Copyright ⓒ 2023 KSII

A Blockchain-enabled Multi-domain DDoS Collaborative Defense Mechanism

Huifen Feng [1*], Ying Liu [1,2], Xincheng Yan[2,3], Na Zhou[2,3], and Zhihong Jiang[3]
[1] National Engineering Research Center for Advanced Network Technologies, Beijing Jiaotong University
Beijing, 100044 China
[e-mail: huifenfeng@bjtu.edu.cn, yliu@bjtu.edu.cn]
[2] State Key Laboratory of Mobile Network and Mobile Multimedia Technology
Shenzhen, 518055, China
[3] ZTE Corporation
Nanjing, 210012, China
[e-mail: yan.xincheng@zte.com.cn, zhou.na@zte.com.cn, jiang.zhihong@zte.com.cn]
[*]Corresponding author: Huifen Feng

Received April 21, 2022; revised October 19, 2022; revised December 28, 2022; accepted March 1, 2023; published March 31, 2023

Abstract

Most of the existing Distributed Denial-of-Service mitigation schemes in Software-Defined Networking are only implemented in the network domain managed by a single controller. In fact, the zombies for attackers to launch large-scale DDoS attacks are actually not in the same network domain. Therefore, abnormal traffic of DDoS attack will affect multiple paths and network domains. A single defense method is difficult to deal with large-scale DDoS attacks. The cooperative defense of multiple domains becomes an important means to effectively solve cross-domain DDoS attacks. We propose an efficient multi-domain DDoS cooperative defense mechanism by integrating blockchain and SDN architecture. It includes attack traceability, inter-domain information sharing and attack mitigation. In order to reduce the length of the marking path and shorten the traceability time, we propose an AS-level packet traceability method called ASPM. We propose an information sharing method across multiple domains based on blockchain and smart contract. It effectively solves the impact of DDoS illegal traffic on multiple domains. According to the traceability results, we designed a DDoS attack mitigation method by replacing the ACL list with the IP address black/gray list. The experimental results show that our ASPM traceability method requires less data packets, high traceability precision and low overhead. And blockchain-based inter-domain sharing scheme has low cost, high scalability and high security. Attack mitigation measures can prevent illegal data flow in a timely and efficient manner.

Keywords: Autonomous System (AS), Blockchain, Smart Contract, IP traceback, Autonomous System Number (ASN), DDoS defense.

Parts of this paper on inter-domain information sharing have been appeared in ACM ICEA, December 28–29, 2021, Jinan, China. This paper includes the specific content and experimental results analysis of multi-domain DDoS attack traceback, inter-domain information sharing and attack mitigation.

http://doi.org/10.3837/tiis.2023.03.013 **ISSN : 1976-7277**

1. Introduction

The Internet of Everything a trend of network development in the future. The large number, variety, and limited security functions of IoT devices are vulnerable to hackers. According to a recent research report [1], the number of connected IoT smart devices will exceed 75 billion by 2025. And more than half of unsecured IoT devices will be vulnerable to serious attacks [2]. It is increasingly common for attackers to launch DDoS attacks on the network by using botnets created by Internet of Things devices. IoT devices have become a major source of growth in DDoS attacks. The numerically controlled separation model of SDN allows users and upper-layer applications to have more control over the network [3], and provides a new method for defending against DDoS attacks. However, it is found through investigation that most DDoS attack defense solutions based on SDN architecture are implemented under a single controller in the same network domain. [4]-[6]. And in the actual network environment, attackers can use these IoT devices with security risks to launch DDoS attacks on other IoT devices in the IoT from different locations or sources. Therefore, the Zombie hosts used by attackers to create botnets are usually in different network domains. Multiple network domains or paths will be affected by abnormal traffic of DDoS attacks [7] [8]. Defending against DDoS attacks in multiple network domains requires consideration of the following aspects:

- ***Flexibility / scalability:*** Most of the DDoS attack defense schemes based on SDN architecture are centralized solutions with low flexibility and poor scalability [5] [9]. And vulnerable to a single point of failure.
- ***Inter-domain sharing capability:*** DDoS attack traffic usually affects multiple network domains. The lack of data sharing among multiple network domains makes it difficult to defend against cross-domain DDoS attacks.
- ***Cost:*** Attackers form botnets to launch a large-scale DDoS attack, usually sending out a lot of meaningless packet traffic. To effectively reduce the huge overhead of forwarding packets across multiple network domains.
- ***Security:*** To ensure the security of data packets when they are forwarded between multiple domains. And trusted with multiple cooperating network domains.

Currently, defense against DDoS attacks across multiple network domains can be divided into two categories. One is to respond immediately when a DDoS attack is detected (e.g., packet loss, port blocking) [10]-[12]. This method can quickly respond to attacks in the network. But it does not pose any threat to attackers. The other is to trace the attack source first, and to prevent the DDoS attack from the source according to the traceability results [13]-[15]. This method can find the real attack source. It can quickly prevent the continuous occurrence of attacks from the source of the attack, and conduct network behavior investigation and forensics. This paper aims at how to effectively track the real source of the attacker and take appropriate mitigation measures when DDoS is detected in the network, which is the key to DDoS defense. The DDoS attack detection section is beyond the scope of this article. The main contributions of this paper can be summarized as follows:

- We propose an AS-oriented multi-domain DDoS attack traceability method. We achieve traceability across multiple network domains by tracing the AS paths between attackers and victims.
- We propose a multi-domain DDoS information sharing solution based on blockchain and smart contracts. IP address black/gray list sharing between ASs through blockchain. It can prevent IP addresses with low reputation from attacking the network again, and effectively reduce the threats to each domain.

- We propose a method to use IP address grey list and blacklist instead of ACL list to respond to the results of DDoS attack detection and traceability. The controller dynamically modifies or adds flow entries to effectively and quickly mitigate DDoS attacks.

The rest of this paper is organized as follows. In Section 2, we summarize the related work on DDoS attack defense. In Section 3, we describe in detail the system scheme. In Section 4, we introduced the implementation platform and steps of our scheme. Section 5, evaluates our scheme. Finally, Section 6, we conclude this paper.

2. Related Work

In this section, we first introduce blockchain technologies. We summarize the existing DDoS defense solutions from the aspects of single-domain and multi-domain.

2.1 Blockchain Technology

Blockchain technology [16] is a combination of a series of existing technologies (e.g., Cryptography, Merkle Tree, Consensus mechanism). Blockchain's decentralization capabilities, data security protection, data sharing functions and distributed characteristics provide a good infrastructure for DDoS collaborative defense system [17]. As shown in **Fig. 1**. It allows attack data to be shared in a fully automated and distributed manner, enabling trustless networks. It solves data sharing among multiple domains and trust issues between domains that cooperate with each other [8].

Smart contracts [18] are used to describe an "agreement executed by the parties". It formulates the agreement between the parties to a contract into a set of regular lines of code. When a transaction that meets the conditions occurs, the smart contract will automatically execute the corresponding contract terms. Smart contract is a reusable, modular, automatically executed scripts running on the blockchain. Currently, smart contracts have been used in DDoS attack mitigation schemes that enable secure data sharing, access, and transmission [19] [20].

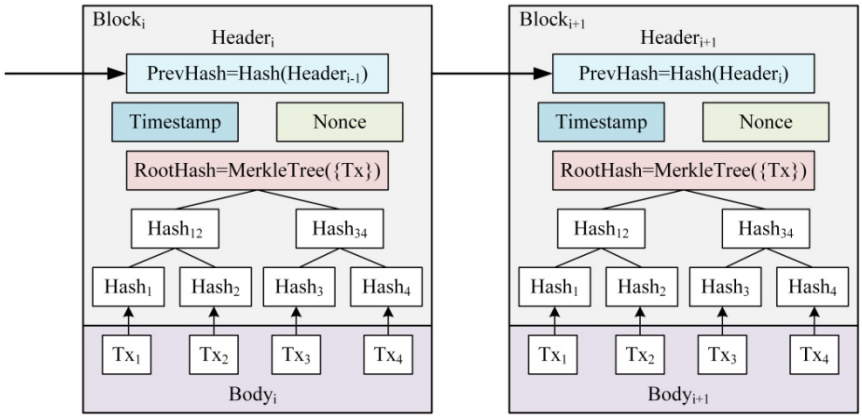

Fig. 1. The basic structure of the blockchain.

