HYBRID TRACE BACK SCHEME FOR IP ADDRESS

G.Soujanya[1], M. Sampath Kumar[2]

[1]M.Tech, Dept of Computer Science & Systems Engineering., Andhra Pradesh, Visakhpatnam.
[2]AssociateProfessor, Dept of Computer Science & Systems Engineering., Andhra Pradesh, Visakhpatnam.

Abstract *- **Internet is a worldwide used network in almost every field. Security is been an important issue for huge amount of data transactions and confidential information. Some networks need more, and some need less security. Now a days various attacks are developed such as viruses, denial of service and spoofing. In computer networking, spoofing is a technique in which attackers masks itself by some other user's IP address. Thus, it is difficult to find the original attackers. For this reason, a traceback scheme is proposed to trace the source of these attacks. Only one packet is used in their packet logging scheme to achieve IP tracking. To create hybrid IP traceback schemes, we combine packet marking with packet logging having less storage. In this, we propose a new scheme with fixed storage requirement for each router in packet logging without the need to refresh the logged tracking information and to achieve zero false positive and false negative rates in path reconstruction.***

Key Words: *DOS/DDOS attack, IP hybrid traceback, IP spoofing, packet logging and packet marking.*

1. INTRODUCTION

Internet is growing day by day in every field. The main issue is security, through internet the exchange of data transactions and other confidential information is done. Among many attacks, DOS attack is classified into flooding attacks and software exploits. In flooding attack large number of packets are flooded to the victim machine. We can easily find the attackers, even if they send low number of packets to victim. In software exploit attack host use other hosts vulnerabilites with few packets. Software exploit attacks include the IP spoofing attack.

The spoofed packets are traced by argumenting the packets with partial information called as packet marking and by storing the packet digest at the intermediate routers called as packet logging. Traceback scheme gives the false positive and false negative problems. Due to excess load traffic the upstream router is formed against the attack pakcets. iTrace scheme use ICMP packet having both forward and backward link of router to control the triggering packet. This iTrack scheme make use of probabilistic packet marking(PPM) and deterministic packet marking(DPM).Full IP address is divided into many segments. Segments and digests are choosen to mark on packets passing. In IP traceback scheme, the location is classified into two heads, packet marking and packet logging. In packet marking, routers write their identification in header field of forwarded IP packets. Flow of marked packets is used to form the network path towards origin. In logging scheme, the packets are logged into routers on network path towards destination

2. RELATED WORK

[1] B.Al.Duwari and M.Govindarasan

One challenging problem of address spoofing is tracing Dos attack. By argmenting the packets with partial path information and by storing packet digests at intermediate routers a traditional traceback scheme provides spoofing packets.For this reason,large number of attack packets to be collected by victim. But later a novel scheme is adopted, that is small number of attack packets to conduct small process and achieve small amount of resources. A scheme is made to preserve the marking based approach. This is called as distributed linklist traceback (DLTT).

The second scheme is based on the concept of pipelined for propagation marking information for router to reach its destination. This is called probabilistic pipelined packet marking.

Advantages

- Efficient packet marking
- Requires fixed storage
- No need to refresh often

Disadvantages

- Attackers hide themselves by spoofing their own IP address and then launch attacks.

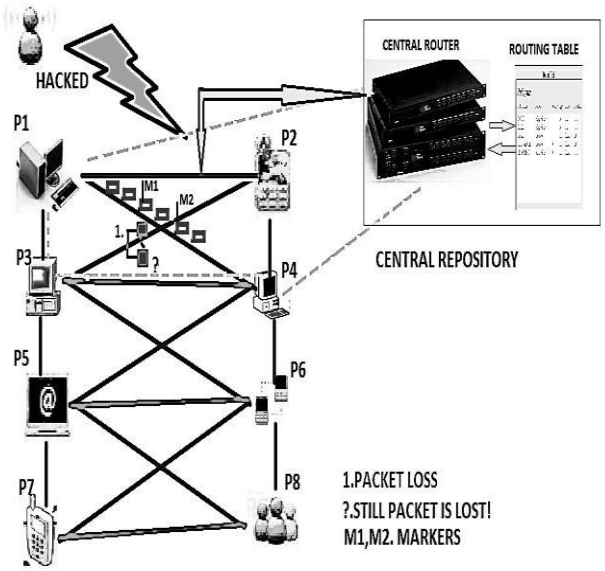
[2] For packet marking and logging Gong & Sarc proposed a new hybrid traceback scheme. To reduce storage requirement and reduce the number of routers requirement for logging. Eg: Huffman codes , Modulo/reverse modulo technique(MRT) and Modulo/reverse modulo(MORE). We take a router set as R={R1,R2,R3,……Rl).Degree of router is denoted as D(Ri),

