Mansoura Engineering Journal

Volume 46 | Issue 4 Article 16

12-1-2021

HDSL: A Hybrid Distributed Single-packet Low-storage IP Traceback Framework.

Magdy M. Fadel
Chief engineer of computers and systems, Faculty of Engineering, Mansoura University

Mansoura University
Faculty of Engineering
Mansoura Engineering Journal

HDSL: A Hybrid Distributed Single-packet Low-storage IP Traceback Framework

Magdy M. Fadel*

KEYWORDS:
DDoS attacks; IP traceback; packet marking; packet logging; storage overhead; pushback.

***Abstract*— Many problems with IP protocol design facilitate the mission of the Distributed Denial of Service (DDoS) attackers. This paper proposes a new Hybrid Distributed Single-packet Low-storage (HDSL) IP traceback framework, which consists of three enhanced DDoS defense mechanisms. The first mechanism is a Deterministic Packet Marking (DPM) to compose a unique path identifier for validating network paths. The second is a low-storage space packet logging to locally log routed packets information which is used later for locating the source of even a single attacking packet. The third, pushes the aggregates of the attacking packets upstream one or more levels to alleviate the congestion occurred at or near the target to legitimate packets dropping. Three algorithms are developed for this purpose. An Intrusion Detection System (IDS) is also used to administrate the defense modules of the framework, and managing network information. Experimental results show that the traceback performance is improved from many aspects. First, the percentage of false edges returned is decreased as a result of the proposed accurate low collision path identifiers. Also, the required logging space is reduced to more than 70% of other mechanisms. Finally, the ratio of the legitimate packets dropped due to attacking packets congestion aggregates potentially decreased for deploying the pushback principle.**

I- INTRODUCTION

There is no doubt that many DDoS attacks have been reported daily all over the world, these attacks aim to hinder legitimate users from accessing a corporation services or resources, resulting in a revenue loss. The attackers rely on the fact that Internet routing infrastructure is mainly concerned by scalability rather security, since routers neither validate source IP address nor log information regarding the forwarded packets [1]. DDoS attack methods could be classified according to the number of attacking packets into flooding and vulnerability attack [2, 3]. In flooding attack, which is the most common, the attacker sends a huge number of packets to the victim to overwhelm its resources, which may be network level like routers computation power or bandwidth [4], or application level like storage capacity, or CPU processing power [5]. In vulnerability attack, which is harder to defend, the attackers exploit some weaknesses in the design of victim's protocols or applications and send a few or even one packet to get the target system down [3, 4].

Research community spent a lot of time and effort developing various DDoS attacks combating mechanisms before, during and after they take place, identifying the pros and cons of each. The majority of these mechanisms could be grouped into three main categories, packet marking [6-8], packet logging [9, 10], and ICMP Traceback [11-13]. Packet marking mechanisms are composed of two procedures, the first is carried out by routers to encode routers information into

Received: (04 October, 2021) - Revised: (15 December, 2021) - Accepted: (22 December, 2021)

***Corresponding Author*:** *Magdy M. Fadel, Chief engineer of computers and systems, Faculty of Engineering, Mansoura University, (e-mail: mfares@mans.edu.eg & mfares73@yahoo.com).*

(Ser. NO. BFEMU-2110-1173)

packets, whereas the second is carried out by victim to use these encoded information for reconstructing the attack graph [14, 15]. Packet marking by itself is classified into two types, Probabilistic Packet Marking (PPM) and Deterministic Packet Marking (DPM) [16-19]. In PPM every packet contains information about only one router of the attack path, so the victim must collect many packets to reconstruct the attack path correctly defining a critical term known as termination condition [20]. Whereas in DPM, every packet should carry information about all the routers in the attack path which represents a great challenge, then this information could be used to distinguish attack packets from legitimate packets at the victim's firewall [21-24]. Packet logging mechanisms, logs for packets digests are stored locally at routers or packet monitors that listen to router interfaces, these routers form an overlay network that victim could query them later about specific packet to locate its source. But large space and access time requirements are the main drawback of this mechanism [25, 26].

Internet Control Message Protocol (ICMP) Traceback or simply (ITrace) mechanisms, the routers along the attacking path probabilistically send traceback messages generated for some packets. These traceback messages are later collected to reconstruct the attacking path. However, these traceback messages will exhaust the network bandwidth specially with large scale DDoS attacks.

Previous methods failed to satisfy the requirements of accurate trackback with a little computation and storage overhead, so nowadays all proposed mechanisms tend to consolidate them getting a hybrid system that mitigate drawbacks and empower strength points [27-34].

The main contribution of this paper is:

- Implement a deterministic packet marking technique to compose a distinctive mark for every network path, these marks are used to differentiate the source of each packet regardless its source IP address.
- Use a low storage packet marking technique to locally log packet information, for tracing back the source of even a single packet.
- Finally, the deployed packet marking facilitate the applicability of the pushback principle decreasing the number of legitimate packets dropped due to congestion at the victim.

The proposed framework is evaluated by comparing it with the Source Path Isolation Engine (SPIE), Hybrid IP Traceback (HIT), Precise and Practical IP Traceback (PPIT), and Path Address Scheme (PAS) both mathematically and experimentally.

The reminder of this paper is composed of the following sections. Section 2 presents the background and related work. Section 3 introduces the proposed IP traceback framework in detail. Section 4 shows the ability of using the pushing back technique with the proposed framework to move the filtering process upstream one or more level, Section 5 evaluates and compares the proposed framework by mathematical analysis and simulation with other different frameworks, Section 6 introduces deployment and security points, Finally the paper is summarized in section 7.

II- BACKGROUND AND RELATED WORK

Any traceback framework should have some important features, first sharp differentiation between legitimate and attack packets that enable the victim's firewall system to precisely filter out attack packets, resulting in low false positive and negative rates. Moreover, ability of locating the source of the attack packets even if the attacker spoofed the source IP address of the packet. Also, a minimum amount of storage space and access time at routers while logging packet digests. Finally, a low computation overhead in the process of reconstructing the attack path.

2.1 Packet Marking Mechanism

All introduced packet marking mechanisms aim to encode path information represented by edge router's IP addresses or identifiers inside the header of routed packets as they traverse from their source to destination. Probabilistic Packet Marking (PPM) and Deterministic Packet Marking (DPM) are its most common types. Savage et al. [14] first introduce PPM, in which intermediate routers encode their IP addresses or identification information into the 16-bit identification field. The drawback of this mechanism is that each packet carries the IP address of only one router from the whole path, so the victim must collect a huge number of packets which directly proportional with the length of packet's path. Introducing a critical point known as termination condition [20], defined as the exact number of packets collected by the victim not less or more than necessary to reconstruct the exact attack path. Despite that this method was the first step in packet marking and many improvements have been done on it, but it is not that practical in tracing DDoS attacks, since it needs a high computation overhead at the victim to reconstruct the attack path, also gives a higher rate of false positive and negative edges.

Yaar et al. [21, 23] show that tracing back the attack packet to know its source is not that important, introducing a new Deterministic Packet Marking (DPM) mechanism. In this mechanism all packets are marked by all routers identifiers through their paths, resulting in packets originated from the same source carry the same mark or path identifier, so these marks could be used later to precisely differentiate attack packets and filter them out at the victim. However the main challenge in this mechanism is to find a perfect method to accumulate all routers identifiers in this restricted 16-bit packet identification field in the packet's header, without getting multiple packets coming from different sources having the same mark, known as mark's collision problem. DPM mechanisms are packet tracing rather than source traceback mechanism, they need a low computation and storage overhead at victim and routers.

2.2 Packet Logging Mechanism

In router based traceback mechanisms, routers calculate the packet digest of every packet passing through them, space efficient data structure (Bloom filter) [35] could be deployed to minimize the size of these digests, so reducing the storage capacity required by each router. The digests and their time-stamps are stored locally at routers for future use in the packet's traceback process.

Source Path Isolation Engine (SPIE) is the first practical deployment of packet logging mechanism, introduced by Snoeren et al. [36]. In SPIE, a traceback server/group of servers that are previously supplied by the network topology start(s) the traceback process from the victim's side towards the attack by querying the routers about a certain packet at specified time, then the queried router calculates the packet digest searching it in its digest table at the specified time, returning the result of searching back to the traceback server that repeats the query with next and next routers until identifying the exact packet's source (attacker).

SPIE has many advantages, for example it can be used to traceback even a single packet, also it is immune against attacks like IP spoofing and packet transformation like fragmentation and tunneling.

However, there are some disadvantages for SPIE especially for router's memory, since they should have a large storage space with a high speed access time which improves the digests storing and searching processes.

Any perfect traceback framework should be continuously monitor, precisely differentiate between legitimate and attack packets, and instantly response, which could be achieved using a hybrid distributed framework.

III- THE NEW HYBRID DISTRIBUTED SINGLE-PACKET LOW-STORAGE (HDSL) IP TRACEBACK FRAMEWORK

A new Hybrid Distributed Single-packet Low-storage (HDSL) IP Traceback Framework is presented in this section. It is mainly based on the marking mechanisms in [28, 37], and the logging mechanisms in [26, 38] utilizing the 16-bit identification field in the packet's header. 4-bits is defined for the Logging Flag (LF) part for packet logging, and 12-bit to hold the path identifiers of the Deterministic Packet Marking (DPM). This way the utilization reduces the storage and computation overhead required at routers. Figure 1 shows the main phases of the proposed framework. It is composed of four phases, Protecting Communication Network (PCN), Attack Detection (AD), Attack Traffic Control (ATC), and Attack Packet Source Traceback (APST), which will be introduced in the following subsections in detail.

