International Journal of Advanced Technology in Engineering and Science
Vol. No.5, Issue No. 03, March 2017
www.ijates.com
ijates
ISSN 2348 - 7550

DESIGN & IMPLEMENTATION OF HYBRID IP TRACEBACK INVOLVING PATH RECONSTRUCTION USING QoS

P.Sravanthi[1], P.Swathi[2]

[1]Asst.Professor, Dept.of CSE, Sphoorthy Engineering College

[2]Asst.Professor, Dept.of CSE, Hyderabad Institute of Technology & Management

ABSTRACT

Ease of access to internet in this present situation has increased the threats to the security of the network. Illegitimate user's causes damage to the resources, thereby hindering the service rendered to the users. Defending against Denial of Service (DoS) attacks requires not only mitigating its effect but also to identify the corresponding source of attack. This process is called IP traceback. In this paper we enumerate a hybrid IP traceback involving both packet marking and packet logging. During the process of tracing back the attack source if any router or node goes down positive feedback is not possible. We use link state routing method and Open Shortest Path First (OSPF) routing protocol to overcome such a hindrance. Also Internet Service Provider (ISP) sets constraints to monitor our network thereby identifying the attackers easily and increasing the security.

Index Terms: Internet security, Denial-of-Service (DOS), IP traceback, ISP (Internet Service Provider), Quality of Service (QoS).

I. INTRODUCTION

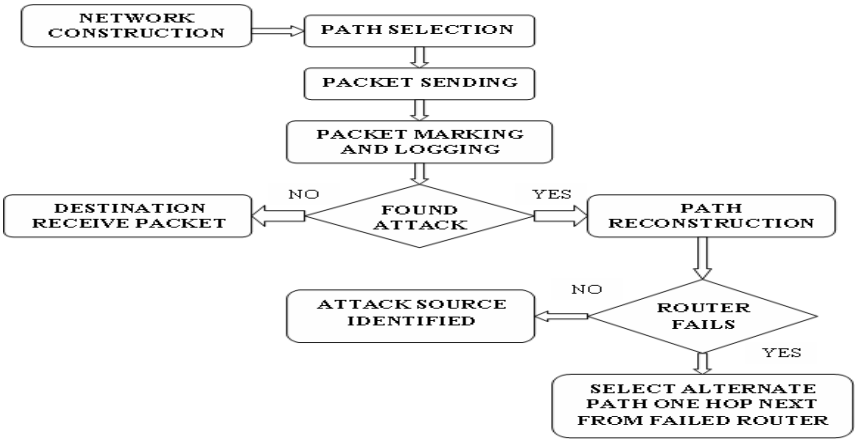
Denial-of-Service (DoS) attacks have been threatening the security of the Internet and the network [6]. A DoS attack is an attempt to make a computer resource unavailable to its intended users [5]. Although the means, motives and targets of a DoS attack may vary, it generally consists preventing an Internet site or service from functioning efficiently.

DoS attacks can be classified into flooding attacks and software exploits [6]. Flooding attacks work by flooding a victim with large amounts of packets leading to heavy traffic in the network and finally resulting in unavailability of resources. Software exploits attack a victim by sending as few as a single packet aiming to create bugs in system OS or software. Attackers send packets with arbitrary source address leading to IP spoofing. Tracing the paths of IP packets back to their origin, known as IP traceback, is an important step in defending against DoS attacks employing IP spoofing. The existing approaches for IP traceback can be grouped into several dimensions: Ingress filtering, link testing, packet marking and packet logging [7]. We mainly concentrate on Packet marking and packet logging in this paper. The main idea behind packet marking is to record network path information in packets. In mark based IP traceback, routers write their identification

information (e.g., IP addresses) into a header field of forwarded packets. The destination node then retrieves the marking information from the received packets and determines the network path.

The basic idea in packet logging is to record the path information at routers. In log-based IP traceback, packets are logged by the routers on the path toward the destination. The network path is then derived based on the logged information at the routers. In this paper we discuss about hybrid IP traceback approach based on both packet logging and packet marking, proposed by Ming-Hour Yang and Ming-Chien Yang [1]. The main design goal in this approach is to maintain the fixed memory storage in each router to alleviate the need of router to refresh the logged information periodically.

