symmetry

MDPI

*Article*

# An Invitation Model Protocol (IMP) for the Bitcoin Asymmetric Lightning Network

**Ali Abdullah and A. M. Mutawa ***

Computer Engineering Department, Kuwait University, P.O. Box 5969, Safat 13060, Kuwait; ali.abdullah@grad.ku.edu.kw

* Correspondence: dr.mutawa@ku.edu.kw; Tel.: +(965)-249-87-160

**Abstract:** The Lightning Network (LN), a second-layer protocol built atop Bitcoin, promises swift, low-cost transactions, thereby addressing blockchain scalability and enhancing user privacy. As the global financial technology landscape evolves, the LN's importance in the future of fintech and the Fourth Industrial Revolution (4IR) becomes increasingly pivotal. The anticipated rise of blockchain-based payments and smart contracts in businesses demands a more agile and secure payment system. However, the LN's early stage raises valid concerns about security and reliability, especially when implemented on a huge asymmetric network such as the Internet, potentially hindering its broader adoption. Malicious nodes could intentionally cause payment failures or initiate attacks, such as DDoS attacks, by overwhelming other nodes in the network with channel-opening requests. As a result, users will be discouraged from using the LN; hence, the technology will become obsolete as individuals will not waste the time and power investment required for using this technology. Addressing these issues, this paper proposes an innovative invitation model protocol (IMP) to reinforce the LN's security and reliability. The IMP creates an exclusive 'Club' within the LN, admitting only nodes verified as honest, thereby bolstering network security and reliability. The protocol empowers Club Founders to expel members exhibiting malicious activities, thereby preserving the invested time, energy, and funds of the network's users. The IMP was rigorously tested using Amazon Web Services Virtual Machines within the Bitcoin and Lightning Network's Testnet environment, which is a highly asymmetric network. The results demonstrated the protocol's efficacy in fulfilling its objectives, marking a significant step towards a safer and more efficient blockchain transaction network. As the blockchain continues to revolutionize the financial sector, implementing robust security measures such as the IMP becomes essential. This research paper introduces a novel approach to enhancing the reliability and security of a Lightning Network (LN), and thus distinguishes itself from the existing literature, by introducing an additional step before establishing or joining such a network. The research underscores the critical role of such protocols in realizing the potential of the LN in powering the next wave of fintech and industrial innovation.

**Keywords:** bitcoin; blockchain; Asymmetric Lightning Network; Club membership; trustless peer-to-peer network



**Citation:** Abdullah, A.; Mutawa, A.M. An Invitation Model Protocol (IMP) for the Bitcoin Asymmetric Lightning Network. *Symmetry* **2023**, *15*, 1273. https://doi.org/10.3390/sym15061273

Academic Editor: Alexander Shelupanov

Received: 17 May 2023
Revised: 12 June 2023
Accepted: 14 June 2023
Published: 16 June 2023



## 1. Introduction

Technology is constantly evolving, resulting in continuous global change in people's lives. For instance, people can use smartphones and tablets to pay for travel and even groceries within seconds while sitting at home. Moreover, technology impacts the world's financial and economic future dramatically. For example, business transactions are continuously moving from cash-based to electronic-based transactions. Further, the worldwide expansion of the Internet in recent years, and its quick acceptance and usage among people, have contributed to simplifying electronic commerce in business environments globally [1]. As a result, electronic payment systems surfaced to replace cash payment systems as busi-

ness owners headed towards e-commerce platforms, which led most businesses worldwide to adopt this technology [2].

E-payment systems have gained a lot of interest and attracted researchers and developers in information technology due to the crucial role these systems have in modern e-commerce. Hence, the importance of e-payment systems has prompted several researchers to perform extensive research that resulted in various views on such systems [3,4].

Bitcoin, which was first introduced in 2008 by Satoshi Nakamoto, is a peer-to-peer electronic cash system based on encryption and a distributed public ledger [5]. According to Kim and Jeong [6], the distributed ledger is also referred to as the blockchain. Additionally, the blockchain is used to record every executed transaction in the system without any need for a trusted third party. This distributed database leverages distributed ledger technology to prevent data tampering and fabrication. In addition, the researchers stated that the blockchain offers several advantages over a typical centralized system, including efficiency, security, resilience, and transparency. In the literature, many studies have been performed to implement blockchain in different daily life applications, such as: securing cloud-based peer-to-peer transactions [7]; manufacturing processes [8]; the Internet of Vehicles [9]; higher education certificate authentication [10]; and many more.

Still, the Bitcoin network has a significant limitation that many researchers are currently trying to overcome, which is the scalability of the blockchain. Specifically, the Bitcoin blockchain allows only a limited number of transactions to be validated through the network, ranging from three to seven transactions per second. Therefore, numerous researchers have sought to resolve the scalability issue through various methods. For example, El Azzaou et al. [11] presented a lightweight scalable authentication solution based on blockchain to secure videoconferences. The researchers employed a time-based consensus approach to decrease latency, mining processing cost, and boost.

Moreover, one of the major solutions that have been developed is payment channels, such as the Lightning Network (LN). The LN is a second-layer technology built on top of the Bitcoin blockchain that provides users with fast payment channels held off-chain in a trustless environment [6]. The Lightning Network is a routed network proposed for bidirectional payment channels that are end-to-end connected [12]. Members of an LN can route payments through numerous channels without needing trusted intermediaries [13]. Moreover, payment transactions can be executed off-chain with minimal fees through an LN. As a result, not all bitcoin transactions are recorded in the main blockchain [14]. Thus, the LN is less expensive, faster, and provides additional privacy for transactions that are not visible on the public blockchain. In a Lightning Network, a channel between two parties is established temporarily for a period, during which each party locks in an identical amount of money as collateral. Furthermore, both parties can exchange money back-and-forth during the channel time with just the netting transaction being validated and preserved in the Bitcoin blockchain [15]. Additionally, a malicious party that fails to update the channel balance will forfeit the collateral (deposited when the channel was first opened) to the other party. LN provides several benefits like anonymity, speed, and trustless transactions [16], tackling Bitcoin's scalability issue without altering network protocols. Some challenges exist, such as potential transaction delays, dependency on the main blockchain, and the complexity of cryptographic network protocols.

Various studies have been conducted to improve the LN. Valente et al. [17] analyzed the LN's underlying structure and its impact on the system, focusing on the network's liquidity and resilience. However, the paper lacked certain details and clear contribution statements. Nowostawski et al. [18] highlighted the potential for leaked data from the off-chain network to the blockchain, which could reveal user information. They also proposed methods to identify unique transactions in the LN. Conoscenti et al. [19] addressed three issues that need resolution in LN technology: the development of Hashed Timelock Contract (HTLC) payments, estimating optimization effects, and predicting profits from investing in a network hub. Seres et al. [20] assessed LN's topology to enhance its security, demonstrating its robustness against random and targeted attacks. Rohrer et al. [21] analyzed

LN's topology and proposed certain attacks that could disrupt the LN. Pérez-Sola et al. [22] described an attack that could lower a node's channel capacity, limiting its network communication. Zhang et al. [23] introduced a distributed simulator that reduced transaction fees along a payment channel, though it faced constraints regarding the success rate and average transaction value.

Many researchers suggested considering the balance of a channel in an LN as crucial information [24–27]. This would enable users to find a viable payment path quickly [28,29]. Other research explored the balance between privacy and efficiency in payment channel networks. For instance, Tang et al. [30] used noise addition to increase privacy, albeit at efficiency's expense. Integrating blockchain and LN technology with the Internet of Things (IoT) for instant payment applications is another research focus [31–37]. Furthermore, a solution based on a game-theoretic model to analyze griefing attacks in Hash Time-Locked Contracts (HTLCs) called HTLC-GPZ was presnted to provides improved protection against such attacks [38].

Other researchers proposed a study that focused he privacy implications and vulnerabilities present in the Lightning Network. The authors likely conduct a thorough analysis of the network's privacy features, focusing on aspects such as transaction linkability, address reuse, and potential deanonymization attacks. making it more difficult for attackers to identify and target specific payment channels [39].

Given the rise in Bitcoin use around the world, it is expected that the use of the LN for point-of-sale (POS) systems and small payment platforms will grow at an exponential rate. The LN is excellent for small quick payments, such as those made at coffee shops or grocery stores. It avoids the inconvenience of having to wait 10 minutes for blockchain approval at the POS. Because of this, the need for a safer LN is growing. To understand how payments work in the LN, consider a simple case with only four nodes: A, B, C, and D. In this situation, shown in Figure 1, these nodes have opened symmetrical bidirectional payment channels in pairs, represented by blue arrows. Further, each channel is funded with 2 bitcoin by each participant. Consequently, the channel balance for each of the payment channels in the network is 4 bitcoin.

