International Journal of Mathematical, Engineering and Management Sciences
Vol. 6, No. 2, 469-479, 2021
https://doi.org/10.33889/IJMEMS.2021.6.2.028

Dependability Analysis of Bitcoin subject to Eclipse Attacks

Chencheng Zhou
Department of Electrical and Computer Engineering,
University of Massachusetts, Dartmouth, MA, USA.
E-mail: czhou@umassd.edu

Liudong Xing
Department of Electrical and Computer Engineering,
University of Massachusetts, Dartmouth, MA, USA.
Corresponding author: lxing@umassd.edu

Qisi Liu
Department of Electrical and Computer Engineering,
University of Massachusetts, Dartmouth, MA, USA.
E-mail: qliu1@umassd.edu

(Received July 16, 2020; Accepted September 11, 2020)

Abstract
The immense potential of the blockchain technology in diverse and critical applications (e.g., financial services, cryptocurrencies, supply chains, smart contracts, and automotive industry) has led to a new challenge: the dependability modeling and analysis of the blockchain-based systems. In this paper, we model the Bitcoin, a peer-to-peer cryptocurrency system built on the blockchain technology that allows individuals to trade freely without involving banks or other intermediate agents. We analyze the dependability of the Bitcoin system subject to the Eclipse attack. A continuous-time Markov chain-based method is suggested to model the system behavior under the Eclipse attack and further quantify the dependability of the Bitcoin system. The effects of several model parameters (related to the miner's habits in system protection, restart, and mining frequency) on the system dependability are demonstrated through numerical examples. Findings from this work may provide effective guidelines in designing a resilient and robust Bitcoin system.

Keywords- Bitcoin, Blockchain, Eclipse attack, Markov chain, Dependability analysis.

1. Introduction

As one of the most revolutionary invention in the computer science world, the block chain technology has received extensive attention from academia, industries and governments in the past decade (Atzei et al., 2017; Dai et al., 2019; Ferrag et al., 2018; Kang et al., 2018). It has great potential in various critical applications (e.g., financial services, smart contracts, supply chains, voting and the Internet of Things), contributing to the transformation of human society towards smart, efficient, and resilient (Akbari et al., 2017; Frizzo-Barker et al., 2020; Garay et al., 2017; Xing, 2020, 2021). Bitcoin is known as the peer-to-peer cryptocurrency system built on the block chain technology (Satoshi, 2008). Compared with the fiat currency, the Bitcoin is featured with a decentralized property, which allows individuals to trade freely without involving banks. Bitcoin has been widely used in different areas and has a market cap of 185 billion. However, the whole Bitcoin network is vulnerable to a variety of threats or attacks.

For example, an attacker can control the availability of the block chain data by creating incorrect or illegal access to the data. To do so, the attacker only needs to track the correspondence of different addresses including the IP address and the Bitcoin address (Koshy et al., 2014). An attacker can easily track the relationship between addresses from transactions by taking advantage of the open network of the Bitcoin system. Thus, the users' privacy may be exposed and their personal information is in danger (Reid and Harrigan, 2013). An attacker can also temper the block chain data by attacking the block chain consensus mechanism (Bag et al., 2016). The vulnerability of a smart contract can cause serious damages to the network if the attacker takes over the control of the block chain data from the user. In addition to the above examples, the Bitcoin system is facing many other security problems, such as Sybil attacks (Zhang and Lee, 2019), Selfish mining (Eyal and Sirer, 2014), mining pool attacks (Bahack, 2013), re-identification attacks (Meiklejohn et al., 2013), miner attacks (Rosenfeld, 2011) and Crypto Locker based attacks (Liao et al., 2016) (a family of ransomware that encrypts a victim's files until a ransom is paid). Considerable research efforts have been expended in studying the Bitcoin security against those attacks.

For example, in Eyal and Sirer (2014) a mitigation strategy based on a practical modification to the Bitcoin protocol was suggested to defend the Bitcoin system against the concluding selfish mining attack. In Gervais et al. (2015), it was shown that an attacker can delay the propagation of Bitcoin transactions in multiple ways to a specific node. Different countermeasures were explored to improve the security of the network, including dynamic timeouts, updating block advertisements and penalizing non-responding nodes. In Biryukov and Pustogarov (2015a, 2015b), it was examined and proved that the Bitcoin over Tor system is not promising in solving the security problem. In Bamert et al. (2014), a hardware token was proposed to secure the sign Bitcoin transaction. In Ben-Sasson et al. (2014) and Monaco (2015), the Bitcoin's weakness in privacy protection was examined and a decentralized anonymous payment scheme was proposed for providing privacy protection. In Kroll et al. (2013), it was shown that there are infinitely many Nash equilibria for mining strategies and argued requirements of the governance structures. In Joux (2004), it was shown that the difficulty of finding simultaneous collisions are not higher than finding individual ones in multiple hash functions. In Bastiaan (2015), the threat to the Bitcoin network from the pool mining was discussed and the Markov Chain was applied for the stochastic analysis of the two phase proof-of-work (2P-POW) including an average reward for pools under different difficulties. In Göbel et al. (2016), the Markov Chain model was applied to demonstrate the possible detection of block-hiding attacks (selfish mining) by monitoring the production rate of orphan blocks. Existing research works mostly focus on the detection of possible threats and estimation of effects from the malicious behavior (protocol or encryption-wise). To the best of our knowledge, no systematic efforts have been dedicated to quantitative dependability analysis of the Bitcoin system and investigating impacts from parameters related to miner's habits.

