based attacks is larger than that of timeout-based attacks, the length of the pulse is quite short compared with the case of timeout-based attacks. Hence, average rate of attacks keeps low and detection of the attacks is a hard task.

2.2 Existing Countermeasures

There are two approaches as countermeasures to pulsing DoS attacks: router-based approach and end-point TCP improvement.

2.2.1 Router-based Approach

Active Queue Management (AQM) scheme is used for achieve high link utilization with a low queuing delay. In [3], the authors investigate the mitigation effect of AQM against pulsing DoS attack, and evaluate two representatives: Random Early Detection with Preferential Dropping (RED-PD) [7] and CHOKe [8] algorithms. As a result of evaluation using NS-2 [9], the authors concludes that it is difficult to detect a pulse which has short duration and to defeat pulsing DoS attacks, even if these AQM algorithms are implemented in a router.

S. Sarat *et al.* investigate the relation between buffer size and the rate of attack pulse required to cause congestion [10]. The evaluation results shows that moderate increase in buffer size coupled with the use of RED-PD is sufficient to minimize the impact of the attacks. However, increase of buffer size can generate more queuing delay for normal flows.

In [11], the authors proposes HAWK algorithm. It is a kind of flow table driven packet dropping technique. HAWK algorithm can identify high burst rate flow with the small flow table. However, if every packet which consists an attack pulse has different source address by IP spoofing, it results that each

packet is classified in separate flows and it is difficult to find high burst rate flows by HAWK algorithm. [12] proposes another flow-based approach for detecting attack flows at edge routers that perform the transport layer function. However, the proposed scheme is also ineffective for an attack pulse which is composed by spoofed packets.

Signal analysis techniques are useful for detecting attack pulses. In [13], the authors propose to monitor network traffic in short time scale (e.g. 1ms), and to analyse the amplitude spectrum distribution in the frequency domain. The proposed method uses Discrete Fourier Transform to convert time-domain series of traffic into its frequency domain representation. X. Luo *et al.* suggest to detect pulsing DoS attacks by observing anomalies in fluctuation of incoming traffic using a discrete wavelet transform technique [4]. [14] proposes to extract the periodic signatures of the attacks by using auto-correlation. The proposed method detects pulsing DoS attacks by matching with the extracted signature, then use the deficit round robin algorithm to provide bandwidth allocation and protection between flows.

Although many router-based approaches are proposed and show good performance against pulsing DoS attacks, all of the approaches need to solve deployment issues. Therefore, we belive that end-point TCP improvement solutions should be focused.

2.2.2 End-point TCP Improvement

Comparing with router-based approaches, there are only a few approaches for improving end-point TCP. In [3], two modification in TCP parameters, initial window size and minRTO, are investigated.

When packet losses occur for a flow which has small window sizes, the flow cannot retransmit lost packets by the fast retransmit algorithm [15] and it results in retransmission timeout. In other words, such flow is more vulnerable to pulsing DoS attacks. To avoid such unwanted retransmission timeout, RFC 3390 [16] recommends an increase in the permitted upper bound for TCP's initial window from one or two segment(s) to between two and four segments. Although this modification may increase robustness of TCP against pulsing DoS attacks, there are no noticeable improvement as reported in [3].

Concept of the minRTO randomization is proposed in [2], then [17] shows the attack mitigation effect through simulation. This approach makes the attackers are difficult to synchronize their pulse sending interval to minRTO. However, minRTO randomization may cause the reduction of TCP connection performance under non-attack conditions [12].

Comparing with router-based approach, deployment issues in end-point TCP improvements are not so many. In this study, we focus on TCP improvements for mitigating the effect of pulsing DoS attacks.