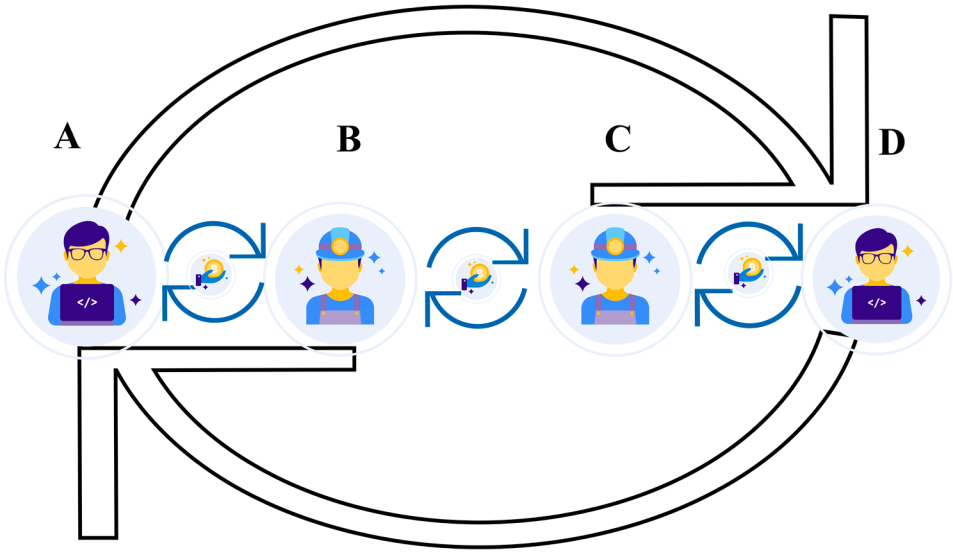


**Figure 1.** Standard LN payment transaction from node A to node D through nodes B and C.

The Lightning Network logic states that node A does not need a direct payment channel with node D to send any bitcoin amount. Instead, node A will use paths A, B, C, and D to reach node D indirectly and send the intended bitcoin amount, without spending additional funds to open a new channel with node D. For instance, if node A intends to send 1 BTC to node D, A will discover the route between payment channels (A through D) and connect to node D through the Internet to receive a hash value (H) of a secret key (R) that is

owned by node D. Later, A will construct a Hashed Timelock Contract (HTLC); this virtual channel is represented by a white arrow in Figure 1, which is a type of smart contract that permits participants to commit funds to a redeemable secret key with a time limit and is utilized in both bidirectional and routed payment channels. The constructed HTLC would include a refund after a 9-block timeout for the amount of 1.002, payable to the hash H. The extra 0.002 BTC amount offered by A is used to pay to the intermediaries B and C and is deducted from the channel balance of node A as a fee to forward the payment. Afterwards, node A offers the HTLC to node B, which indicates that B will receive 1.002 BTC from A only if B can provide the secret key (R). Otherwise, a transaction timeout will occur after 9 blocks. Node A will get the amount back. Afterwards, node B will repeat the same process with node C. Still, instead of committing 1.002 BTC of the channel balance and the same 9-block timeout, node B will create another HTLC that is slightly different to the one offered by node A. More precisely, node B will construct an HTLC with 1.001 BTC and an 8-block timeout for a refund payable to the hash H, taking 0.001 BTC as a commission for forwarding the payment to the next node down the same path. Then, node C will repeat the same process, constructing and committing an HTLC with 1.000 BTC and a 7-block timeout for node D. Finally, node D will present the secret R to node C to claim the 1 BTC from the HTLC that node C offered and add the received amount to the channel balance. Moving back along the same route, each node sends the secret key (R) to the previous node, claims the HTLC committed previously, and updates the channel balance. In the end, node A will have paid node D an amount of 1 BTC without opening a direct payment channel and spending more funds. Instead, node A is only required to pay a minimal amount of BTC to intermediate nodes that can forward the payment to the intended destination. Figure 2 illustrates a ladder diagram that illustrates the step-by-step process explained above.

Nonetheless, Lightning Network (LN) technology is still in its infancy. As a result, academics have performed several studies to develop this technology in various ways. However, other LN vulnerabilities have not been explored in the literature; in particular, rogue nodes intending to perform specific attacks that might undermine network security and reliability. The literature, however, allowed for any LN node to join the LN freely and instantaneously. As a result, the danger level of LN technology will grow as malicious nodes can join the network and conduct attacks, such as intentional cause-of-payment failures or the initiation of DDoS attacks by overwhelming other nodes in the network with channel-opening requests. As a result, users will be discouraged from using the LN; hence, the technology will become obsolete as individuals will not waste the time and power investment required for using this technology.

In this paper, we propose an innovative invitation model protocol (IMP) to improve the security and reliability of Lightning Network (LN) technology. The IMP works by establishing a unique 'Club' within the LN that only admits nodes confirmed to be honest and not malicious, thereby enhancing the network's overall security and reliability. Unlike existing research, this paper is the first of its kind that proposes extra metrics for nodes before joining or establishing an LN, which is especially important for trustless networks. The remainder of this paper is organized as follows: materials and methods are discussed next, followed by simulation and results in Section 3; finally, conclusions and future work are discussed in Section 4.

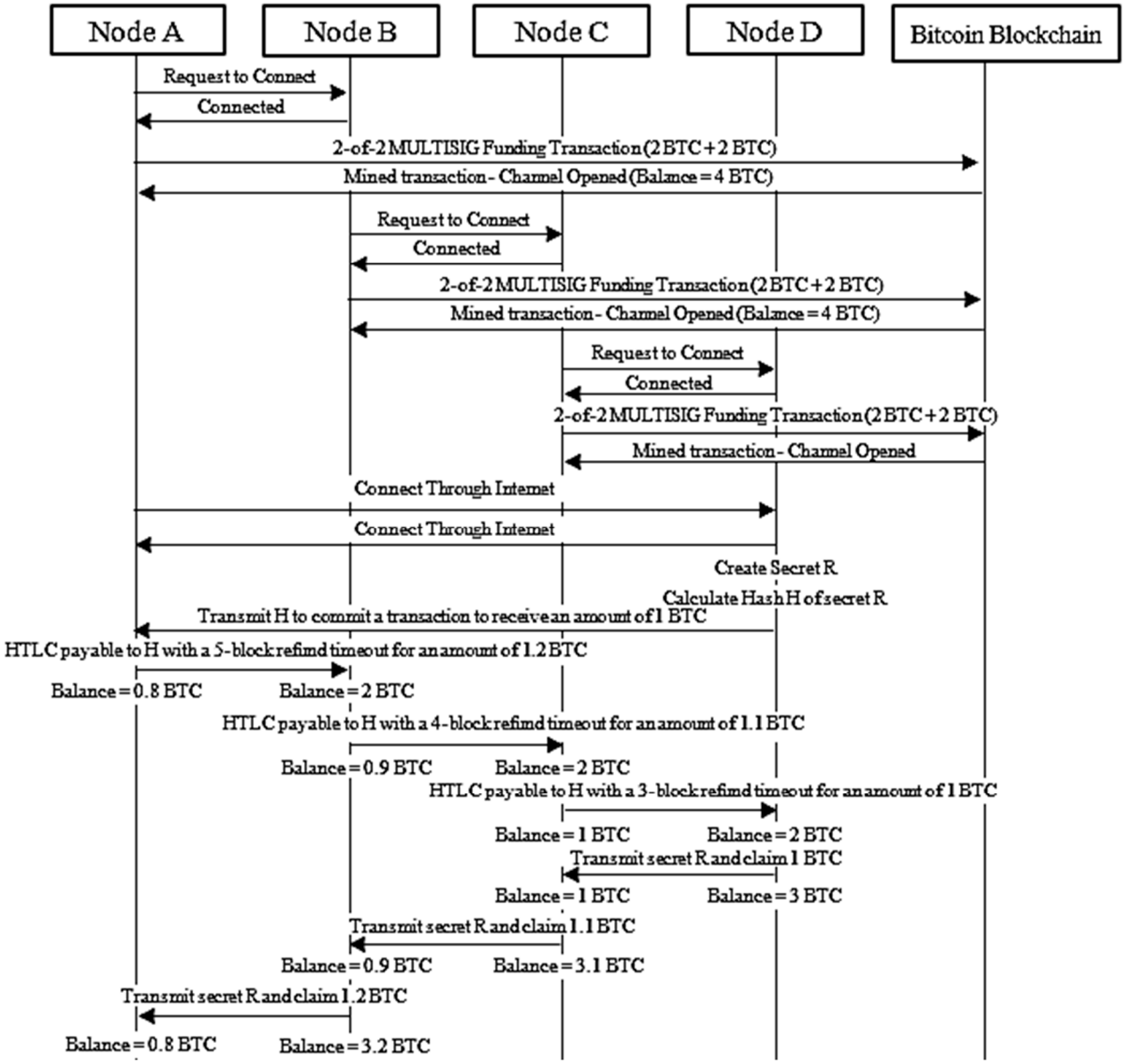


**Figure 2.** Ladder diagram for a Lightning Network transaction to transfer funds between node A and node D.

## 2. Materials and Methods

We introduce an invitation model protocol (IMP) to enhance the security and reliability of LN technology as a solution to transaction failures due to intentionally malicious node failures or bad internet connections. The IMP will be used to add an extra security layer before establishing a Lightning Network between the network nodes. Consequently, this additional layer will improve network security and reliability through ensuring that only authorized honest nodes are permitted to participate in an LN and that connections from unauthorized nodes are denied. Therefore, threats such as a DDoS attack, which overwhelms a node by sending many channel-opening requests, can be prevented since malicious and unauthorized nodes cannot communicate and open a channel with other LN nodes.

In addition, the model states that before creating a Lightning Network, nodes must apply a series of protocols to join an exclusive Lightning Network called a Club, which is only accessible by honest nodes in the network. Moreover, the IMP consists of ten fundamental protocols that must be applied before starting a more secure and reliable Club.

The model requires that before starting a Club, nodes need to employ the use of Bitcoin blockchain through initiating a few Bitcoin payment transactions so that all the information of the Club and Club Members are public and recorded in the Bitcoin blockchain. Therefore, the data is accessible and viewable to all the nodes in the network.

Furthermore, a Club has three main types of nodes: Club Founder (CF), Founding Member (FM), and Club Member (CM). Nevertheless, an Applicant Node (AN) will not be considered a Club Member unless this node was invited by a current FM or CM in a Club. In addition, a Club Founder owns eight public Bitcoin addresses that are fixed and dedicated to the Club. The addresses are myClub, Club Founder Member ID, Member Label, Member Expulsion Label, IP1, IP2, IP3, and IP4. The primary reason for obtaining Bitcoin addresses IP1, IP2, IP3, and IP4 is to encapsulate them into a single IP address, formatted as IP1.IP2.IP3.IP4. To illustrate, if a node's IP is 3.234.215.167, then we would denote IP1 as 3, IP2 as 234, IP3 as 215, and IP4 as 167.

Any node that wants to join a Club needs to give a Bitcoin address. This address is then used as its Member ID in that Club. Whether you're a Founding Member or a regular Club Member, you will have a unique Member ID for the specific Club. Before joining, the node applying (Applicant Node) must also provide a Bitcoin Address (ID). If they're accepted into the Club, this ID becomes their Member ID.

### *2.1. Problem Formulation*

The IMP states that the CFs should elect one or more nodes in the network to be FM(s) of the Club. Moreover, FMs are not required to pay a Membership Fee. On the other hand, an FM can invite an Applicant Node (AN) to the Club that is required to pay the Membership Fee. Additionally, the CFs would make a list, called the Member List, that contains all the current Members of a Club (FMs and CMs). Moreover, the CFs need to update the Member List every ten minutes as this is the average time it takes for a block to be validated and added to the blockchain.

