There is an updated version of this paper called "An Inquiry into Money Laundering Tools in the Bitcoin Ecosystem"
available at http://maltemoeser.de/paper/money-laundering.pdf

Anonymity of Bitcoin Transactions

An Analysis of Mixing Services

Malte Möser
University of Münster
Leonardo-Campus 3
48149 Münster, Germany
malte.moeser@uni-muenster.de

ABSTRACT

Bitcoin, a distributed, cryptographic, digital currency, gained a lot of media attention for being an anonymous e-cash system. But as all transactions in the network are stored publicly in the blockchain, allowing anyone to inspect and analyze them, the system does not provide real anonymity but pseudonymity. There have already been studies showing the possibility to deanonymize bitcoin users based on the transaction graph and publicly available data. Furthermore, users could be tracked by bitcoin exchanges or shops, where they have to provide personal information that can then be linked to their bitcoin addresses. Special bitcoin mixing services claim to obfuscate the origin of transactions and thereby increase the anonymity of its users. In this paper we evaluate three of these services – Bitcoin Fog, BitLaundry, and the Send Shared functionality of Blockchain.info – by analyzing the transaction graph. While Bitcoin Fog and Blockchain.info successfully mix our transaction, we are able to find a direct relation between the input and output transactions in the graph of BitLaundry.

Keywords

anonymity, bitcoin, blockchain, laundry, mix, pseudonymity, shared wallet, transaction

1. INTRODUCTION

Bitcoin is a distributed, cryptographic digital currency that is developed by an open source community. The idea behind it was proposed in 2008 under the pseudonym Nakamoto [16]. In order to send and receive bitcoins (BTC), a user has to create a key pair, which consist of a public key, that serves as an account identifier, and a private key, that is used to sign transactions. Each transaction has a list of inputs and outputs. The inputs refer to previous transactions, that contain a certain amount of bitcoins, in order to enable all members of the network to verify, that these coins have not already been spent. The transaction usually has two outputs, one output destination is the address (public key) of the recipient, the other output belongs to the sender of the bitcoins. As the value of a previous bitcoin transaction cannot be spent partially, the surplus is returned to the sender as change. To increase the anonymity of the user, the reference implementation of the bitcoin wallet (Bitcoin-Qt), a software that manages addresses and makes it easy to send transactions, automatically generates new addresses, that are used whenever an address for change is required.

Bitcoin uses a proof-of-work system to verify transactions and to prevent double-spending. Conflicts in the system are resolved by majority decisions, with the weight of the vote based on computational power. On average, every ten minutes a new block is created, which bundles a number of valid transactions and refers to the previous block, thereby extending the blockchain. To check, whether the inputs of a transaction have already been spent, all clients keep an index of unspent transactions and reject those with invalid inputs from being integrated into a block [20].

Bitcoin gained a lot of media attention for being an anonymous digital currency (e.g., [25]), especially since organizations like WikiLeaks described it as a "secure and anonymous digital currency", that "cannot be easily traced back to you" [10]. However, due to the fact that all transactions are stored publicly in the blockchain, the anonymity of a sender relies on the pseudonym not being linked to his true identity. The bitcoin community itself states that "the current implementation is not very anonymous" [2].

Usually, people have to provide personal information in order to buy bitcoins. The popular bitcoin exchange Mt. Gox for example just recently increased its identity requirements. Anyone who wants to deposit or withdraw currencies other than BTC has to provide a scan of their national ID [15]. To unlink the bitcoins from a persons true identity, they could try to use a mixing service to transfer bitcoins to a new, anonymous address. Bitcoin mixes are services, that claim to increase anonymity by mixing the coins of multiple users, making it harder to find a relation between input and output transactions in the transaction graph. Another possible scenario could be, that an attacker monitors addresses, which are known to belong to a certain person or organization, e.g. WikiLeaks. A bitcoin user could now use a mixing service to make an anonymous donation without the danger of being tracked down by the attacker.

Münster Bitcoin Conference (MBC), 17–18 July '13, Münster, Germany

There is an updated version of this paper called "An Inquiry into Money Laundering Tools in the Bitcoin Ecosystem"
available at http://maltemoeser.de/paper/money-laundering.pdf

In this work, we evaluate whether bitcoin mixing services can increase the anonymity of its users. We test three services and try to establish connections between the input and output transactions in the transaction graph. Its structure gives us hints on how the service works and how this might affect anonymity. We find out, that while the service BitLaundry does not provide good anonymity, both Bitcoin Fog and Blockchain.info make it impossible for us to find any direct connections in the transaction graph. However, they cannot provide real anonymity because the user has to trust the service not to keep any transaction logs.

The rest of this paper is organized as follows. Section 2 presents the idea of mixes in the context of bitcoin transactions and models to measure the anonymity they provide. Section 3 evaluates three mixing services by analyzing the transaction graph, trying to find connections between the input and output transactions. Section 4 presents related work on digital currencies, the anonymity of Bitcoin and mixing services. Finally, Section 5 discusses the limitations of the analysis and ideas for future work.

2. ANONYMITY OF TRANSACTIONS

In order to analyze, how mixing services can provide or increase anonymity for bitcoin users, we will start by defining anonymity. Anonymity means, that an entity inside a set of other entities (the anonymity set) is not identifiable [18]. In a communication network, the anonymity set can be divided into the sender anonymity set and the recipient anonymity set. Unlinkability in this context means, that an attacker cannot decide, whether a certain sender communicates with a certain recipient.

Although a system might achieve high anonymity on a global level, the anonymity of a certain entity in the system can be low, when an attacker has context information available that enables him to reduce the anonymity set. Instead of looking at the global level of anonymity in the bitcoin system, which was done before in [17, 21], we will focus on the linkability of transactions.

2.1 Transaction Network

In the Bitcoin network, a user does not physically own bitcoins. The possession of bitcoins is stored in the blockchain as outputs of a transaction, that refer to the address of a recipient. A transaction represents a payment, that is digitally signed with the private key of the previous owner of a certain amount of bitcoins, who now wishes to reassign the possession of the coins to the public key specified in the transaction [26]. The amount of bitcoins, a user owns, can be calculated as the sum of all unspent transactions that belong to his addresses.

