Rx Security
The path to safety

- D Brashear
- The OpenAFS Project
- 26 March 2014

The backstory

- AFS began life with its own authentication and encryption systems
  - AuthServer for authentication
  - bcrypt for encryption

Evolution

- Cooperation with MIT
  - Andrew (AFS) was a CMU project
  - Fledgling Kerberos 4 from MIT was adopted to replace AuthServer

Evolution

- DES was the standard of the day
  - But for optimization reasons, a similar algorithm called fcrypt was used for protecting AFS traffic
    - Different set of sboxes

The history of AFS

- CMU spun off AFS to a new company
  - Transarc had IBM as an investor
    - IBM had rights to all of Andrew
  - IBM bought Transarc out
  - Then IBM gave up
    - The result was OpenAFS

Evolution during IBM

- IBM realized DFS had lost
  - Some work on AFS was done
  - In spite of a plan for krb5, it never happened
    - Ben Cox spoke to this at Decorum ’99
    - The planned work looked a lot like rxkad-k5 does… 15 years ago.

OpenAFS takes over

- No love for krb4 or rxkad
  - Work done to allow krb5 tickets as tokens
  - Plan for rxgk
- But slow going

DES cracking machines

- The limited value of DES was known
  - But a service called chapcrack made things really accessible

A student project

- Four MIT students took on a project for 6.858.
  - Peter Iannucci, Alex Chernyakhovsky, Patrick Hurst, Christy Swartz
  - Their advisor was a previous OpenAFS Security Officer (Nickolai Zeldovich)

Kerberos with 2 Heads Napping

- Service ticket requested
- Sent to CloudCracker for cracking
- Recovered key can be used to forge service tickets or compromise subkey negotiation.

rxkad-k5

- Basically the same as rxkad
  - Still a DES session key
  - krb5 ticket instead of krb4
    - kvno 256 signifies krb5 ticket
- No client changes

rxkad-k5

- Allows non-DES service keys
  - but KDC must still support DES for session keys
  - principal's service key not vulnerable
  - Client should only request non-DES in TGS-REQ if kdf is supported in the client
    - aklog and afslog already did this correctly, so no new client needed

Wither DES

- Some sites would not be willing to continue to support DES on KDCs
  - Key derivation would allow KDCs to use only strong key types
  - But every server then also needs updated software

rxkad-kdf

- Uses non-DES session keys
  - KDC doesn't need DES.
  - But a new aklog required on the client

Key derivation

- NIST SP800-108 for derivation
  - using HMAC-MD5 in counter mode as the pseudo-random function
    - requires a (pseudo)random input key! Some enctypes need not apply
  - de-parity DES3 keys so they are random
  - use DES keys directly
  - rounds of derivation done until a non-weak DES key is produced, or rounds exhausted

rxkad limitations

- No way to negotiate features
  - KDC returning service key stronger than DES means rxkad-k5 supported.
  - KDC returning session key stronger than DES means rxkad-kdf supported.

Deploying it

- Update the servers
- Rekey the AFS key with new enctypes
  - Requires a dance to avoid an outage
  - And there are Heimdal bugs for older versions
- Restart the servers to make server-server communication use new keys
- Remove the old KeyFiles after any old keys have expired

Details

- If you are using afs@REALM, you should convert to afs/cell@REALM (and there are instructions)
- You can create a keytab from the old key and use it to run kadmin if your Kerberos admins are unhelpful.
  - Requires a dance to avoid an outage
  - And there are Heimdal bugs for older versions
- You can use tricks to get a keytab before the database contains it live.

Are we safe yet?

- rxkad-k5 session key still (crackable) DES
  - But it gets you history only (24hr life vs ~24hr crack time)
- rxkad-kdf provides stronger session key
- but in the end, rxkad still uses a 56 bit key

rxgk

- Required for real cipher suites
- More on the topic later from Ben Kaduk

Fin

- Questions?
- Bibliography:
  - Kerberos with Two Heads Asleep: https://www.dropbox.com/s/hcfua3htd6k7xqw/proj-final.pdf
  - Improving the OpenAFS Security Model Without Client-side Changes http://web.mit.edu/achernya/Public/thesis.pdf
  - krb5 based key derivation for rxkad http://tools.ietf.org/pdf/draft-kaduk-afs3-rxkad-k5-kdf-00.pdf
  - How to deploy rxkad k5 and kdf http://www.openafs.org/pages/security/install-rxkad-k5-1.6.txt
  - How to generate rxkad.keytab for deployment http://www.openafs.org/pages/security/how-to-rekey.txt