3.1 Protecting Communication Network (PCN) phase.

Since the DDoS attacks are originated from various sources in different networks, so the defense framework components should be distributed in proper points all over the network. So, a secure channel for exchanging network and control information among these components should be found.

3.2 Attack Detection (AD) phase.

Figure 2 shows the main modules of the Attack Detection (AD) phase and the interaction between them. The Attack Detection (AD) phase consists of two modules, packet marking and detection. The packet marking module is implemented at every traceback enabled router to insert the router's mark or identification information in every packet passing through it.

Many packet marking mechanisms have been proposed previously in different papers, the proposed marking mechanism introduced in this paper is shown in figure 3, the 16-bit identification field in the IP Header of the packet will be used as usual since it is used for packet fragmentation which are rarely used in practice, but it will be divided into two parts. The first 4-bits named logging flag (LF) field, this field is used by the routers in the packet logging mechanism as introduced in the next sub-sections. The next 12-bit are used in packet marking mechanism, since as Muthuprasanna et al. introduced in [39], 12-bit are enough as routers ID to distinguish them within two-hop neighboring.

So, a 32-bit to 12-bit hash function H ( ) is used to convert the 32-bit router's IP address (A) to a 12-bit length hash value. As the IP addresses of Internet routers can be easily known, and by knowing also the hash function used by the router, the router mark could be easily spoofed by attacker, so a 12-bit router's random number (KR) is generated. This value is XORed with the hash value of the router IP address getting the current router mark (MR), ($MR = H(A) \oplus KR$), router will compute and store the value of its mark (MR) locally only once.

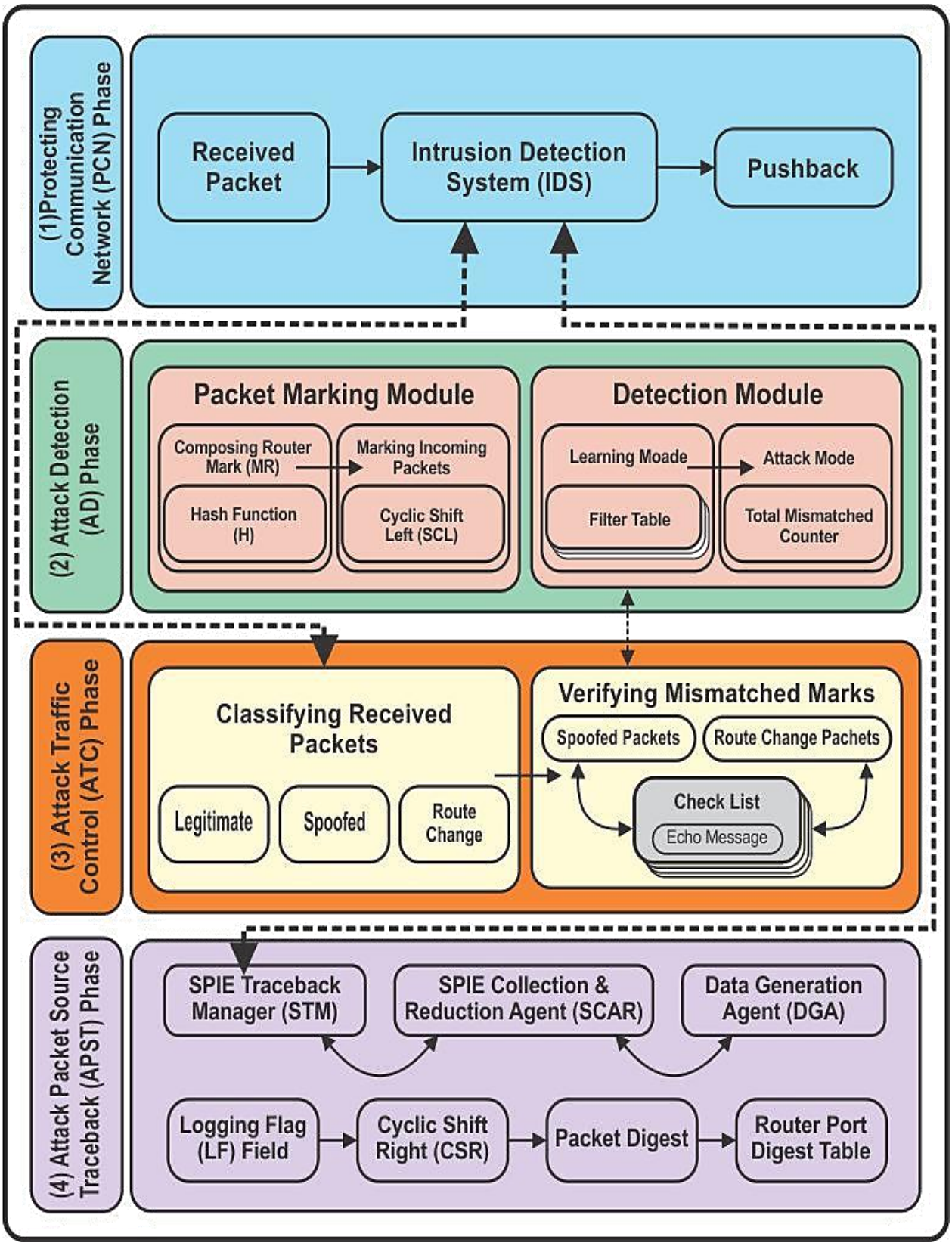


Figure 1: The new Hybrid Distributed Single-packet Low-storage (HDSL) IP Traceback Framework.

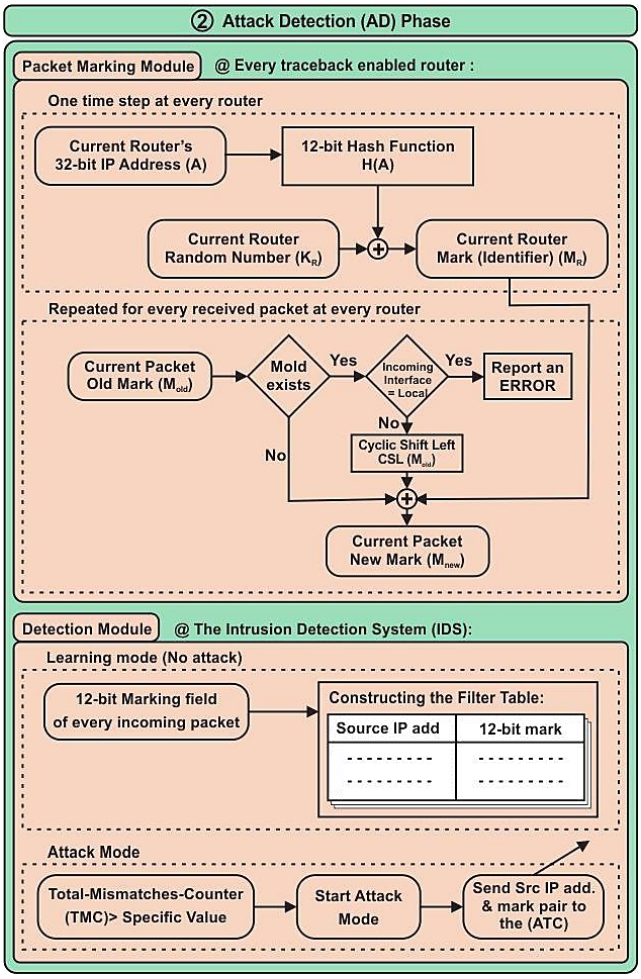


Figure 2: The Attack Detection (AD) phase.

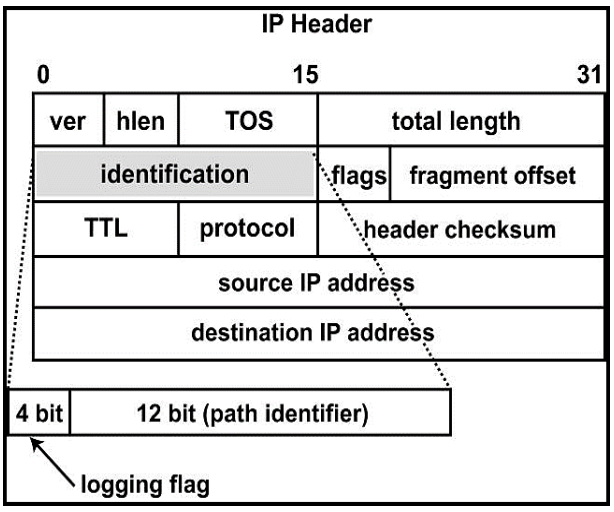


Figure 3: Utilizing the Identification field for packet logging and marking [39]

This router's identification value (MR) is encoded in the header of every going packet later, Chen et al. in [37] proposed using the XOR operation to accumulate the identification values of all routers the packet will pass through along its path from the sender to the receiver, without any increase in the size of 12-bit identification field in the IP header of the packet. Besides this fixed size advantage of XOR, it is also so easy to remove the router's ID value from the path identifier in the packet header by XORing the router's ID again with that mark (where: (A ⊕ B) ⊕ B = A). However, there is a drawback for using XOR operation in composing the packet's path identifier, if two packets pass through the same group of routers but in different order, the path identifier for both of them will be equal (Since: (A ⊕ B) ⊕ C = (C ⊕ B) ⊕ A), returning an incorrect attacker in some cases, figure 4 shows the problem.

