On Detecting Cold Storage Transactions on Bitcoin's Blockchain

Ivan da Silva Sendin[1]

[1]Faculdade De Computação - FACOM – Universidade Federal do Uberlândia (UFU)
Uberlândia – MG – Brazil

sendin@ufu.br

Abstract. *There is a disparity between Bitcoin addresses and real-world entities: the same entity can have many addresses. In Blockchain's analysis, a common technique used for clustering addresses is to view addresses present at the input of the same transaction as a single entity. A common practice to make Bitcoin safer is the use of cold wallets. The use of cold wallets by exchanges - that control the wallets of many users - may disrupt Blockchain's current methods of analysis. In this work we define these scenarios and introduce an heuristic and an algorithm to detect these occurrences on Blockchain. We show that the data obtained using the proposed heuristic are consistent with what was expected.*

1. Introduction

Bitcoin is the oldest and most widespread cryptocurrency. In Bitcoin system, users have addresses - abstractions for public keys - where their Bitcoins are stored, like *wallets* in real world. Bitcoins are *sent* to other addresses through signed transactions stored in a public ledger called Blockchain. Transactions are public and can be accessed by any individual through the Bitcoin P2P network. In a simplified manner, a transaction moves the unspent values from previous transactions to set of addresses. In Figure 1 we show a Bitcoin transaction, each entry in **Input** column ($TxHash_i, pos_i$) refers a specific entry in **Output** column of some previous transaction, identified by $TxHash_i$. Each entry must be signed by the corresponding private key. Details on how Bitcoin works can be found in its original paper [Nakamoto 2008] and in the following references: [Antonopoulos 2014, Narayanan et al. 2016].

The expansion of the adoption of Bitcoin led to the interest in the analysis of the transactions stored in the Blockchain. These analyzes are used to characterize the behavior and profile of system users[Ron and Shamir 2013b] or even detect illegal activities [Ranshous et al. 2017, Liao et al. 2016, Ron and Shamir 2013a]. The Blockchain analysis can be done using only the Blockchain data [Filtz et al. 2017, Androulaki et al. 2013, Ober et al. 2013, Koshy et al. 2014, Spagnuolo et al. 2014, Maesa et al. 2016, Tschorsch and Scheuermann 2016, McGinn et al. 2016] or also using external information (e.g. IP address) to Blockchain [Fleder et al. 2015, Meiklejohn et al. 2016].

One difficulty that arises when analyzing Blockchain is that each real-world entity can have a large number of addresses and it is necessary to determine the pool of addresses that represent each entity, this process is called **address clustering**.

A widely adopted heuristic [Androulaki et al. 2013, Ron and Shamir 2013b, Ober et al. 2013, Ortega 2013, Meiklejohn et al. 2016, Zhao 2014, Koshy et al. 2014,

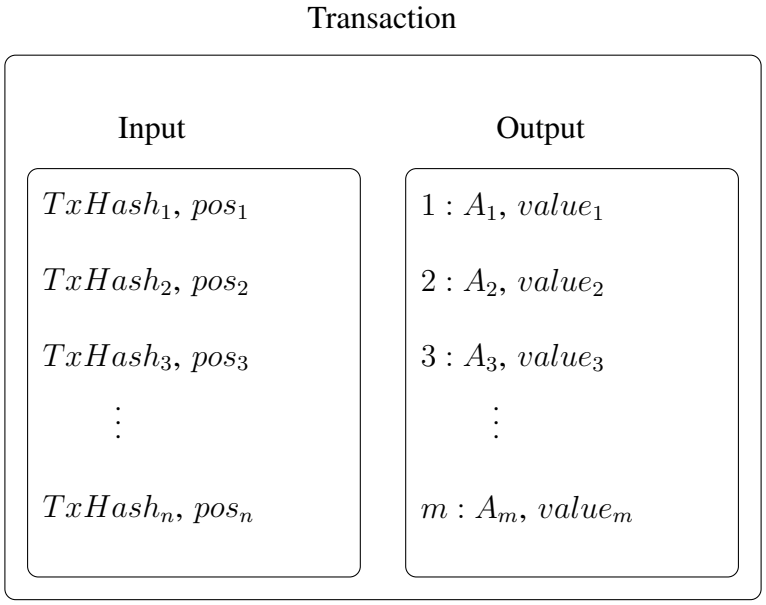

Figure 1. A Bitcoin transaction.