In this paper we make contributions by modeling and analyzing the dependability of the Bitcoin system subject to Eclipse attacks. The process and mechanism of the Eclipse attack are examined. A continuous-time Markov Chain (CTMC)-based method is applied to model the behavior of the Bitcoin system under the Eclipse attack and quantify the dependability of the Bitcoin system. Numerical examples are provided to demonstrate the effects of different parameters reflecting the miner's habits in system protection, restart, and mining frequency on the Bitcoin dependability.

The rest of the paper is arranged as follows: Section 2 examines the working mechanism of the Eclipse attack. Section 3 identifies key states of the Bitcoin system under the Eclipse attack and presents the CTMC-based method for the dependability modeling and analysis of the Bitcoin system. Section 4 presents example analysis results and examines the effects of several model parameters on the Bitcoin dependability. Section 5 draws the conclusion of our study and discusses future research directions.

2. The Eclipse Attack

During the Eclipse attack, an attacker aims to control the information flow of a victim node including reception and transmission so that the victim node loses its connection to other legitimate nodes. To do so, the attacker node maliciously fills the victim node's routing table before the victim node of the block chain restarts. The victim node can be forced to restart, or the attacker can simply wait for the victim node to restart. After the restart, the victim node establishes an outgoing connection with the attack address in the routing table. At the same time, the attacker node continuously establishes an incoming connection with the victim node. Finally, the information flow channel of the victim node is monopolized or controlled so that the victim node can only receive fake or even malicious information sent by the attacker node (Heilman et al., 2015). Figure 1 gives the flowchart of a successful Eclipse attack.

If the attacker node can successfully implement Eclipse attacks on more nodes, it can control the block chain channels and information flows of more nearby nodes, and gradually control most of the block chain network. Thus, a successful Eclipse attack can result in other attacks like double-spending, selfish mining, and block withholding (Heilman et al., 2015).

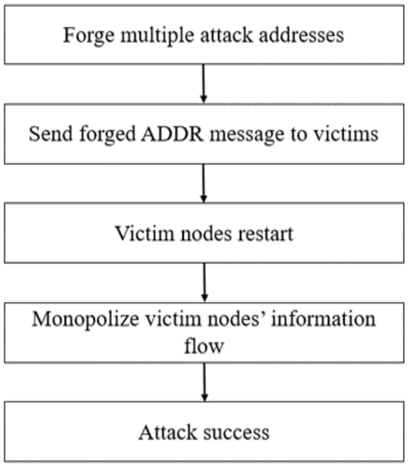
Forge multiple attack addresses
Send forged ADDR message to victims
Victim nodes restart
Monopolize victim nodes' information flow
Attack success

Figure 1. Flowchart of a successful Eclipse attack.

3. Dependability Modeling and Analysis

In this section, we apply the CTMC to model and analyze the dependability of the Bitcoin system under the Eclipse attack. Figure 2 illustrates the state transition diagram in the CTMC-based solution. Based on the working mechanism of the Eclipse attack presented in Section 2, five key states are identified: 0 (original), 1 (table hacked), 2 (restart), 3 (connected), and 4 (monopolized).

Specifically, in the original state 0, the Bitcoin system functions normally without being impacted by the attack. Under state 0, an attack node can send an ADDR message containing a lot of "trash" IP addresses that will gradually overwrite all legal addresses of the node table, causing the system to transit from state 0 to state 1. The transition rate is λ_{01} (i.e., the table hackling rate). Under state 1, the victim node performs the restart causing the system to transit to state 2 with transition rate

