Pervasive and Mobile Computing 61 (2020) 101108

Contents lists available at ScienceDirect

Pervasive and Mobile Computing

journal homepage: www.elsevier.com/locate/pmc

On the robustness of Lightning Network in Bitcoin

Seungjin Lee, Hyoungshick Kim *

Department of Computer Science and Engineering, Sungkyunkwan University, Suwon, South Korea

ARTICLE INFO

Article history:
Received 6 December 2018
Received in revised form 3 October 2019
Accepted 4 October 2019
Available online 29 November 2019

Keywords:
Lightning Network
Bitcoin
Blockchain
Network robustness

ABSTRACT

An off-chain transaction protocol called Lightning Network was recently introduced to enable low-fee and micropayment transactions on the top of the Bitcoin network. This technology has the potential to solve the inherent scalability issue of the Bitcoin system, but is still questionable whether the proposed system is highly secure and robust against various cyberattacks.

In this paper, we analyzed the network characteristics of the Lightning Network mainnet which was launched on March 15, 2018 and found that the Lightning Network topology shows strong scale-free network characteristics. This implies that Lightning Network can be vulnerable to DDoS attacks targeting some specific nodes in the network because there are a few highly connected "hub" nodes in a scale-free network. We experimentally analyze the robustness of Lightning Network via the simulation of network attack model. Our simulation results demonstrate that the connectivity of the existing Lightning Network cannot be maintained sufficiently against target attacks that repeatedly destroy a few nodes with high centrality.

1. Introduction

Since the concept of Bitcoin was introduced as a white paper in 2008 [1], it is an emerging digital currency system that enables online payments to anyone and anywhere in the world. Bitcoin has the potential to revolutionize payment systems on the Internet because it can solve the double spending problem without a trusted centralized entity unlike existing credit card payment systems. Therefore, many companies and stores have already begun to integrate payment options for Bitcoin [2]. Nowadays Coinbase, one of the largest exchanges, already has over 13 million users. It is estimated there are over 22 million Bitcoin wallets and already an estimated 5% of US citizens already hold Bitcoin [3].

However, Bitcoin has also been severely criticized for the lack of scalability. While traditional payment system such as Visa can handle 4000 transactions per second on average and has been stress-tested in 2013 to process 48,000 transactions per second [4], Bitcoin can process only 7 transactions per second in principle due to the limitations of 1 megabyte block size and 10-minutes block creating interval time. The idea of Lightning Network [5] was developed to solve this problem by avoiding frequent Bitcoin transactions on the blockchain, which are inherently expensive to maintain the consensus among all Bitcoin nodes. Lightning Network is a "second layer" payment protocol that operates on top of a blockchain. Lightning Network consists of nodes running the Bitcoin protocol and (payment) channels between the nodes [5]. We found that, similar to other real-world networks such as World Wide Web [6], the Internet, and various social and biological networks, the Lightning Network also has a scale-free property that exhibits a power-law distribution of degree. In general, the network connectivity of scale-free networks can dramatically be reduced by attacking some network nodes and edges with high network centrality [7].

* Corresponding author.
E-mail address: hyoung@skku.edu (H. Kim).

https://doi.org/10.1016/j.pmcj.2019.101108
1574-1192/

In this paper, we present a simulation of the network attacks and defenses on the Lightning Network topology using the framework used in [8,9] to model iterated attack and defense operations, to empirically analyze the robustness of Lightning Network. In this framework, we can analyze the effects of dynamic interactions between an attacker who is capable of removing some nodes from network and a defender who adds some nodes to the network. We can also try to find the optimal attack and defense strategies among several strategies. These analysis results would be helpful for designing network configuration protocols in blockchain systems. In fact, maintaining the robustness of Lightning Network is a challenging issue because failures of some nodes and/or channels can cause critical financial damages to individuals and organizations who have used Lightning Network.

The main contributions of this paper can be summarized as follows:

- We found that the real-world Bitcoin Lightning Network exhibits the scale-free properties, which makes the network structure vulnerable to target attacks.
- We present a framework to evaluate the robustness of Lightning Network through simulations modeling several network attack and defense strategies.
- The simulation results demonstrate that the real-world Lightning Network is vulnerable to target attacks that destroys a few nodes with high degree. We found that the attack targeting high-degree nodes completely destroy the network within about 20 rounds of attack process. On the other hand, the defense strategy balancing the edge distribution through replenishment of nodes enables the largest connected component of network to be maintained at over 80% of the existing network.

The remainder of this paper is organized as follows: Section 2 describes the background on Lightning Network and community structure and motivations of this paper. Section 3 describes the simulation model that we implemented to analyze the robustness of Lightning Network. Section 4 discusses the results of several simulations. Section 5 describes the existing works in the area of network attack modeling and we conclude in Section 6.

2. Background

2.1. Bitcoin Lightning Network

Lightning Network [5] is a second layer payment protocol built on the blockchain of Bitcoin and developed to solve the Bitcoin scalability problem–Bitcoin (7 transactions per second (TPS)) is significantly slower than mainstream payment systems such as Visa (2000 TPS) using a centralized database.

The key idea of Lightning Network is to avoid computationally expensive and slow on-chain transactions on the blockchain requiring consensus among Bitcoin nodes as much as possible. Instead, off-chain payment channels (also referred to as "*state channel*") can be used where they create a relationship between two parties to perpetually update balances, deferring what is broadcast to the blockchain in a single transaction netting out the total balance between those two parties.

Such payment channels can be securely implemented by setting up a multi-signature address that is shared by two participants of Lightning Network. When a payment channel between two parties is established, both parties create an actual Bitcoin transaction for the multi-signature address on the Bitcoin blockchain. Under constructed payment channels, off-chain transactions can be efficiently performed with a local database between two parties without broadcasting to the public blockchain. Unlike conventional database systems, however, the local database for off-chain payment channels can be updated when both parties simultaneously agree to update because both their digital signatures are needed to create update transactions. In other words, this multi-signature address serves as a kind of vault that can only be accessed with both participants' digital signatures and manages a payment channel ledger containing the balance of each participant. The sum of these balances is called a *channel capacity*. When the channel is closed, the final version of the transaction to distribute the channel's remaining funds is broadcasted to the Bitcoin blockchain system.

