*Computer Systems Science & Engineering*
DOI: 10.32604/csse.2023.032862
***Article***

Tech Science Press

# A Secure Framework for Blockchain Transactions Protection

**Wafaa N. Al-Sharu[1,*], Majdi K. Qabalin[2], Muawya Naser[2] and Omar A. Saraerh[1]**

[1]Department of Electrical Engineering, Faculty of Engineering, The Hashemite University, Zarqa, 13133, Jordan
[2]Department of Computer Science, King Hussein School of Computing Sciences, Princess Sumaya University for Technology, Amman, 11941, Jordan
*Corresponding Author: Wafaa N. Al-Sharu. Email: wafaa.al-sharo3@hu.edu.jo

Received: 31 May 2022; Accepted: 05 July 2022

**Abstract:** One of the most extensively used technologies for improving the security of IoT devices is blockchain technology. It is a new technology that can be utilized to boost the security. It is a decentralized peer-to-peer network with no central authority. Multiple nodes on the network mine or verify the data recorded on the Blockchain. It is a distributed ledger that may be used to keep track of transactions between several parties. No one can tamper with the data on the blockchain since it is unchangeable. Because the blocks are connected by hashes, the transaction data is safe. It is managed by a system that is based on the consensus of network users rather than a central authority. The immutability and tamper-proof nature of blockchain security is based on asymmetric cryptography and hashing. Furthermore, Blockchain has an immutable and tamper-proof smart contract, which is a logic that enforces the Blockchain's laws. There is a conflict between the privacy protection needs of cyber-security threat intelligent (CTI) sharing and the necessity to establish a comprehensive attack chain during blockchain transactions. This paper presents a blockchain-based data sharing paradigm that protects the privacy of CTI sharing parties while also preventing unlawful sharing and ensuring the benefit of legitimate sharing parties. It builds a full attack chain using encrypted threat intelligence and exploits the blockchain's backtracking capacity to finish the decryption of the threat source in the attack chain. Smart contracts are also used to send automatic early warning replies to possible attack targets. Simulation tests are used to verify the feasibility and efficacy of the suggested model.

**Keywords:** Manuscript; preparation; typeset; format

## 1 Introduction

In recent years, network technology has been renovated day by day, and at the same time, it has brought more complex network attack methods or means, such as zero-day exploit [1], advanced persistent threat (APT), social engineering, etc. [2]. Due to the asymmetry of information, security defenders are at a natural disadvantage in the "speed battle" of complex system security attack and defense as shown in Fig. 1, according to Verizon's 2018 report, attackers can attack and defend complex systems within minutes (87%

of enterprises' network systems were compromised, while 68% of enterprises would only find out that their network systems were compromised several months later) [3,4]. In the face of complex attack forms and serious attack consequences, relying on the technical strength of individuals or a single organization can only obtain partial attack information [5,6], cannot build a complete attack chain [7,8], and cannot accurately and effectively prevent attackers. Network security threat intelligence sharing and utilization, as a technical method of "exchanging space for time", can timely use the efficient threat intelligence generated in other networks to improve the response capability of the defender and shorten the response time, thereby forming a mechanism to alleviate the asymmetric situation of attack and defense [9–11].

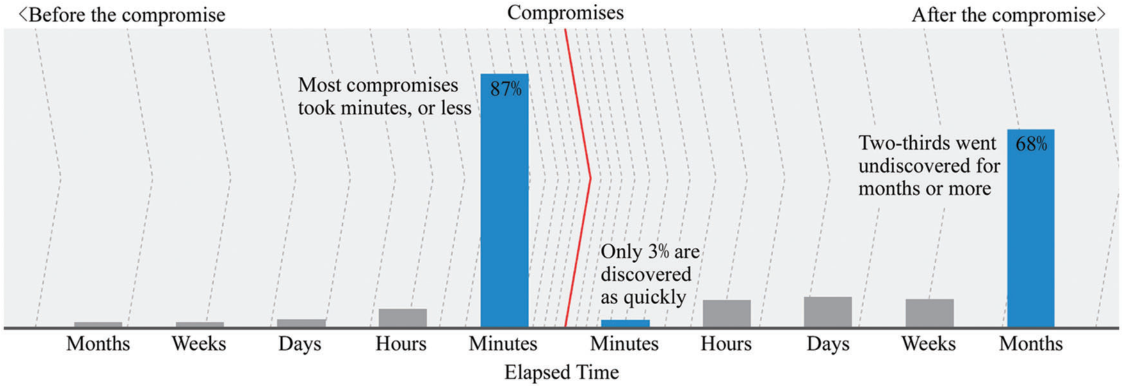


**Figure 1:** Evaluation of data breaching [11]

Information sharing may lead to the leakage of private information, and the sharing of cybersecurity threat intelligence is no exception [12]. In the real society, there have been cases where the leakage of private information has resulted in loss of business economy or reputation [13], which in turn affects the enthusiasm of enterprises to participate in cybersecurity threat intelligence sharing. Aiming at the privacy protection issue in cybersecurity threat intelligence sharing, literature [14–17] progressed from the aspects of protecting intelligence sharing organization identity information, protecting intelligence using the membership information and improving the privacy leakage in Structured Threaten Formation Expression (STIX) sharing standard. However, strict privacy protection also hinders the inference construction of the complete attack chain. For example, a server with IP address 12.1.2.3 of a bank was attacked by malicious code *M* and became a command and control (C2) server. In order to protect the private information, the bank released generalized threat intelligence: Malicious code *M* attacks the bank server, compromised server assumes C2 server function, then analysts cannot use the IP address 12.1.2.3 for reasoning analysis of the attack chain. That is, it is impossible to build a complete attack chain, so it is necessary to propose a method that can not only meet the needs of privacy protection, but also use threat intelligence to analyze and build a network security threat intelligence sharing model for a complete attack chain.

Aiming at the contradiction between privacy protection and threat intelligence utilization, this paper proposes a blockchain-based network security threat intelligence sharing model.

The key contributions are as follows:

- It deploys the account anonymity of the blockchain to protect the identity of the threat intelligence sharing party and the user.
- It uses the encrypted threat intelligence to build a complete attack chain, and complete the decryption of the threat source in the attack chain with the backtracking capability of the blockchain.
- It also deploys smart contracts to automatically issue early warning responses to potential attack targets.

## 2 Related Work

Researchers have carried out a lot of research on the framework or model of cybersecurity threat intelligence sharing, privacy protection in cybersecurity threat intelligence sharing, and the use of blockchain in information sharing, which provides the basis and reference for the work of this paper.

In terms of cybersecurity threat intelligence sharing, reference [18] discussed the necessity of cybersecurity information sharing and provided guidelines for types of shared information, defined community cybersecurity threat alert levels, and discussed the effects of different alert levels on information sharing. In this paper, a collaborative information sharing framework is proposed, and then the security, trustworthiness, privacy and other issues that may be encountered in the sharing process and future research directions are expounded. Reference [19] analyzed the historical background of information sharing with the help of Microsoft's experience in infrastructure security management, and expounded the classification of information sharing in terms of models, methods and mechanisms. The above research work has fully affirmed the necessity of network security threat intelligence sharing, and also put forward the demand for privacy protection, but failed to provide effective privacy protection methods or measures.

In terms of the combination of blockchain technology and information sharing, reference [20] applied the blockchain to the sharing and exchange of medical files, and realized the sharing of data while protecting the privacy of patients. Reference [21] applied the blockchain to in-vehicle edge computing and network data sharing, and achieved certain results. Reference [22] proposed a blockchain-based iShare framework, in which the members participating in the iShare framework can only share the scheme or overview of network security protection, and used the game theory to analyze the possible malicious behavior in the framework. Some progress has been made in the research of blockchain technology in the information sharing in other industries, but these research results cannot be directly applied to the sharing of cybersecurity threat intelligence. The iShare framework can only share the network security protection scheme, and does not involve the sharing of threat intelligence information [23].

