Securing the Private Key in Your Blockchain Wallet: A Continuous Authentication Approach Based on Behavioral Biometric

Teng Hu\textsuperscript{1,2}, Xiaolei Liu\textsuperscript{2}, Weina Niu\textsuperscript{1}, Kangyi Ding\textsuperscript{1}, Yanping Wang\textsuperscript{1} and Xiaosong Zhang\textsuperscript{1,3*}

\textsuperscript{1} Institute for Cyber Security, School of Computer Science and Engineering, University of Electronic Science and Technology of China, China
\textsuperscript{2} Institute of Computer Application, China Academy of Engineering Physics, China
\textsuperscript{3} Peng Cheng Laboratory, Cyberspace Security Research Center, China

Email: johnsonzxs@uestc.edu.cn

Abstract. As blockchain is increasingly valued, the application of blockchain and its security has become a research hotspot. Not only that, more and more ordinary users are paying attention to the blockchain and beginning to trade commodities in cryptocurrencies. The blockchain wallet is a program for users to manage and stores their private keys. Furthermore, the private key is the user’s unique credential to the cryptocurrency in the blockchain. But for now, blockchain wallets are only using passwords to authenticate users, which poses a potential risk to private key security. To mitigate this risk, this paper proposes a continuous authentication approach based on user mouse behavioral biometrics. The approach was experimentally shown to be able to authenticate once per second with an average FAR and FRR of 6.92\% and 6.18\%, respectively.

1. Introduction
The development of blockchain technology makes it possible to transfer value between anonymous participants without relying on an authoritative third party [1]. Nowadays, blockchain has become a research hotspot in academia and industry [2, 3], blockchain technology is also often referred to as the next generation internet [4], and there have been many successful applications based on it [5]. For example, Bitcoin [6], it is the first successful application of blockchain technology. It was named the best performing currency in 2015 [7] and the best performing commodity in 2016 [5] in May 2017. Another example is Ethereum [8], which has a market cap of around $25 billion (as of June 2019) and is the second-largest blockchain platform after Bitcoin. At the same time, it is the first blockchain platform to support smart contracts and the largest platform [9].

The main activities that ordinary users will use cryptocurrencies for include exchanging cryptocurrencies, saving their bitcoins and ether, or using them for commodity trading [10]. To support these, we needed a blockchain wallet, a program is used to control access to the blockchain main chain, manage private keys and addresses, track user account balances, create and sign transactions. However, there is no specific "money" stored in the wallet, but the address of the cryptocurrency (similar to the bank card account) and the private key (similar to the bank card password) are stored. The core function of the wallet is to protect the security of the private key [11]. That is because in the world of blockchains, whoever holds the private key is the real owner of the
A digital asset, so once the private key is leaked, it means that the ownership of the cryptocurrency is lost. Therefore, the security of private keys is important for users and the wallet is actually a management program for managing private keys.

At present, there are two mainstream wallet forms. One is the full-node wallet [12], which represents the Bitcoin-Core wallet, Geth, Parity, etc. This kind of wallet does not directly provide an interface to the outside. It is theoretically the most secure, but the shortcomings are also obvious. To maintain huge blockchain data and the data must be synchronized to the latest. Otherwise, it will lead to the inability to query the transaction status, or even unable to trade.

The other is a light wallet, such as MetaMask [13], which is convenient for ordinary users. Most mobile wallets and web wallets are light node wallets. Light wallets do not need to maintain huge block data. The light wallet only needs to keep the private key, sign the transaction, and then send it to the trusted complete node. Of course, there needs to be an honest and full node open to this function and willing to provide these services. When using wallets, we should beware of the leakage of our private keys.

At present, most wallet software provided by exchanges still uses passwords to verify user identities, and also uses user passwords to encrypt private keys stored in wallets. The user logs in to the wallet through password verification, and the user can perform operations such as transfer after login. In this way, there is a risk of private key leakage. Once the user password is leaked, it would lead the private key leakage, which means that the user has lost the ownership of the digital assets in the blockchain. Unlike other management systems, there is no way to take ownership back of the private keys through other channels. What is worse, to use the wallet conveniently, many users are habitually set to automatic password remembering, which gives attackers many opportunities to obtain private keys.