Such payment channels can be used together to form Lightning Network by connecting multiple channels for a path between participants who are not directly connected with a channel. For payment routing processes on a path, the capacity of all the channels constituting the path should be sufficient to route the amount to transact. Therefore, the participants of Lightning Network are gathered into channels with as much capacity as possible. Therefore, the requirement of the channel capacity would generate some hub nodes which have many connections to other node which make the network vulnerable to intentional attacks targeting the hub nodes. In this paper, we experimentally analyze the robustness of Bitcoin Lightning Network through simulations modeling several attack and defense strategies.

2.2. DDoS attack on Lightning Network

As mentioned previously, Lightning Network has a scale-free property, so it may vulnerable to an attack targeting some specific nodes. Indeed, in March 2018, DDoS (Distributed Denial of Service) attack was occurred on some nodes of Lightning Network, and about 200 of 1050 nodes of Lightning Network were offline due to the attack [10]. This attack was implemented simply by sending a huge amount of new payment channel creation requests to some Lightning Network nodes, where target nodes could not create a new channel to be used for the actual transaction. In this attack, target nodes

are selected randomly. However, Lightning Network is a scale-free network, so it has a few hub nodes highly connected to many other nodes. Thus, if this DDoS attack can be performed by carefully selecting the hub nodes, the connectivity of Lightning Network may be seriously damaged. So we simulate several types of DDoS attacks according to the strategy how the attacker select nodes to destroy.

2.3. Community structure

One of the common approach in research on large and complex networks, such as the Web, social networks, and mobile networks, is to extract and visualize the community structures [11]. There are several studies to quickly find good partitions in relation to extraction algorithms of community structure [11–13]. In this paper, we use the Louvain method [14] based on the modularity optimization approach as a community extraction algorithm. The Louvain method has an advantage over the other community extraction algorithms in terms of time complexity, so that it can find a community structure effectively even when there are many network nodes and edges. The algorithm is repeated several times in a step-by-step manner. In each step, the procedures of the two processes of modularity optimization and community aggregation are performed in order. In the modularity optimization process, each node in the network is moved to the community to which the neighbor node belongs, so that the distance between the nodes in a community is short, the distance between the communities is long, and the value of modularity is maximized. For a weighted graph, the modularity of community is defined as below:

$$Q = \frac{1}{2m} \sum_{ij} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta \left(c_i, c_j \right) \tag{1}$$

where A_{ij} is the edge weight between nodes i and j, k_i is the sum of the weights of the edges connected to the node i, $2m$ is the sum of all of the edge weights in the graph, c_i is the community of the node i and δ is a *Kronecker* delta function. In the community aggregation process, a new network is constructed by organizing the extracted communities into one node. The algorithm repeats these two steps until the modularity value is no longer improved.

3. Simulation model

Our simulation modeling attacks and defenses on Lightning Network is implemented based on the framework suggested in the existing work [9]. Thus, the simulation model can be formalized as a game on a graph G by iterating attack and defense phase during a certain number of rounds. In this model, the objective of an attacker is to maximize disruption to Lightning Network so that the connectivity of network decreases, on the other hand, the objective of a defender is to minimize the disruption by deploying new resources. In attack phase, the attacker selects k_a nodes from the graph G according to attack strategies, then the attacker removes them. When the node is removed, all connected edges are also removed. As presented in Section 2.2, there may be various attack strategies according to how the attacker select k_a nodes to destroy. In defense phase, the defender adds k_d nodes into the graph G with m edges to m different nodes in G. Because the best defensive strategy can be to restore the immediately previous graph state, we assume that the defender cannot know which nodes are removed from previous attack phase. The number of the added edges in defense phase, m, is defined as below:

$$m = Round \left(w * d \left(G \right) \right) \tag{2}$$

where w is the edge construction weight representing the ability of defender to create edge, and $d\left(G\right)$ is the average degree of graph G. Similar to the attack phase, there may be various defense strategies according to how the defender select the m existing nodes to connect the newly replenished nodes. The simulation model consists of multiple rounds, each round consisting of one attack phase and one defense phase. We evaluate the robustness of Lightning Network in terms of network's connectivity after some rounds. As a metric of the connectivity, we use the average degree, the size of largest connected component (LCC) and the average connectivity in a network. We vary the parameters, k_a, k_d and w across the simulations for measuring the impact of each variable change on the effectiveness of attack and defense strategies in terms of the connectivity.

3.1. Attack and defense strategies

In this paper, we experimentally analyze the robustness of Lightning Network through simulations modeling attack and defense on the network. Based on the results of previous studies [8,9], we analyzed the effectiveness of several attack and defense strategies using network centrality. In this paper, we specifically propose a new attack strategy using network community structures. We consider the following attack strategies in our attack and defense simulations.

- **Random attack** selects nodes to be removed randomly from a graph G. That is, in attack phase, the attacker selects k_a nodes randomly, then removes them from G. This attack assumes that the attacker have no knowledge of the network topology and is used as a baseline to compare the effectiveness of other attack strategies. This strategy requires no knowledge about the network topology. We use A^{random} to denote the random attack strategy.

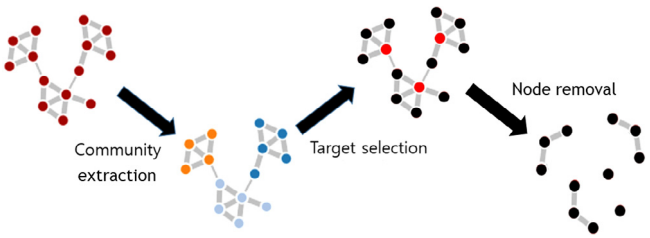


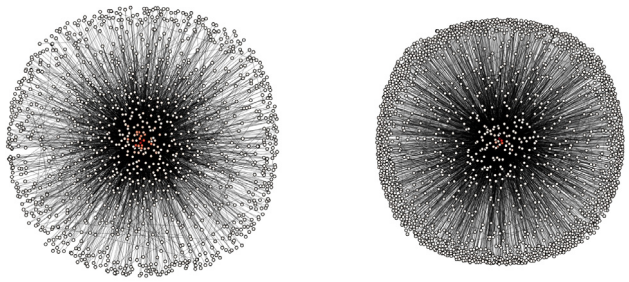
Fig. 1. Process of community-based attack strategy.

