Accessing Research systems remotely
from Windows systems

Unlike the department's teaching Linux farm, which can be accessed directly from home machines (desktop or laptop), connecting to research systems on the departmental internal network is a two-stage process, using a *Secure Shell* connection (SSH).   Windows systems usually require an `ssh` client to be explicitly installed (as this is not included as part of a typical installation).
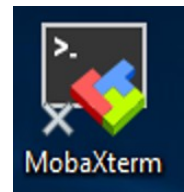
An alternative approach wouch be to set up a *VPN connection* to the departmental internal network, which avoids the need to go via the SSH gateway systems (see below).   This approach is documented separately – see the section *Guides* on the departmental intranet server for more details.

The software that we recommend is called **MobaXterm**, and this can be obtained from
https://mobaxterm.mobatek.net/download-home-edition.html
Download the `Installer edition` zip archive, open it, and run the MSI installer found there.   Then follow the Setup Wizard as usual.
This will typically install an icon on your desktop (as well as an entry in the Start Menu)

Configuring MobaXterm

The first time you use MobaXterm, you will need to supply details of the connection to set up. This information will then be saved, and can be used again for subsequent connections. MobaXterm supports a variety of different types of session – in this document we will only consider a simple SSH connection via the departmental SSH gateways.

There are two such systems – named `ssh1.csc.liv.ac.uk` and `ssh2.csc.liv.ac.uk`. (They may also be accessed via the names `linux1` and `linux2`, but we recommend use of the `ssh{1,2}` names as shown – the earlier forms may be withdrawn at some stage).

The available software and configuration of these two systems are slightly different – however their primary purpose is simply to provide a stepping stone to other systems within the department. They can handle simple processing tasks, but anything more demanding should be run on more powerful or specialist equipment. If either machine is not available, please use the other one (and inform the technical staff of the problem).
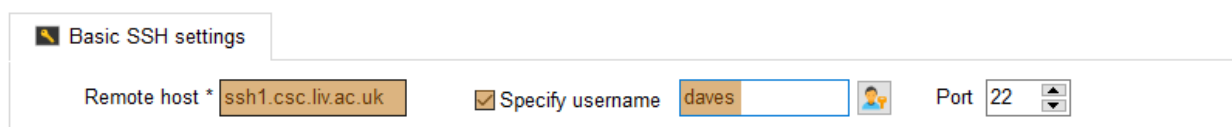
Click the *Session* icon *(left)* in the buttons bar across the top of the window.
Then click the *SSH* session button *(right)* and fill in the fields as follows:

```
Remote Host      ssh{X}.csc.liv.ac.uk
```
Check the box `Specify username` and enter your ***departmental*** username

