Crypto-Book:
An Architecture for Privacy Preserving Online Identities

John Maheswaran,
David Wolinsky, Bryan Ford

The Problem

- Users increasingly use cross-site authentication schemes
  - OAuth, OpenID
  - *Log in with Facebook/Twitter/Google+*
- Users rely on social networking sites for their online digital identities
- However, use of these identities brings privacy/tracking risks
  - Cross-site tracking, browsing history, actions across different sites

OAuth: Cross-site authentication

1. User clicks on Website A to log in using OAuth
2. Website A redirects user to Facebook.
3. User logs into Facebook.
4. User gives permission for Website A to access their Facebook data
5. Facebook generates a temporary OAuth token
6. Facebook redirects user back to Website A
7. Website A can now use the OAuth token to query Facebook for user data

The Problem

- Users would like better privacy protection
- Need to balance the following:
  - Supporting **free speech**, fighting censorship, oppression and allowing individuals to freely express their opinion
  - Improving the **quality of public discourse**. By allowing people to fully hide behind an anonymous veil, they often say or do things they might not otherwise do.

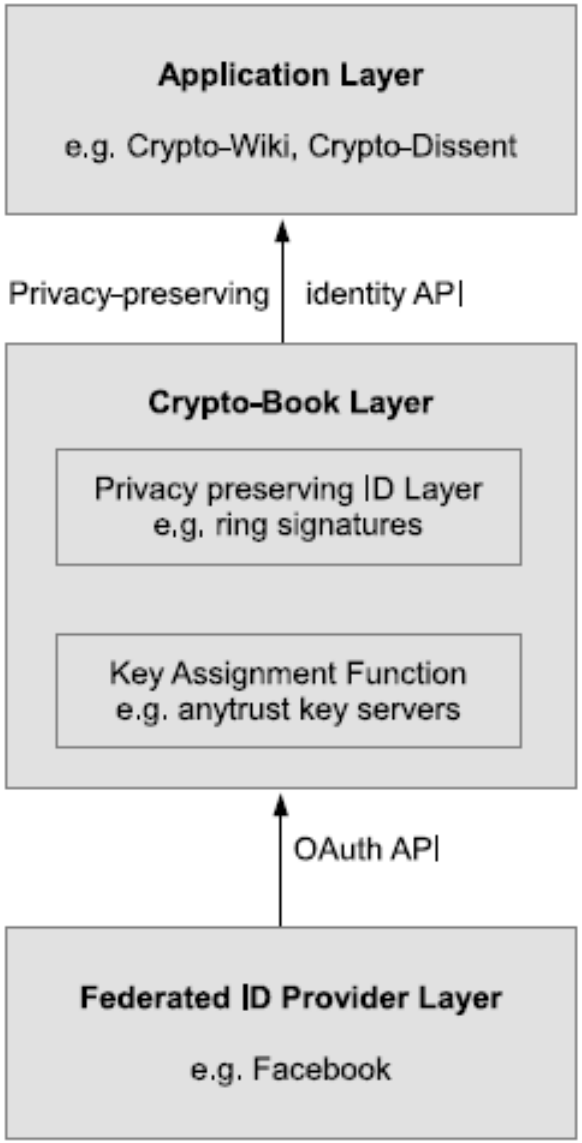
The Problem

- Solution needs to provide both
  - Privacy
  - Accountability
- If we had absolute anonymity with no accountability, system would be open to abuse
  - e.g. anonymous edits on Wikipedia often lead to vandalism

Our Solution: Crypto-Book

- extension to existing digital identity infrastructures
- offers privacy-preserving, digital identities
- adds privacy-preserving crypto layer atop existing social network identities
- *anytrust* cloud of third-party key servers
- turn social network identities into key pairs

