A Social Messaging System for GNUnet

Gabor Toth

July 3, 2013

Design goals

A social messaging system, which is

- scalable
- extensible
- end-to-end encrypted

Federated systems

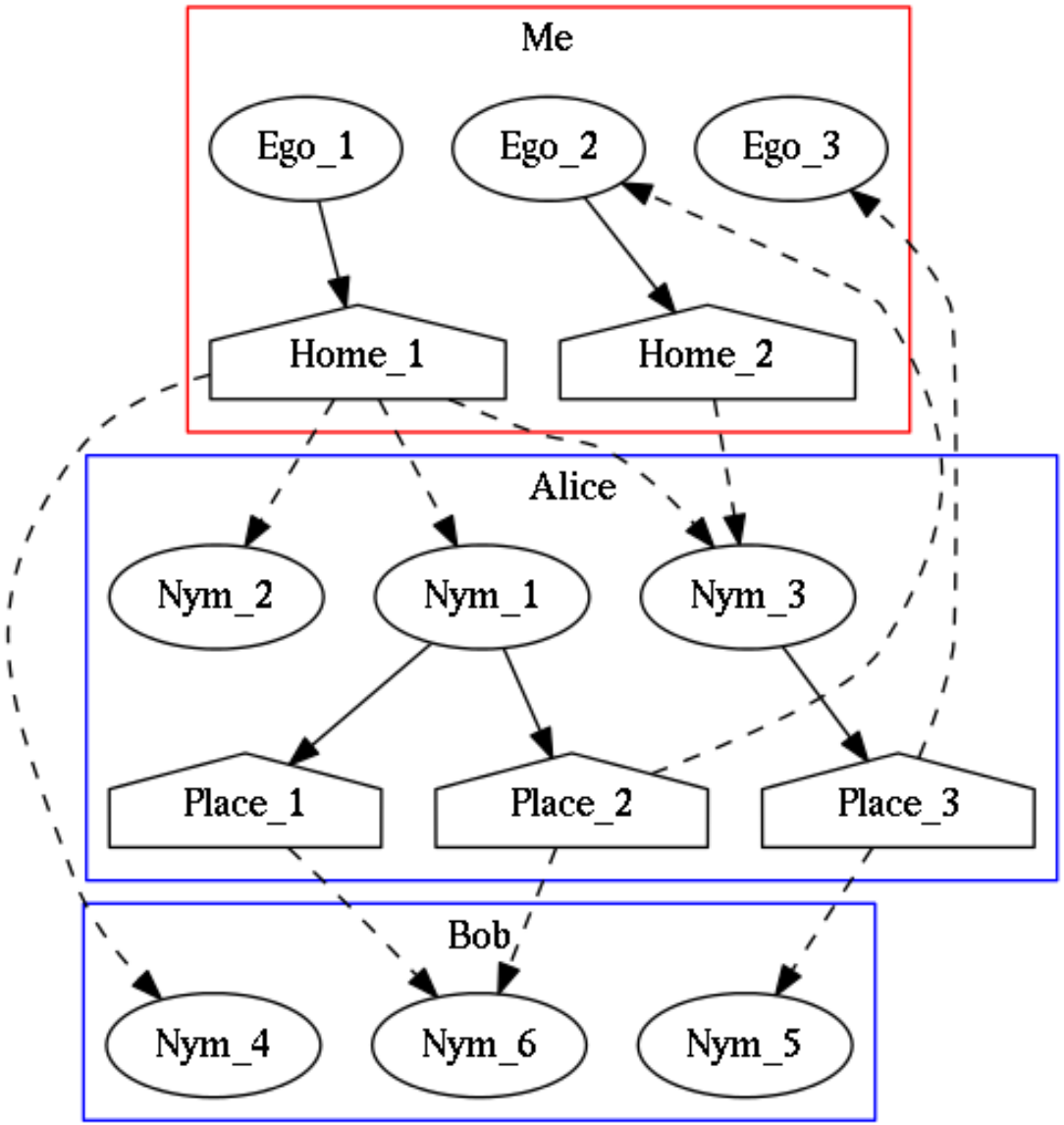
- e.g. XMPP, OStatus
- they only provide link-level encryption
- servers see all communication
- large providers see much of the traffic