Spagnuolo et al. 2014, Fleder et al. 2015, Monaco 2015, Filtz et al. 2017, Tschorsch and Scheuermann 2016, McGinn et al. 2016, Akcora et al. 2017, Maesa et al. 2016, Harrigan and Fretter 2016] is to cluster all addresses present in the input of one transaction as being the same entity: as the transaction must be signed with the private key, the one who built the transaction knows the private key of each address, therefore must represent the same entity. Once established that a set of addresses is associated with the same entity, all transactions sharing at least one address must belong to the same entity. For example, transactions inputs containing $[A_1, A_2]$,$[A_2, A_3]$ and $[A_3, A_4]$ produce a cluster composed of addresses $[A_1, A_2, A_3, A_4]$.

Although correct in most cases, we will show in this work that this heuristic can lead to false positives, producing spurious and meaningless clusters: the exchange can control the Bitcoins through the knowledge of the private key but do not actually has their ownership.

1.1. Bitcoin for the masses

The canonical way of using Bitcoin is to download the reference implementation and, after that, download the Blockchain and manage your addresses on your personal computer. With the popularity of Bitcoin, people uninitiated in the world of cryptography may own cryptocoins by simple communicating with an exchange server using a web interface or smartphone application. In this operation model, the user uses a login and a password to manage their coins, all issues about encryption are hidden from users [Chuen 2015, Chapter 28].

The exchanges works similarly to traditional banks, and one of their tasks is to protect their user's coins from hacker attacks. A common procedure to increase security is called **Cold Storage** or **Cold Wallets**[1]: exchanges keeps only a fraction of Bitcoins

[1]See `https://bitcoin.org/en/secure-your-wallet`.

on line; most of their assets are stored on off line addresses. One way to implement a Cold Wallet is to periodically move the deposits made on client's addresses into an off line address, keeping a zero balance on client's addresses. If the exchange suffer a hacker attack most of coins held by the exchange is protected.

In a recent study[Hileman and Rauchs 2017] it was found that 92% of the exchanges use cold storage and that on average 87% of the funds are protected with this method. Despite the widespread adoption, the use of cold storage does not follow a standard protocol. Thus, issues such as frequency limits or values used to trigger a cold storage transaction are defined by each exchange.

This storage procedure produces false positives in the current adopted process of address clustering: exchanges often produce transactions with a few wallets in the Input, so the clients of an exchange are considered as a unique entity by the clustering process.

In this paper we present and explore the hypothesis that cold storage transactions can be detected using a simple and easily implemented heuristic. This heuristic selects transactions according to the following criteria:

Minimum input size It is expected that the exchange waits for a minimum number of customer addresses with balance before making the storage in a cold wallet;

Small output The output list of the transaction should be small, since cold wallets have a higher operating costs;

Inputs with recent deposits Since cold storage operations must be frequent, addresses in input of one transaction must have recent deposits.

2. Methods

Initially, we verified the presence of Cold Storage transactions according to the heuristic defined in Section 1, analyzing the occurrences of this type of transactions during the years 2014 to 2017, this period corresponds from the first block of Blockchain to the block 50195.

This information was obtained from Blockchain using a Python script using the *Bitcoin Blockchain Parser* [2] and *BitcoinLib*[3] libraries. In the selection of Cold Storage transactions, we considered transactions with 20, 40 and 80 different addresses in the Input with respective deposits funding occurring approximately in the 12 hours prior to the transaction and up to two addresses at the Output. The results are presented in Figure 2, where we show the evolution of the frequency and volume - in Satoshis[4] - of those transactions, is possible to observe a steady increase on this type of transaction, which may be a reflection of the adoption of exchanges by Bitcoin users.

Detailed analyzes have been made over the year 2017 - block 446033 to block 501950 - we believe that the growth of Bitcoin's adoption and the variation of its value could make it impossible to analyze over a large period. For this type of analysis we also use information from the website `blockchain.info` through its JSON API. The results are presented in Table 1.

Using the proposed heuristic we made the clustering of the addresses considering:

[2]https://github.com/alecalve/python-bitcoin-blockchain-parser/

[3]https://github.com/petertodd/python-bitcoinlib

[4]One Satoshi corresponds to 10^{-8} Bitcoins.

