Blockchain based Security Framework for P2P Filesharing system

Srikanta Pradhan
Department of Computer Science & Engineering
Indian Institute of Technology Patna
India
Email: srikanta.pcs16@iitp.ac.in

Somanath Tripathy
Department of Computer Science & Engineering
Indian Institute of Technology Patna
India
Email: som@iitp.ac.in

Sukumar Nandi
Department of Computer Science & Engineering
Indian Institute of Technology Guwahati
India
Email: sukumar@iitg.ernet.in

***Abstract*—Peer to Peer (P2P) is a dynamic and self-organized technology, popularly used in File sharing applications to achieve better performance and avoids single point of failure. The popularity of this network has attracted many attackers framing different attacks including Sybil attack, Routing Table Insertion attack (RTI) and Free Riding. Many mitigation methods are also proposed to defend or reduce the impact of such attacks. However, most of those approaches are protocol specific. In this work, we propose a Blockchain based security framework for P2P network to address such security issues. which can be tailored to any P2P file-sharing system.**

***Index Terms*—Peer-to-Peer Network, Blockchain, File Sharing**

I. Introduction

In the world of high popularity of centralized system like cloud computing, the Peer-to-Peer (P2P) system still able to maintain its relevance due to its distributed underlying concept. This paradigm has a wide range of applications in data dissemination and data discover including file sharing, multimedia streaming, and the Internet of Things (IoT) enabled communication systems [1], [2]. The current generation P2P system has been improved to provides confidentiality and user privacy. Further, some P2P systems like TOR [3] provides anonymous routing to hide the message source.

Meanwhile, the popularity of P2P system attracts many threats for itself. A malicious node can force the target node to insert a malicious node-id entry in its routing table (RT). Such an attack, to insert maliciousIDs into the peer's routing table is referred as Routing Table Poisoning (RTP) or Routing Table Insertion (RTI) attack. RTP and its impact have been studied in [4], [5]. Many authors including [6]–[8] have proposed different countermeasures to mitigate such attacks. Most of the existing works have not considered the collaborative attacks in which a group of malicious peers colluded to perform the RT poisoning. In [9], authors have demonstrated the Routing Table Insertion (RTI) attack on Freenet. Baumeister et.al, [10], proposed a countermeasure for such attack in Freenet. Authors in [11], presented an RTP attack called the collaborative attack on Pastry [12] and suggested a defense mechanism which reduces the impact of such attacks. However, these mitigation techniques are observed to be network protocol specific and cannot be applicable to different networks.

This work proposes a Blockchain based security framework tailored to the P2P network to defend against the collaborative attacks. Blockchain is used in several crypto-currencies where it supported such a huge volume. We have also used the Blockchain in our mechanism for transfer of respect scores. These transfer of respect score are treated as a transaction. The transactions are collected to create a block. Collection of blocks are maintained in the Blockchain record; as manipulating the transactions in Blockchain is difficult. The framework provides security from various attacks.

The proposed framework uses a remuneration method called respect score. Each participating node in the service gets an incentive to increase the respect score for providing services. The requesting node transfers respect score to all the participated nodes after receiving successful reply message. We have considered that the network is very large and many service requests are served at any moment in time. This will create a large number of credit transfer from the requesting nodes to others. It is necessary to have a powerful mechanism to handle such a large volume of transactions.

The rest of the paper is organized a follows. The related works are discussed in Section II. The Section III describes the structure of Blockchain. The P2P system model and potential threats are discussed in Section IV. Our proposed system is presented in Section V. In Section VI we have discussed security analysis of our system. And finally, we concluded the paper in Section VII.

II. Related Work

In this work, we emphasize on the performance degradation by malicious nodes present inside the network. Attackers have discovered various methods to exploit the routing table of a structured P2P network. Routing table poisoning attack [6] forged information about proximity and increases the fraction of bad entries in routing table. The aim of routing table insertion(RTI) attack is to insert a few malicious nodeIDs into routing table of a target node.