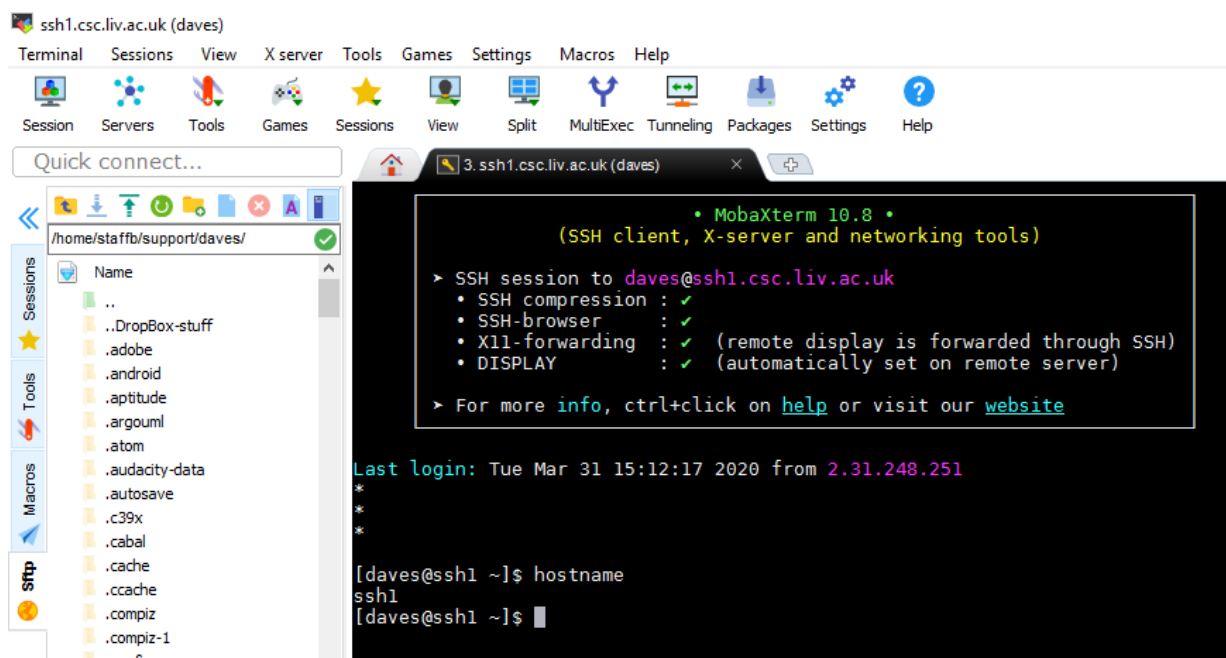The rest of the settings (*Advanced, Terminal*, etc) can be left at the default values.

Then click OK

You will be asked to supply your *departmental* password (and press Return).

This will log you into the chosen system, and you can run simple Linux commands on that machine. For text-based commands, the output will be shown in the same *MobaXterm* terminal window – graphical commands should open up a new window on your local display.

Note that immediately after logging in, you will be informed of the time and source (IP address or hostname) of the last successful login session.   If you do not recognise this connection, please inform the CS technical team immediately by emailing CSC-HELPDESK@liverpool.ac.uk
You may also wish to change your password.

When reporting this, or any other issue with the system, it is extremely helpful to include details of which Linux machine you were using (which should now appear as part of the command prompt), and the date & time any problem occurred. If you're not sure which system you're on, the command `hostname` will also show this information *(see above)*.

Having connected to the SSH gateway machine, the next step is to make a second *SSH* connection to the target system. This will commonly be a personal desktop machine (typically running Linux or MacOS), or a dedicated research group server – see your research group lead or supervisor for more information about any such resources available to you.
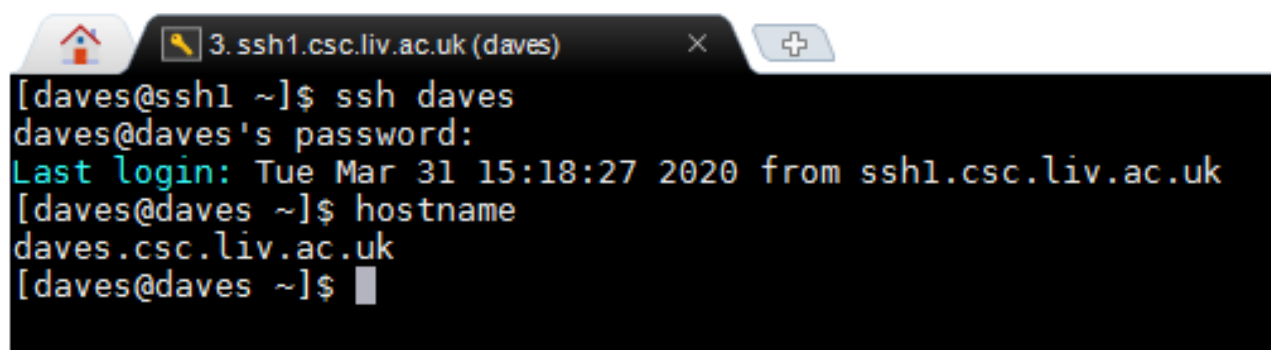
The department also has a number of general-purpose research systems available for staff/PhD use. These include:

O/S	Host names	
Scientific Linux 6	`core01` .. `core08`	*Now withdrawn*
Scientific Linux 7	`atlas01` .. `atlas08`	*Systems failing – will be withdrawn shortly*
Ubuntu 18.04	`ubuntu01` .. `ubuntu05`	*Plan to upgrade to Ubuntu 20.04 (once released) & expand to 8 systems*
Fedora 31	`fedora01` .. `fedora08`	`fedora04` *and* `fedora05` *should not be used*

Connect to the selected machine using the command: `ssh {host}`
If graphical commands will need to be run (with the output appearing on your local display), then use the command: `ssh -X {host}`
Note that there is no need to specify an explicit username (unless this is different on the target system).

When you have finished with the session, use the command `exit` to disconnect from the internal system, and return you to the SSH gateway machine. Then type `exit` again to disconnect from that, and return you to the local machine. You can then close the *Terminal* window, as usual.
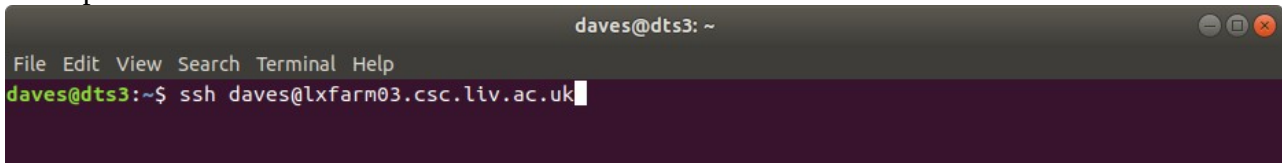 Note that closing the *Terminal* window *without* first `exit`ing cleanly may leave processes running on the internal and/or gateway machines, and could cause problems in the future.

This will then log you into the selected teaching Linux farm machine, and give you a command prompt.
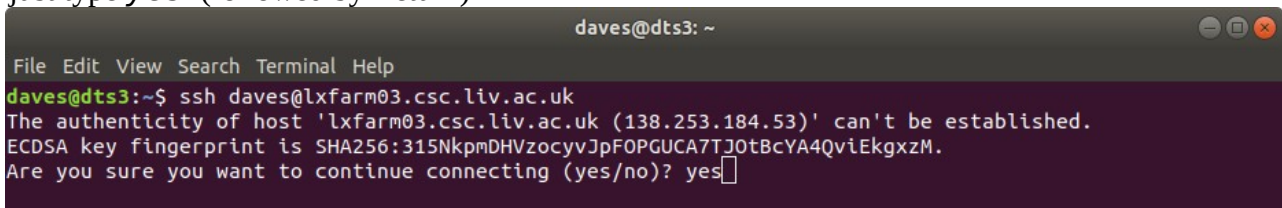
To connect to one of these systems, type the command `ssh` into the ***Terminal*** window, together with your University username, and the name of the chosen machine (separated by the @ sign). Then press Return

```
                              daves@dts3: ~                          ● ⊡ ⊗
 File  Edit  View  Search  Terminal  Help
daves@dts3:~$ ssh daves@lxfarm03.csc.liv.ac.uk
```

(Note that if your username on the local machine happens to be the same as your University username, then you can omit the *username@* element.   But if in any doubt, keep it in)

