



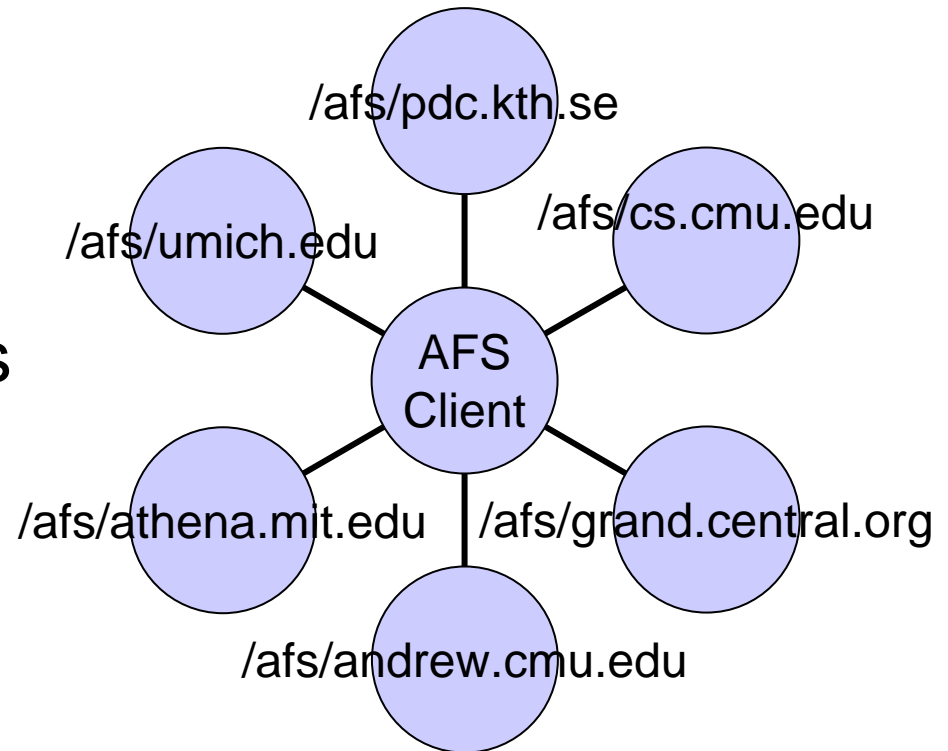
*An Open Source Wide-Area
Distributed File System*

Jeffrey Eric Altman

jaltman *at* secure-endpoints *dot* com

What is AFS?

- A global wide-area Distributed File System providing location independent authenticated access to resources administered by unaffiliated organizations



A Brief History of AFS

- 1980's: Carnegie-Mellon University
 - Information Technology Center (Andrew Research Project)
 - Original goal: one site supporting 10,000 users on private workstations or public clusters
 - Coda (another dfs research project has shared roots)
 - The AFS RPC protocol Rx is a peer of SunRPC. They both encode data with xdr
- May 1989 Transarc Inc.
 - Commercial distribution and support for UNIX platforms of AFSv3
 - Multiple cell architecture
 - Began work on AFSv4 which became DCE DFS



A Brief History of AFS

- 1995 IBM Pittsburgh Labs
 - Microsoft Windows port
 - DCE/DFS took precedence
- 2000: OpenAFS <<http://www.openafs.org>>
 - IBM forked the code base and open sourced
 - <http://www-106.ibm.com/developerworks/linux/library/os-afs.html>
 - Better support, new platforms, cool stuff

Basic Concepts

- **Cell**
 - a collection of servers (file, volume database, protection database, ...) administered as a unit
- **Volume**
 - a unit of storage owned by a single user that stores directories
- **Directory**
 - a container owned by a single user that includes directories, files, mount points, symlinks, and access control lists



Basic Concepts

- **Mount Point**
 - a directory entry that refers to a volume within a cell.
 - mount points can specify a read-only or read-write path
- **File Server**
 - a server that manages access to the contents of volumes
- **Volume Database Server**
 - a server that manages the file server(s) volumes can be found on