λ_{12}. Under state 1, if the user detects and deletes the suspicious message, which contains forged addresses, then the system can go back to state 0 with μ_{10}. Under state 2 (the victim node has been restarted), the victim node can be connected to the attack addresses with rate λ_{23}, causing the system to transit to state 3. Under state 2, if the user cleans his/her node table with tools, the system can transit back to state 1 with μ_{21}. Under state 3, the victim node is forced to select an address from the hacked table to establish an outgoing connection, causing the system to transit to state 4 with rate λ_{34}. Under state 3, if the user successfully restores the healthy connection through some maintenance operation, the system can transit back to state 0 with μ_{30}. Under state 4, the attacker controls all incoming connections to the victim node, truly monopolizing the victim node; the Eclipse attack succeeds. In state 4, if the user detects the adversary connection and rebuilds partial connections with honest nodes, the system is able to transit back to state 3 with rate μ_{43}. Note that transition rates μ_{10}, μ_{21}, μ_{30}, μ_{43} are generally considered as recovery rates and their values are mostly associated with the miner/user's habits. In Section 4, the effects of some of these transition rates on the dependability of the Bitcoin system are investigated through numerical results.

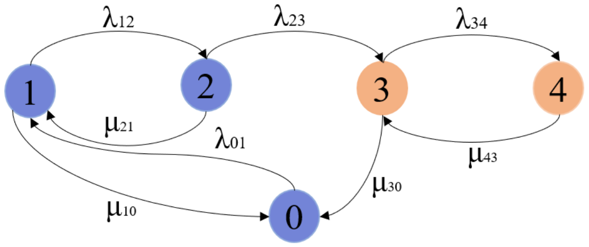

Figure 2. CTMC model of the Bitcoin under the Eclipse attack.

Based on the state transition diagram in Figure 2, the state equations in the matrix form are given in Eq. (1), where the left-most matrix is the transition rate matrix of the CTMC, $P_j(t)$ represents the probability of the Bitcoin system being in state *j* (*j*=0,1,2,3,4), and $\dot{P}_j(t)$ represents the derivative of the state *j* probability. Eq. (1) can also be detailed using separate differential equations as shown in Eqs. (2)-(6).

$$\begin{bmatrix} -\lambda_{01} & \mu_{10} & 0 & \mu_{30} & 0 \\ \lambda_{01} & -(\mu_{10}+\lambda_{12}) & \mu_{21} & 0 & 0 \\ 0 & \lambda_{12} & -(\mu_{21}+\lambda_{23}) & 0 & 0 \\ 0 & 0 & \lambda_{23} & -(\mu_{30}+\lambda_{34}) & \mu_{43} \\ 0 & 0 & 0 & \lambda_{34} & -\mu_{43} \end{bmatrix} \begin{bmatrix} P_0(t) \\ P_1(t) \\ P_2(t) \\ P_3(t) \\ P_4(t) \end{bmatrix} = \begin{bmatrix} \dot{P}_0(t) \\ \dot{P}_1(t) \\ \dot{P}_2(t) \\ \dot{P}_3(t) \\ \dot{P}_4(t) \end{bmatrix} \tag{1}$$

$$\dot{P}_0(t) = -\lambda_{01}P_0(t) + \mu_{10}P_1(t) + \mu_{30}P_3(t) \tag{2}$$

$$\dot{P}_1(t) = \lambda_{01}P_0(t) - (\mu_{10} + \lambda_{12})P_1(t) + \mu_{21}P_2(t) \tag{3}$$

$$\dot{P}_2(t) = \lambda_{12}P_1(t) - (\mu_{21} + \lambda_{23})P_2(t) \tag{4}$$

$$\dot{P}_3(t) = \lambda_{23}P_2(t) - (\mu_{30} + \lambda_{34})P_3(t) + \mu_{43}P_4(t) \tag{5}$$

$$\dot{P}_4(t) = \lambda_{34}P_3(t) - \mu_{43}P_4(t) \tag{6}$$

The system state probabilities can be obtained by aapplying the Laplace transform-based method to solve Eqs. (2)-(6) with the initial state probability $P_0(0) = 1$ and $\sum_{i=0}^{4} P_i(t) = 1$ (Xing et al., 2019). In particular, Laplace transforms of the five system state probabilities are obtained as:

