# DoSoffensivemeasuresandgeneraldefensivemeasures.

*by* LinForrest


| | | | |
|---|---|---|---|
| FILE | DOSOFFENSIVEMEASURESANDGENERALDEFENSIVEMEASURES..TXT (11.6K) | | |
| TIME SUBMITTED | 22-MAY-2018 09:50PM (UTC+0800) | WORD COUNT | 1886 |
| SUBMISSION ID | 967164592 | CHARACTER COUNT | 9819 |

DoS offensive measures and general defensive measures.

## 1.Introduction

With the increasing application of network, accompanying network security issues have become a major obstacle to the development of internet. Denial of service (DoS) is an attack on network devices exploiting defects of network protocol and vulnerabilities of system which renders network or system unavailable, namely, a denial of service. A distributed denial of service (DDoS) is an attack on a host orchestrated by multiple computers scattered in a network, which is more harmful. Currently, DDoS has become a worldwide attack method exploiting vulnerabilities of system. Many business sites and government departments have been targeted by DoS and thus suffered huge economic losses. Therefore, it makes good sense to effectively prevent and stop DoS on network users, especially network managers. TCP-based DoS accounts for 90%-94% of all DoS. According to statistics in 2014, the number of attacks identified as large-scale DDoS has reached an average of 28 per hour (Preimesberger, 2014). This paper conducted a research on SYN Flood and proposed a feasible defense method.

## 2.SYN Flood

SYN Flood is the most common and the most effective DoS form. This paper begins with TCP three-way handshake for connection establishment to understand the principle of SYN Flood. According to Tanenbaum & Wetherall (1996), the first-step handshake process: a host sends a TCP message including a SYN (Synchronize) to a server, informing it of port and initial sequence number for TCP connection. The second-step handshake process: In response, the server replies with a SYN+ACK (Acknowledge). The third-step handshake process: after receiving the SYN+ACK, the host sends an ACK to the server. At this point, connection is established.

If the host suddenly crashes or goes offline after sending a SYN message to the server, the server cannot receive an ACK acknowledgment message after sending a SYN+ACK message. In this circumstance, the server will usually resend a message and abandon unfinished connection after waiting a period of time, which refers to half-open connection (Lemon, 2002).

Lemon (2002) presents that an attacker sends a TCP connection request using a forged IP address. If the forged IP address is occupied, after a SYN+ACK message sent by the server is received, a RST message will be sent to inform the server that there is no need to wait for an invalid connection. If the forged IP address is not occupied, the server will continuously retry within SYN Timeout value before abandoning the connection (a vulnerability of TCP protocol). According to figure 1, in an attack, an attacker sends multiple SYN requests to the server but does not respond to ACK acknowledgment message, which results in many half-open connections. The server is busy with processing falsified TCP connection requests and has no spare time for requests of legitimate users. In severe cases, stack overflow and server crash will be caused.

At present, there are a couple of simple and effective countermeasures to SYN Flood. The first is to reduce SYN Timeout value. Max. quantity of half-open connections which can be maintained by a server

Max. quantity of half-open connections = SYN attack frequencySYN Timeout

By reducing SYN Timeout value, a host will release resources occupied by half-open connections as soon as possible, so that server load will be reduced exponentially. The disadvantage of this method is that when SYN Timeout value is reduced, visit of legitimate users may be affected (Eddy, 2007).

The second is to adopt SYN Cookie technique. A server will generate an encrypted sequence number instead of memorizing a sequence number. In the third-step handshake process, the sequence number of ACK message returned by the client is the encrypted initial sequence number plus 1. At this point, the server will run the same encryption algorithm to regenerate correct sequence number. Connection will be established if the regenerated sequence number is matching to the sequence number of ACK message (Zuquete, 2002). The disadvantage of this method is that the server has to perform hash operations for many times in order to validate the effectiveness of ACK message returned by the client. Hash algorithm has high complexity and will occupy a considerable amount of RAM and CUP resources when attacked. A hacker can wage a targeted ACK Flood attack exploiting this defect (Kiesel, 2002).

