JIST Journal of Information Science and Technology
Volume 11, NO 2 | JUL – DEC 2021 | 1-9
ISSN: 2651-1053 (Online)
ISSN: 1906-9553 (Print)

Prevention of Cryptocurrency Loss from Wrong Receiver Address

Nanta Janpitak[1] and Chanboon Sathitwiriyawong [2]*
[1] Faculty of Engineering at Sriracha, Kasetsart University Sri Racha Campus
[2] Faculty of Information Technology, King Mongkut's Institute of Technology Ladkrabang

Received: March 19, 2021; Revised: Jume 04, 2021; Accepted: July 21, 2021; Published: July 27, 2021

ABSTRACT – Digital money such as Bitcoin, Ethereum and Litecoin are blockchain-based cryptocurrencies. In today's financial age, many people are holding and trading cryptocurrencies instead of fiat money or other assets. To transfer money in the centralized banking application, the valid receiver will be checked before enable transaction to be process. To transfer cryptocurrency in blockchain network, there are steps to verify the transactions by the blockchain nodes instead of the central authority. However, most of the verification processes are based on sender's properties such as sender's digital signature and sender's account balance compared to the sending amount, etc. There is no process to verify whether the receiver address is valid or not. From this vulnerability, there is a risk that the sender may input the wrong receiver address. If this mistake happens and the transaction has been confirmed, the sender will lose his cryptocurrency without recovery option. The amount will be transfer to that receiving address, but we cannot know that it is belong to whom or may not belong to anybody. There are many topics in cryptocurrency forum and news that the owner cried about losing of their cryptocurrency. Many of them had mistaken input in receiver address. Up until now, there is no mechanism to protect this mistake. In this paper, we propose a process to verify the receiver address using the blockchain API before constructing and submitting the transaction in blockchain application. If the receiver address does not exist in the same blockchain network, there is a process to notify the sender. This process can prevent the loss of cryptocurrency not only from wrong receiver addresses but also from different network addresses.

KEYWORDS: Blockchain, Bitcoin, Ethereum, Cryptocurrency, API, Cryptocurrency Address

[*]*Corresponding Author: njanpita@hotmail.com*

1. Introduction

A blockchain [1] technology was invented by Satoshi Nakamoto in October 2008 via paper titled “Bitcoin: A Peer-to-Peer Electronic Cash System. Blockchain is running based on a peer-to-peer network in which transactions are sent from one party to another without a centralized administration. This characteristic of blockchain can improve system availability because the system does not depend on any centralized server. Every full node that is online can serve as a server. The node can leave and come back to join anytime at will. The blockchain will record the hash of transactions as a reference so that it is not possible to edit any transaction and make the same hash to the original transaction. This characteristic of blockchain can ensure that the transactions on blockchain are immutable records.

Bitcoin is the first cryptocurrency that was implemented successfully using blockchain technology. By using blockchain technology, Bitcoin can be sent from one account to other accounts without mint or central authority. Satoshi has defined the transactions of Bitcoin as a chain of digital signatures. The hash of the previous transaction and the public key of the receiver is signed by the sender’s private key and attached to the end of the coin prior to the transfer of the coin. A blockchain node can verify the signatures to verify the chain of ownership.

Ethereum [2], Litecoin [3], Monero [4], and Dash [5] are other cryptocurrencies that are implemented based on blockchain. These cryptocurrencies were called altcoin because they are alternate to Bitcoin. Bitcoin and altcoin are increasingly popular because users have believed in their several advantages over fiat money such as low fees and irreversible transactions.

The next section will explain more details about blockchain node and verification processes.

2. Blockchain Node and Verification Processes

According to the definition of a node in [6], there are three main types of blockchain nodes as follows.

1. Full-node client: A full client, or “full node”, is a client that stores the entire history of Bitcoin transactions, manages users’ wallets, and can initiate transactions directly on the Bitcoin network. Full nodes act as a server in a decentralized network. They handle all aspects of the protocol and can independently validate the entire blockchain and any transaction. A full-node client consumes substantial computer resources (e.g., more than 125 GB of disk, 2 GB of RAM) but offers complete autonomy and independent transaction verification.

