A Probabilistic Packet Marking scheme with LT Code for IP Traceback

Shih-Hao Peng, Kai-Di Chang, Jiann-Liang Chen, I-Long Lin, and Han-Chieh Chao

***Abstract*—Cybercrime has become an important issue in the cyber-society. Distributed Denial of Service attack is the most popular attack, which uses many zombies to attack the victim, makes victim crashed and interrupt services. We propose the LT Code IP Traceback scheme to reconstruct the attack graph and find the source of attacker. LTCIP overcomes the collision problem in traditional packet marking scheme. It uses fewer packets to reconstruct the attack graph. Finally, our LTCIP is a reliable IP Traceback scheme, which can find the source of DDoS and avoid the attack**

***Index Terms*—IP Traceback, DDoS attack, Packet Marking, Network Forensics**

I. Introduction

Distributed Denial of Service (DDoS) attack is one of the popular attacks and causes damage severely. DDoS attack sends large amount of packets to the victim and let the victim cannot serve legitimate users[1]. DDoS have affected many famous companies such as Yahoo, eBay and Twitter…etc. Nowadays, finding the true source of DDoS attack is difficult. DDoS attack is easy to implement and hard to defend due to the stateless behaviour of the internet. Many business products such as intrusion detection system or firewall can detect the DDoS attack, but they could not find the attack source. In order to find the DDoS source, IP Traceback is proposed to tracing back to the source address of the attacker by overcoming IP spoofing [2].

Probabilistic Packet Marking (PPM) is an efficient marking scheme, which marks part of router's information into IP Header. It uses constant probability to decide whether the packet should be marked or not. This scheme can reconstruct the attack graph with enough packets [3], which means that PPM needs many packets to complete the reconstruction.

LT Code IP Traceback (LTCIP) scheme is based on Dynamic Probability and Luby Transform Code (LT Code) to complete the marking procedure. It uses link list to collect the marked packets and decode the received packets. Dynamic Marking Probability uses this method to receive every router's partial information with the same probability. LT Code can be used in the IP Traceback, which reduces the collision of the packets. Thus, LTCIP uses fewer packets to traceback the DDoS attacking source accurately.

Manuscript received April 27, 2012; revised May 16, 2012. This research was partly funded by the National Science Council of the R.O.C. under grants NSC 100-2219-E-197-001, NSC 100-2219-E-197-002 and NSC 100-2219-E-007-011.

Shih-Hao Peng is with the Institute of Computer Science and Information Engineering, National ILan University, ILan, Taiwan. (e-mail: solarorz@hotmail.com).

Kai-Di Chang and Jiann-Liang Chen are with Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan (e-mail: kedy@ieee.org).

I-Long Lin and Han-Chieh Chao are with the Institute of Computer Science and Information Engineering, National ILan University, ILan, Taiwan. (e-mail: paul@mail.cpu.edu.tw, hcc@niu.edu.tw).

II. Proposed LT Code IP Traceback Scheme

We propose a LTCIP by considering LT Code and Dynamic Marking Probability. There are three procedures in the LTCIP: 1) marking procedure, 2) collection procedure and 3) reconstruction procedure. The marking procedure finds the source of the attacker. The marking procedure uses LT Code encode 32-bits IP Address primitively. The collection procedure collects packets from the attacker. Finally, we use ink list to store the marked packets. The reconstruction procedure decodes the collected packets and extracts the attack paths.