Blockchain is considered an effective solution to improve SDN network security as it can provide a distributed, decentralized and distrusted chain system structure. As a naturally distributed architecture, blockchain does not require dedicated registries and other distributed collaborative mechanisms or protocols among multiple domains. Blockchain and smart

contract technologies can be used to share DDoS attack information in a distributed manner across multiple domains. It enables each AS domain to use the attack information provided by other domains to defend against DDoS attacks in advance. Each peer can interact with others even if they do not trust each other. It breaks the information barrier between each network domain and ensures secure and efficient data transmission. Additionally, the blockchain and smart contract solution is decentralized and does not require a central authority to maintain the collaborative system. Therefore, there will be no single point of failure when running on this system. Consequently, this paper uses it to enhance a distributed and secure DDoS defense scheme across multiple domains.

2.2 Single-domain DDoS Defense Solution

In [11], they propose a learning-driven detection mitigation mechanism. They divided the list of malicious IoTs and developed separate attack mitigation strategies for each. However, this method is only used for UDP flooding attacks, and only one dataset is used to test the detection model. In [21], they proposed a DDoS attack traceability mechanism based on Probabilistic Packet Marking (PPM). This scheme needs to mark all routers on the attack path to mark all passing data packets. The computational and communication overhead is large. Liu et al. [22] proposes a new method for dynamic Probabilistic Packet Marking. It is superior to the traditional method of marking with fixed probability. In [23], They proposed a DDoS attack backtracking method based on Deterministic Packet Marking (DPM). It uses the IP address information of the ingress routers on the path to mark each packet that passes through ingress routers. Compared to PPM, DPM generates fewer backtracking data and processes faster. To be compatible with different network environments, [24] and [25] proposed a flexible Deterministic Packet Marking mechanism. In [26], they proposed a multi-layered DDoS defense framework called MLDMF to mitigate against DDoS attacks. It integrates the programmability of SDN, the rapid response capability of edge computing, the state perception capability of fog computing, and the powerful computing capability of cloud computing. But the security of the edge server itself may be ignored. Guo et al. [27] proposed a blockchain-based lightweight DDoS attack traceability defense scheme. The scheme installs a digest program that generates transmission packets on all routers in the LAN. And store it on the blockchain. When a DDoS attack occurs, trace the source based on the router's summary information and block malicious traffic.

2.3 Multi-domain DDoS Defense Solution

Multiple network domains are affected by abnormal traffic from DDoS attacks. Therefore, a single-domain DDoS attack defense is insufficient to deal with large-scale DDoS attacks. In [28], they designed a Controller-to-Controller communication protocol for different AS. This protocol allows SDN controllers in the local domain to communicate with controllers communicate with other controllers in adjacent domains. In addition, it informs neighboring domains of ongoing attacks and transmits DDoS attack information for effective early warning. In [8], they proposed an inter-domain and intra-domain DDoS mitigation solution using smart contracts and blockchain technology. The scheme realizes data sharing of DDoS attacks across multiple network domains through smart contracts. But it doesn't show how DDoS mitigation works. In [10], they proposed a simple message queue mechanism (RSMQ). When one of the controllers detects a DDoS attack, it will be shared and published by Pub/Sub to every controller in the network to block the identified DDoS attack. By using RSMQ method in SDN environment, the controller can prevent malicious traffic in the network and notify neighboring nodes in time. In [29], they propose to construct an AS-level network for IP

backtracking by extending the BGP UPDATE attribute. On the contrary, [30] believes that identifying some key points along the forwarding path of the packet is sufficient for the purpose of backtracking. Information can be passed between ASs that are not necessarily involved in the overlay network. Gao et al. [31] proposed a probabilistic labeling method called ASEM. The method utilizes the AS_PATH feature to ensure that all packets will be marked by a router on the path before reaching the destination. However, tag all packets will increase the burden on the router, resulting in the difficulty of traceability and time. In [32], they proposed a Fast Autonomous System Traceback scheme (FAST). The tag field of this scheme only allows the preceding five ASs to tag packets. For attacks involving more than five ASs, FAST can only record the data of the last five ASs. And in the process of path reconstruction, it is assumed that the victim already knows the AS topology of the network. Unlike previous packet schemes, [33] and [13] use IP protocol record routing option fields to build a path graph from source to destination. It is mainly to gradually add the path information of packet forwarding to the record routing option field. The advantage of this approach is that it is an option in the IP header that can be implemented without a new deployment. And it does not need to store enough packets at the victim for traceability.

This paper summarizes the feasibility and shortcomings of the above schemes. On this basis, a DDoS collaborative defense solution across multiple network domains is proposed by integrating blockchain and SDN architecture. Our solution is more scalable and flexible than the current multi-domain DDoS defense solution. This solution implements the multi-domain security policy among each AS and adds additional security mechanisms.

3. Multi-domain DDoS Collaborative Defense Mechanism

3.1 Overview

In this section, we describe a system framework for a blockchain-enabled multi-domain DDoS collaborative defense mechanism. More specifically, we describe in detail how the scheme performs cross-domain DDoS attack traceability, mitigation, and inter-domain information sharing. Our system structure is shown in **Fig. 2**.

Our proposed scheme for multi-domain DDoS cooperative defense based on blockchain-SDN consists of three parts. Firstly, in order to accurately locate the attack source and restore the attack path, we propose an AS-oriented packet marking algorithm (ASPM). We redefine the IP header format to gradually add the AS forwarding path between attacker and victim to *AS_Path*. We only mark limited packets with the same destination address at the ingress router of each SDN domain. When the victim domain detects a DDoS attack, the server in this domain can obtain complete path information by parsing the data packets. Then, the controller will deliver the filtering mechanism to the corresponding ingress router according to the traceability result and the black/grey ACL list. Therefore, the filtering of attack traffic is implemented at the attack source. In order to prevent malicious hosts with low reputation from launching attacks on other hosts in other network domains again. We propose a blockchain-based cross-domain information sharing scheme. In this way, effective DDoS attack defense can be implemented on the attack path and attack source, and network security can be guaranteed to the greatest extent.

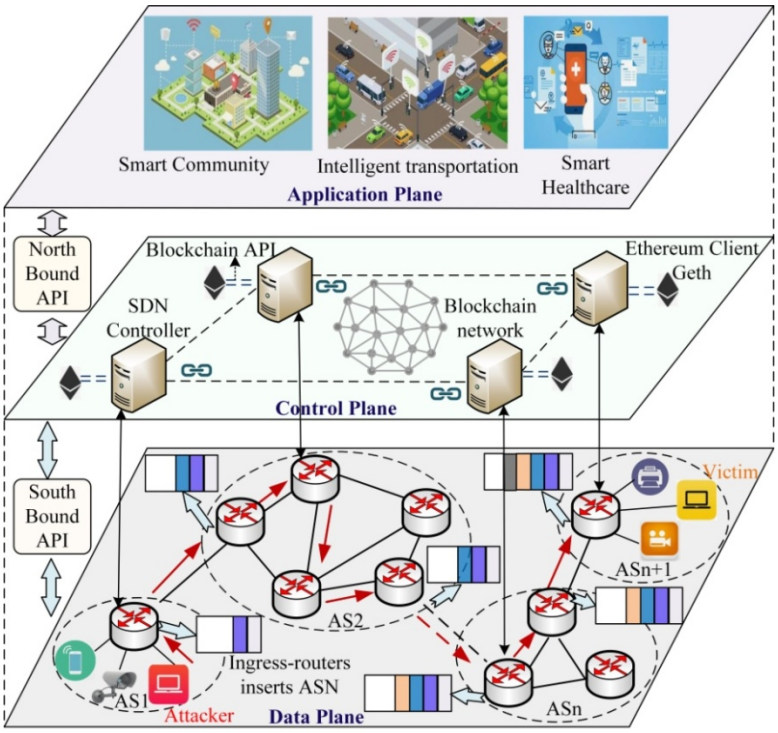

Fig. 2. Overview of the high-level architecture integrating SDN and blockchain.

3.2 DDoS Traceability Method Based On ASPM

After detecting a DDoS attack, how to find the true location of the attacker without relying on the source IP address. Taking targeted defense measures at the root is the key to combating DDoS attacks. We use ASN to replace the traditional IP address marking scheme, and propose a multi-domain DDoS traceability method for AS. Among them, ASN [34] is managed and controlled by Internet Assigned Numbers Authority (IANA). It is the unique global number of the Internet identification AS. ASN is represented by two different formats: 2-byte and 4-byte [35]. With the dramatic increase in the number of ASs in the Internet, the IETF [36] proposes that all ASNs should be considered 4-byte. We define ASN as an unsigned 32-bit integer. The specific process is as follows.