A bitcoin transaction has a list of one or more previous transactions as an input. It has to spend the whole cumulative value of the input transactions, otherwise bitcoins would be lost. Therefore, a standard transaction usually has two output addresses, of which one belongs to the sender who receives the change of the transaction, the other belongs to the payee. Using the references to the previous transactions in the list of inputs, it is possible to build a transaction graph. A simplified example is shown in Figure 1, where an output of t_A is used as an input in t_B, and outputs of t_B and t_C are

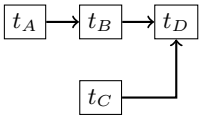

Figure 1: Example transaction graph.

inputs to t_D. Although a transaction keeps a pointer to the previous transaction, we will draw the edges pointing to the next transaction, visualizing the flow of bitcoins.

By grouping together the addresses of inputs to a multi-input transaction, a user graph can be created, which has been analyzed in [21, 22]. However, in this paper we will focus on the transaction graph to trace back a number of output transactions (transactions going out of a mix) to the related input transactions (that went into the service).

2.2 Mixes

The services analyzed in this paper are often referred to as *mixing services*. The basic idea of a mix, which was presented by Chaum [7] in 1981, is to ensure anonymous communication between two parties. Figure 2 shows the basic idea, where the relation between Alice and Bob is hidden by the service. A mix takes a number of inputs, that have been encrypted with the public key c_M of the mix and contain an encrypted message $c_A(z_0, m)$, a destination address A, and a random string z_1 in order to make the size of each incoming message equal. The mix then decrypts the message, removes the random string (cf. Equation (1)) and forwards the again equal sized, encrypted messages to the associated addresses in batches. If the number of inputs is large enough, it is not possible to link inputs to the corresponding outputs.

$$c_M(z_1, c_A(z_0, m), A) \rightarrow c_A(z_0, m), A \tag{1}$$

In order to reduce the danger of a single mix being the attacker, who would known the relation between inputs and outputs, multiple mixes can be linked together, creating a mix cascade. The user then has to encrypt his message with the public keys of all mixes, which ensures that each mix only sees an encrypted message and the next destination address.

Figure 2: A mixing service, that hides the relation between Alice and Bob.

2.3 Shared Wallets

Due to the design of the bitcoin system, all transactions are publicly stored in the blockchain. It is not possible to bundle encrypted transactions and forward them anonymously, because the origin of a transaction input must always be specified in order to prevent double-spending. Therefore it is not possible to design a bitcoin mixing service like a traditional mix.

There is an updated version of this paper called "An Inquiry into Money Laundering Tools in the Bitcoin Ecosystem"
available at http://maltemoeser.de/paper/money-laundering.pdf

For the sender it makes no difference, whether a payee receives a bitcoin, that in the past belonged to him, or to another random bitcoin user, as long as the amount of bitcoins stays the same. Mixing services can therefore use the concept of a shared wallet [17, 24]. The service provider owns a set of addresses, to which the user can send bitcoins to. Once a payment has been confirmed, the amount of bitcoins is transferred to the destination address using a different address, that is not linked to the first address. Usually, the operator takes a small transaction fee that is deducted from the outgoing transaction. A simplified example of the concept is shown in Figure 3.

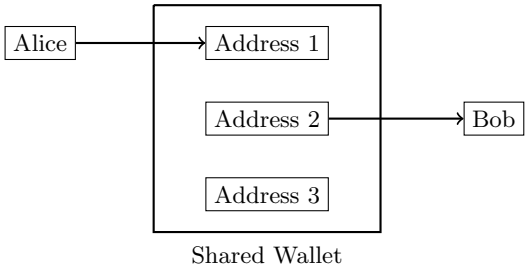

Figure 3: The shared wallet hides the relation between Alice and Bob by using a different address to pay out bitcoins to Bob.

If an attacker monitors the addresses of a user and knows, how many bitcoins he transferred into the service, he could try to use this additional information to attack the anonymity the service provides by searching for an equal-sized output transaction. That's why many services advise their users not to pay out the full amount of bitcoins they previously payed in [3]. Furthermore, they are encouraged to split the outgoing transaction into multiple, smaller transactions and to spread them over a period of time, making it harder for an attacker to link them together.

To achieve anonymity there need to be enough users and bitcoins in the mix, otherwise the same coins might get payed out, that the user just payed in (e.g., [4]). Of course, a service could prevent this from happening by delaying the payout until enough other coins are available in the system. The larger the amount, the user wants to transfer anonymously, the harder it might be to mix the coins with others. Unfortunately, due to our limited amount of bitcoins, we can not evaluate this by paying in large amounts of bitcoins.

The big problem with all bitcoin mixes currently available is, that they require a central instance, that keeps logs for a certain time in order to route the bitcoins through the system. The user has no chance to make sure that these logs are being deleted afterwards. Furthermore, a possible attacker could be the service itself, who would have complete knowledge about who sends which amount of bitcoins to whom. While using multiple mixes can reduce the risk, this comes with a cost increase, which is shown in Section 3.5.

2.4 Measuring Anonymity

In order to evaluate the anonymity a bitcoin mix can provide, we need a way to measure the degree of anonymity.

Diaz et al.

The model of Diaz et al. [8] measures the degree of anonymity by comparing how much influence the information, an attacker was able to gain by observing a system, has on the anonymity set in contrast to the ideal situation, where every sender has the same probability of being the origin of a message. The degree of anonymity d can be calculated by comparing the entropy of the system including the knowledge of the attacker to the maximum entropy:

$$d = \frac{H(X)}{H_M} \tag{2}$$

H_M is the maximum entropy in a system with N users:

$$H_M = log_2(N) \tag{3}$$

$H(X)$ is the entropy of the attacked system, where the attacker assigns probabilities to each possible sender

$$H(X) = -\sum_{i=1}^{N} p_i * log_2(p_i) \tag{4}$$

Taint Analysis

Blockchain.info offers a service called taint analysis[1], which calculates the correlation between two addresses [19]. It is important to note, that there is another understanding of taint in the bitcoin community, which means the percentage of bitcoins, that come from a known theft of bitcoins and have been blacklisted by popular exchange markets.

The taint analysis works by calculating the percentage of the amount of bitcoins that might origin from another address, thus revealing connections in the transaction graph. In the simplified example in Figure 4, A_1 and A_3 would have a taint of 75% and A_2 a taint of 25%. However, it can only detect direct connections in the graph and does not consider any context information.

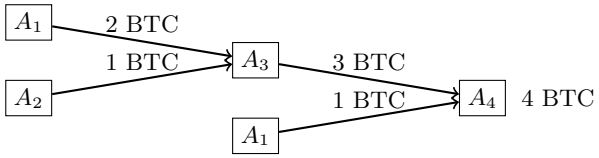

Figure 4: Taint Analysis.

