Leveraging RFC 4533 to build a heterogeneous replication system

Emmanuel Lécharny
elecharny@apache.org
Emmanuel Lécharny

Apache Software Foundation member
Former chairman of the Apache Directory Project
PMC of the Apache Directory Project
PMC of the MINA Project
Works at IKTEK, a small company based on Identity Management and Open Source technologies
Leveraging RFC 4533 to build a heterogeneous LDAP server replication system

**Agenda**

- Introduction
- A bit of history
- RFC 4533, what's in the box?
- Using it in a heterogeneous environment
- What for?
- Roadmap
- Future steps
- Links
- Q/A
Introduction
Replication:

- Critical to any production LDAP server
- Has to be reliable
- Has to be fast
- No exit option
- Not a standard until RFC 4533 was written

This RFC opens many doors

It's not just about replication...
A bit of history
X.500 is the root
  Caching
  Shadowing
Replication is not a part of LDAP specifications
Many published drafts since 1997
A few RFCs since 2002
  RFC 3384
  RFC 4530/4533
LDUP working group 'failed' to produce a RFC
February 2004, Kurt Zeilenga's draft: *LDAP Multi-master Replication Considered Harmful*

Many servers have already implemented a LDUP like replication system, but each system is vendor specific. OpenLDAP has implemented two different system: slurpd (now obsoleted) and Syncrepl. Still looking for a common base to build an interoperable replication system...
III What's in the box?

RFC 4533, what's in the BOX?
“... and I think **syncrepl** is the best thing since copulation.”

(seen on the OpenLDAP mailing list, 18/9/2009)

Probably a bit emphatic!
What's in the box?

A standard
A protocol
Fixes some existing replication issues

- Failure to ensure a reasonable level of convergence
- Failure to detect that convergence cannot be achieved (without reload);
- Require pre-arranged synchronization agreements
- Require the server to maintain histories of past changes to DIT content and/or meta information
- Require the server to maintain synchronization state on a per-client basis
- Overly chatty protocols.
What's in the box?

Implemented so far by OpenLDAP
Replaces the defunct LDUP group
Is currently being implemented in Apache Directory Server
Replication in a heterogeneous environment
It does not need a specific protocol: LDAP is enough.

As soon as a server implements the producer part of the protocol, it can replicate itself with another consumer.

Implementing a consumer makes your server a working 'slave'.

To have the producer and consumer is not enough: you have to implement a conflict resolution system.
The consumer is the easiest part to implement

- Needs a client API
- Implement the controls
- Implement the protocol handling
- Inject the modifications into the server

Done in ADS, as a proof of concept
Can be implemented as a standalone component
The producer is more complex
   Implement the controls
   Implement the protocol handling
   Support for persistent search
   Support for polling
   Have to keep a local state (with a journal)

Not yet done in ADS

Can also be a standalone component, a kind of replication proxy.
Conflict resolution

The most complex part
Easy only in Master-Slave situation
When in multi-master, conflicts are likely to happen
   Need synchronized servers (NTP)
   Based on entryCSN
   The better the precision, the better the resolution
   Last writer wins
This is a deterministic system, it does not need a human being to resolve conflicts
What for ?
Implementing a standard

RFC 4533 is a de facto standard: it guarantees our users that they can switch from one server to another one if needed.

Maybe not the best solution ever, but what else?

In OSS world, interoperability matters.

Allows a cross replication between openLDAP and Apache Directory Server.
You can't ignore the installed servers

OpenLDAP is already installed in many places

Apache Directory Server serves a different set of needs and a heterogeneous cluster is ideal for providing the features you need based on the differing strengths offered by various servers

By implementing this RFC, we are offering more than just LDAP, but we also guarantee the users' assets

Some applications are not critical but need more extensible servers to work: we see that as an opportunity beside OpenLDAP
Apache DS offers extended functionalities

- We have implemented Stored Procedures and Triggers
- This can be leveraged in a global system where the central storage is OpenLDAP and ADS is used as an e-provisioning solution
- Apache Directory Server can be embedded, and replicated with an external server
- Can also be a solution for remote applications, when not connected
Other benefits

In companies where many different LDAP servers are installed, cross replication can help.

Dedicated system using replication
  * Auditing
  * Backups

The protocol itself can be implemented without the backend: as an API.
Roadmap for Apache DS
Roadmap for ADS

Apache Directory Server implementation status

- Remove Mitosis code from the server
- Include support for `entryUUID` and `entryCSN`
- Implement a journal to efficiently implement synrecpl
- Define a client-API being able to communicate using LDAP protocol with a remote server
- Implement the needed controls (SyncRequest, SyncInfo, SyncDone, SyncState)
Roadmap for ADS

Apache Directory Server implementation status:

- Implement the consumer part
- Write a proof of concept, with ADS being a consumer and OpenLDAP as producer
- Implement the producer part
- Implement the conflict resolution system
- Define and implement integration tests
Leveraging RFC 4533 to build a heterogeneous LDAP server replication system
Future steps
Future

Delta-Syncrepl
Syncrepl on other servers too?
Schema replication
Tooling
Leveraging RFC 4533 to build a heterogeneous LDAP server replication system