**ACSE CRUISE - COMPUTER SYSTEM STUFF**

**SSH ISSUES**

To avoid use of passwords with ssh and scp can set up public/private key pairs for authentication, on the system you wish to connect FROM run: ssh-keygen and just hit return to all questions. This will generate 2 files in $HOME/.ssh: id\_rsa and id\_rsa.pub. Copy id\_rsa.pub to the remote system you you wish to connect TO, and append it’s contents to $HOME/.ssh/authorized\_keys.

You can now ssh from the account that created the keys, to the account on remote system to which they were copied without needing to give a password...which means we can automate scp copying of files.

NB. Two caveats for use with the moxa loggers:

1. The version of ssh-keygen on the moxas needs ssh protocol version explicitly defining...but shouldn’t need to ssh out of moxas anyway.

2. The Moxa UC7420s will allow you to copy the public key into /root/.ssh (/root being the home directory of the root user), however, the UC7112 and IA241 units have large parts of the filesystem – including /root set as read only, so we can’t set up password-less ssh login to that account.

To enable automated scp from these systems I’ve created a new ‘acse’ user account with home directory /home/acse under which the .ssh directory is created and authorized\_keys file created.

The authorized\_keys file on all the moxas contains the public key of admin@DiskStation\_1. Backup scripts are run from the diskstation to scp files into /volume1/shared/RAWDATA.

**BACKUPS ON DiskStation\_1**

Changes to DiskStation\_1 config:

* The ‘user home’ option has been enabled for users on DiskStation\_1 (from user admin panel) – this enables a unix home directory, which is created under /var/services/homes.
* The SSH option has been enabled from control panel | terminal menu. This allows an SSH login to provide direct terminal access to the underlying linux system

Creating new user

adduser username

and respond to request for password

user = ‘acse’ : pw = ‘acsebackup’