Contents

This is a series of documents written at different dates about aspects of the NMR service network operations, data backup, and related matters.

The final paper version ought to include discs ??

Software maintenance relevant to the NMR support systems

General

These are notes about the software which may need to be dealt with to support the NMR spectrometers, rather than the spectrometer computers themselves. I have tried to follow a convention used in Bruker and other books to illustrate Linux commands and so on by a different type face. For example, a user's data is stored in

```
/opt/topspin/data/<username>/nmr
```

where `<username>` means a name like seg and the brackets <> must not actually be put into a command.

NMR network

The spectrometers and several workstations are on a small private network, as is 1 network port of each of the NMR file server units. The second network port of the file servers links to the standard college system: the whole idea is to minimise the possibility of intrusion into the spectrometers from outside while allowing data out to the main network. Any serious change to the network will probably need ICT cooperation. We are probably stuck with the phrase NMR file servers, but these machines do not do what most users expect - there is little or no data stored in them, and their function is more of a firewall plus a switching device to connect users to the required data, which is normally on the spectrometer computers.

Command files

Command files or shell scripts are simply text files of conventional Linux/Unix commands which can be executed by entering the file name as a command. There are conventions for file headers (see any of mine) and a file must be made executable before it can be used by e.g.

```
chmod 755 <filename>
```

An important feature in the use of these files is the idea of passing "arguments" into a file. When the file is constructed, any piece of text can be replaced by the dummy items `$1` to `$9` (and probably more). If the file is then executed by something like

```
filename arg1 arg2 arg3
```

then the text string arg1 (which can be anything in principle) replaces each and every occurrence of $1. The strings arg2 and arg3 replace $2 and $3 and so on. NOTE that if the numbers of arguments and $variables do not match then strange things can happen.

This feature is used in much of the back up software to pass machine codes, month codes, user IDs and so on into the files. Reading some files should make this clear.

Multiple disc drives

Linux/Unix always creates one file structure: any extra discs are "mounted" on dummy directories on the main disc (the one which contains /boot). For example, in the workstation Hpsys3, the second and third disc drives are mounted on `/opt` and `/srv` .

Adding users to the NMR systems

General

NMR users of the various IC spectrometers are NOT users in the standard Unix/Linux sense - these user identities are decribed as "additional users of the Linux user nmr" and are a feature of the Topspin program and nothing else. The only effect of changing NMR user is to choose an appropriate data directory and save accounting information correctly.

Users can be added/removed by the configuration section of IconNMR. (In IconNMR's terms, they are additional users of the Linux user nmr.) Alternatively, the appropriate data directory can be set up via a shell window. A user's data is stored in

```
/opt/topspin/data/<username>/nmr
```

(IconNMR creates this when the user is first used)

NOTE that I have used a common convention 2 lines above to illustrate a Linux command or directory: `<username>` means a name like seg and the brackets <> must not actually be put into a command.

Since all of the contents of

```
/opt/topspin/data
```

are exported by the NFS system, no other changes have to be made at the spectrometer computer when a user is added.

Changes at chnmr3

The original design of chnmr3 was made as simple as possible. Therefore, there are no changes to be made here.

Changes at chnmrserv

The Samba server in this system is set up to expect login codes. This suits Apple systems better, and works with Windows Vista. The original intention was to use the same login codes as the open-access spectrometers, but this may not have been the best idea: once a login is recognised, the data of any user can be read and changing this looks difficult.

The Samba server will only allow users who are already registered as users of the Linux system in the workstation, so the first step is to create logins which cannot actually log in to a normal terminal session, to preserve security. (This is almost as daft as it sounds.) Once this is done the relevant section of the Samba software will allow the creation of passwords which only allow access to the Samba service.

1) Creating crippled Linux users and passwords
The first step is to create the Linux user IDs which will be used to get into Samba.

Open the menu System > Administration > Users and Groups (at some point the root password will be requested) This provides a table in its own window.

Choose Add User and fill in the pop-up table, selecting the shell `/sbin/nologin` and choosing NOT to make a home directory and to put the user in its own group. The new user adds to the end of the table at once.

Next, in a terminal window become root, and then enter the command

```
passwd -l <new-user>
```

This blocks any use of the password to log in. The effect can be confirmed by the command

```
passwd -S <new-user>
```

which prints a confirmation message.

2) Adding Samba passwords.
This is all done by the command `smbpasswd`, which must be run in a terminal window as root. Change directory to `/etc/samba`

For each new user, enter

```
smbpasswd -a <user-name>
```

This will ask for the new password twice.

This command has many functions - see manual page or `smbpasswd -h`

Changes to data back up systems.

For each new user, it is necessary to make an alteration to the backup software. This software does suffer from having grown in stages and suffering some significant changes of plan following tests: the basic problem is that I found no tidy way of copying just the new data without using the user's names.

There are 2 computer systems which are independent of each other. Hpsys1 collects data twice a month into an image of the spectrometer's data directory so that it may be copied off onto a DVD when enough has collected. (I then remove the data from its hard disc to keep making DVDs as simple as possible.) Hpsys3 has (now) 3 separate hard discs: one contains the Linux system and so on, the other 2 are to contain two copies of the spectrometer data. This computer collects changes once a day. Both of these systems rely on the system service `cron`, which very regularly checks the `crontab` file and runs tasks at specified times.

Hpsys1
This system has a version of the usual Bruker setup supplied by Julian Tombs and almost everything is run as the user nmrsu.

To add a new user to the structure run the command `adduser` as

```
adduser <newID>
```

This puts the user's ID in the correct places in the local file system.

Then alter the backup task files to incorporate the user. Probably the simplest is to extend the file

```
/home/nmrsu/bakfile.1
```

by looking at the existing lines and copying. (There are 5 active backfiles.) Once this is done, the corresponding file (in this case)

```
/home/nmrsu/delkak.1
```

should also be edited - this is used to remove data once it is on DVD.

Hpsys3
This is largely the same, except that it is a standard Red Hat system with the users root and serv. To allow as many files as possible to be copied directly from the computer hpsys1, I made the directory `/home/nmrsu` (which belongs to user serv) which contains many of the files.

The command

```
adduser <newID>
```

makes the relevant directories (on 2 data discs this time) and the actual backup process needs extending by for example modifying

```
/home/nmrsu/bakfile.1
```

again - but in this case there are more lines to add so that data is stored on both the /opt and /srv disc drives. There is no deleting tool - this is supposed to be a permanent archive.

This system is about 30% full with over 3 years worth of data stored on the data disks. In the event that data needs to be retrieved it could be copied to a CD (but not I think a DVD) or copied elsewhere by sftp, probably as a tar file. Alternatively, NFS could be set up and the data exported into the local network including the server(s). None of this has as yet been tested.