2. Lightweight client: A lightweight client, also known as a simple-payment-verification (SPV) client. These types of nodes communicate with the blockchain while relying on full nodes to provide them with the necessary information. As they don’t store a copy of the chain, they only query the current status for which block is last, and broadcast transactions for processing.

3. Third-party API client: A third-party API client is one that interacts with Bitcoin through a third-party system of application programming interfaces (APIs), rather than by connecting to the Bitcoin network directly. The wallet may be stored by the user or by third-party servers, but all transactions go through a third party.

There are four main verification processes (or consensus processes) before a single raw transaction can be added successfully in a blockchain network as follows.

1. Every blockchain validating node that receives a transaction (in this state, called “unconfirmed” transaction) will independently verify the transaction to ensure that only valid transactions are propagated across the network.

2. Blockchain mining node will aggregate transactions into a candidate block and then solve the Proof-of-Work solution in order to get a new mining block.

3. Every blockchain validating node that receives the new block will verify the new block to ensure that only valid blocks are propagated across the network.

4. The final step in blockchain’s decentralized consensus mechanism is the assembly of blocks into chains and the selection of the chain with the most Proof-of-Work.

From a long checklist of verification criteria such as:

- The transaction’s syntax and data structure must be correct.
- The transaction size in bytes is less than MAX_BLOCK_SIZE.
- For each input, the referenced output must exist and cannot already be spent.
- The block data structure is syntactically valid.
- The block header hash is less than the target (enforces the Proof-of-Work).
- The block timestamp is less than two hours in the future (allowing for time errors).
- The block size is within acceptable limits.
- Etc.

We found that none of the verification processes care about the validity of the receiver address. We then tested the transaction in the cryptocurrency wallet application and found that if the sender inputs the invalid receiver address (does not exist in the network), the cryptocurrency balance, known as

unspent transaction outputs or UTXO will be decreased from the sender address and increased in the invalid receiver address. However, no one can claim to be the owner of those balance because no one has the private key of that invalid address. Therefore, this situation is considered as the loss of cryptocurrency without recovery option.

To demonstrate our test result, we use the Ropsten [7] which is a test network for Ethereum developer to develop and to do testing. We use the Metamask [8] as Ethereum wallet to connect to the Ropsten test network and the main network.

Our scenario is that Alice bought a 1 ETH watch from Bob's watch shop. The invalid receiver address can happen in many cases as the following:

1. Bob informs Alice an invalid address.
2. Bob's wallet is connected to a different network, then the valid address of the different network is informed to Alice.
3. Alice inputs the wrong address by herself.
4. Alice creates a fake evidence to deceive Bob that she already sent him 1 ETH using the Ropsten network and then gone with the watch.

The demonstration of this scenario is shown and explained in Figure 1-7.

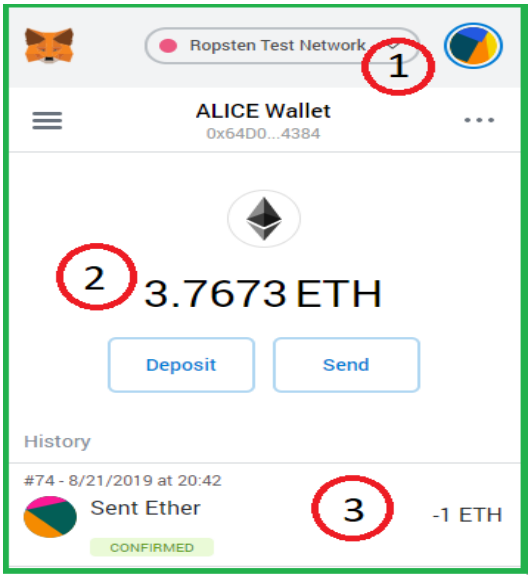

Figure 1. Alice's Ethereum Wallet on Ropsten Test Network using Metamask.