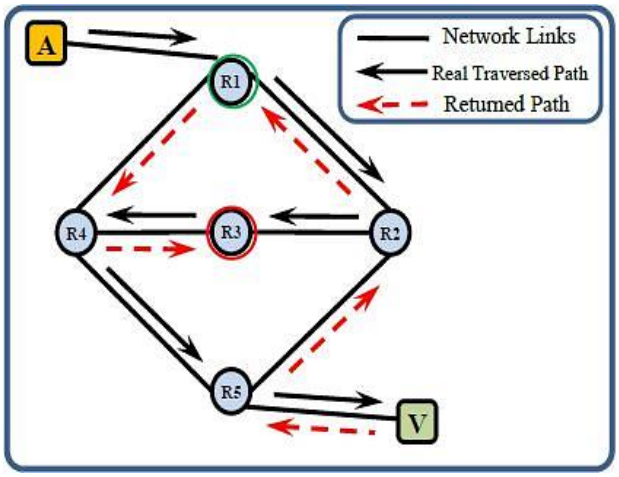
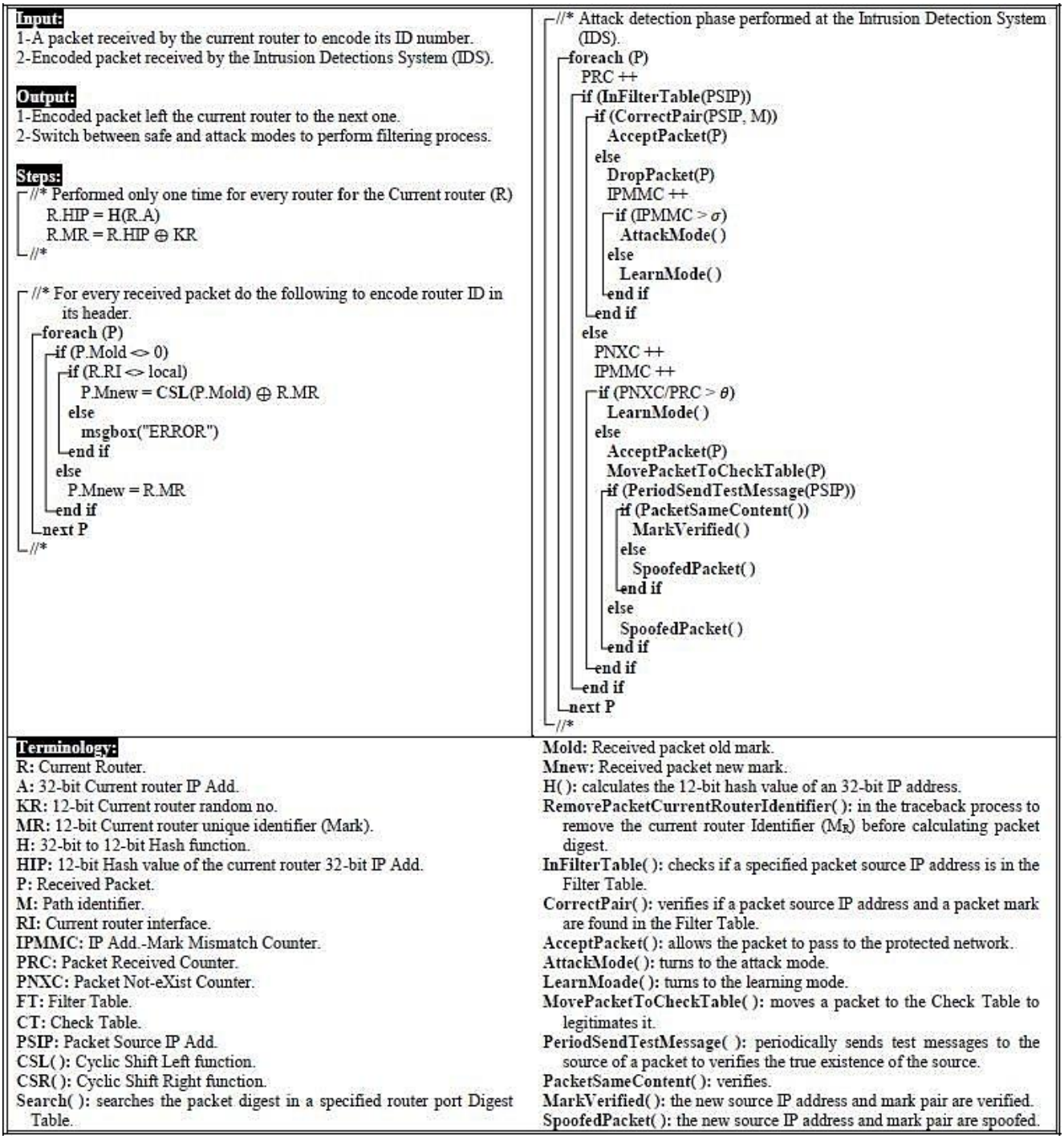


Figure 4: Different orders of the same path [37].

There are two methods to cope this problem, first unmasking the Time-To-Life (TTL) field in the packet header when calculating the packet digest at the router [26], however this method poses some problems for preserving networks confidentiality that is not preferred by service providers. The other method [37] performs a Cyclic-Shift-Lift (CSL) on the old mark (Mold) in the packet header before XORing it with the current router identifier (MR), (i.e.: Mnew = CSL(Mold) MR), and performing Cyclic-Shift-Right (CSR) operation in the traceback operation as shown in the next subsections. The second method is preferred by service providers since it does not exposures their network's confidentiality.

The Detection module is the second part of the Attack Detection (AD) phase that is implemented at the Intrusion Detection Systems (IDS) at the perimeter of the protected networks. Where every IDS constructs a Filter Table (FT) that records source IP address and mark pair. When the number of Mismatches-counter (TMC) exceeds a pre-specified value, the IDS turns from learning mode to the attack mode that will be explained in next sections. Figure 5 shows the Attack Detection (AD) Algorithm.



Input:
1-A packet received by the current router to encode its ID number.
2-Encoded packet received by the Intrusion Detections System (IDS).

Output:
1-Encoded packet left the current router to the next one.
2-Switch between safe and attack modes to perform filtering process.

Steps:

```
//* Performed only one time for every router for the Current router (R)
   R.HIP = H(R.A)
   R.MR = R.HIP ⊕ KR
//*

//* For every received packet do the following to encode router ID in
     its header.
foreach (P)
  if (P.Mold <> 0)
    if (R.RI <> local)
      P.Mnew = CSL(P.Mold) ⊕ R.MR
    else
      msgbox("ERROR")
    end if
  else
    P.Mnew = R.MR
  end if
next P
//*
```

```
//* Attack detection phase performed at the Intrusion Detection System
   (IDS).
foreach (P)
  PRC ++
  if (InFilterTable(PSIP))
    if (CorrectPair(PSIP, M))
      AcceptPacket(P)
    else
      DropPacket(P)
      IPMMC ++
      if (IPMMC > σ)
        AttackMode( )
      else
        LearnMode( )
      end if
    end if
  else
    PNXC ++
    IPMMC ++
    if (PNXC/PRC > θ)
      LearnMode( )
    else
      AcceptPacket(P)
      MovePacketToCheckTable(P)
      if (PeriodSendTestMessage(PSIP))
        if (PacketSameContent( ))
          MarkVerified( )
        else
          SpoofedPacket( )
        end if
      else
        SpoofedPacket( )
      end if
    end if
  end if
next P
//*
```

Terminology:

R: Current Router.
A: 32-bit Current router IP Add.
KR: 12-bit Current router random no.
MR: 12-bit Current router unique identifier (Mark).
H: 32-bit to 12-bit Hash function.
HIP: 12-bit Hash value of the current router 32-bit IP Add.
P: Received Packet.
M: Path identifier.
RI: Current router interface.
IPMMC: IP Add.-Mark Mismatch Counter.
PRC: Packet Received Counter.
PNXC: Packet Not-eXist Counter.
FT: Filter Table.
CT: Check Table.
PSIP: Packet Source IP Add.
CSL( ): Cyclic Shift Left function.
CSR( ): Cyclic Shift Right function.
Search( ): searches the packet digest in a specified router port Digest Table.
Mold: Received packet old mark.
Mnew: Received packet new mark.
H( ): calculates the 12-bit hash value of an 32-bit IP address.
RemovePacketCurrentRouterIdentifier( ): in the traceback process to remove the current router Identifier (M_R) before calculating packet digest.
InFilterTable( ): checks if a specified packet source IP address is in the Filter Table.
CorrectPair( ): verifies if a packet source IP address and a packet mark are found in the Filter Table.
AcceptPacket( ): allows the packet to pass to the protected network.
AttackMode( ): turns to the attack mode.
LearnMoade( ): turns to the learning mode.
MovePacketToCheckTable( ): moves a packet to the Check Table to legitimates it.
PeriodSendTestMessage( ): periodically sends test messages to the source of a packet to verifies the true existence of the source.
PacketSameContent( ): verifies.
MarkVerified( ): the new source IP address and mark pair are verified.
SpoofedPacket( ): the new source IP address and mark pair are spoofed.

Figure 5: Attack Detection (AD) Algorithm.

3.3 Attack Traffic Control (ATC) phase.

Figure 6 shows the block diagram of the Attack Traffic Control (ATC) phase modules and the interaction among them.

This phase is implemented at the IDS when discovering an attack, any packet state could be one of three, the source IP address and the mark pair is found in the Filter Table that means the packet is legitimate and the IDS will accept it. Or, the source IP address found in the Filter Table but with a wrong mark that means it may be a spoofed packet or the packet has changed its route, then the IDS should move this pair to the Check List (CL) to verify its state. Finally, the source IP address is not found in the Filter Table that means this is a new packet and the IDS should move it to the Check List for verification.

