Do we need clients?

Marcin Matuszewski

marcin.matuszewski@nokia.com

P2PSIP WG meeting, IETF70, Vancouver

Difference between Client and SIP UA

- Belong to different layers
  - Peers and clients belong to the overlay
  - SIP UAs work on a "P2PSIP application layer"
  - P2PSIP overlay may support various applications
- SIP UA
  - NOT aware of the overlay
  - requires SIP proxy/registrar functionality in peers
  - does not allow other applications than the conventional SIP applications
  - Security problems: based on trust in a SIP proxy/registrar

Difference between Client and SIP UA   cont.

- Client
  - is aware of the overlay
  - may turn into peers and vice versa if a situation requires it
  - allows different applications to use resources of the overlay
    - Conference announcements, content storage (i.e. pictures in user profiles), SIP location service - policy objects stored in a distributed database ...
  - reuses advanced security mechanisms used in the overlay
  - doesn't require SIP proxy/registrar functionality

The role of Client

- Peer: a "routing node"
  - Overlay maintenance, routing, and storage
- Client: a "non-routing node"
  - Accesses overlay through an associated peer(s)
  - Implements a client protocol
    - If a subset of the peer protocol = It will be implemented in the P2PSIP nodes anyway
  - A device should become a peer whenever possible ...
  - The role should be recognizable by other nodes

Reasons behind having a Client node type

- Different policies may not allow a device to become a peer
  - Security
  - Uptime ...
- If a node does not support a particular DHT it will join the overlay as a client
- NAT traversal may not allow a device to become a peer
  - NAT traversal drains battery -> keepalives

Reasons behind having a Client node type   cont.

- Resource limitations + churn/intermittent connectivity make an undesirable peer

- Undesirable peers may cause:
    - degradation of overlay network performance
    - increase of the network load

- Battery consumption and charges from "always talking" in DHT mode ...

Measurements

- Measurement scenario
  - Implementation of Kademlia for mobile phones
  - Connected to Mainline BitTorrent DHT network ~1M nodes
  - Kademlia software participating in DHT but not issuing any PUT or GET requests
- Results
  - Battery drained in ~5 hours
  - Traffic transmitted ~1MB per hour (in and out)