iRealm: Explorations in using OS X to provide AFS and Kerberos Services

Thomas L. Kula
Information Technology Services
Iowa State University

2006 AFS and Kerberos Best Practices Workshop
Outline

- What this talk covers
- What it doesn't
- Kerberos from Open Directory
- OpenAFS server
- Caveats and Final Thoughts
What This Talk Covers

- A proof of concept
- Caveats and things encountered
- Starting point for further investigation
- It is more of a “I know Kerberos/AFS and want it to work on OS X Server” talk and not “I'm a Mac person and want this Kerberos/AFS stuff”
What This Talk Does Not Cover

• Not an exhaustive investigation
  – Loch LaVerne did not catch on fire
  – I'm not aware of any three-headed calves being born in or around the Greater Ames Metroplex Area
  – Cats and dogs did not, as far as I know, live together
  – I wouldn't base my entire cell on this talk, however

• Does not delve deeply into the Kerberos side
  – Open Directory is documented by Apple
  – Underneath, it contains MIT Kerberos
Scenario

• Configure a Kerberos realm and corresponding AFS cell for irealm.awesmoe.org
• Demo machines have OS X Server 10.4.6
• Also installed is the stock OpenAFS 1.4.1 package for OS X 10.4
Open Directory and Kerberos

• The OS X Server directory service is called Open Directory
• Based on a combination of open and proprietary technologies
• Includes MIT Kerberos and can operate as a KDC
• I will primarily discuss a few hints and caveats since most of this is documented elsewhere
Hints on Realm Naming

• By default, your Kerberos realm is your Open Directory master's hostname, upcased.
• If you configure an OS X server to be an Open Directory master during machine setup, this is what you get.
• I'd rather not have my realm name be SERVER-1.IREALM.AWESMOE.ORG.
Hints on Realm Naming

• Instead, during initial setup of your Open Directory master, make it a “Standalone” machine
• Then, using the Server Configuration Tool, change it into an Open Directory master
• This will allow you to specify a realm name of your choice
• It will not work unless your machine's hostname matches the name for your machine's IP
Kerberos Miscellaneous

• Users that live in Open Directory, have kerberos principals
  – Users local to a machine, of course, do not

• Any user that lives in Open Directory and is marked “User can administer this directory domain” can make changes to the Kerberos database, as can anything with an “admin” instance

## This file autogenerated by KDCSetup ##
*/admin@IREALM.AWESMOE.ORG *
alice
Adding User Instances

• Best practice is to use separate User instances for administrative tasks
  – alice/admin for kerberos administration
  – alice/afs for AFS administration

• There is no way of doing this integrated with the standard OS X Server administration tools

• kadmin:
  addprinc alice/afs
Where are the AFS server binaries

- Located in
  
  `/Library/OpenAFS/Tools/root.server`

- `ls -R /Library/OpenAFS/Tools/root.server`
  
  `etc  usr`
  
  `/Library/OpenAFS/Tools/root.server/usr/afs/bin:`

  - `asetkey`
  - `upserver`
  - `fileserverserver`
  - `klog.krb`
  - `salvager`
  - `bos`
  - `vlserver`
  - `fs`
  - `kpwvalid`
  - `tokens`
  - `bos_util`
  - `volinfo`
  - `kas`
  - `pt_util`
  - `tokens.krb`
  - `bosserver`
  - `volserver`
  - `kaserver`
  - `pts`
  - `udebug`
  - `buserver`
  - `vos`
  - `klog`
  - `ptserver`
  - `upclient`
Symlinking stuff to a useful location

- I find it much easier to do this:

  ```bash
  sudo ln -s /Library/OpenAFS/Tools/root.server/usr/afs /usr/afs
  ```

  And use `/usr/afs/...` for commands
Where is other stuff located?

- KeyFile and server configuration files
  - /usr/afs/etc
- Database files
  - /usr/afs/db
- Logs
  - /usr/afs/logs
Vice Partitions

• Make them UFS
  – I'm betting the namei fileservicer makes some assumptions that HFS+ doesn't keep

• Symlinking
  
  ```
  sudo ln -s /Volumes/vicepa /vicepa
  sudo touch /vicepa/AlwaysAttach
  ```

  Repeat as necessary...

• Turn off indexing/Spotlight
  
  ```
  sudo mdutil -i off /Volumes/vicepa
  ```

  Repeat as necessary...
Initial DB Server Setup

- Start the bosserver with -noauth
  
  ```
  sudo /usr/afs/bin/bosserver -noauth
  ```

- Set the cell name
  
  ```
  sudo /usr/afs/bin/bos setcellname \
  server-2.irealm.awesmoe.org \
  irealm.awesmoe.org -noauth
  ```
Initial DB Server Setup

• Create database processes

```bash
sudo /usr/afs/bin/bos create server-2.irealm.awesmoe.org \
  buserver simple /usr/afs/bin/buserver -noauth
sudo /usr/afs/bin/bos create server-2.irealm.awesmoe.org \
  ptserver simple /usr/afs/bin/ptserver -noauth
sudo /usr/afs/bin/bos create server-2.irealm.awesmoe.org \
  vlserver simple /usr/afs/bin/vlserver -noauth
```