- **High-degree attack** selects nodes to be removed according to their degree. That is, the attacker selects k_a highest-degree nodes in sequence, then removes them from G. Contrary to random attack, the attacker must know the topology of Lightning Network for this attack. However, this seems a reasonable assumption because the attacker can easily access the channel information of Lightning Network due to the open nature of blockchain. We use A^{degree} to denote the high-degree attack strategy.
- **High-centrality attack** selects nodes to be removed according to their betweenness centrality. That is, the attacker selects k_a highest-centrality nodes in sequence, then removes them from G. The centrality is one of the measure representing the importance of node in graph. Among the various types of centrality, such as degree centrality, eigenvector centrality and closeness centrality, we only consider betweenness centrality known to be more related to network connectivity than others. This strategy requires knowledge about the node centrality in the network. We use $A^{central}$ to denote the high-centrality attack strategy.
- **Community-based attack** unlike other approaches, first extracts the community structure from the graph G, then selects k_a nodes to be removed in each community sequentially, according to their degree. This strategy requires knowledge about the network topology. We use $A^{community}$ to denote the community-based attack strategy. Fig. 1 represents the process of $A^{community}$ in a round where $k_a = 3$. In original graph, the average degree is 2.714 and the size of largest connected component is 14. The attacker first extracts three community structures in the graph using Louvain method, then selects nodes to be removed in each community sequentially, according to their degree (in total $k_a = 3$ nodes). After the attack phase, the graph is partially collapsed so that the average degree becomes 1.091 and the size of LCC becomes 3.

Next, we explain defense strategies to be considered in this paper.

- **Random defense** selects nodes to be connected with newly replenished node randomly from a graph G. That is, in the defense phase, the defender replenishes k_d nodes into the graph G with m edges to randomly selected m different nodes in G. This defense is used as a baseline to compare the effectiveness of other defense strategies. This strategy requires no knowledge about the network topology. We use D^{random} to denote the random defense strategy.
- **Preferential defense** selects nodes to be connected with newly replenished node according to their degree. That is, the defender replenishes k_d nodes into the graph G with m edges to m different highest-degree nodes in sequence. This strategy requires the knowledge about the node degree in the network. We use D^{prefer} to denote the preferential defense strategy.
- **Balanced defense** selects nodes to be connected with newly replenished node according to their betweenness centrality. In this attack, the defender replenishes k_d nodes into the graph G with m edges to m different lowest-centrality nodes in sequence for balanced edge distribution. This strategy requires the knowledge about the node centrality in the network. We use $D^{balance}$ to denote the balanced defense strategy.

A defense strategy can be implemented as follows: (1) When a new node joins the network, an explorer (e.g., [15]) for Lightning Network can recommend a set of proper nodes as neighbor nodes, which makes the network more reliable against DDoS attacks. We here assume that users select the recommended peer nodes as their neighbors. This assumption would be reasonable because users generally prefer to make the system more secure and reliable in order to minimize disruption of their operations on Lightning Network. We can also develop a neighbor assignment scheme at the protocol level to explicitly encourage users to select the recommended peer nodes as their neighbor nodes—if a node selects a recommended node as its neighbor node, the Lightning Network protocol may grant transaction fee exemption or reduction on the transactions between those two nodes. We note that each node can select its own peer nodes based on their node degree or network centrality and directly try to connect them for the peer node discovery process because the Lightning Network topology is publicly available. In practice, any connection between nodes in Lightning Network can be constructed in principle because Lightning Network is a virtual overlay network which is established above the IP layer.



(a) Mainnet (b) Testnet

Fig. 2. Visualization of Bitcoin Lightning Network.

Table 1
Properties of lightning network.

	$\|V\|$	$\|E\|$	Diameter	Density	$d(G)$	LCC size	$s(G)$
Mainnet	1211	5277	8	0.007	8.715	1194	1.039
Testnet	1898	4289	8	0.002	4.520	1865	1.097

3.2. Network configuration

For our simulation, we first construct the networks of the Lightning Network mainnet and testnet. The information of Lightning Network is scraped from the Lightning Network explorer [15,16]. Fig. 2 represents visualization of the Lightning Network topology constructed from the scraped information. Figs. 2(a) and 2(b) is the graph layouts of the Lightning Network mainnet and testnet, respectively. In the visualized graph, the higher the degree of a node, the more reddish, of which the number is very small. We also summarize the graph properties of the Lightning Network mainnet and testnet used in the simulation in Table 1.

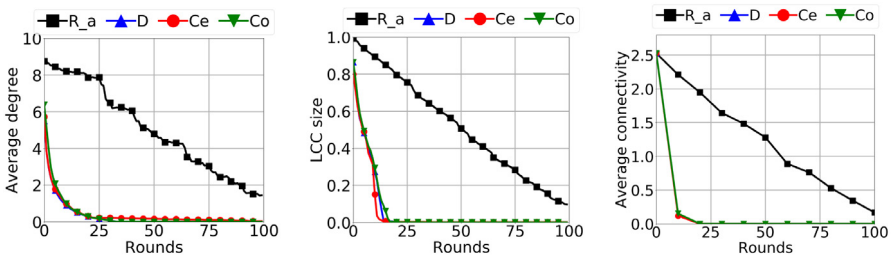
Diameter is the maximum distance between nodes in the graph [17], and Density is a normalized version of the average number of neighbors, which indicates the overall level of interaction between all nodes in the graph [18]. Given a graph G, $d(G)$ and $s(G)$ are the average degree of G, and the average shortest path length, respectively.

