Virtual Machine Configuration

James Duncan
Computer Systems, Infrastructure & Management

Contents

Virtual Machine Beginning State 2

Admins 3

Securing root user 4

User accounts 4

Configuring SSH & Public Key Authentication 4

Known Host Authentication 7

User Groups and Folder permissions 7

Uploading Research Material 8

Critical Reflection & Conclusion 9

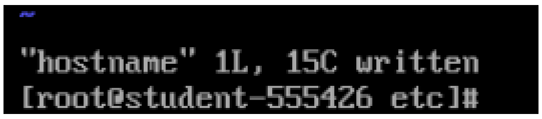
Bibliography 10

Appendix 11

- Check if admins are part of wheel 11
- Validation of admin sudo 11
- Validation of Created Groups & Users 11
- SSH Connections to other VMs 11
- Downloads of all public keys 12
- Public Key Authentication (Connection Tested by Ashley, confirmed on Discord) 13
- Host Based Password Authentication (local IP Through VPN Tests) 13
- Host Based Password Authentication (Connection Tested by Ashley on Discord) 14
- Folder Permissions commands (http Changed to 770 later on) 14
- Annika Tests 14
- Steve Tests 15
- Laura Lance Tests 15
- John Tests 16
- Laura Lance SFTP via Windows UI using WINSCP 17

Virtual Machine Beginning State

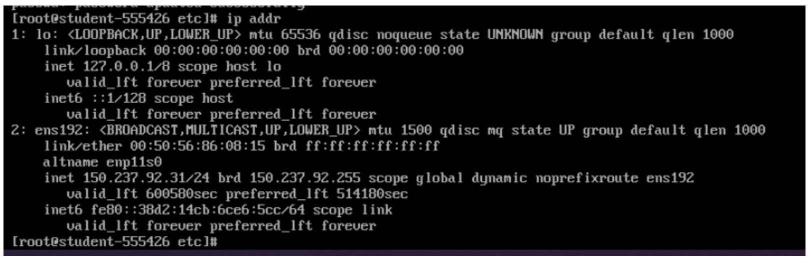
The Virtual Machine (VM) first starts as a fresh install of Arch Linux with none of the settings required for this project. The first thing the administrator (admin) has to do is to change the hostname of the virtual machine to be consistent with the admin's student ID (in this case "student-555426), this is part of establishing an internet connection for the VM which is required for network connectivity used later in remote access using SSH (Secure Shell). This can be achieved by changing directory to the "etc folder" with the "cd" command and finding the file called "hostname". The admin can navigate and find files by using the "ls" command that shows the contents of the directory. By standard the only text editor on the machine is vim so this must be used to edit the hostname file, once edited the admin can save and exit the file by pressing escape and providing the ":wq" command. Confirmation of writing the hostname can be found in figure 1.



```
~
"hostname" 1L, 15C written
[root@student-555426 etc]#
```

Figure 1: Hostname written

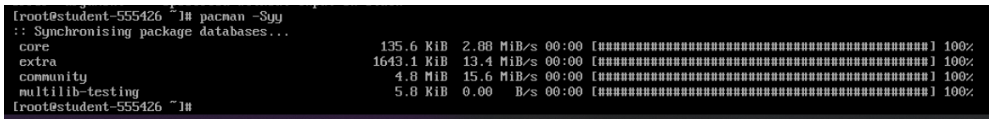
The VM can then be rebooted and connection to the internet can be tested by pinging an IP address (Google DNS 8.8.8.8), your own IP address can also be found by issuing the command "ip addr" which will print all the network connectivity information which can be found in figure 2. In this case the IP address for this VM is 150.237.92.31 as found in section 2 of figure 2.



```
[root@student-555426 etc]# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: ens192: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:50:56:86:08:15 brd ff:ff:ff:ff:ff:ff
    altname enp11s0
    inet 150.237.92.31/24 brd 150.237.92.255 scope global dynamic noprefixroute ens192
       valid_lft 600580sec preferred_lft 514180sec
    inet6 fe80::38d2:14cb:6ce6:5cc/64 scope link
       valid_lft forever preferred_lft forever
[root@student-555426 etc]#
```

Figure 2: Network Information

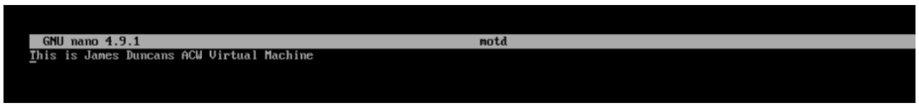
Once a connection has been established the admin can check if all the packages from the Arch Linux Distribution are up to date with "pacman -Syy" which checks all the packages that the install has and what it depends on checking if they are up to date or not. This can be shown in figure 3.



```
[root@student-555426 ~]# pacman -Syy
:: Synchronising package databases...
 core                              135.6 KiB  2.88 MiB/s 00:00 [############################################] 100%
 extra                            1643.1 KiB  13.4 MiB/s 00:00 [############################################] 100%
 community                           4.8 MiB  15.6 MiB/s 00:00 [############################################] 100%
 multilib-testing                    5.8 KiB  0.00   B/s 00:00 [############################################] 100%
[root@student-555426 ~]#
```

Figure 3: Updating Arch Linux Package

Other installs can be done with pacman as well such as installing a new more convenient text editor, so the admin does not have to use vim. Nano (text editor) can be installed by using "pacman -Sy nano" to synchronise and fetch the latest copy of the editor. The manual pages can also be installed so that the user can use "man nano" to get a help sheet of all the provided commands nano offers. Nano can now be tested by changing the "motd" file in the "etc" folder, this is the message that shows up when the VM is successfully logged into. You can find the nano example in figure 4.

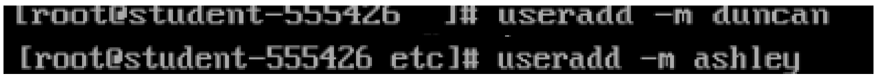


```
  GNU nano 4.9.1                              motd
This is James Duncans ACW Virtual Machine
```

Figure 4: Nano motd

Admins

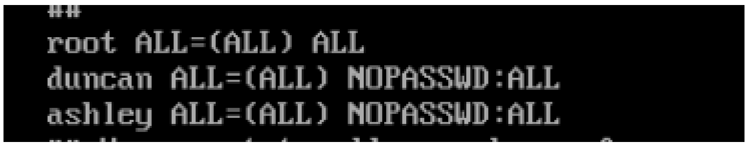
Part of securely locking down a virtual machine is to create admins that have access to run high level commands that a normal user may not have access to. Their prime use is to keep the system functional for the users it supports. To create an admin account the "useradd -m <username>" command can be supplied which will add a user with a home directory (-m) of whatever user name supplied. For this system both James Duncan and Ashley Williamson will require an admin account so both accounts should be created. This is shown in Figure 5.



```
[root@student-555426   ]# useradd -m duncan
[root@student-555426 etc]# useradd -m ashley
```

Figure 5: Create admin accounts

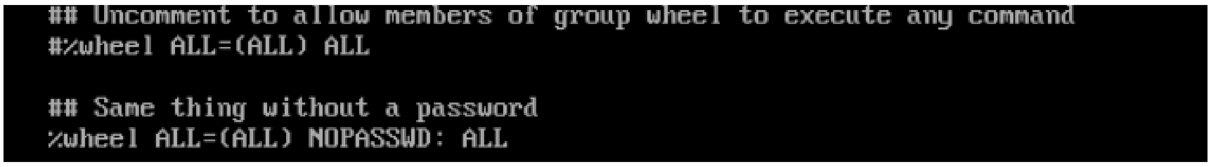
These accounts require SUDO access to perform root level commands or bypass permissions, as a result sudo must be installed on the virtual machine, this can be achieved by using "pacman -Sy sudo" which installs the sudo packages. Once installed it is possible to give the accounts sudo access through the sudoers file, this file contains all the configurations for sudo users, however to access this it is required to set an editor to edit the file, this can be done by running the "export EDITOR=nano" which sets the default text editor. The user can then run the visudo command and it will open the sudoers file in the nano text editor set using the export command. Once access to the sudoers file has been established there are two ways of granting the users sudo access, first you can directly state the user in the sudoers file as shown in figure 6, where the user is declared, and they are given access to sudo without inputting a password.



```
##
root ALL=(ALL) ALL
duncan ALL=(ALL) NOPASSWD:ALL
ashley ALL=(ALL) NOPASSWD:ALL
```

Figure 6: Add users to sudoers file

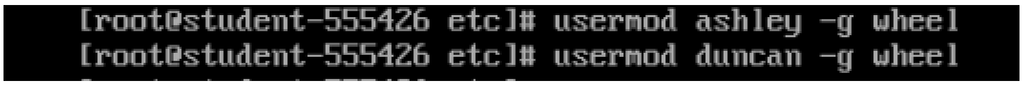
Alternatively, you can enable the "wheel" group to grant sudo access and add both admins to the wheel group. Using the wheel group is preferred as if you want a new admin you can just add this account to the wheel group which directly grants sudo access instead of having to change the sudoers file consistently, this is shown in figure 7.



```
## Uncomment to allow members of group wheel to execute any command
#%wheel ALL=(ALL) ALL

## Same thing without a password
%wheel ALL=(ALL) NOPASSWD: ALL
```

Figure 7: Wheel Group Sudo

These users must now be added to the wheel group in order to use the sudo permissions that wheel has, this can be done by using "usermod <username> -g wheel" this will add the user to the group (-g) wheel. This is shown in figure 8.

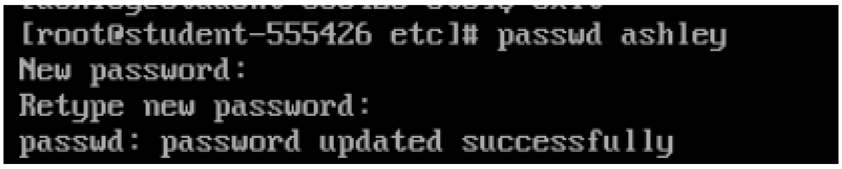


```
[root@student-555426 etc]# usermod ashley -g wheel
[root@student-555426 etc]# usermod duncan -g wheel
```

Figure 8: Add Admins to Wheel Group

These user now have sudo access which makes them incredibly powerful users in the system, these accounts should be the most secure accounts on the network so their passwords should be changed to a secure password. A secure password should not be obvious to any user and should be complicated enough to not be brute forced, therefore a secure password can be made by a user although the user should abide by the Hull

University Password Policy (Hull University , 2020). For these passwords the admin chose to use the website random key gen to generate passwords for these accounts (Random Keygen, n.d.). To change these passwords the root user can issue the command "passwd <username>" as the root user and sudo users can change anyone's passwords but normal users can only change their own. This is shown in Figure 9.



