A RELOAD Usage for Distributed Conference Control (DisCo) – Update

`draft-knauf-p2psip-disco-02`

Alexander Knauf, Gabriel Hege
Thomas Schmidt, Matthias Wählisch

alexander.knauf@haw-hamburg.de, hege@fhtw-berlin.de,
{t.schmidt,waehlisch}@ieee.org

Agenda

- **Status** of Document
- **Overview** of DisCo – a short reminder
- **Update report** of DisCo
- **Proposal** for Media Negotiation in DisCo
- **Introduction** XML Event Package for Distributed Conferences

Status of Document (1)

- draft version -00: Initially presented at IETF 78 (Maastricht)
  - Several encouraging feedbacks
- draft version -01: Submitted 30. Dec 2010
  - Mechanism for generating chained conference certificates
  - USER-CHAIN-MATCH access policy for shared write access to overlay Resources
  - XML Event Package for Distributed Conferences
  - Media negotiation scheme for DisCo

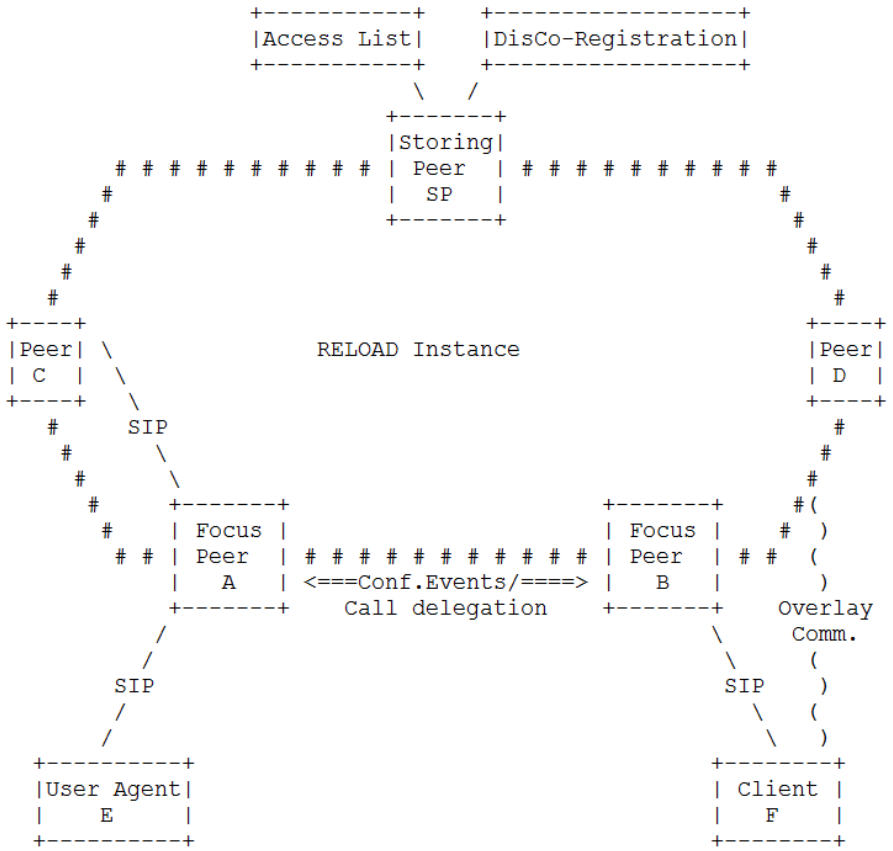
Status of Document (2)

- draft version -02: Submitted 14 Mar 2011
  - Replaced USER-CHAIN-MATCH policy and chained certificate mechanism
    - No adequate solution for revoking chained certificates
  - Using Access Control Policies of **ShaRe**[1] document instead :
    - Access Control Lists manage shared write access
    - Adopted DisCo-Registration Kind to ShaRe requirements

[1] `draft-knauf-p2psip-share-00` (Presentation by Gabriel Hege)

Distributed Conference Control

- A Distributed Conference (**DisCo**) is a multiparty session in a tightly coupled model that is controlled by several independent entities called **Focus Peers**



Conference ID Registration – Update

- Using ShaRe definitions for variable conference identifier corresponding to a naming pattern (as RegEx)
- Update of DisCo-Registration to req. of ShaRe:

New

Changed

```
struct {
  opaque resource_name<0..16^-1>;
  opaque user_name<0..2^16-1>;
  opaque coordinate<0..2^16-1>;
  NodeId node_id;
} DisCoRegistrationData
```