3.2.1 Design of IP Header Tag Field

The scheme based on IP traceability technology requires rewriting the packet header in different ways to mark the path information. ASPM uses the Reserved Flag (RF) field, the Type of Service (TOS) field and the Options filed in the packet header as the tag space to rewrites the IP packet header to store the tag information (See the shaded section in **Fig. 3**). In the IP header field, the TOS field indicates the type of service the packet wants to obtain, and is rarely used at present. The Identification, Fragment Offset and Flag fields are used to segment the packet information. Studies have shown that the proportion of the number of packets that need to be fragmented in the network is only about 0.25% [46]. Therefore, in general these three fields are idle and are used the most in IP traceability. The use of a variable length Option field incurs additional overhead, so this field is rarely used.

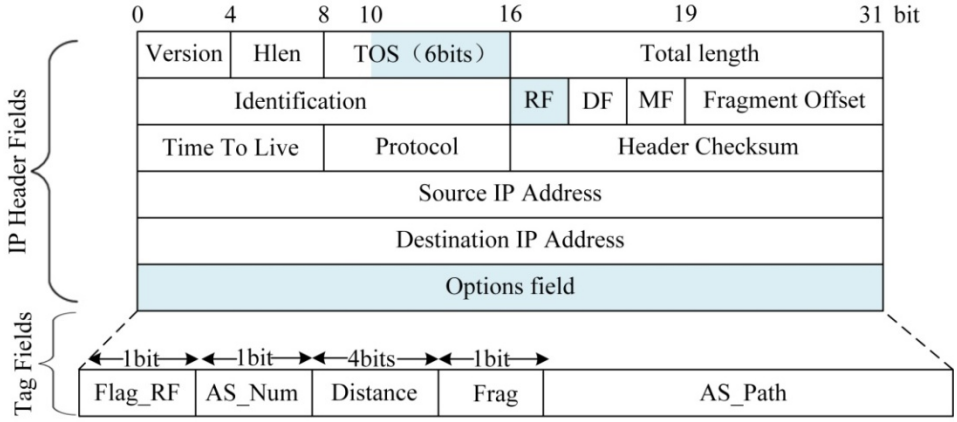

Fig. 3. The IP header fields (shaded) used in our proposed.

In our proposed ASPM scheme, in order to satisfy multiple ASN data storages, we use fields that are rarely used in IP header as the label space, and insert label information into the data packets. It includes the high bit of 1 bit in the Flag field, the low bit of 6 bits in the TOS field, and the variable length Option field. Different tag spaces correspond to different tag information. The above tag space is divided into five parts: ***Flag_RF*** (1bit), ***AS_Num*** (1bit), ***Distance*** (4bits), ***Frag*** (1bit) and ***AS_Path*** (variable). As shown in **Fig. 3**. The definitions are as follows.

- ***ASN_i (16bits)*:** It represents the ASN fragmentation information of the autonomous system. In order to shorten the traceability time, reduce the number of data packets required for path reconstruction, and reduce the marking overhead of the router. We shard the 4-byte (32bits) ASN. Each 2-byte (16bits) is a fragment as marking information. The more fragments an ASN is divided into, the more packets the victim domain requires to reconstruct the attack path.
- ***Flag_RF (1bit)*:** It is used to prompt the downstream ingress border router whether the current data packet is marked. Its values are 0 and 1. The value of *Flag_RF* is set by the packet passing through the first ingress router. When reconstructing the path, the server can filter out untagged packets according to this field to reduce the false positive rate.
- ***AS_Num (1bit)*:** It is used to indicate which fragment of the ASN inserted into the marked packet is stored in the path. Its values are 0 and 1. If *AS_Num* = 0, it means that the first 16 bits of the ASN (e.g., ASN_{i-1}) of the AS domain are inserted into the marked data packet for storage. If *AS_Num* = 1, it means inserting the last 16 bits of the ASN (e.g., ASN_{i-2}) of the AS domain into the marked data packet for storage.
- ***Frag (1bit)*:** It is used to indicate which fragment of the ASN should be inserted into the packet. Its values are 0 and 1.
- ***Distance (4bit)*:** It represents the distance value of the packet from the first marked router to the victim's AS. Its value range is 0-15. [37] shows that more than 99% of the autonomous systems have less than 8 ASs before the packet reaches the destination. For some huge and rare attacks may cross more than 8 AS. In our proposed mark backtracking technique, the marked data packets can store up to 19 ASN fragment information. Therefore, there is no problem in this paper that cannot be backtracked. And the value of this field corresponds to the ASN fragmentation to perform path reconstruction. When the victim and the attacker are in the same domain, set the *Distance* to 0. And the distance between adjacent AS domains to 1. We can achieve traceability at both intra-domain and inter-domain levels.

- ***AS_Path*:** It is used to store ASN shards on the path. Its value is determined by the AS the packet traverses. In our scheme, each time a marked packet passes through an AS, one of its ASN fragments is added to *AS_Path*. Then, traverse the AS from the attacker to the victim.

3.2.2 ASPM Marking Method

In our scheme, ASN is a 32-bit unsigned integer as the tag information of each AS. Then, the ASN marking information is inserted into the IP packet header by the ingress router of each AS. (Since SDN switches have L2-L4 layer network capabilities. Border routers, OpenFlow / SDN edge switches are considered the same type of device in this paper). Specifically, when a data packet is forwarded between different ASs, the ingress router of each AS inserts the ASN fragment of the domain into the corresponding IP data packet header. However, other routers in the AS domain skip the insertion mark information and normally forward packets. In our scheme, only the ingress router of each AS participates in the marking work, and other routers do not participate in the marking.

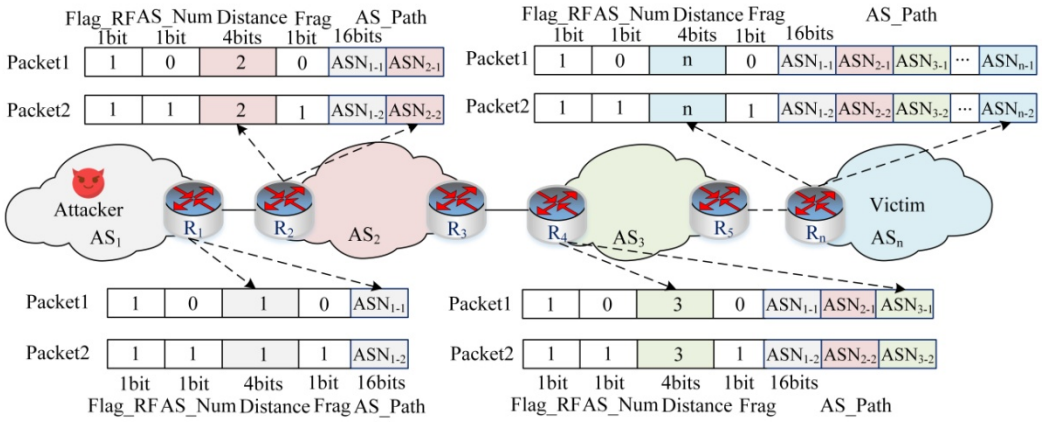

Fig. 4. ASPM marking process across multiple domains.

Fig. 4 shows the ASPM marking process across multiple domains. ASPM's marking scheme starts with the ingress router R_1, which is the closest to the attacker. When the first ingress router receives the packet, it inserts the two fragments of ASN_1 into the two packets respectively. And stored in the *AS_Path* field. And the value of the *Flag_RF* field of the data packet is set to 1. When tracing the source, victims can filter and mark packets according to the value of this field to reduce the false positive rate. And set the value of the *AS_Num* field of the data packet with the first 16 bits of information marked $ASN_{1\text{-}1}$ to 0. The value of the *AS_Num* field of the data packet with the last 16 bits of information marked $ASN_{1\text{-}2}$ is set to 1. Then the packet is forwarded to the next-hop router according to the routing strategy.

When the ingress router R_2 in AS_2 receives the packet, it first checks the value of the *Flag_RF* field of the packet. Based on the value of this field, determine whether the packet is marked. Then, the *Frag* field selects ASN_2 fragments according to the value of the *AS_Num* field of the data packet, and inserts them into the corresponding *AS_Path*. If *AS_Num*=0, the ingress router R_2 chooses to insert the first 16 bits of the ASN_2 value ($ASN_{2\text{-}1}$) into the packet. If *AS_Num*=1, insert the 16bits fragment after the ASN_2 value ($ASN_{2\text{-}2}$) into the packet. Otherwise, the packets are forwarded normally, and the marking information on the packets remains unchanged. Since the ASN of AS_2 has been inserted at the ingress router R_2, the router R_3 skips the insertion and directly forwards the data packet to the next-hop router.

