Bitcoin Mixing Service with Monero

HingOn Miu
hmiu@andrew.cmu.edu

Background

Contrary to popular perception, Bitcoin is unprecedentedly transparent. All Bitcoin transactions are permanently stored and visible to public. Therefore, every transaction is traceable since the sender address, the receiver address and the transaction amount are publicized. While each Bitcoin address is generated privately by user's wallet, once a transaction is completed with a newly generated address, the new address becomes tainted by the transaction history of the involved sender or receiver address. So, anyone can see the balance of a known Bitcoin address. Since Bitcoin users usually have to reveal their identity to trade services or goods, the users expose the ownership of their Bitcoin address once they use them to receive or send payments. Then, the user's identity is linked to the Bitcoin address, and any outside observer could find out how much Bitcoin the user exactly owns as well as the complete payment history of the user.

This is where Bitcoin mixing service comes in. It mixes the transactions of deposited Bitcoin amount to obscure the backtracking to the true original source of funds. The service ensures the requested Bitcoin amount is securely deposited to the given destination Bitcoin address with no traceable link between the original address and the destination address. In other words, the mixing service maintains the users' anonymity whenever they earn or spend their Bitcoin.

Monero

Monero is a cryptocurrency created in April of 2014 to focus on privacy in transactions by shielding sender and receiver addresses and transacted amounts with cryptography. It is widely considered to be one of the most anonymized and truly fungible coin.

Monero uses three main privacy cryptographic algorithms to ensure privacy in transactions. First of all, ring signatures are used to hide the sender identity. When a transaction occurs, ring signatures allow the sender to randomly select other users' transacted amount to be included in the transaction to obfuscate the true source of funds. The number of other users included in the transaction can be adjusted with the transaction's mixin level. This passive mixing scheme means no one could tell who the actual sender is, not even the receiver.

Stealth addresses are used to hide the receiver identity. They allow the sender to generate one-time random addresses for every transaction on behalf of the receiver so that only the fake address in included in public blockchain while the actual receiver address is never publicized. Finally, Monero's ring confidential transactions allow users to hide transacted amounts. The amounts are encrypted in Monero blockchain so that only the sender and the receiver know the exact amount transferred. These three mechanisms together prevent any outside observer to learn information about the identities of sender and receiver in a transaction.