II. ARCHITECTURE OF THE PROPOSED SYSTEM



III. EXISTING SYSTEM

Each router maintains a router interface table which contains numbers of the upstream routers [1]. IP headers identification field, Flag and fragment offset field is used as a 32-bit marking field. When a border router receives a packet from the local network it forwards the packet by setting the marking field as zero. When a core router receives a packet it computes new mark value [1]. Until the mark value does not overflow it is forwarded to next router with the new mark value as computed. When the mark value overflows the packet's mark value is logged onto the router. Hash table is maintained for efficient storage and access of the logged mark values. The corresponding index in the hash table is used for further mark value computation. The packet is now transferred with the new mark value. This process continues until the packet reaches the destination.

When the victim is under attack it sends the upstream router a request for path reconstruction with the received attack packet's mark value [1]. The attack packets upstream router is found iteratively until the source is reached. Hash table containing the mark value is referred while the obtained upstream interface is negative. When the attack source is reached during this process path reconstruction is done. This process proposed by Ming-Hour Yang and Ming-Chien Yang [1] is enumerated with the algorithm and a routing example. If any router in the routing path goes down during the path reconstruction process positive feedback could not be received. In this case another alternate path to continue with the traceback scheme should be discovered.

As per the proposal of Ming-Hour Yang and Ming-Chien Yang the marking and logging with the path reconstruction at times of attack is depicted with the algorithm as below.

International Journal of Advanced Technology in Engineering and Science
Vol. No.5, Issue No. 03, March 2017
www.ijates.com
ijates
ISSN 2348 - 7550

Input: P, UI_i
begin
1. $mark_{new} = P.mark \times (D(R_i) + 1) + UI_i + 1$
2. **if** $mark_{new}$ is overflow **then**
3. $index = h = H(P.mark)$
4. $probe = 0$
5. **while not** ($HT[index]$ is empty or $HT[index]$ is equal to ($P.mark, UI_i$))
6. $probe$++
7. $index = (h + c_1 \times probe + c_2 \times probe^2) \% m$
8. **endwhile**
9. **if** $HT[index]$ is empty **then**
10. $HT[index].mark = P.mark$
11. $HT[index].UI = UI_i$
12. **endif**
13. $mark_{new} = index \times (D(R_i) + 1)$
14. **endif**
15. $P.mark = mark_{new}$
16. *forward the packet to the next router*

end

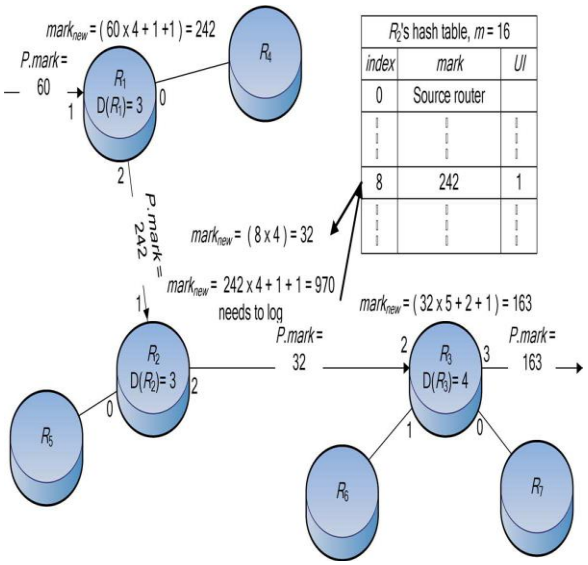


Fig 1- Packet marking and Logging

begin
1. $UI_i = mark_{req} \% (D(R_i) + 1) - 1$
2. **if** UI_i = -1 **then**
3. $index = mark_{req} / (D(R_i) + 1)$
4. **if not** $index = 0$ **then**
5. $UI_i = HT[index].UI$
6. $mark_{old} = HT[index].mark$
7. *send reconstruction request with* $mark_{old}$ *to upstream router by* UI_i
8. **else**
9. *this router is the nearest border router to the attacker*
10. **endif**
11. **else**
12. $mark_{old} = mark_{req} / (D(R_i) + 1)$
13. *send reconstruction request with* $mark_{old}$ *to upstream router by* UI_i
14. **endif**

