

AFS File Services

Peter van der Reest
DESY – IT – Information Fabrics

Linux User Meeting December 2006

DESY

Content

- Basics
- Policies
- Performance
- Trends
- Future of OpenAFS

AFS Basics

AFS is a

- distributed,
- Kerberos secured
- global

filesystem which

- in use in production environments for over 2 decades
- went from academic closed source to commercial to Open Source

AFS Basics II

Basic concept centers around distributed databases for

- authentication (Kerberos IV and V)
- authorisation (users, groups, ACLs)
- data management (backup, restore)
- data location

User data is held on file server machines

where basic entity is a volume

AFS Basics III

Volumes

- hold partial file system trees (e.g. home dir)
- are mounted in the file system
- appear as subdirectories
- lie on one fileserver (RW) – or on several (RW/RO)
- can have a backup clone (~/ .OldFiles /)
- have a limit on available space

Policies



- every DESY account comes with an AFS volume or home directory
- initial quota of ~ 500 MB at present
 - we're looking at 2 GB in the near future
- any volume can be upgraded to 2 GB by group admin
- any volume can be upgraded to 8 GB by IT AFS admin
 - if usage is of a work related nature and provided there is no better alternative

Policies II

Alternatives to 'huge' home volumes are

- not using AFS (not my preference)
- several smaller volumes forming a large home dir
- using group volumes

Group volumes have longer life than user volumes

Group volumes may expand to over 100 GB

in special circumstances

Policies III



Why so little space ?!



Policies III

Why so little space ?!

Because AFS space for home directories is expensive:

- data resides on HQ disks
- data is backed up twice: file level and volume level
 - file level for user driven restoration (6 generations)
 - volume level for admin driven catastrophe recovery
- data is embedded in a high availability environment
 - most near misses, you never know about!

Policies IV

For some time we have been piloting a new course:
for replicated or easily generated data we allow for

- large volumes
 - > 16 GB, maxed by physical partition size of 2 TB
- without doing backup, either file or volume

In special cases we can do file or volume backup,
depending on the use case

Performance

AFS can be slow, yes

Improvements are in

- cache daemon tuning
- newer software releases
- newer hardware in server and client environment

Performance of afsd

Cache performance is dependant on

- cache size
- chunk size
- number of threads and data localisation
- local cache disk hardware
- user driven usage patterns

Trends

as performance goes towards 30+ MB/s the need to use NFS for speed reasons weakens (and losing the hassle of cross mounting NFS servers can really be appealing, not to mention security)

- turning from NFS towards AFS
- consolidating data in one file system on Unix
- consolidating user data across platforms

Future of OpenAFS



almost all development effort goes into securing a future for the maturest of distributed file systems

- performance and reliability
 - rewrites of cache daemon
 - rewrites of file server suite
 - rewrites of the RX protocol, porting to TCP and IPv6
 - hardening the code for today's IT environment

Future of OpenAFS II



improving integration in existing infrastructures:

- providing PAM modules that actually work
- rewriting the database backend to use LDAP
- porting the authentication methods to K5 (finished)
- improving the AFS Windows client
- integrating into MS devel and acceptance cycles
- rewriting the Windows client to be a pluggable FS
- implementing disconnected mode

Future of OpenAFS III



Adaptation to 'modern' views in computing

- providing support for nested structures
- providing support for stateful connections
- enhancing support for authentication models

although these will mean breaking backward compatibility at some point in time

(maybe as soon as OpenAFS 1.6)

Any Questions ?

afs.service@desy.de

peter.van.der.reest@desy.de



HAMBURG • ZEUTHEN