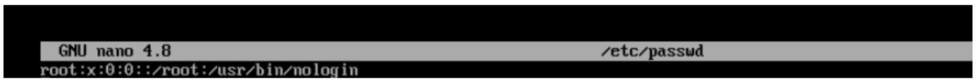
```
[root@student-555426 etc]# passwd ashley
New password:
Retype new password:
passwd: password updated successfully
```

Figure 9: Change Admin Passwords

Duncan: LMvsGplYwW , Ashley: UyQ3xNJTZA

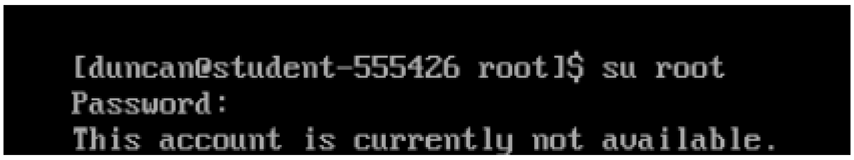
Securing root user

Now that the system has admin accounts that can manage and administrate the root account is no longer required and should be locked down as much as possible as if someone has access to this they could cause significant damage to the system. The root password can be changed, as above this ensures some security although its best that root is entirely disabled so no user can login as root. This is possible by navigating to "etc" and editing the "passwd" file with nano, the admin can now change the root user to have no logins from users, this is shown in figure 10. Once saved no user can log in to root which is shown in figure 11.



```
  GNU nano 4.8                                        /etc/passwd
root:x:0:0::/root:/usr/bin/nologin
```

Figure 10: disable Root Access

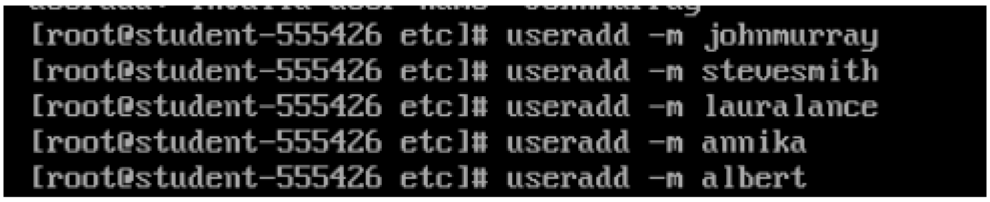


```
[duncan@student-555426 root]$ su root
Password:
This account is currently not available.
```

Figure 11: Root Disabled

User accounts

Now that root is secured and there are admin accounts we can create our users for the system, the same useradd command can be run for these accounts as shown in figure 14. The passwords for these users should be set, however, they are less important than the admin's as they don't have sudo access. The admin can set these passwords to the users first name although, when the user first logs in they should change their own password to be in compliance with Hull University's guidelines (Hull University , 2020)

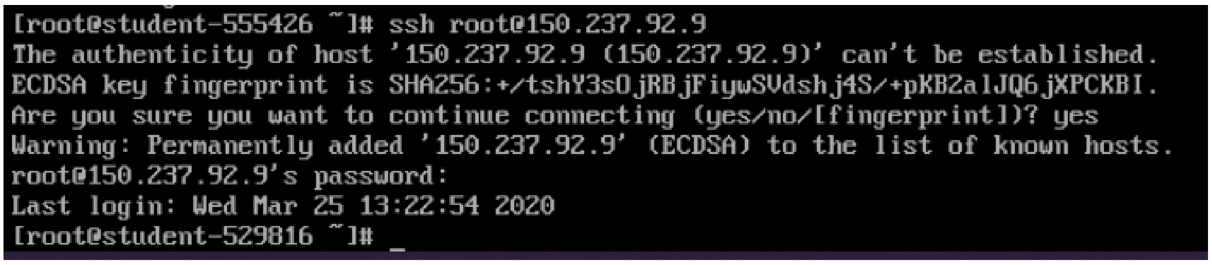


```
[root@student-555426 etc]# useradd -m johnmurray
[root@student-555426 etc]# useradd -m stevesmith
[root@student-555426 etc]# useradd -m lauralance
[root@student-555426 etc]# useradd -m annika
[root@student-555426 etc]# useradd -m albert
```

Figure 12: Add Users

Configuring SSH & Public Key Authentication

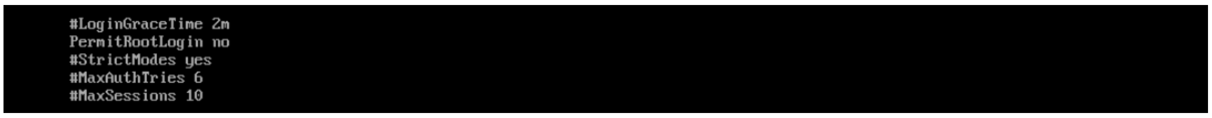
SSH or Secure shell is a way of connecting to the system remotely from a different system. To use this the admin must install "openssh" which can be installed using "pacman -Sy openssh" to fetch the package and install it. This module can then be enabled and started using "systemctl enable sshd" and "systemctl start sshd" which will start the ssh daemon to allow connections. A user can now SSH into another users account, for example you could remotely access another VM as shown in figure 13 where the command "ssh root@150.137.92.9" is executed and the user is logged into the VM at address 150.137.92.9 as root after entering the user password.



```
[root@student-555426 ~]# ssh root@150.237.92.9
The authenticity of host '150.237.92.9 (150.237.92.9)' can't be established.
ECDSA key fingerprint is SHA256:+/tshY3sOjRBjFiywSVdshj4S/+pKB2a1JQ6jXPCKBI.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '150.237.92.9' (ECDSA) to the list of known hosts.
root@150.237.92.9's password:
Last login: Wed Mar 25 13:22:54 2020
[root@student-529816 ~]# _
```

Figure 13: SSH to another VM

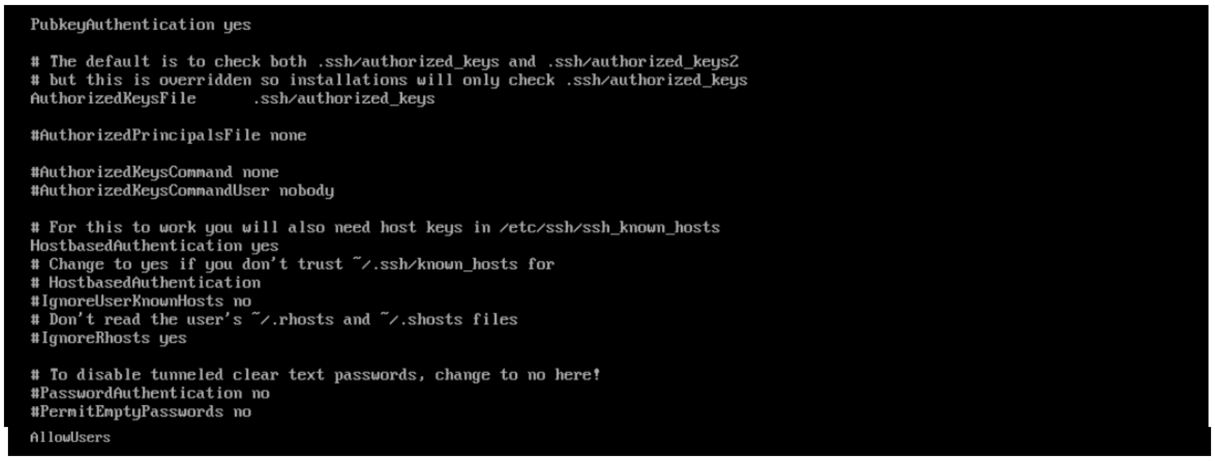
Once a connection has been established the user can use the system just like the current system, this is potentially very dangerous if someone were able to log on as root or as an admin. To check that root login is disabled you have to navigate to "etc/ssh/" and open the "sshd_config" file which has all of the settings for SSH connections. If "PermitRootLogin" is on make sure to turn it off to ensure roots security. This can be shown in Figure 14.



```
#LoginGraceTime 2m
PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
```

Figure 14: sshd_config

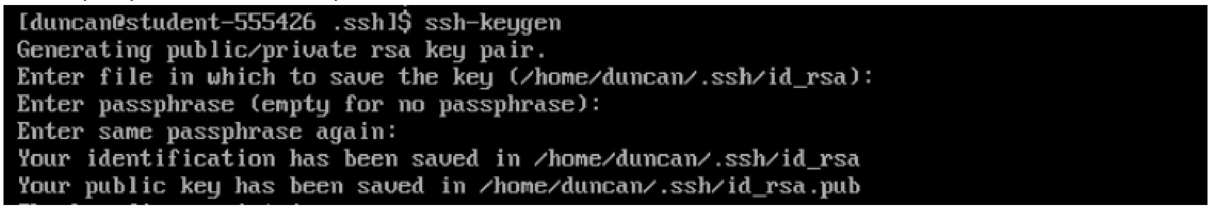
This file also has settings for public key authentication which works based on asymmetric encryption. Asymmetric encryption creates pair of keys, each person in the connection say Alice and Bob both have a public and private key. Both keys can encrypt, and both can decrypt their counterparts data. This is because both keys are mathematically linked to each other normally with use of large prime numbers. Asymmetric encryption works when both parties publish their public key on the internet so that they both can access those keys, say bob wants to send some data to Alice, Bob will encrypt the data using Alice's public key and send the data to Alice, because no malicious actor can decrypt this data without Alice's private key this data is secure. When Alice receives the message, she can decrypt the message safely. In a Linux system the public key can be added to an authenticated keys file that will only allow connections from users that have a public key in this file, therefore authenticating this user. This public key authentication method also allows users to not be required to enter a password, therefore password authentication can be turned off when public key authentication is turned on. These modified settings can be shown in Figure 15.



```
PubkeyAuthentication yes

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
AuthorizedKeysFile      .ssh/authorized_keys

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
HostbasedAuthentication yes
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication no
#PermitEmptyPasswords no
AllowUsers
```

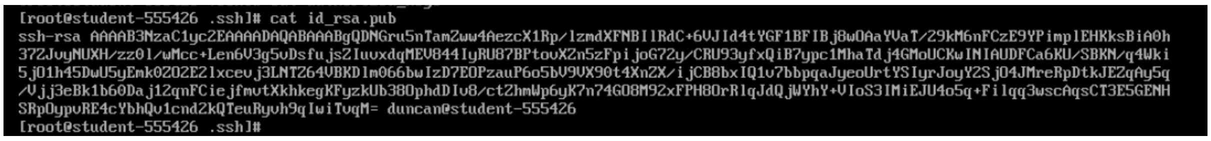
Figure 15: Public Key Authentication Enabled

Once changes to this file are made it is necessary to restart the SSHD service as it is currently running without the adjusted settings, this can be done by issuing the command “systemctl restart sshd”. To create a key pair for a user, say “duncan” the command “ssh-keygen” can be executed, this will create an RSA (Asymmetric Encryption Algorithm) Key pair in the users “.ssh” folder shown in figure 16, additionally, you can find “duncans” public key in figure 17. If you wanted to allow “duncan” to connect to your system you would add that key to your authorised keys file.



```
[duncan@student-555426 .ssh]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/duncan/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/duncan/.ssh/id_rsa
Your public key has been saved in /home/duncan/.ssh/id_rsa.pub
```