Germanus [13] et al. proposed a different type of Eclipse attack on P2P protocol which was named as taLEA. It requires small extent of resources to target a peer in the network. The authors have presented a scheme for the applications

978-1-5386-8134-3/18/$31.00 ©2018 IEEE

where safety and time are most critical aspects. They have observed the path of different lookup request and the point of convergent destination. Tsehayu et al. [14], proposed a mechanism to defend against an attack where a malicious node doesn't forward any lookup messages passing through it, in a Chord overlay. Wang et al. [15], observed the attack traffic to detect a multi-node collaborative attack. Ming He et al. proposed scheme [16], checks the integrity of a message block through the proposed time key scheme. Malicious nodes creates fabricated messages with known time keys. This mechanism defends pollution attacks with arbitrary colluded malicious nodes. Almeida et al. [17] recommended a model, based on reputation mechanism to defend against pollution attack for mesh-pull based live streaming applications.

Uddin et. al. [18] have proposed an intrusion detection system (IDS) to detect DDoS. The IDS uses both signature based and anomaly based detection mapped to artificial immune system(AIS) and proposed generation of detector for the detection of DDoS attacks.

Meng and Liu [19] proposed GeTrust, a trust model for Chord based P2P network, which is inspired from trust relationship in human society. The service peer have guarantee(s) peer which provides guarantee for the service that is going to provide. Both service peer and guarantee peer pledge for the reputation mortgage for the service. And both the peers will get incentives for the successful service. GeTrust gives more chance to peers with high reputation to use authentic services. This improves honest peer's experience and enhance network availability. We have noted that all the above solutions are tailored for a specific P2P network protocol.

III. The Blokchain System

The Blockchain is first used in Bitcoin cryptocurrency network. This network provides virtual currency which does not require any central bank to manage. Bitcoin network can handle a large number of transactions because of the robust nature of underlying Blockchain technology. In Blockchain, once information has been recorded, it is difficult to change. The Blockchain technology consists of a linear list of linked data blocks called Blockchain and a set of nodes called miner who creates these blocks.

A. Blockchain

The Blockchain is useful to keep the record which can be accessible by the all the users of the network. The important feature of the Blockchain is that it is very difficult for any user or group of users to modify the information once recorded into it. A Blockchain is a linear list of blocks. A block of the Blockchain keeps the hash of the previous block (PrevHash) in it along with the transactions. This PrevHash links to the previous block and thus keeps all the blocks linked as shown in Fig 1. A user can traverse to the beginning of the Blockchain by following through the PrevHash value of each block. Any node of the network can keep a copy of this Blockchain. The process of generating a block is called mining which is performed by miner node.

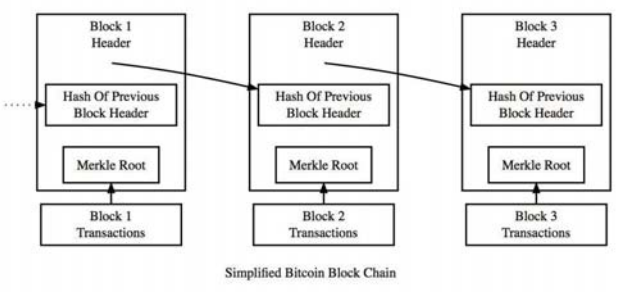

Fig. 1. Typical Structure of Blockchain

B. Miner

A miner in the Blockchain technology is a node with more computational power. It collects all transaction information broadcasted by the nodes of the network and keeps in the block along with the hash of the previous block. The miner has to solve a mathematical problem i.e. to find suitable nonce for which the hash of the block will be less than a predefined value. This is called proof-of-work. After successfully solving the problem i.e. finding appropriate nonce, the miner is allowed to add the block into its Blockchain. The miner broadcast it to the network so that other miner can know and verify it. The miner who solves the mathematical problem at earliest gets incentives as per the network protocol. And then all miners compete for creating a new block.

Many different problems are proposed such as proof-of-stake, proof-of-capacity through which a miner has to prove its superiority over other miners to gain the network incentives. Consequently, Blockchain has provided a mechanism to keep the data integrity, consistency in decentralized networks.

IV. P2P System and Threat Model

A. P2P System

A P2P overlay network can be modeled as a directed graph $G = (V, E)$, in which V $(v_1, v_2, \ldots, v_n)$ is the set of vertex representing Peer and a directed edge $e_{ij} \in E$ exists if the RT of V_i contains V_j. Thus the edge e_{ij} is different from e_{ji}. It is assumed that n number of peers in the network i.e $|v| = n$. Let M and G be the set of malicious peers and Genuine peers in the network. Thus $V = M \cup G$. Also, if $|M| = m$ and $|G| = g$ then $g = n - m$.