Verifying an entry in the Check List is done by sending periodically a fixed number of echo messages with a specific content to the tested source IP address, and upon receiving a reply with the same content, the source IP address and mark pair is verified and they are moved to the Filter Table. Otherwise, the source IP address is spoofed and it will be moved to the

Filter Table with a special mark "ex. **" that indicates a spoofed source IP address to be dropped.

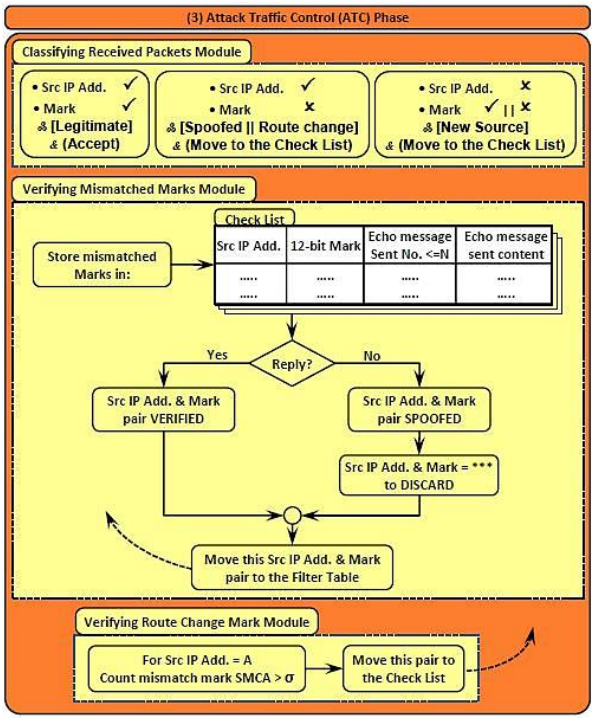


Figure 6: The Attack Traffic Control (ATC) phase.

Figure 7 shows the Attack Traffic Control (ATC) algorithm.

3.4 Attack Packet Source Traceback (APST) phase.

Figure 8 shows the Attack Packet Source Traceback (APST) phase modules, used to traceback the attack packet to its source using the logging flag (LF) field in the packet's header and the packet's digests logged at the routers.

It is consisting of the three modules, Source Path Isolation Engine (SPIE) [26], SPIE Traceback Manager (STM), SPIE Collection & Reduction Agent (SCAR), and Data Generation Agent (DGA). But the contribution in this paper is that, instead of logging the packet's digests in every router along the path, a new LF field of 4-bit (maximum of 16 decimal values from 0 to 15) has been introduced used to decide if the logging will be done every 2, 3 or even 15 routers in the path. Resulting in a decreased storage space required at the routers.

Input:
1-Packet Received by the Intrusion Detection System (IDS)

Output:
1-Specifying the different states of the received packet
2-Verify the packet with a mismatch Source IP Address or Mark

Steps:

```
//* Specify the Received Packet State (Legitimate, Spoofed or Route
   change, or a new packet source)

  if ( CorrectPair(P.PSIP, P.M) )
     msgbox("Legitimate Packet")
     AcceptPacket(P)
  else if ( InFilterTable(P.PSIP) )
     msgbox("Spoofed or Route change")
     IPMMC++
     MovePacketToCheckTable(P)
     else
        msgbox("A new Packet")
        IPMMC++
        AcceptPacket(P)
        MovePacketToCheckTable(P)
  end if
end if
//*
```

```
//* Verifying Mismatched Marks (Spoofed or Route change, or a New
   Packet Source)

  foreach (P) in the Check Table
     SIPAEMC[i]++
     while (SIPAEMC[i] < 10)
        PeriodSendTestMessage (PSIP, SIPAVT[i])
        Delay (20)
        if (GotReply( ) && PacketSameContent( ))
           MarkVerified( )
           MovePacketToFilterTable("Verified")
           SIPAEMC[i] = 10
        end if
        if(SIPAEMC[i]==9)
           SpoofedPacket( )
           MovePacketToFilterTable("Spoofed")
        end if
     next
  next P
//*
```

Terminology:
P: Received Packet.
M: Path Identifier.
PSIP: Packet Source IP Add.
IPMMC: IP Add.-Mark Mismatch Counter.
SIPAEMC: Source IP Add. Echo Message Counter.
SIPAVT: Source IP Add. Verification Text.

Delay( ): Delay in seconds.
GotReply( ): get a reply from a specific source IP Add.
MovePacketToFilterTable( ): Moves the Specified Packet to the Filter Table with "Legitimate" or "Spoofed" sign.
CorrectPair( ): Check if the Packet Source IP Address and Mark pair exist in the Filter Table.

Figure 7: The Attack Traffic Control (ATC) Algorithm.

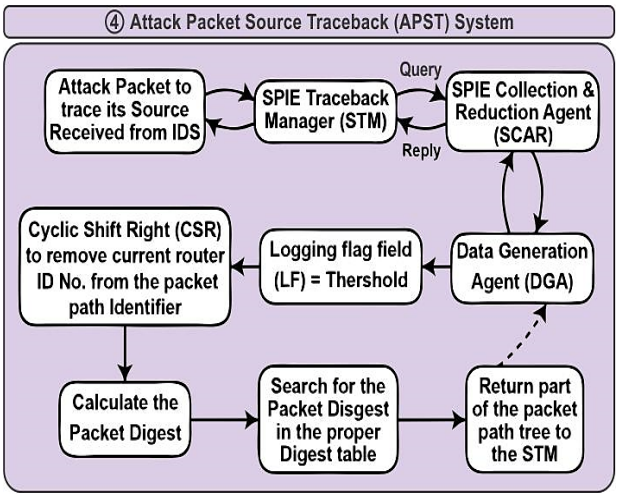


Figure 8: The Attack Packet Source Traceback (APST) phase.

Figure (9-a) shows the complete packet marking and logging with the logging flag (LF) field less than 2, this means that the LF field will take only two values 0 and 1, and the logging process will happen when the router finds the LF field equals to 0. This means that the logging ratio is reduced to 50% of SPIE technique.

Figure (9-b) shows the source traceback operation, to trace the source of a given packet from the victim to the attacker.

Figure 10 shows the Attack Packet Source Traceback (APST) algorithm

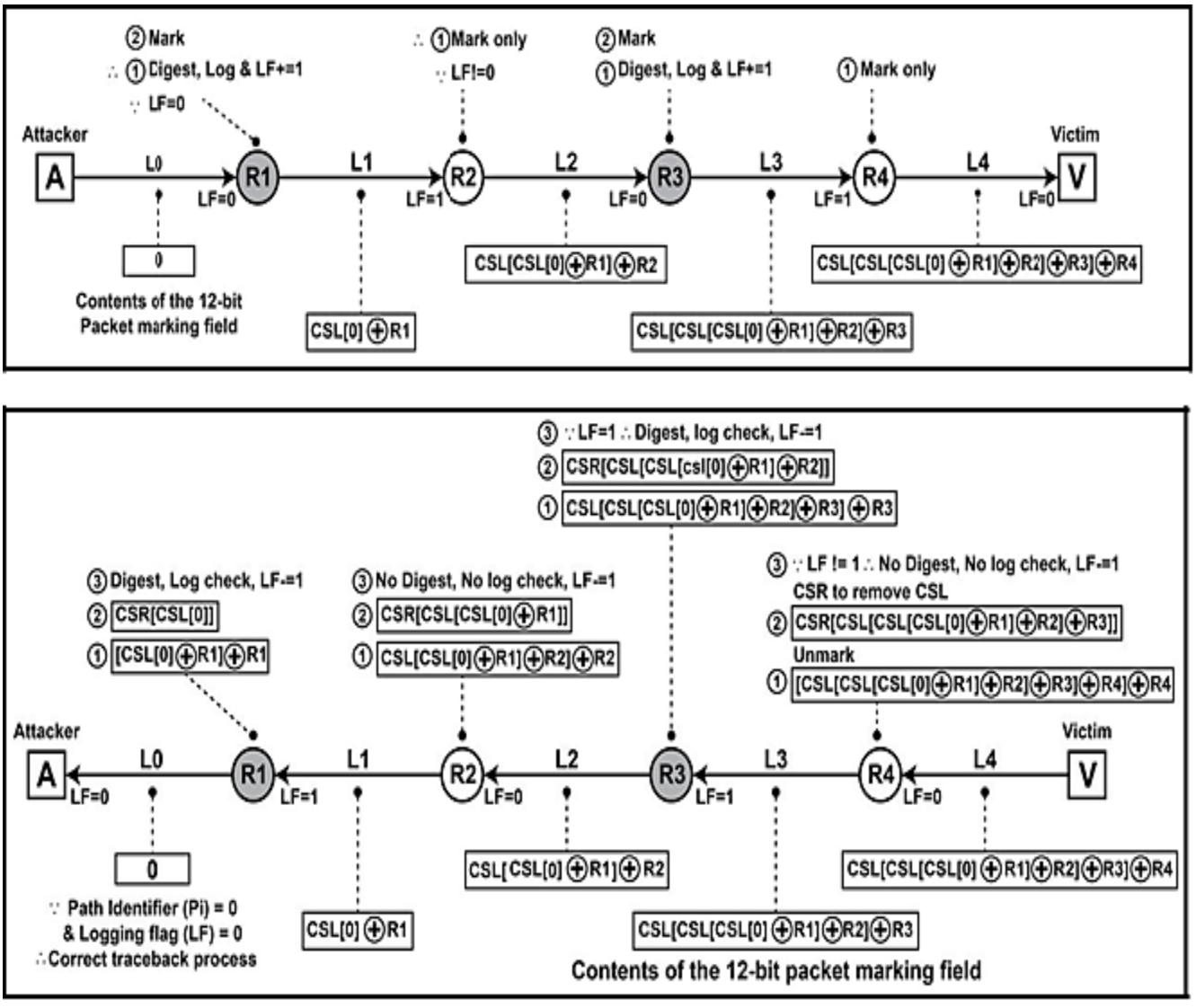


Figure 9: With LF<2 a complete (a) Packet marking and Logging, (b) Packet Traceback.