4. Simulation results

In this section, we describe our results of some simulations on the Lightning Network mainnet and testnet. We also performed simulations on the Lightning Network testnet and confirmed that attack and defense strategies produced similar results. The simulation results for the Lightning Network testnet are presented in Appendix.

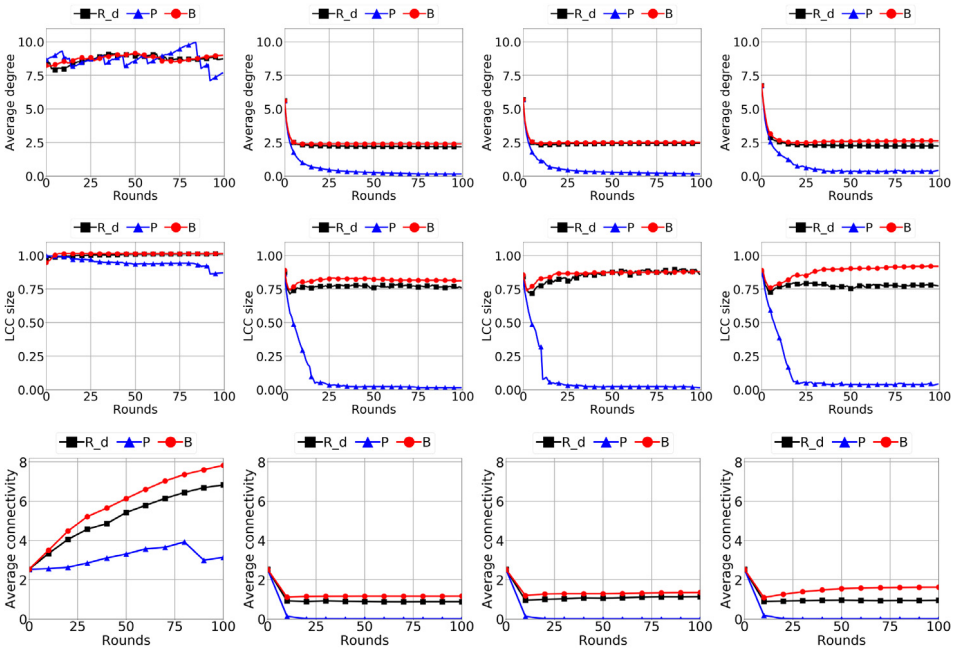
4.1. Effectiveness of attack strategies

For the baseline, we first evaluate the effectiveness of target attacks on the Lightning Network mainnet. In this simulation, we evaluate the effectiveness of attack strategies with the average degree, the LCC size and the average connectivity of the Lightning Network mainnet when the four attack strategies described above, random attack, high-degree attack, high-centrality attack, and community-based attack are used. Fig. 3 shows the changes in the average degree, the size of LCC and the average connectivity of the Lightning Network under four attack strategies. The abbreviations R, D, Ce, and Co in the legend of graph represent random attack, high-degree attack, high-centrality attack and community-based attack, respectively. As shown in Fig. 3, regardless of defense strategy, the average degree, the LCC size and the average connectivity of Lightning Network dramatically decreased, respectively. In all the attacks except random attack, the average degree, the size of LCC and the average connectivity approaches zero before 20 rounds. Among target attacks, the high-centrality attack strategy performed slightly better than the other strategies in terms of the size of LCC. The random attack steadily reduced the average degree, the size of LCC and the average connectivity over rounds, but the metrics did not become zero even after 100 rounds. This result shows that if a target attack such as DDoS attack mentioned in the previous section occurs on the Lightning Network, and the proper defense strategy is not used, the connectivity of the network can be damaged in a short time.



(a) Average degree (b) Size of LCC (c) Average connectivity

Fig. 3. Changes in the average degree, the size of LCC and the average connectivity for the mainnet of Lightning Network under four attack strategies (R_a: random attack, D: high-degree attack, Ce: high-centrality attack, Co: community-based attack).



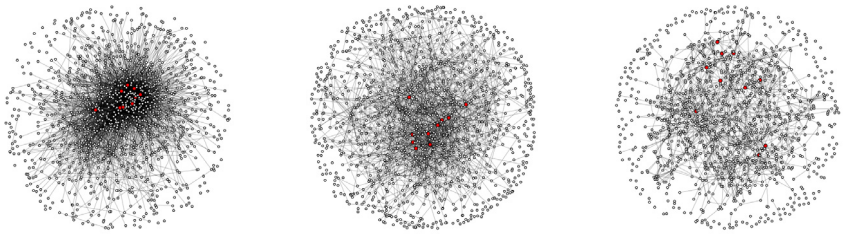
(a) A^{random} (b) A^{degree} (c) $A^{central}$ (d) $A^{community}$

Fig. 4. Changes in the average degree, the size of LCC and the average connectivity for the mainnet of Lightning Network under four attack strategies and three defense strategies. (R_d: random defense, P: preferential defense, B: balanced defense).

4.2. Connectivity evolution over the multiple rounds

In this paper, our objective is to experimentally analyze the robustness of Lightning Network through simulations modeling attacks on the network and find the best attack and defense strategies. To observe how the average degree, the size of LCC and the average connectivity change as attack and defense phases are processed, we first fixed k_a, k_d, and m. For example, Fig. 4 shows the changes of the average degree, the size of LCC and the average connectivity in the Lightning Network mainnet under iterated attack and defense operations with $k_a = 10$, $k_d = 10$ and $w = 1.0$. Figs. 4(a)–4(d) represent the changes of metrics in each attack strategies, A^{random}, A^{degree}, $A^{central}$ and $A^{community}$, in sequence. The abbreviations R, P and B in legend represent random defense, preferential defense and balanced defense, respectively. The size of LCC is normalized by dividing by the size of LCC in the original graph.