The following important features can be observed in a structured P2P File-sharing.

1) Overlay ID space: Both the Peer and file are referred as the resource of the system and can be identified by a unique ID belong to the same overlay ID space. In common, a cryptographic hash function is used to compute the ID. Hash of IP or MAC is referred to the nodeID while hash of file\filename\title is referred to be fileID or key.

2) Routing Table (RT): The routing table is the most crucial data structure used by each peer in an overlay network. Each RT entry is a tuple containing nodeID of a neighbor and its corresponding underlay network connection like IP and

port etc. Usually, there are $t = c * log_2^n$ entries in an RT to hold t neighbor address where c is a constant specific to a P2P protocol. As the peers crash frequently, each peer in the network updates the correct peer found during the RT management deleting the unused peers.

B. Adversary model

The malicious nodes can exploit the vulnerabilities of existing P2P network with the aim to launch different attacks. In this example, the malicious nodes are colluded and can share the necessary information among them. Malicious node remembers what keys other malicious nodes are having. So that it can initiate a lookup request to search those keys. These malicious nodes further share information about their neighbor node. These nodes have the capability to modify the message. In the case of routing table insertion attack, these nodes modify the lookup response message such that the message would be received by the target node. An adversary can create a number of pseudonyms with the aim of launching a Sybil attack.

C. Potential Threats

The performance and security of the network are based on assumption that the nodes are uniformly distributed. By attacking on the node(s), the attacker can gain control over those node(s) and start behaving differently. In a P2P network, an attack by a single attacker is very difficult and also the gain is very less in such attacks. So the attackers may collude together and launch different attacks. Thus a part of the network can be controlled by the attacker. Few of such collaborative attacks are discussed below.

1) Sybil Attack: In a Sybil attack, the attacker creates a number of pseudonymous identities. Then the attacker uses these ids to gain disproportionate influences. The attacker uses this method to effect a voting outcome. By manipulating an election, attackers can control a part of the network or whole network and can interrupt network services. Rowaihy et al. [20] proposed a hierarchical system based on computational puzzles to defend against Sybil attacks. This system suggested creating a tree with root as trusted and reliable node. The root can allow other trusted nodes to join the system, such as major Internet Service Providers (ISPs). Thus obtaining multiple identities become difficult for the attackers.

2) Eclipse Attack: In Eclipse attack, an attacker can control a substantial part of the neighbors of the target node. All of these compromised nodes work together to fool the targeted nodes. So the overlay nodes unable to forward the messages properly. Thus, the attacker targets most of the nodes of an overlay network. Then it can regulate the majority of genuine peers. Singh et al. [21] have identified that, during an Eclipse attack, the in-degree of the malicious nodes is higher than the average in-degree of nodes in the overlay. And the authors have suggested that genuine nodes should use a threshold for in-degree to counter Eclipse attack. The genuine node should forward to the nodes having less than or equal to the threshold number of in-degree.

3) Routing Table Insertion Attack: In a P2P network, the nodes keep all of its neighbor node id in the routing table. These nodes regularly update the information of the routing table. The attacker can exploit this process to force the peers to insert/ update the compromised node in the routing table of neighbor peers. This attack can cause severe damage to the network as it diverts the route of the message. Baumeister et al. [9] has proposed the mitigation approach specific to Freenet P2P network. They have proposed a route prediction model to find out the path of the message and detect if it diverts the path.

4) Free-riders: There are few users in a network who only uses the resources for their own benefit without offering much to the network. These users are only using the network resources. Solving free riding issue in P2P network is very complex, especially in the network where most peers are free riders. A most suggested solution is detect the free riders and disconnecting such peers from the network. But this solution causes network partitioning. Few methods have proposed a contribution aware P2P network, in which node can use the services proportional to its contribution towards the network. This contribution can be monetary, reciprocity or reputation-based approaches. In monetary based approach, users have to pay to access the services. This is a micropayment-based solution as the service charge is very low. In reciprocity based approach, the peer supervise the behaviors of other nodes and calculates their contribution level. In reputation based approach, the node maintains reputation information about other peers. The peers having better reputation score are offered better services. A peer gets reputation information about a peer through the feedback from peers who have earlier interacted with that peer.