3 TCP with Adaptive Bandwidth Estimation Mechanism

Under pulsing DoS attacks, a TCP sender suffers from false congestion signals caused by malicious pulses. Needless to say, this issue is very similar to the issue in the TCP over wireless networks. Nevertheless, to the best of our knowledge, none of research evaluate the performance of TCP modifications for wireless networks, under pulsing DoS attacks. Hence, in the reminder of this paper, we study the performance of TCP Westwood [18] [19] based approaches. This section gives the brief explanation of TCP Westwood and our proposed approach, SACK-based TCP with adaptive bandwidth estimation.

3.1 TCP Westwood Bandwidth Estimation and Rate Estimation

TCP Westwood is a sender-side modification of the TCP congestion window algorithm. The general idea of TCP Westwood is to estimate available bandwidth from received ACKs, and to calculate the appropriate slow start threshold (ssthresh) after a congestion based on estimated available bandwidth. Since available bandwidth indicates the current network capacity, available bandwidth will keep large values if packet losses are caused by only wireless link errors. The congestion window (cwin) is set based on ssthresh according to a basic TCP algorithm, then a sender avoids unnecessary reduction of sending rate after fast recovery phase cause by packet losses due to wireless link errors. Moreover, the sender is allowed to quickly increase its cwin after a timeout event.

There are two TCP Westwood versions depending that have different bandwidth estimation algorithms, the bandwidth estimation (TCPW-BE) [18] and the rate estimation (TCPW-RE) [19]. In TCPW-BE, the sender measures a sample of bandwidth b_k based on the received two successive ACKs and their interarrival time. After timeout event or receiving three

duplicate ACKs, ssthresh is calculated as follows:

$$ssthresh[segments] = \frac{RTT_{min} \cdot \hat{b_k}[bps]/8}{seg\_size[bytes]} \quad (1)$$

where RTT_{min} and $\hat{b_k}$ denote minimum RTT value and smoothed value of b_k, respectively.

In TCPW-RE, b_k is calculated from all ACKs arrived in measurement period T. This measurement corresponds to the rate actually achieved recently by the connection over time T [19].

TCPW-BE and TCP-RE have the different features each other. TCPW-BE provides a reasonably good estimate of the instantly available bandwidth at a bottleneck link and it is robust to random error. However, the estimation algorithm tends to be overestimate the connection fair share and it results in problem on fairness and friendliness with conventional TCP. Although estimation algorithm of TCPW-RE is good for friendliness, it cannot estimate accurate bandwidth unless there are sufficient number of packets on the connection.

3.2 SACK-based TCP with Adaptive Bandwidth Estimation

The proposed method combines Selective ACK (SACK) modification [20] and TCP Westwood. Introducing SACK modification enables a TCP sender to retransmit multiple packets at once, and it is good for performance improvement. Since friendliness with conventional TCP is an important factor in TCP improvement, the proposed method is based on combination of TCPW-RE and SACK.

Fig. 2 shows the state transition diagram of the proposed adaptive bandwidth estimation scheme. As shown in the figure, the proposed method basically uses TCPW-RE with SACK, and TCPW-BE with SACK is used only right after the connection start and timeout events. After the fast recovery phase, TCPW-RE with SACK is used until a timeout event occurs. This switching of estimation algorithms contributes to avoid the estimation accuracy reduction problem of TCPW-RE when insufficient packets are on the connection. As a result, the proposed method can estimate more accurate available bandwidth throughout the connection.

Moreover, the proposed method recalculate $RTTmin$ after a timeout event, in order to reflect up-to-date network condition in $RTTmin$. Since $RTTmin$ is used to calculate ssthresh, this recalculation contributes to set valid ssthresh.