The NMR private data network.

History.

The private NMR data network was set up at a time when our systems were scattered over the buildings and there was a demand to deliver data "to users' desktops".

It was considered completely impractical to connect our then existing spectrometers directly to the college network and not very likely we could set up something in each NMR area to isolate the spectrometer. It had recently become possible to program the network switches to set up closed groups of connected sockets - a VLAN. (Virtual local area network.) Network staff set up such a system linking the two JEOL 270 areas, the 300MHZ on level 8, and the DRX-400 area in 004. This private network was used to link the original set of spectrometers (2 modernised 270MHz and the two DRX systems) to a single server, which isolated them from the main network while supplying read-only data to users in Chemistry. The network also permitted some remote management of the spectrometers.

The refurbishment project left all the new or remaining equipment in 004 and 005 and the VLAN is now concentrated in 2 switch units in the data closet on the mezzanine corridor. Other ways of making the private network would now be practical and may become necessary in future.

Computer registration.

A new requirement (October 07) is that all equipment connected to College network ports MUST have its hardware address (MAC address) registered in a network database, or service will be refused. This applies to all items connected to the VLAN. Things like servers need both interfaces registered. It should be possible to do this through local ICT staff.

IP numbering.

It looked fairly inconvenient to modify the numbers used by the JEOL Delta hardware then in use: everything else was fairly straightforward. I therefore standardised on the number series formally assigned to JEOL USA:

192.160.103.x

where x can nominally be 2 to 254. In a simple closed network using this type of numbering, the netmask for each device should always be

255.255.255.0

It is important that messages using these addresses are confined to the private network.

Maintenance.

In its present form the hardware is wholly maintained by others.

The only attention required by the network is to assign unique IP numbers to devices and maintain files (normally /etc/hosts) in **each** machine linking names and numbers. These do not have to be identical in each machine, and the name used in the hosts file for another machine does not have to be its official hostname - it can be shorter or more convenient. Bruker gave some of the workstations long and unmemorable names, so I have not used these in most host files. BEWARE - these names are embedded in the Topspin software licences - do not try to change them !

I have tried to keep a basic block of the same information in the host file of **each** machine - not always successfully - with the machine's entry commented out and put at the head of the file.

Some of the computers listed in the host files are no longer on line and a few have been discarded. I have started pruning these out of lists, but quite a number of the absent machines are still on shelves and not quite dead yet.....

A peculiar bug: Topspin will not operate correctly in a computer which does not have a "fully qualified" hostname. Machines which run Topspin therefore have an entirely bogus full name listed in their host file - e.g. hpsys1.ch.ic.ac.uk

These names are bogus and **must not** be used on the main college network !! There is normally no need to use these full names - they are purely for Topspin's benefit.

Example host file.

This is from hpsys1 - see notes at the end of the listing.

```
[nmrsu@hpsys1 nmrsu]$ cat /etc/hosts
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1       localhost.localdomain   localhost

#149.236.99.1   ASP_ST2     Spectrometer control
#149.236.99.99  spect       Spectrometer control

192.160.103.160 hpsys1.ch.ic.ac.uk      hpsys1

# new unified nmr net
#
192.160.103.20        Ubox
192.160.103.28        DRX4
192.160.103.42        winpc
192.160.103.44        nmrs2
192.160.103.50        ch3
#
192.160.103.60        IRIS
#
192.160.103.110        eclipse
#
192.160.103.130        eclipse2
#
192.160.103.135        delta1
192.160.103.136        delta2
192.160.103.137        delta3
#
192.160.103.99         pdp

#
# new systems 2005-6
#
192.160.103.150        a500
192.160.103.151        a400
192.160.103.152        b400
192.160.103.153        c400
192.160.103.154        d400

#
# new workstations 2006-7
#
#192.160.103.160       hpsys1 (see above)
192.160.103.161        hpsys2
192.160.103.180        hpsys3

192.160.103.200        serv2
```

This list does not include the HP 4250n network printer, which is assigned the address

```
192.160.103.10
```

Non-obvious names

Ubox and IRIS are an old SGI and the 300's O2 (both off line in room 006).

winpc and nmrs2 still exist.

pdp was the old ex-BP 270 - scrapped.

ch3 and serv2 are the private network ports of chnmr3 and chnmrserv respectively - it is simpler to use different names for each port of these machines.

File Servers for NMR data distribution.

Basic concepts.

The private NMR data network is linked to the main College network by our NMR file server(s). The server hardware is a fairly ordinary PC with 2 network ports - one connected to the private network, the other to the main system. All the spectrometers export their data to the private network as NFS volumes: the server computer mounts these and all the data becomes part of its file system. (At the present time some computers removed from spectrometer service are also still exporting data to the server.) The utility Samba (supplied with Linux) runs in the server and provides a read-only file service to Chemistry Department computers via the main network - this is controlled by limiting the range of IP addresses to which service is supplied. It is important to note that none of the current spectrometer data is actually stored in the server itself - it is fetched from the spectrometers when requested.

The original server is chnmr3.ch.ic.ac.uk - this is a fairly ancient PC with a decidedly obsolete version of Red Hat Linux.

The newer server is chnmrserv.ch.ic.ac.uk - this is an HP xw4300 workstation supplied through Bruker. It runs under ICT's version of Red Hat workstation version 5 and uses the appropriate more modern version of Samba.

Operations.

The servers do not in general need any sort of routine attention - at this moment chnmr3 has been running 82 days since last reboot and will often run much longer than that.

When a reboot does become necessary the procedure is exactly the same as any of our other Linux-based computers. The system will automatically attempt to mount all of the spectrometer NFS volumes during the reboot.

Once the system is up, it is **necessary** to log in, check that data is mounted, and start Samba. The command `df -k` will list all the file systems which have mounted. For example

```
[serv]df -k
Filesystem           1k-blocks      Used Available Use% Mounted on
/dev/hda5             14674280   2434300  11494568  18% /
/dev/hda1                23302      2476     19623  12% /boot
d400:/opt/topspin/data
                     133546340   7289196 119473300   6% /disk1
DRX4:/v/data           8919156   4791648   4127508  54% /disk3
delta1:/home          15583168  11981944   2809624  82% /disk4
a500:/opt/topspin/data
                      56820980  16356436  37531644  31% /disk5
a400:/opt/topspin/data
                      58649344  19522900  36147192  36% /disk6
b400:/opt/topspin/data
                      56820980  20892332  32995748  39% /disk7
c400:/opt/topspin/data
                      56820980   3419132  50468948   7% /disk8
[serv]
```

Problems **will** occur if one of the spectrometers is not running its NFS export correctly. The command `df -k` may not even complete correctly and spectrometers which are listed in the file `/etc/fstab` after the faulty system usually will not mount correctly. The immediate remedy is to switch to root (`su` command) and try the command `mount -a`. If the NFS file systems are still not all mounted, the next step should be to check the spectrometers.