Furthermore, the IMP states that the CFs will expel any malicious Member from the Club that exhibits malicious behavior, such as deliberate causing of multiple consecutive payment transaction failures. After Expulsion of a malicious Member, the CFs will create a list of Expelled Members that contains the Member ID for those malicious Members. In addition, the CFs will update the Expelled Members List every ten minutes. Moreover, the CFs will compare the Members List with the Expelled Members List and remove all the expelled Members that exist in the Members List based on the most recent block height. The following eight equations show how the proposed IMP will operate:

$$\mathrm{CF_i}\ (\mathrm{ML_i}),\ \mathrm{I} \in \mathrm{N} \tag{1}$$

where:

- N is the total number of CFs in the system.

$$\mathrm{EML_i}\ (\mathrm{MN_1}, \mathrm{MN_2}, \ldots, \mathrm{MN_k}),\ \mathrm{i} \in \mathrm{K} \tag{2}$$

where:

- MN is a malicious node.
- K is the total number of malicious nodes that were expelled from a Club.

To calculate the total number of blocks up to the most recent block:

$$\mathrm{BTC}(\mathrm{B_1}, \mathrm{B_2}, \mathrm{B_3}, \cdots, \mathrm{B_h}) \tag{3}$$

$$\mathrm{h} = 6(8640(\mathrm{Y} - 2009) + 720\mathrm{M} + 24\mathrm{D} + \mathrm{H}),$$

where:

- B is a registered block in the blockchain.
- Y is current year.

- M is current month.
- D is current day.

Each block containing the total number of confirmed transactions:

$$\mathrm{Bi} = (\mathrm{T}_1, \mathrm{T}_2, \mathrm{T}_3, \cdots, \mathrm{T}_n), n \in \mathrm{N} \tag{4}$$

where:

- T is a confirmed bitcoin transaction in a block.

Transactions that contain a Bitcoin Address and an amount paid to that Address:

$$\mathrm{Ti}(\mathrm{Address}, \mathrm{Value}), \tag{5}$$

where:

- Address is a Bitcoin address.
- Value is a bitcoin amount.

Member List that contains all Founding Members and Club Members:

$$\mathrm{ML}_i = (\mathrm{FM}_1, \mathrm{FM}_1, \cdots, \mathrm{FM}_f, \mathrm{CM}_1, \mathrm{CM}_2, \cdots, \mathrm{CM}_c), \tag{6}$$

where:

- ML is the Member List of a Club.
- i is the ith Club in the system.
- FM is Founding Member.
- f is the total number of Founding Members in a Club.
- CM is Club Member.
- c is the total number of Club Members in a Club.

The established Club with the Member List and Expulsion List owned by the Club Founder:

$$\mathrm{Club}(i, h) = \begin{cases} \lambda\left(\mathrm{CF}_j, \mathrm{ML}_{i,h}\right), i, j \in \mathrm{N} \\ \mathrm{EML}_{i,h} \end{cases} \tag{7}$$

where:

$$\mathrm{ML}_{i,h} = (\mathrm{FM}_1, \mathrm{FM}_1, \cdots, \mathrm{FM}_f, \mathrm{CM}_1, \mathrm{CM}_2, \cdots, \mathrm{CM}_c) \text{ at block h}$$

- $\mathrm{ML}_{i,h}$ is the Member List that contains all the members of a Club in a specific block h.

Invitation transaction that states that elected FMs will not have to pay a Membership Fee, while invited Applicant Nodes will need to pay the fee to become Club Members:

$$\lambda(\mathrm{X}, (\mathrm{L}_1, \mathrm{L}_2, \cdots, \mathrm{L}_n)) = \begin{cases} \forall_i \mathrm{invite}(\mathrm{X}, \mathrm{L}_i) \text{ and accept } (\mathrm{L}_i, \mathrm{X}) \\ \quad \mathrm{Status}(\mathrm{L}_i, \mathrm{FM}, \mathrm{Value} = 0) \\ \quad \mathrm{Status}(\mathrm{L}_i, \mathrm{CM}, \mathrm{Value} = \varepsilon) \end{cases} \tag{8}$$

where:

- $\forall_i$ is the invitation transaction Sponsor.
- X is the Sponsor.
- $\mathrm{L}_i$ is the invited Member.
- $\varepsilon$ is the Membership Fee.

### *2.2. FM Invitations and Acceptance Protocols*

The selected FM should reply to the invitation with a Bitcoin payment transaction that includes the IP address so that the Club Founder would grant access to this Founding Member through the firewall.

The transaction will be signed by a Founding Member and be of the following form:

Founding Member initiates and signs a bitcoin transaction as follows:

a. Pay the Bitcoin address myClub an amount of dust value
b. Pay the Bitcoin address Member ID an amount of dust value
c. Pay the Bitcoin address IP1 an amount of Bitcoin that is equal to concatenating 0.000 with the first segment of the FM IP address (e.g., if IP is 3.234.215.167, then pay IP1 0.0003 BTC).
d. Pay the Bitcoin address IP2 an amount of Bitcoin that is equal to concatenating 0.000 with the second segment of the FM IP address (e.g., if IP is 3.234.215.167, then pay IP2 0.000234 BTC).
e. Pay the Bitcoin address IP3 an amount of Bitcoin that is equal of concatenating 0.000 with the third segment of the FM IP address (e.g., if IP is 3.234.215.167, then pay IP3 0.000215 BTC).
f. Pay the Bitcoin address IP4 an amount of Bitcoin that is equal of concatenating 0.000 with the fourth segment of the FM IP address (e.g., if IP is 3.234.215.167, then pay IP4 0.000167 BTC).

Then, the Club Founder will check the blockchain, take the information of the selected FMs that accepted the Club invitations, and make a list of Club Members that initiated an acceptance transaction. Finally, actual FMs will be given access through the firewall to connect and open Lightning Network payment channels with each other.

Figure 3 represents the pseudocode for the process of establishing a Club and electing node(s) A(i) to be an FM through an invitation initiated by the CF as well as the invitation's acceptance by the elected FM.

```
1. Start
2. #Each Club Founder assigns a unique member ID to his Club
   #(CF(i).memberID)
3. CF(i).clubAddress = myClub
4. #Send invitation for each elected FM by CF
5.    for j=1 to e do    #e is number of elected FM by CF
6.    {
7.      #CF initiates a Bitcoin transaction to invite node
        #A(j)to his Club
8.       bitcoin-cli sendmany{myClub:dustValue,
                             CF(i).memberID:dustValue,
                             memberLabel:dustValue,
                             A(j).memberID:lowAmount}
9.      #A checks blockchain for own ID in the past n blocks
        #and accepts to be an FM by initiating a Bitcoin
        #transaction as a reply of acceptance
10.       for index = currentBlock - n to currentBlock do
11.       {
12.      #If a block at[index] contains both A(j).memberID and
         #memberLabel, then node A(j) has been invited and
         #should reply with its IP address
13.        if (block[index].transaction contains A(j).memberID
               && memberLabel)
14.        then
15.         bitcoin-cli sendmany{CF(i).clubAddress:dustValue,
                                 A(j).memberID:dustValue,
                                 IP1 :0.000ip1,
                                 IP2 :0.000ip2,
                                 IP3 :0.000ip3,
                                 IP4 :0.000ip4}
16.        end if
17.       } end for
18.      } end for
19.      #CF checks blockchain for A's invitation acceptance
         #through searching for A's member ID with IP
         #addresses in the same transaction
20.      #For each elected node A by CF(i) validates the IP
         #address then grant access through the firewall
21.      for j = 1 to e do
22.       {
23.         for index = currentBlock -n to currentBlock do
24.         {
25.          if (block[index].transaction contains A.memberID
                  && IP1 && IP2 && IP3 && IP4)
26.              #set A as an FM in Club CF(i)
27.              then CF(i).FM(j) = A(j)
28.          end if
29.         } end for
30.       } end for
31.      End
```

**Figure 3.** FM Invitation and Invitation Acceptance.

The CF will initiate a bitcoin transaction to be registered in the main blockchain as an invitation. Moreover, the invitation transaction will start by bitcoin-cli sendmany and takes a Bitcoin address and a value as a parameter, such as bitcoin-cli sendmany (address:value). In addition, a transaction can have as many addresses as needed, separated by a comma. For instance, a bitcoin transaction can be written as follows:

```
bitcoin-cli sendmany {address1 :value1, address2 :value2, ...,
                      addressn :valuen}
```

### 2.3. New Member Invitations

Applicant Nodes (ANs) that intend to be Club members must be invited by a Sponsor. A Sponsor can be either a current FM or a current CM in the Club. Moreover, the IMP gives a chance to all ANs that are honest and referenced by current Club members (Sponsors). Thus, a Club Founder only accepts ANs that were invited by Sponsors. In fact, when a Sponsor invites an Applicant Node, the CF will understand that this referenced AN is honest and can be trusted to join the Club. Accordingly, the AN sends the node ID to be used as a Member ID in the Club to the Sponsor via the Internet. Next, an invitation to the AN will be a bitcoin transaction registered on the Bitcoin blockchain. The invitation transaction will be similar to the one sent to the FM by the CF (as explained in the Club Establishment protocol above) except for the Member Label, which will be excluded from the transaction. Consequently, the Sponsor will construct an invitation transaction that includes the Club address, Sponsor Member ID, and AN ID.

Afterwards, to show that the invitation was accepted, the AN must pay a Membership Fee to both the CF and the Sponsor. The Membership Fee is decided by the CF. However, the fee resembles a minor payment, such as the price of a cup of coffee. Accordingly, the AN will initiate and sign a bitcoin transaction that will do the following:

a. Pay Club Address myClub an amount of coffee price.
b. Pay Sponsor Member ID an amount of coffee price.
c. Pay Applicant Node ID (to be used as a Member ID in the Club) an amount of dust value.