Then, repeat the above marking method until the packet reaches the victim domain. The marking algorithm of the ingress router is as in **Algorithm 1**. Specifically, each time the ingress router of each AS performs a marking action, the value of the *Distance* field of the marked data packet is incremented by 1. When the marked packet is forwarded in the same domain, the value of the *Distance* field does not change. Finally, after the attacker collects data packets containing two complete ASN fragments, the AS path information of the attacker can be determined according to the value of the *Distance* field corresponding to the ASN fragments. In order to further improve the accuracy of attack source tracing, we use fixed time intervals to mark the packets sent from the attacker with ASPM.

Algorithm 1: The marking algorithm of the ingress router

```
Making procedure at ingress router R：
for each incoming packet w
check the Flag_RF filed of each packet w
if Flag_RF=1 then
check the AS_Num filed of each packet w
   if AS_Num=0 then
   The Flag filed writes ASN_{i_1} into w.AS_Path
   else
      if AS_Num=1 then
      The Flag filed writes ASN_{i_2} into w.AS_Path
     end if
  Distance=Distance_i+1
 end if
else
forward packet w
```

3.2.3 Path Reconstruction Procedure

When the DDoS attack detection algorithm in the victim domain detects the attack, the victim starts inter-domain path reconstruction from its own AS domain. As shown in **Fig. 5**. The specific process is as follows.

Step1: Firstly, the server in the victim domain filters and extracts packets with flag information according to the value of the *Flag_RF* field. The marked data packet contains the ASN value of the AS domain traversed from the attacker to the victim and the distance to the victim AS.

Step2: According to the *Distance* field and the *AS_Num* field in the marked data packet, the fragmentation information of the ASN stored in the *AS_Path* is extracted into the reconstructed path table. The reconstructed path table is maintained on the server within the victim domain. Because the packet is marked once, the value of the *Distance* field in the packet is updated. That is, when the data packet leaves the attacker's AS domain and reaches the ingress router of the adjacent domain, the value of the *Distance* field will be changed to the distance between the current AS domain and the attacker's AS domain. The value of the corresponding ASN fragment is the path information between the attacker and the victim.

Step3: When the victim collects two complete packets with different marks, extract the packets in the order of *AS_Num*=0, 1. Arrange the shards corresponding to the ASN in order according to the *Distance*=1,2, …, i, and reconstructs the ASN_i to obtain the attack path: ASN_0, ASN_1, ASN_2, …, ASN_i.

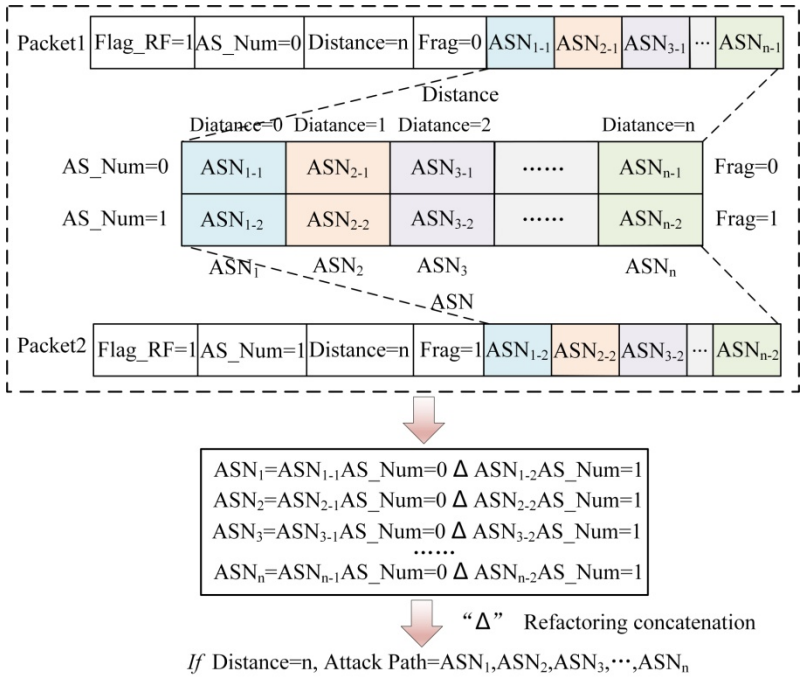

Fig. 5. Path reconstruction process.

3.3 Blockchain-based Inter-Domain Sharing Method

In multi-domain DDoS attacks, there is the transmission of malicious traffic between interconnected AS domains. Consequently, only DDoS attack mitigation in the victim's AS cannot completely prevent the transmit of malicious traffic. And a less reputable attacker could launch an attack on other domains again. In order to effectively address the impact of DDoS attacks across multi-domain on the whole network, cooperation between ASs is required. In this way, the threat caused by malicious traffic to the network domain is minimized.

We utilize emerging blockchain and smart contract technologies to facilitate collaboration between different AS domains. The attack information is shared among ASs in a decentralized manner. As shown in **Fig. 6**. Each AS domain runs only one SDN controller that receives and reports information about the attacker. We use the SDN controllers as the nodes of the blockchain to form a blockchain network. We connect the SDN controller of each AS domain to an Ethereum *Geth* client [38] through the API interface of the blockchain. It enables various domains to communicate and cooperate with each other, receive and share DDoS attack information.

When the victim domain detects a DDoS attack, the ASPM-based attack traceability method is enabled to obtain the AS where the attacker is located. The SDN controller in the AS domain sends illegal data flow information to the *Geth* client through the blockchain API interface. *Geth* clients report illegal data flow information to smart contracts. Then, we classify the illegal data flow in the smart contract. We calculate the credibility of the source IP address of the illegal data flow, and add the IP address to the corresponding IP address grey list or blacklist. By designing decentralized, information sharing program code deployed on the blockchain, the code and state of the smart contract is stored in the block as a kind of transaction data. Finally, smart contracts are proliferated across the blockchain network in a P2P manner after consensus and processing.

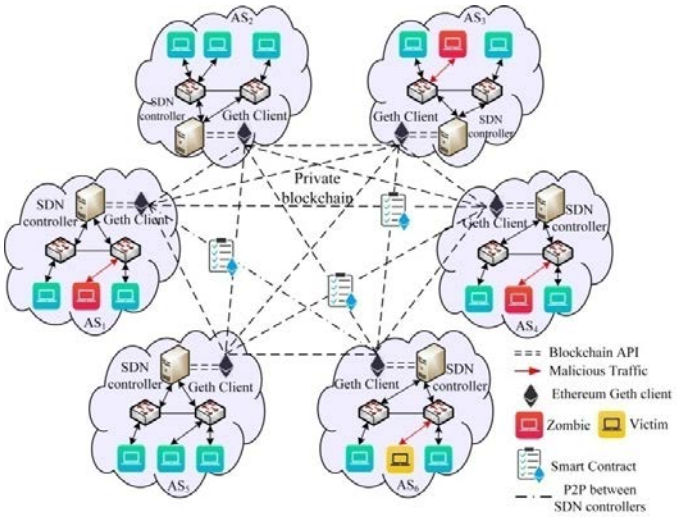

Fig. 6. Multi-domain sharing architecture based on SDN-blockchain.

The smart contract maintains this black/grey list. Specifically, it is first determined whether the source IP address is in the existing blacklist or grey list according to the reported illegal data flow information. If it is, the SDN controller in the AS domain delivers corresponding mitigation strategy to the edge switch, thereby achieving the goal of DDoS attack defense. Conversely, the credibility of the source IP address of the data flow is calculated from the reported illegal data flow information. If the credibility is below the threshold, it means that the source IP address is strongly correlated with the DDoS relationship in the SDN. Then, add the source IP address to the blacklist and update the blacklist. Otherwise, add the source IP address to the grey list, and update the grey list. As shown in **Algorithm 2**.

Algorithm 2: Classification algorithm of source IP address

Input: The number of suspicious source IP addresses x_i;
the total number of source IP addresses n;
Output: Blacklist and grey list
The arithmetic average of the source IP address $\leftarrow \mu$
The standard deviation of the source IP address $\leftarrow \sigma$
Calculate μ by $\sum_{i=1}^{n} x_i/n \leftarrow \mu$
Calculate σ by $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i-\mu)^2} \leftarrow \sigma$
The credibility of suspicious IP address $\leftarrow \varphi$
The threshold of suspicious IP address $\leftarrow T_{thr}$
Calculate by $x_j/\sum_{i=1}^{n} x_i \leftarrow \varphi$
Calculate T_{thr} by $\mu + \sigma \leftarrow T_{thr}$
if IP credibility $\varphi < T_{thr}$ **then**
update the blacklist;
else
if IP credibility $\varphi > T_{thr}$ **then**
update the grey list;
end
end

Then, we deploy the compiled smart contract code on the blockchain. As shown in **Listing 1**. All participating nodes on the blockchain generate new blocks after passing consensus verification. Smart contracts broadcast events across the blockchain. All participants in each network domain receive this event and then add suspicious IP addresses to the blacklist and gray list maintained by that network domain. Then, the SDN controller in the AS domain delivers corresponding mitigation policies to edge switches. Therefore, it implements DDoS defense at attack source and attack path, and maximizes the security of the whole network environment.