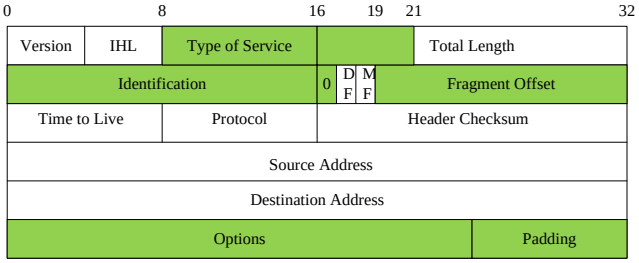

Fig. 1. Available Marking Fields

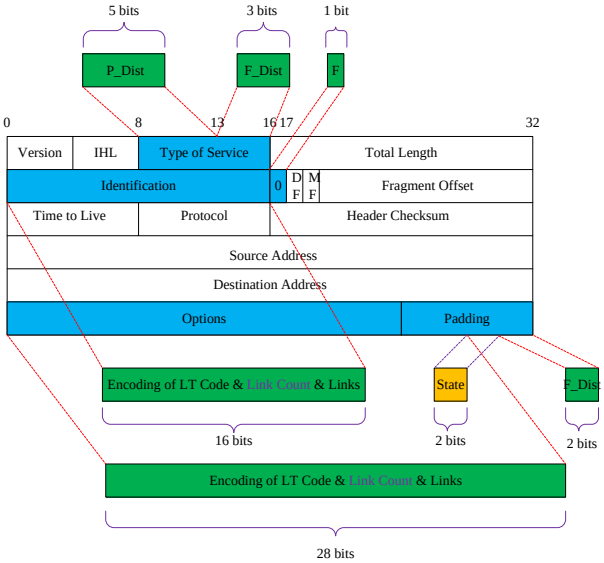

Fig. 2. Marking Fields of LTCIP

We use the 8-bits ToS Field, 16-bits ID Field, 1-bit Reserved Field and 32-bits Option and Padding Field to store

mark information. It overcomes 25 hops count and uses 5 bits to represent it[13]. Then, we divide ToS Fields into two parts. The 2 bits of Padding Field to assist to complete the 10-bits distance field in order to store the hop count and the information of beginning and terminated router. TABLE I shows the purpose of each used fields, defines the variables of marking algorithm.

TABLE I: THE REPRESENTATION OF MARKING FIELD

Marking Field	Representation
Previous Distance**(P_Dist)**	Store first hop count of marked packet
Following Distance**(F_Dist)**	Store last hop count of marked packet
Flag**(F)**	In order to let the current router know whether he marking fields of the packet has to be initialized t or not. It is used in the first LTCIP marking procedure. **(0:Initialized / 1: Not Initialized)**
Encoding of LT Code**(ELC)** & Link Count**(LC)** & Links**(L)**	Store Encoding of LT Code and links which represent the positions of encoding symbols. Link Count isn't used in the second LTCIP marking procedure.
State	In order to let the current router know whether the packet has been marked by previous routers or not. It is used in the second LTCIP marking procedure.

Dynamic Marking Procedure of LT Code

```
1. BEGIN
2. For each packet P at router R
3. Find initial TTL value of P;
4. TDistance := initial TTL value – current value of TTL;
5. LTCode ltc := NULL; Integer i :=0;
6. p := 1 / TDistance;
7. Degree distribution d is 1 or 2;
8. Let r be a random number from [0,1)
9. IF r < p THEN
10.    IF F = 0 THEN
11.      Initialize the Marking Fields;
12.      //LT Code Encoding Procedure
13.      Choose d distinct bits b[i] from 32 bits IP Address of
         current router randomly;
14.      IF d = 2 THEN
15.        ltc := b[i] ⊕ b[i+1]; LC := 1;
16.      ELSE
17.        ltc := b[i]; LC := 0;
18.      END IF
19.      ELC := ltc; L := 1 or 2 links; F :=1; P_Dist := TDistance;
20.    ELSE
21.      z := CheckEmptyField();
22.      IF z = 0 THEN
23.        Execute LT Code Encoding Procedure;
24.        ELC := ltc; L := 1 or 2 links;
25.      ELSE
26.        Execute (F = 0) step;
27.      END IF
28.    END IF
29.  ELSE
30.    IF F = 1 THEN
31.      F := 0; F_Dist := TDistance - 1;
32.    END IF
33.  END IF
34.  Forward P to the next router;
35.  END.
```