GNUnet

- GNU's Framework for Secure Peer-to-Peer Networking
- encrypted communication between peers
- GADS: GNU's Alternative Domain System, offers PKI

Social network model

- Users can have multiple pseudonyms
- Each hosting multiple places, where guests can enter


Me
Ego_1
Ego_2
Ego_3
Home_1
Home_2
Alice
Nym_2
Nym_1
Nym_3
Place_1
Place_2
Place_3
Bob
Nym_4
Nym_6
Nym_5

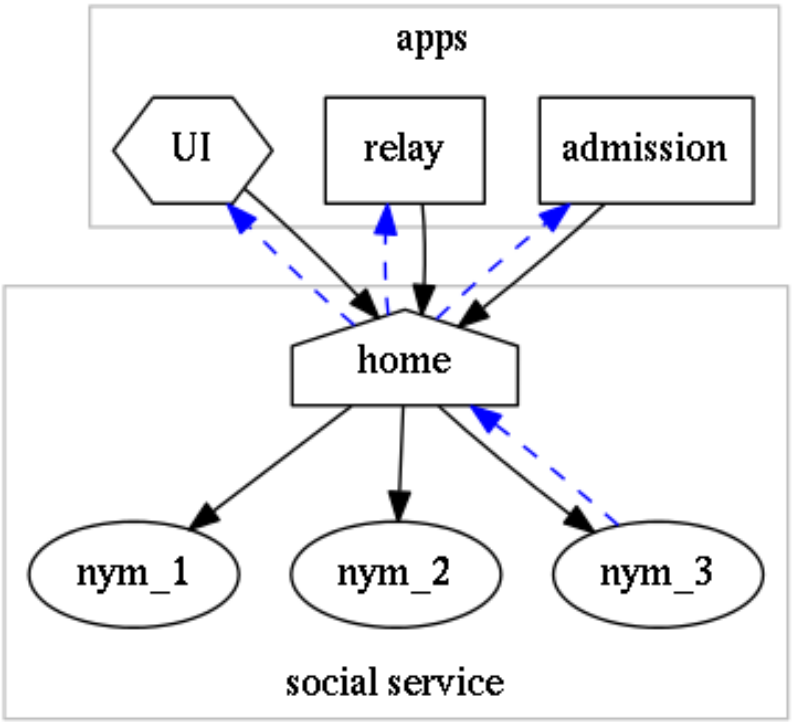
Identities

- Pseudonyms and places are identified by an ECC key pair
- GADS zone for each pseudonym
- the zone is published in the DHT under $H(Nym_{pub})$
- the zone is signed by the pseudonym
- PLACE record type for pointing to places
- empty label (+) points to a place for initial contact

+	PLACE	$H(PlaceA_{pub})$
tech	PLACE	$H(PlaceB_{pub})$
music	PLACE	$H(PlaceC_{pub})$

Place

- one-to-many messaging model
- host sends messages to guests
- guest can send requests to host
- hosts decorate their homes
- history stored locally
- applications handle method calls
- messages use the PSYC syntax



PSYC syntax

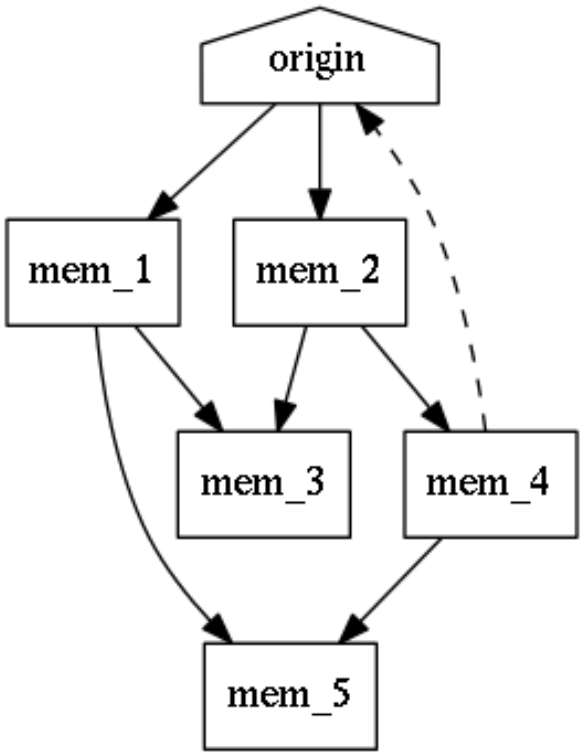
- extensible syntax and semantics
- method is mandatory, state ops and body are optional

```
:_volume 100
_message_public_shout
Hello,
world!
```

```
=_location_city Amsterdam
=_location_country Netherlands
_notice_profile_location
```