Consequently, a Club Founder will keep the received Membership Fee (coffee price). A Sponsor must refund that payment to the invited AN, which shows the CF that this Applicant Node is honest and can be labeled as a CM. The refund transaction will be similar to the invitation transaction sent to the FM by the CF (Club Establishment). Still, the Sponsor will replace the CF Member ID with the Sponsor Member ID. In addition, the Sponsor will pay the AN Member ID an amount of coffee price. The AN can be recognized by the address receiving an amount of a coffee price (e.g., 0.0000111 BTC) through a payment transaction registered in the Bitcoin blockchain and initiated by a Sponsor node and a Member Label in the same transaction. After viewing the blockchain and confirming the success of all trades discussed previously, a CF will update the Club member list by adding the AN as a CM.

As a result, the new CM should reply with a Bitcoin payment transaction that includes its IP address in a similar construction to the transaction that contains the FM IP. Therefore, the CF would grant access to the new CM through the firewall and add the new CM to the Member List. Hence, all Club members are known and can start an exclusive Lightning Network by making payment channels and exchanging as many payments as needed. Table 1 shows an example of a Member List owned by the Club Founder. In contrast, Table 2 shows an example of a list of approved IP addresses of myClub Members. Furthermore, Figure 4 shows the pseudocode for the process of inviting new members to a Club.

**Table 1.** Example of a Members List Owned by a CF.

| Club Address | Member Label | New Member ID | Sponsor | Block Height |
|---|---|---|---|---|
| myClub | Member Label | Node 1 Member ID | CF | B2 |
| myClub | Member Label | Node 2 Member ID | FM | B8 |
| myClub | Member Label | Node 3 Member ID | CM | B14 |

**Table 2.** List of Approved IP Addresses of myClub Members.

| Club Address | Member Label | New Member | Sponsor | Block Height | | | | |
|---|---|---|---|---|---|---|---|---|
| myClub | Member Label | Node 1 Member ID | CF | B2 | | | | |
| myClub | Member Label | Node 2 Member ID | FM | B8 | | | | |
| myClub | Member Label | Node 3 Member ID | CM | B14 | | | | |
| | | | | **IP1 Label** | | | | |
| | | | | **IP2 Label** | | | | |
| | | | | **IP3 Label** | | | | |
| | | | | **IP4 Label** | | | | |
| **Member** | **Value 1** | **Value 2** | **Value 3** | **Value 4** | **IP1** | **IP2** | **IP3** | **IP4** | **IP** |
| Node 1 Member ID | 0.000003 | 0.000234 | 0.000215 | 0.000167 | 3 | 234 | 215 | 167 | 3.234.215.167 |
| Node 2 Member ID | 0.0001 | 0.000002 | 0.000156 | 0.000216 | 100 | 2 | 156 | 216 | 100.2.156.216 |
| Node 3 Member ID | 0.00035 | 0.000174 | 0.000182 | 0.000237 | 35 | 174 | 182 | 237 | 35.147.182.237 |

```
1. Start
2. #Inviting an applicant node AN sponsored by either FM or CM
3. bitcoin-cli sendmany {myClub :dustValue,
4.                       Sponsor(i).memberID :dusValue,
5.                       AN.memberID :dustValue}
6.  #AN will check the blockchain for own ID in the past n blocks and
    #pays Membership Fee through a Bitcoin transaction
7.     for index = currentBlock to currentBlock − n do
8.     {
9.       #If a block at [index] contains both AN.memberID and
          memberLabel, then node AN has been invited and should
          reply with its IP address
10.      if (block[index].transaction contains AN.memberID
             && block[index].transaction contains memberLabel)
11.      then
12.         bitcoin-cli sendmany {myClub :CoffeePrice,
                                  Sponsor(i).memberID :CoffeePrice,
                                  AN.memberID :dustValue}
13.      end if
14.      } end for
15.      #The sponsor checks the blockchain whether membership fee
          #has been paid
16.      #If membership fee was paid, the sponsor will refund 50% of
          #the fee to the AN
17.      for index = currentBlock to currentBlock − n do
18.      {
19.       if (block[index].transaction contains AN.memberID
20.                && myClub :CoffeePrice && Sponsor(i).memberID
                    :CoffeePrice)
21.       then
22.           bitcoin-cli sendmany {myClub :dustValue,
23.                                 Sponsor(i).memberID :dustValue,
24.                                 Member Label :dustValue,
25.                                 AN.memberID :CoffeePrice}
26.       end if
27.      } end for
32.      for index = currentBlock to currentBlock − n do
33.       {
34.        #If a block at [index] contains both AN memberID paid a
           #Coffee Price and memberLabel, then node AN has become a
           #CM and should reply with its IP address
35.        if(block[index].transaction contains
               AN.memberID:CofeePrice && block[index].transaction
               contains memberLabel)
36.        then
37.            bitcoin-cli sendmany{CF(i).clubAddress:dust value,
                                    AN.memberID:dust value,
                                    IP1 :0.000ip1,
                                    IP2 :0.000ip2,
                                    IP3 :0.000ip3,
                                    IP4 :0.000ip4}
38.        end if
39.        } end for
28.      End
```

**Figure 4.** AN Invitation by a Sponsor (FM or CM).

### *2.4. Malicious Nodes Expulsion*

A node is considered malicious if it does not adhere to the rules and conditions of a Club or causes an (n) number of failures in the Lightning Network. For example, a refund transaction to an Applicant Node signed by a Sponsor is not registered in the Bitcoin blockchain. Other examples of malicious behavior could be an intentional causing of transaction timeouts or a node being non-responsive. Thus, in any instance of malicious behavior perpetrated by an FM, a CF will expel the malicious node and elect another CM to be an FM. Furthermore, if a Sponsor exhibits malicious behavior, such as not refunding 50% of the Membership Fee to an Applicant Node, a CF will expel that Sponsor and Label that Applicant Node as a Club Member.

The expulsion transaction will only be constructed and signed by the CF(s). Moreover, the expulsion transaction will be similar to an FM invitation transaction except that the Member Label will be replaced by the Expulsion Label, as follows:

a. Pay to the address *myClub* an amount of *dust value*.
b. Pay to CF Member ID an amount of *dust value*.
c. Pay Member Expulsion Label an amount of *dust value*.
d. Tag the malicious Member by paying their ID a small pre-set amount (i.e., 0.0000112 BTC).

Figure 5 shows the pseudocode for the expulsion transaction that will be initiated by a CF(i) against a malicious Member (FM or CM). Figure 6 shows the abstract of the ten protocols of the proposed IMP. In contrast, Figure 7 illustrates a graphical representation of the workflow of the proposed IMP with an example of malicious node expulsion. Figure 8 represents a ladder diagram of the proposed IMP, including a case of malicious CM expulsion by the CF after causing *three* consecutive LN payment transaction failures.

```
1. Start
2. #If a current Member (CM or FM) causes m transaction
    #failures, then Member is malicious and will be
    #expelled by CF(i)through a Bitcoin transaction
3. if (member.txFailures >= m)
4. then
5.     bitcoin-cli sendmany {myClub :dustValue,
                             CF(i).memberID :dustValue,
                             expulsionLabel :dustValue,
                             Bad_CM.memberID :lowAmount}
6. endif
7. #In case of a malicious FM, CF(i) will elect another
   node to be an FM
8. End
```

**Figure 5.** Pseudocode for expulsion of a malicious Member (FM or CM).

Additionally, the network nodes can identify a malicious CM by viewing the blockchain and finding a transaction that includes the CF Member ID, Expulsion Label, and CM Member ID receiving a set BTC amount (e.g., 0.0000112 BTC). Table 3 shows an example of a List of Expelled Members of myClub owned by the Club Founder.