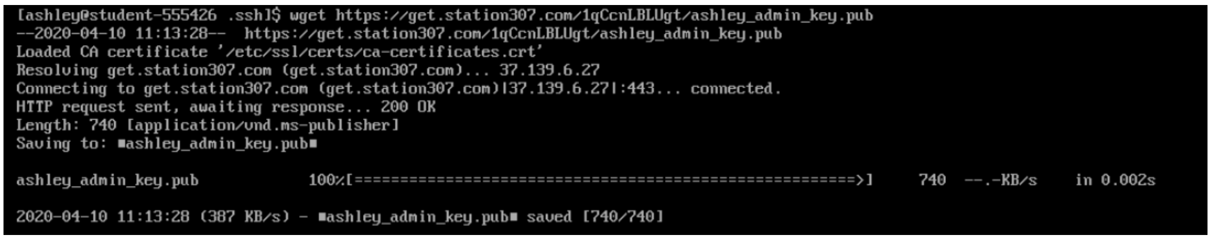
Figure 16: Generating RSA Key Pair



```
[root@student-555426 .ssh]# cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDNGru5nTamZww4AezcX1Rp/lzmdXFNBIlRdC+6VJId4tYGF1BFIBj8wOAaYVaT/29kM6nFCzE9YPimplEHKksBiA0h
37ZJvyNUXH/zz0l/wMcc+Len6V3g5vDsfujsZIuvxdqMEV844IyRU87BPtovXZn5zFpijoG72y/CRU93yfxQiB7ypc1MhaTdj4GMoUCKwINIAUDFCa6KU/SBKN/q4Wki
5jO1h45DwU5yEmk0Z02E2lxcevj3LNTZ64VBKDlm066bwIzD7EOPzauP6o5bV9VX90t4XnZX/ijCB8bxIQ1v7bbpqaJyeoUrtYSIyrJoyY2SjO4JMreRpDtkJEZqAy5q
/Vjj3eBk1b60Daj12qnFCiejfmvtXkhkegKFyzkUb38OphdDIv8/ctZhmWp6yK7n74GO8M92xFPH8OrRlqJdQjWYhY+VIoS3IMiEJU4o5q+Filqq3wscAqsCT3E5GENH
SRpOypvRE4cYbhQv1cnd2kQTeuRyvh9qIwiTvqM= duncan@student-555426
[root@student-555426 .ssh]#
```

Figure 17: “duncans” Public Key

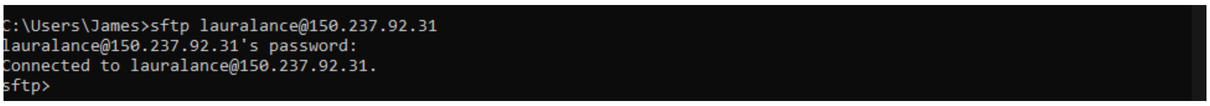
Supplied with the system specification there are public keys for each user that have been created on a separate system, if these keys are added to the “authorized_keys” file on each user then they could use public key authentication to remote ssh to their user account on this system. To do this the user must first load their public key on the system, for this “wget” was installed using “pacman -Sy wget” as “wget” will pull a file from a link and store it as a file which can be manipulated on the system. The public keys were then uploaded to a temporary file host (Station307, 2020) which gives a link to pull a file. The file being pulled using “wget” can be seen in figure 18.



```
[ashley@student-555426 .ssh]$ wget https://get.station307.com/1qCcnLBLUgt/ashley_admin_key.pub
--2020-04-10 11:13:28--  https://get.station307.com/1qCcnLBLUgt/ashley_admin_key.pub
Loaded CA certificate '/etc/ssl/certs/ca-certificates.crt'
Resolving get.station307.com (get.station307.com)... 37.139.6.27
Connecting to get.station307.com (get.station307.com)|37.139.6.27|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 740 [application/vnd.ms-publisher]
Saving to: ■ashley_admin_key.pub■

ashley_admin_key.pub          100%[=======================================================>]     740  --.-KB/s    in 0.002s

2020-04-10 11:13:28 (387 KB/s) - ■ashley_admin_key.pub■ saved [740/740]
```

Figure 18: Pulling Station307 link

Once all users keys are pulled from the temporary host the user can create an “authorized_keys” file by using “touch authorized_keys” and appending the key by issuing the command “cat Key >> authorized_keys”. This will add the key to the authorised keys file and allow connections using that key. Once all the keys are added each user can SSH into their own account and in the case of Laura lance she can access the http folder under “srv/http” by using the “sftp” command in windows or though a UI which she can install. This can be seen in Figure 19 and the UI of WinSCP with an active connection to Laura can be found in the appendix.



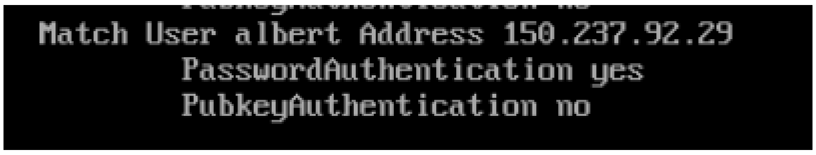
```
C:\Users\James>sftp lauralance@150.237.92.31
lauralance@150.237.92.31's password:
Connected to lauralance@150.237.92.31.
sftp>
```

Figure 19: SFTP Connection

Additionally you can utilise the “Match” keyword in “sshd_config” to force the user into a folder and force a command to the user, as a result of doing this you jail the user to only that folder meaning they cannot access any other folder outside of the one they are in. If we use the commands “ChrootDirectory <filepath>” to lock Laura to srv then she can’t leave that directory. Additionally, if we use the command “ForceCommand internal-sftp”she can only use the sftp command when connecting to the server. The experience is the same on her end but on the admin’s end this is a much more secure way of handling Laura.

Known Host Authentication

In the “sshd_config” file it is possible to specify which host a connection is coming in from using “Match Host” or In this case “Match Address” this can also take a specific user, in Alberts case he wants to connect to his account with a password via a specific host without a key pair. This can be achieved my adding “Match User <username> Match Address <Ip Address>, PasswordAuthentication yes”. This code will allow password authentication from only albert on this host or Ip address, this can be shown in figure 19.

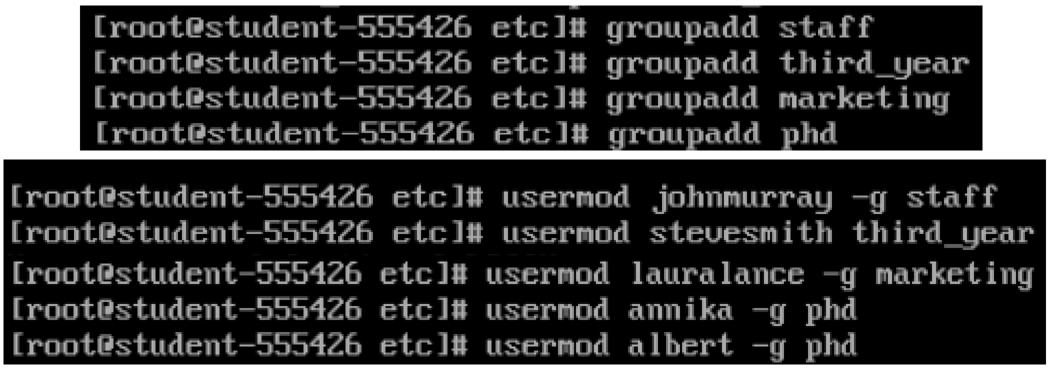


```
Match User albert Address 150.237.92.29
        PasswordAuthentication yes
        PubkeyAuthentication no
```

Figure 19: Host Based Password Authentication

User Groups and Folder permissions

Each user is required to have access to different folders depending on their user, it is best to split down the users into what access they require. Both Annika and Albert are PHD Students that require access to a PHD folder managed by John Murray, for this reason adding them to a PHD user group would make sense as they both will have the same user permissions. Laura Lance’s department is marketing and therefore only requires access to the http folder, she can be given a marketing group that allows her group access to the http folder and if any other marketing user needs access to this folder they can also be added. Steve is the only third year student on the system that requires a folder managed by John Murray, he can be given a group of his own for this as no other student or user is required to access this folder. John Murray on the other hand manages most of the students on this system, as he is a staff member he can be given a staff group although John may not use this group as much as the other users. These groups can be added and given to each user in the system using “groupadd <group>” and “usermod <username> -g <group>”, this can be shown in figure(s) 20.

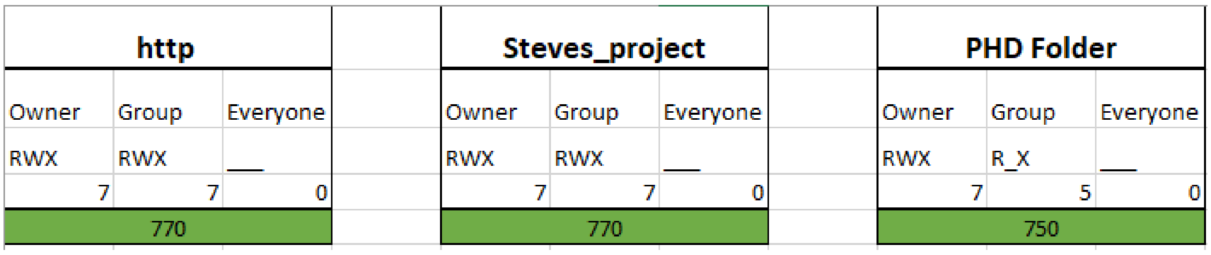


```
[root@student-555426 etc]# groupadd staff
[root@student-555426 etc]# groupadd third_year
[root@student-555426 etc]# groupadd marketing
[root@student-555426 etc]# groupadd phd
```

```
[root@student-555426 etc]# usermod johnmurray -g staff
[root@student-555426 etc]# usermod stevesmith third_year
[root@student-555426 etc]# usermod lauralance -g marketing
[root@student-555426 etc]# usermod annika -g phd
[root@student-555426 etc]# usermod albert -g phd
```

Figure(s) 20: Creating and adding groups

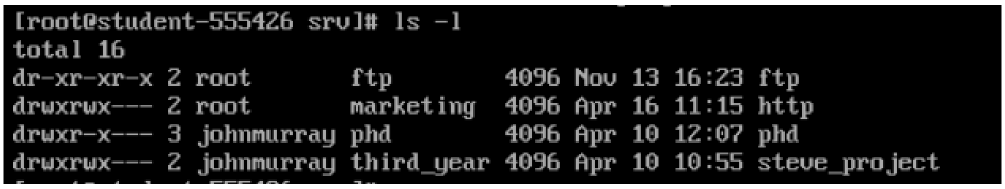
These folders can then be made by navigating to the “/srv” directory and issuing the command “mkdir <folder name>”. Permissions can then be set on these folders, Linux Permissions work by assigning Read, Write and Execute permissions to the Owner of the file, The group that the file is owned by and then everyone else. To give Laura special access to http her group can be granted Read Write and Execute permissions and the folder owner could be root or an admin with full permissions. Other users should not be able to access this folder so all permissions should be off. For Steve’s folder the folder owner should be John Murray as he is the main admin for the resources in the folder, Steve can then have full permissions to his folder as he is managing the project from it, everyone else should not have permissions for this folder. Finally, the PHD folder should also be owned by John Murray as he is the main admin for resources here and requires full permissions, it is important that the users of PHD do not edit so the PHD group can be given read and execute permissions but not write permissions. These permissions are represented in the form of octal numbers and the break down to these numbers can be found in figure 21.



http				Steves_project				PHD Folder		
Owner	Group	Everyone		Owner	Group	Everyone		Owner	Group	Everyone
RWX	RWX	___		RWX	RWX	___		RWX	R_X	___
7	7	0		7	7	0		7	5	0
770				770				750		

Figure 21: Octal Representation

These folder permissions can be set up by changing the folder owner using “chown <user> <folder>” changing the group owner using “chown :<group> <folder>”. Full permissions can then be changed by using “chmod <octal> <folder>” and permissions can be checked by running the command “ls -l” which will list the folder owners and permissions. An example of this can be found in figure 22.