Input:
1- Sending the attack packet detected by the IDS to the STM.
2- Sending the time stamp of the attack packet from IDS to the STM.

Output:
1-The full path of the attack packet (The attacker source).

Steps:

```
//* For every packet received by the Data Generation Agent (DGA) do
the following to locate partial path of its final source.

//* Remove the Current Router ID mark performing both XOR
operation and Cyclic Shift Right (CSR).

foreach (P)
  if (P.M <> NULL)
    if (P.LF = 0)
      msgbox("Correct Packet Source")
    else
      msgbox("Spoofed Packet")
    end if
  else
    P.M = P.M⊕R.MR
    P.M = CSR(P.M)
  end if
next P
//*

//* Check the value of the Logging Flag (LF) field if it equals the max
value (1) with a threshold value equals 2 (takes 0 or 1)

foreach (P)
  if (P.LF == 1)
    PDV = PDF(P)
    foreach (R.RP)
      RRP = search(P.PDV, R.RP)
    next R.RP
  end if
  P.LF –
next P
//*
```

Terminology:
RP: Router Port
RRP: Returned Router Port
K: pre-specified logging flag threshold
PDV: Packet Digest Value
PDT: Packet Digest Table
RPDT: Router Port Digest Table
CSR( ): Cyclic Shift Right
PDF( ): Packet Digest Function
Search( ): search a packet digest in a specified router port digest table.

Figure 10: The Attack Packet Source Traceback (APST) algorithm.

Also figure 11 shows the proposed framework's flow chart.

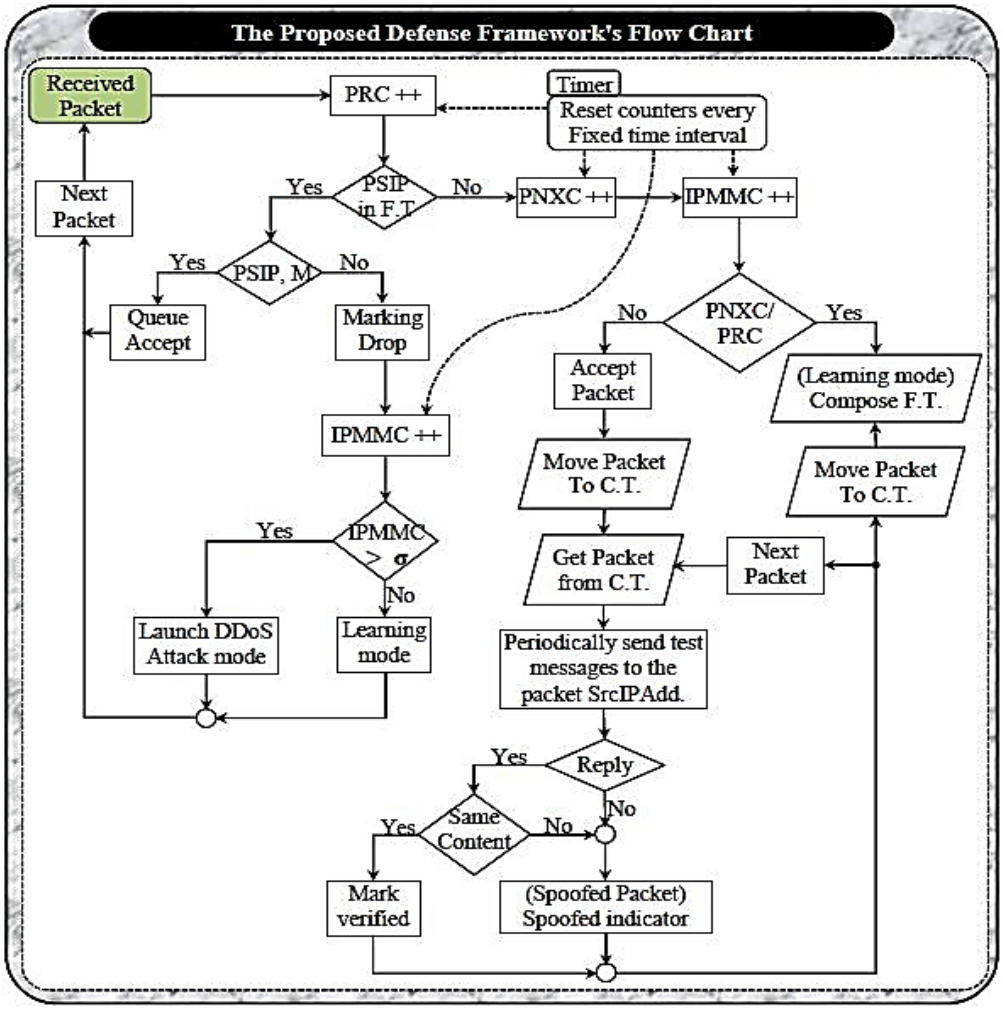


Figure 11: The proposed framework's flow chart

IV. PUSHBACK TECHNIQUE APPLICABILITY

Mahajan et al. introduced in [40] due to a congestion in some network links, some legitimate traffic could be dropped resulting in a degrading performance of the network, figure 12 shows a clear case of this situation. Since link L0 is congested due to DDoS attack, so good traffic coming from link L1 may be dropped before reaching its destination's D. So, if the attack traffic could be filtered one level up, i.e. at routers R2 and R3, traffic coming from link L1 could be flow smoothly to its destination D. But the situation of good traffic from link L1 may be repeated for traffic coming from link L5 and L6, so pushing the attack traffic up another one level to be filtered at routers R4 and R7 respectively, the traffic coming from L5 and L6 can be smoothly flow toward its destinations D. This process of pushing attack traffic far away from the victim guarantees that no legitimate traffic may be dropped due to traffic of DDoS attack. Thanks to the proposed packet marking mechanism introduced in this paper the pushback technique could be easily deployed, as introduced in the next section of performance evaluation and experimental results.

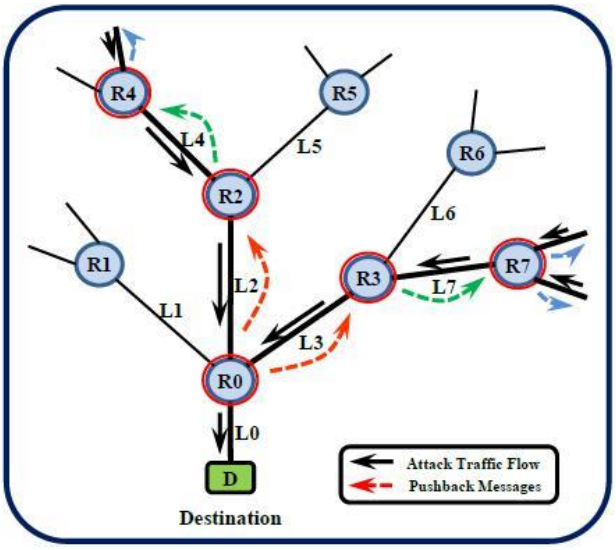


Figure 12: Network Congestion problem [40].

V. PERFORMANCE EVALUATION

The storage space, traceback process overhead, and the accuracy of the proposed framework will be evaluated analytically and by simulation, then the obtained results will be compared with SPIE [26], HIT [28], PPIT [38], and PAS [24] frameworks. SPIE is a direct implementation of the traditional log-based IP traceback mechanism, in which the digests of all passing packets will be logged in every intermediate router that requires a huge logging space and traceback convergence time.

HIT integrates two DDoS defense mechanisms getting a hybrid one, Packet Marking to encode the path information in every passing packet and Packet Logging to store all passing packet digests in the intermediate routers. This integration reduces the traceback process time and the number of false edges returned. PPIT improves the performance of the HIT by utilizing the 16-bit limited marking space in the packet header into a 2-bit hop count used to selectively log the passing packet digests, and 14-bit to encode the passing router's ID numbers into the packet header. Resulting in a noticeable decrease in the storage space required at intermediate routers. Finally, in PAS IP addresses stored at every terminal host are replaced by path addresses stored at intermediate routers. Where every traceback enabled router is assigned a unique number used to compose an authentic mark for each network path, these authentic marks could be used to filter out and push back the attacking traffic improving the overall performance of the defense system.

5.1 Routers storage space overhead

Packets will be logged locally by routers in the proposed framework in the following cases:

(1) IP fragments,

(2) Non-fragmented packets need to be logged at routers, which include:

(a) Non-fragmented packets not logged in the k upstream routers.

(b) Non-fragmented packets logged at the direct upstream router but transformed at the current router.

(c) Non-fragmented packets logged in one of the upstream routers between case (a) and case (b) above.

TABLE 1
PERCENTAGE OF LOGGED PACKETS AT DIFFERENT CONDITIONS.