3. ANALYSIS

The mixing services evaluated in this section are either directly accessible on the internet, or require us to connect via the Tor[2] network. Depending on the functionality they offer, we have to create an account, or specify the relevant parameters directly on the website and receive a single-use address to send bitcoins to. The account-based websites usually act like a virtual wallet, allowing us to deposit and withdraw bitcoins. We pay in small amounts of bitcoins, that can be payed out once the input transaction is confirmed. We can specify the amount of bitcoins to withdraw, one or more destination addresses, the number of output transactions and

[1] `http://blockchain.info/de/taint/_HASH_`
[2] `https://www.torproject.org/`

There is an updated version of this paper called "An Inquiry into Money Laundering Tools in the Bitcoin Ecosystem"
available at http://maltemoeser.de/paper/money-laundering.pdf

a time period, over which the transactions are spread. For each experiment, we use one or multiple newly generated destination addresses belonging to our own, private bitcoin wallet.

Once we receive the payment, we gather the relevant blockchain data using the API of *Blockchain.info*[3]. We build the transaction graph by following the inputs of the outgoing transaction, as described in Section 2.1, and visualize it using the open source software Gephi[4]. Inspecting the transaction graph, we try to understand how the service works and to identify patterns or special characteristics. Furthermore, we try to find direct connections between the input and output transactions and using both a local search as well as the taint analysis tool presented in Section 2.4.

3.1 Services

As of May 2013, there are multiple bitcoin mixing services available:

OnionBC[5] is an online bitcoin wallet accessible only via Tor. It offers the functionality to send transactions anonymously, for which it takes a fee of 3%[6] with a minimum transaction size of 0.5 BTC. Furthermore, it offers an escrow service, that can be used by users buying goods online and paying with bitcoins to hold back a payment until they received the goods they ordered.

Bitcoin Fog[7] is another service only accessible via Tor. It allows the generation of up to 5 addresses for depositing bitcoins and takes a (random) fee between 1–3% of the transaction value. Bitcoins can be withdrawn to a maximum of 20 addresses, spread over a timespan of 6–96 hours with a minimum total of 0.2 BTC.

BitLaundry[8] is a simple mixing service, that, in contrast to OnionBC and Bitcoin Fog, does not allow to deposit bitcoins into a virtual wallet. Instead, the destination addresses, the number of outgoing transactions and a time span have to be specified and a single-use address is generated, where the user has to send at least 0.25 BTC to. The mixing fee for BitLaundry is split into two parts. The first is 2.49% of the total, the second is 0.00249 BTC per outgoing transaction.

Blockchain.info offers a service called *Send Shared*[9] that uses a shared wallet to swap the bitcoins between different users. It takes a mixing fee of 0.5%, making it the cheapest service in this comparison, and requires a minimum transaction size of 0.2 BTC.

On 13 April 2013 the bitcoin forum user *BlindMixrDR* announced a new mixing service[10] that would combine bitcoin and a blind signature scheme. Unfortunately, the service and detailed information about the system are not available anymore.

In the following we analyze the three services Bitcoin Fog, BitLaundry and Send Shared of Blockchain.info, a comparison is given in Table 1. We exclude OnionBC from our analysis due to concerns regarding the trustworthiness of the service, as we were not able to find any positive reviews of it on the bitcoin boards and the minimum deposit size of 0.5 BTC is rather high.

3.2 Bitcoin Fog

Experiment

After creating an account for the service Bitcoin Fog, we get a newly generated address for deposits. For our first attempt, we send 0.3 BTC to this address (cf. Table 2). As of 28 June, almost two months later, these bitcoins have not been moved yet. After the deposit is confirmed by Bitcoin Fog, we withdraw the whole amount using three destination addresses, of which only two receive a transaction later on, an offset and a time span over which the transactions will be spread.

We can now analyze the transaction graph of the outgoing transactions. Building the graph reveals an interesting pattern: both transactions t_2 and t_3 have only one large input transaction with a size of about 474 BTC. The time difference between the transactions is only 15 minutes, and as the graph in Figure 5 shows, there is only one transaction between them.

Figure 5: Chain of input transactions.

We extend this graph, trying to identify the origin of the large transaction. After 1445 single-input transactions is a transaction[11] that took place on 20 April and combines five big transactions with a total of 6,013 BTC. Following those 5 transactions and using a community detection algorithm [5], we can identify five big communities, where a large number of transactions are bundled into one (cf. Figure 6). On the right side, they are connected by a few single-input chains, that were probably used to pay out bitcoins to other users. We cut off the graph at the edges of the communities.

In one of the communities we find a transaction with a size of 44,039 BTC. The coins origin from an even larger transaction[12] that bundles a large number of inputs to a total of 50,000 BTC. While we cannot be sure that these belong to the same service, the transactions show the same pattern of a long, single-input chain paying out small amounts to different bitcoin addresses.

We take a closer look at the first chain of single-input transactions. By comparing the size of a transaction with the size of the previous one, we calculate the amount of bitcoins that has been payed out in each transaction. The minimum

[3] `http://blockchain.info/api/blockchain_api`
[4] `https://gephi.org/`
[5] `http://6fgd4togcynxyclb.onion`
[6] While the frontpage states a fee of 2%, the transaction view says 3%.
[7] `http://bitcoinfog.com`
[8] `http://app.bitlaundry.com`
[9] `https://blockchain.info/de/wallet/send-shared`
[10] `https://bitcointalk.org/index.php?topic=175959.0`
[11] e315f8c1cb7a85762d07511d41c7e621bcd83000185ed51443a9a72370346667
[12] 443d8f0511ec1f77132b06c739bb6bf29f008dc58a373fa511ab1b182390c4fe

There is an updated version of this paper called "An Inquiry into Money Laundering Tools in the Bitcoin Ecosystem"
available at http://maltemoeser.de/paper/money-laundering.pdf

Table 1: Comparison of mixing services.

Service	Input		Output			
	No. of Addresses	Online Wallet	Fee	Multiple Transactions	Time Span	Minimum Transaction Size
Bitcoin Fog	5 per Account	yes	1–3%	1–20	6–96h	0.2 BTC
BitLaundry	1 per Tx	no	2.49% + 0.00249 per Tx	1–10 per day	1–10 days	0.25 BTC
Blockchain.info	unlimited	yes	0.5%	no	no	0.2 BTC

Table 2: Bitcoin Fog transactions.