Figure 1 is Alice's Ethereum Wallet that has the following details:

1. Metamask wallet is connected to Ropsten test network.
2. Current balance or UTXO is 3.7673 ETH.
3. History of Alice transactions.

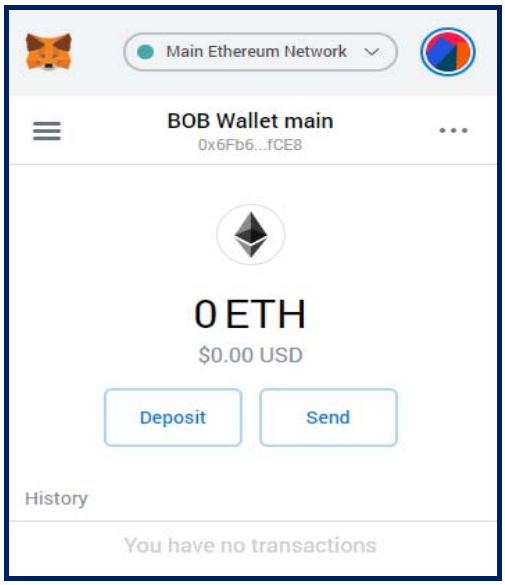

Figure 2. Bob's Ethereum Wallet on main network using Metamask.

Figure 2 shows Bob's Ethereum Wallet that has the following details:

1. Metamask wallet is connected to the main network.
2. Current balance or UTXO is 0 ETH.

Figure 3. Bob's Ethereum address.

Figure 3 shows Bob's address to which Alice must pay. It is "0x6Fb6a09A9273dDdA0393E694d24Da9B28981fCE8".

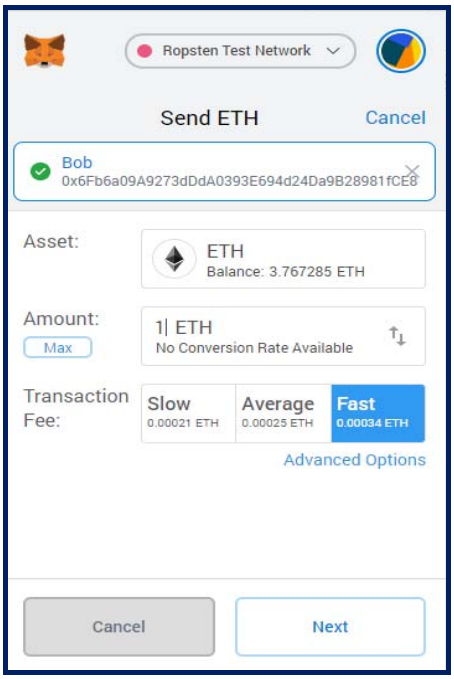

Figure 4. Alice is going to send 1 ETH to Bob's address.

Figure 4 shows the screen that Alice input 1 ETH and then she must push "Next" button for the next process.

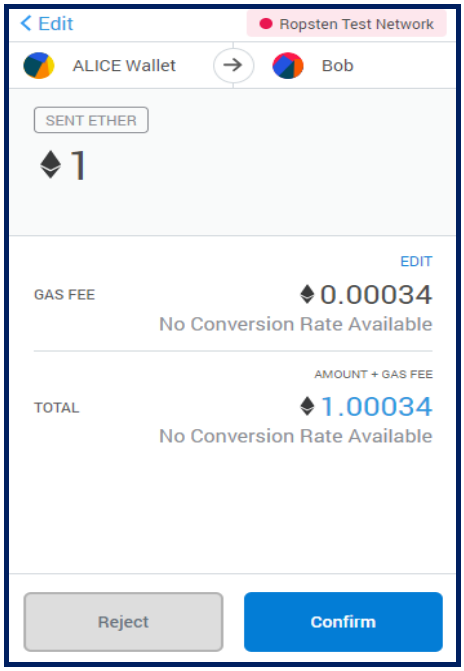

Figure 5. Alice is going to send 1 ETH to Bob's address.