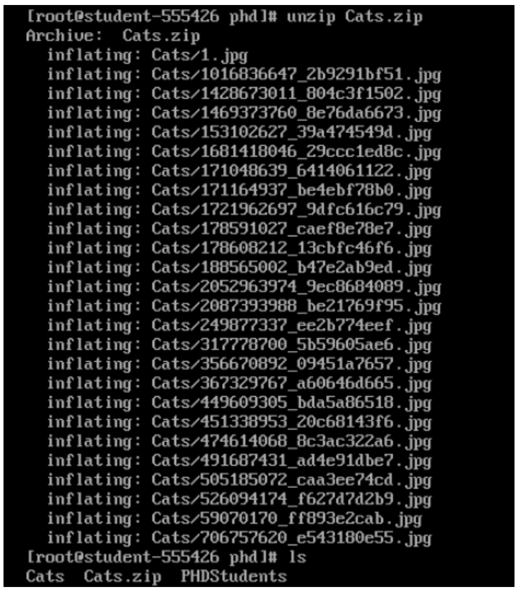
```
[root@student-555426 srv]# ls -l
total 16
dr-xr-xr-x 2 root       ftp        4096 Nov 13 16:23 ftp
drwxrwx--- 2 root       marketing  4096 Apr 16 11:15 http
drwxr-x--- 3 johnmurray phd        4096 Apr 10 12:07 phd
drwxrwx--- 2 johnmurray third_year 4096 Apr 10 10:55 steve_project
```

Figure 22: Folder Permssions

These folder permissions fulfil all of the requirements for the system specification and you can find in the appendix tests to show how each user can access these folders. Additionally, it is worth noting that files can also be given these permissions so when the file owner creates a file they can ensure that permissions are set to make sure no user modifies the data. John may wish to use this for the PHD students and Steve.

Uploading Research Material

John Murray has some research material to upload to the PHD users folder, this can be done in the same way that the public keys were uploaded using station307 (Station307, 2020) and “wget” however this only takes a singular file so John will have to send a zip file to the system. In order to unzip this “unzip” will have to be installed using “pacman -Sy unzip” this can then be used to unzip all of the files from the zip file sent through wget as shown in figure 23. John may not wish to use this mode for all research so he could also use sftp much like Laura does for future uploads.



```
[root@student-555426 phd]# unzip Cats.zip
Archive:  Cats.zip
  inflating: Cats/1.jpg
  inflating: Cats/1016836647_2b9291bf51.jpg
  inflating: Cats/1428673011_804c3f1502.jpg
  inflating: Cats/1469373760_8e76da6673.jpg
  inflating: Cats/153102627_39a474549d.jpg
  inflating: Cats/1681418046_29ccc1ed8c.jpg
  inflating: Cats/171048639_6414061122.jpg
  inflating: Cats/171164937_be4ebf78b0.jpg
  inflating: Cats/1721962697_9dfc616c79.jpg
  inflating: Cats/178591027_caef8e78e7.jpg
  inflating: Cats/178608212_13cbfc46f6.jpg
  inflating: Cats/188565002_b47e2ab9ed.jpg
  inflating: Cats/2052963974_9ec8684089.jpg
  inflating: Cats/2087393988_be21769f95.jpg
  inflating: Cats/249877337_ee2b774eef.jpg
  inflating: Cats/317778700_5b59605ae6.jpg
  inflating: Cats/356670892_09451a7657.jpg
  inflating: Cats/367329767_a60646d665.jpg
  inflating: Cats/449609305_bda5a86518.jpg
  inflating: Cats/451338953_20c68143f6.jpg
  inflating: Cats/474614068_8c3ac322a6.jpg
  inflating: Cats/491687431_ad4e91dbe7.jpg
  inflating: Cats/505185072_caa3ee74cd.jpg
  inflating: Cats/526094174_f627d7d2b9.jpg
  inflating: Cats/59070170_ff893e2cab.jpg
  inflating: Cats/706757620_e543180e55.jpg
[root@student-555426 phd]# ls
Cats  Cats.zip  PHDStudents
```

Figure 23: Unzip

Critical Reflection & Conclusion

In conclusion setting up this system was fun, as finding different solutions to the problems I faced such as getting public keys onto the system challenged me to find an alternative way around using "wget" and Station307s temporary file hosting. There are some small things that I left unchanged for convenience however, I suggest these should be changed if the system was deployed. These were small things such as the "wheel" group not being required to enter a password for sudo, this is potentially very insecure if someone were able to access these accounts they could cause major damage to the system so they should be password protected. It was also nice looking at how to build this system from the bottom up and build in good security practices. Testing for all these steps can also be found in the appendix and tests for public key authentication and host-based authentication although I was not personally able to test them were tested by Ashley with Discord screenshots provided. Any bugs that were encountered were first searched by looking at the manual pages, if a solution was not found Google was used to narrow down the possibilities before consulting Ashley about particular issues. This was particularly relevant when attempting to enable host-based authentication.

Bibliography

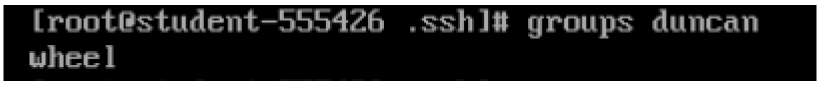
Hull University . (2020, Febuary 11). *Hull Univeristy Password Policy*. Retrieved from Hull University : https://www.hull.ac.uk/editor-assets/docs/password-policy.pdf

Random Keygen. (n.d.). *Random Keygen*. Retrieved from Random Keygen: https://randomkeygen.com/

Station307. (2020). *Station307*. Retrieved from Station307: https://www.station307.com/#/

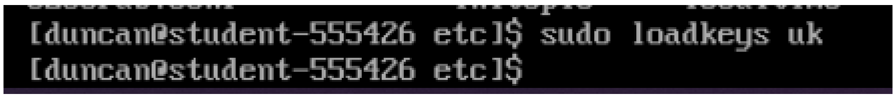
Appendix

Check if admins are part of wheel



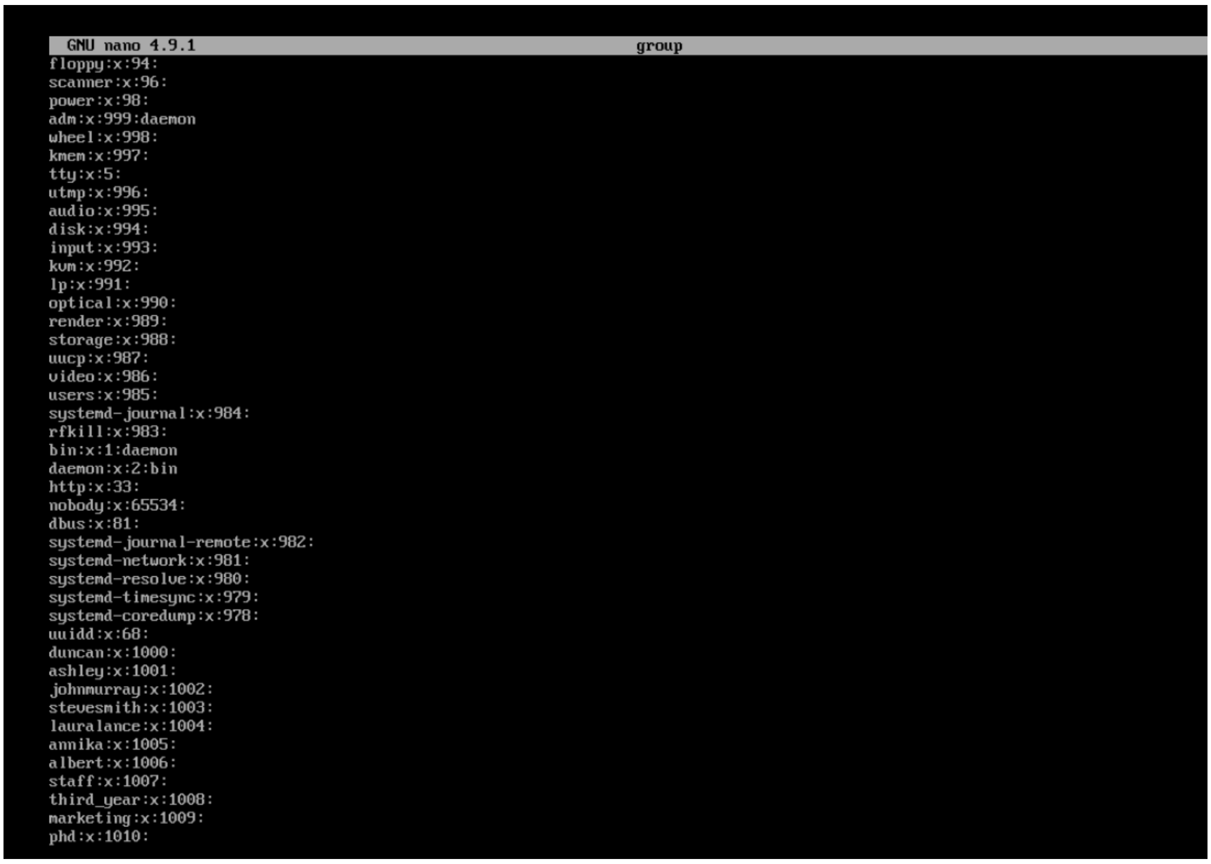
```
[root@student-555426 .ssh]# groups duncan
wheel
```