end

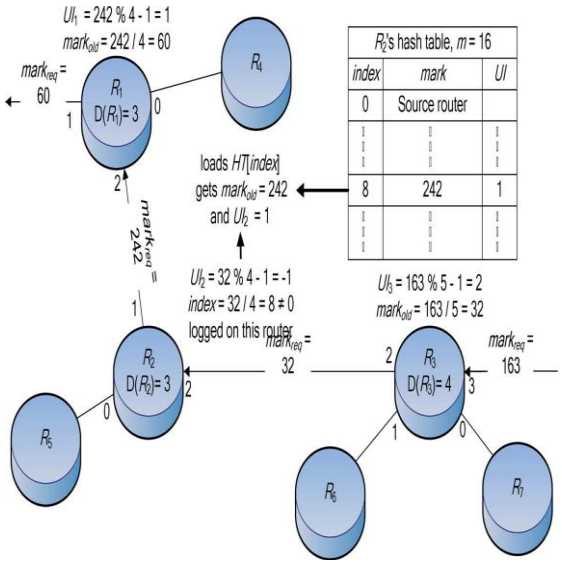


Fig 2- Path reconstruction

IV. PROPOSED ALTERNATE PATH SELECTION

Link state routing approach makes available detailed information about the network topology and connectivity found in the network to each and every node and routers through flooding [4]. By this each router has complete knowledge about its network. Once a node possesses the network-state information, it performs QoS routing, by computing paths based on multiple QoS constraints [9]. This is called multi-constrained path (MCP) selection. OSPF is an ingress routing protocol aimed at selecting an alternate MCP. As soon as the alternate path is defined, along with the network information the constructed interface table and the hash table that stores the mark value is also exchanged between the routers. With this interface values the correct mark values could be derived by the router in the newly selected alternate path.

Begin

1. ***If*** *router does not support the traceback process* ***then***
2. *Discover the router one hop next in the routing path*
3. $mark_{intermediate} = mark_{req} / (D(R) + 1)$
4. *check in the hash table of failed router*
5. ***if*** $mark_{intermediate}$ *is a valid index entry* ***then***
6. *Make its corresponding mark value from HT as* $mark_{req}$
7. $mark_{req}$ *=mark value in HT*
8. *obtain UI from the same HT row*
9. ***else***
10. $mark_{req} = mark_{intermediate}$
11. ***endif***
12. *send reconstruction request with* $mark_{req}$
13. ***endif***

end

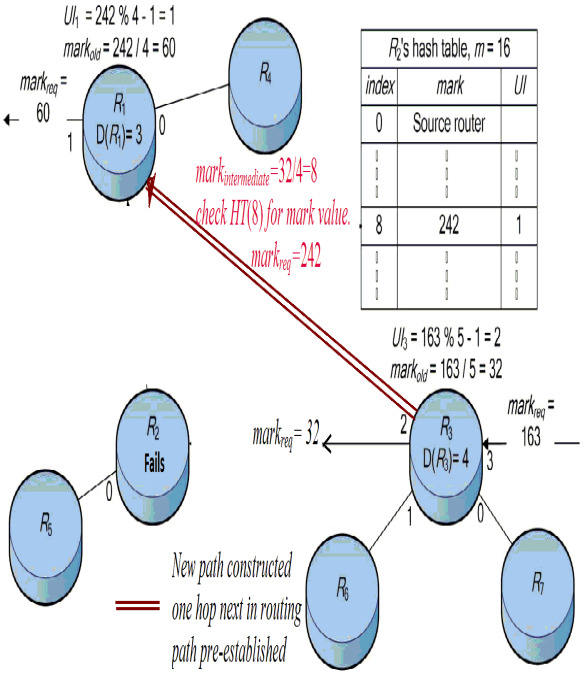


Fig 3- Alternate Path for reconstruction

During the traceback, if any router goes down its impossible to continue with the traceback. For example, if the router $R_{2\ fails}$, the path reconstruction could not continue further with the mark value 32. Through OSPF routing the hash table, interface table of each router is established to the other routers in the network. Hence R1, R3, R4, R5, R6, R7 has all the information regarding R2. Two ways are possible now. One is to find the nearest adjacent router of R2 to continue or the second way is to continue with the next router (one hop away) in the routing path pre-established.