	Time	Type	Value	Hash
t_1	2013-04-29 07:23	In	0.3	`97e723ded27cd1e4f9954689c503d092fe5a1b79747d6c45b18ad8f90bf61c62`
t_2	2013-04-30 08:45	Out	0.2052473	`56a4f35b4a2fb5eb15549befdb1285e831a5dd67bc1b559c1b2ef8e145627856`
t_3	2013-04-30 09:00	Out	0.08804699	`8f4bf3e95c00025d42fc2c6a9f28e66c7ed75eb08560b7675c712accb1d75b2c`
t_4	2013-05-07 20:13	In	0.3141593	`ac8d82b3c3088a633fc4b48562e8c5794f502acbfbec360b406958e0acc92451`
t_5	2013-05-14 08:36	Out	0.1104155	`18ee1ea93a9c84dd5f1e7bd758410368e545a45a989aafcd78584f51c3da4566`
t_6	2013-05-15 20:22	Out	0.1019295	`a95e2fea5498dae5ec3419d8d5c62dea23b09d69923eb15e829a562a6975a962`

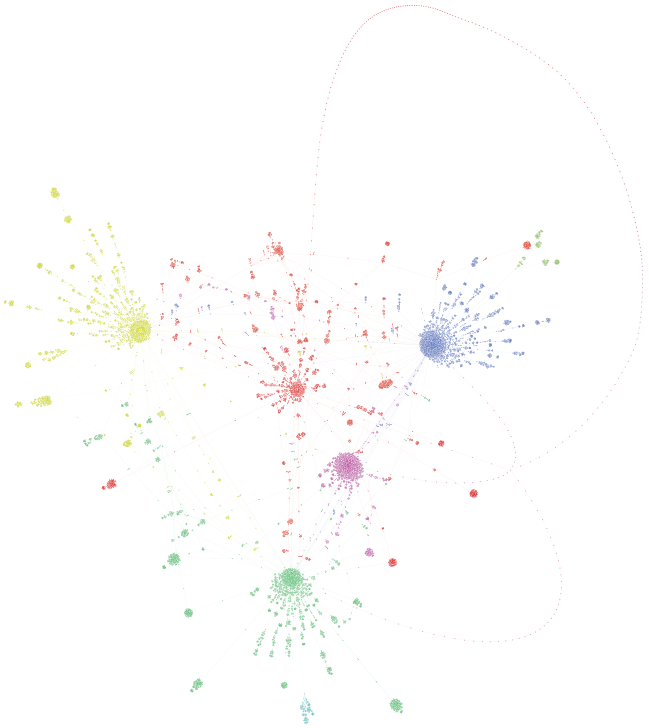

Figure 6: Communities in the first Bitcoin Fog transaction graph.

payout amounts to 0.04239 BTC, the maximum to 717.94096 BTC. The average payout size is 3.8328 BTC with a standard deviation of 24.5344 BTC. The distribution of the payout sizes is shown in Table 3. Most transactions have a size between 0.1 and 5 BTC, with a median of 0.80111 BTC. The large difference between median and mean can be ascribed to a few, large transactions. As the anonymity set for large transactions is small, it can be easier to detect those.

Table 3: Distribution of the payout size.

Larger or equal to	Smaller than	# of transactions
0	0.1	98
0.1	0.5	438
0.5	1	275
1	2	272
2	5	217
5	10	79
10	50	52
50	100	5
100	500	8
500	1000	1

Using the measurement model of Diaz et al. [8], we can evaluate how the knowledge of an attacker on the size of a transaction influences the degree of anonymity in the anonymity set. Assuming, that a passive attacker knows whether the sender, he is interested in, had only a little or a large amount of bitcoins available, we form two groups. The first group contains all transactions up to a size of 2 BTC, the second group the transactions larger than 2 BTC. The attacker can now assign a probability p to the first group, that the sender is within this anonymity set.

$$p_i = \frac{p}{1083}, 1 \leq i \leq 1083;\ p_i = \frac{1-p}{362}, 1084 \leq i \leq 1445$$

In Figure 7, the distribution of the degree of anonymity for p is shown. It never drops below 0.8, which Diaz et al. consider to be the lower bound a system should provide, and reaches its maximum at $p = 0.75$. Thus, the anonymity of the sender in this scenario is high.

A week after the first experiment, we make a second deposit of 0.3141593 BTC. This time we withdraw 0.212345 BTC, spread over two transactions and two days. Again, we create the transaction graph of the ingoing transactions and see a long chain of single input transactions. It originates from a

There is an updated version of this paper called "An Inquiry into Money Laundering Tools in the Bitcoin Ecosystem"
available at http://maltemoeser.de/paper/money-laundering.pdf

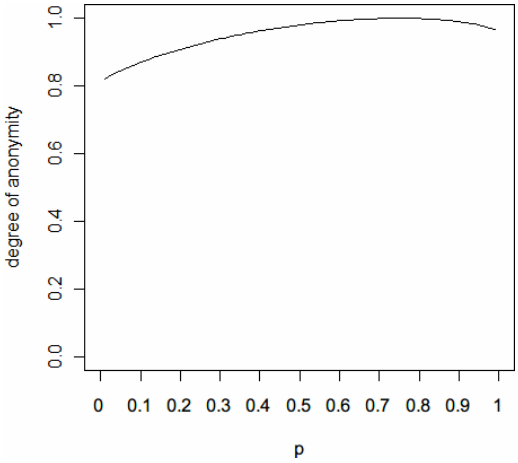

Figure 7: Distribution of the degree of anonymity.

transaction[13] that, similar to the communities in the first experiment, combines multiple transactions into one, with a total value of 942.88 BTC. In the graph, shown in Figure 8, are 30 coinbase transactions, with a total value of 683. If we increase the depth of the graph this size increases, however we cannot determine whether they belong to the service or not.

The transaction size ranges between 0.04 and 36.83 BTC, with an average transaction size of 1.89 BTC at a standard deviation of 3.72. Again, the median of 0.745 BTC is lower than the average due to some large output transactions. Similar to the first experiment, our input transaction has not been spent yet, making it impossible to find connections in the transaction graph.

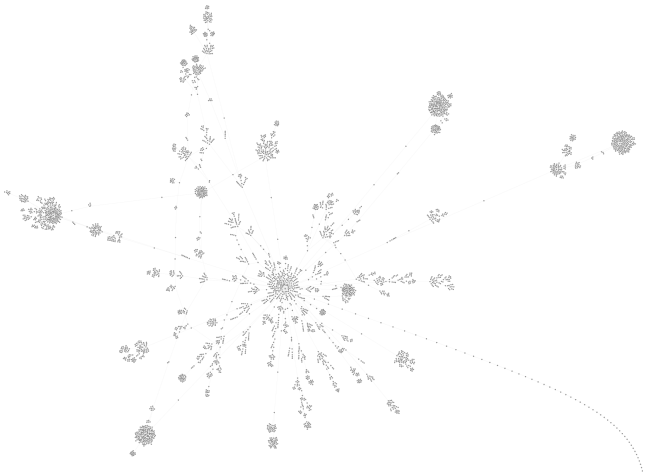

Figure 8: Transaction Graph of the second Bitcoin Fog experiment.

