EtherShipLite

EtherShip is a longshot idea to create a decentralized, anonymous postal service using ethereum smart contracts. EtherShipLite is a non-anonymous, proof-of-concept version of EtherShip, as the challenges presented by anonymizing data in Ethereum smart contracts is worthy of its own separate discussion.

Nodes

A moving node is a node which can move. A moving node has a latitude and a longitude (home) location, and a radius which represents how far the node is willing to go for a package delivery or handoff. A moving node should register for a larger radius than their real radius in order to maximize their driving options.

A stationary node is a node which doesn't drive. It has a home location but no radius.

Shipping a package

1. Package sender calls a `ship` function on the smart contract. The sender must already be a registered node, either driving, or stationary. They specify what node they are shipping to. They also specify when they want their package to get picked up, and how much they want to deposit for insurance.
    1.1. This would have to be anonymized from people looking at the blockchain in the future, including miners.
2. The smart contract generates a shipment `ticket` object with the origin and destination nodes, time, and insurance.
3. Next, the smart contract needs to figure out different possible paths from the sender to the destination. This is tricky.
4. Once the smart contract has a list of different paths:
    4.1. Smart contract loops through all of the different paths and creates an `offer` object for each of first nodes in each path. This offer tells the node:
        - Where they need to go to pick it up
        - How much ether they need to deposit (stake) to accept the offer
        - What time they have to meet up with the previous node in order to pick up the package
    4.2. Some of these nodes will accept their offers, some of them will not. When the node accepts the offer, they set what time they want to meet to hand off the package.
    4.3. For each node that accepts their offer, the next nodes in all possible paths following that node are sent offers. Similarly, these offers tell them:
        - Where they need to go to accept the package (location of the previous node that accepted)

- How much ether they need to stake for the insurance deposit
- What time they have to meet up with the previous node in order to pick up the package.

4.4. This process is repeated until the receiving node accepts. EtherShip requires the receiving node to consent, unlike traditional snail mail which allows people to just send random unwanted junk mail to recipients.

4.5. At this point, there may be no possible routes from the sender to the receiver, or there could be at least one. All possible paths across willing nodes from the sender to the receiver now “bid” on the jobs.

4.6. The original sender is presented with a list of options. Each option will have the price, and the date and time of delivery. The sender will choose the best option for them. They may choose a certain path option because of its speed, or price. It’s up to them.

4.7. Once the sender chooses the path, the shipment is live. The sender then sends his insurance deposit to the smart contract.

4.8. The next node in the chain sends his insurance deposit to the smart contract.

4.9. The sender, and first node meet at the agreed upon time, and they both must verify that the hand-off was successful.

4.10. Once both parties verify that the hand-off was successful, the second node now being in possession of the package, repeats the process with the third node.

4.11. After the 2nd and 3rd nodes verify the handoff, the 2nd node receives his payment and deposit back.

5.