

Well, for a jump server not much is needed.

Getting the tools

1. Login via SSH.
2. Shift into root: `$ sudo su`
3. Ensure git is installed: `$ apt-get update -y && apt-get install git -y`
4. Run: `$ git clone https://github.com/csr13/aux_nuit.git ~/tools`

Creating Users

1. Change to directory: `$ cd ~/tools/admin`
2. Execute `./create_user.sh`
 - a. This will prompt username it will ask if you want ssh keys [y/n]
 - b. It will create `/home/<username>/` and create group `<username>` and add the user to it.
 - c. Keys for users created will be on `/root/ssh_keys/<username1>,<username2>`
3. Before handing over their keys to users, you need to run a few commands
4. Change ownership of each users created `authorized_keys` file under `/home/<user>/.ssh`
 - a. Run: `$ chown <user>:<user> /home/<user>/.ssh/authorized_keys`

SFTP the keys to your localhost 2 give to users.

1. First make sure the keys created for your users are on the user account that you are sftping to in this case most accounts are called ubuntu, your sys admin might vary.
2. Run: `$ ssh -i <key> ubuntu@<dev-ip>`
3. Change accounts to root: `$ sudo su`
4. Go to `/root/ssh_keys/<user1,2,3>`
5. Copy `id_rsa` file into `/home/ubuntu/<name of the file>`
6. Give ownership to ubuntu: `$ chown ubuntu:ubuntu /home/ubuntu/<name of user key>`
7. Exit ssh session.
8. Open a new sftp, under ubuntu user, session or scp, whatever u want.
9. Once you are in via sftp just run a get command: `> get <user pem file> <host output file>`

Setting up desktop

1. Install ubuntu core (minimal): `$ sudo apt-get install -y xubuntu-core`
2. Install x2go: `$ sudo apt-get update && sudo apt-get install -y x2goserver x2goserver-xsession sudo`
3. Enable x2go systemctl service: `$ systemctl enable x2goserver`

Setting up VSCODE:

1. As root: `$ snap install -classic code`

Extra - In case users need sudo access, they need to be added to the sudo group, and a password.

1. Add a user to sudo group: `$ usermod -aG sudo <username>`
2. Make password for user: `$ passwd <user>`

At last make sure login via password is disabled on the main ssh config file, and ensure key access is the only one available, so users can only use password to execute sudo actions, not login with password.