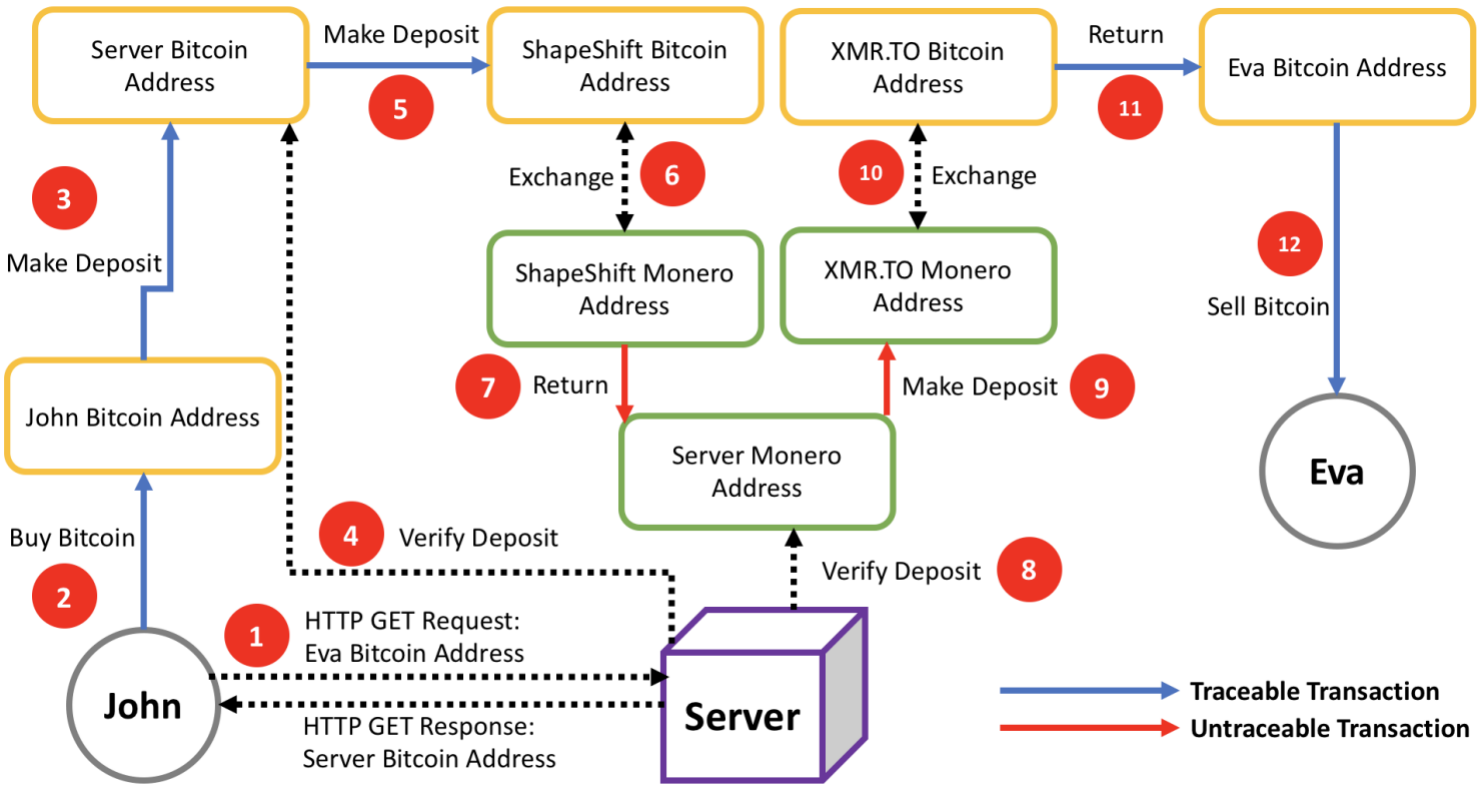
Usage

1. User sends HTTP GET request to server and provides three parameters. *Destination Amount* is the exact Bitcoin amount to be deposited after mixing. *Destination Address* is the Bitcoin address to deposit destination amount. *Refund Address* is the Bitcoin address to return deposited amount if anything goes wrong.

2. Server responds the user with two parameters to inform that the mixing service order is successfully created. *Deposit Amount* is the exact Bitcoin amount for the user to deposit before the mixing service begins. This amount should include the requested destination amount with around 2% mixing service fee. *Deposit Address* is the Bitcoin address provided by server for user to send deposit amount.

3. User should have Bitcoin ready to send deposit amount to deposit address. The service begins immediately after the deposit amount is received.

4. After deposit, user should wait for a couple minutes to check the destination Bitcoin address and verify that the destination amount is successfully deposited.



Implementation

After processing user's GET request, our server first contacts XMR.TO to check how much Monero should be sent to XMR.TO so that the destination Bitcoin amount can be deposited at the destination Bitcoin address. As XMR.TO is an online service that exchanges Monero to Bitcoin, it responds our server with the exact Monero amount to deposit and the Monero address to deposit to.

And then, our server contacts ShapeShift to check how much Bitcoin should be sent to ShapeShift so that the exact Monero requested by XMR.TO is returned by ShapeShift. As ShapeShift is an online service that exchanges Bitcoin to Monero, it responds our server with the exact Bitcoin amount to deposit and the Bitcoin address to deposit to.

Next, our server generates a new Monero address for the intermediary transfer of Monero from ShapeShift to XMR.TO. Also, our server generates a new Bitcoin address for the user to deposit the exact Bitcoin required to complete the mixing service order. The Bitcoin amount and the new Bitcoin address are returned to the user by GET response in JSON.

Upon receiving the full Bitcoin amount from the user, our server sends the Bitcoin amount requested by ShapeShift to begin exchanging Bitcoin to Monero. ShapeShift then returns our server the exact Monero amount requested by XMR.TO. Next, our server sends XMR.TO the Monero amount requested to begin exchanging Monero to Bitcoin. XMR.TO then returns the exact Bitcoin amount deposited at the destination address requested by the user. The mixing service order is thus completed.