Multicast service

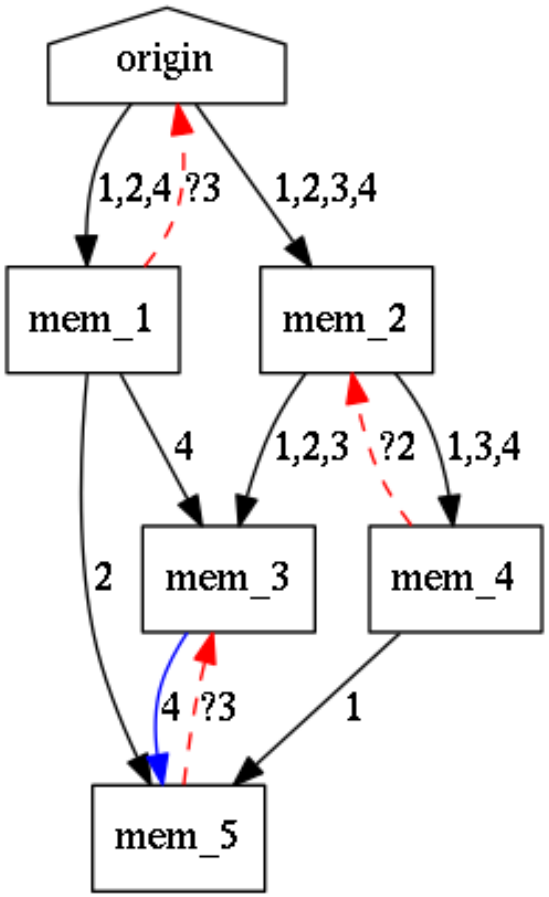
- a place is modelled as a multicast group
- origin: multicast messages originate from here
- group members are peers, no pseudonyms at this level
- messages are signed with the place's key



Joining a multicast group

- place to origin mapping: $H(PLACE_{pub}) \rightarrow H(PEER_{pub})$, signed with $PLACE_{priv}$
- look up peer of origin and send a join request there
- join request answered by application layer
- if admitted, the peer receives a list of other group members to connect, and starts receiving messages