Under the A^{random}, the results show that the average degree and the size of LCC remained mostly unchanged when D^{random} and $D^{balance}$ defense strategies are used. Interestingly, the average connectivity in the network gradually increases over rounds when D^{random} and $D^{balance}$ defense strategies are used. In the case of D^{prefer}, there were some fluctuations in the average degree because of the randomness of target nodes in attack phase, but it remained near the average degree of the original graph. That is, D^{random}, D^{prefer} and $D^{balance}$ strategies are effective for the random attack or random node



(a) After 1 round (b) After 5 rounds (c) After 20 rounds

Fig. 5. Evolution of the mainnet of Lightning Network over rounds when $D^{balance}$ is used against A^{degree}, where $k_a = 10$, $k_d = 10$, and $w = 1.0$.

Table 2
Summary of the effectiveness (in the average degree (DEG), the size of LCC (LCC) and the average connectivity (AC)) of all attack and defense strategies.

	Measure	A^{random}	A^{degree}	$A^{central}$	$A^{community}$
D^{random}	DEG	≈8.72	≈**2.17**	≈2.43	≈2.57
	LCC	≈1.0	≈0.76	≈0.86	≈**0.75**
	AC	≈6.82	≈**0.88**	≈1.13	≈0.95
D^{prefer}	DEG	≈7.67	≈**0.14**	≈0.15	≈0.33
	LCC	≈0.87	≈**0.01**	≈**0.01**	≈0.03
	AC	≈3.14	≈**0.01**	≈**0.01**	≈0.01
$D^{balance}$	DEG	≈8.95	≈**2.39**	≈2.51	≈2.62
	LCC	≈1.0	≈**0.81**	≈0.87	≈0.92
	AC	≈7.83	≈**1.16**	≈1.35	≈1.62

failure. This result is consistent with the fact that the scale-free graph is robust against random attack or random node failure.

Contrary to random attack scenario, under other attack strategies, the average degree of the mainnet tended to decrease dramatically. Especially, when D^{prefer} was used as a defense strategy, all the metrics under the attack strategies other than A^{random} was almost zero within about 25 rounds. Even with D^{random} and $D^{balance}$ defense strategies, the average degree was decreased dramatically (see Fig. 5) but not below 2. For the average connectivity, we can also see similar trends.

As discussed in the previous study [9], we can observe a relationship between the average degree and the size of LCC; the size of LCC also plunged to zero in the case of D^{prefer} as the average degree decreased to less than 2. For other defense strategies where the average degree was not decreased below 2, there was a slight decrease in the size of LCC. In summary, the effectiveness of all attack and defense strategies can be presented in Table 2. We highlighted the best attack results in bold.

In the view of attackers, a clear winning strategy is "high-degree attack" (i.e., A^{degree}) where high degree nodes are first targeted. Overall, A^{degree} produced better attack results than $A^{central}$ and $A^{community}$ against $D^{balance}$ and D^{random} in terms of all metrics except the size of LCC against D^{random}.

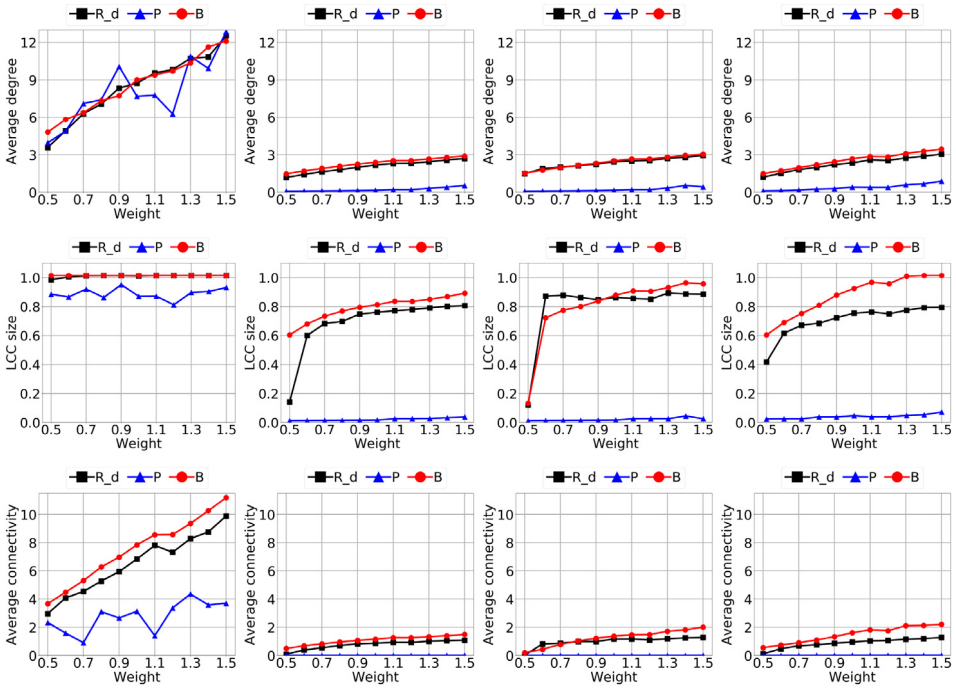
We can see that the best defense strategy is "balanced defense" (i.e., $D^{balance}$) where new nodes tend to be connected to low centrality nodes. $D^{balance}$ overall produced slightly better results than D^{random}. In particular, $D^{balance}$ is significantly more effective than D^{random} in the average connectivity against A^{degree} and $A^{community}$.

Interestingly, for target attacks (A^{degree}, $A^{central}$, and $A^{community}$), the average node degree always converges to around 2.5 over rounds if either $D^{balanced}$ or D^{random} is used as defense strategy.

4.3. Connectivity according to edge construction weight w

We also discuss the changes in connectivity metrics with the number of edges connected to newly added nodes in the defense phase. To observe the effects of edge construction weight w on the average degree, the size of LCC and the average connectivity, we fix $k_a = k_d = 10$ and analyze the metrics after 100 rounds with varying w ranging from 0.5 to 1.5. Through this experiment, we explore the cost required to maintain the original graph's connectivity. The experimental results for each attack strategy are demonstrated in Fig. 6.