Basic Concepts

- Protection Server
 - stores group membership information
 - computes the rights permitted for a given client when accessing the contents of a particular directory
- Token
 - a credential used to authenticate a user to an afs server
 - contains the service portion of a kerberos ticket



Strength: Scalability

- Single name space accessible from everywhere
- AFS was designed for a ratio of 200 clients per server
- Servers can be added and removed as needed
- Servers can be distributed across the globe



Strength and Weakness: I/O Performance

- Read-only volumes can be replicated
 - Redundancy
 - Per-client server preferences allow nearby copies to be accessed first
 - Randomization of server lists balance the load
- Read-write volumes
 - Single copy
 - Write on close



Strength: Authentication

- Kerberos 4 authentication server built-in
- Kerberos 5 authentication should be used but requires extra setup for the realm
- Cross-realm support
- Multiple cells per realm
- Access multiple cells at one time from a given logon session or Process Authentication Group

Strength: Authorization

- Per volume or directory; not per file
- Access Control Lists
 - Directory rights
 - (a)dminister
 - (l)ookup
 - (i)nsert
 - (d)elete
 - File rights (affects all files in the directory)
 - (r)ead
 - (w)rite
 - loc(k)
- Protection Groups (max 5000 members)
 - System-wide
 - User-created
- Foreign principals can be granted access



Strength: Caching

- Caching improves performance
 - Memory cache
 - Persistent disk cache
- Distributed cache coherency algorithms

Weakness: File Locking

- Posix style full file locks supported
- Byte range locks not supported
 - Do not store databases in AFS
 - Microsoft Office apps only utilize byte range locks
- Don't store databases in AFS
 - Databases require byte range locks
 - Databases have their own replication engines



Strengths: Volume Management

- Location independence
 - Volumes are referred to by names
 - Volume DB specifies on which file servers they can be found
- Contain multiple directories
- Can be moved while in use
- Can be replicated as read-only copies
- Can be mounted anywhere within AFS



Weakness: Disconnected Operations

- No support for disconnected operations
- Even when data is in your local cache you cannot use it without cell access

Strength: Low-cost hardware

- Custom hardware is not required
- RAIDs can be used
- Replicated files servers containing read-only volumes are expendable

Usage Models

- Some sites utilize AFS only for read-only data such applications and static web content
- Some sites make extensive use of AFS for home directories and Windows roaming profiles
- Some sites do both

OpenAFS release 1.2: Supported Operating Systems

- UNIX
 - Solaris
 - AIX
 - HP/UX
- Linux
- Microsoft Windows (dead)
- MacOS X
- NetBSD / FreeBSD



OpenAFS Release 1.4: New Operating Systems

- Solaris 10 (Sparc and X86)
- Linux 2.6 kernels
- AIX 5.3
- MacOS X 1.4 (Tiger)

OpenAFS release 1.4: General Improvements

- Large File support
- Multi-Threaded Servers
- Threading optimizations for RPC libraries
- Faster volume releases
- Super Groups (groups of groups)
- Kerberos 5 Large Ticket Support



OpenAFS release 1.4: More improvements

- Fixes for file server issues which have been present in every release since Transarc days
- MacOS X improvements for the Finder

OpenAFS release 1.4: Microsoft Windows improvements

- Too many to list (200 changes since Nov 2003)
- Monthly Status reports published
<http://www.secure-endpoints.com/>

Demo?

If we have network
connectivity

AFS and Kerberos Best Practices Conference

- June 20-24, 2005
Carnegie Mellon University
 - Two days of tutorials
 - AFS administration
 - Kerberos administration
 - Three days of talks
- <http://www.pmw.org/afsbpw05/>



Useful Links

- <http://www.openafs.org>
- <http://grand.central.org/twiki/bin/view/AFSLore/WebHome>
- <http://www-2.cs.cmu.edu/afs/andrew.cmu.edu/usr/sadow/www/afs.html>

Q&A

You ask, I answer



Contact Information

Jeffrey Eric Altman

jaltman *at* secure-endpoints *dot* com