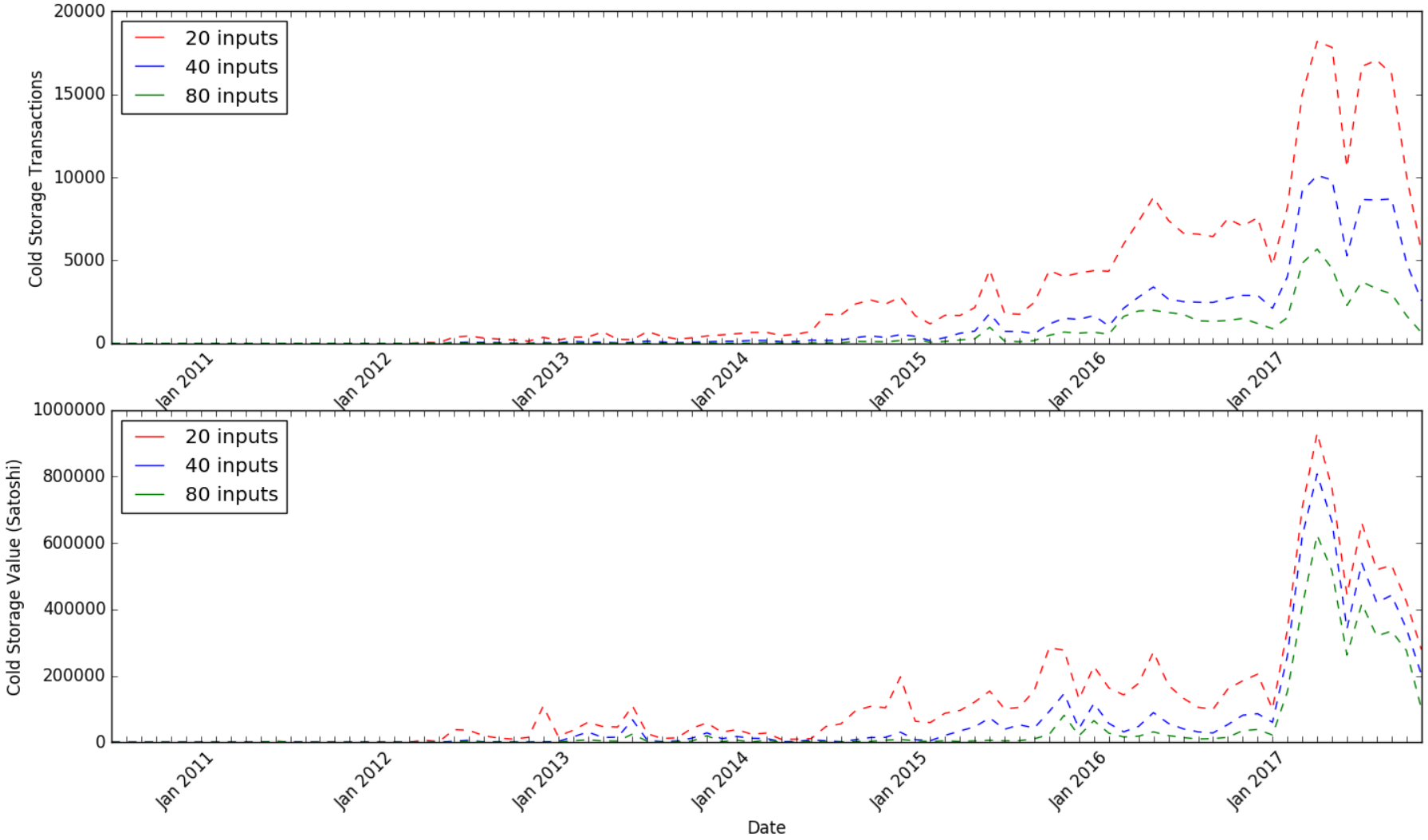

Figure 2. Cold Storage Transactions Evolution.

Evolution of the Cold Storage transactions detected in the Blockchain. The transactions with at least 20, 40 and 80 inputs with respective deposits funding occurring approximately in the 12 hours prior to the transaction.

Table 1. Summary of Cold Storage Transaction and Addresses detected for the year 2017.

Cold Storage Transaction	296,536
Cold Wallets Inputs	5,926,267
Cold Wallets Outputs	192,984

- Transactions sharing same addresses in Input;
- Transactions sharing same addresses in Output.

In Figure 3 we show a histogram with the result of clustering. We can observe that there are few clusters formed by a large number of transactions, this fact should reflect the small number of existing exchanges. In Figure 4, the activity days[5] of each cluster are shown, it is possible to observe that only some clusters have more than 200 days of activity per year. The large number of singleton transactions (not shown in the figures) or small clustering can be attributed to both the failure in the parameters used in heuristic (size of the transactions, period of observation of deposits) and other regular transactions in the Blockchain falsely selected.

