AAA: Lithium: Release Review

Contents

Project Name

AAA

Features

- odl-aaa-authn - AAA Authentication and Token services, IdmLight API, MD-SAL token store, H2 SQL user store
- odl-aaa-authn-no-cluster - Previous version of aaa-authn. No MD-SAL token store.

- odl-aaa-sssd-plugin - Apache SSSD Federated identity plugin. Pulls in odl-aaa-authn.
- odl-aaa-authn-sssd-no-cluster - Previous version of apache SSSD Federated identity plugin. Pulls in odl-aaa-authn-no-cluster

- odl-aaa-netconf-plugin - Plugin allowing authentication of Netconf clients. Pulls in odl-aaa-authn.
- odl-aaa-authn-sssd-no-cluster - Previous version of Plugin allowing authentication of Netconf clients. Pulls in odl-aaa-authn-no-cluster.

- odl-aaa-authz - AAA Authorization Service. Provides Authorization Policiy decisions for RESTCONF API access (Experimental feature)

Migration from Helium is automatic, with previous (MD-SAL less) version retained as backup.

Non-Code Aspects (user docs, examples, tutorials, articles)

AAA docs, in docs project, have been updated to reflect IdmLight usage and features.

Architectural Issues

The MD-SAL Token store is the first step in moving the AAA user-store to use MD-SAL and actually full cluster support. In this release, the SQLite backend was swapped out for H2, which is more portabe and compatible with Java 8, but it still needs to be replicated & synchronized out-of-band to deploy ODL in a cluster. The MD-SAL does not have a "timed entry" capability, thus the token store approximates one. The design decision was taken to flush entries on access, rather than to have a sweeper process. This does mean that stale tokens will show up in the cache, however in all cases only valid tokens will be honored.

Pending the full development of the AuthZ component, any authenticated user into the system has effectively full access rights.

Security Considerations

Any authenticated user, irrespective of their role, is able to access/edit the user credential store, or view Netconf device credentials. This is on par with Helium and intended to be fixed by the Authorization component. The SQL Database stores passwords in un-encrypted form.

Quality Assurance (test coverage, etc)

Standard Unit testing + Robot Tests for Token Store.

End-of-life (API/Features EOLed in Release)

None

Bugzilla (summary of bug situation)

No critical bugs, although several show stopper issues were addressed after API freeze (SQL injection vulnerability, broken IdmLight, no auth on IdmLight).

Known bugs likely to affect users are:

- Bug 3855: Special characters not allowed in user credentials.
    - No known workaround
- Bug 3820: Incorrect database initialization. The SQL database in not editable by a PSQL client.
    - Workaround: Use IdmLight API to edit entries
- Bug 3528: "Authentication service unavailable" error encountered non-deterministically during aaa feature installation
    - Workaround: Start AAA features before services using them.

Standards (summary of standard compliance)

N/A.

Schedule (initial schedule and changes over the release cycle)

Entire group of contributors, bar one, was swapped out twice. New project lead.