MRT uses 32-bit marking field, MORE uses 16-bit marking field and seperates a log table into D(Ri) parts. The D(Ri)+UIi is a computed router, new marking field = marking field +D(Ri)+UIi. Even through the marking field of packet in Huffman codes, MRT and MORE can store a path of longer length than the fixed length coding. But sometimes they form a collision in the log table, it causes false positive during path reconstruction. While reconstructing a path, logged router for a packet needs to search to find the old marking field. Due to this problem in Huffman codes, MRT and MORE schemes, we propose a traceback scheme that marks router interface numbers and packet logging with hash table(RIHT).

3. SYSTEM ARCHITECTURE

This figure shows the packet loss from a central repository. During the transfer of packets from one system to other system. Attackers spoof the IP address inorder to track the status of any confidential data. The lost packets are marked to log the routers used in that table.

Like MORE and MRT, RIHT marks interface number of routers to trace the path packets. Marking field of one packet is limited, our packet marking field is needed to log the marking field into hash table and store the table index on the packet.

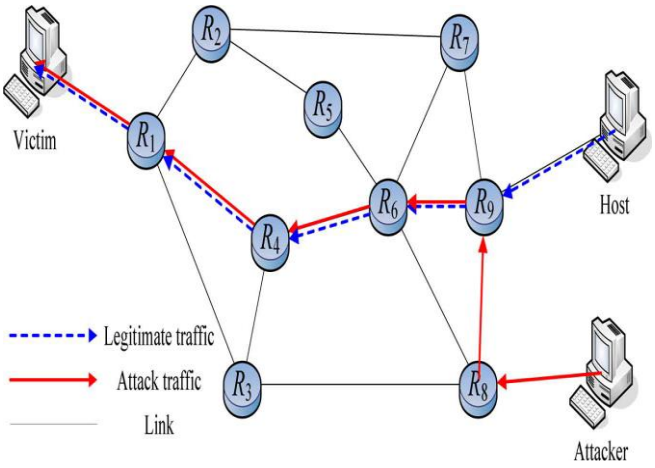


4. PROCEDURE

4.1 Network topology

A router is connected to other router or local networks. The router receives packets from its local network. The core router receives packets from other routers. Attackers use encapsulating security payload (ESP) packets to evade IDS, they generate ESP packets which can never be decrypted at victims site due to lack of shared keys. If we make ESP packets with large probability, it is enough to trace the attackers source. As mentioned above, use of fragment and identification fields does not affect packets. We overwrite two fields in our traceback scheme, IP headers identification field, flag field and fragment offset field as 32-bit marking field. The assumptions of this scheme are:

1. A router creates an interface table and then numbers the upstream interfaces from zero to D(Ri).
2. A router can identify whether the packet is coming from a router or a local network.
3. The network topology and traceback scheme may be changed but not always.



4.2 Packet Logging and Marking

Packet marking is the phase where the efficient packet marking algorithm is applied to each router along the defined path. It calculates the Pmark value and stores in hash table. If the Pmark is not overflow than the capacity of the router, then it is send to next router. Otherwise it refers the hash table and again applies the algorithm. In this when a border router receives a packet from its local network, it sets the packets marking field as zero and then forwards the packet to next router. The core router receives a packet, and it computes marknew=P.If marknew is not overflow, then the core router overwrites P.mark with marknew and then forwards the packet to next core router. Marking scheme uses a quadratic probing algorithm to search P.mark and UIi in HT. If P.mark and UIi are not found, a core router is inserted to form a pair into the table.