As shown in Fig. 6, the average degree of the network increases as w increases in all cases. For A^{random}, the average degree and the average connectivity of the network linearly increase with w even though D^{prefer} leads to some fluctuations. Based on such connectivity results, the size of LCC also remains almost unchanged. Under the target attack strategies rather than A^{random}, the effects of w are rather limited. In particular, w is not effective when D^{prefer} is used. For example, even when D^{prefer} is used with $w = 1.5$, all connectivity metrics are extremely low, which means that most nodes are disconnected. With $D^{balance}$ and D^{random}, the size of LCC increases significantly even against all target attacks when $w \geq 0.7$. Interestingly, $D^{balance}$ is particularly effective against $A^{community}$; the size of LCC is almost the same of the original network against $A^{community}$ when $D^{balance}$ is used with $w \geq 1.3$.



(a) A^{random} (b) A^{degree} (c) $A^{central}$ (d) $A^{community}$

Fig. 6. Changes in the average degree, the size of LCC and the average connectivity for the mainnet of Lightning Network under four attack strategies and three defense strategies with varying w (R_d: random defense, P: preferential defense, B: balanced defense).

4.4. Connectivity with varying k_d

We finally discuss the effects of the number of newly recruited nodes K_d in the defense phases. Fig. 7 shows the effects of varying K_d from 7 to 13 after 100 rounds when $K_a = 10$ and $w = 1.0$.

Unsurprisingly, an increase of K_d is effective except D^{prefer} to enhance the robustness of the Lightning Network structure. As shown in Fig. 7, all the connectivity metrics are *linearly* increased with K_d when either D^{random} or $D^{balance}$ was used for defense. Interestingly, D^{random} produces results comparable with $D^{balance}$ except for $A^{community}$. This implies that a more straightforward defense strategy is increasing the number of new network nodes rather than deploying a complicated defense strategy requiring the knowledge about the network topology. Therefore, we need to develop a mechanism to encourage users to increase the number of connections as many as possible. A simple solution is to enforce a policy so that every new node connects to at least a minimum number of peer nodes. Also, we can provide financial incentives to nodes when processing Lightning Network transactions based on the number of connections in the network.

However, if we deploy D^{prefer} as defense strategy, increasing the number of new nodes would not be helpful against the intentional targeting attacks (i.e., A^{degree}, $A^{central}$, and $A^{community}$). This shows that a network that grow by preferential attachment can finally acquire a power-law distribution of vertex order that in turn makes the network vulnerable to attacks targeted on high degree nodes.

5. Related work

Network robustness is defined as the measure to which a network can function in the presence of nodes/edges failures. Albert et al. [7] analyzed network robustness of various technological networks such as the Internet, the electrical power grid, and transportation networks and found that those networks are robust to random failures but vulnerable to targeted attacks. Holme et al. [19] presented similar experiment results by performing simulations to selectively remove edges, and also suggested using centrality as an alternative to degree for targeting. Zhao et al. [20] studied the circumstances under which a scale-free network suffers cascading breakdown caused by the successive failures of hub nodes.

Nagaraja and Anderson [8] introduced a framework based on evolutionary game theory to explore the dynamic interaction of iterated attack and defense strategies over multiple periods. The suggested framework consists of an attacker whose objective is to maximize disruption to the network so that the network connectivity or efficiency decreases and a defender who tries to minimize the disruption by deploying new resources. According to the strategies of attacker and

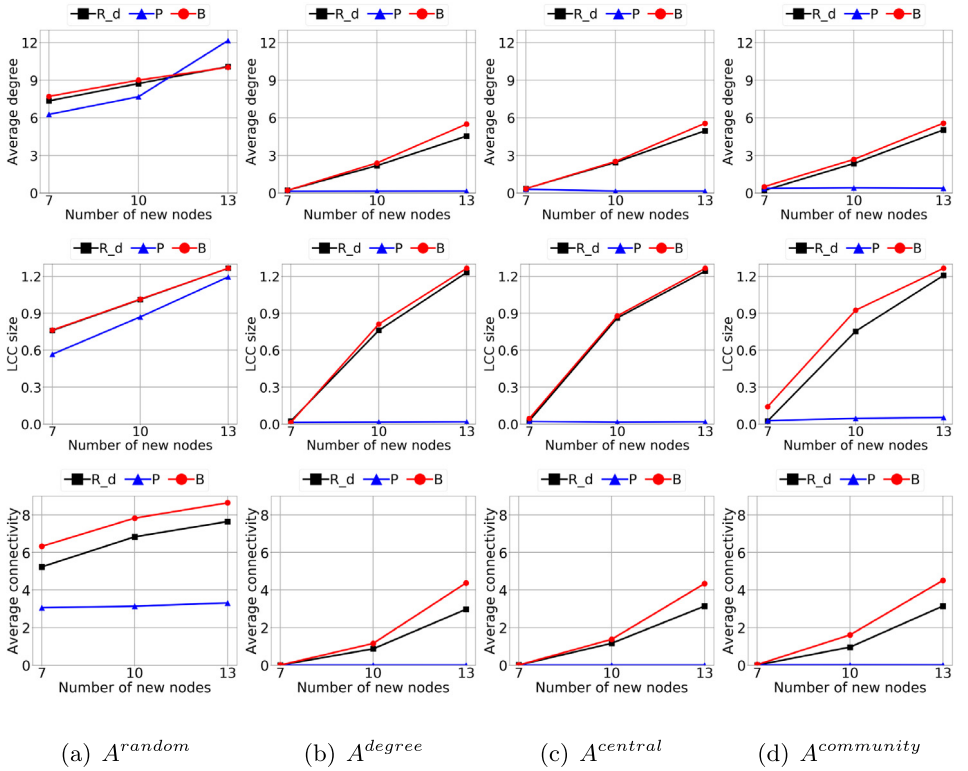


Fig. 7. Changes in the average degree, the size of LCC and the average connectivity for the mainnet of Lightning Network under four attack strategies and three defense strategies with varying K_d (R_d: random defense, P: preferential defense, B: balanced defense).

defender, the attacker picks k_a nodes and removes them in attack phase and the defender adds k_d nodes with m edges in defense phase. For multiple rounds, the attack phase and the defense phase are iterated sequentially. This framework thus provides a methodology to analyze defense and attack in networks where topology matters based on the evolutionary game theory.