One of server-based defense methods is introduced as below, which has been designed to address the shortcomings of the aforesaid two commonly used methods. By receiving two same source SYN messages and screening connection requests, DoS will be blocked.

3.Receiving two same source SYN messages

According to JIAN Xiao-chun, WU Zhen-qiang, HUO Cheng-yi, ZHANG Jie (2008, March), two same source SYN messages refer to two SYN messages have the same source address and source port, which request the same service from the same server. The two SYN messages are respectively marked as SYN1 and SYN2. The principle behind this method is that if the client doesn't receive ACK acknowledgement within a certain period of time after the client sends SYN, the client will retransmit a new SYN connection after timeout. As a result, the server will receive two SYN messages with the same source address and port number within a certain period of time. When the server receives the first SYN message, it will not respond with ACK, waiting for the second SYN message, and judge whether the request is legitimate based on time difference.

Therefore, it's crucial to set up reasonable time period. Firstly, retransmission time difference is defined

X = the time when SYN packet is received for the second time - the time when SYN packet is received for the first time

Reasonable time period X = {x|<x<}, in which Tmin refers to min. value of retransmission time difference, which is defined as the time of timeout retransmission of the server (generally several seconds). Tmax refers to max. value of retransmission time, which is defined as double of the time of timeout retransmission of the server(JIAN et al., 2008).

Therefore, a reasonable connection request needs to satisfy the below conditions:

1.

2.

3.

Only SYN message which satisfies the above-mentioned three conditions can enter the third-step handshake process, while other SYN requests will be abandoned with only source address and reception time saved to retransmission time . In DDoS, different from DoS in which superfluous SYN connection requests are sent using an address, SYN messages are sent using different IP and ports. The method of receiving two same source SYN messages can effectively judge whether there is a DDoS malicious attack (JIAN et al., 2008).

As shown below, this paper introduces in details how to receive and record two same source SYN messages. Firstly, a HASH table is created to record SYN message sent from the client for the first time. Secondly, a HASH function is determined with 16-bit port number of 32-bit source IP address as a key. In consideration that an attacker usually uses an random function to forge its IP source address, therefore use HASH(s)=s mod65535. In which, “s” refers to 48-bit source IP address and source port number. The definition domain is (00000000,0000~FFFFFFFF,FFFF). Each HASH table node stores contents such as source IP, port and time (JIAN et al., 2008).

The below formula can be concluded usingη(usage ratio of total memory) of the HASH table:

Based on the optimum η given by experts from the server side, update time T of HASH table can be achieved. With a cycle of T seconds, traverse the HASH table to release space of timeout nodes so as to prevent the overflow of HASH table (JIAN et al., 2008).

JIAN et al.,(2008) explain that if the server is not overloaded, it means an attack has not been received. Go through TCP three-way handshake process as normal. If the server is overloaded, use this method to check if each SYN message received has been stored in the HASH table. The below three circumstances exist:

·If there is no record of the SYN in the HASH table, establish a new record and abandon the SYN message

·If there is a record of the SYN in the HASH table, but time difference falls out of a reasonable scope, abandon the SYN request and update the time recorded in the HASH table.

·If there is a record of the SYN in the HASH table, and time difference falls within a reasonable scope, a legitimate connection request is determined. In such case, go through TCP three-way handshake process and release node space in the HASH table.

4.Method Summary

If a SYN Flood attack has not been received, a server functions normally. If such an attack is received, this method works in various circumstances:

An attacker continuously sends SYN packets using a forged IP address and then forges an IP address again. Time difference of sending continuous SYN is obviously less than reasonable time period X. The method works because a server abandons a SYN packet after receiving it each time and updates the time in a HASH table. In other words, no half-open state of connection exists.

An attacker discontinuously sends SYN messages and directly changes IP addresses. A server abandons SYN packets and a HASH table is updated after timeout. The method works because if an forged IP is reused for a long time, time difference is obviously larger than normal interval.

This method effectively blocks a SYN Flood attack and addresses the shortcomings of the aforesaid methods. Its advantages lie in simple strategy and calculation as well as low consumption of system resources.