Traceability

In the usual case, an outsider attempts to trace the transaction history from John to Eva. Since all Bitcoin transactions are public knowledge, the outsider can track John's Bitcoin transaction to our server Bitcoin address and our server's Bitcoin transaction to ShapeShift Bitcoin address. Since ShapeShift then exchanges Bitcoin to Monero internally, the actual tracing should swap from ShapeShift Bitcoin address to ShapeShift Monero address. However, the outsider cannot possibly know about the swap from observing public Bitcoin transaction history, so the tracing would move on to other ShapeShift Bitcoin addresses or some ShapeShift client's Bitcoin address. This would never lead to Eva.

A similar scenario happens in backtracking. Since all Bitcoin transactions are public knowledge, the outsider can backtrack Eva's Bitcoin transaction to XMR.TO Bitcoin address. Since XMR.TO exchanges Monero to Bitcoin internally, the actual backtracking should swap from XMR.TO Bitcoin address to XMR.TO Monero address. However, the outsider cannot possibly know about the swap from observing public Bitcoin transaction history, so the backtracking would move on to other XMR.TO Bitcoin addresses or some XMR.TO client's Bitcoin address. This would never lead back to John.

XMR.TO Breach

Now, let's consider some extreme cases. Since the destination address belongs to Eva and that all Bitcoin transactions are public knowledge, an outsider can backtrack the transaction history of Eva Bitcoin address to XMR.TO Bitcoin address. So, the outsider finds out XMR.TO is involved. For the sake of argument, say XMR.TO could be compromised by hackers or would willingly cooperate with law enforcements such that all exchanges performed by XMR.TO are visible to the outsider. The outsider now gains all of XMR.TO's exchange logs and identifies that specific XMR.TO Bitcoin address was used to exchange Monero to Bitcoin. Therefore, the outsider learns the XMR.TO Monero address is involved.

Recall the design of Monero's ring signatures. Its cryptographical design is actually a default transaction mixing procedure that other random senders are included in a transaction to obfuscate the true source of fund, and so everyone, even the receiver, cannot identify the real sender. Thus, the outsider's backtracking from Eva Bitcoin address ends here, after learning the XMR.TO Monero address.

ShapeShift Breach

Say the outsider then decides to trace from John's end to see if a possible connection can be found that links to the XMR.TO Monero address. Since all Bitcoin transactions are public knowledge, the outsider traces the transaction from John Bitcoin address to our server Bitcoin address. Our server Bitcoin address was generated for one-time use to receive Bitcoin deposit from John to initiate the mixing service.

Now that the outsider learns our server Bitcoin address, the transaction to ShapeShift Bitcoin address is also exposed. So, the outsider finds out ShapeShift is involved. For the sake of argument, say ShapeShift could also be compromised by

hackers or would willingly cooperate with law enforcements such that all exchanges performed by ShapeShift are visible to the outsider. The outsider now gains all of ShapeShift's exchange logs and identifies that specific ShapeShift Bitcoin address was used to exchange Bitcoin to Monero. Therefore, the outsider learns the ShapeShift Monero address is involved.

Recall the design of Monero's stealth addresses. The sender generates one-time addresses for every transaction on behalf of the receiver so that the actual Monero address of the receiver is never shown in public record. Therefore, any outsider cannot determine where a payment is sent even when sender's identity is exposed. In this design, only the sender knows the actual Monero address of the receiver, which links to the receiver's true identity. Normally, through observing public blockchain, the outsider cannot determine the Monero payment is actually sent to our server even when the outsider knows the ShapeShift Monero address. However, given that ShapeShift, the sender, is compromised, the outsider could then discover the actual receiver address from ShapeShift. Now that the outsider learns our server Monero address, the outsider concludes the Monero amount was received by our server.

Server Breach

Now, the outsider would have to compromise our server to attempt to find out if our server Monero address made payment to that XMR.TO Monero address in order to make the ultimate connection between John and Eva. For the sake of argument, say our server is entirely compromised so that all information stored in our database is leaked. First of all, the private view key of our server's Monero account can only be used to display incoming transactions. So, even if the outsider has control over the private view key, it can only prove the transaction from ShapeShift, but not the transaction to XMR.TO.

