Is It Congestion or a DDoS Attack?[1]

Ferhat ELMAS, (*Student Member*, IEEE)[2]

2006101102, Computer Engineering, Bogazici University, Turkey

ferhat.elmas@boun.edu.tr

***Abstract* – In this paper, quiet attack which is a sort of DDoS attack is proposed. Attack utilizes of TCP traffic and botnets and network feedback control are the main parts of the model of the attack. Short lived TCP flows are abused by the attack and the attack is hard to detect because currently used defense schemes such as adaptative queue management and aggregate congestion control couldn't detect.**

***Index Terms* – DDoS, router, TCP**

INTRODUCTION

Internet is the essential part of our lives in banking or shopping and DoS is a major threat to the Internet. Since internet has an interdisciplinary structure and more people use the services due to their ease-of-use, attackers are also professionals, they know back-end of the structure so they invent new methods and threats. For example, low rate DoS attack is a new threat and it is reported on the Internet2 experimental network and it is really hard to detect. War between attackers and developers has been going on since the birth of the Internet and the only way to stop is developers to stay ahead of the attackers. Therefore, in this paper, we present quiet attack which will get you to the front of the attackers. Quiet attack is similar to low rate DoS and uses TCP which is a network adaptive protocol and is important since it has large slice in current usage. Attack hides itself by behaving as a normal traffic and uses available botnets to send traffic**.**

QUIET ATTACK BASICS

The attack uses short lived TCP flows to affect long lived ones. TCP is an adaptive network protocol so strictly obeys what the congestion algorithms such as slow start and congestion avoidance say. Therefore, TCP monitors the network and takes the necessary steps, as a result of this long lived TCP flows usually start and end in congestion avoidance phase while short ones in slow start phase. Exploit works as long lived flows are forced to enter slow start phase.

In the model, firstly a set of short lived TCP flows are started where the rate of this set must be proportional to the link under attack because we want to limit the bandwidth. After every short time T(random in 0-1s to prevent determinism), a new set is injected into the network. Meanwhile, we don't apply the exact rate but we make the available bandwidth negligible since TCP flow rate is congestion dependent. Link capacity can be estimated by capprobe and available bandwidth can be monitored by abget or pathchirp. After these calculations, we hold available bandwidth negligible and force it not to increase. If we can succeed this, congestion will be in router queue and when queue is filled, packets are started to be dropped by droptail queue mechanism which means no priority for any packets. Therefore, long lived TCP flows time out, enter slow start phase and congestion window is reduced results lower sending rate and throughput. If congestion continues sufficiently long, random drops start in the queue and much lower throughput results. In short, long lived TCP flows are suffered from available bandwidth which is negligible to the capacity and is limited by sending enough short lived TCP flows.

Exploit is fed from end-to-end window flow control trade-off. When # of the connections increased, delay and congestion increase so what is the right congestion window siz? TCP uses slow start mechanism when size increased which will favor the

short-lived flows. In the attack, short lived flows are persistent so long lived flows wait retransmmission timeout and this results sharp decrease in throughput.

In old times, short lived flows were in huge numbers but occupied very low bandwidth but today with botnets short lived flows can limit the bandwidth so quiet attack is a major threat for Internet structure.

QUIET ATTACK EXECUTION

Network reconnaissance phase is to determine the router to be attacked. We assume the attacker has an access to the botnet. Botmaster sends the tracert command, trace the path from the bot to the target website, to active bots and all bots run the command and return the answer to the botmaster. Then, botmaster chooses the router that is an edge router whose IP address is observed in all tracerts. Important assumption here is that all bots are nearly in the same time zone and there are few ISPs in this time zone and there are few points to enter the backbone. Other methods include espionage during a cyber war, stealing confidential information of ISP or network mapping such as cartoreso or traceroute. Then, web servers are found which will be used to generate short lived TCP flows. Bots request files from web servers via HTTP so they can make use of CAPTCHA or pages don't include CAPTCHA. Bots are limited in the context of traffic to make the attack hard to detect. Moreover, CDN web servers shouldn't be chosen since their traffic doesn't pass over them. However, most of the servers can't afford the CDN.

For example, the target link has 10Gbps connection so to fill the link we need 20000 bots and each has 500 Kbps connection. To make the attack undetectable by IDS, we use 20000 website and in each second each bot requests different page of different website. No pattern so no detection. Meanwhile, botmaster needs the network feedback or congestion traffic at the target link because it must calculate available bandwidth and parameters to add more traffic. For example, ISP can make load sharing and botmaster can learn this using a feedback tool such as abget and add more traffic.

Basically, access a botnet, choose target router and webserver to generate short lived TCP flows and feedback mechanism to monitor the target link. Then, instruct the bots to start the traffic in every T seconds, monitor the link and add more bots.

SIMULATION

Quiet attack is more dangerous than other attacks such as shrew and RoQ attacks. However, web traffic is't affected so much from quiet traffic since much of the traffic is composed short lived flows and the attack favors them. Traffic is also bursty. However, more short lived flow by the attack can degrade the throughput but this is left for future work. No mitigation is available since attack uses TCP which is congestion adaptive and attack monitors the link and behaves dynamically. Shrew and RoQ attacks are detectable since they send high rate packets but quiet attack generates traffic which is similar to legitimate traffic so it is unique.

CONCLUSION and FUTURE WORK

An important DdoS attack model is proposed which can be easily confused by real congestion but the attack requires a botnet. Today, there is no mitigation but CAPTCHA seems a good solution. We will be building a defense system for this attack.

ACKNOWLEDGMENT

This paper is prepared for CmpE475 course. A summary of understanding of original article.

REFERENCES

[1] Amey Shevtekar and Nirwan Ansari, Fellow, IEEE

IEEE Communications Letters – 7 July 2009

1089-7798 vol 13, issue 7, pp. 546-548

[2] Just I wish