NOTATIONS

R_i	{R1,R2,…….Ri,…..RX}, routers in the internet
$D(R_i)$	the degree of Ri
UI_i	the upstream interface number of the router Ri in the router r(or UIi if there is no ambiguity)
P	the received packet
H()	a hash function
m	the size of a hash table (i.e. the number of slots in a hash table)
C_1,c_2	Constants
H T	An m entries hash table
HT[index]	HT[index]: the entry of the hash table HT with address index HT[index].mark: HT[index]'s mark field HT[index].UI:HT[index]'s UI field
%	the modulo operation

4.3 Path Selection

The path is said to be the way in which the selected path or file has to be have to be found and it is stored in the interface table. With the help of that interface table, the desired path between the selected source and destination can be found.

4.4 Packet Sending

One of the packet or file is to be selected for the transformation process. The packet is sent along the defined path from the source LAN to destination LAN. The destination LAN receives the packet and checks whether it has been sent along the defined path or not.

4.5 Path Reconstruction

Once the packet reached the destination after applying the algorithm, there it checks whether it has sent from the current upstream interfaces .If any of the attack is found, it requests for path reconstruction. Path reconstruction is the process of finding the new path for the same source and destination in which no attack can be made.

5. ALGORITHMS

5.1 Path marking and logging scheme

We use this algorithm technique to mark the packets and log them in the table. During the packets transfer from source to destination, to find the lost packets we must mark the path. The routers in the log table are achieved to know the address of the path marked. It takes multiple number of inputs and then compute to give the matched number of the router.

Input: P,UI_i

begin

$mark_{new}=P.mark*(D(R_i)+1)+UI_i+1$

if $mark_{new}$ is overflow ***then***

index= h=H(P.mark)

probe=0

while not (*HT*[*index*] is empty or *HT*[*index*] is equal to (*P.mark,*UI_i))

probe++

$index=(h+c_1*probe+c_2*probe^2)\%m$

endwhile

if HT[*index*] is empty ***then***

HT[*index*].*mark=P.mark*

HT[*index*].*UI=*UI_i

endif

$mark_{new}=index*(D(R_i)+1)$

endif

$P.mark=mark_{new}$

forward the packet to the next router

end

5.2 Path reconstruction

Reconstruction is the process of getting back the packet and sending them one by one by denial of service. This helps in improper packets and also helps in avoiding the loss the packets further. A reconstruction request is send to the upstream router which include packet marking field, whenever there is a victim under attack. It marks the path and then replaces by the new upstream router taken. If index is zero, then the requested router becomes the source and the path reconstruction is done.

Input: P,UI_i

begin

$UI_i = mark_{req} \% (D(R_i) + 1) - 1$

if $UI_i = -1$ ***then***

$index = mark_{req}/(D(R_i) + 1)$

if not index = 0 ***then***

$UI_i = HT[index].UI$

$mark_{old} = HT[index].mark$

send reconstruction request with $mark_{old}$ *to upstream router by* UI_i

else

this router is the nearest border router to the attacker

endif

else

$mark_{old} = mark_{req} / (D(R_i)+1)$

send reconstruction request with $mark_{old}$ *to upstream router by* UI_i

endif

end

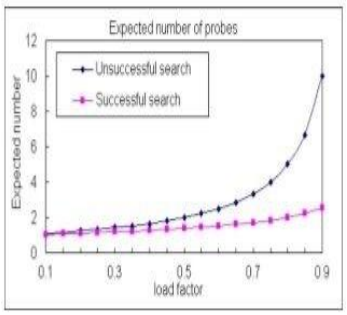
6. RESULTS AND DISCUSSIONS

In this section the storage requirements precision and performance is evaluated. Huffman code is similar to MRT and MORE, but marking field is fixed in Huffman code so we only compare with MRT and MORE. Huffman code marking performance is limited. The environment consists of a PC with Intel P4 930 3 Ghz, 2 G RAM and FreeBSD 6.2.

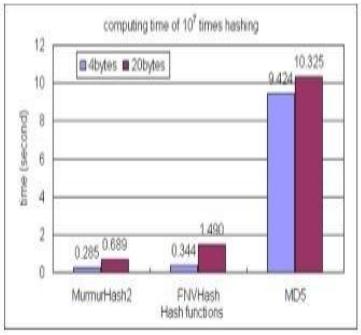
6.1 Computation Analysis

Here we compare the computing line of logging and path reconstruction in RIHT with that in MRT and MORE. It will have a hash table collision problem. A new entry is inserted, starting with the hashed to slot in some probe sequences, whenever there is collision, we have to take a hash tables load factors α.