Figure 5 is the screen that shows the transfer of information to Alice and waiting for her to push "Confirm" button to proceed to the next process.

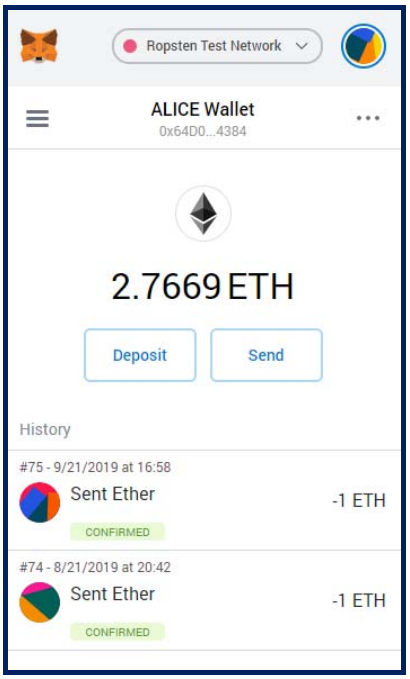

Figure 6. Alice has successfully sent 1 ETH to Bob's address.

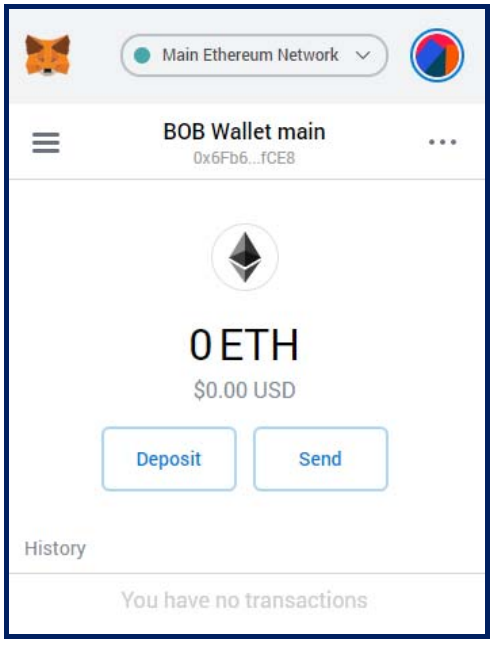

Figure 7. Bob's Ethereum Wallet after Alice has sent 1 ETH.

Figure 6 shows that the balance of Alice wallet has decreased from 3.7673 ETH to 2.7669 ETH while the balance of Bob in Figure 7 has not increased anyhow because the address is in a different network.

From this scenario, there is the cases that may happen,

1. Alice has attention to cheat, and she made the fake evidence shows to Bob that she

already paid. Bob may believe because he saw the figure 5. Alice got the watch for free. Bob loss his watch.

2. Alice has no attention to cheat, and she paid to Bob's valid address but difference network. If Bob waits for the transaction to be completed before giving away the watch, Alice must pay again. So, Alice loss her money.

To prevent a similar problem, we propose a process to verify the receiver address using the blockchain API before constructing and submitting the transaction across the network. If the receiver address does not exist in the same blockchain network, there is the process to notify the sender. This process can prevent the loss of cryptocurrency not only from invalid receiver addresses but also from different network addresses as well. This process needs to be developed in a wallet application.

3. Related Research

The anonymity is one core property of Bitcoin and other alternate coins. The purpose of anonymity is preserving the privacy of users by hiding user's identity from user's transactions. However, anonymity may affect the process of sending Bitcoin related to the target receivers addresses as mentioned before.

Many researchers have identified several problems associated with the privacy and anonymity of Bitcoin and then provide the solutions to address these issues. Niluka Amarasinghe, Xavier Boyen and Matthew McKague [9] conducted a survey of those solutions and concluded that many solutions do not provide an acceptable level of anonymity.