$$P_0^*(s) = \frac{A}{A+B+C+D+E} \tag{7}$$

$$P_1^*(s) = \frac{B}{A+B+C+D+E} \tag{8}$$

$$P_2^*(s) = \frac{C}{A+B+C+D+E} \tag{9}$$

$$P_3^*(s) = \frac{D}{A+B+C+D+E} \tag{10}$$

$$P_4^*(s) = \frac{E}{A+B+C+D+E} \tag{11}$$

where,

$$A = \lambda_{12}*s^3 + \lambda_{23}*s^3 + \lambda_{34}* s^3 + s^3*\mu_{10} + s^3*\mu_{21} + s^3*\mu_{30} + s^3*\mu_{43} + s^4 + \lambda_{12}*\lambda_{23}*s^2 + \lambda_{12}*\lambda_{34}*s^2 + \lambda_{23}*\lambda_{34}*s^2 + \lambda_{23}*s^2*\mu_{10} + \lambda_{12}*s^2*\mu_{30} + \lambda_{34}*s^2*\mu_{10} + \lambda_{12}*s^2*\mu_{43} + \lambda_{23}*s^2*\mu_{30} + \lambda_{34}*s^2*\mu_{21} + \lambda_{23}*s^2*\mu_{43} + s^2*\mu_{10}*\mu_{21} + s^2*\mu_{10}*\mu_{30} + s^2*\mu_{10}*\mu_{43} + s^2*\mu_{21}*\mu_{30} + s^2*\mu_{21}*\mu_{43} + s^2*\mu_{30}*\mu_{43} + \lambda_{12}*\lambda_{23}*\lambda_{34}*s + \lambda_{12}*\lambda_{23}*s*\mu_{30} + \lambda_{23}*\lambda_{34}*s*\mu_{10} + \lambda_{12}*\lambda_{23}*s*\mu_{43} + \lambda_{12}*\lambda_{23}*\mu_{30}*\mu_{43} + \lambda_{23}*s*\mu_{10}*\mu_{30} + \lambda_{34}*s*\mu_{10}*\mu_{21} + \lambda_{23}*s*\mu_{10}*\mu_{43} + \lambda_{12}*s*\mu_{30}*\mu_{43} + \lambda_{23}*s*\mu_{30}*\mu_{43} + \lambda_{23}*\mu_{10}*\mu_{30}*\mu_{43} + s*\mu_{10}*\mu_{21}*\mu_{30} + s*\mu_{10}*\mu_{21}*\mu_{43} + s*\mu_{10}*\mu_{30}*\mu_{43} + s*\mu_{21}*\mu_{30}*\mu_{43} + \mu_{10}*\mu_{21}*\mu_{30}*\mu_{43}$$

$$B = \lambda_{01}*s^3 + \lambda_{01}*\lambda_{23}*s^2 + \lambda_{01}*\lambda_{34}*s^2 + \lambda_{01}*s^2*\mu_{21} + \lambda_{01}*s^2*\mu_{30} + \lambda_{01}*s^2*\mu_{43} + \lambda_{01}*\lambda_{23}*\lambda_{34}*s + \lambda_{01}*\lambda_{23}*s*\mu_{30} + \lambda_{01}*\lambda_{34}*s*\mu_{21} + \lambda_{01}*\lambda_{23}*s*\mu_{43} + \lambda_{01}*\lambda_{23}*\mu_{30}*\mu_{43} + \lambda_{01}*s*\mu_{21}*\mu_{30} + \lambda_{01}*s*\mu_{21}*\mu_{43} + \lambda_{01}*s*\mu_{30}*\mu_{43} + \lambda_{01}*\mu_{21}*\mu_{30}*\mu_{43}$$

$$C = \lambda_{01}*\lambda_{12}*s^2 + \lambda_{01}*\lambda_{12}*\lambda_{34}*s + \lambda_{01}*\lambda_{12}*s*\mu_{30} + \lambda_{01}*\lambda_{12}*s*\mu_{43} + \lambda_{01}*\lambda_{12}*\mu_{30}*\mu_{43}$$

$$D = \lambda_{01}*\lambda_{12}*\lambda_{23}*(s + \mu_{43})$$

$$E = \lambda_{01}*\lambda_{12}*\lambda_{23}*\lambda_{34}.$$

We apply the inverse Laplace transform of $P_i^*(s)$ obtained in Eqs. (7)-(11) to derive the system state probabilities in the time domain $P_j(t)$ (j=0,1,2,3,4). This conversion is completed by Matlab in our study.

With the system state probabilities evaluated, the dependability of the Bitcoin system can be obtained as $D(\mathrm{t}) = P_0(t) + P_1(t) + P_2(t)$, which is the probability that the Bitcoin system can function normally. Under states 3 and 4, the Eclipse attack is considered successful; the Bitcoin system is infected and not dependable. Thus, we define $\overline{D}(t) = P_3(t) + P_4(t)$.

4. Example Analysis Results

This section illustrates the CTMC-based method and impacts of several model parameters on the Bitcoin dependability, gaining insights into the Eclipse attack and defense behaviors of the Bitcoin system.

