

How did I install AFS client on heppc16

Tomasz Wlodek

Abstract: I give instructions how to install AFS client software on Linux machine. Proceed at your own risk.

What is AFS: AFS (Andrew File System) is a program that allows you mount and access disks on foreign computers, in a way similar to NFS. They will appear as a directory on your machine. Unlike NFS, which is best suited for connecting machines locally, AFS is best when connecting machines in remote places.

What is AFS client: AFS comes in two parts: client and server. Client allows you to see files located at remote machines. Server allows others to see your files. I describe how to install client only. For the Atlas work we do not need server capability, and installing a server is much harder.

Where to learn more: There are a number of places on WWW you can look at. One is www.openafs.org. On FNAL site you can check <http://home.fnal.gov/~dawson/afs/index.html>. Both sites have extensive documentation.

Where to get AFS from: You can get the rpm's from open afs site, mentioned above, but they did not work for me. Other place is <http://home.fnal.gov/~dawson/afs/afsrpms.html>. Take the rpm which corresponds to your version of Linux and run rpm -install on it. You have to do it as root.

You can get the rpm which has been installed on heppc16 from /root/afs/ afs-fermi-3.6-10.i386.rpm

How to install it: do

```
mkdir /usr/vice  
mkdir /usr/vice/cache  
mkdir /usr/vice/etc  
mkdir /afs
```

```
rpm -install afs-fermi-3.6-10.i386.rpm  
once this is done AFS code will be installed in /usr/afsws. This will be your AFS area, let us call it AFSPATH.
```

Now do:

```
cp -rp $AFSPATH/root.client/usr/vice/etc /usr/vice  
cp $AFSPATH/lib/pam_afs.so.1 /lib/security/pam_afs.so.1
```

```
cp $AFSPATH/lib/pam_afs.krb.so.1 /lib/security/pam_afs.so.1
cp -p /usr/vice/etc/afs.conf /etc/sysconfig/afs
cp -p /usr/vice/etc/afs.rc /etc/rc.d/init.d/afs
```

The cell name: An important concept in AFS is “the cell”. The cell in AFS is more or less what NIS domain name is in yellow pages on Linux: a name of a group computers sharing files via AFS. The US Atlas cell name is `usatlas.bnl.gov`, Fermilab cell name is `fnal.gov`. You need to know two things: what is the cell name of the group of computers you want to connect to and what are the AFS server IP’s for this cell.

Let us assume that your cell name is *cellname* Open the file `/usr/vice/etc/ThisCell` and write *cellname* in it. Save the file.

Now you have to edit a file, which describes locations of your afs servers. The file is `/usr/vice/etc/CellServDB`. It should look more or less like that:

```
>usatlas.bnl.gov      #US Atlas Tier 1 Facility at BNL
130.199.48.32        #aafs01.usatlas.bnl.gov
130.199.48.33        #aafs02.usatlas.bnl.gov
130.199.48.34        #aafs03.usatlas.bnl.gov
>fnal.gov            # Fermi National Accelerator Laboratory
131.225.68.1         #fsus01.fnal.gov
131.225.68.14        #fsus04.fnal.gov
131.225.68.4         #fsus03.fnal.gov
```

Replace the cell names (after > sign) with your cell name; replace the ip addresses and names of your AFS server with the correct ones. (Ask the AFS administrator of the system you try to connect to what they are).

Create some important links:

Those links are needed for AFS to run:

```
ln -s /lib/security/pam_afs.krb.so.1 /lib/security/pam_afs.so
ln -s /afs/cellname/afs36/i386_linux22/ /usr/afsws
```

Modify the file with login permissions: to the file `/etc/pam.d/login` add the line

```
auth sufficient /lib/security/pam_afs.so try_first_pass ignore_root
```

so that the file looks like:

```
##%PAM-1.0
auth    required  /lib/security/pam_securetty.so
auth    required  /lib/security/pam_pwdb.so shadow nullok
auth    required  /lib/security/pam_nologin.so
auth    sufficient /lib/security/pam_afs.so try_first_pass ignore_root
account required  /lib/security/pam_pwdb.so
```

```
password required /lib/security/pam_cracklib.so
password required /lib/security/pam_pwdb.so nullok use_authok md5 shadow
session required /lib/security/pam_pwdb.so
session optional /lib/security/pam_console.so
```

Bypass UTA security: UTA disables pinging addresses at remote sites. AFS scripts are not aware of that and when they cannot ping remote site they think that AFS servers are not available and stop. For that reason you have to modify AFS script: open the file `/etc/rc.d/init.d/afs`, find lines

```
findServer
if [ $? -eq 0 ] ; then
and replace them by
#findServer
if [ 0 -eq 0 ] ; then
```

The reason for this is the following: AFS script tries to ping server, to see if it is alive. Since UTA disables pinging, it thinks that AFS server is dead. When you comment out the above lines you disable this check, and the AFS starts to work. I know, this is a brutal fix, but what can I do.

Start AFS: As root do

```
/etc/rc.d/init.d/afs start
/sbin/chkconfig --add afs
```

The first line will start AFS daemons. The second one will add afs daemons to startup scripts.

You can stop afs when needed by doing (as root):

```
/etc/rc.d/init.d/afs stop
```

Acknowledgements: Many thanks to Troy Dawson (dawson@fnal.gov) for his help.