Any user may:

- request verification of a public key associated with her address, by broadcasting her `[hash(address | optional appended string) -> public key]` tuple to the verification pending queue

A verified user may:

- add a new public key by creating a `[hash(address | optional appended string) -> public key]` mapping
- revoke any public key associated with their address
- change any public key associated with their address

A validator may:

- compete with other validators for the right to write a block and send a secret message to the addresses on the verification pending queue, and validate the signed responses of the previous block's verifications.

Anybody may:

- look up the public key for a given address hash (or address hash || string concatenation) in the verified user database.

2.2 Aggregating Reputation Signals through Encrypted EigenTrust

Once there exists a decentralized mapping of phone numbers to public keys, it can be used to bootstrap a reputation system that helps users determine the trustworthiness of any new users they may transact with.

A person's cell phone contact list is a rough first-order proxy for a list of people in whom she has a certain level of trust. One can imagine refining this trust proxy through explicit signals (for example, a user may rate people in her contact list in an application-specific manner, or attest to whether a contact in their address book is a person or not), and implicit signals (for example, if a user makes a payment to somebody in her contact list). These signals can be maintained locally, on the user's cell phone, without sharing them with anybody else.

Such address-book based trust signals define a trust network that is both logically decentralized and functionally decentralized. No single entity stores or has visibility into the entire trust network; each user simply knows the people whom they trust, and the level to which they trust them. We describe below how to compute sybil-resistant, privacy-preserving aggregate reputation scores given this decentralized trust network.

2.2.1 EigenTrust

EigenTrust [14] is a decentralized algorithm for computing global reputation scores, given pairwise local trust scores. The key intuition behind EigenTrust is that a person's reputation score can be defined as the number of people who trust that person, weighted by their reputation scores. This recursive computation converges for all nodes to the principal eigenvector $\vec{t}$ of the trust matrix T, where T_{ij} is number between 0 and 1, and whose magnitude is proportional to the relative level that node i trusts node j[3].

In EigenTrust, the principal eigenvector of T is computed using a distributed variant of the Power Method [20]. In the context of a social payments network, it would proceed as follows: The trust network T_{ij} would be some variant of the payment network, where T_{ij} would be nonzero if node i has paid node j, and node j is in the address book of node i. Each node stores their own current t_i, and has access to the values of T_{ij} in row i and column j (the people with whom the node has interacted). The principle eigenvector $\vec{t}$ would then be computed in an iterative fashion as follows. At

[3] An alternative way to frame the problem is to compute the stationary distribution of the ergodic Markov chain described by the trust network.