Listing 1. Smart contract structures and some core code.

```
pragma solidity ^0.4.24;
contract SuspiciousIp {
    struct  IpAttributes {···
    }
    string state;
    mapping (uint => IpAttributes) IpList;
    string []  public IpAddrList;
    uint [] public IpThreshold;//IpNum
    mapping (uint => IpAttributes) blackList;
    uint [] public blackThreshold;//blackNum
    mapping (uint => IpAttributes) grayList;
    uint [] public grayThreshold;//grayNum
function IpClassify(string ipAddress,uint timestamp,uint interval,uint Credibility,uint Threshold)···
// ============================check if the ip is already in the stack=========================···
//======================if else :IP address classification according to the Credibility&Threshold=========================···
//================================just use the Threshold count IpNum ======================================···
// ======================count ip ; count ip in blackList/grayList/IpList=================================···
function setIP (address _address,uint _node,string _ipAddress,bool _state,uint256 _timestamp,uint _interval) public {
}
function getIPs() view public returns(address[]){···
}
function  getIP(address _address) view public returns(uint,string,bool,uint256){···
}
function countIPs() view public returns (uint){···
function removeIPs(address _address,uint _node,string _ipAddress,bool _state,uint256 _timestamp,uint _interval) public{···
function checkState(address _address) view public returns(string,bool,uint256){···
```

Our scheme realizes secure distributed DDoS attack data information sharing among multiple ASs. All authorized collaborators on the blockchain can add, share and access blacklists and gray lists. The method breaks the information barriers between AS domains and promotes mutual cooperation among AS domains. And other ASs can quickly verify the authenticity of attacks and respond in time by analyzing statistical information in advance. Experimental results show that it provides flexibility, scalability, low cost, and security for inter-domain DDoS attack defense.

3.4 DDoS Attack Mitigation Method

After completing the reconstruction of the DDoS attack path and tracing the real attack source, the next step is how to effectively defend against DDoS. Based on the characteristics of SDN, we comprehensively use the Access Control List and black and gray list mechanisms to achieve DDoS attack mitigation. ACL is a command list of router or switch interface, used to control the data packets in and out of the port. After configuring the ACL, it can restrict or effectively block the access of some specified source addresses. We use blacklist and grey list instead of traditional ACL list. Then, the SDN controller of each AS domain maps the defined ACL rules to ACL flow entries in the switch. And deliver the ACL flow entry to the network edge switch. This allows or prohibits communication with the victim, thus achieving the goal of mitigating DDoS attacks.

Specifically, after finding the attacker AS domain according to the attack traceability method, the SDN controller in the domain locates the edge switch connected to the attacking host. Then it judges whether the source IP address of the packets entering the edge switch is in

the blacklist or the gray list. If it is determined that the source IP address of the attacker is in the maintained blacklist IP address list. Then, the SDN controller sends the flow entry to the edge switch and discards the data packets from the source IP address. If it is determined that the source IP address of the attacker is in the gray listed IP address list. Then, the SDN controller sends a flow entry to limit the rate of the data flow. Otherwise, if it is determined that the source IP address of the attacker is not the blacklist and the gray list, it can be judged as normal traffic, normal communication is established, and data packets are forwarded.

4. Implementation

In this section, we will describe in detail the simulation environment for experimental simulation in this paper. We use *Mininet* to build a network topology to simulate the real experimental environment. And using Behavior model version 2 (BMv2) switches in the network effectively simulates a router. It can run independently or in *Mininet*. BMv2 [39] is the second version of the P4 reference software model. It is specifically designed for developing and debugging the P4 data plane and control plane software written for it. P4 [40] supports the creation of custom headers, parsing of existing or new headers, and custom match-action tables.

In this paper, we implement ASPM tag traceability and DDoS attack mitigation based on the programmability of P4. The SDN controller runs alone on the server. We installed and implemented the Ethereum blockchain on Ubuntu 18.04 LTS platform. The Ethereum *Geth* client (1.8.20) is connected to the SDN controller through the Blockchain API. We use the Remix IDE [41] for smart contract deployment, testing and implementation. By combining blockchain and SDN reasonably, we have realized secure DDoS attack data sharing and defense among multiple ASs. **Table 1** shows the main simulation parameters used for system simulation.

Table 1. Applications in each class

Simulation Parameters	Description
CPU	Intel® Core™ i5-6600 CPU @ 3.30GHz × 2
Simulation platform	Mininet / Ethereum
Simulation System	Ubuntu 18.04 LTS
Ethereum Client	Geth client
Smart Contract	Remix IDE
Type of Switch	BMv2 Switches

5. Evaluation

In this section, we respectively evaluate the performance of the DDoS attack defense methods proposed in this paper across multiple network domains from different aspects.

5.1 Performance evaluation of ASPM

In order to test the effectiveness of our scheme in the actual environment, we use the IPv4 Routed /24 Topology Dataset [42] in CAIDA containing more than 20 million path trajectory information to conduct simulation experiments. The dataset consists of 16352 nodes and 39346 links. We get the RouteViews prefix for the AS mapping dataset for IPv4 and IPv6 from RouteViews Prefix to AS mappings Dataset [43]. We then generate the topology in AS units by mapping the reported IP address in the Traceroute dataset to its corresponding AS. This

dataset contains 154,357 path information from a single source. And consists of 32301 different ASs. Among them, the maximum and minimum hop counts of the dataset we use are 16 and 1. We randomly count the distance probability distribution between two ASs. As shown in **Fig. 7**. Through data collation and statistical analysis, 99% of the packets usually pass through less than 8 AS domains before reaching the destination domain. Therefore, our proposed ASPM traceability scheme does not have untraceable.

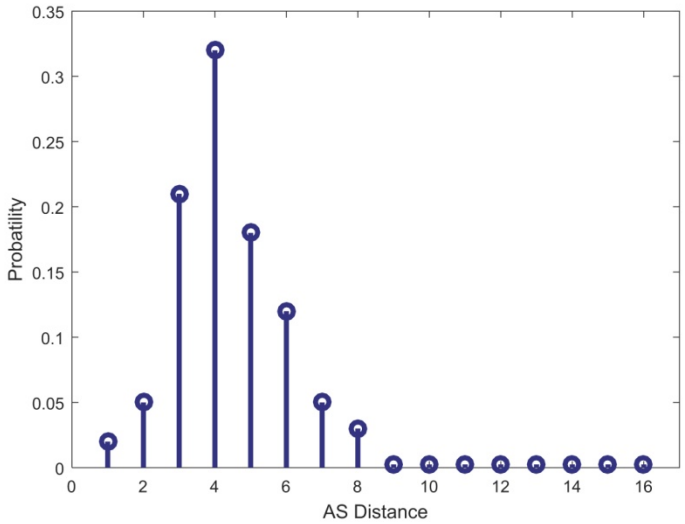

Fig. 7. Multi-domain sharing architecture based on SDN-blockchain.

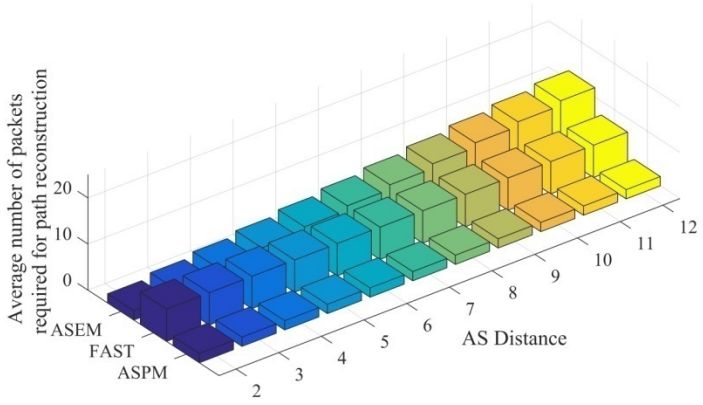

Fig. 8. Comparison of average packets required for path reconstruction.