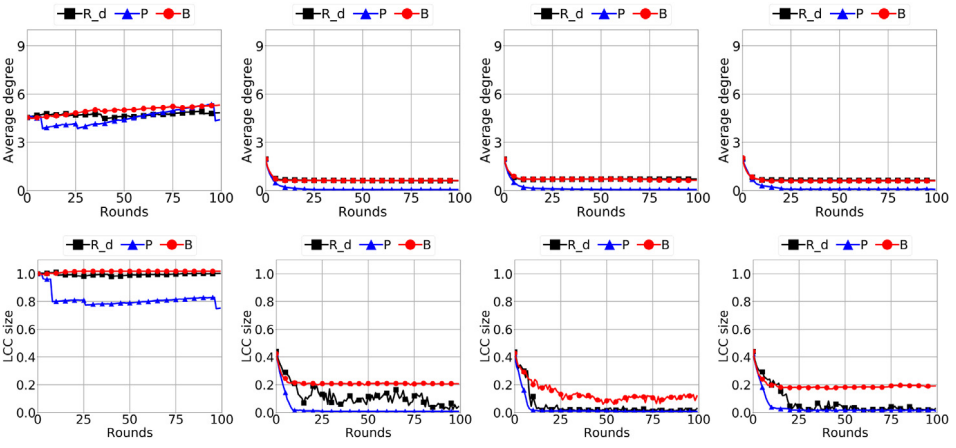
The framework in [8] is extended into a more generalized model in [9], because the original framework does not model the costs of creating new nodes and edges realistically so that the defender only creates a fixed number of newly recruited nodes and an arbitrary number of edges. So, they varied the budgets to limit newly added nodes and their connections and showed the correlation between network density and resilience to random failures or attacks. Also, they analyzed the efficiency of several attack and defense strategies on various well-known networks including real-world networks. The strategies they used to select nodes in attack and defense phases had approaches based on the properties of individual node such as the degree of a node or the centrality of a node. Wu et al. [21] also introduced a theoretical framework for attack information and analyzed the impact of attack information on the structural robustness of scale-free networks.

Recently, András et al. [22] analyzed the scale network properties of Bitcoin Lightning Network and demonstrated that Bitcoin Lightning Network would be vulnerable to target attacks. In this paper, we extend their work to specifically investigate the effectiveness of various attack and defense strategies on a real-world network structure constructed by the Lightning Network protocol.

6. Conclusion

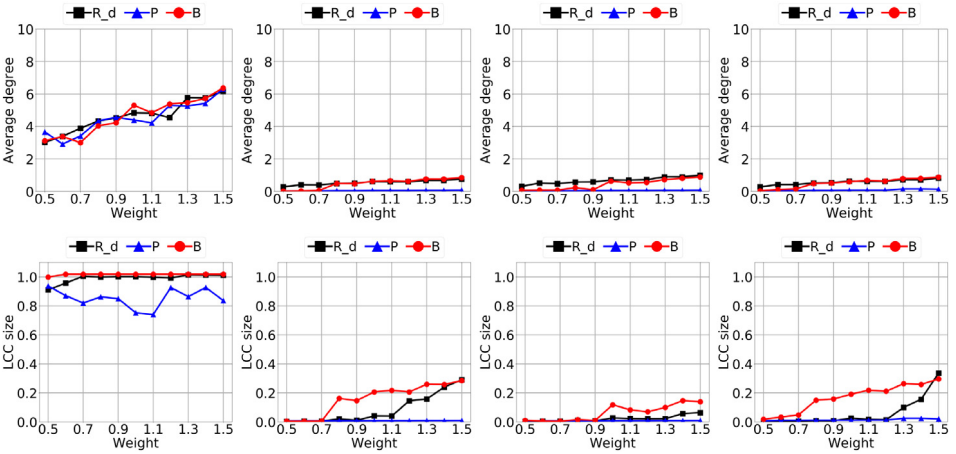
In this paper, we experimentally analyze the robustness of Lightning Network using a framework to evaluate attack and defense strategies on network topology. We found that Lightning Network has scale-free properties in which the number of edges at a given node have distributions that decay with power law tails. To evaluate the robustness of Lightning Network, we considered four attack strategies (random attack, high-degree attack, high-centrality attack, and community-based attack) and three defense strategies (random defense, preferential defense, and balanced defense) using the network characteristics. In summary, our key observations are as follows:

- The real-world Lightning Network is highly centralized, which could be vulnerable to targeted DDoS attacks (A^{degree}, $A^{central}$, and $A^{community}$). Even if the network has grown continuously up with new nodes and connections over time, Lightning Network would still be vulnerable to target attacks because it exhibits preferential connectivity.



(a) A^{random} (b) A^{degree} (c) $A^{central}$ (d) $A^{community}$

Fig. A.8. Changes in the average degree, the size of largest connected component and the average size of CC for the testnet over rounds (R_d: random defense, P: preferential defense, B: balanced defense).