	Packet type	Percentage
1.	IP fragments.	a
2.	Non-fragmented packets not logged in the upstream routers (includes 2.1. 2.2 and 2.3 below).	(1-a)Q
2.1	Non-fragmented packets not logged at any k upstream routers.	$(1-a)(1-Q)^k$
2.2	Non-fragmented packets not logged at the direct upstream router but transformed at the current router.	(1-a)Qb
2.3	Non-fragmented packets logged in one k upstream router and not logged at direct upstream router, but transformed in the current router.	$(1-a)[\sum_{n=1}^{k}(1-Q)^{n-1}]$

Suppose the following letters refer to the types of packets forwarded by a router: P the percentage of packets to be logged at the router, the percentage of IP fragmented packets, b the percentage of transformed packets and Q the percentage of packet not fragmented but needed to be logged at the router. Table 1 show the percentage of logged packets at different conditions. From these parameters we get:

$$P = a + (1-a)\,Q \quad (1)$$

And Q could have the following value:

$$Q = f(x) = \begin{cases} 1, & k = 0 \\ (1-Q)^k + Qb\sum_{n=1}^{k}(1-Q)^{n-1}, & k > 0 \end{cases} \quad (2)$$

Where: k refers to the number of routers between two successive logs that can take values from 0 to maximum of 15,

since it represents the Logging Flag (LF) field of 4-bits in the identification field of the packet header. From Equation (1) the value of Q could be expressed as:

$$Q = \frac{(P-a)}{(1-a)} \qquad (3)$$

TABLE 2
DIFFERENT PERCENTAGES OF PACKET LOGGING WITH DIFFERENT K VALUES

Value of k	Approximate value of P	Notes
0	1	SPIE
1	0.5	HIT
2	0.375	PPIT
3	0.275	HDSL
4	0.225	HDSL

As McCreary et al. mentioned in [41] a ≤ 0.25% and Border et al. mentioned in [42] b ≤ 3% and substituting the value of Q from equation 3 in equation 2 with values of k ranging from 0 to 15 the approximate value of the percentage of packets needed to be logged at the router P is shown in Table 2.

5.2 Traceback process overhead

To traceback the source of a packet, the traceback server queries the routers in the network starting from its nearest router with the help of the value of k in the logging flag field, figure 9-b shows the complete traceback process with the value of k equals 2 and a path length of 5 network segments and 4 intermediate traceback enabled routers with a separate digest table for every router interface. Logically, the traceback process overhead is directly proportional to the number of queries at routers and digest tables that will decrease directly by increasing the value of k. Let NRk and NRa represent the queried routers during the traceback process in the proposed algorithm and SPIE algorithm, respectively. So,

$$NRk = P \times NRa \qquad (5)$$

5.3 Traceback accuracy

The traceback system accuracy could be defined as the number of false positive edges obtained from the process compared with the actual packet route. And the reason of this inaccuracy problem may result from the deployment of bloom filters for optimizing the process of recording packet digests in the routers, or form the traceback process by itself.

(a) Using a separate digest table for logging packet digests will result in decreasing the number of searches, which leads to a reduction in errors resulting from using Bloom filters.
(b) Using the simple XOR operation for accumulating and extracting the routers identifiers gives no chance for generating errors leading to reduced number of generated false positive edges.

5.4 Simulation

Network Simulator (NS2) [43] simulations have been done based on AT&T POP-level topology collected by Rocketfuel [44]. These experimental simulations are composed of 200 nodes and 290 links, 50 out of these 200 nodes are core routers which have more than 2 neighbors. Assuming terminal routers are directly connected with end hosts, sending and receiving hosts are randomly chosen, and no fragmentation or transformation operations are applied on packets as they move through the network.

These simulations will measure important DDoS defense metrics concerning packet logging overhead, traceback procedure overhead, and traceback accuracy. The results of these simulations will confirm the analytical results obtained above.

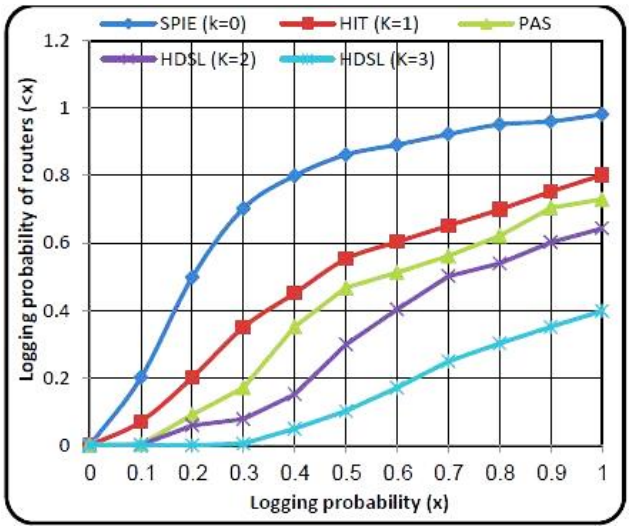


Figure 13: Logging probability vs. various values of k.

The storage overhead in the network resulting from routers logging packets could be measured by calculating the logging probability of routers with different values of k. As shown in figure 13, by increasing the value of k the logging probability of routers will decrease, resulting in a more utilization of storage overhead required by the routers in the network and better performance than HIT and PPIT algorithms, that support the results obtained in Table 1.

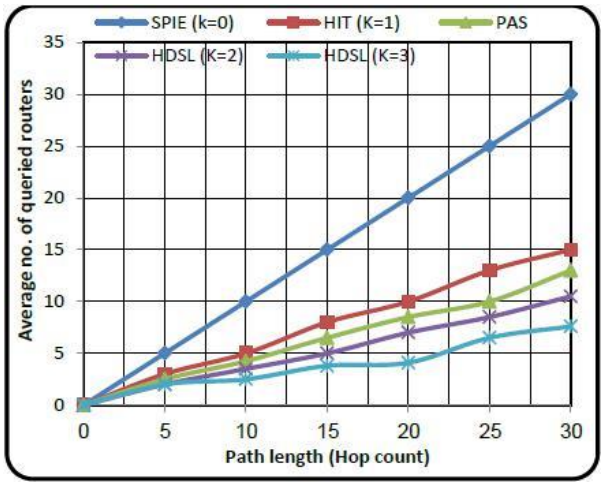


Figure 14: Number of queries vs. path of different lengths with different values of k.

Also examining the number of routers queried to traceback a packet to its source with paths with different length ranging from directly connected to path length of maximum 30 hops. These numbers of queries are used to express the traceback

process overhead, figure 14 shows a comparison between different DDoS algorithms, it is clear that as the value of k increases the number of queries decreases.

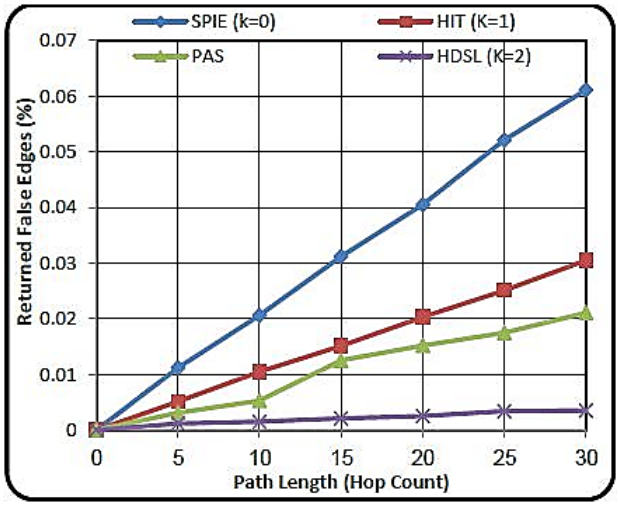


Figure 15: Accuracy of different algorithms with different path lengths.

The number of false edges returned in the attack graph could be used as a tool to measure the accuracy of the traceback system; these false edges could be resulted from the collision between different packet marks, which are very low as a result of using these simple and straight forward XOR operations in the marking accumulating process. Figure 15 shows that with different algorithms and different path lengths.

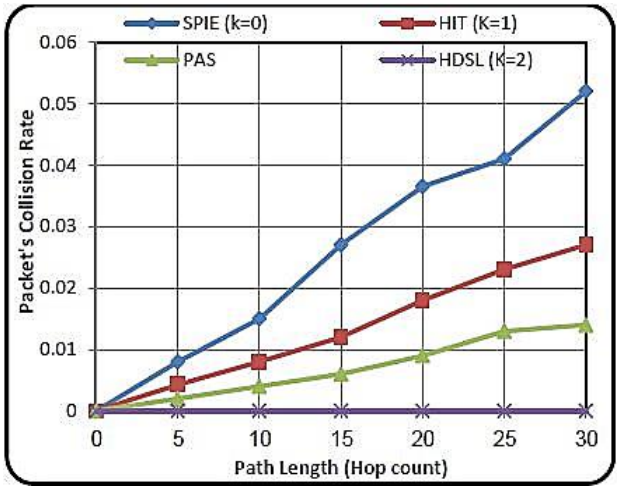


Figure 16: Collision rates of packet identifiers for different algorithms.

Two or more packets of different sources may have the same packet identifiers which known as identifier's collision that results in a failure in the traceback process and filtering process. But thanks to using Cyclic Shit (CS) operations in marking and unmarking processes, the collision rates are very low or approximately zero. Figure 16 shows a comparison between different algorithms in the collision problem.

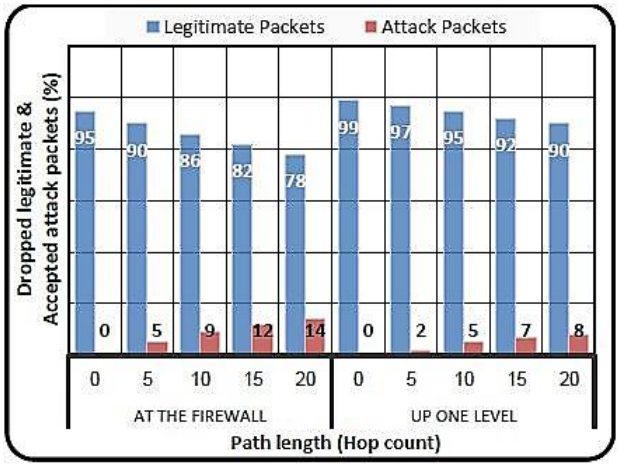


Figure 17: The effect of applying pushback technique in the ratio of acceptance of legitimate and attack packets.