All the solution for the above mention attacks is network specific. The proposed solutions may not be applicable to different networks. In our proposed method, we use the Blockchain technique which can be adapted to any P2P network. In next section, we discuss the framework.

V. The Proposed Blockchain based Security Framework

In this section, we present a Blockchain based security framework for a P2P system. The prime objective of this security frameworks is to prevent variety of attacks including collaborative attacks. In collaborative attack, two or more malicious nodes are collude to target one or a group of genuine nodes in the network. One of such of attacks is the routing table insertion (RTI) attack in which a malicious node (or a group of malicious nodes colluded) targets to insert malicious entries into the routing table of a (victim) node. When victim node wants to send a request, it selects a node from its routing table., thus there is a high probability that the selected node is malicious which could drop the received request or forward to another malicious node. This would decrease the overall network performance.

In the proposed framework, we focus on fair payment and distribution of respect score based management. When a peer

TABLE I
LOOKUP REQUEST TABLE OF NODE B

Request ID	Sender ID	Forward Hop Count	Responder ID	Backward Hop Count
REQID	A	0	C	1

participates in message forwarding, it gets own share of respect score from the service requested peer. A peer sends request message to its neighbor peers. If the neighboring node has the requested file, then it serves the request by sending the file. Otherwise, it forwards the request to its neighbor nodes. Nodes follow this process until the requested file is found or TTL value becomes zero. For a request, if the requested node successfully received the reply, then it performs a transaction of the unit value of respect score which would be equally divided between all the participating nodes of that service. Further, a node can use accumulated respect scores to access the services.

Node joining and Message routing are the two important procedures of a P2P system. In this work we focus on message routing and any existing secure node joining mechanism can be used. We assume that each node has its Public and private keys. For simplicity, we assume that NodeID is its public key and it holds the corresponding private key in secret.

A. Message Routing

In the proposed framework, we have introduced lookup request table (LRT) for nodes. This table keeps an entry for each REQID (id of request message) along with requested nodeID, forward-hop-count, responderID and backward-hop-count. As shown in Fig. 2 node A sends a request message REQID with forward hop count h_u as 0 to few of its neighbors. Node B is one of its neighbors, receives the message. As shown in Table I, node B keeps the REQID, requesting nodeID A in $RequestID$ and $SenderID$ column respectively for the received request. Node B increases hop count and forward the request to its neighbors. Nodes follow this process until the message reaches the node having requested file or TTL becomes zero. As shown in Fig 2, request sent by node A finds the requested file at node D following the path $A \rightarrow B \rightarrow C \rightarrow D$.

Reply message follows deterministic routing. Each node knows the predecessor (from which it has received the request message), and forwards the corresponding reply message. As shown in Fig. 2, node D sends reply to C with the downward hop count h_d as 0 which then increased by C and sends to B. When node B receives reply from node C, it fills $ResponderID$ and $Backward\_Hop\_Count$ column with C and 1 respectively.

After successfully receiving the reply, A sends unit respect score from its account to B which would be distributed between all the participating nodes i.e. B, C and D. This transfer of respect score is treated as transaction.

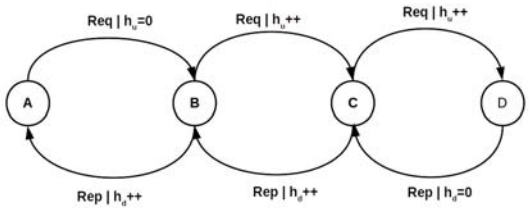

Fig. 2. Lookup request forward

TABLE II
TRANSACTION STRUCTURE

TxnID
PrevTxnID
SenderID
Signature
ReceiverID
MessageID
MessageType
Respect Score

B. Transaction

Respect score is an incentive from the requested node to all nodes participated in reply message forwarding process. The transfer of this respect score ($Resp$) from one node to other is termed as transaction. Node gets sufficient score by participating in multiple processes, which would be used to access the service.