It may be necessary to get the system running with a spectrometer out of action. The short term solution is to mount the missing NFS volumes individually with commands like

```
mount -t nfs  <spectrometerID>:/opt/topspin/data   /<relevant directory>
```

where the spectrometer ID and relevant directory can be quickly checked by looking in the file `/etc/fstab` . If there is likely to be serious downtime on a spectrometer, it may be better to edit `/etc/fstab` (as root) and comment out the line for the faulty system before trying the command `mount -a` again.

Once the NFS data is mounted, start the Samba service (as root) by the command

```
/sbin/service smb start
```

which should produce 2 messages confirming the start of the smbd and nmbd daemons. It is then possible to log off.

The server does not usually notice if a spectrometer is shut down and restarted once Samba is running. Additional root-only commands for control of Samba are

```
/sbin/service smb stop
/sbin/service smb restart
```

which should generate appropriate messages.

On rare occasions there are genuine reports that the server is no longer accessible from machines where it used to be. Experience shows that it is not sufficient to restart Samba - it is necessary to reboot the Linux, check the NFS mounts, and then restart Samba. These events are **rare**.

There are no other significant operation (as opposed to maintenance) tasks.

Maintenance.

Maintenance splits into 2 areas - the OS and Samba.

The Red Hat OS should require very little attention unless matters go very badly wrong. In chnmr3 the system is a very old version (7.3 ?) and no tools are available to repair major problems. A really major failure or disk disaster might be terminal.

The modern system in chnmrserv should be supported by ICT for some time to come - some help with major restoration should be available. This system is linked to an automatic update service which (I think) is only visible as root. It is worth logging in from time to time to see if anything is being offered - I have mostly seen security updates.

In extreme circumstances the procedure to start from scratch has been written up and is not really very complex if the relevant files are still available from CD. Very little more advice can be sensibly given.

The Samba file server may require attention in a number of areas, as follows.

(1) Alteration of IP addresses served and queries about access.

Both versions of the file server will only accept enquiries from specific ranges of IP addresses or individual addresses, specified in the file `/etc/samba/smb.conf` as “hosts allow”. These specifications are either in the form x.y.z. or are the actual IP address of a specific computer. In the first case x y and z are the first 3 groups of numbers for a range of addresses - e.g. 155.198.226. - these are assigned to level 1 in RCS1. The objective is to limit access to users in Chemistry - there is no other form of security. I am very doubtful we will be adding many more IP ranges, but it is easy in principle. Note that the same range of addresses

appears in the smb.conf file as part of "remote announce". This command overcame some problems in the early stages of the project - leaving it in seems to do no harm.

A practical aside: these lines are very long. Chnmr3 does not mind, but chnmrserv didn't like them, so they are split by the Unix convention of \ at the end of one line and continued on the next. The list includes 127. for the "localhost"

It is possible that these commands might get reconstructed in the form of ranges of machine names in future.

There are many queries about failure to access the server: almost always these come down to an unsuitable IP address of the computer concerned. A few are due to unusual operating systems on the computer - XP home edition (which is not supposed to be used on ICT network) or Vista - which only works with the newer Samba version in chnmrserv.

Many of the complaints involve a disbelief that we can or should make any control of access - "but I can use all the other servers" is a popular line. Usually it turns out that access is being attempted from home, a college residence, an insecure wireless connection, or a laptop which has not been properly registered for the college network. (Its owner may not be aware or believe this is necessary.)

In practical terms, all that can be done is to get the user to check the IP of their machine with the command ipconfig. If this is unfamiliar but they insist it is in Chemistry then check with ITC ! Be particularly careful of wireless connections - many of these seem to be college-wide. I have in general refused to support them. Also get them to ping the server from their machine. Any complaint that it worked for a particular machine until recently is usually incomplete - have they moved it and been given a different IP, or have they switched to a wireless connection ?

Sadly most of these complaints cannot be resolved to the user's satisfaction.

Note.
The present list (from Nick Davies / ICT) of IP ranges used within Chemistry is as follows:

155.198.36	CHM teaching
155.198.224	CHM levels 3 / 4
155.198.225	RCS1 level 3 / CHM level 7 / 8
155.198.226	RCS1
155.198.228	CHM level 2
155.198.231	CHM level 0
155.198.232	CHM levels 5 / 6 / 7
155.198.234	CHM level 1

I assume that CHM implies C1 and C2.

The Samba definitions include a couple of specific IP addresses - these are for Ed Smith and friend.

(2) Modification of Samba shares - the data directories made available.

There are two type of share - those containing local content and those derived from other computers via NFS. The latter can only be changed from the spectrometers !

Shares with local content can be modified by moving files into or out of the appropriate directories - make sure that the file permissions make them readable by others. There is a catch in preparing text files to go into these directories: all varieties of Unix use a different convention for line ends to all varieties of Windows - it is too easy to make a file which is unintelligible on Windows. The Bruker-configured Linux systems do have an editor named kwrite which can be set in a menu to produce Windows-style line ends: most of the systems also have the Open Office package which can produce quite satisfactory *.doc files. Otherwise prepare material on a PC and transfer it.

A share can quickly be removed by commenting out all the lines of its definition in the file smb.conf. A new share definition is easily added to the file by basically copying an existing similar one and making sure that the directories it refers to actually exist. I have always kept previous versions of the configuration in files named smb.conf.number so it is easy to go back to a previous version. For chnmr3 nothing else is required, but for chnmrserv there are extra complications caused by the SELinux security package. Very briefly,

(1) the directory referred in the share definition to should be in `/srv`
(2) the command `chcon` **must** be used to set access permission - see the separate description of setting up chnmrserv for details of this.

Failure to observe these requirements will cause large numbers of SELinux messages and the new share will not be accessible.

Changes to smb.conf **only** take effect when Samba is stopped and started by appropriate commands. (See operations section.) If changes do not have the required effect, try stopping Samba and running its test program `testparm`. This will report what the smb.conf should actually do and pick up common errors.

If it is difficult to put new information in a directory in `/srv`, it is possible to put a link file in a new directory in `/srv` which points to information elsewhere. This is covered in the description of setting up chnmrserv.

(3) Modifying the data which is obtained via NFS.