The very first time that you connect to a particular machine, you will be asked to confirm that this is the system you expect – in most circumstances this *digital fingerprint* is pretty meaningless, so just type *yes* (followed by Return)
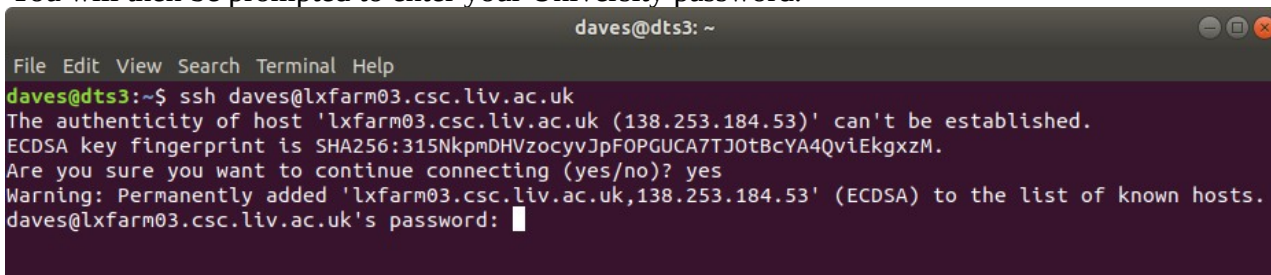
```
                              daves@dts3: ~                          ● ⊡ ⊗
 File  Edit  View  Search  Terminal  Help
daves@dts3:~$ ssh daves@lxfarm03.csc.liv.ac.uk
The authenticity of host 'lxfarm03.csc.liv.ac.uk (138.253.184.53)' can't be established.
ECDSA key fingerprint is SHA256:315NkpmDHVzocyvJpFOPGUCA7TJOtBcYA4QviEkgxzM.
Are you sure you want to continue connecting (yes/no)? yes
```

The next time you connect **to the same machine**, the fingerprint will be checked automatically, and you should not be prompted for this again (as long as the fingerprint matches).
If you choose a different system within the teaching Linux farm, then you'll get a similar prompt for that new machine (but with a different fingerprint).

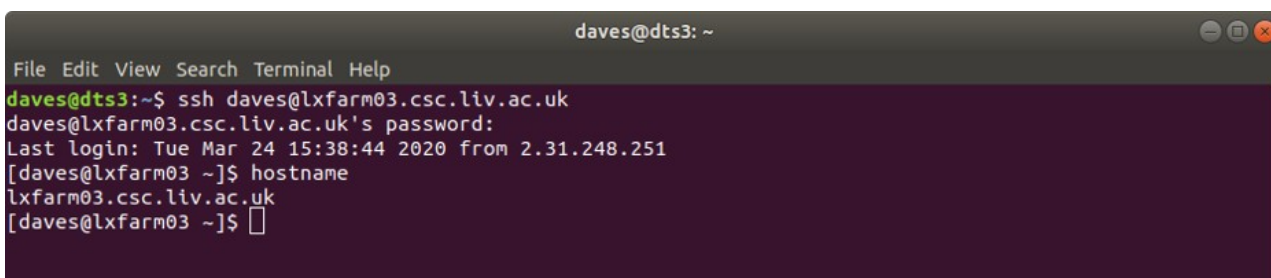You will then be prompted to enter your University password.

Note that this whole connection is encrypted, so the password (or anything else you type) cannot be intercepted or read while crossing the Internet. This is the same idea as secure web pages (`https`)

Once you have supplied the (correct) password, you will be logged into the chosen system, and can run Linux commands on that machine. For text-based commands, the output will be shown in the same *Terminal* window – graphical commands *should* open up a new window on your local display. If this does not happen (and you need to run such a graphical application), then disconnect from the Linux system (see below), and re-connect using the command

ssh **-X** *username*@lxfarm*NN*.csc.liv.ac.uk

Note that immediately after logging in, you will be informed of the time and source (IP address or hostname) of the last successful login session. If you do not recognise this connection, please inform the CS technical team immediately by emailing CSC-HELPDESK@liverpool.ac.uk
You may also wish to change your password.

When reporting this, or any other issue with the system, it is extremely helpful to include details of which Linux machine you were using (which should now appear as part of the command prompt), and the date & time any problem occurred. If you're not sure which system you're on, the command `hostname` will also show this information (see above).

When you have finished with the session, use the command `exit` to disconnect, and return you to the local machine. You can then close the *Terminal* window, as usual.
  Note that closing the *Terminal* window *without* first `exit`ing from the Linux system may leave processes running on that Linux farm machine, and could cause problems in the future.