Fig. 3. Dynamic marking procedure of LT code

Dynamic Marking Procedure of Determination of LT Code

```
1. BEGIN
2. For each packet P at router R
3. Find initial TTL value of P;
4. TDistance := initial TTL value – current value of TTL;
5. LTCode ltc := NULL; Integer i :=0;
6. p := 1 / TDistance;
7. Degree distribution d is 1 or 2;
8. Let r be a random number from [0,1)
9. IF state = 00 THEN
10.    IF r < p THEN
11.      Initialize the Marking Fields;
12.      //LT Code Encoding Procedure
13.      Choose d distinct bits b[i] from 32 bits IP Address of
         current router randomly;
14.      IF d = 2 THEN
15.        ltc := b[i] ⊕ b[i+1]; LC := 1;
16.      ELSE
17.        ltc := b[i]; LC := 0;
18.      END IF
19.      ELC := ltc; L := 2 links; state := 01; P_Dist := TDistance;
20.    END IF
21.  ELSE IF state = 01 THEN
22.    Execute LT Code Encoding Procedure;
23.    ELC := ltc; L := 2 links; state := 10;
24.  ELSE IF state = 10 THEN
25.    Execute LT Code Encoding Procedure;
26.    ELC := ltc; L := 2 links; state := 11;
27.  ELSE
28.    Execute LT Code Encoding Procedure;
29.    ELC := ltc; L := 2 links; state := 00; F_Dist := TDistance;
30.  END IF
31.  Forward P to the next router;
END.
```

Fig. 4. Dynamic marking algorithm of determination of LT code

A. *Marking Procedure*

We propose two types of LTCIP. The first LTCIP uses dynamic storing method to store the encoding symbols into the marking fields of the packet, that is to say higher storage. It uses 1 bit to represent the encoding symbol and uses 5 or 10 bits to represent the links which represent the positions of encoding symbols and are decided by the random degree distribution between 1 and 2. It uses 1 bit to represent the link count. The storing order of marking information is encoding symbol, link count and the last is maybe one link position or two link positions. The best performance of the first LTCIP is that it could store at most six encoding symbols of the routers. The following will show the first LTCIP marking algorithm and steps.

The steps of Dynamic Marking Procedure of LT Code

- Step1: Finding initial TTL value and computing dynamic marking probability.
- Step2: Decide whether the packet should be marked or not through dynamic marking probability.
- Step3: Checking the flag of packet. If the flag is 0, using LT Code encoding method and saving hops count to the Previous Distance Field and storing specific information to the marking fields.
- Step4: If the flag is 1, checking the marking field whether it is full or not. If it is not full, storing specific information. If it is full, executing the step which the flag is 0.

- Step5: If the packet is not marked by the router, storing previous hops count to the Following Distance Field and forward the packet to the next router.

The marking procedure of the second LTCIP is called Dynamic Marking Algorithm of Determination of LT Code, which its marking method is a little bit different with the first LTCIP. It uses State Field to check the packet whether the marking field is full or not. Its marking method also uses dynamic marking probability to decide whether the current packet will be marked or not. It uses fixed storing method and the storing order of marking information is encoding symbol, first link and the last is second link. If the router decides to mark the packet, the encoding symbols of the following three routers will be marked into the packet too. The best performance of the second LTCIP is that it could store at most four encoding symbols of the routers and ensure saving four encoding symbols invariably. The second LTCIP marking algorithm and steps are shown as follows:

The steps of Dynamic Marking Algorithm of Determination of LT Code

- Step1: Finding initial TTL value and computing dynamic marking probability.
- Step2: Checking the packet state.
- Step3: If the packet state is 00 and the dynamic marking probability is greater than random number, using LT Code encoding method and saving hops count to the Previous Distance Field and storing specific information to the marking fields.
- Step4: If the packet state is 01 and 10, the router marks the packet determinately through the same method which is just like state of 00.
- Step5: If the packet state is 11, this step is the same as step 4 and need to store the hops count into the Following Distance Field. Finally, forwarding the packet to the next router.