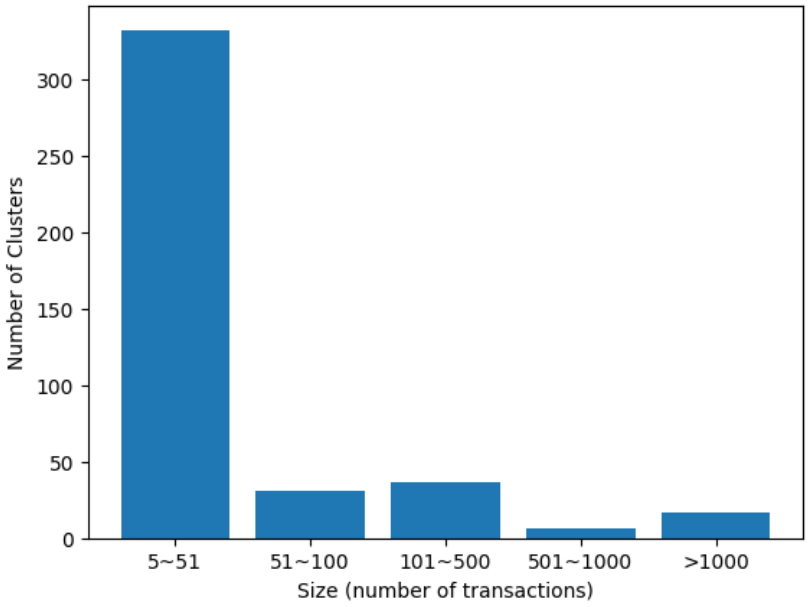

Figure 3. Histogram of clusters by cluster size.

3. Results

Once Cold Storage transactions have been selected and clustered, we can perform analysis to verify the efficiency of the proposed heuristics. Due to limitations of time, computational power and network band the following data sets were used in the analysis:

ColdClients From the 5 million addresses detected as exchange clients, 4250 addresses were sampled;

[5]Number of days with at least one transaction executed by a wallet in the cluster.

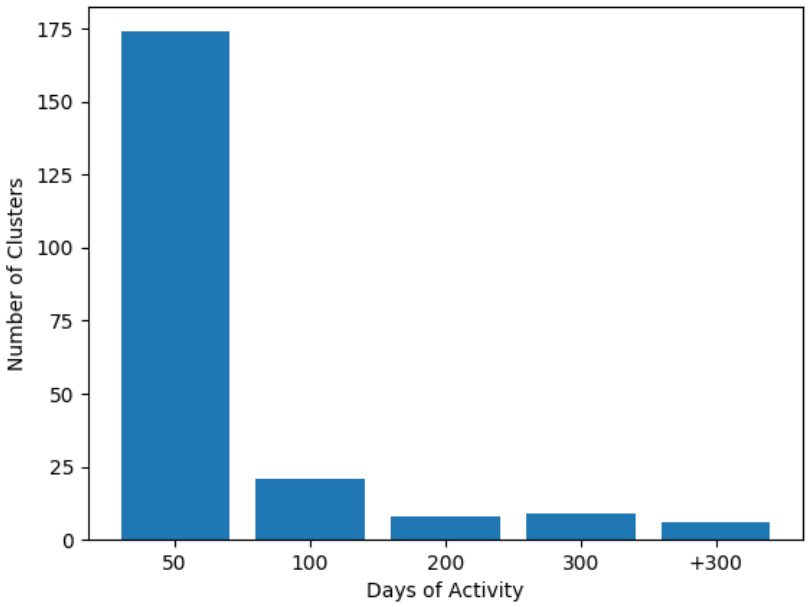

Figure 4. Histogram of clusters by days of activity.

AddressesSampled From active addresses in 2017, 31,000 were sampled;
ColdExchange This data set if formed by all 192,984 Cold Wallets detected;
ColdTransaction This data set if formed by all 296,536 Cold Storage transactions;
SampledTransactioin 6,000 samples of transactions with 20 or more entries in Input occurred in 2017.

For all data sets used, additional information was obtained from `blockchain.info`.

3.1. Financial Profile

The Gini coefficient[Gini 1921] and Lorenz Curve[Lorenz 1905] are statistical indicators used to describe the dispersion of the wealth of a population. These two indicators have been used to characterize and describe the financial aspects of Blockchain in several recent studies [Kondor et al. 2014, Vasek and Moore 2015].