Transaction Structure: The structure of a transaction is presented in Table II. $TxnID$ is current transaction ID and $PrevTxnID$ is ID of previous transaction. While a node transfers a respect score, it sets own-ID and its signature in the corresponding $SenderId$ and $Signature$ field. $ReceiverID$ field sets to the ID of node to which, the sender targets to transfer the respect score. $MessageID$ field is to keep ID of

TABLE III
TRANSACTION FROM A TO B

TxnID: TxnAB
PrevTxnID: —-
SenderID: A
Signature: $Sign_A$
ReceiverID: B
MessageID: REQID
MessageType: Lookup
Respect Score: 1

TABLE IV
TRANSACTION FROM B TO C

TxnID: TxnBC
PrevTxnID: TxnAB
SenderID: B
Signature: Sign_B
ReceiverID: C
MessageID: REQID
MessageType: Lookup
Respect Score: 0.67

the message for which sender has sent respect score through this transaction. A node checks this field to know for which message it has received the score. Because a node may have sent multiple replies to the sender. The P2P network has different types of messages like lookup message, bootstrap message, etc. This is to be mentioned in $MessageType$ field. Sender keeps respect value in $Resepct\_Score$ field, that it sends to other nodes. Transaction (intermediate transaction) at each intermediate node is slightly different from that of the beginning (initial transaction).

Initial Transaction: After receiving the reply, the initiating node A performs a transaction as shown in Table III to transfer a respect score to its neighbor node (B) which will be distributed among all the participating nodes.

The transaction id $TxnID$ field is initialized by $TxnAB$ and $PrevTxnID$ is kept blank as this is the first transaction. The transaction is from A to B, therefore $SenderID$ field is A and $ReceiverID$ is B. A signs the transaction and put it in $Signature$ field. Node A mentions $REQID$ in $MessageID$ field and $Lookup$ in $MessageType$ field. The requested node always sends a unit respect score to the participating node. It is the job of intermediate nodes to calculate their share to keep and pass on the remaining score for others. Thus this transaction deducts a unit respect score from A's account crediting to B and others appropriately.

Intermediate Transaction: Respect score of B increased by a unit through $TxnAB$ transaction. After receiving the respect score, B calculates the score that to be transferred to node C as

$$NewRespectScore = RespectScore - \frac{RespectScore}{h_d + 2}. \quad (1)$$

B uses this new respect score i.e. 0.67 (in this example) in $RespectScore$ field of transaction. The new transaction ID is $TxnBC$ and $PrevTxnID$ is $TxnAB$ as B received the respect score through this transaction. B sets other field of transaction as per its specification and sends to C. Similarly, C calculates the respect score and creates a transaction to send to D.

Inserting Block to chain:

The network is divided into multiple groups or clusters. A node is elected as cluster head (CH) by members of the cluster. The CH keeps the transaction in a pool of transactions when both the sender and receiver of the transaction are members of the cluster. If a node is not a member, then CH sends the transaction to other CHs to add it into its transaction pool. CHs are called as Overlay Blockchain Managers (OBMs) as these nodes collect the transaction and create a new block using the consensus algorithm as discussed in [22]. Then, OBM broadcasts the new block to others. As per the time-based consensus algorithm, each OBM waits for a random period called consensus-period before generating a new block. When OBM receives a block which contains some or all of the transactions of its pool, then it discards those transactions. It creates a new block out of the remaining transaction in the pool after the completion of the consensus-period. Other OBMs can verify the newly created block. An OBM can have direct evidence about other OBM when it correctly verifies a block generated by other OBM. The direct evidence increases the trust between OBMs. OBM can have indirect evidence if it has no direct evidence but other OBMs have verified the block. OBMs calculates the Blockchain utilization after every consensus-period. It is the ratio between the number of transactions created to the total number of transactions added to Blockchain. Since all OBMs has the necessary information about transactions and blocks, the computed ratio should be similar. If the ratio exceeds the predefined limits, then OBMs need to restructure their clusters.

C. Possible Anomaly Attempts in Transaction

A malicious node would attempt to tamper a transaction to forbid a genuine node to have the desired score. Malicious node can be a requesting node or an intermediate node.