Figure 17 shows two scenarios of the ratio of legitimate and attack packets accepted with different number of attackers at the firewall and one level up. Since as introduced above in section 4, as the number of attack packets increase there may be some legitimate packets dropped due to congestion at the firewall, resulting in a degrading in the network performance. As a solution for this case the filtering process could be pushed up one or more level resulting in an improving in the network performance.

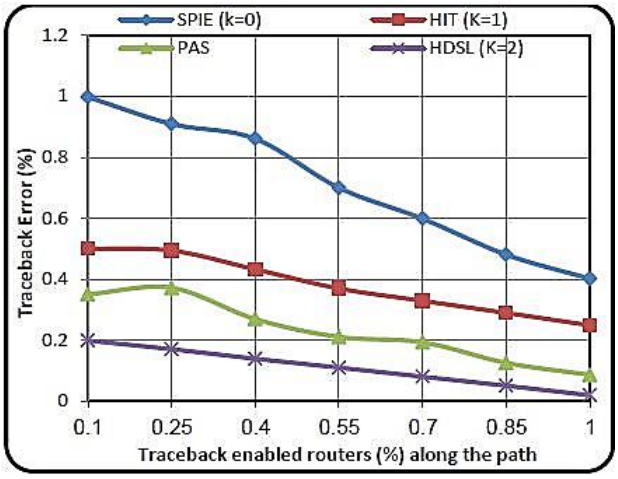


Figure 18: Traceback accuracy vs. legacy routers existence.

In some algorithms the existence of legacy routers may hinder the processes of defense and traceback, for example using the Time-To-Life (TTL) field in the packet header in Pi scheme [23] for keeping the correct order of the passing nodes which results in an error ratio in reconstructing the attack path and filtering packet processes. But in the proposed framework the number of traceback enabled router does not affect greatly the traceback process as shown in figure 18.

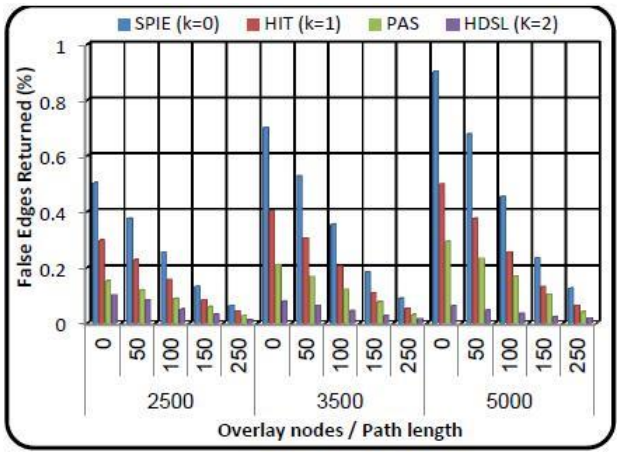


Figure 19: Relation between overlay nodes no. and false edges.

Also, the number of traceback enabled routers deployed on the protected network forms an overlay network. As the number of overlay nodes increase the collision between path identifiers decrease resulting in a high accuracy in the filtering process performed at the IDS, and a high accuracy in the traceback process. Figure 19 shows this point.

VI. Discussion

6.1 Deployment

Packet traceback frameworks could combine two well-known DDoS defense mechanisms, packet logging and packet marking. Packet logging which is used to log the packet digests locally, could be deployed by the router itself or an independent device named network tap. Packet marking encodes the ID information of the passing packets inside their header. The two proposed modifications handle the two drawbacks of the Internet routing infrastructure.

The routers capable of doing these two modifications are named traceback enabled routers, and as the number of these routers increase in the network the more accurate results of the traceback process. These routers together form an overlay network and every router should know its overlay neighbors, besides the traceback server should learn the topology of the overlay network.

6.2 Security

There are two main security problems, the first is spoofing the source IP address of the packet by the attacker, hence the Intrusion Detection System (IDS) could not filter out the packets based on their Source IP address, and instead it will rely on the packet mark that represents the actual path traversed by the packet. The second security problem is that the attacker could write an initial forged mark in the packet marking field which produce a final wrong mark for the packet at the victim, but this case could be discovered by the traceback system since the traceback process will continue until the packet is not logged by any port of the final router, and at this point the traceback server will stop the trace process whatever the marking field is empty or not.

VII. Conclusion

Defending against DDoS attacks is an important searching point today. Many mechanisms have been proposed like packet marking, packet logging, and iTrace. However to precisely filter out attack packets and accurately traceback the attacker cooperation and collaboration among these mechanisms to produce a hybrid defense framework should be done. Actually, many defense frameworks such HIT and PPIT combined the packet marking and logging mechanisms to enhance the defense framework.

This paper utilizes the limited space of the 16-bit identification field in the packet header to log the packet IDs to compose a unique mark for each network path, also the router's packet logging process is managed to selectively log packet's information used later for tracing back even a single attacking packet and the pushback principle is deployed to overcome the traffic congestion problem. Based on these three principles a new Hybrid Distributed Single-packet Low-storage (HDSL) framework is proposed.

The work presented in this paper could be extended by optimizing the Logging Flag (LF) field determining its optimal value that eliminate the tradeoffs between small convergence time and accurate traceback edges returned. Also, despite testing the performance and applicability of the proposed framework in NS2, examining it in a real network is an important stage.

Funding statement:
No financial support was received

Declaration of Conflicting interests statement:

The author declared that there are no potential conflicts of interest with respect to the research authorship or publication of this article.

References

[1] P. J. Criscuolo, "Distributed Denial of Service Tools Trinoo, Tribe Flood Network," Tribe Flood Network 2000, and Stacheldraht CIAC-2319, Department of Energy Computer Incident Advisory Capability (CIAC), UCRL-ID-136939, Rev. 1., Lawrence Livermore National Laboratory, February 14, 2000.

[2] M. Vijayalakshmi, M. Shalinie and N. Neethimani, "A Brief Survey of IP Traceback Methodologies," Department of Computer Science and Engineering, Thigarajar College of Engineering, Thiruparankundram, Madurai, Tamilnadu, India, Vol. 11, no. 9, 2014.

[3] S. T. Zargar, J. Joshi and D. Tipper, "A Survey of Defense Mechanisms Against Distributed Denial of Service (DDoS) Flooding Attacks," IEEE Communications Surveys & Tutorials, Vol. 15, no. 4, 4th Quarter 2013.

[4] B. B. Gupta and Omkar P. Badve, "Taxonomy of DoS and DDoS attacks and desirable defense mechanism in a Cloud computing environment," Springer Neural Computing and Applications Vol. 28, PP. 3655–3682, 2017.

[5] Xiaolin Zhao, Hui Peng, Xiang Li, Yue Li, Jingfeng Xue, Yaoyuan Liang and Mingzhe Pei, "Defending Application Layer DDoS Attacks via Multidimensional Parallelotope," Hindawi Security and Communication Networks, Vol. 2020, 2020.

[6] Mazhar Javed Awan, Umar Farooq, Hafiz Muhammad Aqeel Babar, Awais Yasin, Haitham Nobanee, Muzammil Hussain, Owais Hakeem and Azlan Mohd Zain, "Real-Time DDoS Attack Detection System Using Big Data Approach," MDPI Sustainability, Vol. 13, Issue 19, 2021.

[7] Y. Bhavani, V. Janaki and R. Sridevi, "Survey on Packet Marking Algorithms for IP Traceback," Oriental Journal of Computer Science & Technology, Vol. 10, Issue 2, PP. 507-512, May 2017.

[8] H. Zhang, J. Reich and J. Rexford, "Packet Traceback for Software-Defined Networks," Master's Thesis, Department of Computer Science, Princeton University, 2015.

[9] EunHee Jeong and ByungKwan Lee, "An IP Traceback Protocol using a Compressed Hash Table, a Sinkhole Router and Data Mining based on Network Forensics against Network Attacks," Elsevier Future Generation Computer Systems, Vol. 33, PP. 42-52, April 2014.

[10] Vijayalakshmi Murugesan, Mercy Shalinie Selvaraj and Ming-Hour Yang, "HPSIPT: A high-precision single-packet IP traceback scheme," Elsevier Computer Networks, Vol. 143, PP. 275-288, Oct. 2018.

[11] M. Vijayalakshmi and S. M. Shalinie, "Single Packet ICMP Traceback Technique using Router Interface," Journal of Information Science and Engineering, Vol. 30, PP. 1673-1694, 2014.

[12] Bo-Chao Cheng, Guo-Tan Liao, Ching-Kai Lin, Shih-Chun Hsu, Ping-Hai Hsu and Jong Park "MIB-ITrace-CP: An Improvement of ICMP-Based Traceback Efficiency in Network Forensic Analysis," 9th International Conference on Network and Parallel Computing (NPC), PP.101-109, Sep. 2012.

[13] Brian Cusack, Zhuang Tian and Ar Kar Kyaw, "Identifying DOS and DDOS Attack Origin: IP Traceback Methods Comparison and Evaluation for IoT," International Conference on Interoperability in IoT International Conference on Safety and Security in IoT, Vol. 190, PP. 127-138, Feb. 2017.