If the number of external computers from which Samba distributes data is to be modified, then the file `/etc/fstab` will have to be changed to mount the data from a new machine or stop mounting data from a retired one. To add a new NFS system, a new empty directory has to be created to act as the "mount point" - preferably in the same location as the existing ones. (Read `/etc/fstab`)

Before making any permanent changes, it is worth testing that a new NFS volume will work correctly by doing a temporary mount with a command like

```
mount -t nfs  (computerID):/datapath   (empty directory)
```

and checking the required data is visible. Once this works correctly unmount the new NFS volume before proceeding.

In the case of chnmrserv, the new mount point directory must next have its attributes set by the `chcon` command (see above) while nothing is mounted to it and before inclusion in Samba. (Otherwise thousands of error messages.....)

Make appropriate changes to `/etc/fstab` (much can be guessed from the existing file) and check it works as intended by `mount -a`. Make appropriate changes to smb.conf, stop and start Samba, and check the results are as expected.

(4) Password access to chnmrserv.

The Samba service on chnmrserv has been modified to require a username/password authentication before allowing access. (See also the notes on chnmrserv itself.)

The system set up will only need attention to add new user groups. The login username is the abbreviation used on the 400s (e.g. agmb, vcg, mimi) and the password is the same as required for Topspin in the 400s. Samba passwords are created by the command `smbpasswd` - unfortunately this command only works for usernames already created in Linux.

The first step is therefore to create a completely crippled Linux login as root using the users and groups panel. The new user is given the shell `/sbin/nologin` and is not allowed a home directory. Once this is done the password is blocked by the command

```
passwd -l username
```

which makes the password indecipherable and further blocks login. The command `smbpasswd` can then be used to assign the password of this new user.

Appendix 1 - setting up spectrometer NFS to export data.

All Bruker systems already have NFS running to provide the link to the spectrometer control - usually spect. It is only necessary to add to the definitions of exported data.

NFS is set up by the contents of the file `/etc/exports`. In most of the Bruker configurations there is a menu entry leading to a control panel tool, but the file can be edited directly. In either case, the information required is the path to the directory to be exported, the possible destinations, and options such as read-only. When the remote machine "mounts" the export to an empty directory, that directory then contains the contents of the exported path. In our applications, when the path /opt/topspin/data is exported, the directory to which it is mounted contains the username directories.

The export destination may be set to * meaning anyone as this is to a small closed network. The file for the new DRX-400 system d400 is typical: the final line was added to export data

```
[nmrsu@d400 ~]$ cat /etc/exports
#
/usr/diskless/dl_usr              spect(sync,ro)
/usr/diskless/clients/spect       spect(sync,rw,no_root_squash)
/opt/topspin/data                 *(ro,sync)
[nmrsu@d400 ~]$
```

Setting the exports file is only part of the job: it is also necessary to alter the network Security and Firewall control panel via one of the system menus. NFS will not be exported correctly until the external interface (eth0) is set to be trusted. I suspect this significantly reduces security - more reason for not being on the main network. After making these changes, it may be necessary to reboot to get the required effects.

Making a system which is not already running NFS export data will probably mean starting this service - usually via a control panel of some sort. A quick check on the state of NFS is

```
ps -ef | grep nfs
```

which should show several instances of nfsd if the service is running.

Added note July 09

Installing Topspin 2.1.n caused modifications to the `/etc/exports` file (adding extra lines) but the same modification is still relevant.

A brief description of automated NMR data backup in its present form.

Given that the NMR computer discs have a finite capacity and that discs can fail, I set up a system to automatically copy data to another computer ready to copy to DVD. Later I added a similar system to copy the data to large hard discs in yet another computer. The DVDs only get made when enough material has been collected, so are usually more or less behind the current date. The hard-disc based version gets updated daily - this at least means only a day's data will be lost if spectrometer discs fail.

For each new NMR user, it is unfortunately necessary to make an alteration to the backup software. This software does suffer from having grown in stages and suffering some significant changes of plan following tests: the basic problem is that I found no tidy way of copying just the new data without using the user's names.

The 2 systems are independent of each other and run in HP xw4xxx computers in room 005. Hpsys1 collects NMR data twice a month into an image of the spectrometer's data directory so that it may be copied off onto a DVD when enough has collected. (I then remove the data from the Hpsys1 disc to keep making DVDs as simple as possible.) Hpsys3 has (now) 3 separate hard discs: one contains the Linux system and so on, the other 2 are to contain copies of the spectrometer data. This computer collects changes once a day. Both of these systems rely on the system service `cron`, which very regularly checks the `crontab` file and runs tasks at specified times.

Hpsys1

This system has a version of the usual Bruker Red Hat Linux setup supplied by Julian Tombs and almost everything is run as the user nmrsu. This system includes the package K3b which provides a convenient tool for making DVDs. (K3b is supposed to also make CD-R discs but the version on Hpsys1 fails to do this.) The backup system is based on regular `cron` calls to the file `callbak` - these specify the month to be backed up. File `callbak` in turn calls the file `tsbak` once for each spectrometer. File `tsbak` contains the definition of the current year at the head of the file - this needs changing in January after the last of the December files are collected. This file in turn calls a series of files named `bakfile.n` which do the actual copying. (I originally planned to run them on different nights but this proved unnecessary.) All these files copy data into directory structures like

`/opt/temp/<machine-name>/data/<user-name>/nmr`

When full enough, the directory `<machine-name>` and all its contents get put on DVD. The size of the data waiting to go to DVD can be checked by the command `baksize` in a terminal window - a disc will take something like 3.5-4 Gb of data. (This probably needs checking every 2 weeks or so.) The machine names used in all the file-copying routines are 400A, 400B, 400C, 500, and D400 (the DRX-400).

I have written a separate note on making DVD/CD-R discs with our Linux-based HP machines.

The whole system writes a log file `bak.log` (read with `more`) - at intervals this is copied to `obak.log` and a new file started.

If a new user gets missed for a while it is possible to put extra one-off calls to the required months onto the end of the `crontab` file to sort things out - these should re removed or commented out once the job is done..

Once data has been put on a DVD, the file `delbak` removes the specified data from the directories - this uses files named `delfile.n` (which must be altered to keep in step with `bakfile.n`). The file is run in a terminal window - commands like

```
delbak 500 Apr 2009
```

will delete data for the specified machine-month-year.

Adding a new user involves 3 operations:

1) make new empty directories by using the command `adduser`
2) alter one of the `bakfile.n` files by adding 2 extra lines for each new user - copied from the existing ones
3) alter the matching `delfile.n` file by adding a new line for each user

More detail is in the note on adding new users to the NMR systems.

Hpsys3

This has a standard Red Hat system loaded from the disc set delivered by HP: the users are root and serv. As far as possible the routines from Hpsys1 were copied over and reused - to simplify this the directory /home/nmrsu was used for many of the files. (It belongs to the user serv) The NMR data gets copied to

```
/opt/temp/<machine-name>/data/<user-name>/nmr
```

and also to

```
/srv/temp/<machine-name>/data/<user-name>/nmr
```

and kept. In practice `/opt` and `/srv` are the mount points for two 700Gb discs used only for data.

