mSLP - Mesh-enhanced Service Location Protocol

Weibin Zhao, Henning Schulzrinne
{zwb,hgs}@cs.columbia.edu
Department of Computer Science
Columbia University

Erik Guttman
erik.guttman@germany.sun.com
Sun Microsystems

October 18, 2000
Introduction

- Service Discovery Systems
  - Goals
    * automatically discover available network services and devices
  - Applications
    * mobile, wireless, ad-hoc, home network
  - Typical systems
    * SLP, Jini, UPnP, Bluetooth, INS, SDS, Salutation, etc.
  - Models
    * directory-centric: registration/lookup
    * peer-to-peer: multicast
Service Location Protocol

- IETF standard for IP networks
  - supports both directory-centric and peer-to-peer models
- Components
  - Service Agent (SA), User Agent (UA), Directory Agent (DA)
- DA discovery
  - active
    - Multicast SrvRqst (service:directory–agent)
    - Unicast DAAdvert
  - passive
    - Multicast DAAdvert
SLP Architecture

- Scalability
  - DAs
  - service scopes

- Reliability
  - multiple DAs for each scope
SLP Deployment

- (1) small, (2) mid-size, (3) large
mSLP - Mesh-enhanced Service Location Protocol

- Interactions of SLP DAs
  - DAs within the same scope: registration forwarding (mSLP)
  - DAs in different scopes: query routing (open)

- mSLP motivations
  - improve reliability and consistency of SLP directory services
  - simplify SA registrations
    - SLPv2: SA registers with ALL DAs
    - mSLP: SA registers with ONE mesh-enhanced DAs; registrations are propagated automatically
  - scalability: thin-client SAs
  - compatibility: incremental deployment
mSLP Architecture

- Peer DAs
  - share some service scopes
  - maintain same data for common scopes (forwarding registrations)

- Peering connection
  - persistent TCP connection
  - closing: terminates a peer relationship

- Fully-meshed connection
  - greatly facilitates message exchange among peer DAs
  - a small peering DA set (sufficient to achieve high reliability)
Message Forwarding

- Mesh-forwarding extension (ID = 6)
  - used by service registration messages (SrvReg/SrvDeReg)
  - forwarding flag: on/off

- Peer DAs
  - exchange existing data when setting up a peer relationship
  - forward new registrations and updates

- Forwarding rules
  - explicit forwarding (default is not)
  - one-hop forwarding (full mesh)
Peer Relationship Management

- Three stages
  - peering setup
  - peering maintenance
  - peering tear-down

- Peer information (peer-table)
  - URL, scopes, reference to peering connection, mesh-flag, etc.

- Mesh-control message (MeshCtrl, ID = 12)
  - Pconn_Indication: peering connection indication
  - Peers_Indication: peers indication
  - Data_Get_Rqst: request for getting data
  - Data_Put_Done: done with putting data
  - Peer_Keepalive: peer keepalive
Learning about New Peers

- Configuration file
- DHCP
- DA advertisement multicast
  - mesh-enhanced DA advertisement (“mesh-enhanced” keyword)

  Mesh–enhanced DA  
  Multicast DAAdvert (Attr = "mesh–enhanced")  
  DA/UA/SA

- DA advertisement forwarding
  - from a new/rebooted non-mesh-enhanced peer
  - forwarded to mesh-enhanced peers
  - forwarded only once
    * forwarded DAAdvert: sending DA and advertised DA are different

- Peer information exchange in peering setup stage
Peering Setup

- Setup procedure
  - get peer’s advertisement
  - establish peering connection

(1) "Pconn_Indication" MeshCtrl
(2) Unsolicited DAAdvert

- exchange information about peers
- exchange data if needed
- handling new peers
Peering Setup (2)

- Exchanging information about peers

- CPL: common peer list

- Two purposes
  * learn about new peers from known peers
  * decide which scopes of data are needed to exchange
Peering Setup (3)

- Exchanging data

1. SrvReg (data of requested scopes)
2. "Data_Put.Done" MeshCtrl

Example

- DA1 (a, b)
- DA2 (a, c)
- DA3 (b, c)
- DA4 (a, b, c)
Peering Maintenance and Tear-down

- **Peering maintenance**
  - boot timestamp
  - peering connection keepalive
  - stay synchronized

- **Peering tear-down**
  - peering connection was closed
  - DAAdvert boot timestamp = 0
  - "Peer_Keepalive" MeshCtrl is timeout
Implementation and Example

- Implementation
  - extends DA functionality
    * peer relationship management
    * message forwarding control
  - simplify SAs: thin-client

- Example
  - (1) normal operation, (2) DA failure, (3) recovering from a failure

Diagram:

- DA1 (a, b)
- UA / SA
- DA2 (a, c)
- DA3 (b, c)
Conclusions

- mSLP summary
  - a fully-meshed peering DA architecture
  - improve reliability and consistency of SLP directory services
  - peer relationship management; message forwarding control
  - simplify SA registrations
  - fully compatible with SLPv2
  - mesh-enhanced DA can be deployed incrementally

- Future work
  - synchronization in peering setup with multiple peers simultaneously
  - bulk data exchange in peering setup
  - advance to RFC
  - interactions of DAs in different scopes: query routing