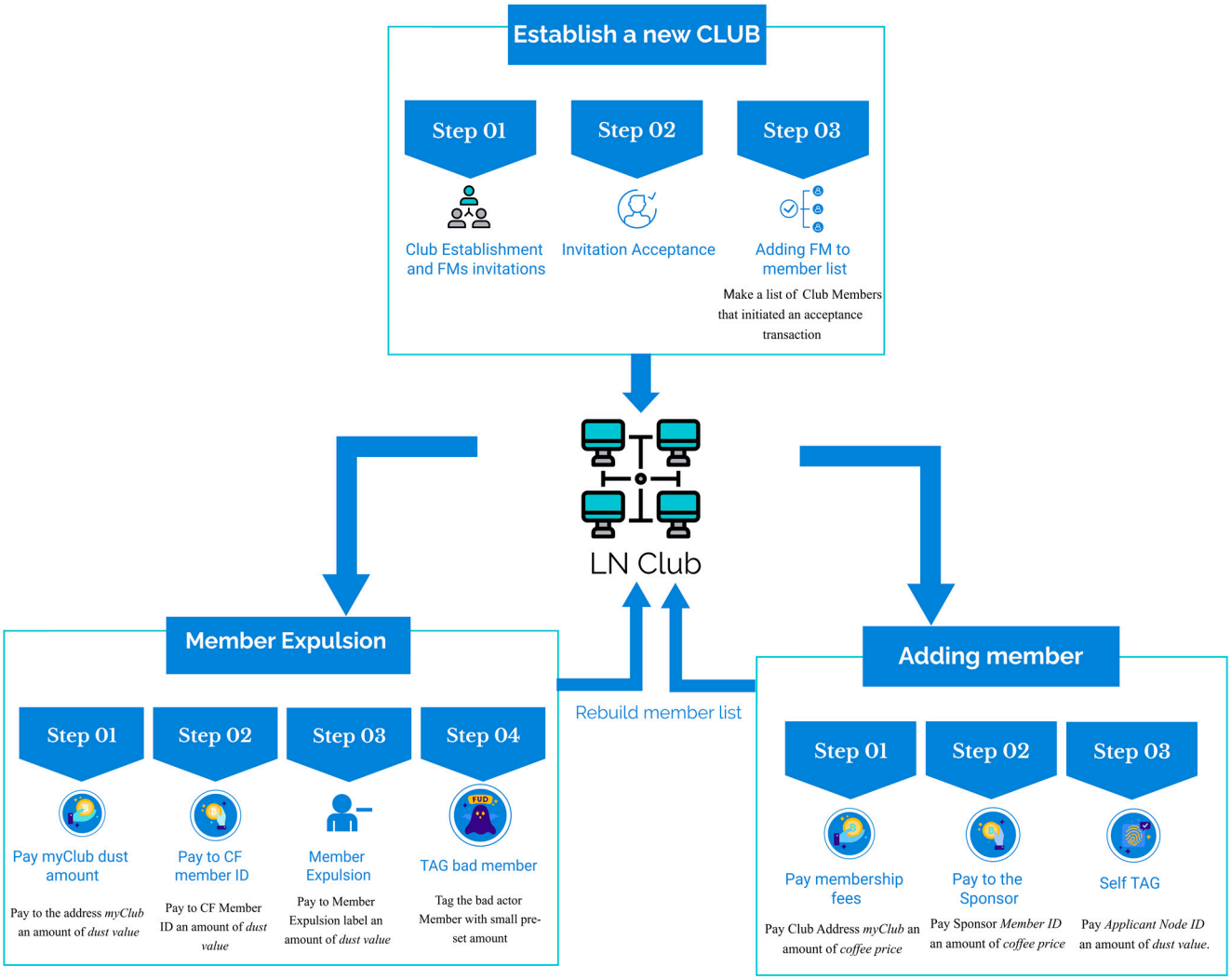


**Figure 6.** The proposed IMP.

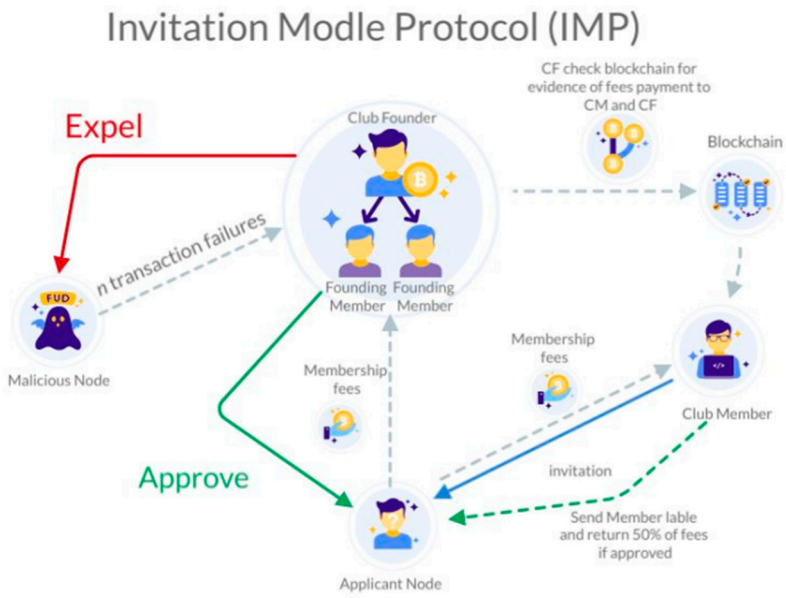


**Figure 7.** An example workflow of the IMP with a malicious node.

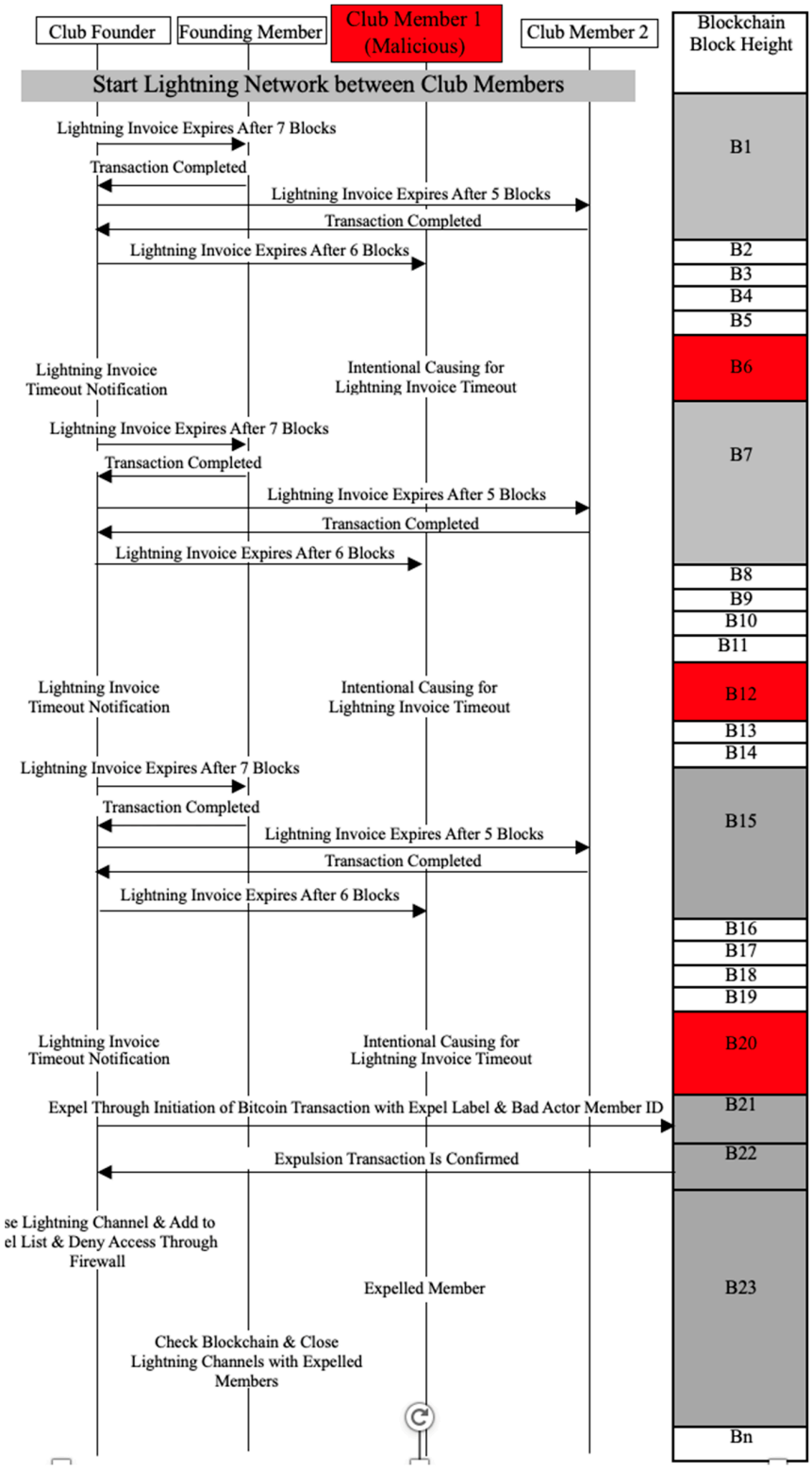


**Figure 8.** Timeline of the process of expelling malicious node CM1.

**Table 3.** List of Expelled Members of myClub.

| myClub | Expulsion Label | Club Member | Sponsor | CF | Block Height |
|---|---|---|---|---|---|
| myClub | Expulsion Label | CM 1 Member ID | Sponsor Member ID | CF Member ID | B23 |

### *2.5. Removal of Expelled Members from Member List*

A CF will take several measures to build a list of honest CMs in a Club. A CF of a Club will follow multiple steps to remove previously expelled CMs from that Club. First, a CF will create a list of Club Members after checking the Bitcoin blockchain for transactions that include the Club Address and Member Label. Second, the CF will save all new Club Members in separate records. The Club Address, Member Label, Club Member ID, and Block Height in which AN was labeled as a CM are all included in each record. Then, after examining the Bitcoin blockchain for transactions that include the Club Address and the Expulsion Label, the CF will create a list of Expelled Members. The list of Expelled Members will contain a record for each member that contains the Club Address, Expulsion Label, Club Member ID, and Block Height in which a Club Member was Labeled as an Expelled Member. Afterwards, each record in the Expelled Members list will be compared to the Club Members list by the CF. As a result, if a record in the Expelled Members list has a greater Block Height than the record in the Club Members list, the CF will remove that CM record from the Club Member list and create a new one that contains current CMs only. Moreover, after generating the final CM list that includes only the honest nodes, each record in the list will be parsed by the CF, who will look up the appropriate IP address in the Bitcoin blockchain. As a result, the CF will have a list of IP addresses of the current CMs along with other default IP addresses for admin nodes (other CFs). Therefore, The CF will go through the IP Addresses list and grant access to CMs through the firewall, allowing them to start Lightning Channels with the CF. Thus, targeted attacks by malicious nodes can be avoided as the firewall would not allow non-permitted nodes to connect to the CF. Moreover, malicious nodes cannot reach the CF to send channel-opening requests. Table 4 shows an example of the Member List of myClub after removal of Expelled Members.

**Table 4.** Member List of myClub after removal of Expelled Members.

| myClub | Member Label | Member ID | Sponsor | Block Height |
|---|---|---|---|---|
| My Club Address | Member Label | FM Member ID | CF | B2 |
| My Club Address | Member Label | CM 2 Member ID | FM | B8 |

Nevertheless, after the expulsion of a malicious node, CF and CMs will not block any node in the network. For instance, if a malicious node is expelled from a Club, it can rejoin on the condition that a Membership Fee is paid again to ensure Club membership. There are two reasons why the no-block policy is used. First, malicious nodes can easily modify their IDs and Bitcoin addresses to impersonate another Applicant Node requesting to be a CM. As a result, the CF and CMs cannot distinguish the malicious node that was earlier expelled. However, to rejoin the Club, the node will always have to pay a Membership Fee. Hence, every time a malicious node tries to rejoin a Club, more funds will be paid, making them the only losers in the network.

The second reason for enforcing the no-block policy is that some nodes may experience a temporarily bad Internet connection, resulting in LN payment transaction failures. Those CMs would be considered malicious and expelled. Thus, blocking those nodes would be unjust. Yet, expelled nodes must still pay a Membership Fee to rejoin a Club, which improves the network reliability.

### *2.6. Rebuilding the Member List*

The CF will repeat the process of removing Expelled Members from the Member List every ten minutes, which is the average time it takes for a new block to be added to the blockchain. Therefore, the Member List is constantly being updated.