In terms of privacy protection in threat intelligence sharing, reference [24] used an aggregated blind signature (based on BBS + signature scheme) mechanism in order to protect the identity information of organizations sharing threat intelligence, and proposed a registration, sharing, and demonstration method. In order to prevent the private information from being leaked to untrusted participants or hackers in the process of threat intelligence sharing, reference [25] proposed a network security threat intelligence sharing and utilization framework based on the homomorphic encryption. Reference [26] analyzed the possible privacy information leakage problem in STIX network security threat intelligence sharing standard, and tried to solve the privacy information leakage problem by using an improved data sharing protocol. The above research work has played a role in privacy protection from a unilateral dimension, but cannot protect the privacy information of the intelligence sharing party, the intelligence user and the intelligence-related parties at the same time [27–32].

In view of the limitations of existing research work, this paper constructs a network security threat intelligence sharing model based on blockchain. This model utilizes the account anonymity of the blockchain and the unified one-way encryption function to fully protect the privacy information of the intelligence sharing party, the user and the intelligence involved other parties. At the same time, the encrypted threat intelligence can be correlated and analyzed to construct a complete attack chain, improve the efficiency, and ability of security protection.

## 3 System Model

As shown in Tab. 1, the features or functions of blockchain such as decentralization, account anonymity, openness, autonomy, immutability, and smart contract mechanism can meet the privacy protection in cybersecurity threat intelligence sharing, according to the contribution value for reward, threat intelligence traceability, automatic early warning response and other requirements, among which:

1) The anonymity of the blockchain is determined by the Bitcoin address generation process. The Bitcoin address is generated by a series of encoding algorithms and hashing algorithms on the public key of the elliptic curve.
2) The traceability of the blockchain is determined by the blockchain construction process, $Block(N) = Hash(tp(N), Merkle(N), Block(N-1), nounce)$, where $tp(N)$ represents the Timestamp, $Merkle(N) = Hash(Tx(N))$ means the Merkle root containing the existing transaction, and *nounce* means the random number.
3) Smart contracts can trigger transactions when the contract conditions are met, that is:

$$is_{execute} = \begin{cases} True, \ x \in conditions \ or \\ \quad x = conditions; \\ \quad False, \ otherwise \end{cases} \tag{1}$$

In view of this, this paper proposes a blockchain-based network security threat intelligence sharing model.

**Table 1:** Features and requirements

| Features | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| Blockchain features | Decentralized technology and account anonymity | Openness and autonomy, distributed ledger | Tamper-free | Smart contract |
| Cyber security threat intelligence | Privacy preservation | Rewarding | Traceability | Automating early warning and response |

### 3.1 Definition

Before expounding the model and algorithm in this paper, the relevant definitions are given first.

**Definition 1.** One Meta-Class Network Security Threat Intelligence (OneCti). It mainly includes network security threat indicator (indicator of compromise, IoC) or threat object type intelligence, such as email, IP address, domain name, malicious code, organization, domain name owner, attacker, etc. It is described using a quadruple *tp*, *type*, *value*, *label*, where *tp* represents the timestamp, *type* represents the element type $(type \in \{tp, type, value, label\})$, *value* represents the element value, and *label* represents the label tag; such as: ⟨2019-05-16T10:00:00, IP, 12.6.5.3, C2⟩ indicates that the IP address 12.6.5.3 was detected at 2019-05-16T10:00:00 at C2 server.

**Definition 2.** Binary Class Network Security Threat Intelligence (TwoCti). It mainly includes network security event type intelligence, which is described using the 7-tuple: $< tp, type_1, value_1, rel, type_2, value_2, desc >$. It represents the relationship between two elements $(rel \in \{connect, inject, scan, \ldots\}$, *desc* represents the descriptive information related to the intelligence. For example: < 2019-05-16T10:00:00, ip, 12.6.5.3, connect, ip, 13.5.6.6, connect server > means that when 2019-05-16T10:00:00, the IP address is 12.6.5.3 and it is connected to the server whose IP address is 13.5.6.6.

**Definition 3.** Cybersecurity Intelligence Sharing Transaction (STrans). It refers to the process in which organizations share cybersecurity threat intelligence to the threat intelligence center, and the threat intelligence center verifies the threat intelligence, gives a certain *ticket* after evaluation, and returns the intelligence sharing certificate (*ticket*, using the six-tuple $\langle tp, O_{\mathrm{acc}}, C_{\mathrm{acc}}, reward, SENc(C_{pub_k}, Cti), ticket\rangle$ description, where *tp* represents the timestamp, $O_{\mathrm{acc}}$, $C_{\mathrm{acc}}$ and $SENc(C_{pub_k}, Cti)$ indicates the threat intelligence encrypted with the public key $C_{pub_k}$ of the threat intelligence center.

**Definition 4.** Network security threat intelligence map. It refers to the directed graph composed of the element values of network security threat intelligence using one-way encryption, described by $G = \langle V, E, L, R \rangle$. Among them, $V$ represents the one-way encrypted ciphertext of IP, domain name, mailbox and other element values. If there is a binary network security threat intelligence between $u, v \in V$, forming a directed edge from $u$ to $v$, then $(u, v) \in E$; $L$ is the set of labels of nodes $v \in V$; $R$ is the set of relations between nodes.

**Definition 5.** Network Security Intelligence Analysis Transaction (*ATrans*). It refers to the organization's request for threat intelligence analysis to the threat intelligence center. After the intelligence center correlates the threat intelligence provided by the organization with the intelligence in the database, it returns the analysis result or threat disposal suggestion (*result*) and charges a certain amount of intelligence usage fee (*uf*) process, using the six-tuple: $\langle tp,\ O_{\mathrm{acc}},\ C_{\mathrm{acc}},\ uf,\ SEnc\left(C_{\mathrm{pub_k}}, Cti\right),\ SEnc\left(C_{\mathrm{pub_k}}, result\right)\rangle$ description, where $tp,\ O_{\mathrm{acc}},\ C_{\mathrm{acc}},\ uf,\ SEnc\left(C_{\mathrm{pub_k}}, Cti\right)$ have the same meaning as *STrans*. The meaning is the same as in, $SEnc\left(O_{\mathrm{pub_k}}, result\right) >$ represents the result of intelligence analysis encrypted with the public key $O_{\mathrm{pub_k}}$ of the organization.

### *3.2 Model Description*

As shown in Fig. 2, the blockchain-based network security threat intelligence sharing model is represented by the octet: $\langle Org,\ Center,\ BlockNet,\ CtiDB,\ Cti,\ Trans,\ SC,\ Operation\rangle$, where:

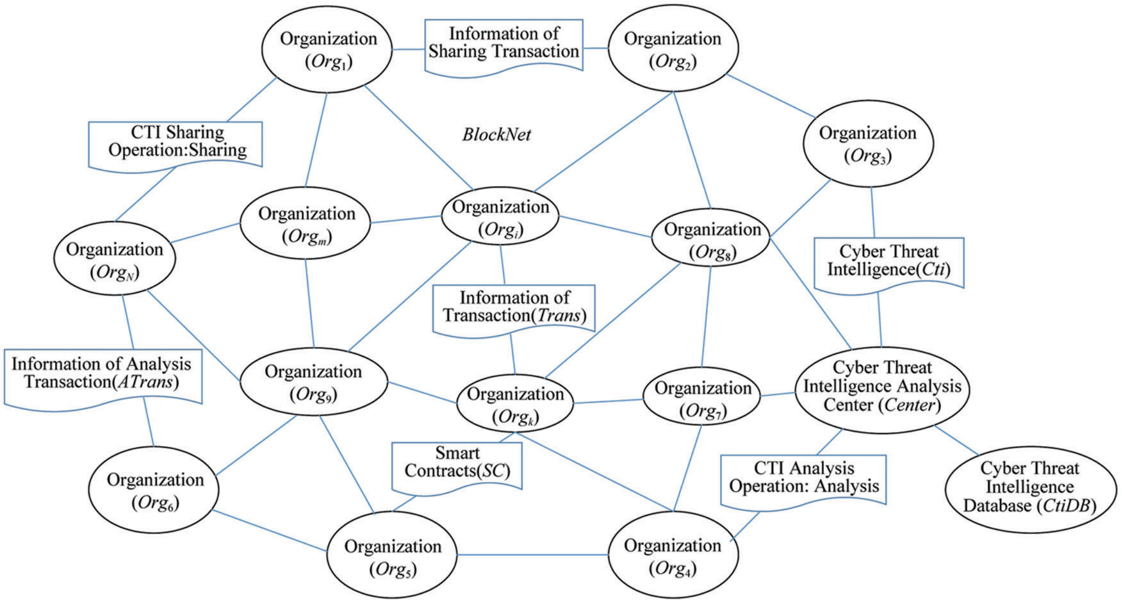


**Figure 2:** Blockchain model

1) *Org* represents an organization that can share and use cybersecurity threat intelligence. There are $N$ organizations $Org_i(1 \le i \le N)$ in the model. Each organization $Org_i$ acts as a node of the blockchain and has the blockchain account address $O_{\mathrm{acc}}$. The organizations use the network security threat intelligence in the process of sharing and using the network security threat intelligence. $O_{\mathrm{acc}}$ appears in the form which can effectively protect the identity information of organizations.
2) Center represents the network security threat intelligence analysis center, which has the function of analyzing network security threat intelligence, and is an indispensable trusted third party for reasoning and constructing a complete attack chain. Have a blockchain account address $C_{\mathrm{acc}}$.'
3) BlockNet represents the blockchain network, which consists of *Org* and Center.

4) *CtiDB* represents network security threat intelligence database, which can store network security threat intelligence. In this model, the intelligence in the network security threat intelligence database is encrypted intelligence Hash (*Cti*), which can effectively prevent the leakage of private information in the intelligence.
5) *Cti* represents network security threat intelligence $(Cti \in \{OneCti,\ TwoCti\})$. This paper mainly discusses one-dimensional and two-dimensional network security threat intelligence, and other multi-category network security threat intelligence can be divided into multiple binary or one-dimensional categories. A combination of cybersecurity threat intelligence.
6) Trans represents the transaction information on the blockchain, including the network security threat intelligence sharing transaction and the network security threat intelligence analysis transaction, namely $Trans \in \{STrans,\ ATrans\}$.
7) *SC* represents the smart contract created by the organization, which is composed of creator account address $O_{\mathrm{acc}}$, trigger condition, early warning response measure response, intelligence usage fee $uf$, using the quadruple: $\langle O_{\mathrm{acc}},\ condition,\ response,\ uf\rangle$. When *Center* is found in the intelligence analysis and reasoning that the triggering condition of the smart contract is met, the early warning response measures will be executed, and the fee $uf$ will be deducted from the account $O_{\mathrm{acc}},$ of the creator of the smart contract.
8) *Operation* represents the action operation between the main body *Org*, *Center*, *CtiDB* and *BlockNet*, including intelligence node registration (registry), threat intelligence sharing (sharing), evaluation, analysis, transaction broadcast, storage, extraction (get) and smart contract creation, etc. As shown in Fig. 3, when the organization $Org_1$ and *Center* share information, different subjects will involve sharing, get, evaluate, store, broadcast and other actions.

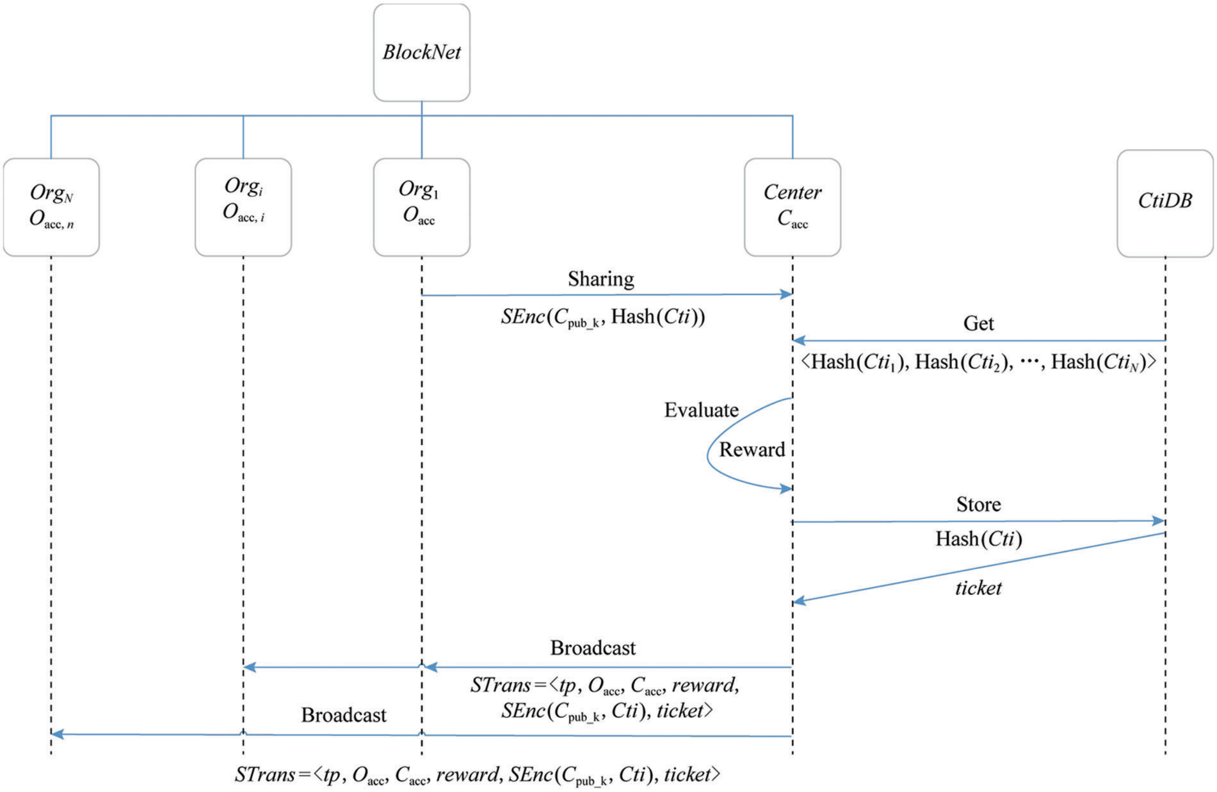

BlockNet
$Org_N$ $O_{\mathrm{acc},n}$
$Org_i$ $O_{\mathrm{acc},i}$
$Org_1$ $O_{\mathrm{acc}}$
Center $C_{\mathrm{acc}}$
CtiDB
Sharing
$SEnc(C_{\mathrm{pub\_k}}, \mathrm{Hash}(Cti))$
Get
$\langle \mathrm{Hash}(Cti_1), \mathrm{Hash}(Cti_2), \cdots, \mathrm{Hash}(Cti_N)\rangle$
Evaluate
Reward
Store
$\mathrm{Hash}(Cti)$
ticket
Broadcast
$STrans = \langle tp, O_{\mathrm{acc}}, C_{\mathrm{acc}}, reward, SEnc(C_{\mathrm{pub\_k}}, Cti), ticket\rangle$
Broadcast
$STrans = \langle tp, O_{\mathrm{acc}}, C_{\mathrm{acc}}, reward, SEnc(C_{\mathrm{pub\_k}}, Cti), ticket\rangle$