B. Packet Collection Procedure

We create a Packet Collection List Table(PCLTbl) at the victim. It has two slots, the first slot will store the source IP Address of the attackers and the second slot will use the method of link list to store the marked packets by the upstream routers. When each packet forwards to the victim, the victim will check the table and insert the marking information to the appropriate place. The marking information will sort dynamically when the packet enters into the victim. In order to decrease the amount of storage at the victim, we use the behaviour of the link list to store the marking information dynamically. The Packet Collection algorithm and the steps are shown as follows:

The steps of Dynamic Marking Algorithm of Determination of LT Code

- Step1: Initializing TableEntry and NodePointer variables.
- Step2: Finding the table entry and checking the packet source which is sent by attacker.
- Step3: If the source of the checked packet is existence, comparing with each node and inserting the marking information of packet into the right place.
- Step4: If the checked packet source does not exist, creating table entry of packet source and inserting the marking information of packet into the new node.

C. Reconstruction Procedure

This finds the marking information of same distance and same source. It puts the marking information of the same features into the decoding box. The decoding box executes LT Code decoding. After decoding procedure, the result gets the IP Address from one router to the others. Finally, the procedure puts the decoded information into the stack. This method could get the full attack graphs and find the source of the attackers. The following will show the Reconstruction algorithm and steps.

The steps of Reconstruction Procedure

- Step1: Finding first table entry and first node.
- Step2: Finding the same distance through node pointer.
- Step3: Throwing the same distance node into Decoding Box and using LT Code decoding method to decode the symbols which are in the Decoding Box.
- Step4: The IP Address which is decoded through LT Code decoding method will store into the stack.
- Step5: Extracting the full attack path from stack through pop operation.

Packet Collection Procedure

```
1. BEGIN
2. For each packet P from attacker
3. Let PCLTbl to be the Packet Collecting List Table
4. TableEntry *te := NULL;
5. NodePointer *nptr := NULL, *currptr := NULL;
6. te := FindTableEntry(P.Source);
7. IF te != NULL THEN
8.   nptr := head;
9.   WHILE nptr != NULL DO
10.      IF nptr → data.P_Dist = P.P_Dist && nptr → data.F_Dist >=
         P.F_Dist || nptr →link
           = NULL THEN
11.        newNode → link := nptr → link;
12.        nptr → link := newNode;
13.        BREAK;
14.      ELSE IF nptr → data.P_Dist > P.P_Dist THEN
15.        currptr := head;
16.        WHILE currptr != nptr DO
17.          IF currptr → link = nptr THEN
18.            newNode → link := currptr → link;
19.            currptr → link := newNode;
20.            BREAK;
21.          END IF
22.            currptr := currptr → link;
23.        END WHILE
24.        BREAK;
25.      ELSE
26.        nptr := nptr → link;
27.      END IF
28.    END WHILE
29.  ELSE
30.    te := FindEmptyEntry();
31.    CreateTableEntry(te,PSource);
32.    nptr := head;
33.    nptr → link := newNode;
34.  END IF
35.  END.
```

Fig. 5. Packet collection procedure

TABLE II: SCHEME COMPARISON

Category	Scheme	Computation Overhead	Packet Required	False Positive	Packet Field Usage
Edge sampling	FMS	High	Large	High	Low
	Authenticated Packet Marking	Medium	Medium	Medium	High
Node sampling	ASPPM	Low	Medium	Medium	Medium
	DPPM	High	Medium	Low	Low
	LTCIP	Medium	Low	Low	High

Reconstruction Procedure

```
1. BEGIN
2. Let PCLTbl to be the Packet Collecting List Table
3. TableEntry *te := NULL;
4. NodePointer *nptr := NULL;
5. DeconingBox *deb := NULL;
6. Integer i;
7. te := FindFirstRow();
8. WHILE te != NULL DO
9.   nptr := head;
10.    WHILE nptr != NULL DO
11.      i := 0;
12.      p_dist := nptr → data.P_Dist;
13.      f_dist := nptr → data.F_Dist;
14.      Insert(deb,nptr → data);
15.      WHILE i != 1 DO
16.        nptr := nptr → link;
17.        cpr := Compare(p_dist,f_dist, nptr → data.P_Dist, nptr →
       data.F_Dist);
18.        IF cpr = TRUE THEN
19.          Insert(deb,nptr → data);
20.        ELSE
21.          i := 1
22.        END IF
23.      END WHILE
24.    ipaddr := Calculate(deb);
25.    Stack(ipaddr);
26.    Initialize(deb);
27.    END WHILE
28.  te := te + 1;
29.  END WHILE
30.  Extract attack path through pop operator from Stack;
     END.
```

