

Private Anonymous Messaging With Friends!

Ruchith Fernando

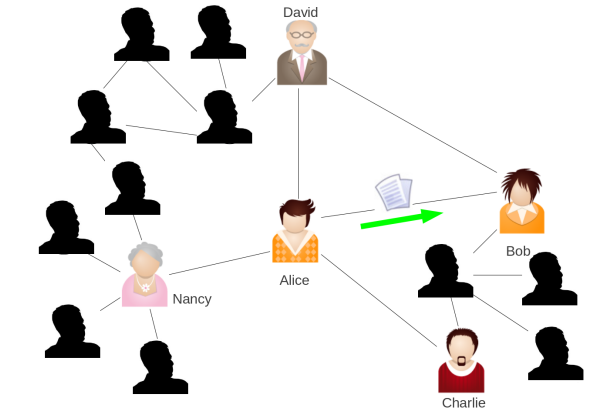
Purdue University

April 26, 2011

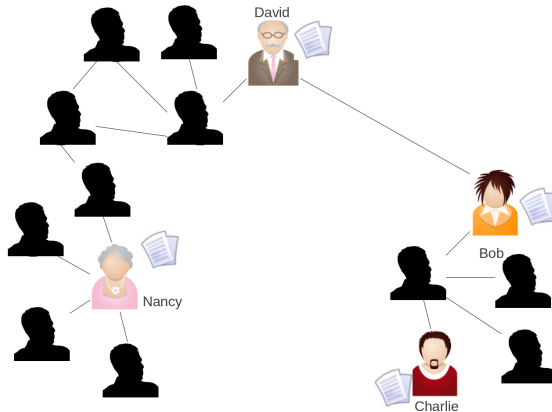
Problem

- A user wants to send a message to all his current contacts
- Even if they are **offline**!
- **Only** trusts his/her immediate contacts
- A contact can re-distribute messages on requests

Problem



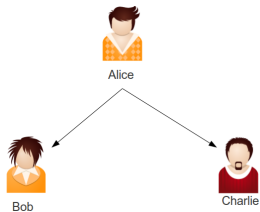
Problem



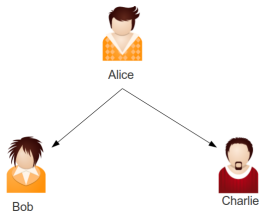
Proposed Solution

- Modify HIBE (Hierarchical Identity Based Encryption with Constant Size Ciphertext, Boneh et.al)
- Each contact is issued a private key (only private channel for key exchange)
- Contacts generate **anonymous** public keys using their private keys
- Broadcast update request to be processed by other contacts
- Re-key mechanism with **public** data (no private channel requirement)

Hierarchical Identity Based Encryption



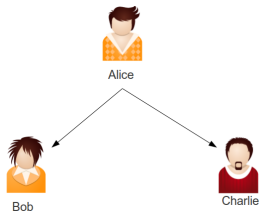
Hierarchical Identity Based Encryption



- Identities:

- Alice: I_1
- Bob: I_1, I_{2_1}
- Charlie: I_1, I_{2_2}

Hierarchical Identity Based Encryption

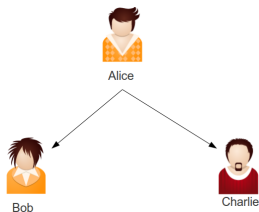


- Identities:

- Alice: I_1
- Bob: I_1, I_{2_1}
- Charlie: I_1, I_{2_2}

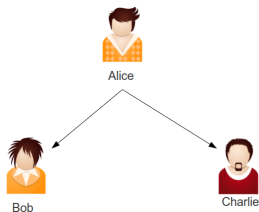
- $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_1, |\mathbb{G}| = p$

Hierarchical Identity Based Encryption



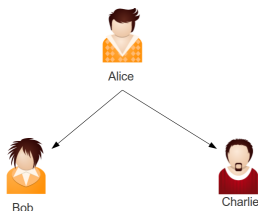
- Identities:
 - Alice: I_1
 - Bob: I_1, I_{2_1}
 - Charlie: I_1, I_{2_2}
- $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_1, |\mathbb{G}| = p$
- $params = (g, g_1, g_2, g_3, h_1, h_2), g_1 = g^\alpha, \alpha \in \mathbb{Z}_p$

Hierarchical Identity Based Encryption



- Identities:
 - Alice: I_1
 - Bob: I_1, I_{2_1}
 - Charlie: I_1, I_{2_2}
- $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_1, |\mathbb{G}| = p$
- $params = (g, g_1, g_2, g_3, h_1, h_2), g_1 = g^\alpha, \alpha \in \mathbb{Z}_p$
- $mk = g_2^\alpha$

Hierarchical Identity Based Encryption



- Alice :
 - $K_{priv_alice} = KeyGen(I_1, params, mk)$
 - $K_{pub_alice} = I_1$
- Bob :
 - $K_{priv_bob} = KeyGen(I_1, I_2, params, K_{priv_alice})$
 - $K_{pub_bob} = I_1, I_{21}$
- To encrypt for Bob
 $CT = Encrypt(msg, I_1, I_{21}, params)$

Changes to HIBE

- Update $Encrypt()$ to work with $h_1^{l_1} h_2^{l_{2_1}} = ID$
- To encrypt for Bob
 $CT = Encrypt'(msg, ID, params)$
- On re-key update α and only generate minimum public data for existing contacts.

Usage : Encryption

- First level identity (I_{r_1}) and private key to a contact
- A contact issue him/herself a second level identity with a random I_{r_2}
- Broadcasts a request for data ($\langle user, ID_r \rangle$) where $ID_r = h_1^{I_{r_1}} h_2^{I_{r_2}}$
- Any other contact of the *user* can respond to the request, by encrypting with $params_{user}$:
$$CT = \text{Encrypt}'(msg, ID_r, params_{user})$$

- If the *user* removes a contact
- Re-key : Parameters
 - Generate a new master key : $\alpha' \in \mathbb{Z}_p, g_2^{\alpha'}$
 - $params_{user}$ update : Only $g_1 = g^{\alpha'}$
 - Current contacts' private keys
- Re-key : Contact private key
 - $K_{priv} = (g_2^{\alpha'} \cdot (h_1^{l_1} \cdot g_3)^r, g^r, h_2^r, h_3^r) = (C_1, C_2, C_3)$
 - Only C_1 need to be published along with l_1
 - Indexed by blinded $ID_{contact}$

Implementation

- <http://code.google.com/p/anon-encrypt/>
- Using Java Pairing Based Cryptography Library (JPBC)¹
- Implemented as a library
- Demo application

¹<http://gas.dia.unisa.it/projects/jpbc/>

DEMO!