Specifically, we examine the impacts of parameters (μ_{21}, μ_{30}, μ_{43}), λ_{12}, and λ_{34} on the system dependability using the parameter set as shown in Table 1. Parameters (μ_{21}, μ_{30}, μ_{43}) stand for the user's sense of system protection, and their effects are examined through parameter sets *a*, *b*, and *c* in Table 1. Parameter λ_{12} models the user's habits of turning off the system after using it, and its effects are estimated through parameter sets *d*, *b* and *e*. Parameter λ_{34} models the user's mining habits, and its effects are investigated using sets *f*, *b*, and *g* in Table 1.

Table 1. Model transition rate parameters (per hour).

Rate	Set a	Set b	Set c	Set d	Set e	Set f	Set g
μ_{10}	0.05	0.05	0.05	0.05	0.05	0.05	0.05
μ_{21}	0.01	0.12	0.38	0.12	0.12	0.12	0.12
μ_{30}	0.05	0.18	0.53	0.18	0.18	0.18	0.18
μ_{43}	0.02	0.16	0.45	0.16	0.16	0.16	0.16
λ_{01}	0.03	0.03	0.03	0.03	0.03	0.03	0.03
λ_{12}	0.25	0.25	0.25	0.05	0.65	0.25	0.25
λ_{23}	0.34	0.34	0.34	0.34	0.34	0.34	0.34
λ_{34}	0.16	0.16	0.16	0.16	0.16	0.04	0.56

4.1 Effects of Protection Parameters ($\boldsymbol{\mu_{21}}$, $\boldsymbol{\mu_{30}}$, $\boldsymbol{\mu_{43}}$)

Parameter sets *a*, *b* and *c* of Table 1 model different levels of the user's sense of system protection, from low to high. Specially, set *a* represents a user who has the least sense of system protection and rarely checks the status of the system health. Set *b* represents a normal user who checks the system status with a common frequency. Set *c* represents *a* user with a strong sense of system security and experiences in system protection.

Table 2 shows the dependability of the Bitcoin system under parameter sets *a*, *b*, and *c* for several values of mission time. Figure 3 shows the graphical representation of the dependability results.

Table 2. Dependability of the bitcoin system.

t (hrs)	Set a	Set b	Set c
6	0.999957	0.974337	0.988463
12	0.864533	0.923578	0.976648
18	0.755948	0.883587	0.971561
24	0.657072	0.857726	0.969674
30	0.570580	0.842074	0.969004
36	0.495490	0.832862	0.968770

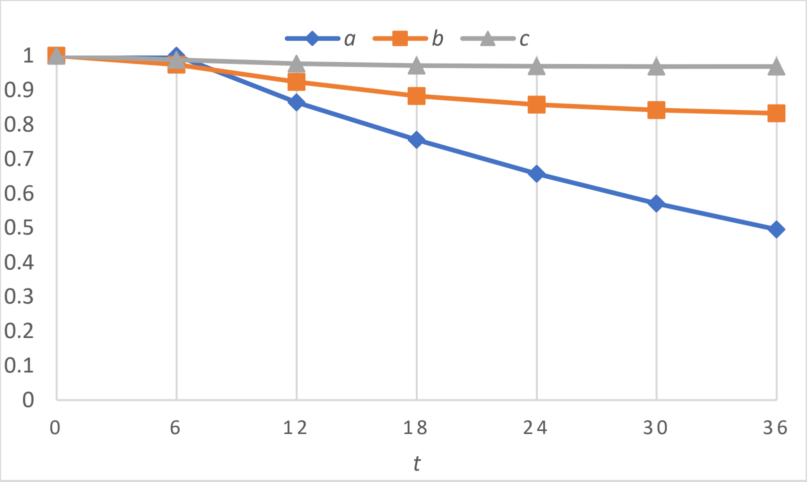

Figure 3. Effects of parameters μ_{21}, μ_{30}, μ_{43} on the system dependability.

From Figure 3, we can observe that the system dependability decreases with time. The system dependability *D* under set *a* (user with the least sense of system protection) is the lowest and decreases quickly with time; *D* under set *c* (experienced user with a strong sense of system protection) appears the highest and decreases slightly with time; *D* under set *b* appears in between the former two cases. The above results support the intuition that the system is more likely to stay in the dependable state when its user has a higher sense of system protection.

4.2 Effects of User Restart Habits Parameter λ_{12}

Effects of user's restart habits parameter λ_{12} are investigated through parameter sets *d*, *b*, and *e* in Table 1. Set *d* models a user who almost never turns off his/her computer after finish using it. Set *b* models a user who turns off the system with a certain frequency. Set *e* models a user who always shuts down the system (due to, e.g., strong security sense, economic concern, or obsessive-compulsive disorder). The system dependability results under set *d*, *b*, and *e* are presented in Table 3. Figure 4 gives the graphical representation of the dependability results.