The Crypto-Book Stack

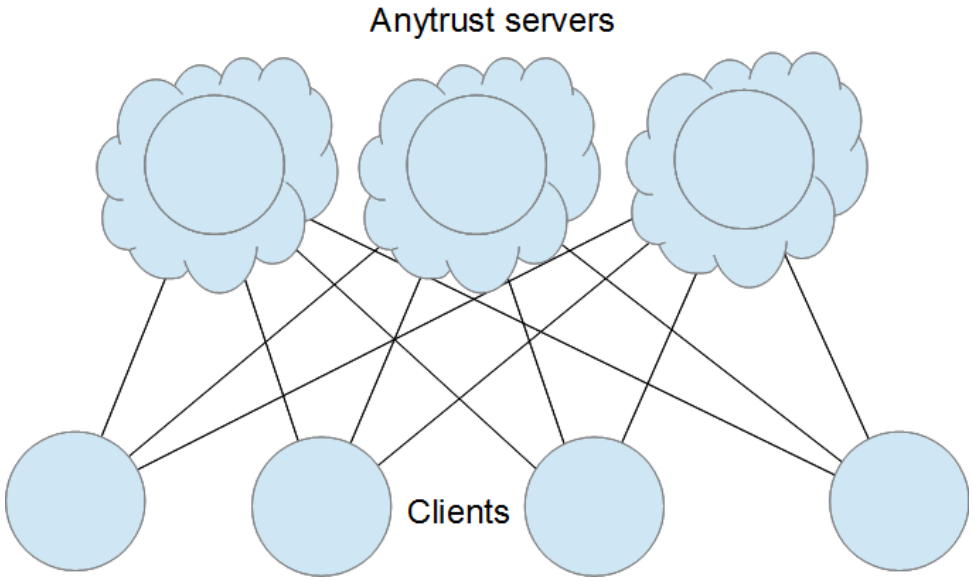


Key Assignment

- In order to use privacy preserving cryptographic technologies, have to assign public/private keypairs to users
- Use *anytrust* cloud of key servers that *splits trust* across them

Anytrust key servers

- An anytrust cloud is:
  - a decentralized client/server network model
  - trust there is *at least one* honest server
- We use an anytrust cloud of key servers
  - assigns key pairs to each social network user

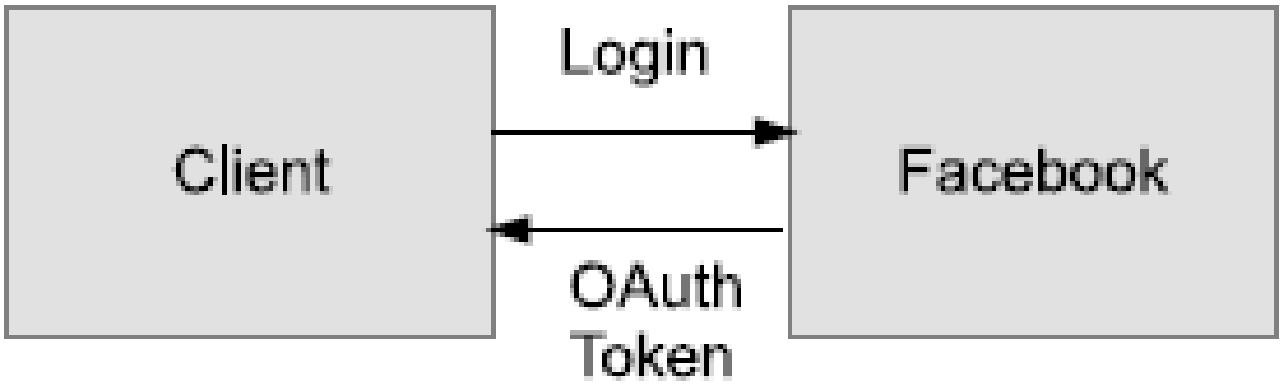


Key assignment

- User authenticates with social network
  - e.g. Log in with Facebook
- User obtains OAuth token which is sent to Crypto-Book key servers
- Each server generates a private key for user
- Servers send private keys back to user
- User combines private keys to get composite private key
  - So long as one server is honest, no server knows the user's composite private key