Mauro Conti et al. [10] conducted a survey on security and privacy issues of Bitcoin and one of the issues is anonymity. Even anonymity may affect some processes, but it is still required in terms of cryptocurrency properties. So, it can be said that anonymity needs to be improved but not removed.

QingChun ShenTu and JianPing Yu [11] also studied anonymization and de-anonymization technologies. They discussed some anonymization methods such as Analysis of Transaction Chain (ATC), coin-mixing and transaction remote release (TRR) that were used to cover the relationship between Bitcoin address and the user. Finally, they made some concluded that:

- ATC could not reach the practical stage. Many stolen Bitcoins failed to be identified its owner.
- The security and practicability of the decentralized coin-mixing protocols have not been estimated adequately. More de-anonymization research is expected to attack decentralized coin-mixing protocols.
- Group signature, group blind signature, privacy sharing, homomorphic encryption, lattice cryptography and other algorithms should be applied on Bitcoin system.

Liu et al. [12] discussed that the transactions can be attacked after submitting from wallet by interception, modification, and rebroadcast of a transaction into the Bitcoin network. They proposed the mechanism to recheck the balance of Bitcoin address after spending to confirm the completion of a transaction in case of transaction malleability. However, this mechanism is for mistake detection but not for mistake prevention.

Sai, Buckley, and Le Gear [13] discussed the privacy and security of cryptocurrency mobile applications by referring to the OWASP mobile top 10. They performed the test and investigation on wallet application vulnerabilities but none of the result discussed about validation of receiver address.

In the Bitcoin talk forum [14], there is the discussion header "If I send my Bitcoins to a wrong address, what happens?". Until today, the most discussion still saying that the Bitcoin will be lost forever if it has been sent to the wrong address.

Coinbase Q&A [15] also mentioned that "Due to the irreversible nature of digital currency protocols, transactions can neither be canceled nor reversed once sent. In this scenario, it would be necessary to contact the receiving party and seek their cooperation in returning the funds. If you do not know the owner of the address, there are no possible actions you can to take to retrieve the funds."

In Quora's discussion [16] header "What happens if I send my Bitcoin to an incorrect address?" also mention that the cross-chain deposits are rare but possible; many people have lost their money by sending Bitcoin to a Bitcoin cash or sending NEO to Bitcoin address for example.

4. Proposed Receiver Address Verification Process

Since most of the blockchain verification processes will take time around 10 minutes, so it does not make sense to reject the transaction in the later steps if invalid receiver address is found. It will incur more workload to the blockchain network. The invalid receiver address should be tracked at the earliest stage possible. From this point, we proposed to verify the receiver address before constructing and submitting the transaction to the network. This can be better done by the wallet application. However, most of the wallet applications especially mobile wallets are implemented based on lite nodes that do not have the full data of blockchain.

So, we would propose to include a verification script into the wallet application by using the blockchain API from the third party to verify the receiver address. If the receiver address is not found in the

block explorer, the wallet application should notify user to re-input receiver address before proceeding to the next process. User can decide to continue or re-input receiver address at will (just in case the wallet application is in the test environment).

Since we cannot modify the worldwide wallet application such as Metamask, so we use our own developed wallet application developed by the Geth [17] command and Web3.js [18] interface which are used by most Ethereum wallet applications in the real market. The process demonstration can be explained by Figure 8-11.