## 3. Simulation and Results

Four virtual machines were established using Amazon Web Services (AWS) EC2 to build and test the proposed IMP. Furthermore, each machine features the Ubuntu 18.04

operating system, two virtual CPUs, 8GB of RAM, and 500 GB of General-Purpose SSD (gp2) storage. In addition, each virtual machine runs a full Bitcoin node, which necessitates downloading and syncing the whole Bitcoin blockchain containing all of the Bitcoin network transactions since 2009. As a result, downloading and syncing the entire blockchain takes three to four days for each machine. Furthermore, the nodes were configured to work on Bitcoin Testnet, a development and testing environment for Bitcoin developers to experiment with new ideas and approaches to Bitcoin technology. Furthermore, each of the four Bitcoin nodes includes a fully functional Lightning node (c-lightning). The four virtual machines (VMs) were used to create and test the model.

### *3.1. Testing the IMP*

A CF can start a Club after advertising the availability of that Club officially and presenting a list of the addresses linked to that specific Club. For example, by publishing the Club addresses and terms and conditions for joining the Club through an officially owned website or a trustworthy platform. The published addresses will be myClub Address, Member Label, Expel Label, IP1, IP2, IP3, IP4, and Club Founder Member ID, as shown in Table 5.

**Table 5.** List of Bitcoin addresses owned by the CF.

| CF IP Address | 35.171.207.143 |
|---|---|
| myClub | tb1qvtmz3gfwakwyvwr47atc597vr3u7mpr7s6zmqd |
| Member Label | tb1qsd46p2qezu4hf57ypnqvlt344cyuwp96lzxzhw |
| Expulsion Label | tb1q7azzpj8jgncekyjrsvl4w7pt847se8mnf5kd0g |
| IP1 | tb1qcfesjr2hgrggtxsre90kyxafps7dsdy6audwlh |
| IP2 | tb1qqcuy77j9att8l9nfppcww42csgk5afjqcprhxp |
| IP3 | tb1qgsadg8el5xxcksjm38v88h0rp7mwe32yx3z72k |
| IP4 | tb1q6mac0p0spg82swf64vhpj0r363jr9l39ch0kpm |
| CF Member ID | tb1qqgn8k933njz6khpgh23mc29pduv98pfe2rkvgu |

Then, after communicating and obtaining the Member ID of a chosen FM, a CF will send an invitation to the Club. The invitation transaction, as explained previously, will include the myClub address, CF Member ID, Member Label, and FM Member ID, and each Bitcoin address for the IP is followed with the corresponding value, as shown in Figure 9.


**Figure 9.** Ubuntu command invitation transaction to an FM to join a Club.

Figure 10 shows the invitation transaction after it was confirmed and included in a mined block in the blockchain.

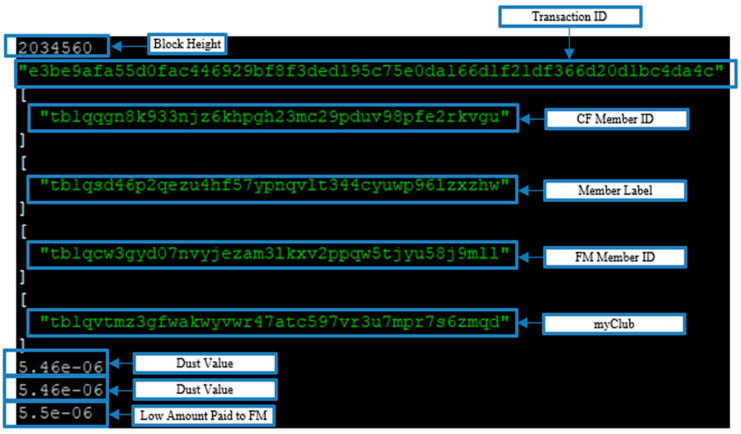


**Figure 10.** Mined FM invitation transaction.

Afterwards, the FM will initiate a bitcoin transaction that includes the Club Address, owned Member ID, and owned IP address along with the corresponding BTC value for each Bitcoin address to pass through the CF firewall. Figure 11 shows the reply to a transaction by the FM.


**Figure 11.** FM sends IP address to pass through the CF firewall.

Next, the CF will view the blockchain to check that the transaction for the FM IP address is confirmed, then extract the data from the blockchain, such as block height, transaction ID, and each Bitcoin address for the IP along with the corresponding BTC values. Figure 12 shows the extracted mined transaction.

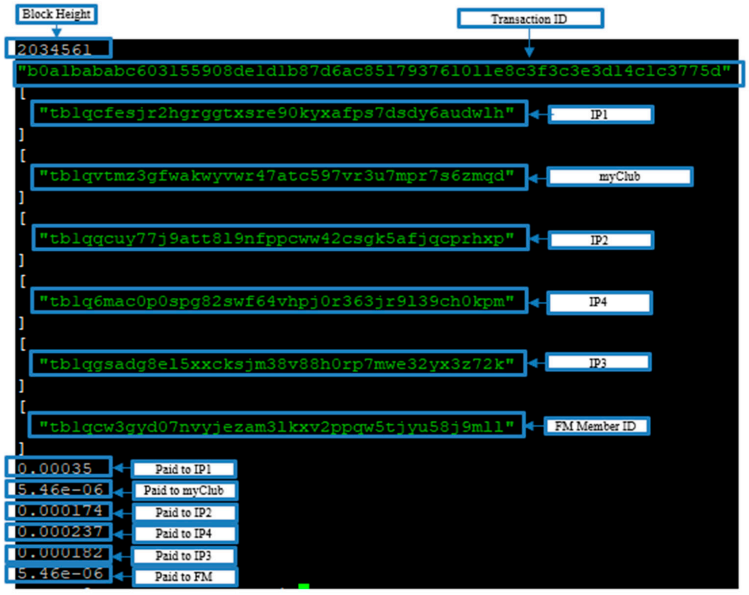


**Figure 12.** Extracted FM IP transaction from blockchain.

Then, the CF will grant the FM access through the firewall, as shown in Figure 13.

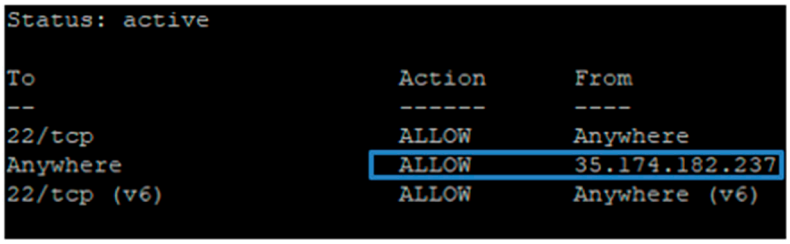


**Figure 13.** CF Allows FM to pass through the firewall to initiate a connection.

After implementing the proposed IMP on the four VMs (one CF, one FM, and two ANs), the list of Club Members (following invitation and access being granted by the Club Founder) is shown in Table 6.

**Table 6.** List of three CMs of a Club after applying the proposed IMP.

| n | Club Address | Member Label | Member ID | Block Height |
|---|---|---|---|---|
| 1 | tb1qvtmz3gfwakwyvwr47atc597vr3u7mpr7s6zmqd | tb1qsd46p2qezu4hf57ypnqvlt344cyuwp96lzxzhw | tb1qcw3gyd07nvyjezam3lkxv2ppqw5tjyu58j9mll | 2,034,560 |
| 2 | tb1qvtmz3gfwakwyvwr47atc597vr3u7mpr7s6zmqd | tb1qsd46p2qezu4hf57ypnqvlt344cyuwp96lzxzhw | tb1qaty98s9us9sur0awklkrh08rxhhc85tey4lzkf | 2,034,564 |
| 3 | tb1qvtmz3gfwakwyvwr47atc597vr3u7mpr7s6zmqd | tb1qsd46p2qezu4hf57ypnqvlt344cyuwp96lzxzhw | tb1qsdk3std3p4qqahg02jt42z67srl4q0pvm44j0x | 2,034,718 |

In contrast, the CF will expel a malicious node that actively disrupts the network and causes payment transaction failures. In this research paper, as an example of a malicious act, a malicious CM will cause $n$ consecutive invoice expirations without paying those invoices. Eventually, the CF will expel that malicious CM from the Club.

Assuming the malicious behavior already occurred, the CF will initiate an expulsion transaction through the Bitcoin network to remove the malicious CM from the Club. The malicious CM should then be added to the list of expelled CMs, which will be automatically updated every ten minutes along with the list of CMs. Figure 14 represents the mined expulsion transaction initiated by the CF against the malicious CM, while Table 7 shows the list of malicious CMs expelled after causing $n$ transaction failures.

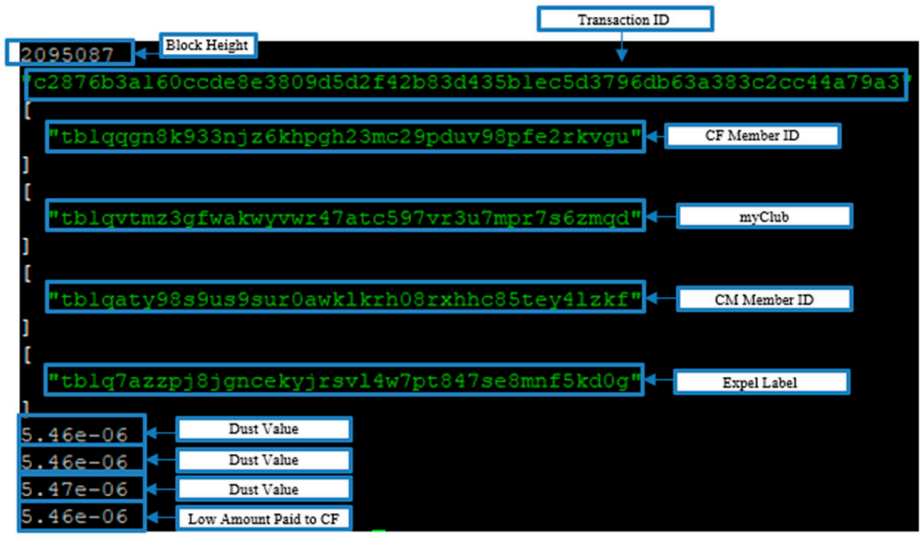


**Figure 14.** Mined expulsion transaction of malicious CM1.

**Table 7.** Expelled List of nodes that caused multiple transaction failures.

| n | Club Address | Expulsion Label | Member ID | Block Height |
|---|---|---|---|---|
| 1 | tb1qvtmz3gfwakwyvwr47atc 597vr3u7mpr7s6zmqd | tb1q7azzpj8jgnceky jrsvl4w7pt847se8mnf5kd0g | tb1qaty98s9us9sur0awklk rh08rxhhc85tey4lzkf | 2,095,087 |

Consequently, the CF will close the Lightning Channel with malicious CM1 using the command *close* followed by the *channel id*, as shown in Figure 15.