The results of collision time may vary all the time, we have successful and unsuccessful search when logging. The expected number of probes in unsuccessful search using atmost 1/1-α. The successful search means 1/1-α ln (1/1-α) .We use various number of hash values in this time logging schemes is FNVH hash, MD5, murmur Hash2.



Expected number of probes.

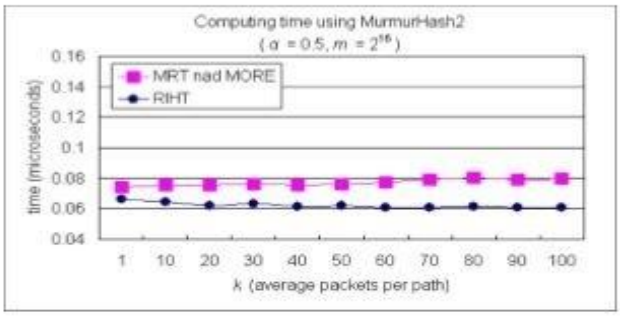


Computing time of hash functions with different input length.

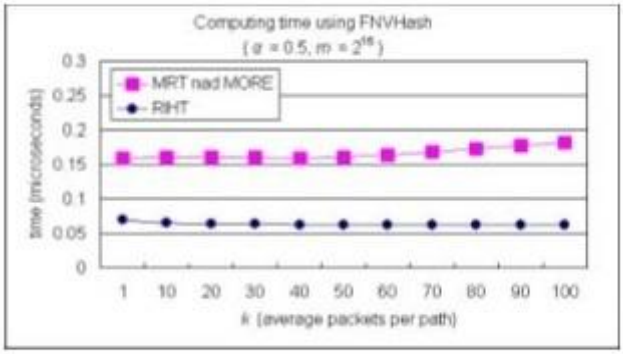
6.2 Storage requirement

Since the storage requirement of an interface table is negligible we leave it for our storage requirement. The size of a hash table can decide how many paths can be logged on one router. A hash tables load factor α=l/m where l is the number of logged paths in a hash table.

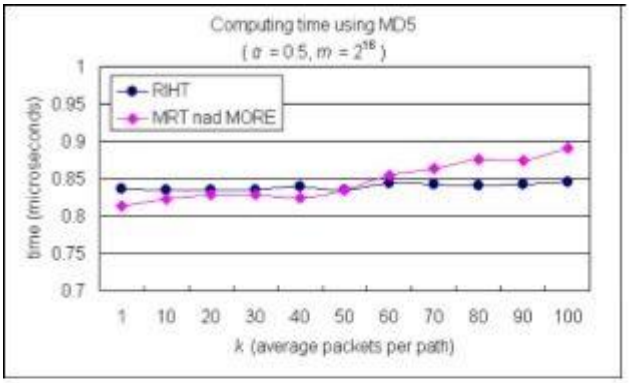
Theorem: Let N be number of paths to be logged on a router R in RIHT. Then the required size of a hash table is 2N and the storage requirement for R is 80N bits. To know the actual size of hash table on router, we use the skitter project topology distributed by CAIDA's as our sample data set of topology. We analyse 197003 complete paths in total to host and average hop count of paths is 15.46. Its average upstream is calculated as 3.89. The result shows that we need to log 159641 paths and three paths of routers as log23462, 22149,13381. Hence by this theorem, size of each hash table set 2^{16}=65536.



Computing time of logging schemes using MurmurHash2.



Computing time of logging schemes using FNVHash.



Computing time of logging schemes using MD5.

6.3 False positive and False negative rates

False positive means for any attack router there is a router mistaken and false negative means attacker fails to traceback the attack. We can use them only when the data is refreshed. There are no false positive and false negative problems for RIHT. In MRT, the router logs the marking field on to the log table. In MORE, a router uses many number of log tables which are associated with UI_i ,to log the marking fields. For both MRT and MORE, the false positive rates are greater than zero. On the other hand, in RIHT we mark index of any packet marking field under the index of a hash table, hence there will be no collision and false positive rates is zero with higher accuracy.