**Figure 3:** Intelligence sharing process

## 4 Intelligence Node Registration Algorithm

In order to ensure the credibility of intelligence nodes and protect intelligence sharing and use of private information, this paper introduces the intelligence node registration algorithm as shown in Algorithm 1. The organizations try to join the threat intelligence blockchain and become intelligence nodes. They need to report to the network. The Security Threat Intelligence Center performs registration, and the registration process relies on the standard encryption system to complete. Among them, the asymmetric encryption consists of triples: $\langle G_{\mathrm{sig}}, \ SEnc, \ SDec\rangle$. $G_{\mathrm{sig}}$ generates a public-private key pair, *SEnc* is an asymmetric encryption algorithm, and *SDec* is an asymmetric decryption algorithm. The one-way encryption function is represented by $H_{\mathrm{alg}}$. Specifically, steps 2 ~ 4 of Algorithm 1 generate the public-private key pair $\left(O_{\mathrm{pub_k}}, O_{\mathrm{pri\_k}}\right)$ that receives the information returned by the Threat Intelligence Center, and send $O_{\mathrm{pub_k}}$ to the Threat Intelligence Analysis Center. Steps 5 ~ 9 generate the public-private key pair $C_{\mathrm{pub_k}}$, $C_{\mathrm{pri_k}}$ and the one-way encryption function $H$ for the information submitted by the receiving organization. In order to verify whether $O_{\mathrm{pub_k}}$ has been tampered with, use Org's public key $O_{\mathrm{pub_k}}$ to encrypt $C_{\mathrm{pub_k}}$, and send the ciphertext and plaintext of the public key to Org. Steps 10 ~ 15 use its own private key to decrypt and obtain $C_{\mathrm{pub_{k1}}}$, and judge whether it is equal to the plaintext $C_{\mathrm{pub_k}}$. If it is equal, complete the negotiation of the encryption key and the designation of the one-way encryption algorithm during the information exchange. The algorithm is a linear algorithm, and the time complexity is in the range of O(1).

**Algorithm 1:** Information Node Registration

**Input:** Organization blockchain account $O_{\mathrm{acc}}$, Threat Intelligence Analysis Center account $C_{\mathrm{acc}}$

**Output:** Public key $O_{\mathrm{pub_k}}$ received by organization intelligence, public key $C_{\mathrm{pub_k}}$ received by threat intelligence analysis center.

```
1: inra $(O_{\mathrm{acc}}, C_{\mathrm{acc}})$
2. Organization to perform:
3. $O_{\mathrm{pub_k}}, \ O_{\mathrm{pri_k}} = G_{\mathrm{sig}}()$ /*Generate public-private key pair*/
4. Send $O_{\mathrm{pub_k}}$ to $C_{\mathrm{acc}}$
5. CTI Analysis Center performs:
6. $O_{\mathrm{pub_k}}, \ O_{\mathrm{pri_k}} = G_{\mathrm{sig}}()$
7. $H = H_{\mathrm{alg}}()$ /*Specify a one-way encryption algorithm*/
8. $PK_{Center}^{\mathrm{sig}} = SEnc\left(O_{pub_k}, C_{pub\_k}\right)$
9. Send $H, \ PK_{Center}^{\mathrm{sig}}, \ C_{\mathrm{pub_k}}$ to $O_{\mathrm{acc}}$
10. Organizational execution:
11. $C_{\mathrm{pub_{k1}}} = SDec\left(O_{\mathrm{pri_k}}, PK_{Center}^{\mathrm{sig}}\right)$
12. if $C_{\mathrm{pub_{k1}}} == C_{\mathrm{pub_k}}$
13. return $O_{pub_k}, \ C_{\mathrm{pub_k}}, \ H$
14. else
15. return None
16. end if
```

## 5 Intelligence Data Accounting Algorithm

In order to ensure the healthy development of the intelligence sharing community and avoid the "free-rider" behavior of intelligence nodes that only use intelligence and do not share intelligence, this paper designs the intelligence data accounting algorithm shown in Algorithm 2, which realizes the organization intelligence sharing between the agency and the network security threat intelligence center. The shared intelligence is encrypted and recorded in the blockchain, which is convenient for the follow-up threat source traceability and decryption. At the same time, a reward mechanism is introduced to encourage intelligence nodes to actively share intelligence. First, the organization submits the encrypted network security threat intelligence $SEnc\left(C_{\mathrm{pub}_{\mathrm{k}}}, Cti\right)\left(Cti = \langle tp,\ type, H(value), label\rangle || \langle tp, type,\ H(value),\ rel,\ type,\ H(value),\ desc\rangle\right)$. Then, as shown in steps 2 ~ 6 in Algorithm 2, the intelligence analysis center decrypts, evaluates and stores the submitted intelligence after receiving the submitted intelligence. The value assessment of threat intelligence is mainly based on the degree of correlation with the existing intelligence. Finally, as shown in steps 7 ~ 8 in Algorithm 2, the $C_{\mathrm{acc}}$ broadcasts the intelligence sharing transaction information to the entire blockchain network to complete the accounting operation of intelligence sharing.

**Algorithm 2:** Intelligence data accounting algorithm

**Input:** Intelligence received time *receive_time*, organization account $O_{\mathrm{acc}}$, threat intelligence center account $C_{\mathrm{acc}}$, intelligence message $SEnc\left(C_{pub_k}, Cti\right)$ encrypted by asymmetric encryption algorithm, threat intelligence center private key $C_{pri_k}$

**Output:** Receive success True or receive failure False

```
1: $idaa\ \left(receive_{time}, O_{\mathrm{acc}}, C_{\mathrm{acc}},\ SENc\left(C_{pubk}, Cti\right), C_{pri_k}\ \right)$
2: $enc\_msg =\ SEnc\left(C_{pubk}, Cti\right)$
3: $msg = SDec\left(C_{pri_k},\ enc\_msg\right)$
4: $reward = evaluate\_cti(msg);$ /*Assessing threat intelligence value*/
5: $if\ store(msg,\ CtiDB)$
6: $ticket = CtiDB.find(msg).index$
7: $strans = (receive_{time},\ O_{\mathrm{acc}}, C_{\mathrm{acc}}, reward, enc\_msg, ticket)$
8: $broad\_to\_blockchain(strans)$; /*Broadcasting transaction information in the blockchain*/
9: return True
10: else
11: return False
12: end if
```

## 6 Analysis and Trading Algorithms Based on Intelligence Graphs

In order to effectively use the network security threat intelligence shared by the intelligence nodes, this paper introduces the intelligence graph-based analysis and transaction algorithm shown in Algorithm 3, which realizes the intelligence analysis between the organization and the network security threat intelligence center. The combined intelligence clues and intelligence results are recorded in the blockchain. Specifically, the network security threat intelligence center and the existing intelligence are correlated with each other to construct a network security threat intelligence map. Then, use the label propagation algorithm to infer the labels of the threat intelligence elements, and then use the topological sorting algorithm to construct the attack chain. Finally, according to the attack chain and element label, with the help of the traceability function of the blockchain, the one-way encrypted network security threat element value is decrypted, returned to the initiator of the analysis request, and the analysis transaction information is written into the blockchain.

**Algorithm 3:** Analysis and Trading Based on Threat Intelligence Graph

**Input:** Encrypted threat intelligence *msg*, threat intelligence database *CtiDB*

**Output:** Intelligence *label_list*, *attack_chain*

```
$igata\ (msg, CtiDB\ )$
init graph $G$
$hash\_value\_list = get\_hash\_value(msg)$
for $hash\_value$ in $hash\_value\_list$
    if $hash\_value$ not in $G$
        $G.add(hash\_value)$
    end if
end for
for $v$ in $G$
    $two\_cti = CtiDB.find\_two\_cti(v)$
    for $item$ in $[two\_cti, src\_value, two\_cti.dst\_value]$
        if $item! = v$ and not in $G$
            $G.add(item)$
            $G.L[item] = CtiDB.find\_one\_cti(item).label$
        end if
    end for
end for
for $hash\_value$ in $hash\_value\_list$
    $label\_list.append(label\_propagation(G, hash\_value))$
end for
$attack\_chain = top\_sort(G)$
$atrans = construct\_atrons(msg, label\_list, attack\_chain)$
$broad\_to\_blockchain(atrans)$
return $label\_list, attack\_chain$
```

Time complexity analysis of Algorithm 3: The main time of Algorithm 3 is spent in the process of building the intelligence graph. Assuming that the number of nodes in the constructed intelligence graph is $n$, the directed edges are $m$, and *CtiDB* uses the form of key-value pairs for database storage or cloud computing. The storage and access time is O(1), then, the time complexity of building the intelligence graph is $O(n^2)$, the time consumption of the label propagation algorithm is less than $O(m)$, and the time consumption of the attack chain construction process is $O(n + m)$, Then the total time complexity is close to $O(n^2)$.