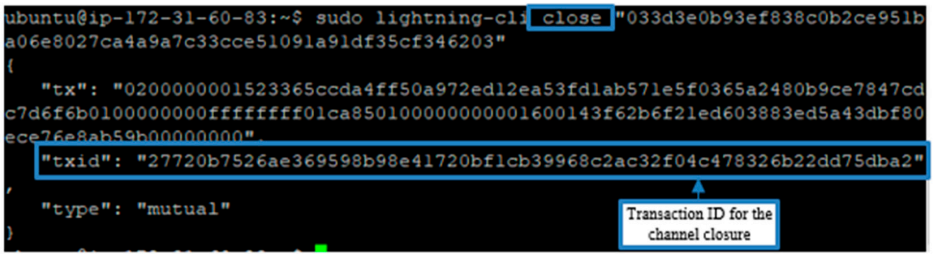


**Figure 15.** CF closing the Lightning Channel with malicious CM1 along with the information of the closure transaction.

Therefore, the CF will deny access to malicious CM1 (IP: 54.146.244.190) through the firewall, as shown in Figure 16.

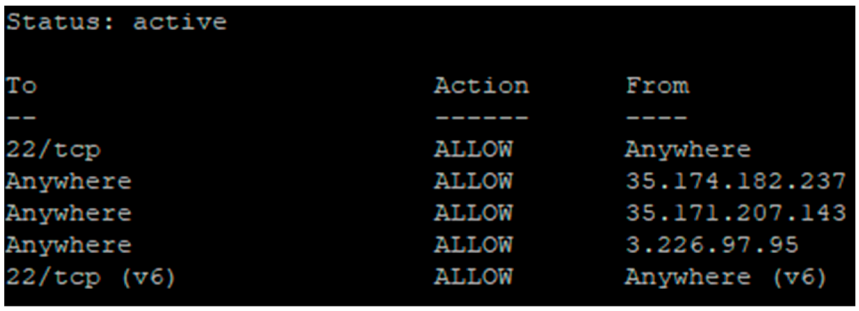


**Figure 16.** Malicious CM1 is removed from the List of CMs granted access through the Firewall.

Accordingly, the expelled CM will never connect to the CF and open a Lightning channel since the malicious CM is blocked by the firewall. Figure 17 shows a failed attempt by expelled CM1 to reconnect to the CF.

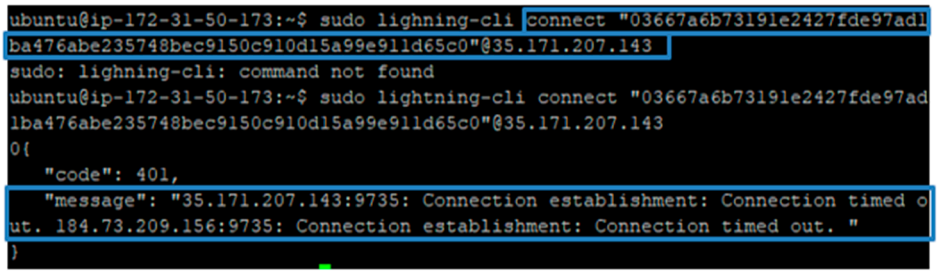


**Figure 17.** Connection failure after expelled CM1 tries to connect to the CF.

Nonetheless, some CMs might experience a temporarily bad internet connection. The CF would send invoices that might expire and be marked as invalid multiple times during that period. As a result, the CF would consider the CM as a malicious Member. Accordingly, the CF will expel the CM even though the invoice expirations were unintentional. Therefore, the IMP allows expelled members to rejoin a Club after being removed under

the condition of repaying the Membership Fee. This condition helps ensure that each CM has an opportunity to rejoin a Club in case of unintentional faults that might occur in the machine that belongs to the CM.

Furthermore, the Membership Fee repayment would also help discourage malicious members from continuing to rejoin a Club as those malicious members would lose more than they could earn after each attempt to rejoin a Club. This protocol helps maintain the security and reliability of the LN targeted by malicious CMs. After the expelled CM pays the Membership Fee to rejoin the Club, the CF will grant firewall access to that node. As indicated in Table 8, after the list of CMs has been updated, the expelled CM will be published among the current valid CMs.

**Table 8.** List of CMs after an expelled node rejoins the Club (node 4).

| n | Club Address | Member Label | Member ID | Block Height |
|---|---|---|---|---|
| 1 | tb1qvtmz3gfwakwyvwr47 atc597vr3u7mpr7s6zmqd | tb1qsd46p2qezu4hf57 ypnqvlt344cyuwp96lzxzhw | tb1qcw3gyd07nvyjezam3lkxv 2ppqw5tjyu58j9mll | 2,034,560 |
| 2 | tb1qvtmz3gfwakwyvwr47 atc597vr3u7mpr7s6zmqd | tb1qsd46p2qezu4hf57 ypnqvlt344cyuwp96lzxzhw | tb1qaty98s9us9sur0awklkrh0 8rxhhc85tey4lzkf | 2,034,564 |
| 3 | tb1qvtmz3gfwakwyvwr47 atc597vr3u7mpr7s6zmqd | tb1qsd46p2qezu4hf57 ypnqvlt344cyuwp96lzxzhw | tb1qsdk3std3p4qqahg02jt42z 67srl4q0pvm44j0x | 2,034,718 |
| 4 | tb1qvtmz3gfwakwyvwr47 atc597vr3u7mpr7s6zmqd | tb1qsd46p2qezu4hf57 ypnqvlt344cyuwp96lzxzhw | tb1qaty98s9us9sur0awklkr h08rxhhc85tey4lzkf | 2,095,235 |

### *3.2. Advantages of the Proposed IMP*

The proposed IMP ensures that a more secure and reliable Lightning Network can be started for the network nodes. The model states that before starting a Lightning Network, nodes must apply a set of protocols that employ the main Bitcoin blockchain to utilize all the security features that the blockchain technology offers. IMP improves the security of the LN by allowing only permitted nodes to connect and start an exclusive LN called a Club that consists only of honest nodes, while unpermitted nodes cannot connect or communicate to send channel-opening requests. Hence, the IMP will improve the security and reliability of an LN through expelling nodes that show malicious behavior, such as causing multiple payment transaction failures or commencing targeted attacks.

## 4. Conclusions and Future Work

This research proposed an invitation model protocol (IMP) for the Bitcoin Lightning Network, which was developed in Bitcoin Testnet through employing four VMs built in AWS. The proposed IMP aims to increase the level of security and reliability of LN technology, especially given that this technology is still in its infancy. Currently, the Bitcoin LN can be prone to different attacks, such as malicious nodes intentionally causing payment failures or DDoS attacks, making the network undesirable and unusable. Users will be discouraged from using the LN, which could potentially render the technology obsolete. The proposed IMP ensures that before starting an LN, nodes must initiate a set of bitcoin transactions to be recorded in the blockchain and then start an exclusive LN called a Club. In addition, a Club has rules that need to be followed by each of the CMs. Following any malicious act by any CM, the CF will expel that CM and mark that member as an Expelled Member. However, an Expelled Member can still rejoin a Club under the condition of repaying the Membership Fee. Therefore, a malicious CM will be at a disadvantage because each malicious act leads to expulsion from the Club, leading to more fund consumption than gain, which acts as a deterrent to malicious behavior. Moreover, the proposed IMP was simulated and shown to be effective against malicious CMs that cause payment transaction failures. Finally, to the best of our knowledge, this work is the first in the literature to explore the level of security and reliability of LN technology via forcing the network nodes to follow a set of protocols to form or join an LN that consists of honest nodes in a trustless environment.

For future work, some features of the IMP can be enhanced as the proposed IMP did not concentrate on efficiency, especially time consumption, as the model was presented

in detailed protocols to make the approach clearer. Hence, the IMP can be optimized by adding more parameters to transactions instead of initiating multiple separate transactions. Moreover, the proposed IMP could be enhanced to facilitate multiple expulsion transactions (i.e., three or more expulsion transactions) against a CM due to malicious acts. The CF announces to the other CMs that this specific CM node has a suspicious IP. Therefore, this node is dishonest. It is best to avoid dealing with such nodes, especially since all the expulsion transactions are registered in the Bitcoin blockchain and will exist forever. Therefore, adding an extra parameter called Reputation for each node in the network would be more efficient. This Reputation metric can also be used by other new CFs with immediate blocking capabilities and provides an extra layer of caution when dealing with nodes with bad Reputation who benefit from the Lightning Network overall. Each CM would try their best to maintain an excellent Reputation to open more channels and be considered trustworthy. Also, an excellent idea for future enhancement is to give limited funding until a suspicious CM node proves that the previous malicious behaviors were unintentional. For example, opening a Lightning Channel with less funding (transactions will not exceed 10 Satoshis). Eventually, an ecosystem will be created where there will be a minimum of dishonest nodes and the LN will function properly. Therefore, the network will mostly contain honest nodes, and malicious nodes will not continue using the network unless those nodes are willing to lose more funds than they can gain.