The purpose of attack traceability is to reconstruct the attack path and locate the source of the attack. Therefore, we use the number of packets needed to rebuild the attack path as one of the measures. **Fig. 8** shows the relationship between path length and the average number of packets required during path reconstruction. We compare the ASPM traceability method with the classic packet-marked AS traceability methods FAST [32] and ASEM [31]. As can be seen from **Fig. 8**, as the length of the AS path increases, the number of data packets required in the path reconstruction process of the ASEM scheme increases linearly. The optimal marking probability for ASEM is determined according to the AS path length. The FAST-marking

scheme requires an average of 7 packets in the entire reconstruction attack path. In our scheme, the *AS_Path* field in the packet with tag information received by the victim domain contains the complete ASN fragment information. The victim domain only needs to collect two complete marked packets to reconstruct the AS path from the attacker to the victim. Therefore, our scheme requires fewer packets to reconstruct the path. It reduces source tracing time and router overhead, and improves source tracing speed. However, as the path length increases, our scheme requires more marker space to record path information. Therefore, our method consumes more bandwidth.

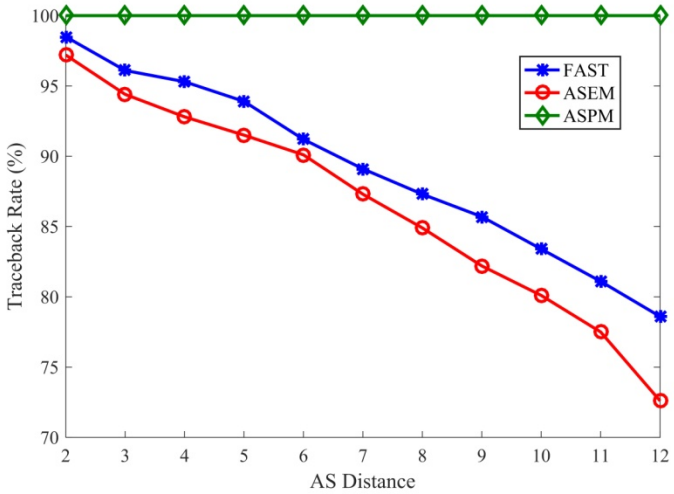

Fig. 9. Comparison of schemes with different traceback rates.

Fig. 9 shows the probability that the victim successfully finds the attacker's AS domain. In our scheme, the victim extracts the marked packet when restoring the attacker's AS domain. Because the *AS_Path* field of the marked data packet stores the fragmentation of the complete ASN. Therefore, as long as the victim extracts the tag information carried by the two packets containing the complete ASN fragment. According to the distance field and the *AS_Num* field, the ASN shards are reassembled in sequence, and the AS domain where the attacker is located can be successfully restored. We assume that the router is secure, and the marking information will not be tampered with during the forwarding of marked packets. Therefore, the probability of successful restoration is 1, regardless of the number of packets with tag information received by the victim domain. Therefore, our scheme has a high backtracking success rate.

5.2 Evaluation of Inter-Domain Information Sharing Methods

We verify and evaluate the cross-domain information sharing scheme based on blockchain in terms of throughput and gas consumption. First, we write and compile smart contracts on Remix. Then, use *Ganache* to deploy the smart contract on a private Ethereum blockchain for functional testing. *Ganache* [44] is an Ethereum emulator for quickly testing smart contract functionality. More detailed contract information is in the official Ethereum test chain. After the contract is deployed, use the ABI and the contract address to invoke the smart contract. **Fig. 10** shows the transaction details when sharing data across multiple network domains.

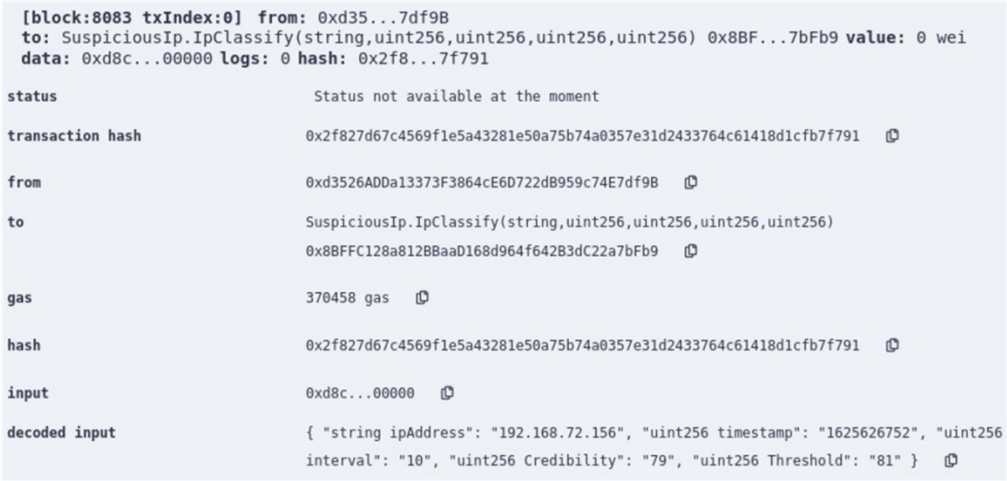

Fig. 10. Information related to transactions on the blockchain.

Each transaction corresponds to a transaction hash value, which is used to identify the transaction. The From and To fields represent the address of the originating account and the receiving account for the transaction, respectively. The originating account in the multi-domain DDoS information sharing method proposed in this paper is generally the client account in the target domain. It's worth noting that each transaction is not free in non-test environments. The resource consumption in Ethereum is called Gas Used. This paper uses this metric as a cost assessment.

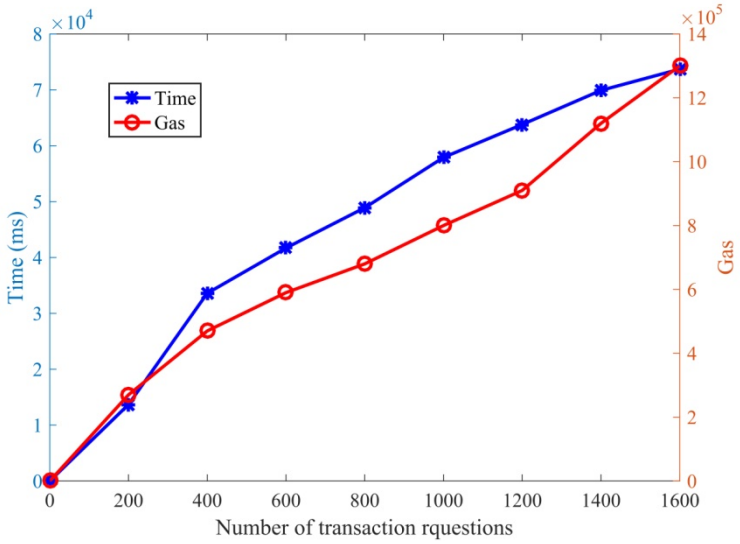

Fig. 11. The correlation of Time and Gas consumption with the Number of Transaction requests.

Fig. 11 shows the corresponding relationship between the processing time and Gas versus respectively with the number of transaction requests. The smallest unit of Gas is *wei*, $1ETH=10^{18}wei=10^{9}Gwei$. Time is expressed in *ms*. In our scheme, a certain amount of Gas is consumed to ensure that the transaction of DDoS attack information is transmitted between network domains. The gas consumption increases approximately linearly with time when the number of transactions is increasing.

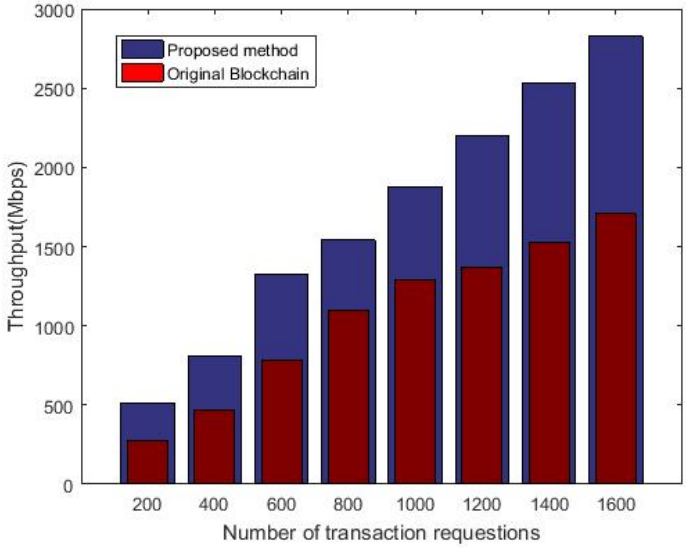

Fig. 12. The correlation between Throughput and the Number of Transaction requests.