Table 3. Dependability of the Bitcoin system.

t (hrs)	Set d	Set b	Set e
6	0.993312	0.974337	0.955837
12	0.975726	0.923578	0.891195
18	0.957196	0.883587	0.849035
24	0.941810	0.857726	0.824199
30	0.93030	0.842074	0.809803
36	0.922198	0.832862	0.801483

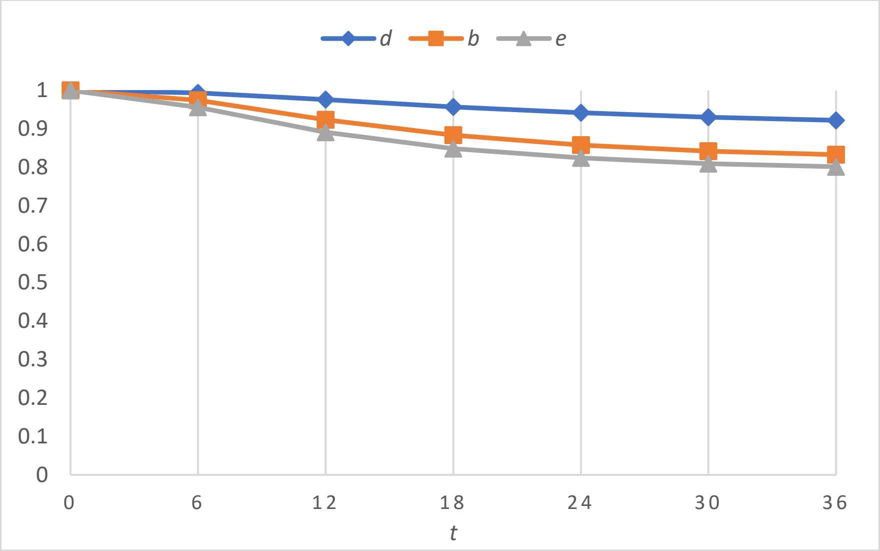

Figure 4. Effects of parameters λ_{12} on the system dependability.

Figure 4 shows that the system dependability D under set d is the highest and has the lowest decreasing speed as time proceeds; D under set e appears the lowest and decreases with the highest speed as time proceeds; D under set b appears in between the former two cases. The numerical results support the intuition that the system is more likely to stay in the dependable state when its user has a habit of not turning off the system after each use. In other words, the system is more likely to get compromised if its user shuts down and restarts the system more frequently. This is due to the special mechanism of the Eclipse attack, which requires the system to reboot to complete the malicious attack.

4.3 Effects of Mining Frequency Parameter λ_{34}

Effects of the mining frequency parameter λ_{34} are investigated through parameter sets *f*, *b*, and *g* in Table 1. Set *f* corresponds to an amateur miner or a beginner level miner who rarely does mining. Set *b* corresponds to a normal miner who does average mining. Set *g* corresponds to a frequent miner who would do mining almost all the time. Table 4 shows the system dependability results under sets *f*, *b* and *g*. Figure 5 gives the graphical representation of the dependability results.

Table 4. Dependability of the Bitcoin system.

***t* (hrs)**	**Set f**	**Set b**	**Set g**
6	0.975206	0.974337	0.972357
12	0.931627	0.923578	0.908163
18	0.904076	0.883587	0.846277
24	0.890514	0.857726	0.797295
30	0.884375	0.842074	0.760980
36	0.881670	0.832862	0.734729

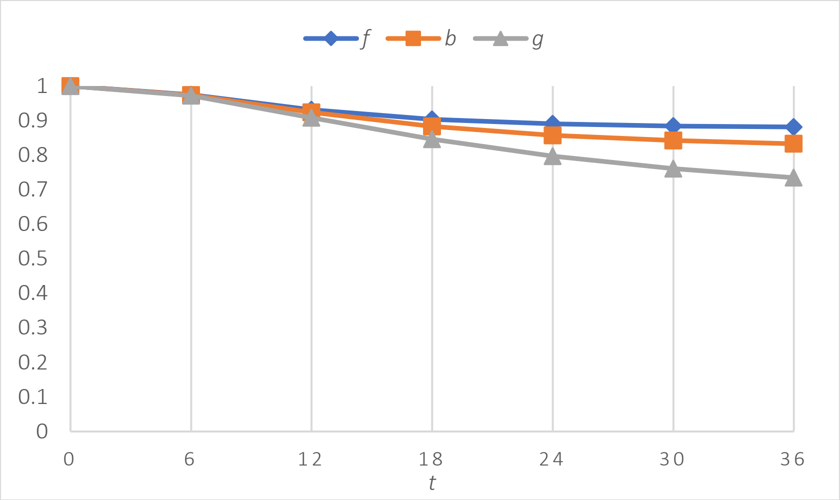

Figure 5. Effects of parameters λ_{34} on the system dependability.

