Requirements for SIP-based Peer-to-Peer Internet Telephony

draft-baset-sipping-p2preq-00

Salman Baset
Henning Schulzrinne
Eunsoo Shim
K. Dhara

Overview

- P2P aka overlay network
- file sharing, VoIP, presence, instant messaging, content distribution, and collaboration
- resources of participants shared to provide services
  - computation, bandwidth, storage
- may use some limited centralized resources

Potential P2P Characteristics

- good scalability
  - self-scaling: resources increase with user population
- reduced management costs
  - "servers" are user-managed
- reduced deployment costs
  - low up-front investment
- easy setup
  - not exclusive to P2P

Terminology

- DHT (distributed hash table): key ◊ value mapping, kept on a set of hosts
  - incremental forwarding of queries to something closer to authoritative source of mapping
  - may be separate from actual computational or storage resource
    - could point to resource elsewhere
- Overlay network: collection of DHTs and their internal pointers (= query paths)
  - can be clients
  - subset of clients ("super nodes")
  - special nodes operated by service provider

Basic goal

- MUST support basic voice, video, interactive text
- SHOULD support asynchronous messaging and presence

Resources to distribute

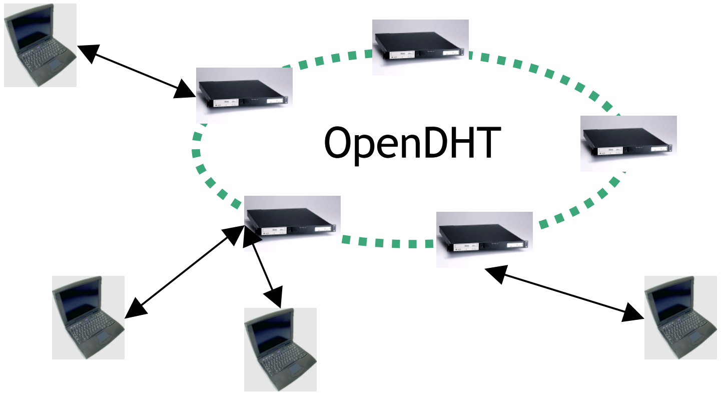
- Location service, NAT and firewall traversal servers, voicemail, address book, and configuration storage
- If possible, generic mechanism ◊ add more services later
- Note: SIP is already close to P2P
  – proxy servers not mandatory
  – proxy servers can be distributed
    - but lookup via DNS limits flexibility (domain only)

Protocol reuse

- Existing protocols such as SSL, TLS, and SIP SHOULD be reused as much as possible such that their usage does not introduce a significant overhead.

Not just one DHT

- accommodate different DHT algorithms:
  - Chord, CAN, Kademlia, Pastry, ...
  - still active research area
  - trade-off look-up costs vs. churn resilience
  - small vs. large scale
- client may be able to ignore DHT if external



NAT traversal

- The peer-to-peer system SHOULD distribute the functionality of NAT and firewall traversal servers to the end-points.
- A peer with NAT and firewall traversal capabilities SHOULD be selected such that it does not introduce significant delay between the communicating peers.

Voice transport

- The peers SHOULD support sending and receiving voice packets over TCP in addition to UDP.
  - Probably not really a P2P requirement.

Deployment scale

- The P2P system will be deployed in small offices and home networks (SOHO), emergency and ad-hoc situations, and globally over the Internet. The protocols SHOULD be flexible to cater for the varying scale requirements of these networks.

Architectural requirements

- SHOULD achieve Internet scale.
- MUST continue to function as peers arrive, depart, and fail. No assumptions on peer uptime or capabilities
  - may affect selection of DHT, however

Naming

- The system SHOULD allow centralized and non-centralized naming authorities.
  - support first-user-keeps naming
  - global naming may not be necessary in small, isolated overlays
    - may be able to qualify with p2p name

Services/resource lookup

- Some services may be centralized ◊ provide discovery
  - e.g., voicemail storage
- Interconnect with PSTN, non-P2P SIP, other P2P systems

Security issues

- Inherently different requirements and trust model
  - trust may be probabilistic ◊ similar to byzantine failure models
    - well-known results: 2/3 better be good
    - need to protect against “mole invasion”
    - but attacker may not be able to choose attacked node
  - different motivations of “evil nodes”:
    - leachers: don’t want to contribute resources
    - curiosity: steal information (but may only get random node)
    - DOS: prevent communications
- Identity
  - avoid identity theft ◊ typically, FCFS
  - sybil attacks (impersonation)

Security issues: signaling and media

- Media and signaling need to be encrypted end-to-end
  - – discourage nosy peers
  - – key exchange is hard problem (MIM)

Open issues

- Distinguish requirements for three models:
  - small-scale (zero-conf & “broadcast”)
  - built-in DHT
  - generic (external) DHT
- Characterizing security issues
  - traditional “provider is trusted” not always applicable