• Put server CellServDB info into client CSDB

```bash
sudo cat /usr/afs/etc/CellServDB >> \n  /var/db/openafs/etc/CellServDB
```

• While you are at it, make client ThisCell contain your cell name
Adding/Extracting AFS Service Key

• Again, there is no way to add service principals using the standard OS X Server administration tools

• Also you will want to limit the AFS service key to have only the “des-cbc-crc:normal” enctype
Adding/Extracting AFS Service Key

• kadmin:

```
addprinc -e des-cbc-crc:normal -randkey afs/irealm.awesmoe.org
ktadd -k /tmp/afs.keytab -e des-cbc-crc:normal
    afs/irealm.awesmoe.org
```

• ktadd will tell you the afs/realm KVNO

• At a shell prompt:

```
sudo /usr/afs/bin/asetkey add 5 /tmp/afs.keytab \
    afs/irealm.awesmoe.org
```

– my KVNO happened to be 5
Initial DB Server Setup

• Create initial pts entries

  
  sudo /usr/afs/bin/pts createuser -name alice \\
  -cell irealm.awesmoe.org -id somenumber -noauth
  sudo /usr/afs/bin/pts createuser -name alice.afs \\
  -cell irealm.awesmoe.org -id anothernumber -noauth

• You can leave out -id somenumber and -id anothernumber if you do not care what the user's pts number is

• Yes, you can do this with pt_util

  – Doing so is left as an exercise to the fatally insane
Initial DB Server Setup

- Add alice.afs to the system:administrators list
  
sudo /usr/afs/bin/pts adduser alice.afs system:administrators \
  -cell irealm.awesmoe.org -noauth
Initial DB Server Setup

• Add alice.afs to UserList

  sudo /usr/afs/bin/bos adduser server-2.irealm.awesmoe.org alice.afs -cell irealm.awesmoe.org -noauth
Restarting bosserver

- `/usr/afs/bin/bos shutdown \server-2.irealm.awesmoe.org -noauth`
- `ps auxww | grep bosserver`
- `sudo kill pid-of-bosserver`
- If you make the symlink to `/usr/afs`... as suggested, bosserver will start up automatically before the local AFS client service does
  `sudo SystemStarter start AFS`
Initial fileserver

• Create fileserver instance
  
  kinit alice/afs
  aklog
  
  /usr/afs/bin/bos create \
  server-2.irealm.awesmoe.org fs fs \
  /usr/afs/bin/fileserver \
  /usr/afs/bin/volserver \
  /usr/afs/bin/salvager \
  -cell irealm.awesmoe.org
Initial fileserver

• Create root.afs and root.cell

  sudo /usr/afs/bin/vos create \n  server-2.irealm.awesmoe.org \n  /vicepa root.afs

  sudo /usr/afs/bin/vos create \n  server-2.irealm.awesmoe.org \n  /vicepa root.cell
Initial fileserver

- Set permissions on /afs and /afs/irealm.awesmoe.org
  - This involves turning on the client w/o dynroot
  - Or various mounting tricks
  - Left as an exercise to the reader
- Create volumes
- Have fun
Final Thoughts: What Works

• It functions and seems stable
• I have gotten one other report of it not working
  – I have no details on this, however
Final Thoughts: Caveats

• vicep* mounting
  – I would prefer that my vicep* partitions actually be mounted at /vicep*, not symlinked from /Volumes
  – I also would prefer that my vicep* partitions not show up in the Finder, like conventional volumes do
  – And, well, I really don't want Spotlight indexing them

• If this is not supported, it should be

• If this is supported, better or easier-to-find documentation on how to do it is needed
Final Thoughts: Caveats

- OS X comes with a lot of stuff
  - Does my KDC really need iTunes?
Final Thoughts: Suggestions

• A way of adding user instances and service principals to the KDC that is integrated into the standard OS X Server administration tools

• Separation of being allowed to “administer this directory domain” and being given kadmin rights
  – If alice is allowed to administer the directory domain, she should be given an admin instance and be asked to supply a password for that to make any changes to the Kerberos database
Final Thoughts: Suggestions

- Integration of OpenAFS as a file service, just like Samba, etc., again, provisioned through the standard OS X Server administration tools
Final Thoughts

- There are caveats to providing Kerberos and AFS services with OS X
- But it does work
- Organizations with a heavy OS X deployment have access to high-quality, distributed, secure authentication and file services, and they can provide it on the platform they know
- A higher degree of integration of OpenAFS with OS X Server would make it even better
Thanks go to...

- Apple Computer for the loan of five demonstration Mac Minis
  - Ernest Prabhakar
  - John Hickey Jr
  - WWPM Product Placement Group
- Stomping Grounds in Ames, Iowa, for continuing to provide high-quality caffeine and free wireless network access
iRealm: Explorations in using OS X to provide AFS and Kerberos Services

Thomas L. Kula
Information Technology Services
Iowa State University

2006 AFS and Kerberos Best Practices Workshop

http://kula.public.iastate.edu/talks/afs-bpw-2006/