RFC1577 Rewrite Update

Server Synchronization Overview

RFC1577+ Server Synchronization

Background

- Lack of a multiserver model at the Amsterdam meeting (July 1993) forced RFC1577 to a single server model as an interim solution (for a year) with the directive that the roll to the future implementation without changes to the host [client].
- Consensus at the Danvers meeting (April 1995) was to include ATMARP server synchronization in the RFC1577 update.
- Update was added to the work plan at Danvers.
- draft-ietf-ipatm-classic2-00.txt was released in August 1995

RFC1577+ Server Synchronization

Goals

- Old RFC1577 clients must be able to use the multi-server service with the understanding that they are limited to connect to only one server in the service
- RFC1577+ clients allowed to use any servers in the service
- Must reliably support Class C sized IP subnetworks and extend to support arbitrarily large IP subnetworks
- Must reliably support an arbitrarily large number of servers distributed in the network according to local topology and administrative deployment decisions
- ATMARP_Requests must be allowed to be arbitrarily distributed to any server in the server and not sent directly or indirectly to a single server. (This allows the size of the LIS to scale independently from single server performance.)
- No changes needed to client code (beyond other RFC1577+ changes)
- No changes needed to client <> server protocol messages

RFC1577+ Server Synchronization

Goals, continued

- The model must minimize the impact of server-to-server synchronization messages to a reasonable percentage of the total network load
- The design of the synchronization protocol must be based on existing database/server synchronization techniques; i.e., if someone else has a proven and workable solution use it
- Maintain RFC1577's LIS model and minimal reliance on the ATM services - keep dependencies to point-to-point VCs and signaling and keep IP layer topology decoupled from ATM layer topology; i.e. if RFC1577 runs there now, RFC1577+ will run there also.
- Provide a packet format and method that allows easy extension other IETF server synchronization problems; i.e., see if the same basic mechanism can be leveraged by others (MARS, NHS)
- Allow the straightforward enhancement of authenticated ARPs

RFC1577+ Server Synchronization

Method

Xerox PARC's"Epidemic Algorithms for Replicated Database Maintenance" was selected on basis of:

- scales to arbitrarily large systems
- robust/resiliant - very few (if any) pathological failure issues
- load spreading, no one server takes the load
- straightforward to implement over pt-to-pt with signaling
- existing real use experience: XeroxClearinghouse
- *anti-entropy* and *rumor mongering* with *counter and feedback* methods appropriate
- permits arbitrary server interconnection topologies and server addition and departure
- handles network partitioning and rejoins
- rumor method separate from rumor content, can rumor anything

RFC1577+ Server Synchronization

General Mechanism

- Arbitrary number of servers are configured to synchronize amongst themselves
- Clients register with one server using same ATMARP client model of RFC1577 (based registration change already done for RFC1577+)
- Server's new information is regarded as "hot". The receiving server becomes *infective* and randomly selects and redistributes to its *susceptible* neighbors via rumor mongering process. Stops after a count of "already heard"s
- Server<>server VC appearance triggers "anti-entropy" update
- Clients have a list of servers (configured administratively or via the new MIB). One server is used until problems (timeout), vc close, etc.

  (Note: mechanism has changed with the recent disappearance of NHRP support from the draft)

RFC1577+ Server Synchronization

General Mechanism, continued

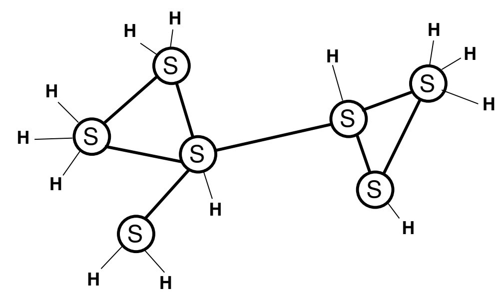
- **Rumors are serialized by the originating server**
- **Loops broken by "heard already" mechanism**
- **Authentication initially via mutual configuration, but extensible to stronger techniques**
- **Servers burdened with knowing their immediate neighbors, but not knowledge of every server**
- **Packet formats allow extensibility and other content rumors**

Note:

- **Epidemic techniques are statistical in nature and produce near complete rumor propagation coverage when the value of the threshold counter is greater then 1 (one). For value greater than 1, proportional redundant traffic is induced into the network. Xerox PARC studies indicate that a value of 2 or 3 is sufficient to induce near complete coverage (little residue)**

RFC1577+ Server Synchronization

A picture for discussion



RFC1577+ Server Synchronization

Reference

Alan Demers, Mark Gealy, Dan Greene, Carl Hauser, Wes Irish, John Larson, Sue Manning, Scott Shenker, Howard Sturgis, Dan Swinehart, Doug Terry, and Don Woods, *Epidemic Algorithms for Replicated Database Maintenance*, Xerox Corporation, Palo Alto Research Corporation, CSL-89-1, January, 1989.

Obtain from: Sandy Heynen, Xerox Corporate, Palo Alto Research Center, 3333 Coyote Hill Road, Palo Alto, CA 94304

+1.415.812.4440 or sandy_heynen@parc.xerox.com

RFC1577+ Server Synchronization

Clarifications Since Morning Presentation

- **When k ≥ #neighbors, behavior is flooding**

  When k < #neighbors, initial hot rumor floods

  Note Counter is based on number of "heards" not the number of "not heards"
- **Residual is small for our needs (see figure)**
- **"Try again" reliability a straightforward tune**
- **Packing > 1 ATMARP in a rumor is straightforward**
- **Server to Server request (forward) is also straightforward addition**
- **New clients will work with old servers**
- **Via the MIB, a new client can be told to use a new registration message that extends to authenticated registration**

RFC1577+ Server Synchronization

General Rumor Mechanism