Results

The service Bitcoin Fog bundles a large amount of transactions into single, large transactions, which are then used

[13] d7cfafaba42d952fee3ec4617f07d40808bc52fd14e507cd7fb2e0e168d40635

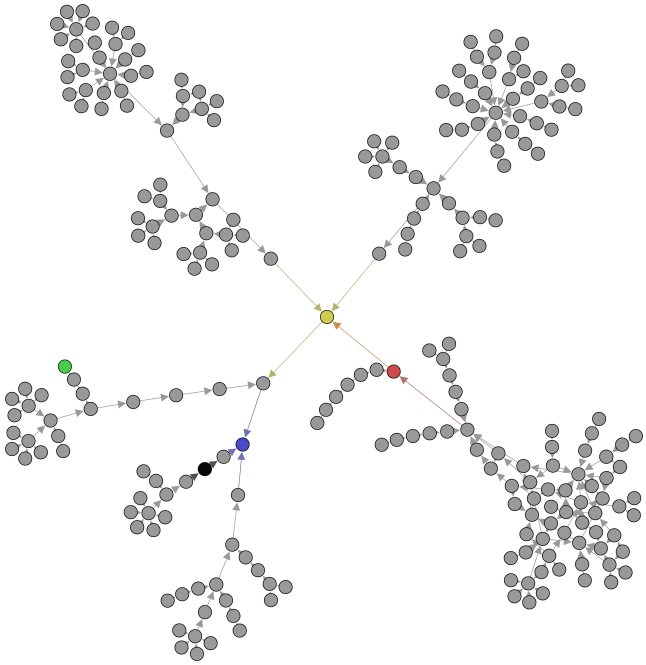

Figure 9: Transaction graph of the first BitLaundry experiment.

to create all outgoing transactions. The input transactions however remain untouched for a long time. Thereby, the service prevents us from detecting any direct connections between the input and output transaction in the transaction graph. It may however be possible to decrease the anonymity of transactions using additional context information due to the clear structure of the service. Therefore, the user should split his output transactions in such a way, that they have the most common size, e.g. between 0.1 and 5 BTC, to decrease the chance of being identified.

3.3 BitLaundry

Experiment

The second service analyzed is BitLaundry. On 13 March we deposit 0.33158651 BTC in order to be transferred to a single address, split up into two transactions over a period of 2 days. Instead of two, we receive four transactions (cf. Table 4). The first observation is, that the payouts seem to take place at 10:45 p.m. and 12:15 a.m.

The transaction graph, visualizing the flow of the incoming transactions, is shown in Figure 9. The transactions are colored as follows: t_8 = red, t_9 = yellow, t_{10} = green, t_{11} = blue. In contrast to our experiment with Bitcoin Fog, we also find our deposit transaction t_7 in the graph, it is colored in black. All five transactions are connected to each other.

A large part of the input transaction is forwarded to an address[14], that over a timespan of 14 days received and sent about 18.45 BTC. A little amount of 0.0244 BTC, 7.79% of the total amount, is going directly into t_{11}, which means that there is a direct connection between our input transaction and one of the output transactions in the transaction graph. The

[14] 1KdPv6GWpg6eoj6cxcV65uc1NwufvhtGGQ

There is an updated version of this paper called "An Inquiry into Money Laundering Tools in the Bitcoin Ecosystem"
available at http://maltemoeser.de/paper/money-laundering.pdf

Table 4: BitLaundry transactions.

	Time	Type	Value	Hash
t_7	2013-05-13 20:04	In	0.33158651	3f574ac9026d265250fb987468346dc84a339d6ae3741356940aed723579aab5
t_8	2013-05-13 22:45	Out	0.09387001	50b78013f4e5a7acea29e721179e9ead6742bc9c9993b41d26c95fd13591f210
t_9	2013-05-14 12:09	Out	0.0818	bbb6320539a61abfde853e1ee684ec9430d19fa40926b1abe27a22bcfa7daf16
t_{10}	2013-05-14 22:43	Out	0.0782	529f930f65001a6ef519c54c7c5ad463db864cce5656fdd706ab4c5d91792845
t_{11}	2013-05-15 12:22	Out	0.0595	5fcf3ea2565672a65a389de99653a9672fa06a0c7ad90c17231bd354d2422767
t_{12}	2013-06-22 20:45	In	0.31415	9809ab21a659724b1c52cdd22427c83420f486df3935f17b0c1e3c0a1fc7b38a
t_{13}	2013-06-23 01:18	Out	0.30383767	2f917d2a38e68b99d87c47b8a78db1c8f4d7310840c23c9f9e84239dabae8cdd
t_{14}	2013-06-24 15:56	In	0.332711	6078f4779354d3cd8902be6703c0f5bb2b13417f43c60e62ed0f6375acd66a09
t_{15}	2013-06-25 00:56	Out	0.16055895	06e5b3c0d5e3be98abd8f1cd18fc91370f6e9161e4085184f11680d35ffd8af8
t_{16}	2013-06-25 16:26	Out	0.1584	238e5c60fbb09a92f8c4b6e0c94ca03658f6177ca6d50a68468aba5c4453f35d

taint analysis of Blockchain.info calculates a taint of 6.01%, which is not much, but the knowledge that the address, a user used to create a transaction, can be related to another address, might still be valuable for an attacker.

In the graph are several hubs, where multiple transactions are combined. If we increase the depth of the graph, the number of coinbase transactions increases. However, we do not know whether they belong to the service, and we assume that the system does not base on a large number of coinbase transactions for better anonymity.

We make two more experiments with BitLaundry in order to find out, if for example low usage of the service might lead to another connection in the transaction graph. Therefore, we first make a transaction with a size of 0.31415 BTC in order to be payed in one transaction withing one day. Its transaction graph does not reveal direct connections between input and output.

After that, we pay 0.332711 BTC into the service, in order to be transferred within one day, spread over two transactions. This time, we can find a direct connection between the input t_{14} and the first output t_{15} in the transaction graph (cf. Figure 10), which results in a taint of around 50%. The second output t_{16} is not connected to the input.