Fig. 6. Reconstruction Procedure

III. ANALYSIS

In this section, we discuss three cases. Then, we compare each IP Traceback scheme and explain the results in Scheme Comparison section. The simulation shows the simulator and partial simulation result.

A. Case Analysis

Case1 represents the worst situation; Case2 represents the best situation; and Case3 represents the state of full marking field. The scenario is shown in Fig. 7.

Case 1:

If the packet sent by the attacker and passes through the router R1, it calculates the hops count of the packet and get the dynamic marking probability which is 1. At first, R1 checks the Flag Field, then execute LT Code encoding procedure and write the specific information to the packet. R1 forwards the packet to the next router R2. If R2 would not mark the packet through dynamic marking probability, R2 writes the previous hops count to the Following Distance Field and set the flag value to be 0. Finally, R3 to R6 would not mark the packet, victim receive only partial information of R1.

Case 2:

If the packet is forwarded from R1 to R6, each router will mark the packet definitely and the degree distribution is always 1. When the packet is forwarded to the R6, the marking field of the packet will store the encoding bits of the router from R1 to R6. When the victim receives enough packets which take the marking information from R1 to R6, the Reconstruction Procedure could use these packets to reconstruct the IP Address of the router from R1 to R6.

Case 3:

If R1, R2 to R4 mark the packet definitely and the degree distribution also is 2. The marking filed of the packet is full at the R4. When the packet is forwarded to the R5 and it decides to mark the packet, R5 checks the packet and finds out that the marking field of the packet is full. Thus, it initializes the marking fields and executes the process which flag is 0. At this moment, the marking field of the packet only has the encoding in formation of the R5.

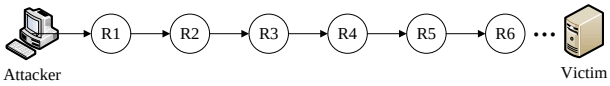

Fig. 7. Marking Router

B. Scheme Comparison

We analyze the pros and cons of the five IP Traceback schemes in TABLE II. Traditional marking schemes always divide IP Address of the router into fragments or use different marking information to write into the available fields of the packet. However, they have the collision problem, which the victim may receive many marking information of the same value. Our proposed LTCIP does not have the collision problem. Owing to the feature of LT Code, if the victim receives many packets of the same encoding symbols, every symbol could be the data of the Reconstruction procedure.

S. Savage[9] uses the method of statistics to define the function, which means the expected result of the number of marked packets. Let i be the current hops count of the packet, p be the marking probability which the value is 0.04, f be the number of fragments where f is greater than 1, d be the attack path, X be the number of packets and E(X) be the expected number of packets required to reconstruct attack paths.

$$E(X) = \frac{f * ln\,(f * d)}{p(1-p)^{d-1}} \quad (1)$$

T. Akyuz etc[12] use dynamic probability to be the marking probability and they use function to prove that each router has the same probability to send the partial information to the victim. That means the information of the downstream routers rarely overwrite the information of the upstream routers. We complete the function of dynamic marking probability which T. Akyuz etc don't define. Let the i be the current hops count of the packet, p be the marking probability which the value is 1/i, f be the number of fragments where f is greater than 1, d be the attack path, c be the distance where packet is the last to be marked by the router, X be the number of packets and E(X) be the expected number of packets required to reconstruct attack paths.

$$E(X) = \frac{f * ln(f * d)}{\frac{1}{c}\sum_{i=1}^{L=d-c}(1-(\frac{1}{c+i}))} \quad (2)$$