Fig. 12 shows the relationship between throughput and the number of transactions in the scheme designed in this paper. Transaction throughput is the number of transactions made per second. In this paper, the throughput of transactions is taken as the vertical axis, and the number of requests of transactions is taken as the horizontal axis. **Fig. 12** shows that as the number of requests increases, the ability of our designed scheme to handle transactions has improved, so the throughput has also increased significantly. We propose a customized new architecture for different SDN network domains that support blockchain. We use the concept of Ethereum's smart contracts to facilitate the collaboration among SDN-based domains to mitigate DDoS attacks. And remove the intermediary for secure communication. Blocks can be added to the blockchain without a public process, which significantly reduces the additional overhead and energy consumption of the original blockchain with POW. Improves the certain throughput compared to the original blockchain scheme.

5.3 Evaluation of DDoS Attack Mitigation Methods

We design simulation validation of multi-domain DDoS attack mitigation methods using the DDoS attack dataset of CAIDA [45]. It can be known from **Fig. 7** that most of the DDoS attack packets can reach the destination through 4 ASs from the attacker's network domain to the victim's network domain. Therefore, this paper simulates four SDN-based AS domains for DDoS attack mitigation validation. We take the change in the overall network rate as the overall time overhead of the DDoS attack mitigation system.

Fig. 13 shows the traffic changes in the network from the start of the attack to the implementation of mitigation. At the 40th second, the attacking host initiates a DDoS attack on the victim host. The traffic of illegal data flow in the network has increased dramatically. After the attack lasts for 25 seconds, the SDN controller in the AS domain sends a flow table to the edge switch in the domain to implement mitigation strategies for DDoS attacks. After the DDoS attack mitigation strategy was implemented, the attack rate in the network dropped rapidly and quickly returned to normal levels. Experiments show that this scheme can effectively mitigate the impact of DDoS attacks.

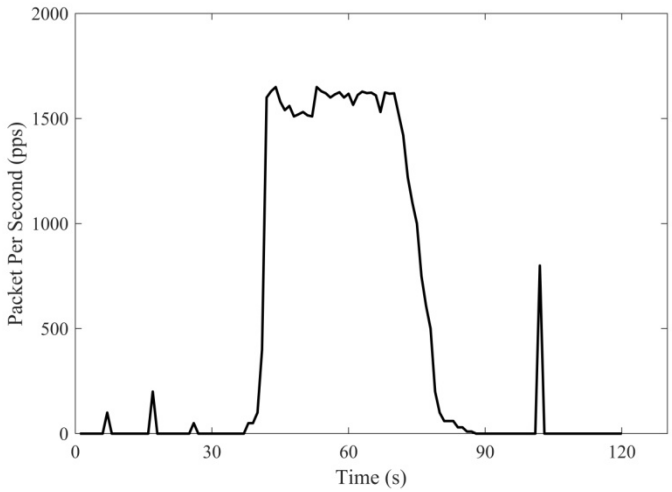

Fig. 13. DDoS attack mitigation effects.

Table 2. Comparison of our scheme and other related proposed solutions

	N. Ravi et al. [11]	Guo et al. [27]	El Houda et al. [8]	Hameed et al. [28]	Durresi et al. [32]	Ours
Multi-domain collaboration	No	No	Yes	Yes	Yes	Yes
Blockchain	No	Yes	Yes	No	No	Yes
Suitable for DDoS data sharing	No	No	Yes	Yes	No	Yes
Scalability	Poor	Poor	Good	Poor	Average	Good
Bandwidth overhead	Low	High	Average	Average	Low	High
Security	Low	Average	High	Low	Low	High

As depicted in **Table 2**, we qualitatively compare the characteristics of different DDoS defense schemes among N. Ravi et al. [11], Guo et al. [27], El Houda et al. [8], Hameed et al. [28], Durresi et al. [32] and ours. First, since our solution runs on the platform of the Ethereum blockchain, it is decentralized and there is no single point of failure problem. Therefore, it has good flexibility and scalability. And it requires neither a third party nor a central system to maintain collaboration between multiple network domains. It can share DDoS attack information among network domains in a distributed manner to ensure secure and efficient data transmission. Therefore, it has high security and reliability. However, as the path length increases, our DDoS attack tag traceability scheme requires more tag space to record the path information. Thus, our approach consumes more bandwidth. Generally, we provide a new approach to address DDoS defense schemes across multiple network domains and add additional security mechanisms.

6. Conclusion

In this paper, we integrate blockchain with SDN architecture to provide new ideas for DDoS attack mitigation methods across multiple domains. In order to deal with the traceability problem of AS-level DDoS attacks, we reload the IP packet header to record the ASN fragments traversing from the attacker to the victim. Our method only needs two marked packets containing complete ASN fragments to reconstruct the attack path. Our method

requires a small number of reconstructed data packets, a short time for backtracking, and a high traceability accuracy. Abnormal traffic is forwarded across multiple network domains and will affect the entire network. The cross-domain DDoS attack information sharing method based on blockchain can effectively reduce the threat to each AS domain and improve the flexibility and security of the system. Experimental results show that our multi-domain DDoS cooperative defense mechanism can effectively mitigate DDoS attacks.

For the future work, we intend to combine intra-domain DDoS mitigation scheme with inter-domain traceability scheme, so as to improve the security of the whole network within and between domains. The problem of protecting private information across multiple network domains is also considered. And how to deploy and verify the effectiveness of the proposed DDoS defense scheme in the actual networks.

Acknowledgement

This work was supported by the Fundamental Research Funds for the Central Universities under Grant (2021YJS007), and by the ZTE industry-university research cooperation fund project "Research on network identity trusted communication technology architecture" (W21L00940), and by the State Key Laboratory of Mobile Network and Mobile Multimedia Technology.

References

[1] The IoT Rundown for 2020: Stats Risks and Solutions, 2020, [online] Available: https://securitytoday.com/Articles/2020/01/13/The-IoT-Rundown-for-2020.aspx?Page=2

[2] More Than Half of IoT Devices Vulnerable to Severe Attacks, 2020, [online] Available: https://threatpost.com/half-iot-devices-vulnerable-severe-attacks/153609/

[3] Sutrala, Anil & Obaidat, Mohammad & Saha, Sourav & Das, Ashok Kumar & Alazab, Mamoun & Park, Youngho, "Authenticated Key Agreement Scheme With User Anonymity and Untraceability for 5G-Enabled Softwarized Industrial Cyber-Physical Systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 3, pp. 2316-2330, 2022. Article (CrossRef Link)

[4] Rochak Swami, Mayank Dave, and Virender Ranga, "Software-defined Networking-based DDoS Defense Mechanisms," *ACM Computing Surveys*, vol. 52, pp. 1-36, 2019. Article (CrossRef Link)

[5] Mishra, A., Gupta, N. & Gupta, B.B., "Defense mechanisms against DDoS attack based on entropy in SDN-cloud using POX controller," *Telecommun Syst*, 77, 47–62, 2021. Article (CrossRef Link)

[6] M. Jia, Y. J. Shu, Q. Guo, et al., "DDoS attack detection method for space-based network based on SDN architecture," *ZTE Communications*, vol. 18, no. 4, pp. 18–25, Dec. 2020. Article (CrossRef Link)

[7] L. Zhu, X. Tang, M. Shen, X. Du and M. Guizani, "Privacy-Preserving DDoS Attack Detection Using Cross-Domain Traffic in Software Defined Networks," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 3, pp. 628-643, March 2018. Article (CrossRef Link)

[8] Z. A. El Houda, A. Hafid and L. Khoukhi, "Co-IoT: A Collaborative DDoS Mitigation Scheme in IoT Environment Based on Blockchain Using SDN," in *Proc. of* 2019 *IEEE Global Communications Conference (GLOBECOM)*, pp. 1-6, 2019. Article (CrossRef Link)

[9] J. A. Pérez-Díaz, I. A. Valdovinos, K. -K. R. Choo and D. Zhu, "A Flexible SDN-Based Architecture for Identifying and Mitigating Low-Rate DDoS Attacks Using Machine Learning," *IEEE Access*, vol. 8, pp. 155859-155872, 2020. Article (CrossRef Link)

[10] Tayfour, O.E., Marsono, M.N., "Collaborative detection and mitigation of DDoS in software-defined networks," *J Super comput*, 77, 13166–13190, 2021. Article (CrossRef Link)