Figure 5 shows that the system dependability *D* under set *f* is the highest and has the slowest decreasing speed as time proceeds among the three cases studied; *D* under set *g* appears the lowest and decreases more quickly with time than the other two cases; *D* under set *b* appears in between the former two cases. From these results we can draw a conclusion that a miner with a higher frequency of mining has a relatively higher chance of being exposed to the attack/risk, leading to lower system dependability.

5. Conclusion and Future Work

An Eclipse attack to a Bitcoin system is a network-level attack, where an attacker essentially takes control of the peer-to-peer network blocking a node’s view of the block chain. The existing works on the Bitcoin security risk are mostly based on protocol and encryption with a focus on threat detection or evaluation of impacts from the malicious behaviour. This paper advances the state of the art by performing the quantitative dependability analysis of the Bitcoin system undergoing the eclipse attack using the CTMC-based method. We further investigate the impacts of several parameters related to the miner’s habits in system protection, restart, and mining frequency on the system dependability through numerical studies.

The CTMC-based method is limited to constant transition rates (or exponentially distributed state transition time). In the future we plan to explore semi-Markov models to relax this limitation for the Bitcoin dependability analysis. We also plan to extend the proposed method to investigate other types of attacks on the Bitcoin network such as selfish mining (Yang et al., 2020) and block withholding mining (Qin et al., 2020). Another direction is to design resilience algorithms and methods that can improve the robustness of the current Bitcoin network model and strengthen its immunity to various threats.

Conflict of Interest
The authors confirm that there is no conflict of interest to declare for this publication.

Acknowledgments
This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. The authors would like to thank the editor and anonymous reviewers for their comments that help improve the quality of this work.

References

Akbari, E., Wu, Q., Zhao, W., Arabnia, H.R., & Yang, M.Q. (2017). From blockchain to internet-based Voting. In *2017 International Conference on Computational Science and Computational Intelligence (CSCI)* (pp. 218-221). IEEE. Las Vegas, USA.

Atzei, N., Bartoletti, M., & Cimoli, T. (2017). A survey of attacks on ethereum smart contracts (sok). In *International Conference on Principles of Security and Trust* (pp. 164-186). Springer, Berlin, Heidelberg.

Bag, S., Ruj, S., & Sakurai, K. (2016). Bitcoin block withholding attack: analysis and mitigation. *IEEE Transactions on Information Forensics and Security*, *12*(8), 1967-1978.

Bahack, L. (2013). Theoretical bitcoin attacks with less than half of the computational power (draft). *arXiv preprint arXiv:1312.7013*.

Bamert, T., Decker, C., Wattenhofer, R., & Welten, S. (2014). Bluewallet: the secure bitcoin wallet. In *International Workshop on Security and Trust Management* (pp. 65-80). Springer, Cham, Switzerland.

Bastiaan, M. (2015). Preventing the 51%-attack: a stochastic analysis of two phase proof of work in Bitcoin. *22nd Twente Student Conference on IT* (pp. 1-10). Enschede, the Netherlands. https://fmt.ewi.utwente.nl/media/175.pdf, Accessed in August 2020.

Ben-Sasson, E., Chiesa, A., Garman, C., Green, M., Miers, I., Tromer, E., & Virza, M. (2014). Zerocash: Decentralized anonymous payments from bitcoin. In *2014 IEEE Symposium on Security and Privacy* (pp. 459-474). IEEE. San Jose, CA, USA.

Biryukov, A., & Pustogarov, I. (2015a). Bitcoin over Tor isn't a good idea. In *2015 IEEE Symposium on Security and Privacy* (pp. 122-134). IEEE. San Jose, CA, USA.

Biryukov, A., & Pustogarov, I. (2015b). Proof-of-work as anonymous micropayment: rewarding a Tor relay. In *International Conference on Financial Cryptography and Data Security* (pp. 445-455). Springer, Berlin, Heidelberg.

Dai, H.N., Zheng, Z., & Zhang, Y. (2019). Blockchain for Internet of Things: a survey. *IEEE Internet of Things Journal*, *6*(5), 8076-8094.

Eyal, I., & Sirer, E.G. (2014). Majority is not enough: bitcoin mining is vulnerable. In *International Conference on Financial Cryptography and Data Security* (pp. 436-454). Springer, Berlin, Heidelberg.

Ferrag, M.A., Derdour, M., Mukherjee, M., Derhab, A., Maglaras, L., & Janicke, H. (2018). Blockchain technologies for the internet of things: Research issues and challenges. *IEEE Internet of Things Journal*, *6*(2), 2188-2204.

