

Configuring Shared Screens in Linux

It is very convenient to share terminals for teaching the command-line, or giving personal help concerning it. In this way, we can run a workshop where some computers (laptops and towers) need to share a console for interactive teaching of the command line. The teacher and all students can see and use the same terminal, while it is vital that we do not expose anyone's passwords.

The Hardware Arrangement

Figure 1 shows what is needed. For example: Each of the computers — `student1`, `student2`, `student3`, etc. — (on the left) connects *via* a hub to the wired ethernet interface of the `teacher` computer (on the right) or each computer gets an IP address (examples shown) from a common wireless access point.



Figure 1: NETWORK CONFIGURATION FOR SCREEN SHARING

Using SCREEN to Share Terminals

This demonstration shows in detail how a teacher with username `guru` on a computer with hostname `teacher-pc` invokes screen sharing with a student with username `newbie` on a computer with hostname `student-pc`, on the Local Area Network, no personal passwords being revealed.

What the Teacher Needs to Do

The teacher should ensure that their computer has the `screen` program (for screen sharing) and the `openssh-server` program (for remote login) installed on their computer, by gaining internet access and running the following command:

```
guru@teacher-pc$ sudo apt install screen openssh-server
```

The teacher cannot just log in to the student computer to see what is going on because that requires the student to give up their password to the teacher. To get around this, the teacher creates a neutral 'placeholder' user `lsga` with password `lsga` and all its defaults thus:

```
guru@teacher-pc$ sudo adduser lsga
Adding user 'lsga' ...
Adding new group 'lsga' (1002) ...
Adding new user 'lsga' (1002) with group 'lsga' ...
```

```
Creating home directory '/home/lsga' ...
Copying files from '/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for lsga
Enter the new value, or press ENTER for the default
Full Name []: Room Number []: Work Phone []: Home Phone []: Other []:
Is the information correct? [Y/n]
```

The teacher could now change to the first desktop, open a terminal allocated to the student, and become the user `lsga` by typing:

```
guru@teacher-pc$ su - lsga
Password: lsga
lsga@teacher-pc$
```

and then start a screen session labelled and dedicated to the student `newbie` like this:

```
lsga@teacher-pc$ screen -S newbie
```

What the Student Needs to Do

The student should ensure their computer has the `openssh-server` program (for remote login) installed on their computer, by gaining internet access, and running the following command:

```
newbie@student-pc$ sudo apt install openssh-server
```

The student logs in to the teacher's computer at IP address `192.168.2.99` as user `lsga` thus:

```
user@student-pc$ ssh lsga@192.168.2.99
lsga@192.168.2.99's password: lsga
lsga@teacher-pc:~ssh$
```

The student can then attach to the teacher's screen session running on the first desktop thus:

```
lsga@teacher-pc$ screen -x newbie
```

Suddenly, the teacher and student see exactly the same screen image on their respective computer. They both see the commands each other is typing and both can see the result of that command, *just as if they were sitting at the other terminal*.

Finally, the student `newbie` returns to their own computer `student-pc` because that is where everything for which they seek help is to be done, by logging back in thus:

```
lsga@teacher-pc$ ssh newbie@192.168.2.11
newbie@student-pc:~ssh$
```

The student and teacher can follow each others commandline typing and either can modify it as necessary. When convenient, the teacher or student may end their screen session by hitting `CTRL+D` (which means: hold the `CTRL` key down and press `D`). They should see:

```
^D
[screen is terminating]
lsga $
```