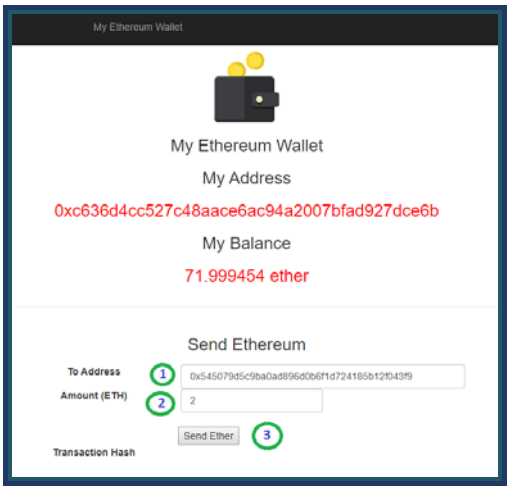

Figure 8. Simple Ethereum Wallet developed by Web3.

```
// Send ether
function sendEther(){
    var to = document.getElementById('send_to').value;
    var amount = document.getElementById('send_amount').value;

    var params = {
        from: web3.eth.coinbase,
        to: to,
        value: web3.toWei(amount, "ether"),
    };

    web3.eth.sendTransaction(params , function (error, result) {
        if(!error){
            document.getElementById('send_output').innerHTML =  result;
        }else{
            document.getElementById('send_output').innerHTML =  error;
        }
    });
}
```

Figure 9. JavaScript with Geth command to send Ether.

Figure 8 shows a simple Ethereum Wallet application. Once the user inputs the receiver address (1) and the amount (2) and clicks button “Send Ether” (3), the script is called to send Ether using the Geth command “web3.eth.sendTransaction()” as shown in Figure 9.

Command sendTransaction() required three important parameters which are: sender address (Figure 8: My Address), receiver address (Figure 8: To Address), and amount (Figure 8: Amount (ETH)). Application will retrieve sender address from coin base which is a primary wallet address of user who logged in. The receiver address 0x545079d5c9ba0ad896d0b6f1d724185b12f043f9 from user input is a valid address and is in the same network with the sender address (0xc636d4cc527c48aace6ac94a2 007bfad927dce6b), so the command sendTransaction() can be executed correctly. Once the last digit of the receiver address was removed giving “0x545079d5c9ba0ad896d0b6f1d724185b12f 043f” and “Send Ether” was clicked, there was an error in Web3.js console as shown in Figure 10.

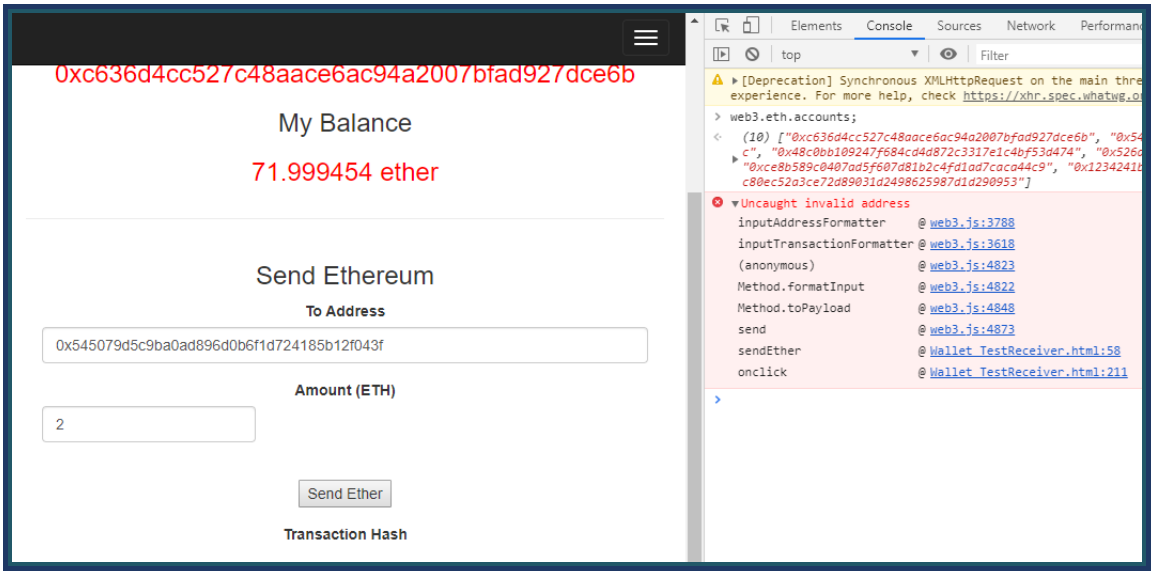

Figure 10. Web3.js error when inputting the wrong format of the receiver address.