7. CONCLUSION

In this paper, we propose a hybrid IP traceback scheme for efficient packet logging to have some fixed storage requirement a CAIDA's skitter data set 320 kbytes is used in packet logging without the need to refresh the logged information. In the attack-path reconstruction the proposed scheme has zero false positive and zero negative rates. This scheme can also deploy a marking field as packet identity to filter the traffic which is intended to do harm and secure against DoS attacks. It is also provided with high accuracy, a low storage requirement, and fast computation, RIHT serve as an secure and efficient scheme for IP hybrid traceback. As for our future work, a new version of RIHT having 16 bit marking field to avoid the problems caused by packet fragmentation.

REFERENCES

1] B.AL-Duwari and M.Govindarasu, "Novel hybrid schemes employing packet marking andlogging for IP traceback", *IEEE Trans. Parallel Distributed Syst.*, vol. 17, no.5, pp.403-418, May 2006.

[2]A.Appleby,Murmurhash2010[Online]. Available:http://sites.google.com/site/murmurhash/

[3] A.Belenky and N.Ansari ,"IP traceback with deterministic packet marking,"*IEEE Commun.Lett.*,vol.7, no.4 , pp.162-164, Apr.2003.

[4] S.M.Bellovin, M.D.Leech, and T.Taylor,"ICMP traceback messages, "*Internet Draft: Draft-left-Itrace-04.Txt*, Feb.2003.

[5] CAIDA's Skitter Project CAIDA, 2010 [Online]. Available at caida.org.

[6] K.H.Choi and H.K.Dai,"A marking scheme using Huffman codes for IP traceback, "in Proc. 7th Int. Symp. Parallel Architectures, Algorithms Networks (SPAN '04), Hong Kong, China, May 2004,pp.421-428.

[7] D.E. Knuth, The Art of Computer Programming, 2nd ed.Redwood City, CA: Addison Wesley Longman,1998, vol. 3, pp.95-108, 2007.

[8] S.Mallinga and A.Tamilarasi, "A proposal for new marking scheme with its performance evaluation for IP traceback. "WSEAS Trans. Computer Res., vol.3, no.4, pp.259-272, Apr. 2008.

[9] The MD5 Message-Digest Algorithm. :IEFT RFC 1321, 1992.

[10] A.Yaar, A.Perrig, and D.Song, "FIT: Fast internet traceback," in Proc. IEEE INFOCOM 2005, Miami, FL, Mar.2005 , pp. 1395-1406

[11] W. John and S. Tafvelin, "Analysis of internet backbone traffic and header anomalies observed," in *Proc. IMC '07: 7th ACM SIGCOMMConf. Internet Measurement*, San Diego, CA, Oct. 2007, pp. 111–116.

[12] W. John and T. Olovsson, "Detection of malicious traffic on backbone links via packet header analysis," *Campus-Wide Inform. Syst.*, vol. 25,no. 5, pp. 342–358, 2008.

[13] D. E. Knuth, *The Art of Computer Programming*, 2nd ed. RedwoodCity, CA: Addison Wesley Longman, 1998, vol. 3, pp. 513–558.

[14] T. Korkmaz, C. Gong, K. Sarac, and S. G. Dykes, "Single packet IPtraceback in AS-level partial deployment scenario," *Int. J. Security Networks*, vol. 2, no. 1/2, pp. 95–108, 2007.

[15] S. Malliga and A. Tamilarasi, "A proposal for new marking scheme with its performance evaluation for IP traceback,"*WSEAS Trans. Computer Res.*, vol. 3, no. 4, pp. 259–272, Apr. 2008.

BIOGRAPHIES

G.Soujanya received the M.Tech degree in computer science and systems engineering from Andhra University, Visakhapatnam, in the year 2013-2015. Her research mainly focuses on network security and system security with particular interest in security issues in RFID and NFC security communication protocols. Topics include: mutual authentication protocols, secure ownership transfer protocols, tracing mobile attackers.

M. Sampath Kumar is working as Associate Professor in Dept of Computer Science & Systems Engineering in Andhra University, Visakhpatnam. His research mainly focuses on network security and system security with particular interest in security issues in RFID and NFC security communication protocols. Topics include: mutual authentication protocols, secure ownership transfer protocols, cryptography, network security, design and analysis of algorithms, and digital home.