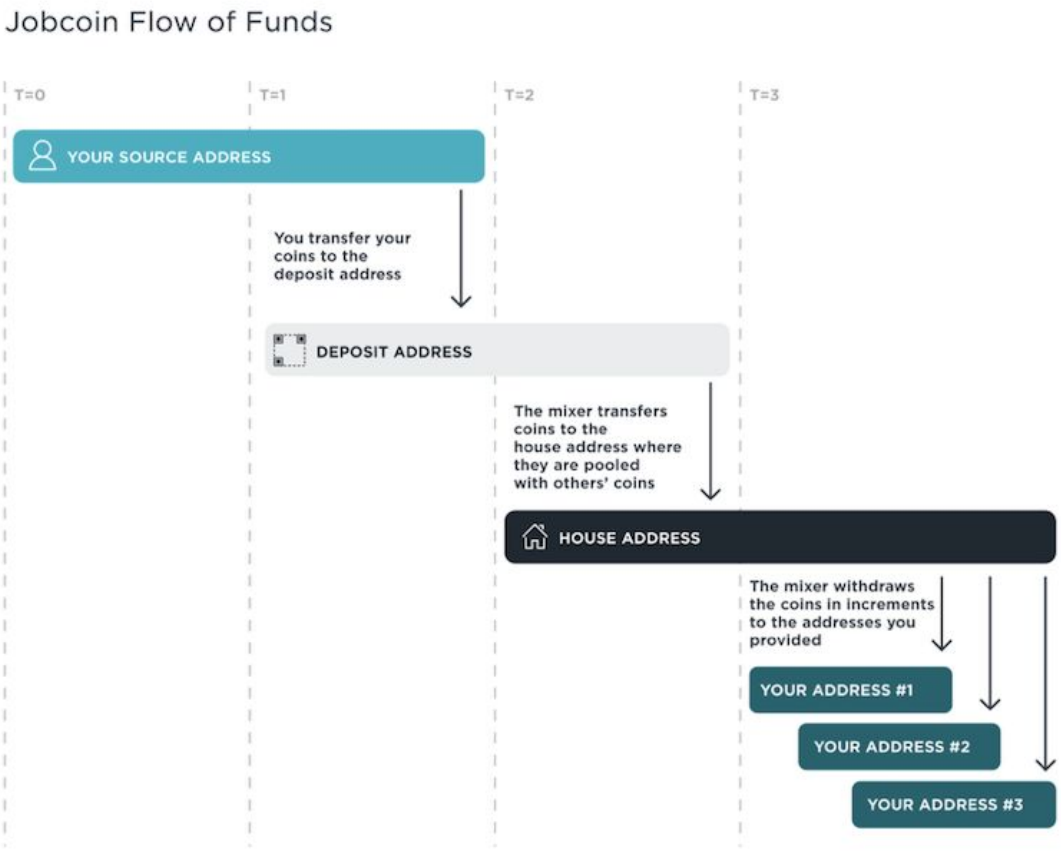
Coin Mixer Challenge

Despite some media reports, Bitcoin is not an anonymous protocol. Instead, it's often referred to as a pseudonymous system. All transactions to or from any Bitcoin address are publicly available, so Bitcoin's "anonymity" hinges on not knowing which addresses belong to which people. But because Bitcoin addresses are so trivial to create (it's essentially just a key pair), you can help ensure your anonymity by using a bunch of addresses instead of just one.

A Bitcoin mixer is one way to maintain your privacy on the Bitcoin network. The diagram below provides a helpful demonstration of how a mixer works:



1. You provide a list of new, unused addresses that you own to the mixer;
2. The mixer provides you with a new deposit address that it owns;
3. You transfer your bitcoins to that address;
4. The mixer will detect your transfer by watching or polling the P2P Bitcoin network;
5. The mixer will transfer your bitcoin from the deposit address into a big “house account” along with all the other bitcoin currently being mixed; and
6. Then, over some time the mixer will use the house account to dole out your bitcoin in smaller increments to the withdrawal addresses that you provided, possibly after deducting a fee.

There are a number of reasons to use a Bitcoin mixer. For instance, if your salary gets paid to the same Bitcoin address every two weeks, and if you buy your morning coffee using that address, it would be fairly easy for your barista to look up your previous transactions and figure out how much money you make. Using a mixer is one of the many ways to hide that transaction flow.

Bitcoin is a very difficult protocol to work with, especially for a newcomer to cryptocurrencies, so this challenge is to create a mixer for a new, much simpler virtual currency, *Jobcoin*. Each Jobcoin has an “address” that is just an arbitrary string, and there’s no mining or transaction signing - anyone can create units of Jobcoin out of thin air and/or send them between any two addresses.

INSTRUCTIONS

Please create a Jobcoin mixer, analogous to the Bitcoin mixer described above. You may collect a fee for your mixing service if you wish. We recommend doing the coding challenge in one of these languages: Scala, Java 8, JavaScript, Python (2 or 3), or C++. If you’re not comfortable coding in any of those languages, please let us know what you’re planning on using. Then send us your source code as a .zip file, git bundle, or link to an online source code repository.

A set of starter templates in Scala, Javascript, and Python are available at https://github.com/gemini/jobcoin-boilerplate.

Please write the test in the language you feel the most comfortable in so we can see your best work!

Mixers can be incredibly complicated, and people have spent years working on them. Please don’t spend time making an overly complicated solution — but be prepared to discuss what privacy vulnerabilities might exist in your mixer as written and how you could mitigate them.