Home directory in AFS

AFS Home Directories: Security Issues

SLAC has traditionally encouraged a policy of open exchange of data and programs in its computer systems. However as the Internet has grown and applications have increased in complexity, this policy needs some updating to provide the proper level of computer and data security.

User home directories in AFS have historically been wide open for read access from other users at SLAC. For example:

```
fs listacl ~vanilla
Access list for /u/sf/vanilla is
Normal rights:
  system:slac rl
  system:administrators rlidwka
  system:authuser rl
  vanilla rlidwka
```

The problematic entries are `system:slac` which means any machine in the SLAC ip address ranges and `system:authuser` which means any one in the world with a SLAC AFS token. While `rl` only allows reading and listing, there are many applications that assume a different file security model than the per directory one that AFS supplies. This causes problems when the application assumes that setting a file with unix permissions of `-rw-------` for user read/write only makes it reasonably secure, when in fact it may be readable by many other users.

In the past, OCIO has gone to some lengths to improve the security of specific apps, (ssh, vnc , X11), but as the number and complexity of applications increases this simply becomes unmanageable. Starting on March 18, 2009, specific directories that have known security issues will have the `system:slac` and `system:authuser` permissions removed. SCCS has been doing this for .ssh and .vnc directories for several years, this just expands the list to . mozilla, .mysql and .gaim. The directories .pgp and .gnupg were added in March, 2012. Other directories will be added as deemed appropriate.

Note that when a new account is created, the subdirectories `private`, `mail`, and `Downloads` are pre-created with more restrictive ACLs, which should meet expectations for privacy.

In addition, we would encourage you to start tightening down the AFS ACLs on your own as much as possible. In particular, for any application specific subdirectories in your home directory that may contain private data, remove the troublesome ACL entries. In order to do this with the minimum possible disruption, OCIO has provided a tool called `batten` to automate as much of this as possible. Please consider using this tool in at least it's minimum mode to secure your home directory.

`batten` has two different modes of use, do the minimum possible and do the most possible. In the minimum mode only directories that start with a dot have their ACL's modified. In maximum mode all directories, including the user's home directory get more restricted ACL's. `batten` in maximum mode also moves the user's `.k5login`, `.forward` and `.procmailrc` into a `system:slac` readable directory and makes symlinks to allow ssh login and email forwarding to continue working.

Using batten

As of 2/16/2022, /usr/local/bin/batten is still available on rhel6-64.slac.stanford.edu. By default, it does nothing put print out a list of the commands it would run. Once it prints out a list you are happy with, you use the `-g` option to actually execute those commands. For example:

```
rhel6-64> /usr/local/bin/batten

#!/bin/sh
# batten is running in test mode.
# This output can be edited and feed to /bin/sh
cd /u/v/vanilla
mkdir .system
/usr/afsws/bin/fs setacl -dir .system -acl system:anyuser none system:authuser none system:slac rl
/usr/afsws/bin/fs setacl -dir .openmpi -acl system:anyuser none system:authuser none system:slac none
/usr/afsws/bin/fs setacl -dir .mozilla -acl system:anyuser none system:authuser none system:slac none
/usr/afsws/bin/fs setacl -dir .dotfiles -acl system:anyuser none system:authuser none system:slac none
/usr/afsws/bin/fs setacl -dir .hepix-scs -acl system:anyuser none system:authuser none system:slac none
/usr/afsws/bin/fs setacl -dir .metacity -acl system:anyuser none system:authuser none system:slac none
/usr/afsws/bin/fs setacl -dir .gnome-desktop -acl system:anyuser none system:authuser none system:slac none
/usr/afsws/bin/fs setacl -dir .gnome -acl system:anyuser none system:authuser none system:slac none
/usr/afsws/bin/fs setacl -dir .eggcups -acl system:anyuser none system:authuser none system:slac none
/usr/afsws/bin/fs setacl -dir .environ-scs -acl system:anyuser none system:authuser none system:slac none
/usr/afsws/bin/fs setacl -dir private -acl system:anyuser none system:authuser none system:slac none
```

The best way to use `batten` is to tweak the command line options until you are satisfied with the results and then use the `-g` option to actually execute the commands. This will provide a "back out" file called `batten.revert` in your home directory that will undo all the commands if needed. For more detailed information see the `batten` man page.

[This page copied from an old web page created by Booker Bense. Copied 2020 March 31.]