



# AFS (Andrew File System) on Linux for S/390 and zSeries

---

David Boyes  
Session L05



## Agenda

---

- What is AFS?
- Virtue, Vice and The Devil: Benefits and Drawbacks of AFS
- Where Do You Get It?
- Planning an AFS Install
- Adapting Applications for AFS



## What is AFS?

---

- AFS is a distributed file system
  - Separate component file and authentication servers
  - Strong authentication server
  - Cache manager



## AFS Implementation Concepts

---

- Cell – a group of machines under a single administrative control org
- Server – a physical machine providing AFS functions as part of a cell
- Server process – the internal components of the AFS software



## AFS Data Concepts

- Volume – a group of files managed as a unit
- Replication – copying of popular volumes to multiple physical disk partitions
- Caching – storing parts of files in use on local disk



## Virtue, Vice and the Devil


- |   |  |
|---|--|
| <ul style="list-style-type: none"><li>■ Benefits of AFS<ul style="list-style-type: none"><li>■ Strong security model</li><li>■ Automatic survivability of file system access for replicated files</li><li>■ Global filespace naming</li></ul></li></ul> | <ul style="list-style-type: none"><li>■ Drawbacks of AFS<ul style="list-style-type: none"><li>■ Invasive install</li><li>■ Complexity of back end server functions</li><li>■ Authentication issues with applications</li></ul></li></ul> |
|---|--|



## Versions of AFS

---

- CMU
- Transarc/IBM
- University of Michigan (OS/390)
- OpenAFS
- ARLA/milko



## OpenAFS

---

- Kernel interface module
- Client/server
- Very, very OLD code
- Deep ties to internal kernel structures
- Probably a bad choice for a port, but we didn't know it at the time...8-)



## OpenAFS

---

- Compile Modules
  - Debug SuSE non-export of kernel memcpy function.
  - make takes literally hours to process due to ancient dependency checking.
  - Dealing with:

```
/* This code will be removed in the rewrite in 1989 */
```

in many, many places...



## OpenAFS

---

- Compile Modules
  - Linux pthreads implementation incompatible with CMU pthreads.
  - Assumption of int = char representation.
  - '1' and x'00001' are NOT identical on S/390 (fixing improper casting of pointer)
  - Fencepost errors in buffer management



## OpenAFS

---

- Compile Modules
  - Interface to setjmp() internal task buffer organized very differently on S/390 than other platforms
  - Implementation of context switch assembler routines (Linux assembler is NOT OS linkage compliant).



## Where Do You Get It?

---

- Current 1.04 code is available at:
  - [www.openafs.org](http://www.openafs.org)
  - [oss.software.ibm.com/developerworks/openafs](http://oss.software.ibm.com/developerworks/openafs)
- Patches for S/390 and performance improvements (pre-1.04):
  - [www.sinenomine.net/downloads](http://www.sinenomine.net/downloads)



## Where Do You Get It?

- Documentation files are available from IBM's developer site ([oss.software.ibm.com](http://oss.software.ibm.com)).
- READ THE DOCUMENTATION.
- READ THE DOCUMENTATION.



## Planning an AFS Install

- To practically implement AFS, you need a minimum of two machines (server and client)
- In production, plan to separate all AFS functions onto separate machines (see next slide)



## AFS Roles and Processes

---

- Five major roles in an AFS cell:
  - System control server
  - Database server
  - Binary distribution server
  - File/volume server
  - Client system



## Planning an AFS Install

---

- Before installing any machines in these roles, plan the location and distribution of your data.
  - File server machines on all major segments of network is recommended



## Planning an AFS Install: File Servers

---

- Build system for I/O performance
- RAID desirable but not necessary
- Large RAM desirable (256M or greater)
- Network performance is critical.
- Separate partition for AFS data
- Plan to dedicate at least one file server to each major group of users



## Planning an AFS Install: Database Servers

---

- RAID with snapshot capability recommended here.
- Multiple channel RAID recommended
- Maximum RAM possible.
- First DB server installed should have the lowest IP address available (saves MANY updates)



## Planning an AFS Install: System Control and Binary Distribution

---

- Often combined with first database server
- Replicate on all DB servers (18M of disk isn't much to give for enhanced reliability)



## Setting Up AFS

---

- Before you start:
  - Read the Quick Start guide completely. Some things you CAN'T back out of.
  - Ensure you have plenty of time to build the AFS code. Directions:
    - [www.sinenomine.net/downloads](http://www.sinenomine.net/downloads)



## AFS and Kerberos

---

- Note that the internal Kerberos implementation in AFS is not mainstream Kerberos 4
- Docs in progress to describe minor adaptations to use mainstream K4 and K5 implementations



## Adapting Applications for AFS

---

- Applications in the AFS environment need to handle some differences in these issues:
  - Authentication
  - Access Control List Management API
  - Ticket Expiration (new)



## Adapting Applications for AFS

- Required for any application using common userid/password
- See AFS Administrator Guide appendix for details of code changes



## Adapting Applications for AFS

- Ticket Expiration
  - New problem – AFS/Kerberos tickets expire
  - Long running applications need to cope with renewing authentication periodically