## 7 Early Warning Response Algorithm Based on Smart Contract

In view of the time asymmetry of network security attack and defense confrontation, how to improve the response speed of the defender has become a focus of attack and defense confrontation. In order to improve the speed of early warning response in the intelligence blockchain network, a smart contract-based early warning response mechanism is introduced into the model, as shown in Algorithm 4. First, the organization (steps 2 ~ 4) will create smart contracts on the blockchain for the systems that need key protection, and broadcast the entire blockchain network. Then, when analyzing the intelligence, the cybersecurity threat intelligence center will analyze the threat intelligence related to the smart contract created by the organization. If there is any threat intelligence that meets the triggering conditions of the smart contract, the model will automatically trigger the smart contract and execute it. Among them, the early warning response process or measures, and the analysis is transformed into network security threat intelligence analysis transaction information broadcast intelligence blockchain network.

**Algorithm 4:** Smart contract-based early warning response

**Input:** Smart contract creator account address $O_{\mathrm{acc}}$, smart contract *condition*, early warning measure *response*, paid fee *uf*

**Output:** Threat Intelligence *Cti* that satisfies the smart contract conditions.

```
1: $sera\ (msg, CtiDB\ )$
2: Organization to perform:
3: $sc = create\_sc(condition,\ response,\ uf)$
4: $broad\_to\_blockchain(sc)$; /*Broadcast smart contracts within the intelligence blockchain network*/
5: CTI Center performs:
6: if $Cti$ match $condition$
7: $execute\ (response)$
8: $atrans = construct\_atrans(Cti)$
9: $broad\_to\_blockchain(atrans)$
10: return $SEnc\left(O_{\mathrm{pub_k}},\ Cti\right)$
11: end if
```

## 8 Experimental Evaluation

In order to verify and evaluate the effect of the above threat intelligence sharing model, this paper designs a network with the topology shown in Fig. 4, among which 16.1.5.20, 16.1.5.23, 16.1.5.26, 16.1.5.29, 16.1.5.30 constitute the blockchain network, 16.1.5.20 carries the function of the network security threat intelligence analysis center, and the network security threat intelligence database is stored on the cloud storage built by ownCloud.

Referring to the kill chain (reconnaissance and tracking, weapon construction, payload delivery, vulnerability exploitation, installation and implantation, command and control, and target achievement), this paper designs a simulated attack scenario as shown in Fig. 5: 2019-05-26T10:00:00, the attacker Alice used the server with the IP of 12.1.5.67 to attack the server of the $Org_0$ enterprise with the IP of 16.1.5.29, and dropped the malicious code payload $M_0$, and then used the 16.1.5.29 as Springboard captured the 16.1.5.30 server and dropped malicious code payload $M_1$ (a variant of $M_0$). The malicious code $M_0$ uses its own propagation mechanism to spread to the server whose IP is 16.1.5.26. At

2019-05-27T01:00:00, the attacker Alice received the new zombie machine 16.1.5.26 and used it to compromise the 16.1.5.23 server of the enterprise $Org_3$ and dropped malicious code payloads $M_1$.

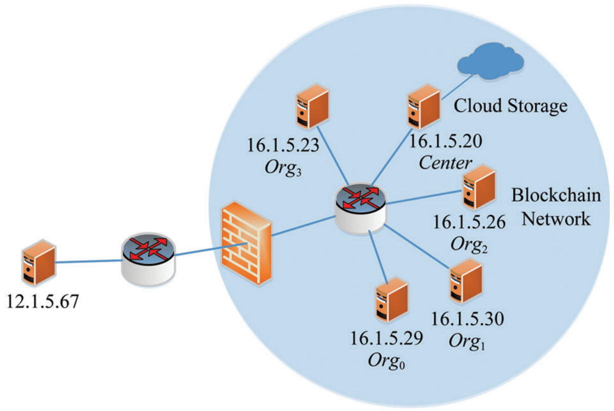


**Figure 4:** Proposed network topology

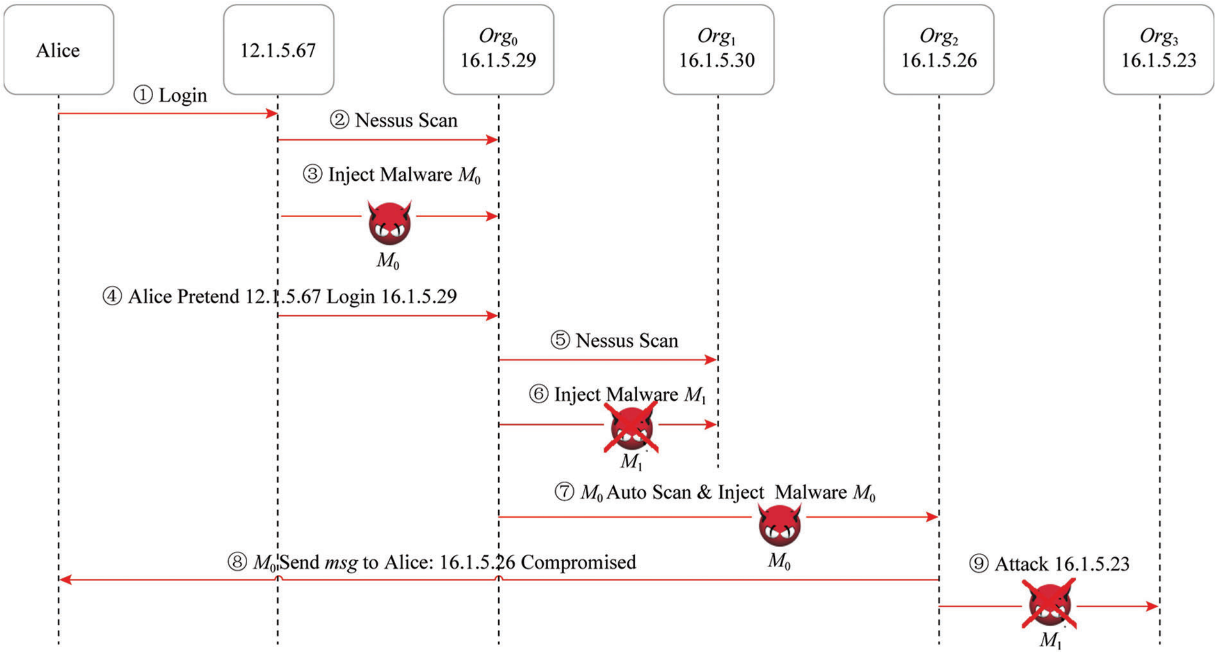
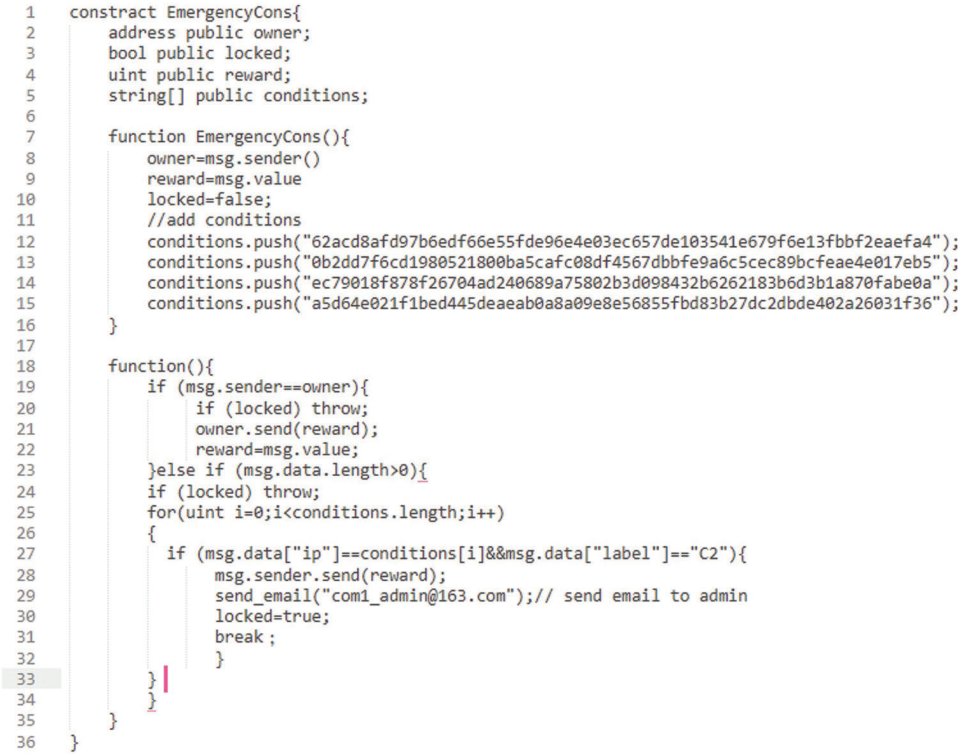