We then tried again with the same address, but the last digit was changed from 9 to 8. As shown in Figure 11, we found that the command sendTransaction() can be executed normally. The transaction was confirmed, and the balance of "My Address: 0xc636d4cc527c48aace6ac94a2007bfad927dce6b" was decreased by the amount that has been sent. This can be concluded that the command sendTransaction() can detect only the invalid address from the wrong format but cannot detect the invalid address in the right format. This problem also happens in Metamask which is the worldwide Ethereum wallet as we have already explained in section 2.

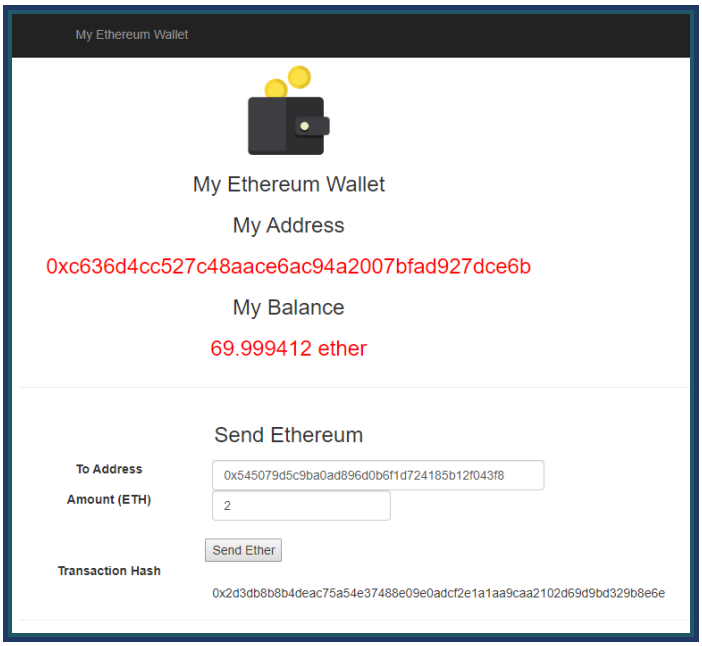

Figure 11. Command sendTransaction() can be executed normally on the invalid address.

To prevent the problem as shown in Figure 11, a script was developed to verify the receiver address before sending it to command sendTransaction() as shown in Figure 12.

Figure 12. JavaScript to verify receiver address using blockchain API.

The verification script shown in Figure 12 uses the third-party API (1) to check and notify the user if the receiver address has never received any amount (2) so far in the same network. As an example, the result of using the direct API link shown in Figure 13 indicates that the total_received column of address "0x64D0c83f7852A741381F16088b469135c6154384 " is 0 which means that this address has never received any amount in the past. So, there will be the notification to the user, as shown in Figure 14. The user can continue to send to this address if he is sure that this address is newly created in the same network.

Figure 13. API checking result of address "0x64D0c83f7852A741381F16088b469135c6154384".

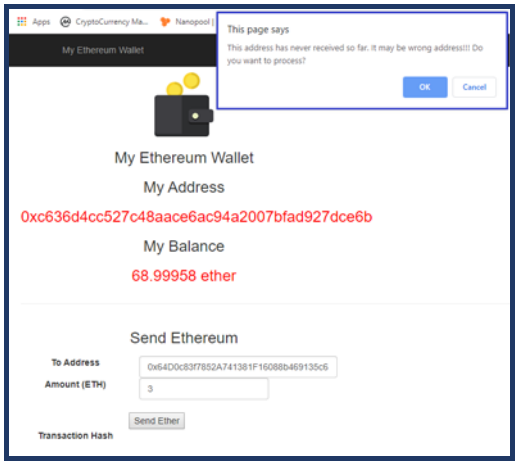

Figure 14. User notification if the receiver address has never been received so far.