In the Figure 5 we present the Lorenz curve of the **ColdTransaction** and **SampledTransactioin**, we can see a marked distinction in this indicator for these groups. Each transaction is composed of a set of inputs, in Figure 6 we show the distribution of the Gini coefficient of the inputs for the transactions **ColdTransaction** and **SampledTransactioin**.

A characteristic of Bitcoin is that the creator of the transaction chooses the fee to be paid to the miners, this rate will influence the time in which the transaction takes to be included in the Blockchain [Antonopoulos 2014]. In Figure 7, the fees paid by **ColdTransaction** and **SampledTransactioin** are shown.

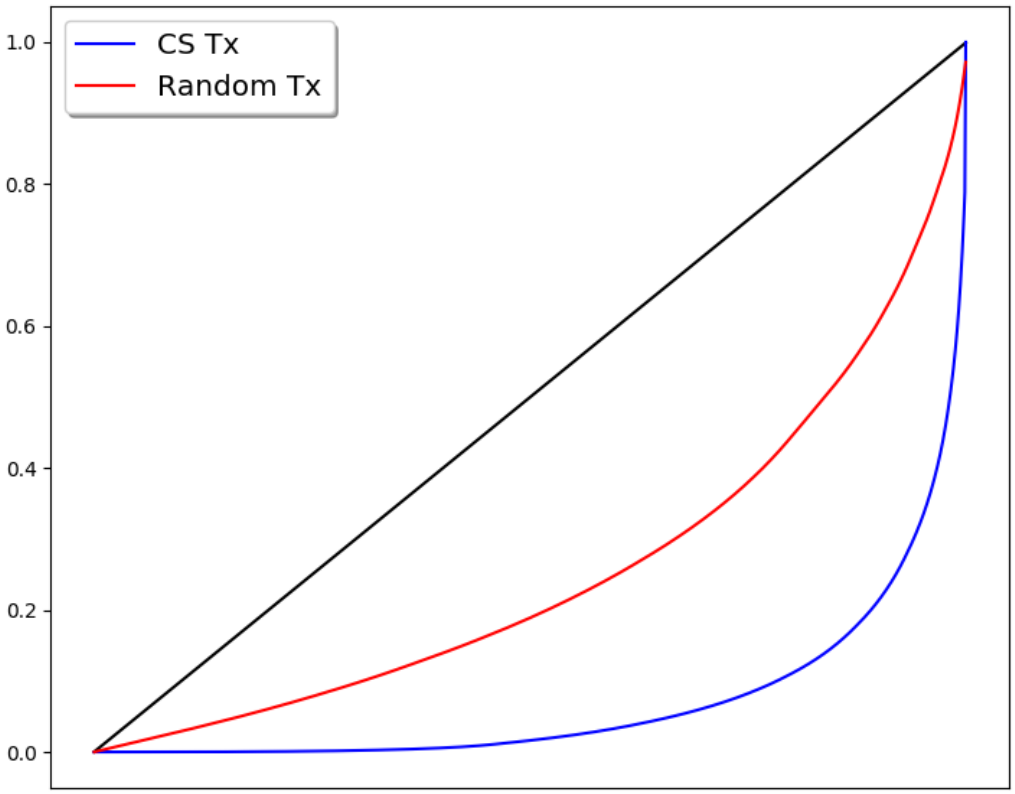

Figure 5. Lorenz Curve for ColdTransaction and SampledTransactioin.

3.2. Behavioral Profile

A address operated by an exchange must have a very specific transaction behavior, caused indeed by Cold Storage operations. In Table 2, we present the comparison between **ColdClients** and **AddressesSampled** of some behaviors that we expect to occur differently in these sets:

Non-Zero Blocks As deposits in customer addresses are periodically transferred to more secure addresses, it is expected that the number of non-zero balance blocks are small;

Full Withdraw We consider a FullWithdraw operation the withdrawals that cause a zero balance in the address. Table 2 shows the ratio of addresses that exclusively performed this type of withdraw.

Table 2. Behavioral Profile of addresses: balance and withdraw.