The traceback scheme in our paper continues with the second approach. When the router R2 fails, the reconstruction scheme now continues with the router R1 which is one hop away from the failed router R2 on the network path in existence prior. When the mark value 32 reaches R1 it is first divided by the total number interfaces plus one of the failed router R2. i.e. $mark_{intermediate}$= mark req/ total number of interfaces + 1 of failed router. Hence the $mark_{intermediate}$=32/4=8. Now the hash table of R2 is checked for any valid mark entry with corresponding to index 8. If no, this $mark_{intermediate}$ is made as $mark_{req}$. Alternatively if there is a valid mark entry for the index value 8, the corresponding mark value entry is made the $mark_{req}$. The upstream interface of R1 is calculated. This newly obtained mark value is passed to router one hop away i.e. R1. Now R1 continues with the traceback process until it reaches the source. Through this approach node failure could not hinder our traceback process.

V. SECURING THE NETWORK THROUGH ISP

DDoS attacks that impact the availability of resources represents a significant opportunity for Internet Service Provider(ISP). A virtual ring is set up around our network. Now, the ISP is completely aware of all the state information of the routers in the network. With Ingress filtering the boundary router should proactively filter all traffic coming from the customer that has a source address of something other than the addresses that have been assigned to that customer. By this, attacks from outside could be minimized. Setting constraints and continuous monitoring of the network through ISP reduces security threats to a extent.

VI. CONCLUSION AND FUTURE WORK

In this paper the way to precede the traceback when the router proves negative to support, is continued by selecting an alternate path with help of OSPF routing. QoS of this routing is checked with parameters like bandwidth and delay. Also, time required to traceback to the attack source when no router fails and the when the router fails leading to alternate path selection is compared. By selecting the alternate path delay caused due to router failure is avoided to an extent. As a future extension to this paper alternate path through shortest path algorithm could be constructed and its efficiency could be analyzed with the path constructed one hop away from the failed router.

REFERENCE

[1] M.Hour Yang and M.Chein Yang, "RIHT- A Novel Hybrid IP Traceback scheme" in Proc. IEEE Trans on Information Forensics and Security, April 2012, vol. 7,no. 2, pg. 789-797

[2] S.Vincent and J.I.Raja, "A Survey of IP Traceback to overcome Denial of service attacks" in Proc. Recent Advances in Networking,VLSI and Signal Processing.

[3]T. Killalea, "Recommended Internet Service Provider Security Services and Procedures" in Network Working Group,BCP: 46, Nov 2000.

[4] H. Badis et al "Optimal Path Selection in a Link State QoS Routing Protocol".

[5] G.Santhi and A.Nachiappan, "A Survey of QoS Routing Protocols For mobile AD HOC Networks" in Proc. IJCSIT Vol.2, No.4, August 2010.

[6] C. Gong and K. Sarac, "A more practical approach for single-packet IP traceback using packet logging and marking," IEEE Trans. Parallel Distributed Syst., vol. 19, no. 10, pp. 1310–1324, Oct. 2008.

[7] S. Savage, D. Wetherall, A. Karlin, and T. Anderson, "Practical network support for IP traceback," in Proc. ACM SIGCOMM2000, Stockholm, Sweden, Aug. 2000, pp. 295–306.

[8] A. C. Snoeren, C. Partridge, L. A. Sanchez, C. E. Jones, F. Tchakountio, B. Schwartz, S. T. Kent, and W. T. Strayer, "Single-packet IP traceback," IEEE/ACM Trans. Networking, vol. 10, no. 6, pp. 721–734, Dec. 2002.

[9] P.V.Mieghem et.al., "Quality of Service routing".

AUTHOR

P. Sravanthi Reddy received B.Tech (CSE) from SRIT, Nizambad (2008) and M.Tech (CSE) from VREC, Nizamabad(2013). She is currently working as Assistant Professor in Sphoorthy Engineering College, Hyderabad, India. She has memberships in ISTE, CSI and IAENG. She has published 4 papers in international journals. Her area of interests includes Networks, Cloud Computing and Data Structures.

P.Swathy post graduate specialized in Computer Science and Engineering from SRIT, Nizamabad 2014. Assistant Professor in department of CSE from Hyderabad Institute of Technology and Management. Areas of interest Networks, Security, Operating System & Compiler Design.