This paper proposes an approach to continuously verify the user’s identity so that to address this issue. Based on this approach, the user’s private key can be protected in real-time. Once the wallet detects that the user’s identity does not match, it can immediately exit or prevent the transaction from proceeding, thereby effectively preventing the leakage of the user’s private key. This approach is based on the user’s mouse behavior biometrics and the CNN network, as shown in figure 1. When a person registers in the wallet as a user, the Wallet provider will first collect the user’s mouse behavior biometrics, then use a CNN network to train the user’s biometric model. The model training process can be on the user’s local machine or on the server to increase the training speed. After that, the user’s biometric model and the private key are paired and stored in the wallet. When the user logging or using the wallet, the mouse biometrics are also generated. The wallet continuously verifies the user’s identity through the stored user model, which can effectively prevent the leakage of the private key. The experiment shows that our approach can complete the identification of the user’s identity in every second, and the FAR and FRR of 6.917% and 6.175%, respectively. The main contribution of this article is to propose an approach for the blockchain wallet to verify the user’s identity in real-time through the user’s biometrics, which can effectively prevent the leakage of the user’s private key.

![Figure 1. The Framework of the Continuous Authentication for Blockchain Wallet.](image-url)
2. Data, Coordinate System, and CNN Model

2.1. Data Collection and Recording
We monitor mouse actions by hooking mouse messages of windows. In this way, we collected a total of five users’ mouse behavior biometric information, and each user has about 500,000 actions. The data stored in chronological order and in the following format: [timestamp, x, y, action], where timestamp represents the time when recording the mouse action, action represents the specific action of the mouse, x and y respectively indicate the coordinate values where the mouse is in action. Figure 2 shows a schematic of the data storage format.

![Figure 2. The schematic figure of the data storage format.](image)

2.2. Coordinate System
The biometric behaviors we collected include these basic actions of the mouse: move, left-press, left-release, right-press, right-release, drag, down-scroll, and up-scroll. We represent these actions with fixed symbols on the coordinate system. The corresponding x and y values at the time of each action record correspond to the position on the coordinate system. As shown in figure 3, the blue line indicates move, the yellow line indicates drag, the red dot indicates the left mouse button left-pressed, the red circle indicates the left mouse button left-released, the green dot and circle respectively indicate the right mouse button right-pressed and right-released, and the upper and lower black triangles indicate the up-scroll and down-scroll.

According to the mouse biometrics we collected, the average number of mouse actions generated by the user per second is shown in table 1. Draw a coordinate system figure every second, including all mouse actions generated by the user within this second, until all mouse actions of a user are completely drawn. Finally, as shown in table 2, five data sets are obtained.

To make the classification of CNN more accurate, we adopted the method of incremental data set. We performed random horizontal flip, vertical flip, 90-degree rotation, 180-degree rotation, and 270-degree rotation operations on all coordinate system figures. And randomly selected 20,000 coordinate system figures as our experimental data set. In other words, for each user, we have a data set containing 20,000 figures.

![Figure 3. Schematic figure of the user’s mouse biometrics represented on the coordinate system.](image)
Table 1. The average number of mouse actions generated by the user per second.

| User | Count (per second) |
|------|-------------------|
| A    | 54.25             |
| B    | 60.87             |
| C    | 74.26             |
| D    | 63.31             |
| E    | 45.96             |

Table 2. Data sets of user mouse biometric.

| User | Count (in total) |
|------|------------------|
| A    | 9217             |
| B    | 8215             |
| C    | 6733             |
| D    | 7898             |
| E    | 10880            |

3. CNN Model

We have constructed a 12-layer CNN network based on Keras, as shown in figure 4. There is a total of four convolutional layers, two maximum pooling layers, three dropout layers to reduce overfitting, one Flatten layer to make two-dimensional input into one-dimensional, and two fully connected layers to implement the classifier function. The output of the last layer is sent to Softmax function and obtained the probability distribution of the classified labels.
4. Experiments

This section introduces the experimental configuration, the evaluation metrics, and the results. We completed the entire model training and experiments in the following experimental environments: Python, Keras, TensorFlow, CUDA and cudnn, and NVIDIA GTX 1060 6GB GPU. We use false acceptance rate (FAR) and false rejection rate (FRR) to evaluate the effectiveness of our approach.

