

Session Changes for Trunking

Proposal to address Issue 30

Outline

- Where we started (RFC3530)
- Changes due to sessions (as of draft-06)
- Need for further changes
- Proposed changes
- Follow-up

RFC3530: No trunking

- Each server IP treated as separate server
 - No way for client to find otherwise
 - Not allowed to act on knowledge anyway
- Each client IP treated as separate client
 - We have client_id4 to name clients
 - But client must put IP in client_id4
 - To make sure server thinks there are two clients
- Summary: Trunking is to be avoided

But wait ...

- We don't want to avoid trunking
- Trunking useful for:
 - Bandwidth aggregation
 - multi-pathing for reliability
- Sessions allow trunking
 - Multiple connections in a single session
 - Multiple sessions for a single clientid
- Great! Problem solved.

Issues (as of draft-06)

- How client knows if 2 IP's are same server
 - He's OK if he knows
 - Could use DNS in some cases
 - No standard reliable way to find out
- How server know if 2 IP's are same client
 - Could use client_id4 and compare
 - But spec still requires IP in client_id4

Addressing Issues

- Address second one first
 - Just take that requirement out
 - Requires first problem to be solved
 - Otherwise you can have server know there is one client, while client thinks there are 2 servers
 - Confusion reigns
 - That's why requirement is in RFC3530
- Need to solve both problems

Server Identification

- Needs to be reliable
 - Avoid false positives and negatives
- Present server id
 - Same uniqueness story as client_id4
 - Minus your own IP!
 - Avoids false negatives
- Can verify using SSV
 - Avoids false positives

Proposed Solution

- Do EXCHANGE_ID op on new connection
 - Subsumes CREATE_CLIENTDID
- Client provides client_id4
 - Without your own IP
- Server provides clientid (new or existing)
- Server provides server identification
 - Server id as above
 - Session set value (uint32 or uint64)

What client does with id's

- See if you have a matching server id
 - If don't, new server
 - Use the clientid to create a new session
- Existing server: check session
 - If it doesn't match,
 - Create second session bound to same client
 - If it does match,
 - Bind new connection to existing section

Implications

- client_id4 change
 - Will have to wait lease time for lock to go away when switching to v4.1
- Matches some stuff in locations_info
 - Should probably drop that stuff
 - More appropriate at server level than fs level
- Could easily address issue 6 this way
 - Could handle by adding a server-set id