Frizzo-Barker, J., Chow-White, P.A., Adams, P.R., Mentanko, J., Ha, D., & Green, S. (2020). Blockchain as a disruptive technology for business: a systematic review. *International Journal of Information Management*, *51*, 102029.

Garay, J., Kiayias, A., & Leonardos, N. (2017). The Bitcoin backbone protocol with chains of variable difficulty. In *Annual International Cryptology Conference* (pp. 291-323). Springer, Cham, Switzerland.

Gervais, A., Ritzdorf, H., Karame, G.O., & Capkun, S. (2015). Tampering with the delivery of blocks and transactions in bitcoin. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security* (pp. 692-705). Denver, USA.

Göbel, J., Keeler, H.P., Krzesinski, A.E., & Taylor, P.G. (2016). Bitcoin blockchain dynamics: the selfish-mine strategy in the presence of propagation delay. *Performance Evaluation*, *104*, 23-41.

Heilman, E., Kendler, A., Zohar, A., & Goldberg, S. (2015). Eclipse attacks on bitcoin's peer-to-peer network. In *24th USENIX Security Symposium* (pp. 129-144). Washington D.C., USA.

Joux, A. (2004). Multicollisions in iterated hash functions. Application to cascaded constructions. In *Annual International Cryptology Conference* (pp. 306-316). Springer, Berlin, Heidelberg.

Kang, J., Yu, R., Huang, X., Wu, M., Maharjan, S., Xie, S., & Zhang, Y. (2018). Blockchain for secure and efficient data sharing in vehicular edge computing and networks. *IEEE Internet of Things Journal*, *6*(3), 4660-4670.

Koshy, P., Koshy, D., & McDaniel, P. (2014). An analysis of anonymity in bitcoin using p2p network traffic. In *International Conference on Financial Cryptography and Data Security* (pp. 469-485). Springer, Berlin, Heidelberg.

Kroll, J.A., Davey, I.C., & Felten, E.W. (2013). The economics of bitcoin mining, or bitcoin in the presence of adversaries. In *Proceedings of WEIS* (Vol. 2013, p. 11). Washington D.C., USA.

Liao, K., Zhao, Z., Doupé, A., & Ahn, G.J. (2016). Behind closed doors: measurement and analysis of CryptoLocker ransoms in bitcoin. In *2016 APWG Symposium on Electronic Crime Research (eCrime)* (pp. 1-13). IEEE. Toronto, Canada.

Meiklejohn, S., Pomarole, M., Jordan, G., Levchenko, K., McCoy, D., Voelker, G.M., & Savage, S. (2013). A fistful of bitcoins: characterizing payments among men with no names. In *Proceedings of the 2013 Conference on Internet Measurement Conference* (pp. 127-140). Barcelona, Spain.

Monaco, J.V. (2015). Identifying bitcoin users by transaction behavior. In *Biometric and Surveillance Technology for Human and Activity Identification XII* (Vol. 9457, p. 945704). International Society for Optics and Photonics. Baltimore, United States.

Qin, R., Yuan, Y., & Wang, F.Y. (2020). Optimal block withholding strategies for blockchain mining pools. *IEEE Transactions on Computational Social Systems*, *7*(3), 709-717, doi: 10.1109/TCSS.2020.2991097.

Reid, F., & Harrigan, M. (2013). An analysis of anonymity in the bitcoin system. In Altshuler Y., Elovici Y., Cremers A.B., Aharony N., & Pentland A. (eds) *Security and Privacy in Social Networks*. Springer, New York, pp. 197-223.

Rosenfeld, M. (2011). Analysis of bitcoin pooled mining reward systems. *arXiv preprint arXiv:1112.4980*.

Satoshi, N. (2008). Bitcoin: a peer-to-peer electronic cash system. *Consulted*, *1*(2012), 28.

Xing, L. (2020). Reliability in internet of things: current status and future perspectives. *IEEE Internet of Things Journal*, *7*(8), 6704-6721.

Xing, L. (2021). Cascading failures in internet of things: review and perspectives on reliability and resilience. *IEEE Internet of Things Journal*, *8*(1), 44-64. doi: 10.1109/JIOT.2020.3018687.

Xing, L., Levitin, G., & Wang, C. (2019). *Dynamic system reliability: modeling and analysis of dynamic and dependent behaviors*. John Wiley & Sons.

Yang, R., Chang, X., Mišić, J., & Mišić, V.B. (2020). Assessing blockchain selfish mining in an imperfect network: honest and selfish miner views. *Computers & Security*, *97*, 101956.

Zhang, S., & Lee, J.H. (2019). Double-spending with a sybil attack in the bitcoin decentralized network. *IEEE Transactions on Industrial Informatics*, *15*(10), 5715-5722.