In the event that the backed-up data is required, there are several possibilities. These include:

1) write to CD (or DVD ??? - failed so far) or memory stick
2) make files into a tar file or files and send to a server by sftp - once there put into a directory exported by the Samba server. (The temporary data one ?)
3) set up NFS service on this computer, export the data to the NMR network, and modify the samba server(s) to export to the main network.

None of these routes has actually been tested so far ! For older data the DVDs made by Hpsys1 would probably be more convenient.

Adding users is essentially the same as for Hpsys1 - more detail in the note on adding users.

Required operator actions

The systems are pretty automatic but the following tasks are ESSENTIAL.

Both systems: Change the year definition in the first few lines of the file `/home/nmrsu/tsbak` in the first couple of weeks in January each year.

Make necessary changes to add new users to the backup files.

Hpsys1
Check the size of the collected data every month or so and convert to DVD when necessary - probably when between 3 and 4Gb of data are collected. Look at the file

```
/home/nmrsu/bak/log
```

from time to time to see if major errors are being reported. (If empty, look at obak.log)

It is usually more convenient to put data from just one spectrometer on each disc. Once data is backed up on DVD, clear out the directory with the command `delbak.` Doing this makes the preparation of subsequent DVDs much easier.

Hpsys3.

This is meant to be very automatic, but it does no harm to look at its bak.log file from time to time.

N.B.

It is probably worth rebooting this machine every 3-4 months: it is very stable but if left on long enough, the next reboot insists on running a full fsck check on all 3 discs and this will take a LONG time.

Making CD-R and DVD discs on NMR network computers.

General

These computers are all HP xw4xxx machines running various versions of Red Hat Linux. They can be divided into 2 basic groups: all but 2 have software configurations provided by Bruker.

Systems with the software package K3b

The more recent spectrometer and workstation computers came with this program as part of the OS: these are the 4200 (part of 400A) and the various 4300 machines. Only the newer machines have a drive which is supposed to make DVDs - the older units have a CD read/write drive and a read-only drive for DVDs.

K3b resembles a good many Windows utilities: it opens its own window which is subdivided into 4 sections. The top left shows the file system as a tree structure: clicking on a directory will show its contents at the top right. The lower part of the main window starts by showing various icons including “new data CD project” and “new DVD data project” - clicking one creates a tree at the left and an initially empty area at bottom right. Dragging folders/files to this area from the top right builds up the data which will go to the disc - a progress bar at the bottom shows total size. In principle a “project” can be saved to hard disc and reused. Once the copying is complete the button Burn creates a sub-window in which almost everything can be left at default, though it is sensible to find the disc title and change it to something meaningful. This subwindow contains another burn button which actually starts the process. Be aware that there is often a considerable delay before anyting obvious happens and the process is quite slow.

In practice Hpsys1 (and probably Hpsys2) refuse to make CD-Rs - only DVDs work.

It is possible to add a verify option to the process before actually starting: this compares the data on the disc with the original files but is VERY slow.

Other machines

This refers to the xw4100 machines on 3 of the spectrometers plus the others running standard (not Bruker) Red Hat. These do have a slightly odd tool on the desktop or easily accessible. The 4100s have an icon which opens to show a window with a burn command and pointing to a very odd directory. The icons of folders and files are copied into this window by means of the copy and paste commands of the Edit menu. They can then be burnt to a CD-R.

The other purely Red Hat macines contain a variation on this idea. The Computer directory opens to show an icon for the optical disc drive. Provided a blank CD-R is in this, it will open to show a similar window into which icons can copied and pasted from other graphical windows, and a button to burn the disc.

Apparently the newest versions of this should make DVDs as well, but I haven’t managed to acheive this as yet.

Unmounting optical discs

Linux/Unix always treats optical disc as being mounted by a specific user. They must be unmounted (right click icon) before they can be ejected.

Building up an NMR server using Linux and Samba.

This document tries to set out as plainly as possible the steps to set up a Samba server for our NMR facility starting from an empty machine - or one which has been so badly upset that the best thing is to start again from scratch.

The underlying concept is that the spectrometer computers and some other workstations are linked by a private and rather old fashioned network: the server or servers have 2 network interfaces, one to the private network and the other to the main network.

Fundamental steps.

- Install a modern and ICT approved OS on the server box - with luck some help will be available if this goes badly wrong.
- Ensure that the Samba software is present on this.
- Make changes to the server system to prevent anyone logging into it from the main network.
  This is essential to minimise the chances of unwanted external access to the spectrometer computers.
- Install a second interface in the server and configure to work with the private network.
- Get the system to mount the NFS volumes exported by the spectrometers.
- Copy in an existing version of the Samba file smb.conf
- Make any necessary changes, any changes needed for security features of the system, and test.

Installing the operating system and checking Samba software.

ICT will load an approved version of Linux onto a workstation and give the system a name. The machine must be ready to connect to the main network for this to be done.

In July 07 I had Nick Davies (ICT) put the current ICT version of RedHat Enterprise 5 onto an HP xw4300 64-bit workstation. This has many security features turned on and is supposed to be safe on the network. This system was given the name chnmrserv.ch.ic.ac.uk

As supplied, this uses the standard college log-ins (same as PCs etc.) The nominated custodian has use of the command sudo to perform root-type operations. The syntax is

```
sudo command
```

which asks for the user's password and then runs command with root privileges.

Current versions of this system have in the applications menu the item "Add/Remove software" This links to a server somewhere and can add Red Hat packages missing from the machine. Use its search option to look for samba: this shows all the possible packages and which ones are actually present. Very often the main part is missing and only a library and smbclient are installed. Mark the required package and load it - be SURE that all the items in the computer will have the same version number. The package swat does not seem worth having.

The approach in the future will probably have to be much the same, though RedHat may have moved on some more.

It might be better to ask for the server version of the software - but I have never tried this since I found an early version did not have the usual desktop interface.

Taking control of access to the server system

There are probably 2 distinct ways of doing this. In principle it is possible to use the files `/etc/hosts.allow` and `/etc/hosts.deny` to specify which external computers if any may log into the system, but I have never tried this and do not know if it would cause problems with the Samba file server.

I resorted to a brute force approach by what would probably be seen as a misuse of the sudo facility. Logged in as rns (the custodian), entering

```
sudo passwd
```

asks the user password and then lets you enter and confirm a new root password without ever knowing the old one.

Log in as root, and find “users and groups” in system admin sub-menu.

Make a new ordinary user - most options can be left as default.