	Non-zero Blocks	Full Withdraw
ColdClients	73.0	0.90
AddressesSampled	1583.2	0.41

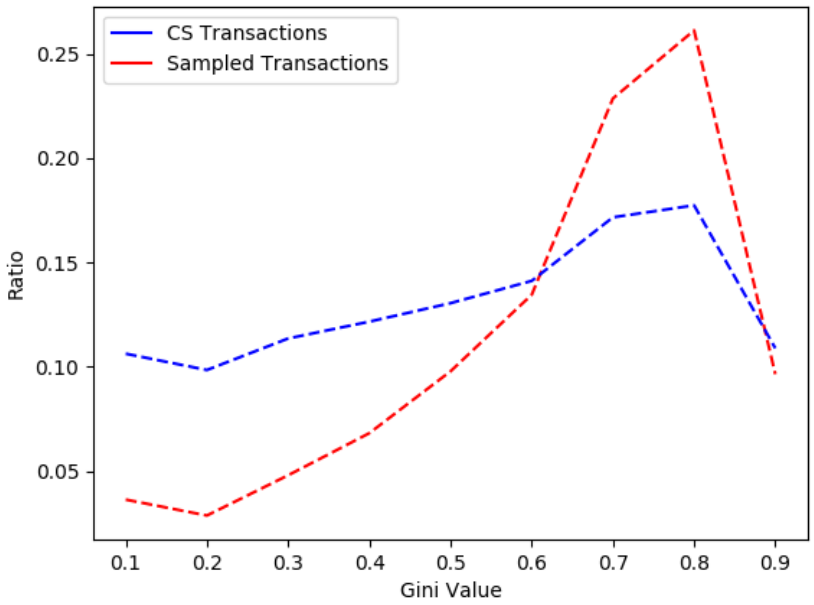

Figure 6. Distribution of the Gini coefficient of the Inputs of the transactions.

Addresses controlled by an exchange must have a close relationship with it:

- Third-party deposits should be moved quickly to some cold wallet;
- Payments to third parties must be previously funded by the exchange, probably from cold wallet.

Thus, each pair of transactions must contain at least one exchange-controlled addresses occurrence. In Figure 8 we present the frequency of occurrences of Cold Storage Addresses on **ColdTransaction**. It can be seen that most **ColdClients** have a close relationship with the Cold Wallets, performing most of their transactions with address operated by exchanges.

3.3. Provision of funds

A **ColdClients** address should have zero balance most of the time, so any expense should be preceded by a deposit made by the exchange. With this observation one should find at Blockchain deposits made by exchange - probably from Cold addresses - for your customers. Another pattern that can be found in Blockchain is *cross-provisioning*: when a Cold Wallet in a cluster makes deposits in a client address of another cluster. The results of the search for these patterns in the selected data set are shown in Table 3.

Table 3. Provision of funds occurrences.

Funding Transactions	44,989
Cross provision Transactions	5,733

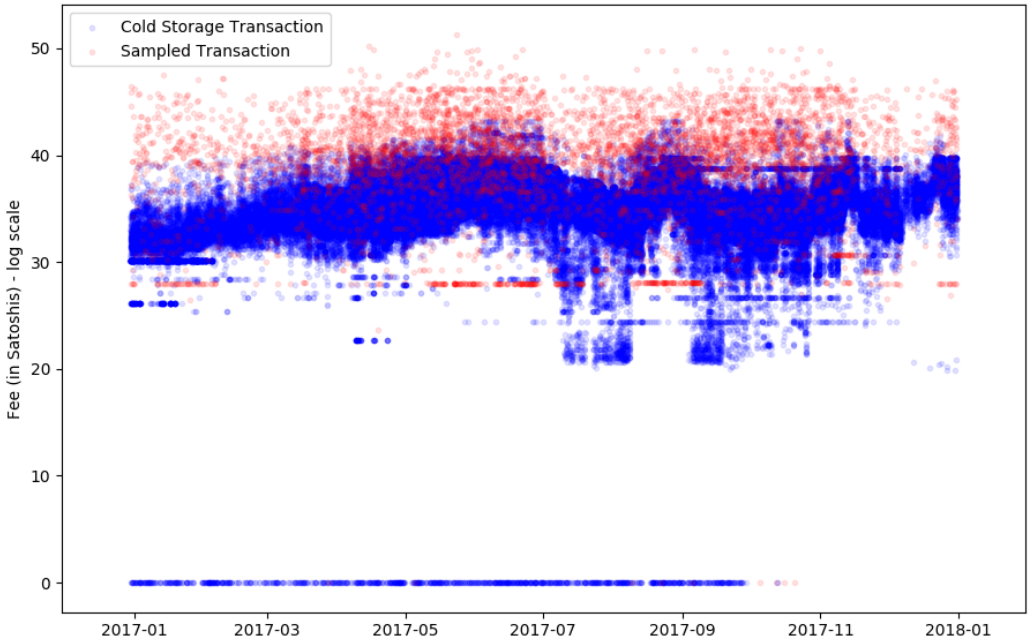

Figure 7. Fee Distribution.