[14] Deepthi S. and Arun P. S., "A Survey on IP Traceback Techniques," International Research Journal of Engineering and Technology (IRJET), Vol. 4, Issue 10, Oct. 2017.

[15] L. Cheng, D. M. Divakaran, W. Y. Lim and V. L. L. Thing, "Opportunistic Piggyback Marking for IP Traceback," IEEE Transactions on Information Forensics and Security (IFS), Vol. 11, no. 2, PP. 273-288, February 2016.

[16] T. Peng, C. Leckie and K. Ramamohanarao, "Survey of network-based defense mechanisms countering the DoS and DDoS problems," Journal of ACM Computing Surveys (CSUR), Vol. 39, no. 1, Article 3, April 2007.

[17] U. Tariq, M. Hong and K. Lhee, "A Comprehensive Categorization of DDoS Attacks and DDoS defense Techniques," International Conference on Advanced Data Mining and Applications, Advanced Data Mining and Applications (ADMA 2006), Lecture Notes in Computer Science (LNCS), Vol. 4093, pp. 1025-1036, 2006, Springer, Berlin, Heidelberg.

[18] S. Yu, W. Zhou, S. Guo and M. Guo, "A Feasible IP Traceback Framework through Dynamic Deterministic Packet Marking," IEEE Transactions on Computers, Vol. 65, no. 5, PP. 1418-1427, May 2016.

[19] P. S. Waghmare and U. A. Mande, "Flexible Deterministic Packet Marking: An IP Traceback Scheme Against DDoS Attacks," International Journal of Scientific Engineering and Applied Science (IJSEAS), Vol. 1, no. 6, PP. 440-444 September 2015.

[20] T. Y. Wong, M. H. Wong and C. S. Lui, "A Precise Termination Condition of the Probabilistic Packet Marking Algorithm," IEEE Transactions on Dependable and Secure Computing, Vol. 5, no. 1, Jan-March 2008.

[21] A. Yaar, A. Perrig and D. Song "SIFF: A Stateless Internet Flow Filter to Mitigate DDoS Flooding Attacks," IEEE Symposium on Security and Privacy, 2004.

[22] Y. Xiang, W. Zhou and M. Guo, "Flexible Deterministic Packet Marking: An IP Traceback System to Find the Real Source of Attacks," IEEE Transactions on Parallel and Distributed Systems, Vol. 20, no. 4, pp. 567-580, April 2009.

[23] Andreas Papalambrou, Kyriakos Stefanidis, John Gialelis and Dimitrios Serpanos, "Detection, traceback and filtering of denial of service attacks in networked embedded systems," Proceedings of the 9th Workshop on Embedded Systems Security (WESS '14), Article No. 5, PP. 1–8, Oct. 2014,

[24] M.-K. Yoon and S. Chen, "An incremental deployable path address scheme," ELSEVIER Journal of Parallel and Distributed Computing, Vol. 72, no. 10, PP. 1215-1225, October 2012.

[25] Ming-Hour Yang, Jia-Ning Luo, M. Vijayalakshmi and S. Mercy Shalinie, "Hybrid Multilayer Network Traceback to the Real Sources of Attack Devices," IEEE Access, Vol. 8, PP. 201087–201097, Jan. 2020.

[26] E. Hilgenstieler, E. P. Duarte, G. Mansfield-Keeni and N. Shiratori, "Extensions to the source path isolation engine for precise and efficient log-based IP traceback," ELSEVIER Computers and Security, Vol. 29, no. 4, pp. 383-392, June 2010.

[27] S. Malliga and A. Tamilarasi, "A hybrid scheme using packet marking and logging for IP traceback," International Journal of Internet Protocol Technology, Vol. 5, no. 1-2, pp. 81–91, April 2010.

[28] C. Gong and K. Sarac, "A more practical approach for single-packet IP traceback using packet logging and marking," IEEE Transactions on Parallel and Distributed Systems, Vol. 19, no. 10, pp. 1310–1324, 2008.

[29] M. H. Yang, "Hybrid Single-Packet IP Traceback with Low Storage and High Accuracy," The Scientific World Journal, Vol. 2014, Article ID 239280, 2014.

[30] M. H. Yang, "Storage-Efficient 16-Bit Hybrid IP Traceback with Single Packet," The Scientific World Journal, Vol. 2014, Article ID 659894, 2014.

[31] S. Malliga, C. S. K. Selvi, and S. V. Kogilavani, "A low storage and traceback overhead system for IP traceback: A hybrid approach," Journal of Information Science and Engineering, 2014.

[32] S. Yu, W. Zhou, S. Guo and M. Guo, "A dynamical Deterministic Packet Marking scheme for DDoS traceback," IEEE Xplore, Global Communications Conference (GLOBECOM), December 2013.

[33] Y. Cui, L. Yan, S. Li, H. Xing, W. Pan, J. Zhu and X. Zheng, "SD-Anti-DDoS: Fast and efficient DDoS defense in software-defined networks," ELSEVIER Journal of Network and Computer Applications (JNCA), Vol. 68, PP. 65-79, June 2016.

[34] J. Francois and Olivier Festor, "Anomaly Traceback using Software Defined Networking," IEEE National Conference on Parallel Computing Technologies (PARCOMPTECH), PP. 203-208, 2015.

[35] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," Communications of the ACM, Vol. 13, no. 7, pp. 422–426, July 1970.

[36] A. Y. Nur and M. E. Tozal, "Single Packet AS Traceback against DoS Attacks," IEEE International Systems Conference (SysCon), PP. 1-8, May 2021.

[37] Merouane Mehdi, "An approach for detecting and preventing DDoS attacks in campus," Automatic Control and Computer Sciences, Vol. 51, Issue 1, PP. 13-23, Jan. 2017.

[38] D. Yan, Y. Wang, S. Su and F. Yang, "A Precise and Practical IP Traceback Technique based on packet marking and logging," Journal of Information Science and Engineering, Vol. 28, no. 3, pp. 453-470, May 2012.

[39] Sangita Roy and Ashok Singh Sairam, "Distributed star coloring of network for IP traceback," International Journal of Information Security, Vol. 17, Issue 3, PP. 315–326, March 2018.

[40] Gagandeep Kaur, Vikas Saxena and J. P. Gupta, "A Novel Multi Scale Approach for Detecting High Bandwidth Aggregates in Network Traffic," International Journal of Security and Its Applications, Vol.7, Issue 5, PP. 81-100, 2013.

[41] Nofel Yaseen, John Sonchack and Vincent Liu, "tpprof: A Network Traffic Pattern Profiler," 17th USENIX Symposium on Networked Systems Design and Implementation, PP. 1015-1030, Feb. 2020.

[42] Shahabeddin Geravand and Mahmood Ahmadi, "Bloom filter applications in network security: A state-of-the-art survey," Elsevier Computer Networks, Vol. 57, Issue 18, PP. 4047-4064, Dec. 2013.

[43] Network Simulator (ns-2). http://www.isi.edu/nsnam/ns/ (accessed September 1, 2021).

[44] Rocketfuel. http://research.cs.washington.edu/networking/rocketfuel/ (accessed September 1, 2021).

AUTHOR BIOGRAPHIES

Magdy M. Fadel received his Ph.D., M.Sc. and B.Sc. degrees in Computers and Control Engineering from Faculty of Engineering, Mansoura University in 2018, 2003 and 1995 respectively. He is serving as Chief Engineer in Faculty of Engineering, Mansoura University. His research interests are in the area of computer networks, network security and Data Security.

Arabic Title

HDSL: إطار عمل هجين لتتبع الحزم من النوع IP موزع، أحادي الحزمة ومنخفض التخزين

Abstract in Arabic:

تسهل العديد من المشكلات المتعلقة بتصميم بروتوكول IP مهمة مهاجمي رفض الخدمة الموزعة (DDoS). تقترح هذه الورقة إطار عمل جديد (HDSL) يتبع حزم بروتوكول IP وهو نظام هجين موزع، أحادي الحزمة ومنخفض التخزين، ويتكون من ثلاث آليات محسنة للدفاع ضد الهجوم من النوع DDoS. الآلية الأولى هي الترقيم الإلزامي للحزم (DPM) لتكوين معرف مسار فريد للتحقق من مصدر وصحة مسارات الشبكة. والثاني هو تسجيل معلومات الحزم منخفض المساحة لتسجيل معلومات الحزم الموجهة محليًا والتي يتم استخدامها لاحقًا لتحديد مصدر حتى حزمة مهاجمة واحدة. والثالث، يدفع مجاميع الحزم المهاجمة إلى مستوى واحد أو أكثر لأعلى للتخفيف من الازدحام الذي حدث عند الهدف أو بالقرب منه والذي قد يتسبب في إسقاط الحزم المشروعة. تم تطوير ثلاث خوارزميات لهذا الغرض. يستخدم نظام كشف التطفل (IDS) أيضًا لإدارة وحدات الدفاع في إطار العمل، وإدارة معلومات الشبكة. تظهر النتائج التجريبية أن أداء التتبع قد تحسن من عدة جوانب. أولاً، تم تقليل النسبة المئوية للحواف الزائفة التي يتم إرجاعها كنتيجة لمعرفات مسار الاصطدام المنخفض الدقيقة المقترحة. أيضًا، تم تقليل مساحة التسجيل المطلوبة إلى أكثر من ٧٠٪ مقارنة بالآليات الأخرى. أخيرًا، نسبة الحزم المشروعة التي تم إسقاطها بسبب الإزدحام الناتج عن مهاجمة مجاميع الحزم قد انخفضت بدرجة كبيرة بفضل تطبيق مبدأ دفع التصدي للأمام