Validation of admin sudo



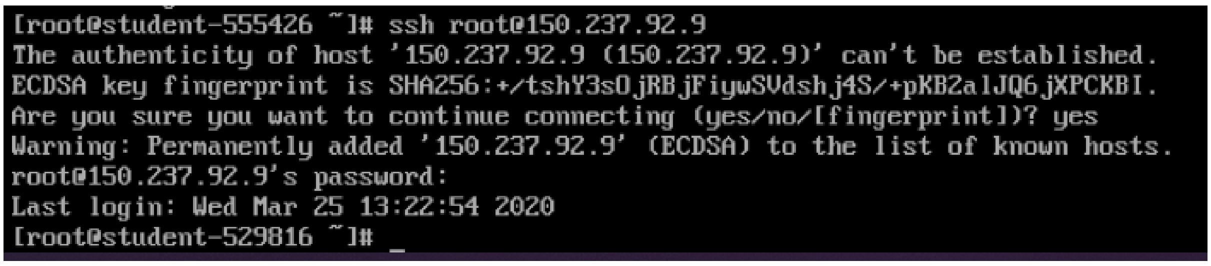
```
[duncan@student-555426 etc]$ sudo loadkeys uk
[duncan@student-555426 etc]$
```

Validation of Created Groups & Users



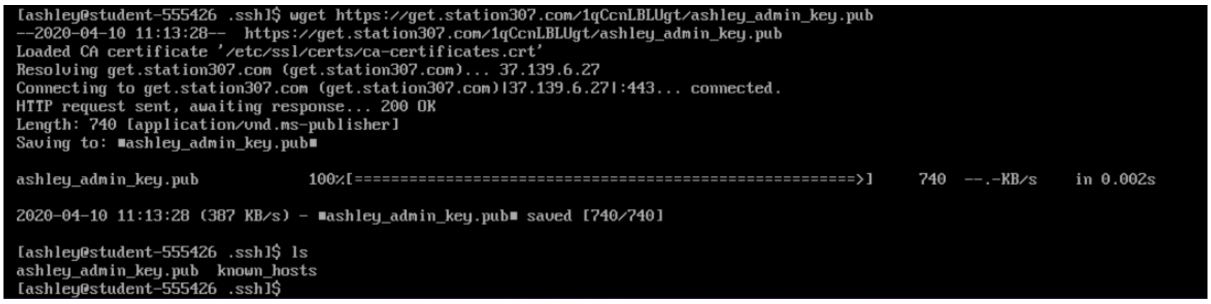
```
  GNU nano 4.9.1                                    group
floppy:x:94:
scanner:x:96:
power:x:98:
adm:x:999:daemon
wheel:x:998:
kmem:x:997:
tty:x:5:
utmp:x:996:
audio:x:995:
disk:x:994:
input:x:993:
kvm:x:992:
lp:x:991:
optical:x:990:
render:x:989:
storage:x:988:
uucp:x:987:
video:x:986:
users:x:985:
systemd-journal:x:984:
rfkill:x:983:
bin:x:1:daemon
daemon:x:2:bin
http:x:33:
nobody:x:65534:
dbus:x:81:
systemd-journal-remote:x:982:
systemd-network:x:981:
systemd-resolve:x:980:
systemd-timesync:x:979:
systemd-coredump:x:978:
uuidd:x:68:
duncan:x:1000:
ashley:x:1001:
johnmurray:x:1002:
stevesmith:x:1003:
lauralance:x:1004:
annika:x:1005:
albert:x:1006:
staff:x:1007:
third_year:x:1008:
marketing:x:1009:
phd:x:1010:
```

SSH Connections to other VMs

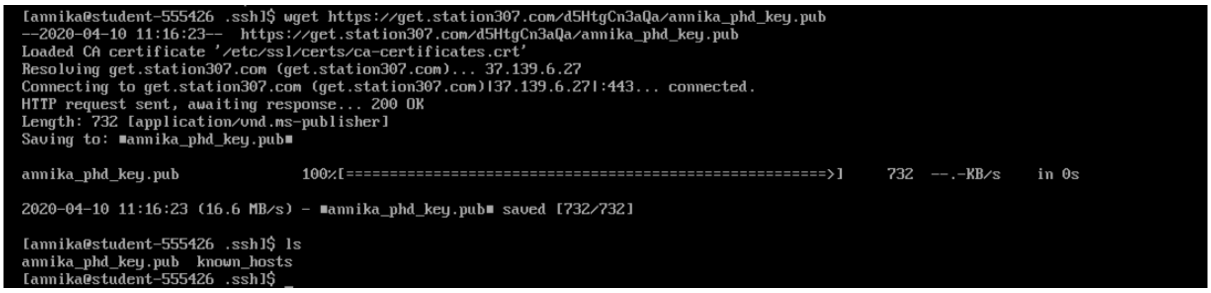


```
[root@student-555426 ~]# ssh root@150.237.92.9
The authenticity of host '150.237.92.9 (150.237.92.9)' can't be established.
ECDSA key fingerprint is SHA256:+/tshY3sOjRBjFiywSVdshj4S/+pKBZalJQ6jXPCKBI.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '150.237.92.9' (ECDSA) to the list of known hosts.
root@150.237.92.9's password:
Last login: Wed Mar 25 13:22:54 2020
[root@student-529816 ~]# _
```

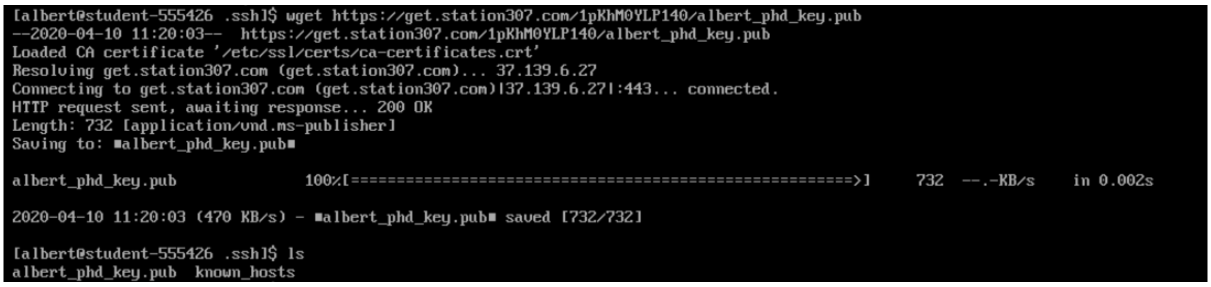
Downloads of all public keys



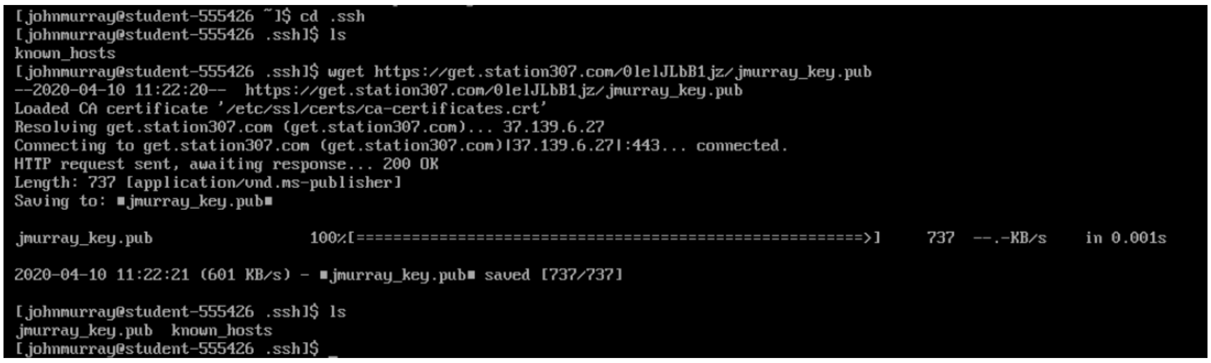
```
[ashley@student-555426 .ssh]$ wget https://get.station307.com/1qCcnLBLUgt/ashley_admin_key.pub
--2020-04-10 11:13:28--  https://get.station307.com/1qCcnLBLUgt/ashley_admin_key.pub
Loaded CA certificate '/etc/ssl/certs/ca-certificates.crt'
Resolving get.station307.com (get.station307.com)... 37.139.6.27
Connecting to get.station307.com (get.station307.com)|37.139.6.27|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 740 [application/vnd.ms-publisher]
Saving to: ■ashley_admin_key.pub■

ashley_admin_key.pub          100%[=====================================================>]     740  --.-KB/s    in 0.002s

2020-04-10 11:13:28 (387 KB/s) - ■ashley_admin_key.pub■ saved [740/740]

[ashley@student-555426 .ssh]$ ls
ashley_admin_key.pub  known_hosts
[ashley@student-555426 .ssh]$
```



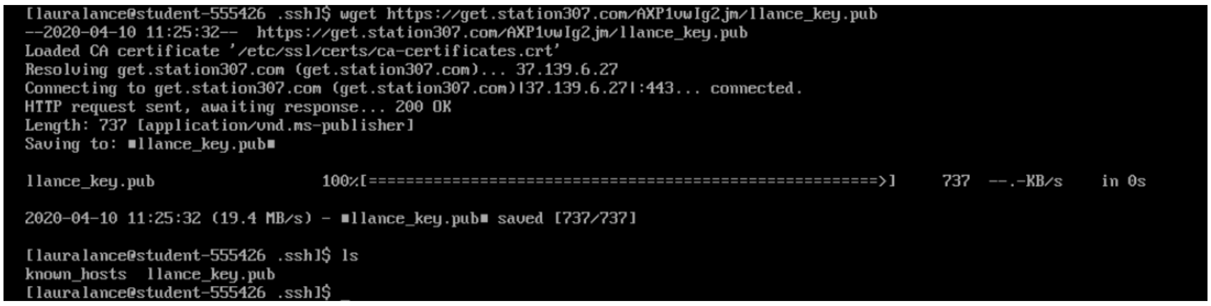
```
[annika@student-555426 .ssh]$ wget https://get.station307.com/d5HtgCn3aQa/annika_phd_key.pub
--2020-04-10 11:16:23--  https://get.station307.com/d5HtgCn3aQa/annika_phd_key.pub
Loaded CA certificate '/etc/ssl/certs/ca-certificates.crt'
Resolving get.station307.com (get.station307.com)... 37.139.6.27
Connecting to get.station307.com (get.station307.com)|37.139.6.27|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 732 [application/vnd.ms-publisher]
Saving to: ■annika_phd_key.pub■

annika_phd_key.pub            100%[=====================================================>]     732  --.-KB/s    in 0s

2020-04-10 11:16:23 (16.6 MB/s) - ■annika_phd_key.pub■ saved [732/732]

[annika@student-555426 .ssh]$ ls
annika_phd_key.pub  known_hosts
[annika@student-555426 .ssh]$ _
```