TEST that both this user and root can log in properly !

Once this is done, log in as root, and go to the menu system>administration>authentication.

In the panel which appears, switch OFF the LDAP option, which controls the college logins. I also set the option that local users only need local authentication. This may have been overkill.

At this stage only root and your ordinary user can log in at the keyboard, and only root over the network.

Installing and configuring a second network interface.

In July 07 I used a cheap network interface card from RS, chosen because the catalogue claimed it was supported in Linux. This item was RS 505-1750 - described as a Belkin Gigabit PCI card.

Switch the PC off and install the card - might need help ?

Once this is done, connect the main network to the original socket and NOTHING to the socket on the new card, and restart the system. No error messages should be generated.

Before going any further, get a recent version of the /etc/hosts file from a system already on the NMR network. This should show the IP numbers of the existing boxes, of the form 192.160.103.x, and pick a new value for x. It is ESSENTIAL that this is not used anywhere else on the NMR network - the paranoid might use one of the existing machines to try to ping this number..... This range of addresses is assigned to JEOL USA and was used by the 270MHz machines (now scrapped) but it is not worth changing to a different series.

The configuration of the ethernet interfaces is controlled by files in `/etc/sysconfig/network-scripts`. File `ifcfg-eth0` controls the original interface and should not be touched: the file `ifcfg-eth1` will probably have been made similar automatically and needs to be edited as root.

The line which must be altered is BOOTPROTO= and the lines which must be added are NETMASK= and IPADDR=.

N.B. The line HWADDR must not be altered in any way !

The essential lines in the edited file are

```
DEVICE=eth1
ONBOOT=yes
```

```
BOOTPROTO=none
NETMASK=255.255.255.0
IPADDR=192.160.103.something
HWADDR=something
TYPE=Ethernet
```

One this is done, reboot. There should be no errors, though if you look at `ifcfg-eth1` you may find some extra bits added. If you ping the IP address defined in the file a normal response should be produced. Pinging this newly defined address from a normal PC on the main college network should produce no reply.

If a reply should appear then the server is "IP-forwarding" which is not wanted as it would prevent the isolation of the private network from the main one. This behaviour is controlled by the file ip_forward. I am assured the following unlikely command

```
echo 0 > /proc/sys/net/ipv4/ip_forward
```

will turn IP-forwarding off. The daemon routed would also have to be running, which by default seems not to be the case.

The private network can now be plugged into the new interface, and attempts made to ping other machines on the private section - e.g.

```
ping 192.160.103.150
```

should get a return from the 500MHz. It should now be possible to log into another machine by commands like

```
ssh  user@ipnumber
```

or

```
sftp user@ipnumber   (for secure file transfer)
```

What I did next was to login with ssh to a machine with an up-to-date version of the local version of the hosts file, list it, open another local window, and use vi to edit the required lines into the new machine's default version of the file /etc/hosts by cutting and pasting. The default file will only be 2 or 3 lines: more needs to be added onto it.

Don't put the network name of the new server or any of its IP addresses into the host file as active lines - though you can put them in as comments of course. Once this is done, you can use machine names in ping, ssh, sftp, and so on to link to the other systems on the private network.

An odd aside: the computers running Topspin have got bogus full names like hpsys1.ch.ic.ac.uk in their host files - this is necessary to defeat a bug in Topspin and for no other reason.

<u>Mounting NFS volumes from the spectrometers.</u>

This all needs to be done as root.

All the data which will be made available via the Samba service will need to be either in or mounted to a directory which is NOT in the path starting /home... (This is an important new rule for the current operating system.) There is usually an empty directory at root level meant for this sort of purpose with a name like /srv or /opt - if not then make one. Make sure you are NOT in this empty directory for the next step.

The command

```
mount -t nfs  a500:/opt/topspin/data  /srv
```

should mount the 500MHz data onto /srv - the command `df -k` should show it and a directory should show the user names. (Similar commands for other spectrometers.)

If this does not work make a note of any error messages and see the appendix about iptables. If you insert -F before -t the command just syntax checks without mounting - will catch most errors.

This turned out not to be any problem at all with the RedHat 5 system.

Once this test has worked unmount the NFS volume by `umount /srv` or similar.

The next step is to prepare directories in the empty directory (/srv in all following examples) to mount the NFS data from the spectrometers and also to hold any local data. By default these will all be created as owned by root: it is worthwhile to use chown and chgrp to make the ones to be used for local data into directories belonging to the normal user, so that the contents can later be edited by that user. The directory of the initial version in chnmrserv was

```
# ls -al /srv
total 232
drwxr-xr-x 16 root root   4096 Jul 20 17:50 .
drwxr-xr-x 24 root root   4096 Jul 20 18:27 ..
drwxrwxrwx 35 root root   4096 Apr  3 12:21 disk1
drwxr-xr-x  3 root root     22 Mar 12  2004 disk3
drwxrwxrwx 39 root root   4096 Jan 30 13:23 disk5
drwxr-xr-x  2 root root   4096 Jul 24 20:29 disk50
drwxrwxrwx 35 root root   4096 Jan 11  2007 disk6
drwxr-xr-x  2 root root   4096 Jul 24 20:24 disk60
drwxrwxrwx 37 root root   4096 Jan 16  2007 disk7
drwxrwxrwx 39 root root   4096 Apr 24 15:27 disk8
drwxr-xr-x  4 ops  ops    4096 Jul 27 16:24 docu
drwxr-xr-x  2 ops  ops    4096 Jul 27 15:38 NMRservice-form
drwxr-xr-x  3 ops  ops  135168 Jul 27 15:29 Old500
drwxr-xr-x 23 ops  ops    4096 Jul 27 15:49 QM
drwxr-xr-x  2 ops  ops    4096 Jul 20 17:49 Temp
drwxr-xr-x  3 ops  ops    4096 Jul 27 16:30 Topspin
```

The next step is then to modify the file `/etc/fstab` to cause automatic mounting of the files on reboot. This is done by leaving the lines built by the system install severely alone and adding a new section on the end. This can probably be done by cutting and pasting the relevant bit from an existing version of the file. The result for chnmrserv was as follows - the first 7 lines are those built automatically:

```
 cat /etc/fstab
/dev/VolGroup00/LogVol00 /                       ext3    defaults        1 1
LABEL=/boot              /boot                   ext3    defaults        1 2
devpts                   /dev/pts                devpts  gid=5,mode=620  0 0
tmpfs                    /dev/shm                tmpfs   defaults        0 0
proc                     /proc                   proc    defaults        0 0
sysfs                    /sys                    sysfs   defaults        0 0
/dev/VolGroup00/LogVol01 swap                    swap    defaults        0 0
#
#   Added nfs mounts for VLAN
#
d400:/opt/topspin/data  /srv/disk1            nfs     ro,user,rsize=32768
###eclipse:/usr/people     /disk2             nfs     ro,user
DRX4:/v/data            /srv/disk3            nfs     ro,user
###nmrs2:/home/serv/v400   /disk3             nfs     ro,user,rsize=8192
###delta1:/home          /disk4                nfs     ro,user,rsize=8192
a500:/opt/topspin/data  /srv/disk5            nfs     ro,user,rsize=32768
a400:/opt/topspin/data  /srv/disk6            nfs     ro,user,rsize=32768
b400:/opt/topspin/data  /srv/disk7            nfs     ro,user,rsize=32768
c400:/opt/topspin/data  /srv/disk8            nfs     ro,user,rsize=32758
```