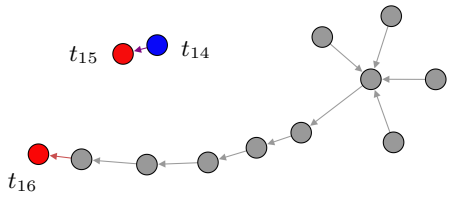

Figure 10: Transaction graph of the third BitLaundry experiment.

Results

In the first experiment, we were able to find a connection between one output and our input. The second experiment did not reveal any connections. However, in the last experiment the service directly used half of the input transaction to create an output transaction. We can conclude, that it does not provide high anonymity. A reason for this could be a low usage of the service as well as a lack of technical measures to ensure that users do not receive their input coins back.

3.4 Blockchain.info

Experiment

The third service analyzed is the *Send Shared* functionality of Blockchain.info. In contrast to Bitcoin Fog and BitLaundry, it offers neither the option to split the transaction into multiple smaller ones, nor to spread the payments over a certain period of time. However, this is not a big problem, as someone can always split a single transaction into multiple manually.

We send 0.40012345 BTC into our online wallet and, only 6 minutes later, use the shared wallet feature to send them to another address. As we cannot detect any special patterns in the transaction graph we create eleven additional transactions in order to increase the chance of, for example, getting multiple coins from the same address.

We are not able to find any direct connections between the input and output transactions. However, instead of twelve there are only eight separate graphs (cf. Figure 11), meaning that there are connections between multiple outputs. Furthermore, there are hubs where a large number of transactions are bundled into one transaction, but we find only a few coinbase transactions, which means that mainly the coins of other users are used. The transactions, that are connected to multiple output transactions, suggest that bitcoin transactions are bundled into larger ones and then split again for payouts. One example is shown in Figure 12, where the red nodes represent output transactions and the green nodes represent transactions, that are connected to multiple output transactions. Following the left, green transaction, we find an address[15] that bundles transactions to a total size of 2,000 BTC, which is then split into eight transactions with a size of around 250 BTC each, and then again into smaller transactions.

Results

Although our input has been used by the service, it is not possible to find any direct connections between the input and output transactions. The service bundles a large number of small transactions into larger ones, which are then split

[15] 13udyfBcdA2PUDCFM69VYDEHRRFnqkjEkx

There is an updated version of this paper called "An Inquiry into Money Laundering Tools in the Bitcoin Ecosystem"
available at http://maltemoeser.de/paper/money-laundering.pdf

Table 5: Blockchain.info transactions.

	Time	Type	Value	Hash
t_{17}	2013-05-27 16:09	In	0.40012345	c8536ce1809f296d9ed82c37a406a5cb01b63c780aa5b76324a2f26c1a7063cd
t_{18}	2013-05-27 16:15	Out	0.39713345	7fa8bf0c9c346a3e1b57ce1540947369342741172 9ac5664487ce6f811016517
t_{19}	2013-05-27 16:18	In	0.21262121	e72bf981bdf893a0acf55f9c54cab361c476a2bdf131d5127cc03ce105e79702
t_{20}	2013-05-28 15:55	In	0.4105	10ce8832084bb1625d180d71eafc79cdea46c24dd647e44e2a50c9309182892d
t_{21}	2013-05-28 16:15	Out	0.2	c70237e203a5d3d70d1b92ced9253240810228e7b947ac73afc4e75ab34393e1
t_{22}	2013-05-28 16:17	Out	0.2	6c4c0a974999c0f83fc2f4a581da223d3cc26f7b2eacccc85ebcf5a302e18f90
t_{23}	2013-05-28 16:19	Out	0.2	b45d9a2a45c9985a9e1236aaff70d6865c562c2d7184303ebadb4303c8246d2c
t_{24}	2013-05-28 20:02	In	0.6305	c2319a47c5811aaa00575343030e80b31fa482f243b297a650dfc8b12b6b660e
t_{25}	2013-05-28 20:05	Out	0.21	a3b0226c4fb44bbf0829c0be13b4dd4613daa517dd0c3616c651c04a3c06f43b
t_{26}	2013-05-28 20:08	Out	0.21	f5c3c844d9c1b7f48c45826059df7608af532d3528e05b60d9fd28c2aca3b78e
t_{27}	2013-05-28 20:13	Out	0.21992121	aab4d3d66f4a08c713e71becdd3c28cf9bf8fb34a29bf5f8d96dceb26bdecbe5
t_{28}	2013-05-28 20:52	In	0.5005	1fca72c0fe447c35a5db1cc6381cc9fde7439847354b01de773053e413ae9404
t_{29}	2013-05-28 20:55	Out	0.204191	d0cf1c9fdcd2e4ac3e0421e8bd5f81ce85a1ed1e7ebc6cb78980e4c0b52b9e4b
t_{30}	2013-05-28 20:57	Out	0.203799	985bd5a528e2992820f5a5a1b64d537b518e29dabd40651662e5fbe09b8caf49
t_{31}	2013-05-28 21:07	In	0.6005	b12e7bb024ab1a98dfe27375eb4b378cbb5e316751cee7faf5cc2c70cd5b738a
t_{32}	2013-05-28 21:13	Out	0.2110955	4de6e9651f3801bfa110dce3e1c3d01c129dcfc87ad098909e508122014fc18f
t_{33}	2013-05-28 21:15	Out	0.21336685	c2bd5ab1a52621684150ad3d4d087c131d9bbbd17d38d0db523da85ab5406bb2
t_{34}	2013-05-28 21:30	Out	0.25707765	e490ad336994f2c570a5d28edc85c80316ed00f4d8cfd0a99a86a5a224ba127a