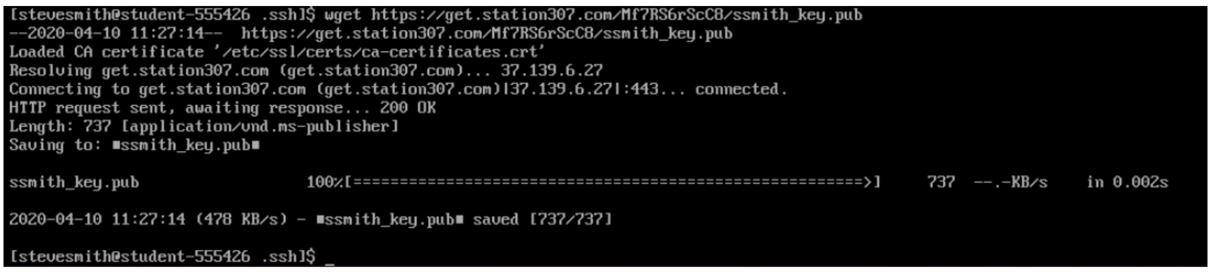
```
[albert@student-555426 .ssh]$ wget https://get.station307.com/1pKhM0YLP140/albert_phd_key.pub
--2020-04-10 11:20:03--  https://get.station307.com/1pKhM0YLP140/albert_phd_key.pub
Loaded CA certificate '/etc/ssl/certs/ca-certificates.crt'
Resolving get.station307.com (get.station307.com)... 37.139.6.27
Connecting to get.station307.com (get.station307.com)|37.139.6.27|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 732 [application/vnd.ms-publisher]
Saving to: ■albert_phd_key.pub■

albert_phd_key.pub            100%[=====================================================>]     732  --.-KB/s    in 0.002s

2020-04-10 11:20:03 (470 KB/s) - ■albert_phd_key.pub■ saved [732/732]

[albert@student-555426 .ssh]$ ls
albert_phd_key.pub  known_hosts
```



```
[johnmurray@student-555426 ~]$ cd .ssh
[johnmurray@student-555426 .ssh]$ ls
known_hosts
[johnmurray@student-555426 .ssh]$ wget https://get.station307.com/0lelJLbB1jz/jmurray_key.pub
--2020-04-10 11:22:20--  https://get.station307.com/0lelJLbB1jz/jmurray_key.pub
Loaded CA certificate '/etc/ssl/certs/ca-certificates.crt'
Resolving get.station307.com (get.station307.com)... 37.139.6.27
Connecting to get.station307.com (get.station307.com)|37.139.6.27|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 737 [application/vnd.ms-publisher]
Saving to: ■jmurray_key.pub■

jmurray_key.pub               100%[=====================================================>]     737  --.-KB/s    in 0.001s

2020-04-10 11:22:21 (601 KB/s) - ■jmurray_key.pub■ saved [737/737]

[johnmurray@student-555426 .ssh]$ ls
jmurray_key.pub  known_hosts
[johnmurray@student-555426 .ssh]$ _
```

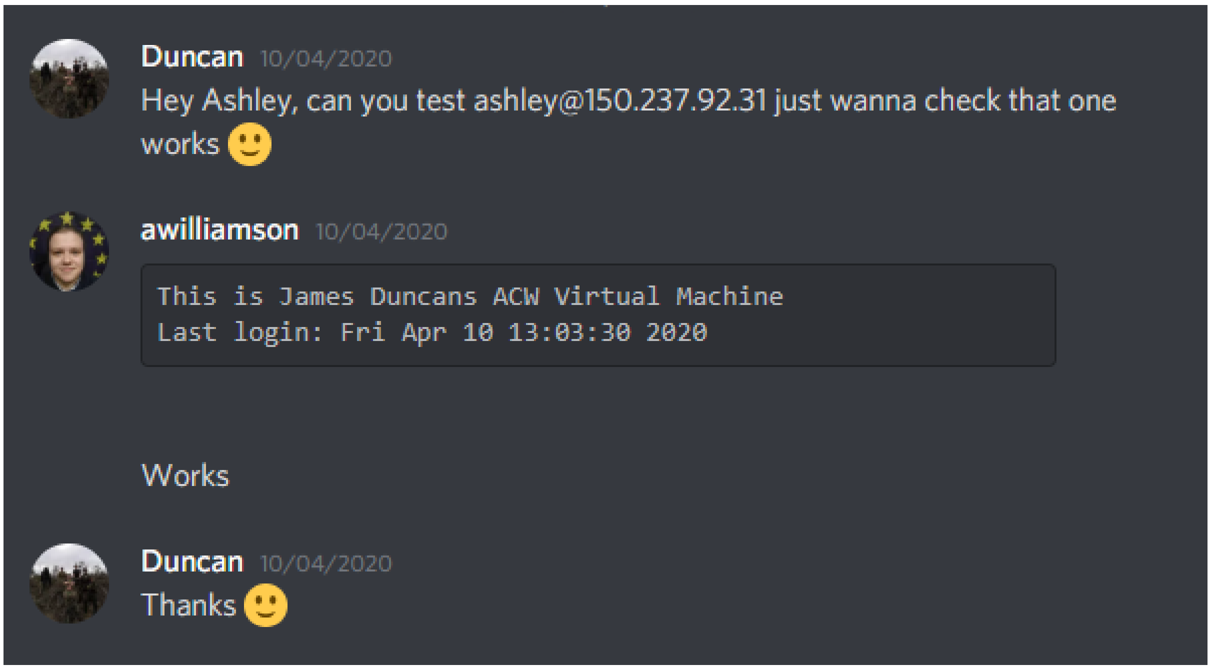


```
[lauralance@student-555426 .ssh]$ wget https://get.station307.com/AXP1vwIg2jm/llance_key.pub
--2020-04-10 11:25:32--  https://get.station307.com/AXP1vwIg2jm/llance_key.pub
Loaded CA certificate '/etc/ssl/certs/ca-certificates.crt'
Resolving get.station307.com (get.station307.com)... 37.139.6.27
Connecting to get.station307.com (get.station307.com)|37.139.6.27|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 737 [application/vnd.ms-publisher]
Saving to: ■llance_key.pub■

llance_key.pub                100%[=====================================================>]     737  --.-KB/s    in 0s

2020-04-10 11:25:32 (19.4 MB/s) - ■llance_key.pub■ saved [737/737]

[lauralance@student-555426 .ssh]$ ls
known_hosts  llance_key.pub
[lauralance@student-555426 .ssh]$ _
```



```
[stevesmith@student-555426 .ssh]$ wget https://get.station307.com/Mf7RS6rScC8/ssmith_key.pub
--2020-04-10 11:27:14--  https://get.station307.com/Mf7RS6rScC8/ssmith_key.pub
Loaded CA certificate '/etc/ssl/certs/ca-certificates.crt'
Resolving get.station307.com (get.station307.com)... 37.139.6.27
Connecting to get.station307.com (get.station307.com)|37.139.6.27|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 737 [application/vnd.ms-publisher]
Saving to: ■ssmith_key.pub■

ssmith_key.pub             100%[=====================================================>]     737  --.-KB/s    in 0.002s

2020-04-10 11:27:14 (478 KB/s) - ■ssmith_key.pub■ saved [737/737]

[stevesmith@student-555426 .ssh]$ _
```

Public Key Authentication (Connection Tested by Ashley, confirmed on Discord)



Duncan 10/04/2020
Hey Ashley, can you test ashley@150.237.92.31 just wanna check that one works 🙂

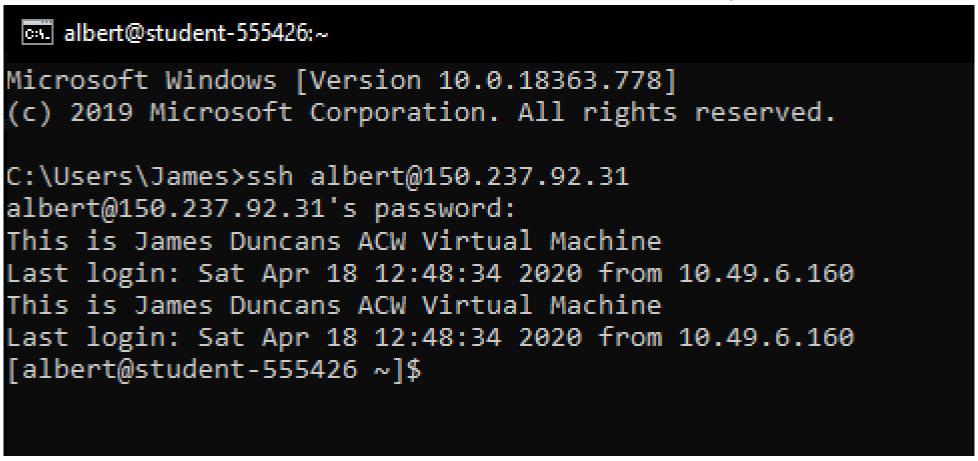
awilliamson 10/04/2020

```
This is James Duncans ACW Virtual Machine
Last login: Fri Apr 10 13:03:30 2020
```

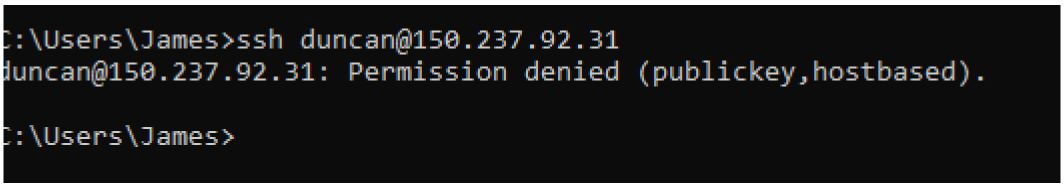
Works

Duncan 10/04/2020
Thanks 🙂

Host Based Password Authentication (local IP Through VPN Tests)

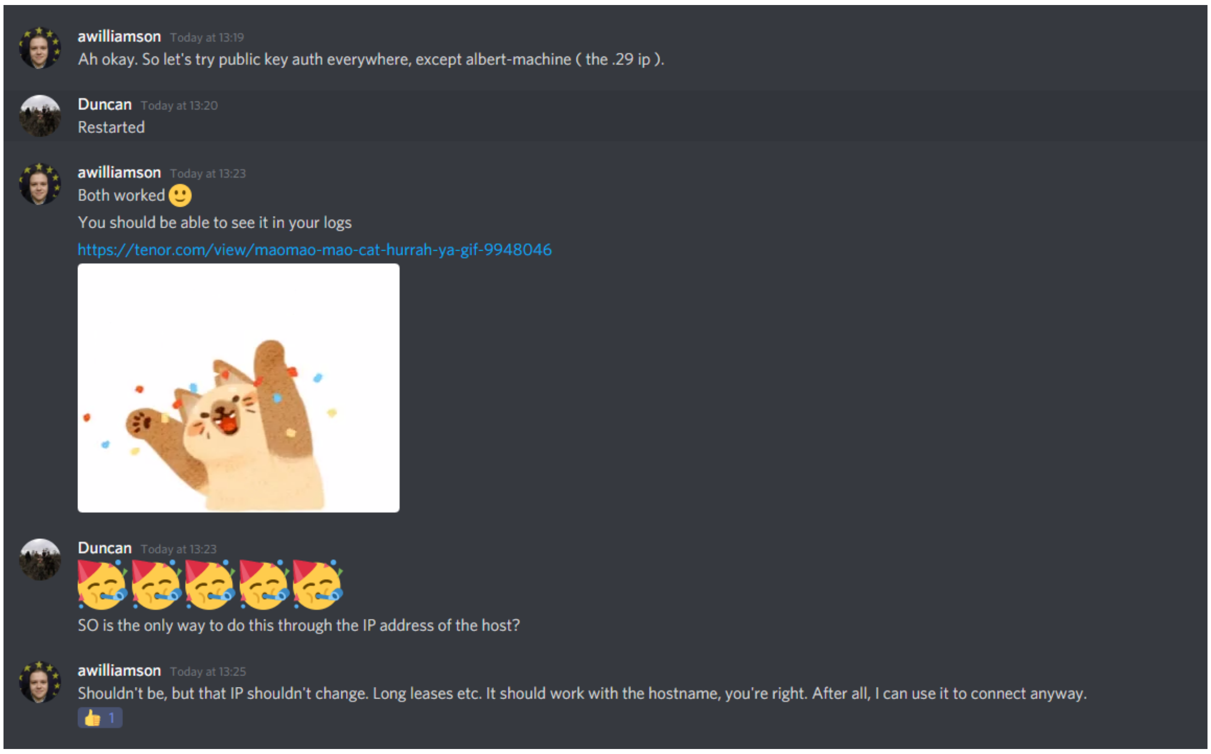


```
Microsoft Windows [Version 10.0.18363.778]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\James>ssh albert@150.237.92.31
albert@150.237.92.31's password:
This is James Duncans ACW Virtual Machine
Last login: Sat Apr 18 12:48:34 2020 from 10.49.6.160
This is James Duncans ACW Virtual Machine
Last login: Sat Apr 18 12:48:34 2020 from 10.49.6.160
[albert@student-555426 ~]$
```



```
:\Users\James>ssh duncan@150.237.92.31
uncan@150.237.92.31: Permission denied (publickey,hostbased).

:\Users\James>
```

