SSH and Shared Service Accounts

(Copied from an old web page. Needs clean up.)

SSH and Shared Accounts

Previously SLAC used a locally customized version of SSH that supported forwarding AFS tokens during login. Unfortunately, the latest versions of OpenSSH make maintaining these set of patches and keeping our software current very difficult. We are switching to a new version of ssh that supports Kerberos TGT forwarding and using this forwarded ticket to obtain an AFS token at login. This will make many things much easier and simpler in the future, but leaves us with a difficult transition period as the behavior of one of the most commonly used tools changes.

Accessing Shared Accounts

The new version will require you to use ssh version 2 to take advantage of the TGT forwarding and automatically get an afs token when logging into a shared access or role account. Version 1 rsa key based connections will continue to be supported until sometime in December 2007, but only a GSSAPI or kerberos login will get you an automatic AFS token on login. Due to DOE computer security requirements ssh version 1 will have to be phased out by 2008.

Using SSH to access a shared account

Once the .k5login file has been created for the shared account. Users simply use these two commands to access the shared account.

> *kinit userid@SLAC.STANFORD.EDU*
> *ssh sharedid@machine.slac.stanford.edu*

Automatic update of .k5login files for shared Accounts

Shared accounts will need to be handled differently, rather than using the ~/.ssh/authorized_keys file to control access, you will want to use the ~/.k5login file to list the authorized users of the account. The ~/.ssh/authorized_keys method will still continue to work, but you will have to kinit after login to obtain an AFS token. Maintaining the .k5login by hand will be very awkward as the file must be owned by the shared account and have specific permissions (0644) in order to work properly. SCCS has created an automated process to allow groups to manage the .k5login file effectively. For each shared account you will need to identify a PTS group that contains all the users that are allowed to log in to the shared account. You will then need to contact unix-admin with both the shared account name and pts group. This will be added to the config file and then an automated process will track the entries in the pts group and update the .k5login file at regular intervals. All known current shared account owners will be contacted by SCCS to provide this information.

Here is typical example, in the past SCCS standard practice has been to set up two pts groups for every shared account. If the account was

> *foobar*

The corresponding pts groups would be

> *foobar:owner-foobar People that can add/delete users from foobar:foobar and ssh into the account*
> *foobar:foobar People that can ssh into the account*

In the new system you would no longer need to maintain ssh keys for the people in foobar:foobar, but simply use the pts command to add/delete them from the group.