rxgk status update

Benjamin Kaduk
kaduk@mit.edu

27 March, 2014

Completed and Ongoing Work

- Standardization: 50% done!
- Core security class routines
- Integration with bos/bosserver

Standardization status

Simon wrote two documents, back in 2010:

- draft-wilkinson-afs3-rxgk
  - core spec for rx security class operation
  - leaves many behaviors as "implementation-defined"

Standardization status

Simon wrote two documents, back in 2010:

- draft-wilkinson-afs3-rxgk
  - core spec for rx security class operation
  - leaves many behaviors as "implementation-defined"
- draft-wilkinson-afs3-rxgk-afs
  - Fills in "implementation defined" behaviors
  - Lays out the workflow for db, file servers
  - Also registering fileservers, extended callback keys

draft-wilkinson-afs3-rxgk-11

Published as AFS-3 Experimental standard!

draft-wilkinson-afs3-rxgk-11

Published as AFS-3 Experimental standard!

- Key negotiation with GSS for initial tokens
- Combining Tokens
- rx security class operation, challenge/response, packet handling
- error codes
- Last several months were just getting the GSS text right

draft-wilkinson-afs3-rxgk-11

Published as AFS-3 Experimental standard!

- Key negotiation with GSS for initial tokens
- Combining Tokens
- rx security class operation, challenge/response, packet handling
- error codes
- Last several months were just getting the GSS text right

Oh, and the RFC 4402 $\mathrm{PRF}+$ was mis-implemented.

draft-kaduk-kitten-gss-loop

It turns out, there *is* no good documentation for how to correctly perform a GSS security context negotiation!

draft-kaduk-kitten-gss-loop

It turns out, there *is* no good documentation for how to correctly perform a GSS security context negotiation!

- Duplicated in: ssh, DNS, GS2/SASL, RPCSEC_GSS
- A good central document with sample code would be useful
- Brought to IETF kitten working group, seems to have interest

draft-wilkinson-afs3-rxgk-afs

Currently at version -05.

draft-wilkinson-afs3-rxgk-afs

Currently at version -05.
Specifies the implementation-defined behavior:

- security index
- authenticator appdata (client UUID)
- maximum size of opaque type
- how to combine tokens

Non-controversial (probably).

draft-wilkinson-afs3-rxgk-afs

Also implementation-specific:

- Token format (tokens can be produced by one server and consumed by another)

draft-wilkinson-afs3-rxgk-afs

Also implementation-specific:

- Token format (tokens can be produced by one server and consumed by another)

New functionality (on top of the core spec):

- cache manager tokens for use with extended callbacks
- AFSCombineTokens to populate that callback id, translate token
- server-to-server communication
- registering file servers as rxgk-capable

draft-wilkinson-afs3-rxgk-afs

Also new functionality on top of core spec:

- SetCallBackKey

This document does not cover extended callbacks (see draft-benjamin-extendedcallbackinfo). However, the need for a secure channel for extended callbacks $\Rightarrow$ affects the token format.

Core rxgk security class for OpenAFS

Core rxgk security class for OpenAFS

- An implementation of the core rxgk security class is in gerrit, changes 10561-10576 (except 10569), 10589-10591, 10936-10938 — the 'rxgk' topic
- All of them need more review.
- Some got more self-review than others.
- Should also review for spec compliance

rxgk-ified bos/bosserver

The BOZO RPCs are neither db server nor file server, so only the core spec applies, and there is lots of leeway.

- GSSAPI negotiation requires pthreads (or a LWP GSSAPI library)
- Thus, rxgk-bos depends on pthread-bos (Chas's work), in gerrit
- rxgk-bos is not in gerrit; waiting for rxgk and pthread-bos to settle
- `https://github.com/kaduk/openafs/commits/rxgkng` is frequently updated, but has rxgk-bos code

rxgk-ified bos/bosserver

The BOZO RPCs are neither db server nor file server, so only the core spec applies, and there is lots of leeway.

- GSSAPI negotiation requires pthreads (or a LWP GSSAPI library)
- Thus, rxgk-bos depends on pthread-bos (Chas's work), in gerrit
- rxgk-bos is not in gerrit; waiting for rxgk and pthread-bos to settle
- `https://github.com/kaduk/openafs/commits/rxgkng` is frequently updated, but has rxgk-bos code

I have rxgk-bos running on a test server; it works.

Lots left to do!

Trying to track outstanding tasks at
`http://wiki.openafs.org/RXGKToDo`

Lots left to do!

Trying to track outstanding tasks at
`http://wiki.openafs.org/RXGKToDo`

- Need vldb and prdb format bumps for full support
  - prdb pretty well understood
  - vldb needs investigation
- Some small tasks, some big tasks
- Some decisions to be made

Open questions

- Non-cell-wide keys on disk. Where to put them?
  - Per-fileserver key
  - bosserver's token-encrypting key? Or ephemeral?
- Combined tokens as user tokens — union or intersection?

Thanks!