Daredevil Kerberos

Kerberos at a Hypothetical Large Financial Institution

Roland Dowdeswell <elric@imrryr.org>

Environment

Imagine that you have 50,000 UNIX machines, 80,000 Windows machines and a few mainframes using Kerberos for SSH, AFS, LDAP, MQ, Sybase, DB2, Oracle, Informix, SMTP and hundreds if not thousands of internally developed applications in an environment where interruptions of service are unacceptable.

Environment Part 2

- What does this mean practically?
  - The market is open 22 hours a day
  - Escalations can occur at any time
  - I've lost many a weekend...
- Kerberos Libraries
  - MIT Kerberos 1.4, 1.3, 1.2, KFW 3.2, 2.6.5
  - Microsoft
  - Cybersafe Kerberos
  - Java JGSS (1.4, 1.5, 1.6)
  - Oracle
  - Mainframe
- Vendor products

MIT Kerberos Libraries

- Issues
  - Code quality
  - Race conditions, edge cases
  - ABI changes
  - Replay cache issues
- Patches
  - forward_with_noaddresses
  - Ignoring addresses
  - TGS_REQ failover
  - Incremental propagation
  - Inetd-mode krb5kdc
  - A number of bugs...

Java Kerberos Issues

- Encryption type support
- Servers perform AS_REQs
- Failover is not enterprise ready
- Keytab processing: appears to throw exceptions if it processes a key in a keytab with an enctype that it doesn't understand even if it is not being used
- Can't use ccache_type < 3 on little endian boxen
- Difficult for Java programs to migrate JVM versions

Other Libraries

- Cybersafe
  - DES only
  - Can't handle addressless tickets
  - Can't use ccache_type >= 3
- Oracle
  - Can't parse krb5.conf with long lines
  - Can use ccache_type < 3
  - Can't do pre-auth on AS_REQs
- Microsoft
- Mainframe

MIT kadmin{,d} Issues

- ACLs not terribly expressive
  - Can't query external databases
  - Can't express complicated transactions
- Can't remove old keys when -keepold
- Can't remove enctypes from existing keys
- Key rotation has race conditions
- Stability issues
- Code quality makes extension and modification difficult and error prone

KNC: Kerberised Netcat

- Works mostly like netcat
- Server:
  - Can run standalone or out of inetd
  - Will fork/exec a program upon accepting a connexion or connect to a UNIX domain socket
  - Communicates the client credentials via either environment variable or as a header prepended to the exchange
- Client:
  - Either connects to the server by initiating a TCP connexion, or
  - Accepts a fd on the command line which it will use as a previously established TCP connexion

SSP: Simple Secure Protocol

- Perl based RPC mechanism
  - Takes a Perl object and exposes it as a program that speaks a protocol that looks very similar to SMTP on stdin/stdout
  - Marshalls and unmarshalls arguments and returns that include an arbitrarily nested list of hash refs and list refs
  - Handles exceptions
  - Clients failover
  - Servers can redirect clients to support read/write master, read only slave models (such as MIT Kerberos)
  - Uses KNC to authenticate and encrypt

krb5_admind

- We used KNC/SSP to write a Kerberos Administration Daemon that is easy to extend
- Allows definitions of new commands with minimal effort:
  - Fetch
  - DES Deco
  - NKKPS
  - ``Proid'' password change

krb5_keytab

- Manages keytabs on servers
- Works under the assumption that host/hostname can create/fetch keys for service/hostname
- Is a suid-root application which checks the entitlements of an ID and determines if it is allowed to create/fetch a particular service key
- Is expected to be run in the server's rc script

Questions?

Kerberos IV Deco

DES Deco

- Different Kerberos libraries support different lists of enctypes
- Developers don't understand the issues and need to be protected from the information
- Split the problem up:
  - TGS Keys
  - Service Keys
  - User Keys
  - Session Keys

DES Deco: Service Keys Part 1

- Constraints
  - Clients do not need to understand the enctype of tickets but they must understand the enctype of session keys
  - KDC selects enctype for session key based on the keys configured for the server in the Kerberos DB
  - And so: each server must be configured with keys that it supports in the Kerberos DB

DES Deco: Service Keys Part 2

- The solution: krb5_keytab creates strong keys by default
  - Needed to define an exception mechanism
  - Users were confused by enctypes
  - We changed the conversation from what enctypes were in keytabs to what Kerberos libraries the server was running by:
    - providing an option to krb5_keytab (-L) which would select the appropriate encryption types based on the reported Kerberos library
    - Providing a -q option which reports what libraries each principal in a keytab will support
    - Providing a -t option which in conjunction with -L will test if the principals in a keytab support a particular lib

DES Deco: The TGS Key

- Ensure that session keys are compatible with all krb5 libs that might be used
- Must use -keepold but there is no way to delete the old key
- Race condition when changing the TGS Key
  - We largely avoid this because we list the KDCs in /etc/krb5.conf and that means that you are overwhelmingly likely to use the same KDC for TGS_REQs each time
  - We also patched the MIT Kerberos libraries to fail back to the master on TGS_REQ failures

DES Deco: User Keys

- MIT Kerberos does not provide a mechanism for different default enctypes for each user or groups of users
  - Ideally, one could attach default enctypes to principals and policies
  - Due to time constraints, we deployed a hack in kadmind: if a policy starts with ``strong_'' then it will use aes256-cts, aes128-cts, rc4-hmac, des3-cbc-crc
- Provided a self-service tool
- Scanned Kerberos logs and upgraded users when appropriate

Questions?

What We'd Like To See

- Better support for HTTP
  - SPNEGO is insufficient
- Implementation of GSSAPI for TLS
  - TLS libraries such as OpenSSL must provide a transition mechanism which simulates certificates for GSSAPI authentication
- EKE
- HSMs
- Better vendor support for Kerberos

Conclusion

Although we have mentioned many of the issues that we face in deploying Kerberos at such a large institution with such stringent requirements, it is working quite well...

Questions?