Host Based Password Authentication (Connection Tested by Ashley on Discord)



awilliamson Today at 13:19
Ah okay. So let's try public key auth everywhere, except albert-machine (the .29 ip).

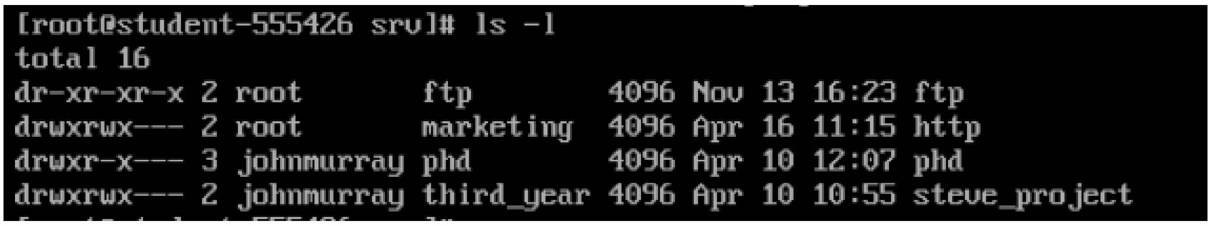
Duncan Today at 13:20
Restarted

awilliamson Today at 13:23
Both worked 🙂
You should be able to see it in your logs
https://tenor.com/view/maomao-mao-cat-hurrah-ya-gif-9948046

Duncan Today at 13:23
🥳🥳🥳🥳🥳
SO is the only way to do this through the IP address of the host?

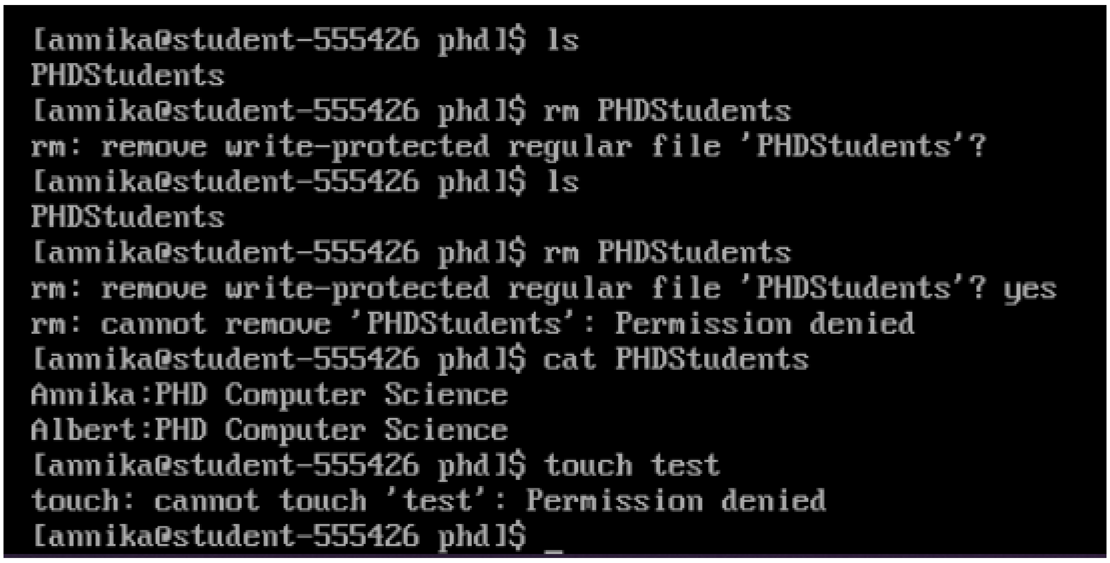
awilliamson Today at 13:25
Shouldn't be, but that IP shouldn't change. Long leases etc. It should work with the hostname, you're right. After all, I can use it to connect anyway.
👍 1

Folder Permissions

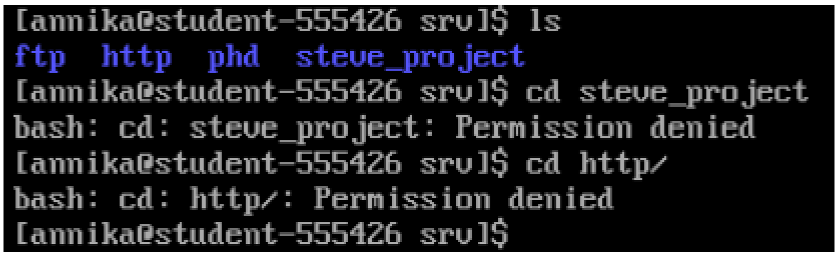


```
[root@student-555426 srv]# ls -l
total 16
dr-xr-xr-x 2 root       ftp         4096 Nov 13 16:23 ftp
drwxrwx--- 2 root       marketing   4096 Apr 16 11:15 http
drwxr-x--- 3 johnmurray phd         4096 Apr 10 12:07 phd
drwxrwx--- 2 johnmurray third_year  4096 Apr 10 10:55 steve_project
```

Annika Tests

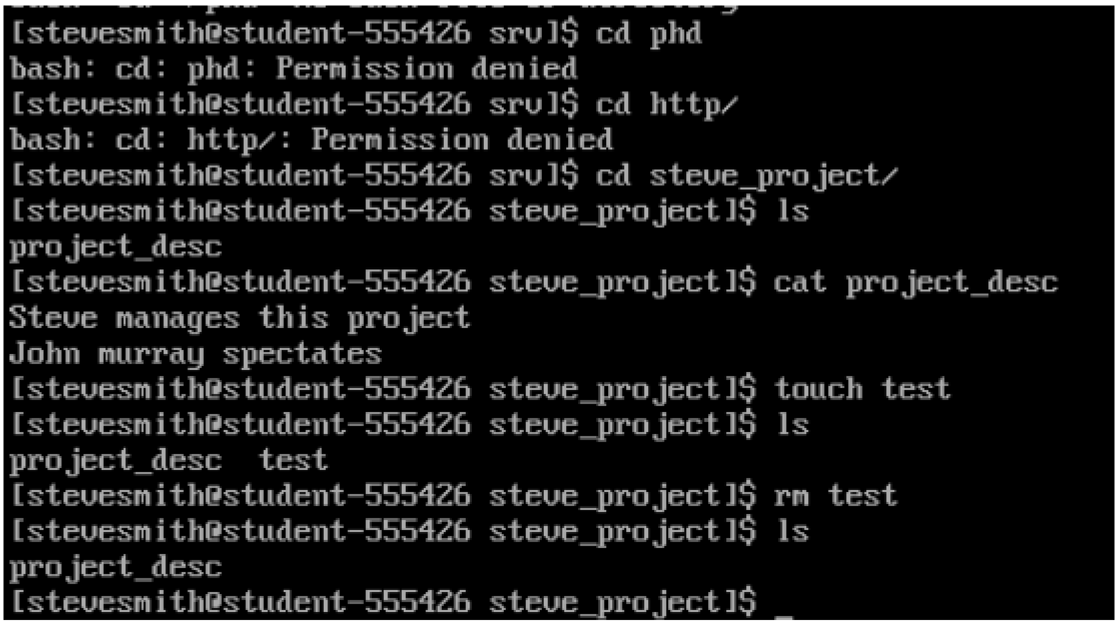


```
[annika@student-555426 phd]$ ls
PHDStudents
[annika@student-555426 phd]$ rm PHDStudents
rm: remove write-protected regular file 'PHDStudents'?
[annika@student-555426 phd]$ ls
PHDStudents
[annika@student-555426 phd]$ rm PHDStudents
rm: remove write-protected regular file 'PHDStudents'? yes
rm: cannot remove 'PHDStudents': Permission denied
[annika@student-555426 phd]$ cat PHDStudents
Annika:PHD Computer Science
Albert:PHD Computer Science
[annika@student-555426 phd]$ touch test
touch: cannot touch 'test': Permission denied
[annika@student-555426 phd]$ _
```



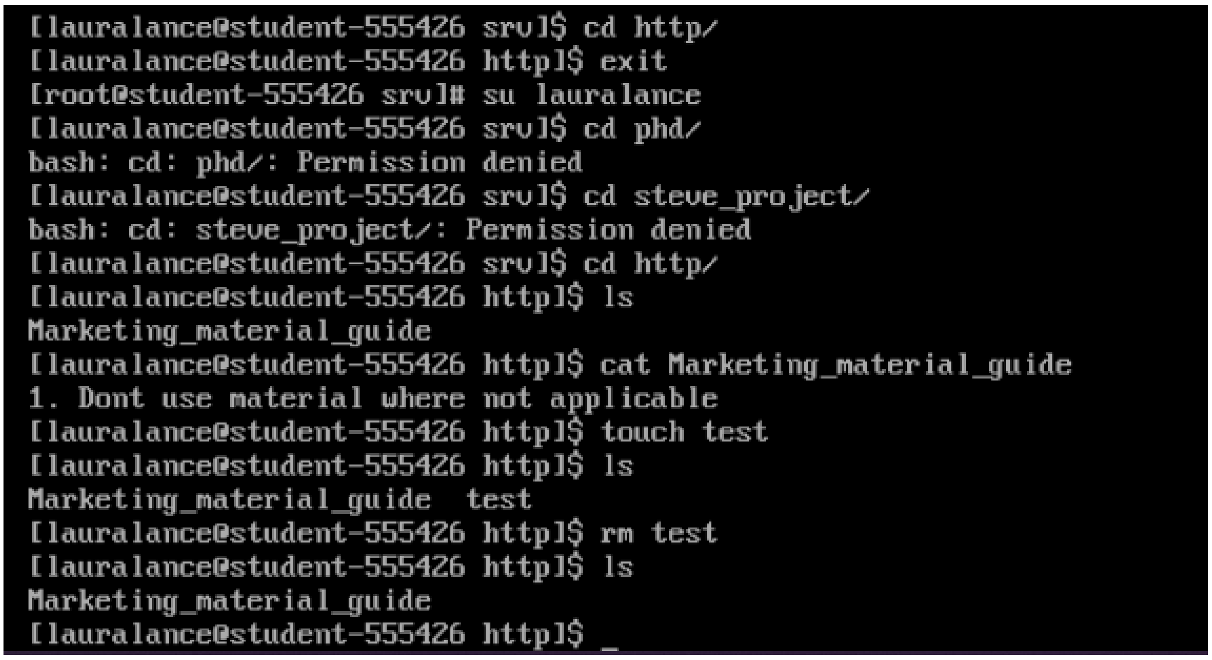
```
[annika@student-555426 srv]$ ls
ftp  http  phd  steve_project
[annika@student-555426 srv]$ cd steve_project
bash: cd: steve_project: Permission denied
[annika@student-555426 srv]$ cd http/
bash: cd: http/: Permission denied
[annika@student-555426 srv]$
```