**Figure 5:** Illustrations of cyber attack

$Org_1$ creates a smart contract as shown in Fig. 6 on the blockchain network, as long as the conditions are met: $ip \in \{\mathrm{SHA256}(16.1.5.23), \ldots \mathrm{SHA256}(16.1.5.29)\}$ & $\mathrm{label} = \mathrm{C2}$, execute the function $send\_email$(“org1_admin@163.com,”) to send alarm information to the server administrator.

In order to meet the model requirements, in the experimental evaluation, the asymmetric encryption adopts the Rivest-Shamir-Adleman (RSA) algorithm, the symmetric encryption adopts the Advanced Encryption Standard (AES) algorithm, and the one-way encryption adopts the SHA256 algorithm. According to the model design, the IP information in this paper appears in the form of SHA256(IP) during intelligence sharing, that is, as shown in the value of SHA256(IP) in Tab. 2, which avoids the leakage of IP information in intelligence. The same applies to domain names, email addresses and names. The STIX standard is referred to when sharing network security threat intelligence in the community. As shown in Fig. 7, first, each organization Org registers with the Cyber Security Threat Intelligence Center. Then, $Org_1$ creates the smart contract as shown in Fig. 6. Subsequently, $Org_0$ shared the first cybersecurity threat intelligence with, because the triggering condition of the smart contract SHA256(16.1.5.2) $\in$ {SHA256(16.1.5.23),…, SHA256(16.1.5.29)} & *label* = C2, the system sends a threat source request to $Org_0$ (because SHA256(16.1.5.29) involves private information, $Org_0$ chooses not to decrypt the IP address, but returns nothing. The behavior and emergency response method of the malicious code $M_0$ involving privacy and trigger the execution of the smart contract of $Org_1$, and send the alarm information to the administrator of 16.1.5.30. Finally, when $Org_3$ issues <2019-05-27T01:00:00, *ip*, SHA256 (16.1.5.26) >, the network security threat intelligence analysis requirements, the system builds an intelligence graph with the help of existing network security threat intelligence, and uses the label propagation algorithm calculates that the label of SHA256 (16.1.5.26) is C2, and at the same time builds a possible complete attack chain: SHA256 (12.1.5.67) → SHA256 (16.1.5.29) → SHA256 (16.1.5.26), and use the blockchain to decrypt SHA256 (12.1.5.67) to get 12.1.5.67. Tab. 3 provides the sharing list of the cyber threat intelligence.



```
constract EmergencyCons{
    address public owner;
    bool public locked;
    uint public reward;
    string[] public conditions;

    function EmergencyCons(){
        owner=msg.sender()
        reward=msg.value
        locked=false;
        //add conditions
        conditions.push("62acd8afd97b6edf66e55fde96e4e03ec657de103541e679f6e13fbbf2eaefa4");
        conditions.push("0b2dd7f6cd1980521800ba5cafc08df4567dbbfe9a6c5cec89bcfeae4e017eb5");
        conditions.push("ec79018f878f26704ad240689a75802b3d098432b6262183b6d3b1a870fabe0a");
        conditions.push("a5d64e021f1bed445deaeab0a8a09e8e56855fbd83b27dc2dbde402a26031f36");
    }

    function(){
        if (msg.sender==owner){
             if (locked) throw;
             owner.send(reward);
             reward=msg.value;
        }else if (msg.data.length>0){
        if (locked) throw;
        for(uint i=0;i<conditions.length;i++)
        {
          if (msg.data["ip"]==conditions[i]&&msg.data["label"]=="C2"){
              msg.sender.send(reward);
              send_email("com1_admin@163.com");// send email to admin
              locked=true;
              break ;
              }
        }
        }
    }
}
```

**Figure 6:** Illustrations $Org_1$ smart contract

The above process verifies that the blockchain-based threat intelligence sharing model can build a complete attack chain while protecting intelligence privacy. In order to further prove the validity of this model, this paper extracts 15 complete attack chains from 2010 pieces of intelligence information collected by open source intelligence, and maps the 30 IP addresses involved to 16.1.5.10~16.1.5.29 experiment in the address segment. The 30 IPs are 4 C2IPs, 11 attack IPs, and

15 attack target IPs. Fig. 8 shows the impact of the proportion of a certain type of tagged IP becoming an intelligence node on the successful construction of an attack chain when other types of tagged IPs have become intelligence nodes. It can be seen that when C2IP becomes an intelligence node, threat intelligence sharing is more effective to the construction of a complete attack chain.

**Table 2:** Cross-reference analysis

| | Attacker/Org | | | | |
|---|---|---|---|---|---|
| | Alice | $Org_0$ | $Org_1$ | $Org_2$ | $Org_3$ |
| IP | 12.1.5.67 | 16.1.5.29 | 16.1.5.30 | 16.1.5.26 | 16.1.5.23 |
| SHA256 (IP) | 558692953c968a26a7187824a229d63be84753a4a0acace95982e1bbb2761a7b | 62acd8afd97b66e55fde96e4e03ec657de103541e679f6e13fbbf2eaefa4 | 0b2dd7f6cd1980521800ba5cafc08df4567dbbfe9a6c5cec89bcfeae4e017eb5 | Ec79018f878f26704ad240689a75802b3d098432b6262183b6d3b1a870fabe0a | A5d64e021f1bed445deaeab0a8a09e8e56855fbd83b27dc2dbde402a26031f36 |