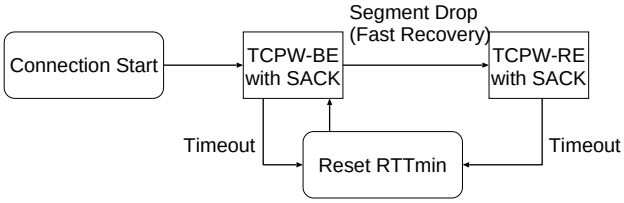

Fig. 2: State Transition Diagram of the Proposed Method

4 Performance Evaluation

The performance evaluation relies on computer simulation, using NS-2 [9]. Figure 3 shows the classic dumb-bell topology we used in our simulations. N pairs of TCP senders and receivers are in the network. The bandwidth of the link between routers is 10Mbps, and this link becomes a bottleneck. All remained links have 100Mbps bandwidth. RTT of each connection is set to 30msec. The start time of each TCP sender is a random variable uniformly distributed from 0 to 5 seconds, and each sender sends data packets for 100 seconds. Also, the attacker begins to send pulses at 15 seconds. The length of a pulse L (see Fig. 1) is 10msec and pulse peak rate R is equal to bandwidth of the bottleneck link.

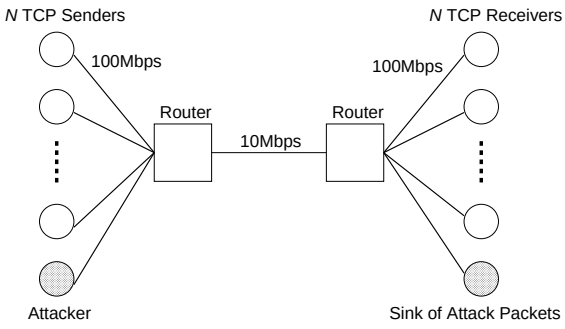

Fig. 3: Simulation Topology

We evaluate the robustness of several types of TCP flows: TCP New Reno, TCP New Reno with RTO randomization [17], TCP Sack, TCPW-BE, TCPW-RE, and the proposed method. The robustness of each method is evaluated using *Normalized Throughput* (ρ) as the comparison metric. ρ is defined as follows:

$$\rho = \frac{S_{PDoS}}{S_{Normal}} \quad (2)$$

where S_{PDoS} denotes aggregate throughput achieved by N TCP connections under the pulsing DoS attack. S_{Normal} denotes aggregate throughput without

the pulsing DoS attack. In addition, to evaluate the magnitude of the attack, average attack rates R_{avg} is computed by

$$R_{avg} = \frac{L \cdot R}{L + T}. \quad (3)$$

Figure 4 shows normalized throughput as the number of TCP connections changes. The pulse interval T is 100msec. In this case, R_{avg} is about 0.91Mbps and less than only 10% of the bottleneck bandwidth, but the normalized throughput of every protocol is clearly decreased. However, our proposed method achieves higher throughput than the others, especially in the case that the number of TCP connections is small. We also find that RTO randomization, which is an existing TCP improvement approach, does not work well in this situation. RTO randomization targets to avoid a drop of a retransmitted packet by an attack pulse. Since pulse length is small but a pulse is sent frequently, a timeout event also occurs frequently and RTO randomization does not work effectively.

The effect of the attacks decreases as T increases, as shown in Fig. 5, 6, 7. For the protocols other than our proposed method, the effect is still remained if the number of connections is one or two. On the other hand, the proposed method keeps high throughput in such cases.

From these results, we conclude that our proposed method has basic robustness to pulsing DoS attacks and effectively mitigate the effect of the attacks.

5 Conclusions

In this paper, we examined using a bandwidth estimation mechanisms for improving the robustness of TCP to pulsing DoS attacks. Then, we proposed SACK-based TCP with adaptive bandwidth estimation mechanism. Two estimation algorithms, a bandwidth estimation and a rate estimation, are used in the proposed method. The proposed method switches those algorithms depending on the situation. The bandwidth estimation algorithm is used only right after a connection start and timeout events, and the rate estimation algorithm used in a steady state of a connection. This approach contributes to get accurate estimation of available bandwidth throughout a connection. From the simulation results, we show that bandwidth estimation mechanisms help to improve the robustness of TCP against pulsing DoS attacks. Simulation results also demonstrated the proposed SACK-based protocol using adaptive bandwidth estimation achieve the highest throughput among the evaluated protocols under pulsing DoS attacks. As a consequence, we conclude that the proposed method has robustness to pulsing DoS attacks and effectively mitigate the effect of the attacks.