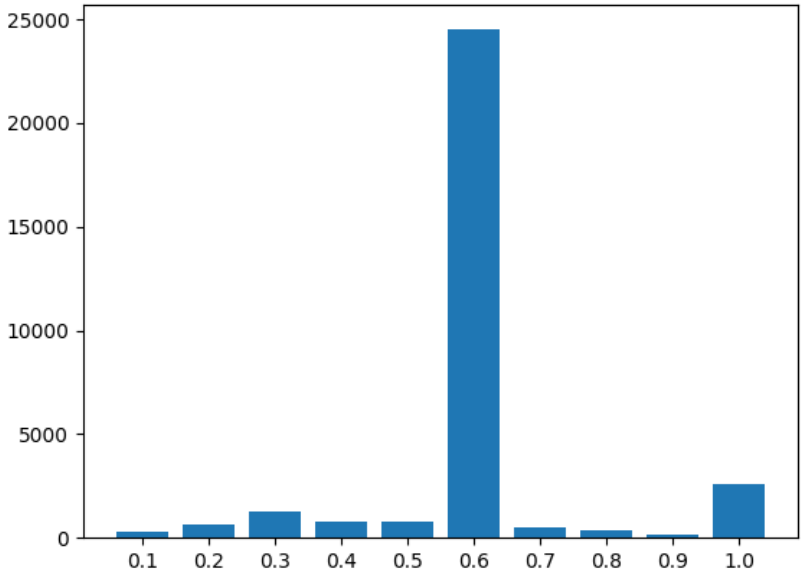

Figure 8. Frequency of Cold Storage Wallets in ColdTransaction.

4. Discussion

Cold Storage operations, despite being a recommended practice, do not have a standard, making it difficult to detect them. The use of Cold Storage has an impact on the clustering

process based on current heuristic: the exchanges that manage several addresses create huge clusters. Comparative analyzes of these cluster with the other Bitcoin users easily become meaningless. Even when exchanges disclose the address of their cold addresses for purposes of proof of ownership of Bitcoins and financial health, the cold operating addresses are still hidden.

The heuristic presented has its validity confirmed by the analyzes: the addresses and transactions have different behavior of the addresses and transactions sampled.

The analysis of the data set made in Sections 2.2 and 2.3 indicate that some addresses and transactions were wrongly classified: in Table 2 all addresses are expected to perform only Full Withdraw transactions. Similarly, in Figure 8 it is shown clients addresses with fewer transactions with the exchange than expected. We believe that these scenarios are a consequence of false positives in determining clients addresses or cold wallets not detected.

In Table 3, the existence of cross-provisioning transactions indicates that some clusters should be clustered.

The problem of wallets controlled by exchanges to form huge clusters has already been addressed in the literature (*e.g.* [Harrigan and Fretter 2016, Ron and Shamir 2013b, Meiklejohn et al. 2016]) , however without a method to detecting and addressing them.

5. Conclusions

The advance of the adoption of the Bitcoin brought interest in the analysis of Blockchain. An important step in these analyses is to cluster the addresses in an attempt to identify a single entity that has ownership over the Bitcoins deposited in them. The widely used clustering method fails when the addresses are operated by exchanges: they have control over addresses but not ownership over Bitcoins. This issue is evidenced in Cold Storage operations, which must be performed periodically by exchanges.

We presented a heuristic that identify such kind of transactions by the size of the entry, size of the output and the age of the transactions of the entries. The effectiveness of this heuristic was verified by comparing the characteristics of the selected transactions and addresses with the Blockchain data.

References

Akcora, C. G., Gel, Y. R., and Kantarcioglu, M. (2017). Blockchain: A graph primer. *CoRR*, abs/1708.08749.

Androulaki, E., Karame, G. O., Roeschlin, M., Scherer, T., and Capkun, S. (2013). Evaluating user privacy in bitcoin. In Sadeghi, A.-R., editor, *Financial Cryptography and Data Security*, pages 34–51, Berlin, Heidelberg. Springer Berlin Heidelberg.

Antonopoulos, A. M. (2014). *Mastering Bitcoin: Unlocking Digital Crypto-Currencies*. O'Reilly Media, Inc., 1st edition.

Chuen, D. (2015). *Handbook of Digital Currency: Bitcoin, Innovation, Financial Instruments, and Big Data*. Elsevier Science.