1) In the first case (initiating node is selfish and does not wish to perform the transaction or perform a transaction of lesser score). In this framework, such activities can be identified by the next node. If the initiating selfish node A does not send respect score after receiving the reply form B, A would not get the service from D as D would not receive a transaction.
   In another situation, let A, the requesting node sends less than unit respect score (i.e. 0.8) to B. Node B refers its LRT to find the forward hop count for REQID which would be 0. That is A initiated request message and B must get unit respect score from A. Thus B can identify the malicious behavior of A
2) If an intermediate node attempts to cheat by sending a lesser score than that desired, it will be identified by the subsequent node too. Let intermediate node B sends a respect scoreless than it deserves, to next node C. In such case, C can refer to its LRT and calculates from hop counts that three nodes are involved in the process. And it must receive 0.67 scores. When it receives score less than 0.67 from node B for that REQID, C can identify the malicious behavior of B.

When a node identifies such anomaly behavior, it raises the alarm, and the corresponding transaction would be rejected by Blockchain miners.

VI. Security Analysis

The Blockchain system is immutable and publicly available and keeps the respect value of each node. A node gets its portion of respect score for participating in providing network service. The service requesting node have to pay from its respect score to other nodes. These transfer of respect scores are recorded into the Blockchain. If any node does not pay for successful service, other nodes won't address further requests from that node. The defense against the attacks are discussed in following subsections.

a) Defense against Sybil Attack: In Sybil attack, the attackers create number of pseudonymous id in the network. These Ids are controlled by the attackers to change the outcome of a voting. But in our system, the process of creating an id is monitored by registration server. To create a new node Id, it has to pass through the registration process. Then in next step, the newly created node will under a miner. The miner can observe the activities of the node. The activities of these Ids are recorded in Blockchain. So that any node can read the Blockchain records and identify the attackers. This makes the process difficult for the attacker to launch a Sybil attack.

b) Defense against Eclipse Attack: In eclipse attack, a node is surrounded by few malicious nodes. The messages or respect score sent by the genuine node likely to be captured by the malicious nodes which may not forward toward correct route. A genuine node expects the corresponding reply but may not get it. All these activities are recorded by the miner in Blockchain. So that other nodes and miners can verify the activity. And the attackers can easily be identified and the network can be protected from the eclipse attack.

c) Defense against Routing table insertion Attack: In RTI attack, the attacker forcefully inserts malicious entry into routing table of genuine node during routing table updating process. For example in GNUnet, when a genuine node gets a reply message from a new node, it keeps the id of new node into its routing table. Because GNUnet keeps the record of the node from where it received the request message. So that when it receives the reply message then it can forward to correct node. In this case, a node get reply from any node. If that node is not present in routing table then it creates a new entry in routing table. We have modified the lookup request table, which keeps id of node to which it has sent the request. A node knows from which node it may get a reply. It discards the reply messages if received from undesired nodes. Thus our mechanism can defend the RTI attacks.

d) Defense against Free Riding Attack: A free riding attack is an almost impossible task for any attacker. Because each node has limited amount of respect score which it has to earn through participating in network activities. A node can only use for limited period. A node can access the services only if it has more respect score. There is no way to gain respect score other than participating in message forwarding. Our system can successfully defend free riding attacks.

VII. Conclusion

Peer to Peer systems are popularly used in file-sharing applications because of the performance and robustness. In this work, we proposed a Blockchain based security framework to reduce Free-riders. This security framework does not need any central server, at the same time, it avoids the attacks like RTI and Sybil attacks. Time required to complete a file searching is computed and seems to be very small.

ACKNOWLEDGMENT

This work is funded by the E-security Division, Ministry of Electronics and Information Technology, Government of India, through project grant number 12(7)/2015-ESD.

References

[1] M. Steinheimer, U. Trick, W. Fuhrmann, and B. Ghita, "P2p-based community concept for m2m applications," in *Future Generation Communication Technology (FGCT), 2013 Second International Conference on*. IEEE, 2013, pp. 114–119.

[2] Y. Wu, Q. Z. Sheng, and D. Ranasinghe, "P2p object tracking in the internet of things," in *Parallel Processing (ICPP), 2011 International Conference on*. IEEE, 2011, pp. 502–511.

[3] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The second-generation onion router," Naval Research Lab Washington DC, Tech. Rep., 2004.