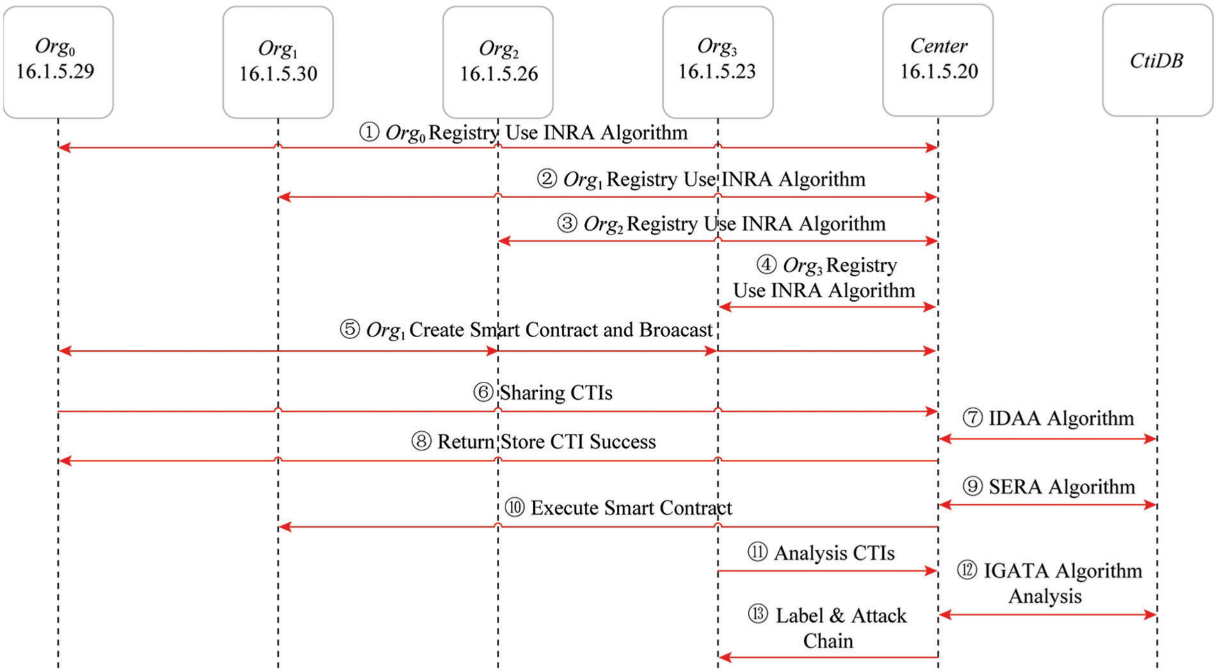

$Org_0$ 16.1.5.29
$Org_1$ 16.1.5.30
$Org_2$ 16.1.5.26
$Org_3$ 16.1.5.23
Center 16.1.5.20
CtiDB
① $Org_0$ Registry Use INRA Algorithm
② $Org_1$ Registry Use INRA Algorithm
③ $Org_2$ Registry Use INRA Algorithm
④ $Org_3$ Registry Use INRA Algorithm
⑤ $Org_1$ Create Smart Contract and Broacast
⑥ Sharing CTIs
⑦ IDAA Algorithm
⑧ Return Store CTI Success
⑨ SERA Algorithm
⑩ Execute Smart Contract
⑪ Analysis CTIs
⑫ IGATA Algorithm Analysis
⑬ Label & Attack Chain

**Figure 7:** Flow illustrations of CTI

**Table 3:** CTI values evaluation

| CTI tuple | $Org_0$ | $Org_1$ | $Org_2$ |
|---|---|---|---|
| | $< 2019 - 05 - 26T10 : 00 : 00, ip,$<br>$SHA256(16.1.5.29,\ C2) >$ | $< 2019 - 05 - 26T10 : 00 : 00,\ ip,$<br>$SHA256(12.1.5.67),$<br>“src ip connect target ip” > | $< 2019 - 05 - 26T11 : 00 : 00, ip,$<br>$SHA256(16.1.5.29),$<br>$scan,\ ip,\ SHA256(16.1.5.26),$<br>“*Mo* malware scan server” > |

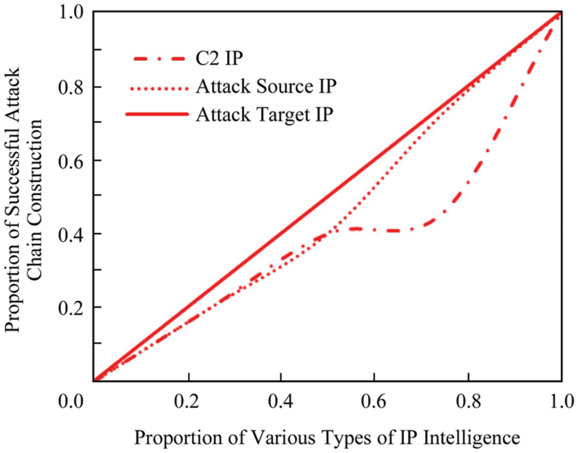


**Figure 8:** Comparison of various attacks

In order to further evaluate the privacy protection of this model in the process of intelligence sharing, this paper conducts analysis and simulation experiments with the existing typical models in terms of the privacy protection strength of the intelligence sharing party, the intelligence utilizing party, and the third party involved in the intelligence. The experimental results are as follows: As shown in Fig. 9, the vertical axis represents the privacy protection strength, and the privacy protection strength is set according to the cracking difficulty of the privacy protection algorithm in Tab. 4. The horizontal axis represents the privacy that needs to be protected in intelligence sharing (that is, the privacy of the intelligence sharing party, the privacy of the intelligence user, and the privacy of the third party involved in the intelligence) and the comprehensive privacy protection evaluation index (according to the importance of the three-party privacy protection to intelligence sharing, the use of Eq. (2) calculated by the weighted average method as:

$$f = 0.5s_s + 0.3s_u + 0.2s_t \tag{2}$$

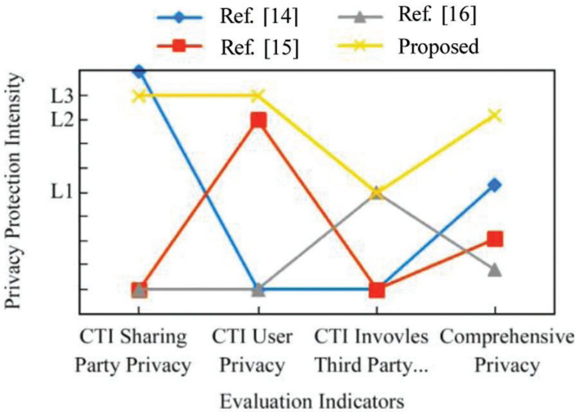


**Figure 9:** Privacy protection comparison of the proposed and existing algorithms

Among them, $s_s$ represents the privacy protection strength of the intelligence sharing party; $s_u$ represents the privacy protection strength of the intelligence user; $s_t$ indicates that intelligence involves the strength of third-party privacy protection.

**Table 4:** Comparison of algorithms complexity

| Algorithm | Complexity |
|---|---|
| Ref. [15] | $O(n^3),\ n = \log N,\ N = pq$ |
| Ref. [16] | $O\left(2^{(k-1)p}\right)$ |
| Proposed | $O(2^n)$ |

In Fig. 9, the method of reference [14] can better solve the privacy protection problem of the intelligence sharing party, but this solution is insufficient for the privacy protection of the intelligence user and the third party involved in the intelligence. Reference [15] protects the privacy of the intelligence user, but the protection strength is still lower than the protection intensity of the intelligence user in this paper. Reference [16] mainly protects the third-party privacy involved in intelligence, and its adoption method is similar to the third-party privacy protection method in this paper, so the protection strength is basically the same. It can be concluded that the proposed scheme to protect the privacy of the intelligence sharing party and the intelligence user by using the blockchain anonymity mechanism and the one-way encryption algorithm to protect the privacy of the third party involved in intelligence has obvious advantages in terms of comprehensive privacy protection strength.

## 9 Conclusion

Aiming at the contradiction between privacy protection and attack chain construction in the current network security threat intelligence sharing process, this paper proposes a blockchain-based network security threat intelligence sharing model, which utilizes the decentralization and anonymity of blockchain technology. It not only protects the private information of participating organizations and involved organizations in cybersecurity threat intelligence sharing, but also facilitates reasoning and analysis of the complete cyberattack chain. Use the backtracking capability of the blockchain to retrospectively restore the threat source in the attack chain. Using the smart contract mechanism to realize automatic early warning and response to network threats. Finally, the feasibility and effectiveness of the model are verified by simulation experiments. Future work is to consider cache analysis of the blockchain network in view of the proposed approach.