Note that I have left in (but made into comments) lines for old machines. Reading the manual section for NFS on this new version of RedHat shows that the recommended value of the parameter rsize is now 32768 - this seems to get bigger in successive versions, so the active mounts were changed to this. (It is the size of a read buffer.)

Found the hard way that leaving a space before the number (rsize= 32768) is a FATAL error.

Rebooting at this stage should mount all the specified NFS data. Logging back in will show icons for each NFS volume: df -k should show all the systems.

Two cautions.
1) If any of the spectrometers is not running and exporting its data then the mount of that line in the file and subsequent lines will go wrong.
2) Some systems don't seem to cope with a long list of NFS mounts - HPSYS1 for a start. After rebootiing that computer there are assorted icons for mounted and unmounted volumes on the screen. Double clicking the unmounted ones mounts them - so (I think) does a mount -a. This has not yet been properly investigated.

Getting a Samba configuration file. (N.B. - see also Appendix 1 about using passwords)

The quickest method is to copy the file smb.conf from directory `/etc/samba` of a working machine to the same location in the new one (after copying any existing file to an alternative name). The example in chnmr3 is a bit out of date for modern versions of the software. Most of the problems in making Samba work in chnmrserv were caused by SELinux (see below) but I decided to start again using the example smb.conf file supplied with the system and adapting as required. The file consists of 2 sections - global settings and share definitions.

The copied file may need minor editing for path changes and to add or remove shares. Some of the features in my files such as defining the IP of the server should probably be removed. A complete smb.conf file should be syntax checked by the command `testparm`. This will produce a report of what the file should do in operation and mark any bad errors.

Do NOT omit this step !

The following is a working smb.conf file from the server chnmrserv. The directives host allow and remote announce specify the ranges of IPs which may read the data in the server, plus a couple of specific machines for Ed Smith et al. # starts a comment line as usual.

```
#======================Global Settings=======================
[global]
   workgroup = IC
#   realm = IC.AC.UK
   server string = NMR tst
#   printcap name = /etc/printcap
   load printers = no
#   printing = cups
#   cups options = raw
   guest account = nobody
   log file = /var/log/samba/%m.log
   max log size = 2
   security = SHARE
   hosts   allow   =   155.198.36.   155.198.224.   155.198.225.   155.198.228.
155.198.232.\
   155.198.234. 155.198.226. 155.198.231. 155.198.213.120 155.198.213.88 127.
   interfaces = 155.198.226.88/255.255.255.0
  remote   announce   =   155.198.36.   155.198.224.   155.198.225.   155.198.228.
155.198.232.\
   155.198.234. 155.198.226. 155.198.231. 155.198.213.120 155.198.213.88 127.
```

```
#   encrypt passwords = yes
   socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
   local master = no
#   wins server = icwins0.cc.ic.ac.uk
   wins server = 155.198.63.70
   dns proxy = no
   idmap uid = 16777216-33554431
   idmap gid = 16777216-33554431
   template shell = /bin/false
   winbind use default domain = no
#============================Share Definitions====================

[Docs]
   comment = Various notes
   guest only = yes
   path = /srv/docu
   writeable = no
   public = yes

[NMRform]
   comment = NMR service form blank
   guest only = yes
   path = /srv/NMRservice-form
   writeable = no
   public = yes

[Temp]
   comment = Old files loaded temporarily
   guest only = yes
   path = /srv/Temp
   writeable = no
   public = yes

[Topspin]
   comment = Topspin information etc.
   guest only = yes
   path = /srv/Topspin
   writeable = no
   public = yes

[AV500]
   comment = Avance 500MHz data
   guest only = yes
   path = /srv/disk5
   writeable = no
   public = yes

[DRX-400]
   comment = Current and old DRX-400 data (service)
   guest only = yes
   path = /srv/disk60
   writeable = no
   public = yes

[AV400A]
   comment = 400 A spectrometer
   guest only = yes
   path = /srv/disk6
   writeable = no
   public = yes

[AV400B]
```

```
   comment = 400 B spectrometer
   guest only = yes
   path = /srv/disk7
   writeable = no
   public = yes

[AV400C]
   comment = 400 C spectrometer
   guest only = yes
   path = /srv/disk8
   writeable = no
   public = yes

[NEW400s]
   comment = 3 Av-400s
   guest only = yes
   path = /srv/disk50
   writeable = no
   public = yes

[Old500]
   comment = Partial AM-500 data (retired)
   guest only = yes
   path = /srv/Old500
   writeable = no
   public = yes

[QM600]
   comment = Queen Mary 600MHz data
   guest only = yes
   path = /srv/QM
   writeable = no
   public = yes
```

Getting Samba to work..

If any of the share directories are going to contain local files (e.g. NMRform) put some files in them so that something will show up in tests.

There are a number of extra steps in getting Samba to work imposed by the security features of the latest versions of RedHat: these should be dealt with before trying any actual tests.

1) The passage of Samba data through the network interface must be permitted by going to the menu System>administration>security level and firewall: set the interface to allow samba to pass.

2) the “SELinux” package is a complex security setup and completely blocks Samba access unless various limits are observed. When it has done this (or anything similar) an icon appears on the top bar (yellow star) which opens a report panel - this can also be found in the system admin menu.

SELinux imposes the following restrictions on Samba:

a) No access should be made to anything in the /home/..... directories.

b) Specific action is required to allow Samba to use mounted NFS volumes. The control panel for SELinux Management is reached via the menu System>administration. The Select menu shows a range of difficult options!

Opening the choice Boolean shows a set of options for many things including Samba. Opening the Samba section shows that these including permitting the use of NFS data. Set

this on. Other choices affect how much smbd and nmbd are controlled - might be needed in future.

(These controls do allow operations with home directories, but meant for keeping many users' home directories on a server I think.)

c) Directories must be permitted to be accessed by Samba by a new command chcon which is supposed to mean 'change security context'

This command is recursive: the required command (according to SELinux) to allow access to directory foo is

```
chcon -R -t  samba_share_t   foo
```

The man page for chcon is pretty uninformative.