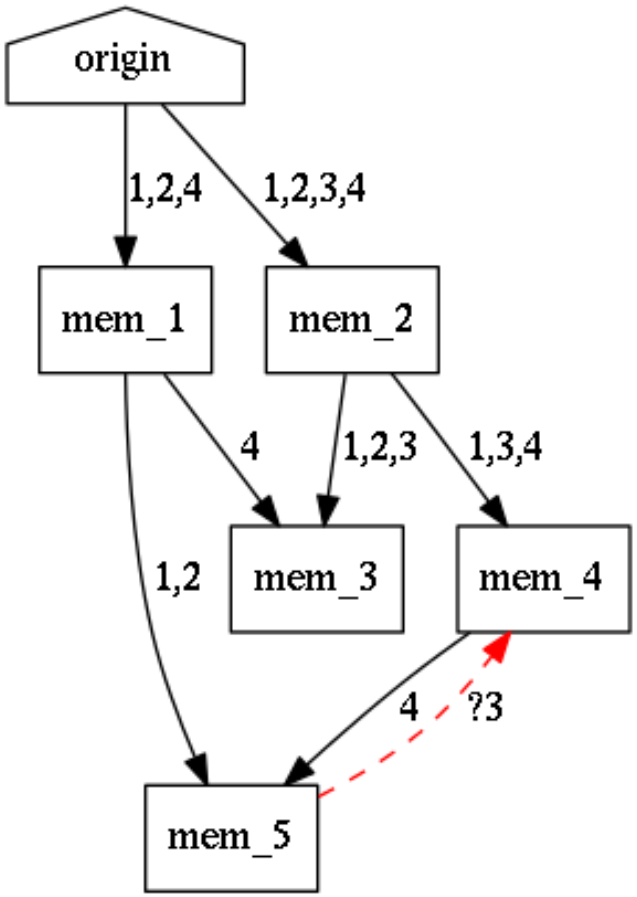
Replay



Confidentiality

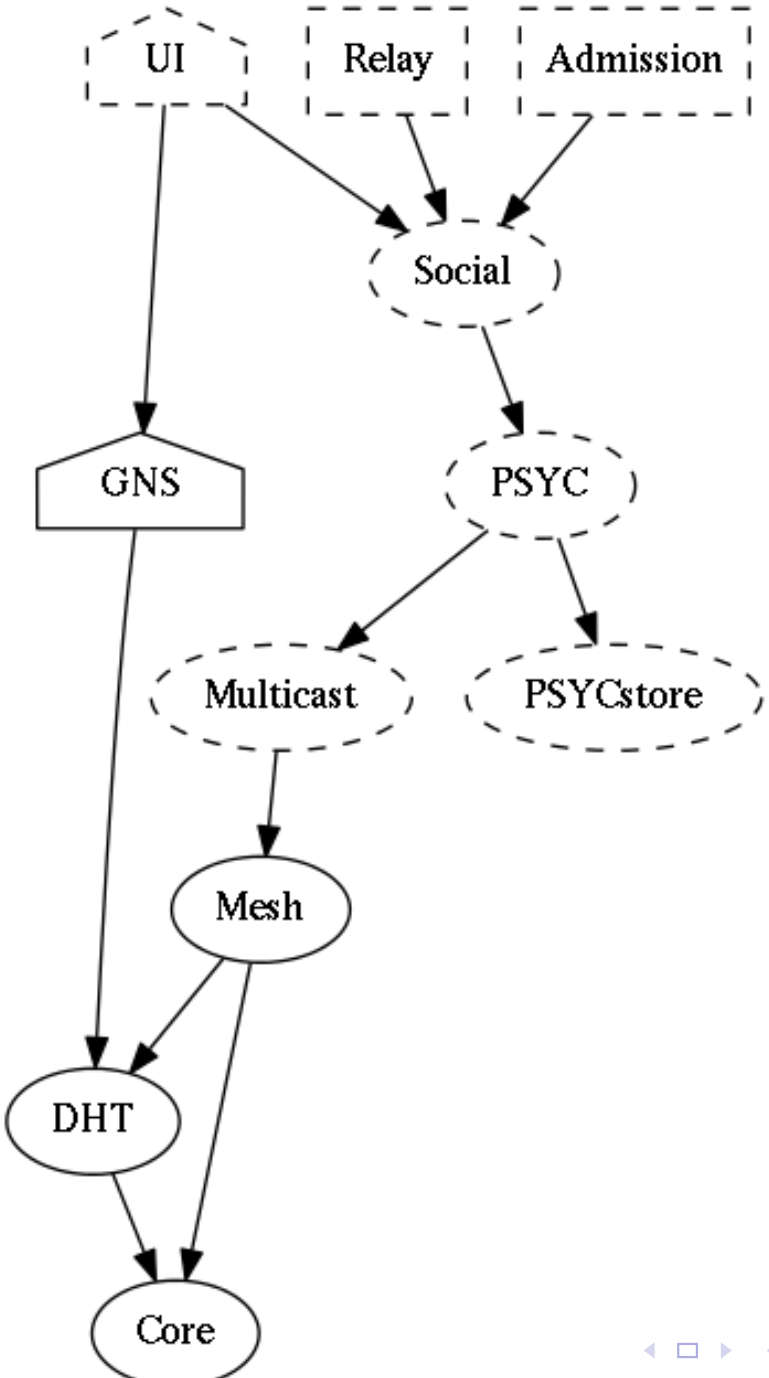
- replay only those messages, which the requester could have seen
- store join/leave events
- group generation: incremented when a member leaves
- members are trusted that they only forward messages to the intended recipients

Group generation



Components of the system

- Applications
- Social: social network model, try-and-slice
- PSYC: parse PSYC syntax and perform state operations
- PSYCstore: message history, state, membership
- Multicast: messaging and replay in multicast groups


UI
Relay
Admission
Social
GNS
PSYC
Multicast
PSYCstore
Mesh
DHT
Core

Summary

- scalability through multicast message delivery
- availability: local storage of messages
- extensibility provided by the PSYC syntax
- ECC keys for nyms & places
- GADS for naming

Questions?