## 5.Conclusion

This paper first introduces DoS and analyzes in detail general attack process and form of SYN Flood. Besides, two methods, which are widely used nowadays to prevent SYN Flood, are introduced. This paper then explains in detail the method of receiving two same source SYN messages. In this method, normal time interval is first set up, a HASH table is then adopted for recording reception time of SYN messages and whether SYN messages should be abandoned is determined after comparing reception time with normal time interval, which effectively prevents and blocks DDoS. In the end, the paper compares this method with other two commonly used methods in terms of advantages and shortcomings.

A simple and effective method to prevent and block DDoS is to increase safety awareness of computer users. Vulnerabilities of operating system should be timely

patched and network security devices and software should be updated so that malicious software attacks can be blocked and DDoS scale can be reduced.

Word count: 1764 words

Reference


Chang, R. K. (2002). Defending against flooding-based distributed denial-of-service attacks: a tutorial. IEEE communications magazine, 40(10), 42-51.

Eddy, W. M. (2007). TCP SYN flooding attacks and common mitigations.

JIAN Xiao-chun, WU Zhen-qiang, HUO Cheng-yi, ZHANG Jie (2008, March). Method of defending SYN Flooding attack by receiving two same source SYN packets:Computer Engineering and Design (Vol. 29 No. 6)

Kiesel, S. (2002). On the use of cryptographic cookies for transport layer connection establishment. Proc. EUNICE Summer School, 177-184.

Lemon, J. (2002, February). Resisting SYN Flood DoS Attacks with a SYN Cache. In BSDCon (Vol. 2002, pp. 89-97).

Preimesberger, C. (2014). DDoS attack volume escalates as new methods emerge. Eweek.

Tanenbaum, A. S., & Wetherall, D. (1996). Computer networks (pp. I-XVII). Prentice hall.

Zuquete, A. (2002). Improving the functionality of SYN cookies. In Advanced Communications and Multimedia Security (pp. 57-77). Springer, Boston, MA.

# DoSoffensivemeasuresandgeneraldefensivemeasures.

ORIGINALITY REPORT

| %11 | %7 | %7 | %6 |
|---|---|---|---|
| SIMILARITY INDEX | INTERNET SOURCES | PUBLICATIONS | STUDENT PAPERS |

PRIMARY SOURCES

| | | |
|---|---|---|
| 1 | Yuli Fu. "An improved algorithm for detecting SYN flooding attacks based on network processor", 2010 International Conference on Anti-Counterfeiting Security and Identification, 07/2010<br>Publication | %1 |
| 2 | oriprobe.com<br>Internet Source | %1 |
| 3 | Submitted to Sheffield Hallam University<br>Student Paper | %1 |
| 4 | Submitted to QA Learning<br>Student Paper | %1 |
| 5 | Submitted to Nottingham Trent University<br>Student Paper | %1 |
| 6 | Submitted to Napier University<br>Student Paper | %1 |
| 7 | Martine Bellaiche, Jean-Charles Gregoire. "Avoiding DDoS with active management of backlog queues", 2011 5th International | %1 |

| | | |
|---|---|---|
| | Conference on Network and System Security, 2011<br>Publication | |
| 8 | Submitted to American Intercontinental University Online<br>Student Paper | %1 |
| 9 | Submitted to Colorado Technical University Online<br>Student Paper | %1 |
| 10 | en.wikipedia.org<br>Internet Source | %1 |
| 11 | arxiv.org<br>Internet Source | <%1 |
| 12 | ftp.kfki.hu<br>Internet Source | <%1 |
| 13 | M. Chouman, H. Safa, H. Artail. "Novel defense mechanism against SYN flooding attacks in IP networks", Canadian Conference on Electrical and Computer Engineering, 2005., 2005<br>Publication | <%1 |
| 14 | Xiao, B.. "An autonomous defense against SYN flooding attacks: Detect and throttle attacks at the victim side independently", Journal of Parallel and Distributed Computing, 200804<br>Publication | <%1 |

| | | | |
|---|---|---|---|
| EXCLUDE QUOTES | OFF | EXCLUDE MATCHES | OFF |
| EXCLUDE BIBLIOGRAPHY | OFF | | |