**Author Contributions:** Conceptualization, A.A. and A.M.M.; methodology, A.A. and A.M.M.; Software, A.A.; validation, A.A.; formal analysis, A.A. and A.M.M.; investigation, A.A.; resources, A.A. and A.M.M.; data curation, A.A.; writing-original draft preparation, A.A. and A.M.M.; writing-review and editing, A.A. and A.M.M.; visualization, A.A. and A.M.M.; supervision, A.M.M.; project administration, A.M.M.; funding acquisition, A.M.M.; All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Data Availability Statement:** Data can be available upon request.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Cui, M.; Li, X.; Kamoche, K. Transforming from Traditional To E-intermediary: A Resource Orchestration Perspective. *Int. J. Electron. Commer.* **2021**, *25*, 338–363. [CrossRef]
2. Jackson, J.E.; Xu, X. Does Scarcity Add Value in Influencing Consumers in the Try-Before-You-Buy Model? *Int. J. Electron. Commer.* **2022**, *26*, 25–48. [CrossRef]
3. Yu, Y.; Chung, T. Electronic Payment Performance: A Trend and Contextual Analysis of Its Social Impact on Secured E-Payment in 2016–2019. In *Handbook of Research on Social Impacts of E-Payment and Blockchain Technology*; IGI Global: Hershey, PA, USA, 2022; pp. 196–228.
4. Asokan, N.; Janson, P.; Steiner, M.; Waidner, M. State of the Art in Electronic Payment Systems. In *Advances in Computers*; Elsevier: Amsterdam, The Netherlands, 2000; Volume 53, pp. 425–449.
5. Nakamoto, S.; Bitcoin, A. A peer-to-peer electronic cash system. *Bitcoin* **2008**, *4*, 2. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 13 February 2023).
6. Kim, H.-W.; Jeong, Y.-S. Secure authentication-management human-centric scheme for trusting personal resource information on mobile cloud computing with blockchain. *Hum.-Cent. Comput. Inf. Sci.* **2018**, *8*, 11. [CrossRef]
7. Park, J.H.; Park, J.H. Blockchain Security in Cloud Computing: Use Cases, Challenges, and Solutions. *Symmetry* **2017**, *9*, 164. [CrossRef]
8. Kumar, A.; Abhishek, K.; Nerurkar, P.; Ghalib, M.R.; Shankar, A.; Cheng, X. Secure smart contracts for cloud-based manufacturing using Ethereum blockchain. *Trans. Emerg. Telecommun. Technol.* **2022**, *33*, e4129. [CrossRef]
9. Xu, G.; Bai, H.; Xing, J.; Luo, T.; Xiong, N.N.; Cheng, X.; Liu, S.; Zheng, X. SG-PBFT: A secure and highly efficient distributed blockchain PBFT consensus algorithm for intelligent Internet of vehicles. *J. Parallel Distrib. Comput.* **2022**, *164*, 1–11. [CrossRef]
10. Ayub Khan, A.; Laghari, A.A.; Shaikh, A.A.; Bourouis, S.; Mamlouk, A.M.; Alshazly, H. Educational Blockchain: A Secure Degree Attestation and Verification Traceability Architecture for Higher Education Commission. *Appl. Sci.* **2021**, *11*, 10917. [CrossRef]
11. El Azzaoui, A.; Choi, M.Y.; Lee, C.H.; Park, J.H. Scalable Lightweight Blockchain-Based Authentication Mechanism for Secure VoIP Communication. *Hum.-Cent. Comput. Inf. Sci.* **2022**, *12*. [CrossRef]

12. Zabka, P.; Foerster, K.-T.; Schmid, S.; Decker, C. Empirical evaluation of nodes and channels of the lightning network. *Pervasive Mob. Comput.* **2022**, *83*, 101584. [CrossRef]
13. Poon, J.; Dryja, T. The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments. 2016. Available online: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwiW8KKDvcf_AhWE1zgGHZUZCh0QFnoECA0QAQ&url=https%3A%2F%2Flightning.network%2Flightning-network-paper.pdf&usg=AOvVaw3PUYDaHMXxEkS9gax1FEeb (accessed on 17 May 2023).
14. van Dam, G.; Kadir, R.A. Hiding payments in lightning network with approximate differentially private payment channels. *Comput. Secur.* **2022**, *115*, 102623. [CrossRef]
15. Bartolucci, S.; Caccioli, F.; Vivo, P. A percolation model for the emergence of the Bitcoin Lightning Network. *Sci. Rep.* **2020**, *10*, 4488. [CrossRef]
16. Antonopoulos, A.M.; Osuntokun, O.; Pickhardt, R. *Mastering the Lightning Network*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2021.
17. Instituto Superior Técnico, Tecnico Lesboa, Portugal 2018. Available online: https://www.researchgate.net/profile/Joao-Valente-7/publication/329809850_A_Lightning_Network_Analysis/links/5c1b9dc8299bf12be38d1a11/A-Lightning-Network-Analysis.pdf (accessed on 17 May 2023).
18. Nowostawski, M.; Tøn, J. Evaluating methods for the identification of off-chain transactions in the lightning network. *Appl. Sci.* **2019**, *9*, 2519. [CrossRef]
19. Conoscenti, M.; Vetrò, A.; De Martin, J.C.; Spini, F. The cloth simulator for htlc payment networks with introductory lightning network performance results. *Information* **2018**, *9*, 223. [CrossRef]
20. Seres, I.A.; Gulyás, L.; Nagy, D.A.; Burcsi, P. Topological Analysis of Bitcoin's Lightning Network. In *Mathematical Research for Blockchain Economy*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 1–12.
21. Rohrer, E.; Malliaris, J.; Tschorsch, F. Discharged payment channels: Quantifying the lightning network's resilience to topology-based attacks. In Proceedings of the 2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Stockholm, Sweden, 17–19 June 2019; pp. 347–356.
22. Pérez-Sola, C.; Ranchal-Pedrosa, A.; Herrera-Joancomartí, J.; Navarro-Arribas, G.; Garcia-Alfaro, J. Lockdown: Balance Availability Attack against Lightning Network Channels. In *International Conference on Financial Cryptography and Data Security*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 245–263.
23. Zhang, Y.; Yang, D.; Xue, G. Cheapay: An optimal algorithm for fee minimization in blockchain-based payment channel networks. In Proceedings of the ICC 2019-2019 IEEE International Conference on Communications (ICC), Shanghai, China, 20–24 May 2019; pp. 1–6.
24. Bai, Q.; Xu, Y.; Wang, X. Understanding the Benefit of Being Patient in Payment Channel Networks. *IEEE Trans. Netw. Sci. Eng.* **2022**, *9*, 1895–1908. [CrossRef]
25. Piatkivskyi, D.; Nowostawski, M. Split Payments in Payment Networks. In *Data Privacy Management, Cryptocurrencies and Blockchain Technology*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 67–75.
26. Ren, A.H.J.; Feng, L.; Cheong, S.A.; Goh, R.S.M. Optimal fee structure for efficient lightning networks. In Proceedings of the 2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS), Singapore, 11–13 December 2018; pp. 980–985.
27. Osuntokun, O. AMP: Atomic Multi-Path Payments over Lightning. 2018. Available online: https://lists.linuxfoundation.org/pipermail/lightning-dev/2018-February/000993.html (accessed on 17 May 2023).
28. Yu, R.; Xue, G.; Kilari, V.T.; Yang, D.; Tang, J. Coinexpress: A fast payment routing mechanism in blockchain-based payment channel networks. In Proceedings of the 2018 27th International Conference on Computer Communication and Networks (ICCCN), Hangzhou, China, 30 July–2 August 2018; pp. 1–9.
29. Qiao, Y.; Wu, K.; Khabbazian, M. Non-intrusive and high-efficient balance tomography in the lightning network. In Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security, Hong Kong, 7–11 June 2021; pp. 832–843.
30. Tang, W.; Wang, W.; Fanti, G.; Oh, S. Privacy-utility tradeoffs in routing cryptocurrency over payment channel networks. *Proc. ACM Meas. Anal. Comput. Syst.* **2020**, *4*, 1–39. [CrossRef]
31. Robert, J.; Kubler, S.; Ghatpande, S. Enhanced Lightning Network (off-chain)-based micropayment in IoT ecosystems. *Future Gener. Comput. Syst.* **2020**, *112*, 283–296. [CrossRef]
32. Mercan, S.; Kurt, A.; Akkaya, K.; Erdin, E. Cryptocurrency solutions to enable micropayments in consumer IoT. *IEEE Consum. Electron. Mag.* **2021**, *11*, 97–103. [CrossRef]
33. Kurt, A.; Mercan, S.; Shlomovits, O.; Erdin, E.; Akkaya, K. Lngate: Powering iot with next generation lightning micro-payments using threshold cryptography. In Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks, Abu Dhabi, United Arab Emirates, 28 June–2 July 2021; pp. 117–128.
34. Pinto, F.; da Silva, C.F.; Moro, S. People-Centered Distributed Ledger Technology-IoT Architectures: A Systematic Literature Review. *Telemat. Inform.* **2022**, *70*, 101812. [CrossRef]
35. Salim, M.M.; Shanmuganathan, V.; Loia, V.; Park, J.H. Deep learning enabled secure IoT handover authentication for blockchain networks. *Hum. Cent. Comput. Inf. Sci* **2021**, *11*, 21.
36. Wang, T.; Ai, S.; Cao, J.; Zhao, Y. A Blockchain-Based Distributed Computational Resource Trading Strategy for Internet of Things Considering Multiple Preferences. *Symmetry* **2023**, *15*, 808. [CrossRef]

37. Mihaljević, M.J.; Knežević, M.; Urošević, D.; Wang, L.; Xu, S. An Approach for Blockchain and Symmetric Keys Broadcast Encryption Based Access Control in IoT. *Symmetry* **2023**, *15*, 299. [CrossRef]
38. Mazumdar, S.; Banerjee, P.; Sinha, A.; Ruj, S.; Roy, B.K. Strategic Analysis of Griefing Attack in Lightning Network. *IEEE Trans. Netw. Serv. Manag.* **2022**. [CrossRef]
39. Kappos, G.; Yousaf, H.; Piotrowska, A.; Kanjalkar, S.; Delgado-Segura, S.; Miller, A.; Meiklejohn, S. An empirical analysis of privacy in the lightning network. In Proceedings of the Financial Cryptography and Data Security: 25th International Conference, FC 2021, Virtual Event, 1–5 March 2021; Revised Selected Papers, Part I 25; Springer: Berlin/Heidelberg, Germany, 2021; pp. 167–186.