[4] T. Cholez, I. Chrisment, O. Festor, and G. Doyen, "Detection and mitigation of localized attacks in a widely deployed p2p network," *Peer-to-Peer Networking and Applications*, vol. 6, no. 2, pp. 155–174, 2013.

[5] H. Lin, R. Ma, L. Guo, P. Zhang, and X. Chen, "Conducting routing table poisoning attack in dht networks," in *Communications, Circuits and Systems (ICCCAS), 2010 International Conference on*. IEEE, 2010, pp. 254–258.

[6] M. Castro, P. Druschel, A. Ganesh, A. Rowstron, and D. S. Wallach, "Secure routing for structured peer-to-peer overlay networks," *ACM SIGOPS Operating Systems Review*, vol. 36, no. SI, pp. 299–314, 2002.

[7] Z. Li and X. Chen, "Misusing kademlia protocol to perform ddos attacks," in *Parallel and Distributed Processing with Applications, 2008. ISPA'08. International Symposium on*. IEEE, 2008, pp. 80–86.

[8] C. Rottondi, A. Panzeri, C. Yagne, and G. Verticale, "Mitigation of the eclipse attack in chord overlays," *Procedia Computer Science*, vol. 32, pp. 1115–1120, 2014.

[9] T. Baumeister, Y. Dong, Z. Duan, and G. Tian, "A routing table insertion (rti) attack on freenet," in *Cyber Security (CyberSecurity), 2012 International Conference on*. IEEE, 2012, pp. 8–15.

[10] T. Baumeister, Y. Dong, G. Tian, and Z. Duan, "Using randomized routing to counter routing table insertion attack on freenet," in *Global Communications Conference (GLOBECOM), 2013 IEEE*. IEEE, 2013, pp. 754–759.

[11] S. Pradhan and S. Tripathy, "Cap: collaborative attack on pastry," in *Proceedings of the 10th International Conference on Security of Information and Networks*. ACM, 2017, pp. 319–322.

[12] A. Rowstron and P. Druschel, "Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems," in *IFIP/ACM International Conference on Distributed Systems Platforms and Open Distributed Processing*. Springer, 2001, pp. 329–350.

[13] D. Germanus, S. Roos, T. Strufe, and N. Suri, "Mitigating eclipse attacks in peer-to-peer networks," in *Communications and Network Security (CNS), 2014 IEEE Conference on*. IEEE, 2014, pp. 400–408.

[14] M. Tsehayu, "Secure routing on structured p2p overlayssimulating secure routing on chord dht," *Master's Thesis*, 2015.

[15] W. Wang and W. Zhao, "Model the p2p attack in computer networks," 2015.

[16] M. He, Z. Gong, L. Chen, H. Wang, F. Dai, and Z. Liu, "Securing network coding against pollution attacks in p2p converged ubiquitous networks," *Peer-to-Peer Networking and Applications*, vol. 8, no. 4, pp. 642–650, 2015.

[17] R. B. de Almeida, J. A. M. Natif, A. P. C. da Silva, and A. B. Vieira, "Pollution and whitewashing attacks in a p2p live streaming system: analysis and counter-attack," in *Communications (ICC), 2013 IEEE International Conference on*. IEEE, 2013, pp. 2006–2010.

[18] M. Uddin, R. Alsaqour, and M. Abdelhaq, "Intrusion detection system to detect ddos attack in gnutella hybrid p2p network," *Indian Journal of Science and Technology*, vol. 6, no. 2, pp. 4045–4057, 2013.

[19] X. Meng and D. Liu, "Getrust: A guarantee-based trust model in chord-based p2p networks," *IEEE Transactions on Dependable and Secure Computing*, 2016.

[20] H. Rowaihy, W. Enck, P. McDaniel, and T. La Porta, "Limiting sybil attacks in structured peer-to-peer networks," *Technical report, Network and Security Research Center, Department of Computer Science and Engineering, Pennsylvania State University, USA*, 2005.

[21] A. Singh *et al.*, "Eclipse attacks on overlay networks: Threats and defenses," in *In IEEE INFOCOM*. Citeseer, 2006.

[22] A. Dorri, M. Steger, S. S. Kanhere, and R. Jurdak, "Blockchain: A distributed solution to automotive security and privacy," *IEEE Communications Magazine*, vol. 55, no. 12, pp. 119–125, 2017.