[11] N. Ravi and S. M. Shalinie, "Learning-Driven Detection and Mitigation of DDoS Attack in IoT via SDN-Cloud Architecture," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3559-3570, April 2020. Article (CrossRef Link)
[12] Narmeen Zakaria Bawany, Jawwad A. Shamsi, "SEAL: SDN based secure and agile framework for protecting smart city applications from DDoS attacks," *Journal of Network and Computer Applications*, Vol. 145, p. 102381, 2019. Article (CrossRef Link)
[13] Abdullah YasinNur, Mehmet EnginTozal, "Record route IP traceback: Combating DoS attacks and the variants," *Computers & Security*, Vol. 72, pp. 13-25, 2018. Article (CrossRef Link)
[14] Praveena, V., Karthik, S. & Jeon, G., "Correction to: Hybrid Approach for IP Traceback Analysis in Wireless Networks," *Wireless Pers Commun*, 113, 691, 2020. Article (CrossRef Link)
[15] Suresh, S., Sankar Ram, N. Feasible, "DDoS attack source traceback scheme by deterministic multiple packet marking mechanism," *J Super comput*, 76, 4232–4246, 2020. Article (CrossRef Link)
[16] Saha, S, Chattaraj, D, Bera, B, Kumar Das, A., "Consortium blockchain-enabled access control mechanism in edge computing based generic Internet of Things environment," *Trans Emerging Tel Tech*., 32, e3995, 2021. Article (CrossRef Link)
[17] D. Chattaraj, S. Saha, B. Bera and A. K. Das, "On the Design of Blockchain-Based Access Control Scheme for Software Defined Networks," in *Proc. of IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 237-242, 2020. Article (CrossRef Link)
[18] A. Abuhashim and C. C. Tan, "Smart Contract Designs on Blockchain Applications," in *Proc. of 2020 IEEE Symposium on Computers and Communications (ISCC)*, Rennes, France, pp. 1-4, 2020. Article (CrossRef Link)
[19] M. Essaid, D. Kim, S. H. Maeng, S. Park and H. T. Ju, "A Collaborative DDoS Mitigation Solution Based on Ethereum Smart Contract and RNN-LSTM," in *Proc. of 2019 20th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, Matsue, Japan, pp. 1-6, 2019. Article (CrossRef Link)
[20] K. Giotis, M. Apostolaki and V. Maglaris, "A reputation-based collaborative schema for the mitigation of distributed attacks in SDN domains," in *Proc. of NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*, Istanbul, Turkey, pp. 495-501, 2016. Article (CrossRef Link)
[21] M. T. Goodrich, "Probabilistic Packet Marking for Large-Scale IP Traceback," *IEEE/ACM Transactions on Networking*, vol. 16, no. 1, pp. 15-24, Feb. 2008. Article (CrossRef Link)
[22] Jenshiuh Liu, Zhi-Jian Lee, and Yeh-Ching Chung, "Dynamic probabilistic packet marking for efficient IP traceback," *Comput. Netw*., 51, 3, 866–882, February, 2007. Article (CrossRef Link)
[23] Andrey Belenky, Nirwan Ansari, "On deterministic packet marking," *Computer Networks*, Vol. 51, no. 10, pp. 2677-2700, 2007. Article (CrossRef Link)
[24] 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. Article (CrossRef Link)
[25] G. D. Barokar and V. S. Mahalle, "Identification of the Real Source of DDOS Attack by FDPM in IP Traceback System," in *Proc. of 2014 European Modelling Symposium*, pp. 392-396, 2014. Article (CrossRef Link)
[26] Q. Yan, W. Huang, X. Luo, Q. Gong and F. R. Yu, "A Multi-Level DDoS Mitigation Framework for the Industrial Internet of Things," *IEEE Communications Magazine*, vol. 56, no. 2, pp. 30-36, Feb. 2018.Article (CrossRef Link)
[27] W. Guo, J. Xu, Y. Pei, L. Yin and C. Jiang, "LDBT: A Lightweight DDoS Attack Tracing Scheme Based on Blockchain," in *Proc. of 2021 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1-6, 2021. Article (CrossRef Link)
[28] Hameed S, Ahmed Khan H., "SDN Based Collaborative Scheme for Mitigation of DDoS Attacks," *Future Internet*, 10(3), 23, 2018. Article (CrossRef Link)
[29] A. Castelucio, A. Ziviani and R. M. Salles, "An AS-level overlay network for IP traceback," *IEEE Network*, vol. 23, no. 1, pp. 36-41, January-February 2009. Article (CrossRef Link)

[30] V. Aghaei-Foroushani and A. N. Zincir-Heywood, "Autonomous system based flow marking scheme for IP-Traceback," in *Proc. of NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*, pp. 121-128, 2016. Article (CrossRef Link)
[31] Zhiqiang Gao, Nirwan Ansari, "A practical and robust inter-domain marking scheme for IP traceback," *Computer Networks*, Vol. 51, no. 3, pp. 732-750, 2007. Article (CrossRef Link)
[32] ArjanDurresi, VamsiParuchuri, Leonard Barolli, "Fast autonomous system traceback," *Journal of Network and Computer Applications*, Vol. 32, no. 2, pp. 448-454, 2009. Article (CrossRef Link)
[33] M. Alenezi and M. J. Reed, "Selective record route DoS traceback," in *Proc. of 2013 International Conference on Risks and Security of Internet and Systems (CRiSIS)*, pp. 1-7, 2013. Article (CrossRef Link)
[34] Internet Assigned Numbers Authority (IANA), Autonomous System Number Allocations. [Online]. Available: https://www.iana.org/assignments/as-numbers/as-numbers.xhtml
[35] Autonomous System Numbers. [Online]. Available: https://www.arin.net/resources/guide/asn/
[36] Internet Engineering Task Force (IETF), 2012. [online] Available: https://datatracker.ietf.org/doc//html/rfc6793
[37] Arjmandpanah-Kalat, M, Abbasinezhad-Mood, D, Mahrooghi, H-R, Aliabadi, S., "Design and performance analysis of an efficient single flow IP traceback technique in the AS level," *Int J Commun Syst*., 33, e4382, 2020. Article (CrossRef Link)
[38] "Geth", [Online]. Available: https://geth.ethereum.org/downloads/
[39] "BMv2", [Online]. Available: https://github.com/p4lang/behavioral-model
[40] "P4", [Online]. Available: https://p4.org/p4-spec/p4-14/v1.0.5/tex/p4.pdf
[41] "Remix", [Online]. Available: http://remix.ethereum.org/
[42] IPv4 Routed /24 AS Links Dataset, [Online]. Available: https://www.caida.org/catalog/datasets/ipv4_routed_topology_aslinks_dataset/
[43] Routeviews Prefix to AS mappings Dataset (pfx2as) for IPv4 and IPv6, 2022-03-12, [Online]. Available: https://www.caida.org/catalog/datasets/routeviews-prefix2as/
[44] "Ganache", [Online]. Available: https://github.com/trufflesuite/ganache-cli
[45] The CAIDA "DDoS Attack 2007" Dataset, [Online]. Available: https://www.caida.org/catalog/datasets/ddos-20070804_dataset
[46] S. Savage, D. Wetherall, A. Karlin and T. Anderson, "Network support for IP traceback," *IEEE/ACM Transactions on Networking*, vol. 9, no. 3, pp. 226-237, June 2001. Article (CrossRef Link)

Huifen Feng received the M.S. degree in Electronic and Electrical Engineering from Henan Normal University, Henan, China, in 2019. Where she is currently pursuing the Ph.D. degree with the School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China. Her research interests include network security, future network architecture, privacy and blockchain.

Ying Liu is a Professor at the School of Electronic and Information Engineering, Beijing Jiaotong University, China. She received her BS degree from Beijing Jiaotong University in 2000. She received her MS and Ph D degrees from Beijing Jiaotong University in 2003 and 2012, respectively. Her research interests include network security, privacy, new protocols and architectures in networking.

Xincheng Yan is Chief System Architecture Expert of ZTE corporation, deputy director of the future network research center of State Key Laboratory of Mobile Networks and Mobile Multimedia Technology. He is a professorate senior engineer, has 20 years of experience in the telecom industry, has more than 40 patents, and has presided over the National Science and Technology Major Project of China in 5G security. He has won several scientific and technological awards, and won the titles of "333" third-level talent and high-level talent in Jiangsu Province.

Na Zhou received the PhD. degree from Nanjing University of Aeronautics and Astronautics University, China in 2004. She is currently the senior technical research expert in ZTE Corporation. She has been awarded Shenzhen Scientific and Technological Award.

Zhihong Jiang received the M.S degree from Posts and Telecommunications University, China in 2003. He is currently the senior technical research expert in ZTE Corporation.