References:

[1] C. Manikopoulos and S. Papavassiliou: "Network Intrusion and Fault Detection: A Statistical Anomaly Approach", IEEE Communications Magazine, pp. 76–82 (2002).

[2] A. Kuzmanovic and E. Knightly: "Low-Rate TCP-Targeted Denial of Service Attacks(The Shrew vs. the Mice and Elephant)", Proc. of ACM SIGCOMM'03 (2003).

[3] A. Kuzmanovic and E. Knightly: "Low-Rate TCP-Targeted Denial of Service Attacks and Counter Strategies", to be appeared in IEEE/ACM Transactions on Networking, **14**, 5 (2006).

[4] X. Luo and R. Chang: "On a New Class of Pulsing Denial-of-Service Attacks and the Defense", Proc. of NDSS2005 (2005).

[5] R. Chang: "Defending against Flooding-Based Distributed Denial-of-Service Attacks: A Tutorial", IEEE Communication Magazine, **40**, 10, pp. 42–51 (2002).

[6] M. Allman and V. Paxson: "On Estimating End-to-End network path properties", Proceedings of ACM SIGCOMM 1999 (1999).

[7] R. Mahajan, S. Floyd and D. Wetherall: "Controlling High-Bandwidths Flows at the Congested Router", Proc. of IEEE ICNP'01 (2001).

[8] R. Pain, B. Prabhakar and K. Psounis: "CHOKe: A Stateless Active Queue Management Scheme for Approximating Fair Bandwidth Allocation", Proc. of IEEE INFOCOM'00 (2000).

[9] S. McCanne and S. Floyd: "ns Network Simulator". `http://www.isi.edu/nsnam/ns/`.

[10] S. Sarat and A. Terzis: "On the Effect of Router Buffer Sizes on Low-Rate Denial of Service Attacks", Proc of ICCCN2005 (2005).

[11] Y. Kwok, R. Tripathi, Y. Chen and K. Hwang: "HAWK: Halting Anomalies with Weighted

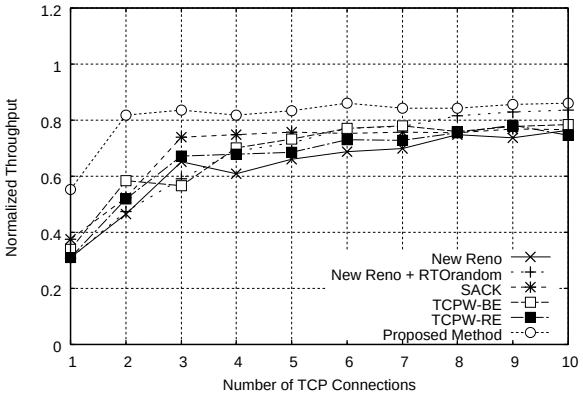

Fig. 4: Effect of the Attack (R=10Mbps, L=10msec, T=100msec, $R_{avg} \approx 0.91Mbps$)

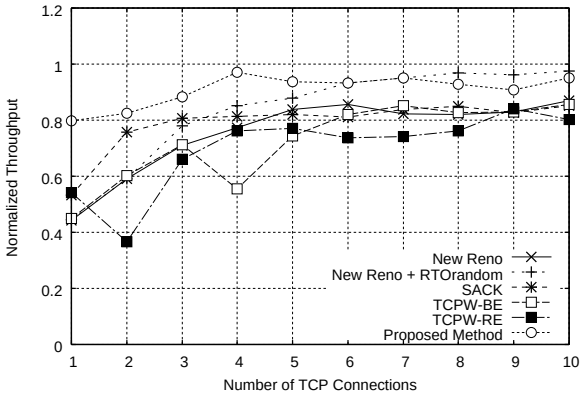

Fig. 5: Effect of the Attack(R=10Mbps, L=10msec, T=200msec, $R_{avg} \approx 0.48Mbps$)