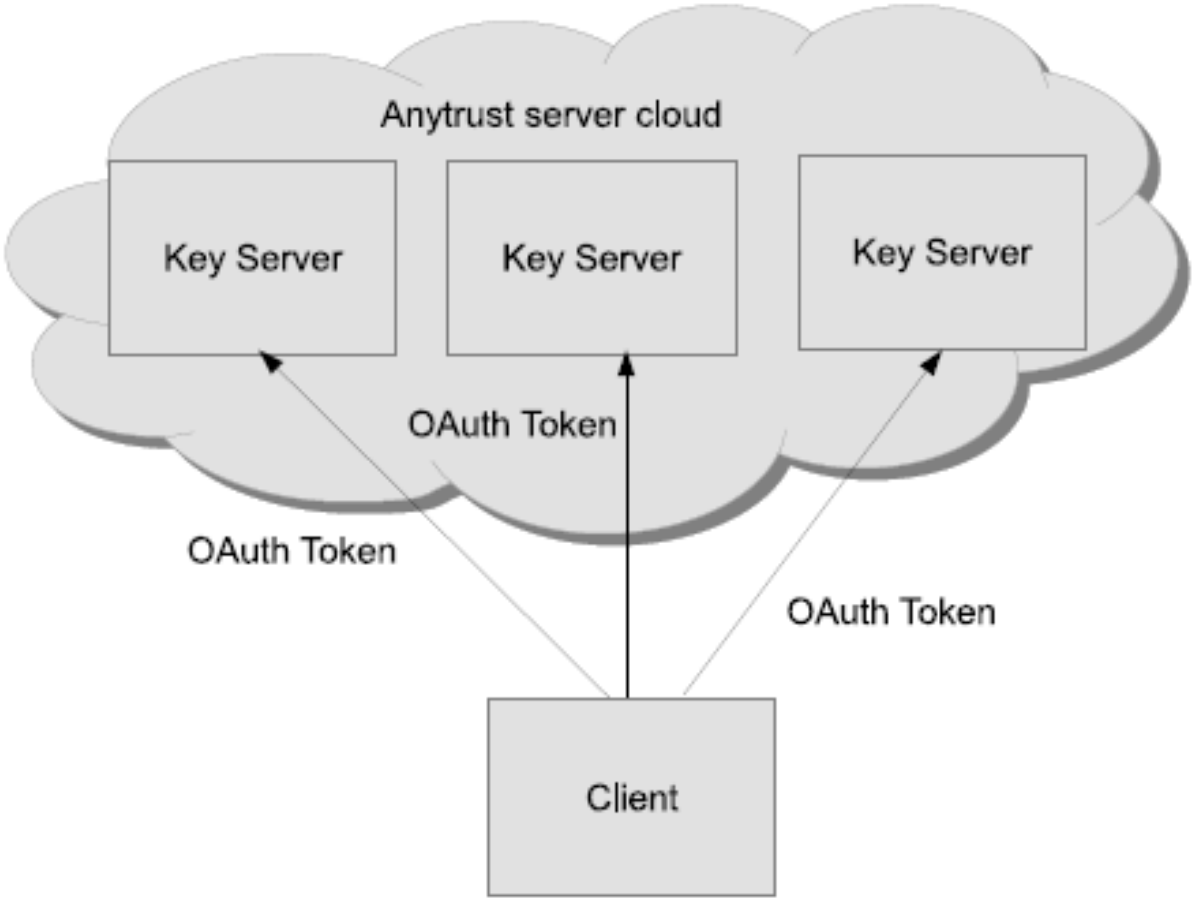
Key assignment

- User first authenticates with their social networking provider



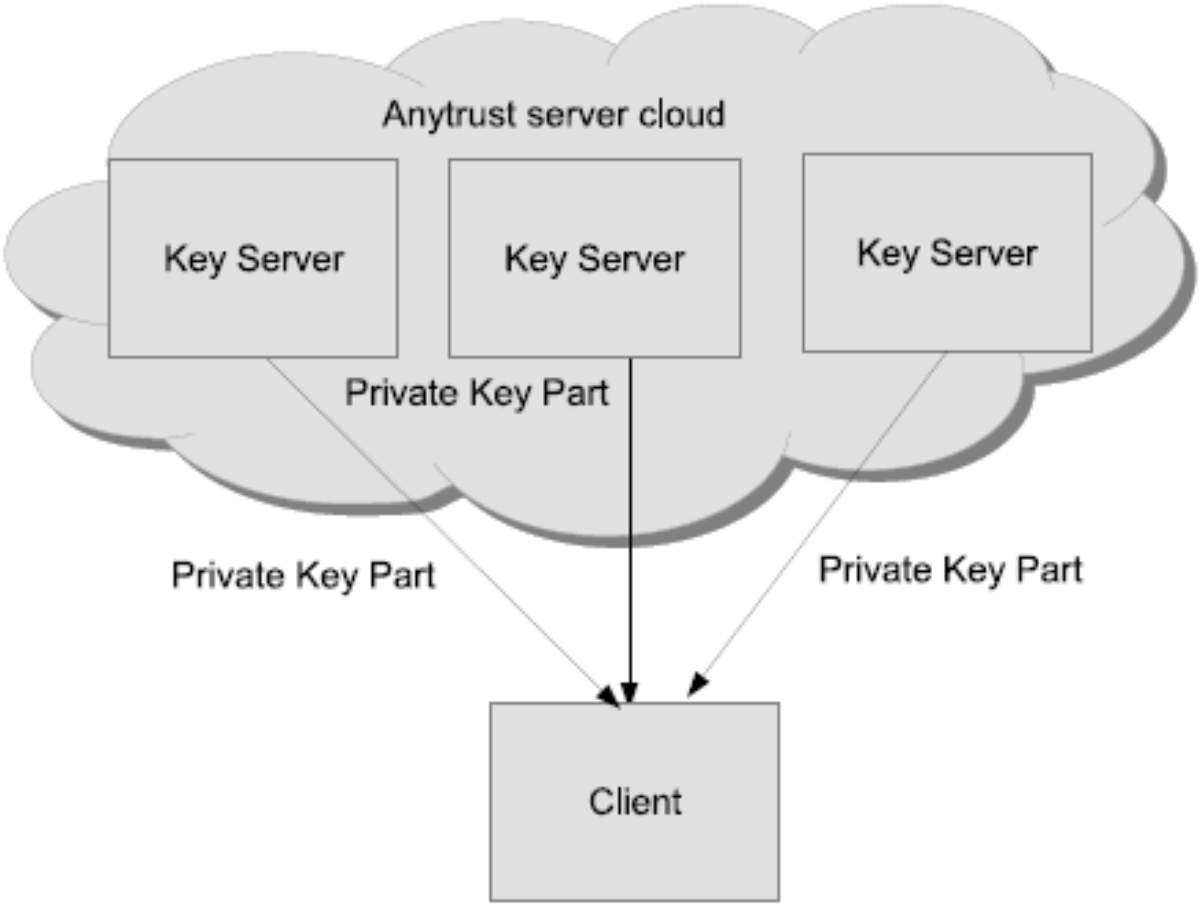
Key assignment

- Client sends OAuth token to key servers



Key assignment

- Key servers send private key parts to client

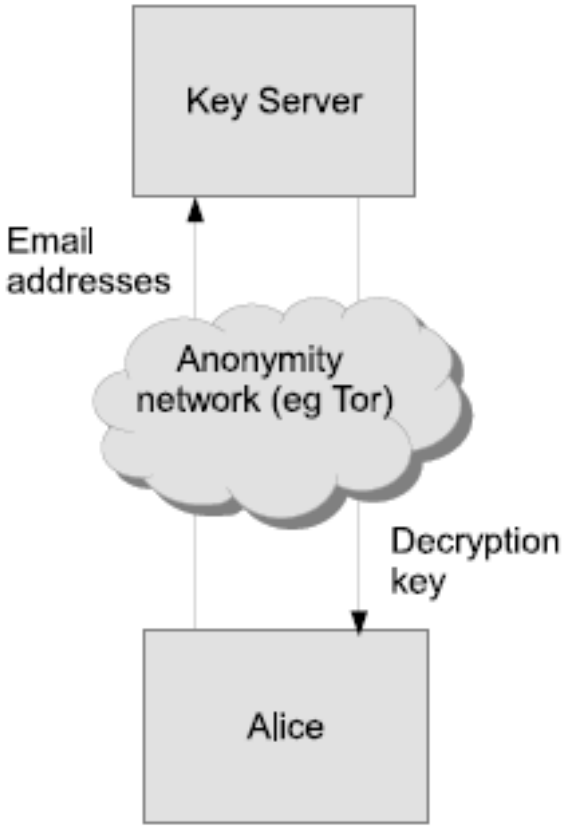


Key assignment

- Public keys can be collected from key servers in similar manner
- Client can now user keys with anonymizing cryptographic techniques
  - Linkable ring signatures (LRS), ring signatures

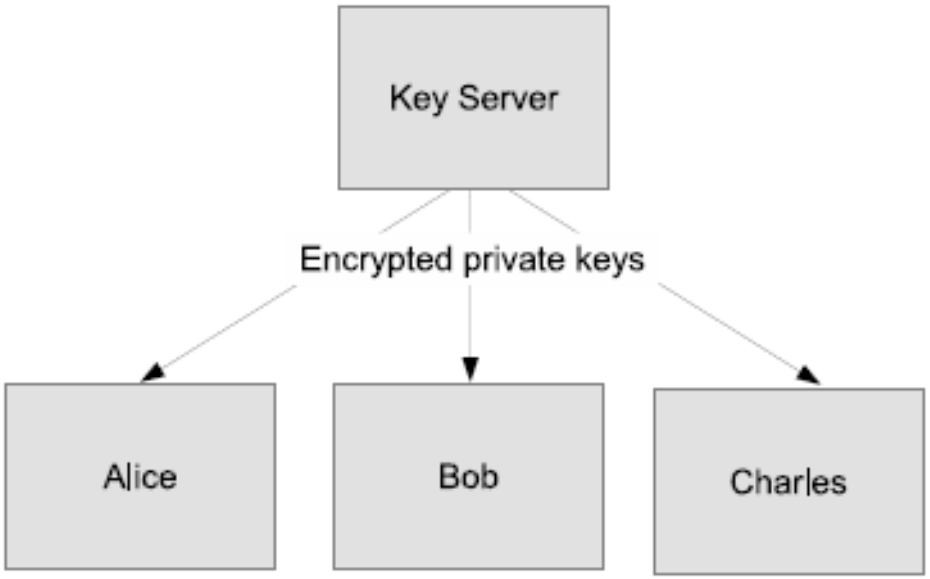
Anonymous key distribution

- Alice anonymously requests her key



Anonymous key distribution

- Messages (e.g. emails) sent with encrypted private key attached
- Alice can decrypt her private key



Privacy Preserving Crypto Layer

- Client generates linkable ring signature (LRS) using their private key and public keys of other Facebook users
- LRS can be verified by third party site as being generated by one member of a group
  - But cannot identify which member
- Preserves privacy

Privacy Preserving Crypto Layer

- LRS has *linkage tag*
  - If a client generates two LRSs, they will have the same linkage tag
  - Means LRSs can be linked across time
- Linkage tag provides *accountability*
  - 1-to-1 mapping between Facebook users and anonymized identities

Anonymous key distribution

- Allows user, Alice, to pick up her private key without server knowing who requested their key
- Alice anonymously requests her key via Tor
  - Provides anonymity set
- Server distributes all keys in anonymity set, encrypted with symmetric key
- Only Alice can decrypt her key

Overall system properties

Goal	Strategy
Usable	Build on top of existing social networks
Dishonest key server resistant	Use anytrust cloud to *split trust* across servers
Privacy protecting	LRS cryptographically anonymizes users
Abuse resistant	LRS ensures 1-to-1 mapping between users and anon IDs
Modular	Could easily use alternative: • key assignment (instead of anytrust cloud) • anonymizing crypto (instead of LRS)