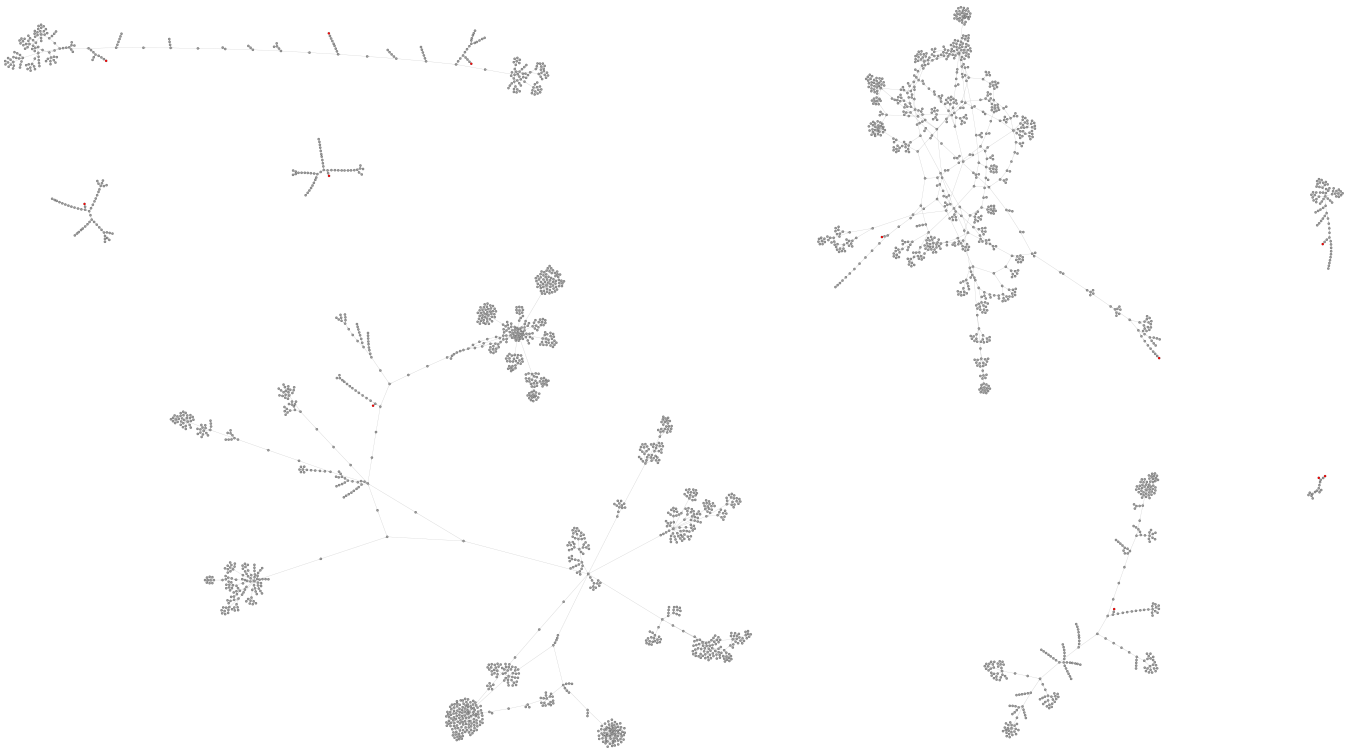

Figure 11: Graphs of the Blockchain.info Send Shared transactions.

again, making it difficult to draw any conclusions on where the bitcoins come from.

3.5 Combinations

All three services still pose the risk, that the operator itself is an attacker or at least cooperates with one. In order to reduce this risk, it would be possible to combine multiple services. However, this comes with a great cost increase. We calculate the cost for using Bitcoin Fog, BitLaundry and Send Shared to obfuscate a transaction. The output O can be calculated by multiplying the input I with the fees of the single services, minus the number of outgoing transactions m of BitLaundry, minus our initial transaction cost. In this case we end up with a total cost of about 5% for using these mixes.

$$O = I * 0.98 * 0.995 * 0.9751 - m * 0.00249 - 0.0005 \quad (5)$$

The real cost are probably even higher, as the risks, that one of the services goes bankrupt (e.g., it gets hacked and

There is an updated version of this paper called "An Inquiry into Money Laundering Tools in the Bitcoin Ecosystem"
available at http://maltemoeser.de/paper/money-laundering.pdf

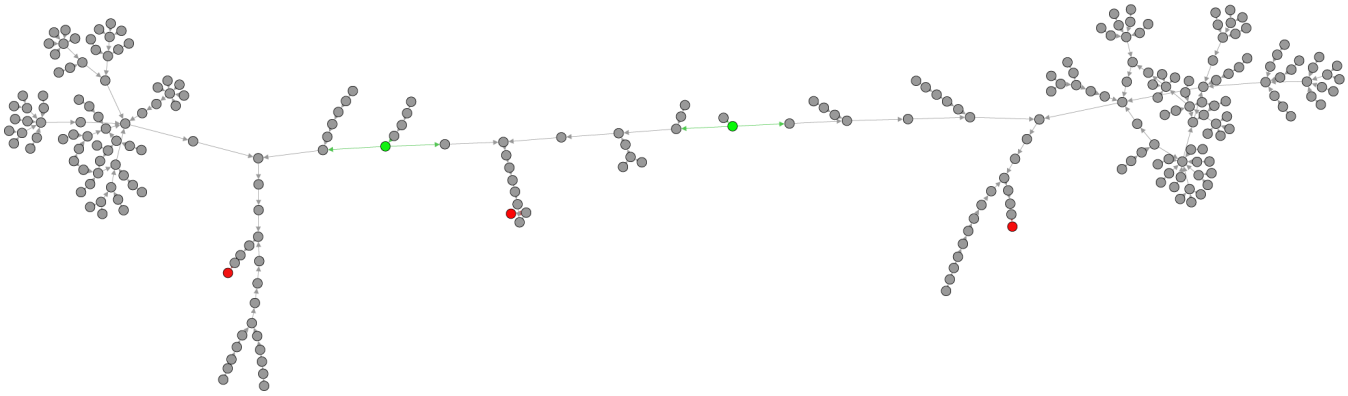

Figure 12: Transactions in the Send Shared graph.

all coins are stolen) or offline (stealing all coins that are in the system), as well as the risk that transactions are not included into the blockchain, have to be calculated as well.

3.6 Conclusion

We can conclude that both Bitcoin Fog and Blockchain.info make it hard for an attacker to relate input and output transaction. In our analysis of Bitcoin Fog we found a clear structure, making it possible to understand how the service works. This might make it easier for an attacker who has additional information available to detect outgoing transactions. We were not able to find any direct connections in the transaction graph of Blockchain.info. In the analysis of the service BitLaundry, we found direct connections in the transaction graph in two of our three experiments, in the last one we directly received half of the coins, we payed in. Thus, BitLaundry cannot be considered to reliably increase anonymity.

4. RELATED WORK

The idea of electronic cash systems is not new. In 1985 Chaum [6] proposed an e-cash system, that allowed anonymous payments using a blind signature scheme. Another idea for electronic currencies are credit networks, e.g. iOwe [13], where every user is able to create digital bonds, which represent his pledge to deliver a certain value or good in the future. These networks heavily rely on trust to prevent double spending and sybill attacks.