This test result shows that the user's mistakes can be found and reduced before the construction and submission of the transaction to the blockchain network.

5. Conclusions and Discussions

Lack of KYC (Know Your Customer) is the main weaknesses of blockchain technology. There is no information helping users to verify whether the address is really belonging to the target receiver as the traditional centralized banking system. With the increasing use of cryptocurrency, new user has obtained new wallet every day without awareness of security. Some user never aware of any mistake that may happen until it happened. Whenever the mistaken input committed, user cannot claim to any centralized authority even the web exchange. They also cannot find out who is the owner of mistake address because there is no information like that in blockchain network. This mistake causes the cryptocurrency loss without recovery forever. To reduce the mistake transactions committed by users and reduce the forever loss of cryptocurrency in the market, we recommend adding our proposed verification process into the existing wallet applications.

References

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008. [Online]. Available: https://Bitcoin.org/ Bitcoin.pdf.
[2] Ethereum. [Online]. Available: https://www.ethereum.org.
[3] Litecoin - Open source P2P digital currency. [Online]. Available: https://litecoin.org/.
[4] Monero - secure, private, untraceable. [Online]. Available: https://www.getmonero.org/
[5] Dash - Dash is Digital Cash You Can Spend Anywhere. [Online]. Available: https://www.dash.org/.
[6] A. M. Antonopoulos, "Mastering Bitcoin: Unlocking Digital Cryptocurrencies," 1st ed. Sebastopol, CA, USA: O'Reilly Media, Inc., 2014.
[7] Testnet Ropsten (ETH) Blockchain Explorer. [Online]. Available: https://ropsten.etherscan.io/.
[8] Metamask, Brings Ethereum to your browser [Online]. Available: https://metamask.io.
[9] N. Amarasinghe, X. Boyen, and M. McKague. "A Survey of Anonymity of Cryptocurrencies,". In Proceedings of the Australasian Computer Science Week Multiconference (ACSW 2019). ACM, New York, NY, USA.
[10] M. Conti, E. Sandeep Kumar, C. Lal and S. Ruj, "A Survey on Security and Privacy Issues of Bitcoin," in IEEE Communications Surveys & Tutorials, vol. 20, no. 4, pp. 3416-3452, Fourth Quarter 2018.
[11] Q. ShenTu and J. Yu, "Research on anonymization and deanonymization in the Bitcoin system," arXiv preprint arXiv:1510.07782, Oct. 2015. [Online]. Available: https://arxiv.org/abs/1510.07782.
[12] Y. Liu, X. Liu, L. Zhang, C. Tang, and H. Kang, "An efficient strategy to eliminate malleability of Bitcoin transaction," 2017 4th International Conference on Systems and Informatics (ICSAI), Hangzhou, 2017, pp. 960-964.
[13] A. R. Sai, J. Buckley and A. Le Gear, "Privacy and Security Analysis of Cryptocurrency Mobile Applications," 2019 Fifth Conference on Mobile and Secure Services (MobiSecServ), Miami Beach, FL, USA, 2019, pp. 1-6.
[14] Bitcoin talk. If i send my Bitcoins to a wrong address, what happens? [Online]. Available: https://Bitcointalk.org/index.php?topic=21127.0.
[15] Coinbase Q&A. "I sent funds to the wrong address. How do I get them back?" [Online]. Available: https://support.coinbase.com/customer/en/portal/articles/1404411-i-sent-funds-to-the-wrong-address-how-do-i-get-them-back-.
[16] Quora. "What happens if I send my Bitcoin to an incorrect address?" [Online]. Available: https://www.quora.com/What-happens-if-I-send-my-Bitcoin-to-an-incorrect-address.
[17] Official Go implementation of the Ethereum protocol. [Online]. Available: https://geth.ethereum.org/.
[18] Ethereum JavaScript API. [Online]. Available: https://github.com/ethereum/web3.js/.