The purpose of our experiment is to test whether our approach can identify the user in a very short time (in one second). The blockchain wallet needs to determine whether the person currently logged in or using the wallet is consistent with the model of the legitimate user stored in the wallet. When a user registers a wallet, it will collect his mouse biometric as model training data. After the model is trained, it will be bound to the user’s private key and stored in the wallet. When the user uses the wallet, the user’s mouse biometric are continuously collected, and the user’s identity is authenticated by the model every second. Therefore, we design the experiment as follows. First, we designate one user as a legal user, such as User A, and four other users as illegal users. Then, using to the datasets generated by section 2.2, divide the dataset into a training dataset (represented by Tr0) and a test dataset (represented by Te0) according to the ratio of 8:2, and set aside 20% from the training set as a validation dataset for adjustment and optimization model parameters. This user’s training dataset is used as the positive data set for model training. Next, randomly select the training dataset (represented by Tr1) and test dataset (represented by Te1) from the remaining four user data sets according to the above proportions. And ensure that there is no intersection between Tr1 and Te1. The Tr1 dataset is used as the negative data set for model training. After that, take the training set Tr0 + Tr1 (total 32,000) as input and use the CNN network constructed in Section 2.3 to train the model. Test Te0 + Te1 (total 8,000) with the generated model and calculate FAR and FRR. Finally, take the above experiments for five users, respectively, and calculate the average FAR and FRR. The experimental results are shown in table 3.

| User | FAR (%) | FRR (%) |
|------|---------|---------|
| A    | 6.59    | 4.51    |
| B    | 6.78    | 5.56    |
| C    | 11.59   | 7.62    |
| D    | 3.67    | 5.56    |
| E    | 5.95    | 7.63    |
| **Average** | **6.916** | **6.176** |

As can be seen from the result of table 3, our approach can complete the task of identity authentication. Because we generate a coordinate system figure every second, this means that each figure contains the mouse behavioral biometrics generate in only one second. Then, we could use the trained model for classification. This shows that our approach can complete identity authentication once every second, meeting the real-time requirements. In addition to user C, the FAR results of the other users are good enough. We think this was caused by the interference of user C’s data during data collection. However, this does not detract from the experimental results indicating that our proposed approach has important implications for the security of private keys in protected blockchain wallets.

5. Related Work

Due to the characteristics of the blockchain technology, the user’s private key is the sole credential for the user’s account use rights and ownership of digital assets within the account [14]. Furthermore, incidents of bitcoin wallets being hacked resulting in the theft of bitcoins occur from time to time [15]. Therefore, many researchers have conducted research on blockchain wallet technology around the management and protection of user private keys. The first blockchain wallet was a full-node wallet, developed by Satoshi Nakamoto when he developed the Bitcoin protocol in 2009, called Bitcoin-Qt
[6]. Bitcoin-Qt’s private key is not password protected and there is no way to import and export the private key. Later, Bitcoin-Qt was renamed to Bitcoin core and added the encrypted function, which can encrypt the file where the private key is stored and import and export the private key. However, the Bitcoin core has obvious defects, its data volume is too large, and private key management is inconvenient [16]. To solve these issues, there came into being the SPV (short for Simplified Payment Verification) wallet and HD (short for Hierarchical deterministic) wallet. The SPV wallet only retains a part of the blockchain, so that the data volume is small and the transaction speed is fast. The HD wallet can effectively manage multiple private keys. However, both wallet technologies sacrifice security. For SPV wallets, an attacker can attack the block header so that it cannot ensure the accuracy of the verification transaction result [17]; or steal the private key when the user uses the private key. Dai et al. [11] proposed a blockchain-based lightweight wallet based on Trustzone, which protects private keys and block headers, so that solves the problem of low SPV security. But it also has defects, that is, many devices do not support Trustzone technology. For the HD wallet, there is a fixed relationship between the private keys. So, the attacker can recover the master private key by adding any one of the child private keys and the master public key. The first wallet with HD technology is Electrum [18], but Electrum has low security. As in Ref. [19], the Electrum wallet was offline violence cracked. Ref. [20] proposed a new HD wallet that uses trapdoor hash functions to send signatures to prevent association attacks.