Filtz, E., Polleres, A., Karl, R., and Haslhofer, B. (2017). Evolution of the bitcoin address graph. In Haber, P., Lampoltshammer, T., and Mayr, M., editors, *Data Science – Analytics and Applications*, pages 77–82, Wiesbaden. Springer Fachmedien Wiesbaden.

Fleder, M., Kester, M. S., and Pillai, S. (2015). Bitcoin transaction graph analysis. *CoRR*, abs/1502.01657.

Gini, C. (1921). Measurement of inequality of incomes. *The Economic Journal*, 31(121):124–126.

Harrigan, M. and Fretter, C. (2016). The unreasonable effectiveness of address clustering. In *2016 Intl IEEE Conferences on Ubiquitous Intelligence Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld)*, pages 368–373.

Hileman, G. and Rauchs, M. (2017). Global cryptocurrency benchmarking study. Technical report, University of Cambridge, Judge Business School.

Kondor, D., Pósfai, M., Csabai, I., and Vattay, G. (2014). Do the rich get richer? An empirical analysis of the Bitcoin transaction network. *PLoS ONE*, 9(2).

Koshy, P., Koshy, D., and McDaniel, P. (2014). An analysis of anonymity in bitcoin using p2p network traffic. In Christin, N. and Safavi-Naini, R., editors, *Financial Cryptography and Data Security*, pages 469–485, Berlin, Heidelberg. Springer Berlin Heidelberg.

Liao, K., Zhao, Z., Doupe, A., and Ahn, G. J. (2016). Behind closed doors: Measurement and analysis of CryptoLocker ransoms in Bitcoin. *eCrime Researchers Summit, eCrime*, 2016-June:1–13.

Lorenz, M. O. (1905). Methods of measuring the concentration of wealth. *Publications of the American Statistical Association*, 9(70):209–219.

Maesa, D. D. F., Marino, A., and Ricci, L. (2016). Uncovering the bitcoin blockchain: An analysis of the full users graph. *Proceedings - 3rd IEEE International Conference on Data Science and Advanced Analytics, DSAA 2016*, pages 537–546.

McGinn, D., Birch, D., Akroyd, D., Molina-Solana, M., Guo, Y., and Knottenbelt, W. J. (2016). Visualizing Dynamic Bitcoin Transaction Patterns. *Big Data*, 4(2):109–119.

Meiklejohn, S., Pomarole, M., Jordan, G., Levchenko, K., McCoy, D., Voelker, G. M., and Savage, S. (2016). A fistful of bitcoins: Characterizing payments among men with no names. *Commun. ACM*, 59(4):86–93.

Monaco, J. V. (2015). Identifying Bitcoin users by transaction behavior. *Proc.SPIE - Biometric and Surveillance Technology for Human and Activity Identification XII*, page 945704.

Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. *Bitcoin.org*.

Narayanan, A., Bonneau, J., Felten, E., Miller, A., and Goldfeder, S. (2016). *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*. Princeton University Press, Princeton, NJ, USA.

Ober, M., Katzenbeisser, S., and Hamacher, K. (2013). Structure and anonymity of the bitcoin transaction graph. *Future Internet*, 5(2):237–250.

Ortega, M. S. (2013). *The Bitcoin Transaction Graph Anonimity*. PhD thesis, Universitat Oberta de Catalunya.

Ranshous, S., Joslyn, C. A., Kreyling, S., Nowak, K., Samatova, N. F., West, C. L., and Winters, S. (2017). Exchange pattern mining in the bitcoin transaction directed hypergraph. In *Financial Cryptography Workshops*, volume 10323 of *Lecture Notes in Computer Science*, pages 248–263. Springer.

Ron, D. and Shamir, A. (2013a). How did dread pirate roberts acquire and protect his bitcoin wealth? *IACR Cryptology ePrint Archive*, 2013:782.

Ron, D. and Shamir, A. (2013b). Quantitative analysis of the full Bitcoin transaction graph. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 7859 LNCS, pages 6–24.

Spagnuolo, M., Maggi, F., and Zanero, S. (2014). Bitiodine: Extracting intelligence from the bitcoin network. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8437:457–468.

Tschorsch, F. and Scheuermann, B. (2016). Bitcoin and beyond: A technical survey on decentralized digital currencies. *IEEE Communications Surveys and Tutorials*, 18(3):2084–2123.

Vasek, M. and Moore, T. (2015). There's no free lunch, even using bitcoin: Tracking the popularity and profits of virtual currency scams. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8975:44–61.

Zhao, C. (2014). *Graph-based forensic investigation of Bitcoin transactions*. PhD thesis, Iowa State University.