- **`resource_name`**: Req. of USER-PATTERN-MATCH policy
- **`user_name`**: Req. of USER-CHAIN-ACL policy
- **Additional**: Storage of Access List Kind
  - List of users allowed to register as focus peer

DisCo using ShaRe

- Creator of a conference stores **two** Kinds:
  a. DisCo-Registration: Mapping Conference ID to its Node-Id
     - Uses USER-NODE-MATCH or USER-PATTERN-MATCH
  b. Access List Kind: Initializing shared write access to DisCo Kind at this Resource-Id
- Creator may delegate write access to potential focus peers
  - Store a new ACL item delegating: creator -> pot. focus
  - Enable potential focus to register as conference controller
  - Decide on delegating write access to further parties

SDP Offer/Answer in DisCo

- Focus peers are responsible for distributing media to connected participants
- Ad-hoc scheme:
  - A Focus distributes all media streams to all connected peers
  - Focus may choose to do mixing/recoding
  - When a new peer joins:
    - Focus offers all media streams it receives to the joining peer
    - Joining peer offers its media streams to the focus
  - **Either:** Focus modifies media sessions to all connected peers, offering the new stream
  - **OR:** Mix the new stream with existing streams to prevent the need for SIP re-INVITE
  - Media streams naturally follow signaling connections

Event Package for Distributed Conferences

- Design Objectives:
  - Partial ordering of events in a distributed conference
  - Convey information about roles and relations of the conference participants
  - Announce local state of the focus peers
  - Reuse of existing XML elements of the Event Package for Conference State [RFC4575] (see figure)

```
distributed-conference
    |
    |-- version-vector
    |     |-- version
    |     |-- version
    |
    |-- conference-description
    |
    |-- focus
    |     |-- focus-state
    |     |     |-- user-count
    |     |     |-- coordinate
    |     |     |-- maximum-user-count
    |     |     |-- active
    |     |     |-- locked
    |     |     |-- conf-uris
    |     |     |-- available-media
    |     |
    |     |-- users
    |     |     |-- user
    |     |     |     |-- endpoint
    |     |     |     |     |-- media
    |     |     |     |     |-- call-info
    |     |
    |     |-- relations
    |     |     |-- relation
    |-- focus
    |     |-- ...
```

Coherent Versioning using a <version-vector>

- Uses principle of vector clocks[2]
- A <version-vector> of a conference with N focus peers has N <version> sub elements
- Each <version> announces the local state of a single focus peer with a counter
- A focus increments its counter if its local state changes and sends an event notification containing the entire <version-vector>
- Allows partial ordering of concurrent change events origin-wise
  - Detects causality violations

[2] Fidge, C., "Timestamps in Message-Passing Systems that Preserve the Partial Ordering", in Proc. of 11th ACSC , pp. 56-66,Feb. 1988.

Announcing the Local State using <focus> Element

- Aggregates state information of a conference party acting as focus peer
- A separate element for each focus
- Maps participants to focus peers
- Changes of local state updates the corresponding <focus> element
  - Increments logical clock of the associated <version> element

```
|-- focus
|     |-- focus-state
|     |     |-- user-count
|     |     |-- coordinate
|     |     |-- maximum-user-count
|     |     |-- active
|     |     |-- locked
|     |     |-- conf-uris
|     |     |-- available-media
|     |
|     |-- users
|     |     |-- user
|     |     |     |-- endpoint
|     |     |     |     |-- media
|     |     |     |     |-- call-info
|     |
|     |-- relations
|     |     |-- relation
```

Interconnecting Focus Peers using the <relations> Element

- <relation> elements used to reflect the state synchronization and media flows between the focus peers
  - ➢ enables reconstruction of conference topology
- <relation> elements contain a string of form:
  - "CONNECTION-TYPE:IDENTIFIER"
- Two connection types defined:
  - **sync**: Indicates subscription for DisCo events
    - Uses SIP call-id as identifier
  - **media**: Indicates a media connection to remote focus
    - Uses SDP 'label' to identify a single media stream
- Connection types can be extended

Next Steps

- Implementation of DisCo and ShaRe in progress
- Ready for adoption as a WG item?

Thanks for your attention!

Questions?

Alexander Knauf, Gabriel Hege, Thomas Schmidt, Matthias Wählisch

http://inet.cpt.haw-hamburg.de/