All the following needs to be done as root.

Before going any further unmount all the NFS volumes, or thousands of error messages will be generated ! (Use commands like `umount /srv/disk8` for my examples.)

Run chcon for each of the directories which will be used by samba - e.g. in my examples

```
chcon -R -t  samba_share_t   /srv/disk1
```

and so on. Then remount the NFS volumes.

It should now be possible to start the server by the command

```
/sbin/service  smb  start
```

which should produce 2 lines of messages.

It should now be possible to go to a PC and see the server operating - most quickly by opening run in the start menu and typing for example \\chnmrserv.ch.ic.ac.uk.

This should open a window showing all the shares. Check they work - if not something is probably wrong with the chcon commands and the SELinux troubleshooter is probably filling with messages. If these are worked through, enough of them make sense to make some progress.

If all the steps above were complete, there should be no problems.

Once the system is working it seems the chcon command is not necessary for changes - when extra files are copied into shared directories they appear at once.

Tidying up.

Some of the Samba "shares' need to point to data mounted from more than 1 place - Nick Davies has requested that 1 share should be able to see all the open-access 400s to simplify setting up for undergraduates.

This is done by the share New400s, which points to /srv/disk50.

This directory contains link files made by commands such as

```
ln -s /srv/disk6 400A
```

(General form is ln -s target-directory local-pointer. The -s specifies a "soft link" which is required for a file to point to a directory.)

A directory produced by `ls -al /srv/disk50` shows the link files pointing to the required places.

In principle it is possible to make the Samba server start automatically on reboot. In the menu system>admin>server settings>services is an editable table of the services which start at reboot. I have not at present done this because I prefer to see what is happening.

The same menu contains an item Samba which seems to do nothing - probably because the package SWAT is not installed.

Remaining queries and oddities.

The PCs persist in showing a share for faxes and printers although none is defined and nothing tried stops this - it seems to be an oddity of the current version of Samba.

Log files of connections are kept in the file /var/log/samba. These should have computer names but mostly only have IP numbers. I suspect the activities of SELinux.....

Appendix1.

Making Samba require password authentication.

The current versions of Samba will comminicate with Apple's OS-X if authentication is turned on and with Windows Vista, which may also need authentication (not checked) Adding authentication by password does increase security at the cost of extra complexity.

This change is made by setting the security choice in the global section of smb.conf to USER. The minimum changes were made to the existing set-up. The original plan was to provide access via the same user/password combinations as are used in the open-access 400MHz spectrometers, but a basic sort of configuration allows all the logins to access all the data and adding users is time-consuming - a single global code might have been better.

The following is the modified header (global) section of smb.conf now in chnmrserv:

```
#======================Global                          Settings
====================
[global]
   workgroup = IC
#   realm = IC.AC.UK
   server string = NMR file_server_2
#   printcap name = /etc/printcap
   load printers = no
#   printing = cups
#   cups options = raw
   guest account = nobody
   log file = /var/log/samba/%m.log
   max log size = 2
   security = USER
  hosts  allow  =  155.198.36.  155.198.224.  155.198.225.
155.198.228. 155.198.232.   155.198.165.26 \
   155.198.234.  155.198.226.  155.198.231.  155.198.213.120
155.198.213.88 127.
   interfaces = 155.198.226.88/255.255.255.0
 remote  announce  =  155.198.36.  155.198.224.  155.198.225.
155.198.228. 155.198.232. 155.198.165.26 \
   155.198.234.  155.198.226.  155.198.231.  155.198.213.120
155.198.213.88 127.

#   encrypt passwords = no
```

```
  socket    options    =    TCP_NODELAY    SO_RCVBUF=8192
SO_SNDBUF=8192
   local master = no
#   wins server = icwins0.cc.ic.ac.uk
   wins server = 155.198.63.70
   dns proxy = no
   idmap uid = 16777216-33554431
   idmap gid = 16777216-33554431
   template shell = /bin/false
   winbind use default domain = no
#============================ShareDefinitions============
 This is exactly the same as before.
```

The Samba server will now only allow access to users who are already registered as users of the Linux system in chnmrserv, so the first step is to create logins which cannot actually log in to a normal terminal session, to preserve security. (This is almost as daft as it sounds.) Once this is done the relevant section of the Samba software will allow the creation of passwords which only allow access to the Samba service.

1) Creating crippled Linux users and passwords
The first step is to create the Linux user IDs which will be used to get into Samba.

Open the menu System > Administration > Users and Groups (at some point the root password will be requested) This provides a table in its own window.

Choose Add User and fill in the pop-up table, selecting the shell `/sbin/nologin` and choosing NOT to make a home directory and to put the user in its own group. Give some sort of cryptic password at the correct point. The new user adds to the end of the table at once.

Next, in a terminal window become root, and for each crippled user enter the command

```
passwd -l <new-user>
```

This blocks any use of the password to log in. The effect can be confirmed by the command

```
passwd -S <new-user>
```

which prints a confirmation message.

2) Adding Samba passwords.
This is all done by the command `smbpasswd`, which must be run in a terminal window as root having changed to the directory `/etc/samba`

For each new user, enter

```
smbpasswd -a <user-name>
```

This will ask for the new password twice - this is the password they are given.

This command has many functions - see manual page or `smbpasswd -h`

Appendix 2

In describing the setting up of NFS mounts the security feature iptables was mentioned. This has given no trouble in working with the July 07 ICT version of RedHat 5 but did with an earlier system.

This feature examines each packet of data as it arrives at the network interfaces and applies rules which control its destination. The main purpose is to reject packets from unwanted sources. The current rules may be printed (as root) by `iptables -L -v`.

The manual pages for this feature are long and somewhat incomplete, but it is clear that incoming packets are either accepted if they pass various tests or rejected. The default produced by the operating system I used seems designed to reject certain types of traffic but not to control where it comes from. The tests are in “chains” - the important one has the name RH-Firewall-1-INPUT in my example.

In the system which gave trouble, some of the data packets used by NFS (the udp type) were not accepted from the private network. Root can alter the checks in the firewall chain: I added an initial step by

```
iptables  -I  <name-of-chain>  -p all  -s=192.160.103.0/24  -j ACCEPT
```

where the chain name should NOT be in brackets. Read the manual pages !!

It is possible to specify that rules only apply to a particular interface, which would probably be best if this sort of thing is needed again.

In the previous system I had, making the change permanent seemed extremely contorted. I finished up using the command `iptables-save` to put everything in a file once it worked and `iptables-restore` to put things back after a reboot. (This was hidden in the final local step of the files `rc.d`)