We define the function to reveal the performance of the LTCIP. We also use the dynamic marking probability which is redefined by us and the concept of LT Code. Let the i be the current hops count of the packet, p be the marking probability which the value is 1/i, d be the attack path, s be the length of stored hops count which the best performance value is 6, N be the encoding symbols of reconstruction, X be the number of packets and E(X) be the expected number of packets required to reconstruct attack paths.

$$E(X) = \frac{N * \log_s{}^{d}}{\frac{1}{c}\sum_{i=1}^{L=d-c}(1-\frac{1}{c+i})} \quad (3)$$

According to the previous functions (1)(2)(3), if the f is 8, d is 25, p is 0.04, s is 6 and the N is k+1 where k is 32, the E(X) of the function (1)(2) will be larger than the function (3). Thus it can be seen, the performance of the LTCIP which is proposed by us and is better than the traditional IP Traceback schemes.

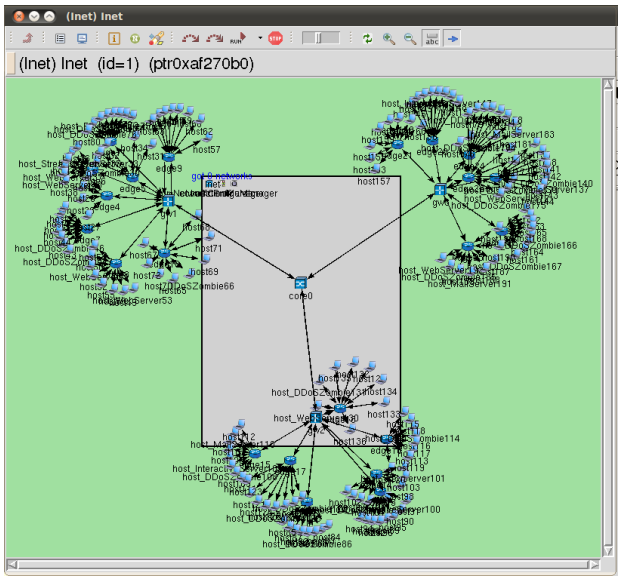

Fig. 8. Testing Topology

C. Simulation

We use OMNet++[16] as the simulator. Then we create the topology to test completed traditional DPPM marking scheme and our marking schemes. The definition of the IP Header of the Inet Framework is not allowed to write the partial information of the router to the marking fields. Thus, we redefine the needed marking fields. Fig. 8 shows the network topology and the Fig. 9 show the value of marking fields, which are marked by the marking router with the marking algorithms.

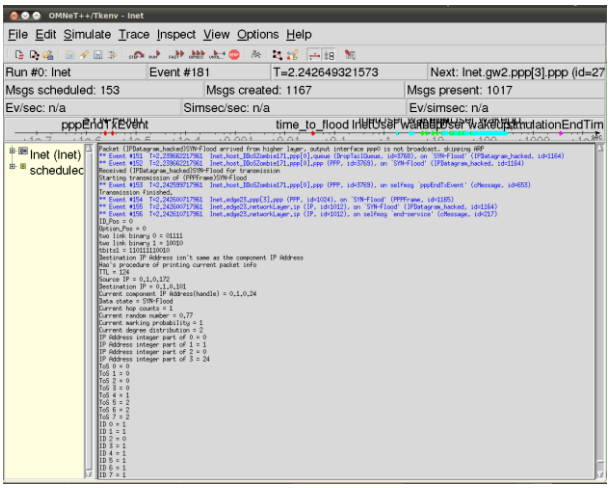

Fig. 9. The Value of Marked IP Header

IV. Conclusion and Future Works.

DDoS threats our daily usage over the internet. In order to overcome this problem, we propose an IP Traceback scheme with three procedures to reconstruct the attack graph and find the attacker. We use dynamic marking probability and LT Code to implement the marking procedure. It uses link list to store the marked packet. Then we also define a function to calculate the necessary number of packets, which is required to reconstruct attack paths. The proposed method can reconstruct the attack paths and its performance is better than the other schemes. Finally, we use OMNet++ simulator to implement the DPPM scheme and our two marking schemes. In the future, we will finish the Packet Collection procedure and the Reconstruction procedure and compare with the DPPM scheme.