**Acknowledgement:** The authors would like to thanks the editors and reviewers for their review and recommendations.

**Funding Statement:** The authors received no specific funding for this study.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

[1] Y. Chen, J. Wu, Y. Hsieh and C. Hsuh, “An oracle-based on-chain privacy,” *Computers Journal*, vol. 9, no. 3, pp. 1–16, 2020.

[2] Y. Liu, X. Liu, C. Tang, J. Wang and L. Zhang, “Unlinkable coin mixing scheme for transaction privacy enhancement of bitcoin,” *IEEE Access*, vol. 6, pp. 23261–23270, 2018.

[3] N. Amarasinghe, X. Boyen and M. Kague, “The cryptographic complexity of anonymous coins: A systematic exploration,” *Cryptography Journal*, vol. 5, no. 1, pp. 1–17, 2021.

[4] A. Z. Junejo, M. A. Hashmani and M. Memon, “Empirical evaluation of privacy efficiency in blockchain networks: Review and open challenges,” *Applied Sciences*, vol. 11, no. 15, pp. 1–18, 2021.

[5] K. Qureshi, L. Shahzad, A. Abdelmaboud, T. Eisa, B. Alamri *et al.,* "A blockchain-based efficient, secure and anonymous conditional privacy-preserving and authentication scheme for the internet of vehicles," *Applied Sciences Journal*, vol. 12, no. 1, pp. 1–18, 2022.

[6] J. Alupotha, X. Boyen and M. Mckague, "Aggregable confidential transactions for efficient quantum-safe cryptocurrencies," *IEEE Access*, vol. 10, pp. 17722–17747, 2022.

[7] A. Silveria, G. Betarte, M. Cristia and C. Luna, "A formal analysis of the mimblewimble cryptocurrency protocol," *Sensors Journal*, vol. 21, no. 17, pp. 1–18, 2021.

[8] Q. Wang, B. Qin, J. Hu and F. Xiao, "Preserving transaction privacy in bitcoin," *Future Generation Computer Systems*, vol. 107, no. 3, pp. 793–804, 2020.

[9] B. Bunz, J. Bootle, D. Boneh, A. Poelstra, P. Wuille *et al.,* "Bulletproofs: Short proofs for confidential transactions and more," in *IEEE Symp. on Security and Privacy (ISP)*, pp. 969–975, 2018.

[10] O. Ersoy, T. Pedersen, K. Kaya, A. Selcuk and E. Anarim, "A crt-based verifiable secret sharing scheme secure against unbounded adversaries," *Security and Communication Networks*, vol. 9, no. 7, pp. 4416–4427, 2016.

[11] K. Caramancion, Y. Li, E. Dubois and E. Jung, "The missing case of disinformation from the cybersecurity risk continuum: A comparative assessment of disinformation with other cyber threats," *Data Journal*, vol. 7, no. 4, pp. 1–24, 2022.

[12] A. Ofori, S. Islam, S. Lee, Z. Shamszaman, K. Muhammad *et al.,* "Cyber threat predictive analytics for improving cyber supply chain security," *IEEE Access*, vol. 9, pp. 94318–94337, 2021.

[13] W. Tounsi and H. Rais, "A survery on technical threat intelligence in the age of sophisticated cyber attacks," *Computers & Security*, vol. 72, no. 3, pp. 212–233, 2018.

[14] H. Wang, K. Fan, K. Zhang, Z. Wang, H. Li *et al.,* "Secure and efficient data privacy-preserving scheme for mobile cyber physical systems," *IEEE Internet of Things Journal*, vol. 7, no. 2, pp. 1018–1029, 2021.

[15] C. Gupta, I. Johri, K. Srivasan, Y. Hu, S. Qaisar *et al.,* "A systematic review on machine learning and deep learning models for electronic information security in mobile networks," *Sensors*, vol. 22, no. 5, pp. 1–31, 2022.

[16] G. Sakellariou, P. Fouliras, I. Mavridis and P. Sarigiannidis, "A reference model for cyber threat intelligence (CTI) systems," *Electronics Journal*, vol. 11, no. 9, pp. 1–18, 2022.

[17] J. Fuentes, L. Gonzalez and J. Tapiador, "PRACIS: Privacy-preserving and aggregatable cybersecurity information sharing," *Computers & Security*, vol. 69, no. 4, pp. 127–141, 2017.

[18] K. Rantos, A. Spyros, A. Papanikolaou, A. Kritsas, C. Ilioudis *et al.,* "Interoperability challenges in the cybersecurity information sharing ecosystem," *Computers*, vol. 9, no. 1, pp. 1–19, 2020.

[19] H. Zhang, Y. Pan, Z. Lu, J. Wang and Z. Liu, "A cyber security evaluation framework for in-vehicle electrical control units," *IEEE Access*, vol. 9, pp. 149690–149706, 2021.

[20] P. Esmaeilzadeh and T. Mirzaei, "The potential of blockchain technology for health information exchange: Experimental study from patients' perspectives," *Journal of Medical Internet Research*, vol. 21, no. 6, pp. 3713–3721, 2019.

[21] K. Jiawen, Y. Rong and H. Xuming, "Blockchain for secure and efficient data sharing in vehicular edge computing and networks," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4660–4670, 2019.

[22] D. Rawat, L. Njilla and K. Kwiat, "iShare: Blockchain-based privacy-aware multi-agent information sharing games for cybersecurity," in *IEEE Int. Conf. on Computing, Networking and Communications (ICNC)*, Maui, USA, pp. 425–431, 2018.

[23] Z. Rashid, U. Noor and J. Altmann, "Economic model for evaluating the value creation through information sharing within the cybersecurity information sharing ecosystem," *Future Generation Computer Systems*, vol. 124, no. 4, pp. 436–466, 2021.

[24] K. Kim, F. Alfouzan and H. Kim, "Cyber-attack scoring model based on the offensive cybersecurity framework," *Applied Sciences*, vol. 11, no. 15, pp. 1–17, 2021.

[25] W. Ruzai, M. Ariffin, M. Asbullah and M. Zahari, "On the improvement attack upon some variants of RSA cryptosystem via the continued fractions method," *IEEE Access*, vol. 8, pp. 80997–81006, 2020.

[26] R. Hamza, A. Hassan, A. Ali, M. Bashir, S. Alqhtani *et al.,* "Towards secure big data analysis via fully homomorphic encryption algorithms," *Entropy*, vol. 24, no. 4, pp. 1–28, 2022.

[27] M. Shuaib, N. Hassan, S. Usman, S. Alam, S. Bhatia *et al.,* "Land registry framework based on self-sovereign identity (SSI) for environmental sustainability," *Sustainability*, vol. 14, no. 9, pp. 1–19, 2022.

[28] M. Shuaib, N. Hassan, S. Usman, S. Alam, S. Bhatia *et al.,* "Self-sovereign identity solution for blockchain-based lang registry system: A comparison," *Mobile Information Systems*, vol. 9, no. 3, pp. 1–16, 2022.

[29] S. Bhatia, S. Alam, M. Shuaib, M. Hameed, F. Jeribi *et al.,* "Retinal vessel extraction via assisted multi-channel feature map and u-net," *Frontiers in Public Health*, vol. 10, no. 3, pp. 1765–1773, 2022.

[30] M. Shuaib, N. Hassan, S. Usman, S. Alam, S. Bhatia *et al.,* "Identity model for blockchain-based land registry system: A comparison," *Mobile Information Systems*, vol. 7, no. 6, pp. 1–16, 2022.

[31] M. Rahmani, M. Shuaib, S. Alam, S. Siddiqui, S. Ahmad *et al.,* "Blockchain-based trust management framework for cloud computing-based internet of medical things (IoMT): A systematic review," *Computational Intelligence and Neuroscience*, vol. 22, no. 1, pp. 1–14, 2022.

[32] S. Alam, M. Shuaib, W. Khan, S. Garg, G. Kaddoum *et al.,* "Blockchain-based initiatives: Current state and challenges," *Computer Networks*, vol. 198, no. 6, pp. 1085–1097, 2021.