There have been several studies on the anonymity of Bitcoin. Ron and Shamir [22] analyze statistical properties of the bitcoin network. Androulaki, Karame, and Roeschlin [1] look at the privacy implications of multi-input transactions and shadow addresses generated by the client for receiving change. They also simulate a local use of Bitcoin and identify 40% of user-profiles based on their behavior. Reid and Harrigan [21] analyze the network of bitcoin users by combining addresses that are inputs of multi-input transactions and therefore must belong to the same sender. They use publicly available data, like forum posts which contain bitcoin addresses, to identify users. Miers et al. [14] propose an e-cash system called *Zerocoin* that is based on Bitcoin and uses zero knowledge proofs to deposit and withdraw special transactions, where input and output cannot be linked together. Ober, Katzenbeisser, and Hamacher [17] analyze structural aspects of the transaction graph and their implications for the anonymity of transactions.

The idea of increasing anonymity by mixing data of multiple users was first presented by Chaum [7]. Probably the most popular mix system is The Onion Router (Tor), which aims at anonymizing applications and users communication on the TCP-layer [9]. Attacks on mix systems are often performed using context or linkability information [12, 23, 27].

5. DISCUSSION AND OUTLOOK

In this paper we presented why Bitcoin is not an anonymous currency, how bitcoin mixes try to increase anonymity by using the concept of a shared wallet and how they differ from traditional mixes. Using two approaches to measure anonymity and the transaction graph to find connections and visualize the structure, we analyzed in which way and how well the three mixing services operate.

A limitation of this work is that we only looked at the transaction graph. Structures in the graph might become more visible when addresses are combined to create a user graph. Regarding the analysis of larger transaction graphs, there is an element of uncertainty, as some transactions in the graph might not belong to the mixing service. Furthermore, due to our limited amount of bitcoins, we were not able to analyze if these mixes can obfuscate large transactions, for example with hundreds of bitcoins.

The big problem of the bitcoin mixing services is the necessity of a central instance, that controls the mix and is able to relate input and output transactions. Using multiple mixes increases the costs, making it too expensive for everyday use, as well as the danger of loosing money to an untrustworthy mix. As we cannot directly apply Chaums mix design to bitcoin, further work could try to build a decentralized mixing service for digital currencies like bitcoin. Another idea would be to have bitcoin purely as a reserve currency and use Chaums tokens as an anonymous payment system [11].

6. ACKNOWLEDGEMENTS

The author would like to thank Raimo Radczewski for the interesting and helpful discussions as well as Dominic Breuker and the anonymous reviewers for their valuable feedback.

There is an updated version of this paper called "An Inquiry into Money Laundering Tools in the Bitcoin Ecosystem" available at http://maltemoeser.de/paper/money-laundering.pdf

7. REFERENCES

[1] E. Androulaki, G. Karame, and M. Roeschlin. Evaluating User Privacy in Bitcoin, 2012.

[2] Anonymity. URL: https://en.bitcoin.it/wiki/Anonymity (visited on 05/23/2013).

[3] Bitcoin Fog. URL: http://bitcoinfog.com/ (visited on 06/22/2013).

[4] Bitcoin Laundry. URL: https://en.bitcoin.it/wiki/Bitcoin_Laundry (visited on 05/21/2013).

[5] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast Unfolding of Communities in Large Networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

[6] D. Chaum. Security Without Identification: Transaction Systems To Make Big Brother Obsolete. *Communications of the acm*, 28(70), 1985.

[7] D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the acm*, 24(2):84–90, Feb. 1981.

[8] C. Diaz, S. Seys, J. Claessens, and B. Preneel. Towards Measuring Anonymity. In R. Dingledine and P. Syverson, editors, *Privacy enhancing technologies*, in Lecture Notes in Computer Science, pp. 54–68. Springer, 2003.

[9] R. Dingledine, N. Mathewson, and P. Syverson. Tor: The second-generation onion router, 2004.

[10] Donate to WikiLeaks. URL: http://shop.wikileaks.org/donate (visited on 05/30/2013).

[11] S. Dörner. Was den Bitcoin-Durchbruch verhindert. 2013. URL: http://www.wallstreetjournal.de/article/SB10001424127887323582904578486772517550396.html (visited on 05/31/2013).

[12] M. Franz, B. Meyer, and A. Pashalidis. Attacking Unlinkability: The Importance of Context. In N. Borisov and P. Golle, editors, *Privacy enhancing technologies*, in Lecture Notes in Computer Science, pp. 1–16. Springer, 2007.

[13] D. Levin, A. Schulman, K. LaCurts, N. Spring, and B. Bhattacharjee. Making Currency Inexpensive with iOwe. *Proceedings of the workshop on the economics of networks, systems, and computation (netecon)*, 2011.

[14] I. Miers, C. Garman, M. Green, and A. D. Rubin. Zerocoin : Anonymous Distributed E-Cash from Bitcoin, 2013.

[15] Mt.Gox. Statement Regarding Account Verifications. 2013. URL: https://mtgox.com/press_release_20130530.html (visited on 05/31/2013).

[16] S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.

[17] M. Ober, S. Katzenbeisser, and K. Hamacher. Structure and Anonymity of the Bitcoin Transaction Graph. *Future internet*, 5(2):237–250, May 2013.

[18] A. Pfitzmann and M. Hansen. A terminology for talking about privacy by data minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management, 2010.

[19] Piuk. What is taint? 2012. URL: https://bitcointalk.org/index.php?topic=92416.msg1018943#msg1018943 (visited on 05/31/2031).

[20] Protocol rules. URL: https://en.bitcoin.it/wiki/Protocol_rules (visited on 05/21/2013).

[21] F. Reid and M. Harrigan. An Analysis of Anonymity in the Bitcoin System. In Y. Altshuler, Y. Elovici, A. Cremers, N. Aharony, and A. Pentland, editors, *Security and privacy in social networks*, pp. 197–223. Springer, 2013.

[22] D. Ron and A. Shamir. Quantitative Analysis of the Full Bitcoin Transaction Graph, 2012.

[23] S. Schiffner and S. Clauß. Using linkability information to attack mix-based anonymity services. In, *Privacy enhancing technologies*, pp. 94–107, 2009.

[24] Send Shared. URL: http://blockchain.info/de/wallet/send-shared (visited on 05/31/2013).

[25] O. Solon. A simple guide to Bitcoin. 2013. URL: http://www.wired.co.uk/news/archive/2013-05/7/bitcoin-101 (visited on 05/23/2013).

[26] Transactions. URL: https://en.bitcoin.it/wiki/Transactions (visited on 05/28/2013).

[27] S. Zhioua. Anonymity attacks on mix systems: a formal analysis. In, *Information hiding*, pp. 133–147, 2011.