Acknowledgment

This research was partly funded by the National Science Council of the R.O.C. under grants NSC 100-2219-E-197-001, NSC 100-2219-E-197-002 and NSC 100-2219-E-007-011.

References

[1] M. Li, C.-H. Chi, W. Jia, W. Zhao, W. Zhou, J. Cao, D. Long, and Q. Meng, "Decision analysis of statistically detecting distributed denial-of-service flooding attacks," *International Journal of Information Technology & Decision Making* vol. 2, no. 3, pp. 397-405, 2003.

[2] E. S. Pilli, R. C. Joshi, and R. Niyogi, "Network forensic frameworks: Survey and research challenges," *Digital Investigation*, vol. 7, issues 1-2, October 2010, pp. 14-27.

[3] S. O. Amin, M. S. Kang, and C. S. Hong, "A Lightweight IP Traceback Mechanism on IPv6," *X. Zhou et al. (Eds.):EUC Workshops 2006*, LNCS 4097, pp. 671-680, 2006.

[4] S. Karthik, V. P. Arunachalam, and T. Ravichandran, "A Comparative Study of Various IP Trace back Strategies and Simulation of IP Trace back," *Asian Journal of Information Technology*, Medwell Journal 2008, pp 454-458.

[5] S. Bellovin *et al*., "ICMP Traceback Messages," *IETF Internet Draft*, Version 4, Feb. 2003 (Work in progress).

[6] C. Gong and K. Sarac, "IP traceback based on packet marking and logging," *in Proc. of IEEE International Conference on Communications*, Seoul, Korea, May 2005.

[7] A. Belenky and N. Ansari, "IP Traceback with Deterministic Packet Marking," *IEEE Commun. Lett.*, vol. 7, no 4, Apr. 2003, pp. 162-164.

[8] R. Shokri, A. Varshovi, H. Mohammadi, N. Yazdani, and B. Sadeghian,"DDPM: Dynamic deterministic packet marking for IP traceback," in *Proc. IEEE International Conference on Networks*, vol. 2, Singapore, Sep. 14–15, 2006, pp. 1–6.

[9] S. Savage, D. Wetherall, A. Karlin and T. Anderson, "Practical network support for ip traceback," in *Proceedings of the 2000 ACM SIGCOMM Conference*, August 2000, pp. 295-306.

[10] C. Gong and K. Sarac, "Toward a Practical Packet Marking Approach for IP Traceback," *International Journal of Network Se-curity (IJNS)*, vol. 8, no, 3, pp. 271‒281, May 2009.

[11] A. Yaar, A. Perrig, D. Song, "FIT: fast Internet traceback," *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, vol.2, no., pp. 1395-1406 vol. 2, 13-17 March 2005.

[12] T. Akyuz and I. Sogukpinar, "Packet Marking With Distance Based Probabilities for IP Traceback," *first national conference Networks and communication*, pp. 433-437, 2009.

[13] Z. Wan, Y. Zhang, T. Cao, M. Wu, and F. Wang, "A novel Authenticated Packet Marking Scheme for IP Trace-back, " *Computer Science and Information Technology*, pp. 150-153, Aug. 2009.

[14] Y. Jing, X. Wang, X. Xiao, and G. Zhang, "A Logless Fast IP Traceback Scheme Against DDoS Attacks in Wireless Ad-hoc Network," *Wireless, Mobile and Multimedia Networks*, Nov. 2006.

[15] L. Yang, S. Song, W. W. Su, Y. F. Wang, and H. Wen, "The Performance Analysis of LT Codes," *Communications in Computer and Information Science*, vol. 265, pp. 227-235, 2012.

[16] T. Gamer and M. Scharf, "Realistic Simulation Environments for IP-based Networks," *OMNeT++ 2008: Proceedings of the 1st International Workshop on OMNeT++* (hosted by SIMUTools 2008).