SRP Replication Interop Report

Ted Lemon <elemon@apple.com>
Abtin Keshvarzian <abtink@google.com>
Jonathan Hui <jjonhui@google.com>

We didn't Interoperate

- But we learned a lot…
- Ted, Abtin and Lin got together at Apple Park
- After an hour or so of fumbling, we were able to actually try to interoperate
- We ran into a lot of issues with how peers were advertised/ discovered
- Mostly resolved those.
- Further issues getting the services to talk to each other
- Spent a bunch of time figuring that out
- Ran out of time
- No actual protocol occurred yet

Lessons Learned

- The service discovery aspect of protocol startup is non-trivial

  - We had some really good discussions about how to improve and also simplify it

  - Abtin, Lin and I are currently the three main implementors of the protocol, so there was some good discussion

Notes from the meeting

- Until we get to the ready state, we do not have a long-lived operation. This means that the idle timer is running. Every time a non-keepalive message arrives, the idle timer gets reset. So keepalives don't keep the connection alive, but continued work does. If no continued work happens and we don't get to the ready state, the idle timer kills the connection.

- When we are in the routine state and discover a new server, connect to the new server and get to the ready state. At ready state, adopt the new dataset ID. For any existing connections, send an SRPLSession message to restart the synchronization process with the new dataset ID (or kill the connection)

- Everything that's advertised in dns-sd is also in the SRPLSession, so we can validate that we have the right information. If a partner connects and it turns out that we ought to have connected to the partner, we close our connection with the partner and do a reconfirm to clear the stale data.

More notes (sequence number strategy)

- If there is no sequence number advertised and we don't remember advertising one, start at random(0..127)
- If we see one and can't sync to it, try to use it; if it doesn't work, advertise a new serial number that is +1 from the stale advertisement we are seeing.
- The new sequence number should always be lower if possible, but never the same one we advertised last time.
- This means that when a client joins the network, it will always try to join the highest number, and we won't slowly creep up to 255 and run out of options.
- We can do this in the stub network document as well, or talk about stub networks in srp replication.

We will not be deterred

- We're planning to do another interop event, possibly at the Thread Group meeting in April.
- Hopefully now that we've gotten past the discovery issues and other transport issues, we can start exchanging actual protocol messages