The only Monero mechanism to actively prove an outgoing transaction is that the sender provides a per-transaction key for the receiver to verify the sender is the true source of fund. However, generating these per-transaction keys is disabled by default, and must be enabled before the actual transaction. Since our server makes sure to disable this feature to not store any per-transaction key, the outsider could not possibly retrieve the per-transaction key of that XMR.TO payment to prove the transaction, even when our server is completely compromised.

Another pitfall is if our server keeps logs of all transactions and addresses, so the outsider can find the XMR.TO transaction details with that XMR.TO Monero address or even the mixing service order details of John's request with Eva Bitcoin address. Thus, our server purges the record of each completed order in the database an hour after creation. Even if the outsider gains control over the entire database of our server, no information of past orders could be found.

Last but not least, the server Monero wallet client locally stores all outgoing and ingoing transactions of the wallet. Therefore, at the end of each day, our server destroys the wallet, along with its login information, private view and spend keys and seed, then our server clears the wallet cache and creates a new wallet. Without the seed, the removed wallet is irrecoverable and permanently lost. So, there would never be any trace that links a past Monero payment from our server to XMR.TO, which means the outsider's tracking from John Bitcoin address ends here, after learning our server Monero address.

Considering XMR.TO, ShapeShift and our server are all compromised by the same outsider is a very unlikely scenario,

and even then, the connection between John and Eva cannot be proved. So, this shows our mixing service privacy design is highly reliable.

Conclusion

This power of deleting Monero wallet to permanently break traceability of a Monero transaction trail is exactly why our server chooses to have an intermediate Monero transaction relay between ShapeShift and XMR.TO, instead of having ShapeShift to send Monero to XMR.TO directly. If ShapeShift transacts with XMR.TO directly, ShapeShift Monero wallet would then have a record of XMR.TO Monero address and there is nothing our server can do to remove this piece of information from ShapeShift wallet. Unlike Monero, Bitcoin is extremely transparent since details of all confirmed transactions are publicly visible on blockchain. So, destroying a Bitcoin wallet permanently would hide nothing and make no difference when an outsider decides to pry its transaction history.

It is now apparent that Monero transaction scheme prevents any third party to learn the true identities of sender and receiver from observing the public blockchain. Even on the same transaction trail, any previous sender would not know where the payment really goes from the current sender. Similarly, any following receiver would not know any true source of payment along the transaction trail. With Monero's ring signatures and stealth addresses, not even the receiver of payment knows true identity of its sender.

Evidently, the only weak link of a Monero transaction trail is potential leak from the sender of a transaction, since only the sender knows the true identity of the receiver and could expose real transaction information. This is unavoidable as the sender must know exactly who to pay before making the payment. In theory, to prove a connection between a source and a destination, malicious attacker or law enforcement could first start with a public Monero address that is known to be linked to true original sender and hack the sender wallet (e.g. pry wallet seed) to find the receiver address, and then hack the receiver wallet to find the next receiver address, and so on, until the destination receiver address is identified.

Backtracking from destination is not feasible since ring signatures of each Monero transaction prevent it. Here we assume no sender on the transaction trail would pass per-transaction key to each receiver to allow the receiver to identify the sender since this would void the whole mechanism of ring signatures, the sole purpose of using Monero. So, forward tracking from true source would be the only feasible way to prove connection between an original source and a destination of a Monero transaction trail.

Hence, it is valid to claim that as long as one of the intermediate senders on Monero transaction trail in between the true original source and the destination securely remove the Monero wallet and any means to recover it, the traceability from the original source to the destination is then permanently broken. And so, this wallet destruction must be carefully implemented and automated by our server. Utilizing Monero privacy schemes and securing data destruction, our server exchanges Bitcoin to Monero, mixes Monero and exchanges Monero back to Bitcoin to break traceability of Bitcoin. Also, theoretically, our server could mix any other cryptocurrency and break traceability with the same design.

Sources

https://bitcoin.org/
https://getmonero.org/
https://monero.org/
https://xmr.to/
https://monero.how/
https://shapeshift.io/