6. Conclusion
To mitigate the potential risk to private key leakage, we propose a continuous authentication approach based on user mouse behavioral biometrics. This approach uses the principle that users use the mouse to generate biometric, combined with the CNN network training classification model, as a means of continuous authentication. The approach was experimentally shown to be able to authenticate once per second with a good performance. We believe that this could solve the problem of secure storage of private keys in blockchain wallets to a certain extent. Future work directions include improving the accuracy of the approach and another way to address the authentication issue in the blockchain.

Acknowledgment
This work was supported by the Institute of Computer Application, China Academy of Engineering Physics (SJ2019A05) and the National Natural Science Foundation of China (U19A2066, 61902262).

References
[1] Chen W, Zheng Z, Cui J, Ngai E, Zheng P and Zhou Y 2018 April Detecting Ponzi schemes on Ethereum: Towards healthier blockchain technology Proceedings of the 2018 World Wide Web Conference pp 1409-1418.
[2] Swan M 2015 Blockchain: Blueprint for a New Economy (O’Reilly Media, Inc.).
[3] Zheng Z, Xie S, Dai H N, Chen X and Wang H 2018 Blockchain challenges and opportunities: A survey International Journal of Web and Grid Services 14 (4) 352-375.
[4] Fernández-Caramés T M and Fraga-Lamas P 2019 A review on the application of blockchain to the next generation of cybersecure industry 40 smart factories IEEE Access 7 45201-45218.
[5] Li X, Jiang P, Chen T, Luo X and Wen Q 2020 A survey on the security of blockchain systems Future Generation Computer Systems 107 841-853.
[6] Nakamoto S 2008 Bitcoin: A Peer-To-Peer Electronic Cash System.
[7] DeVries P D 2016 An analysis of cryptocurrency, bitcoin, and the future International Journal of Business Management and Commerce 1 (2) 1-9.
[8] Wood G 2014 Ethereum: A secure decentralised generalised transaction ledger Ethereum Project Yellow Paper 151 (2014) 1-32.
[9] Chen T, Zhu Y, Li Z, Chen J, Li X, Luo X, Lin X and Zhang X 2018 Understanding ethereum via graph analysis IEEE INFOCOM 2018-IEEE Conference on Computer Communications pp 1484-1492.
[10] White L H 2015 The market for cryptocurrencies Cato J. 35 383.
[11] Dai W, Deng J, Wang Q, Cui C, Zou D and Jin H 2018 SBLWT: A secure blockchain lightweight wallet based on trustzone IEEE Access 6 40638-40648.
[12] Elrom E 2019 EOS IO Wallets and Smart Contracts in the Blockchain Developer (Berkeley, CA: Apress) pp 213-256.
[13] Lee W M 2019 Using the metamask chrome extension Beginning Ethereum Smart Contracts Programming (Berkeley, CA: Apress) pp 93-126.
[14] Zheng Z, Xie S, Dai H, Chen X and Wang H 2017 An overview of blockchain technology: Architecture, consensus, and future trends 2017 IEEE International Congress on Big Data (BigData Congress) pp 557-564.
[15] Peterson B 2017 Thieves stole potentially millions of dollars in bitcoin in a hacking attack on a cryptocurrency company Business Insider 6 12.
[16] Van Der Horst L, Choo K K R and Le-Khac N A 2017 Process memory investigation of the bitcoin clients electrum and bitcoin core IEEE Access 5 22385-22398.
[17] Kogias E K, Jovanovic P, Gailly N, Khoffi I, Gasser L and Ford B 2016 Enhancing bitcoin security and performance with strong consistency via collective signing 25th {Usenix} Security Symposium ({Usenix} Security 16) pp 279-296.
[18] Gutoski G and Stebila D 2015 January Hierarchical deterministic bitcoin wallets that tolerate key leakage International Conference on Financial Cryptography and Data Security (Berlin, Heidelberg: Springer) pp 497-504.
[19] Volety T, Saini S, McGhin T, Liu C Z and Choo K K R 2019 Cracking Bitcoin wallets: I want what you have in the wallets Future Generation Computer Systems 91 136-143.
[20] Fan C I, Tseng Y F, Su H P, Hsu R H and Kikuchi H 2019 Secure hierarchical Bitcoin wallet scheme against privilege escalation attacks International Journal of Information Security 1-11.