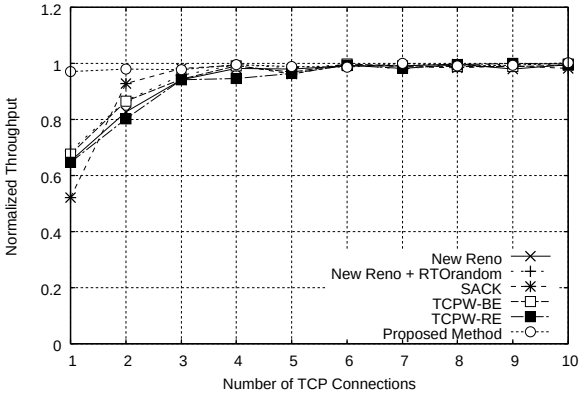

Fig. 6: Effect of the Attack (R=10Mbps, L=10msec, T=500msec, $R_{avg} \approx 0.20Mbps$)

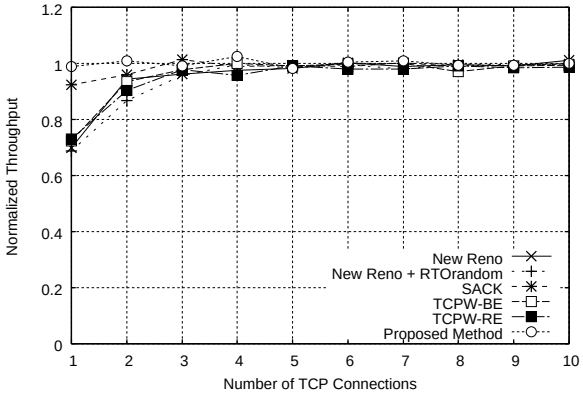

Fig. 7: Effect of the Attack (R=10Mbps, L=10msec, T=1000msec, $R_{avg} \approx 0.10Mbps$)

Choking to Rescue Well-Behaved TCP Sessions from Shrew DDoS Attacks", Proc. of ICCNMC2005 (2005).

[12] A. Shevtekar, K. Anantharam and N. Ansari: "Low Rate TCP Denial-of-Service Attack Detection ad Edge Routers", IEEE Communications Letters, **9**, 4, pp. 363–365 (2005).

[13] Y. Chen and a. Y. K. K. Hwang: "Filtering of Shrew DDoS Attacks in Frequency Domain", Proc of the First IEEE LCN Workshop on Network Security (WoNS) (2005).

[14] D. K. Y. H. Sun, J. C.S. Lui: "Defending Against Low-rate TCP Attacks: Dynamic Detection and Protection", Proc. of ICNP'04 (2004).

[15] M. Allman, V. Paxson and W. Stevens: "TCP Congestion Control", RFC2581 (1999).

[16] M. Allman, S. Floyd and C. Partridge: "Increasing TCP's Initial Window", RFC3390 (2002).

[17] G. Yang, M. Gerla and M. Sanadid: "Defence against Low-rate TCP-targeted Denial-of-Service Attacks", Proc of ISCC2004 (2004).

[18] C. Casetti, M.Gerla, S.Mascolo, M.Y.Sanadidi and R.Wang: "TCP Westwood: Bandwidth Estimation for Enhanced Transport over Wireless Links", ACM Mobicom 2001, pp. 287–297 (2001).

[19] R. Wang, M. Valla, M. Sanadidi, B. Ng and M. Gerla: "Efficiency/Friendliness Tradeoffs in TCP Westwood", 7th IEEE Symposium on Comupter and Communications (2002).

[20] M. Mathis, J. Mahdavi, S. Floyd and A. Romanow: "TCP Selective Acknowledgement Options", RFC2018 (1996).