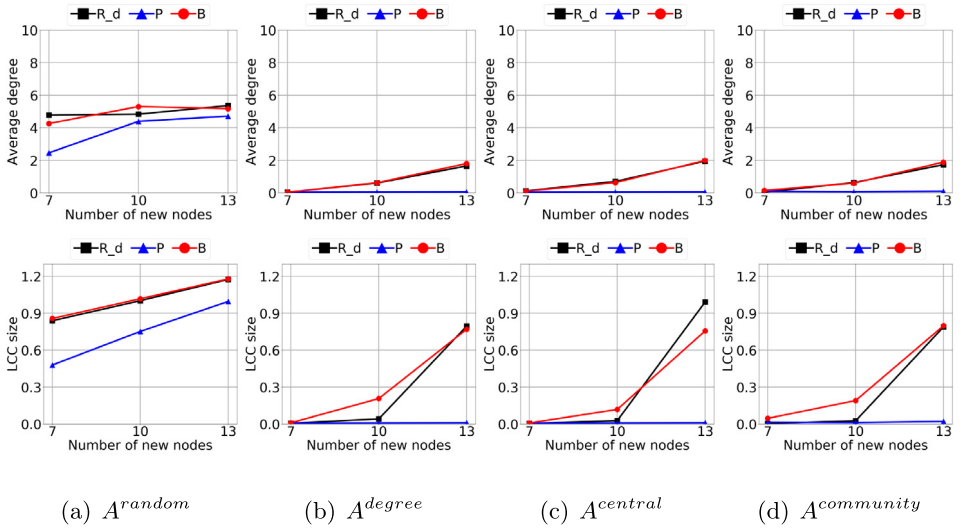


(a) A^{random} (b) A^{degree} (c) $A^{central}$ (d) $A^{community}$

Fig. A.9. Changes in the average degree and the size of LCC for the testnet of Lightning Network under four attack strategies and three defense strategies with varying w (R_d: random defense, P: preferential defense, B: balanced defense).

- The best attack strategy is "high degree attack", A^{degree}, against $D^{balance}$ and D^{random} in terms of all measures (the average degree, the size of LCC, and the average connectivity). This implies that attackers can easily implement the best attack strategy without the knowledge about the whole network topology.
- For defense, our recommendation would be "balanced defense", $D^{balance}$, by connecting new nodes to low centrality nodes. Overall, $D^{balance}$ produced slightly better results than D^{random}. In particular, $D^{balance}$ is significantly more effective than D^{random} against A^{degree} and $A^{community}$.
- For target attacks (A^{degree}, $A^{central}$, and $A^{community}$), the average node degree always converges to around 2.5 over rounds against $D^{balance}$ or D^{random}.

In this paper, we used a simplified model to analyze topological weaknesses of the network structure for Bitcoin Lightning Network. However, our such abstraction may miss some important parameters to understand main characteristics of Lightning Network. In future work, we plan to develop a better model to analyze topological characteristics of Lightning Network with other important parameters (e.g., channel capacity and network transmission time).



(a) A^{random} (b) A^{degree} (c) $A^{central}$ (d) $A^{community}$

Fig. A.10. Changes in the average degree and the size of LCC for the testnet of Lightning Network under four attack strategies and three defense strategies with varying k_d (R_d: random defense, P: preferential defense, B: balanced defense).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported by the ICT R&D program of MSICT/IITP, South Korea. [2017-0-00045, Hyper-connected Intelligent Infrastructure Technology Development].

Appendix. Effectiveness of attack and defense strategies for the testnet in Lightning Network

See Figs. A.8–A.10.

References

[1] S. Nakamoto, Bitcoin: A peer-to-peer electronic Cash system, 2008.
[2] 99Bitcoins, Who Accepts Bitcoins As Payment? List of Companies, 2018, Online (Accessed 3 December 2018).
[3] A. Magazine, Massive worldwide cryptocurrency adoption is about to take place, 2018, Online (Accessed 3 December 2018).
[4] VISA, Stress test prepares VisaNet for the most wonderful time of the year, 2013, Online (Accessed 5 December 2018).
[5] J. Poon, T. Dryja, The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments, Technical report, 2015.
[6] A.-L. Barabási, R. Albert, H. Jeong, Scale-free characteristics of random networks: the topology of the world-wide web, Physica A 281 (1–4) (2000) 69–77.
[7] R. Albert, H. Jeong, A.L. Barabási, Error and attack tolerance of complex networks, Nature 406 (2000) 378–382.
[8] S. Nagaraja, R. Anderson, Topology of covert conflict, in: Proceedings of the 5th Workshop on the Economics of Information Security, 2006.
[9] H. Kim, R. Anderson, An experimental evaluation of robustness of networks, IEEE Syst. J. 7 (2013) 179–188.
[10] Trustnodes, Lightning Network DDoS Sends 20% of Nodes Down, 2018, https://www.trustnodes.com/2018/03/21/lightning-network-ddos-sends-20-nodes, Online (Accessed 22 November 2018).
[11] M. Girvan, M.E.J. Newman, Community structure in social and biological networks, Natl. Acad. Sci. 99 (2002) 7821–7826.
[12] M.A. Porter, J.P. Onnela, P.J. Mucha, Communities in networks, Not. AMS 56 (2009) 1082–1166.
[13] S. Fortunato, C. Castellano, Community structure in graphs, Comput. Complexity 2012 (2012) 490–512.
[14] V.D. Blondel, J.L. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks, J. Stat. Mech. Theory Exp. 2008 (2008) P1008.
[15] Lightning Network Explorer, 2018, https://lnmainnet.gaben.win, Offline (Accessed 15 June 2018).
[16] Lightning Network Explorer [TESTNET], 2018, https://explorer.acinq.co, Online (Accessed 22 November 2018).
[17] P. Hage, F. Harary, Eccentricity and centrality in networks, Soc. Netw. 17 (1995) 57–63.
[18] J. Dong, S. Horvath, Understanding network concepts in modules, BMC Syst. Biol. 1 (2007) 24–43.
[19] P. Holme, B.J. Kim, C.N. Yoon, S.K. Han, Attack vulnerability of complex networks, Phys. Rev. E 65 (5) (2002) 056109.
[20] L. Zhao, K. Park, Y.-C. Lai, Attack vulnerability of scale-free networks due to cascading breakdown, Phys. Rev. E 70 (3) (2004) 035101.
[21] J. Wu, S.-Y. Tan, Z. Liu, Y.-J. Tan, X. Lu, Enhancing structural robustness of scale-free networks by information disturbance, Sci. Rep. 7 (1) (2017).
[22] I.A. Seres, L. Gulyás, D.A. Nagy, P. Burcsi, Topological analysis of bitcoin's lightning network, 2019, CoRR, abs/1901.04972.