Steve Tests



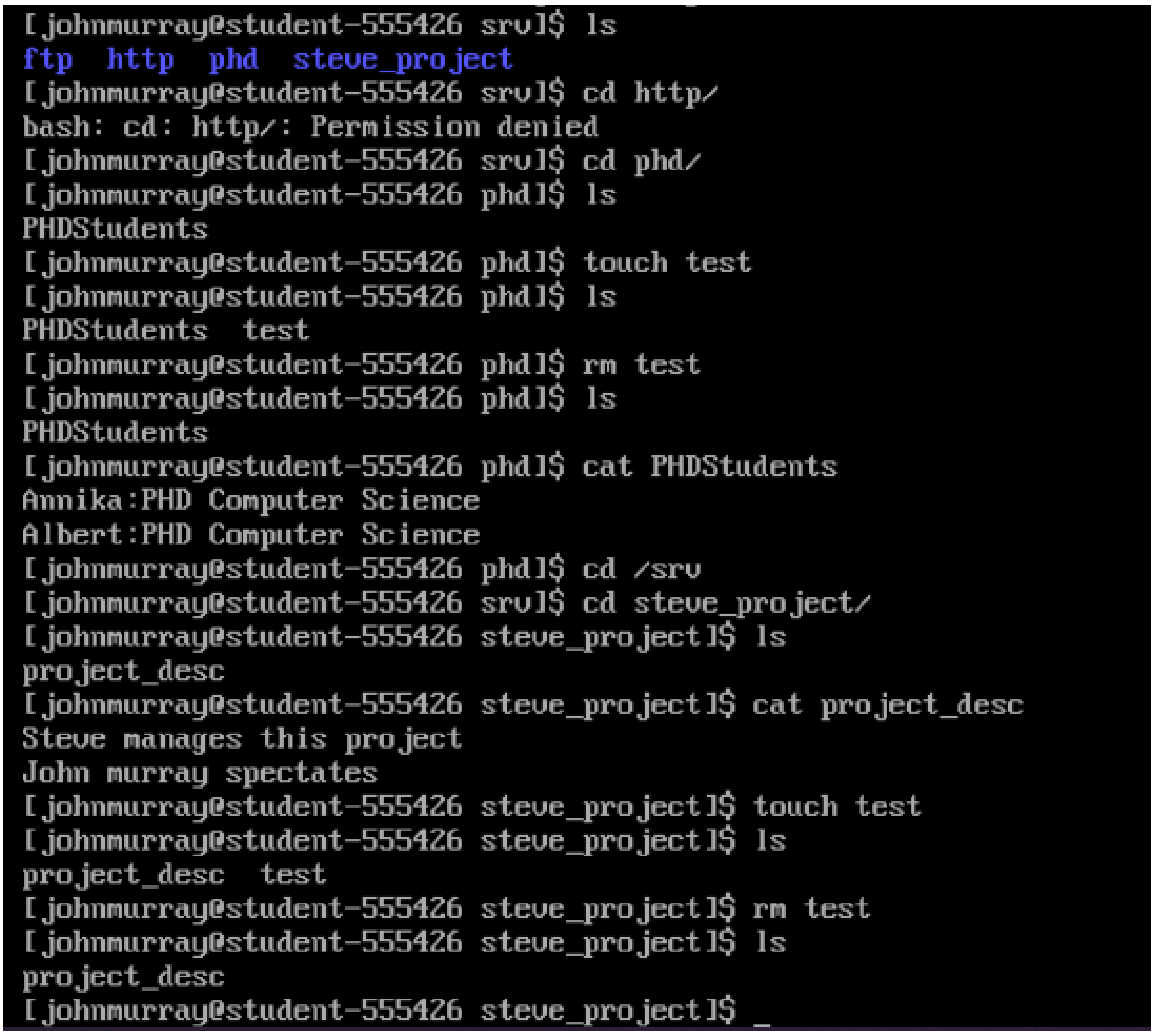
```
[stevesmith@student-555426 srv]$ cd phd
bash: cd: phd: Permission denied
[stevesmith@student-555426 srv]$ cd http/
bash: cd: http/: Permission denied
[stevesmith@student-555426 srv]$ cd steve_project/
[stevesmith@student-555426 steve_project]$ ls
project_desc
[stevesmith@student-555426 steve_project]$ cat project_desc
Steve manages this project
John murray spectates
[stevesmith@student-555426 steve_project]$ touch test
[stevesmith@student-555426 steve_project]$ ls
project_desc  test
[stevesmith@student-555426 steve_project]$ rm test
[stevesmith@student-555426 steve_project]$ ls
project_desc
[stevesmith@student-555426 steve_project]$ _
```

Laura Lance Tests



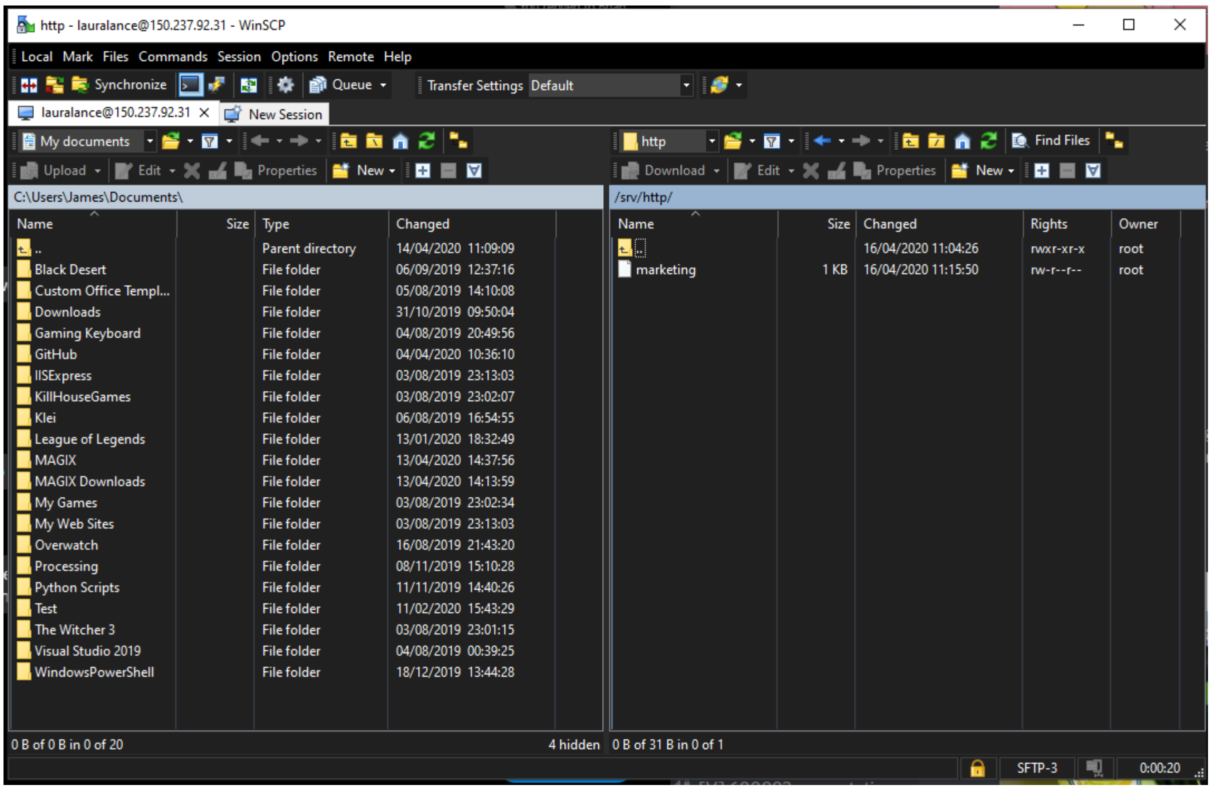
```
[lauralance@student-555426 srv]$ cd http/
[lauralance@student-555426 http]$ exit
[root@student-555426 srv]# su lauralance
[lauralance@student-555426 srv]$ cd phd/
bash: cd: phd/: Permission denied
[lauralance@student-555426 srv]$ cd steve_project/
bash: cd: steve_project/: Permission denied
[lauralance@student-555426 srv]$ cd http/
[lauralance@student-555426 http]$ ls
Marketing_material_guide
[lauralance@student-555426 http]$ cat Marketing_material_guide
1. Dont use material where not applicable
[lauralance@student-555426 http]$ touch test
[lauralance@student-555426 http]$ ls
Marketing_material_guide  test
[lauralance@student-555426 http]$ rm test
[lauralance@student-555426 http]$ ls
Marketing_material_guide
[lauralance@student-555426 http]$ _
```

John Tests



```
[johnmurray@student-555426 srv]$ ls
ftp  http  phd  steve_project
[johnmurray@student-555426 srv]$ cd http/
bash: cd: http/: Permission denied
[johnmurray@student-555426 srv]$ cd phd/
[johnmurray@student-555426 phd]$ ls
PHDStudents
[johnmurray@student-555426 phd]$ touch test
[johnmurray@student-555426 phd]$ ls
PHDStudents  test
[johnmurray@student-555426 phd]$ rm test
[johnmurray@student-555426 phd]$ ls
PHDStudents
[johnmurray@student-555426 phd]$ cat PHDStudents
Annika:PHD Computer Science
Albert:PHD Computer Science
[johnmurray@student-555426 phd]$ cd /srv
[johnmurray@student-555426 srv]$ cd steve_project/
[johnmurray@student-555426 steve_project]$ ls
project_desc
[johnmurray@student-555426 steve_project]$ cat project_desc
Steve manages this project
John murray spectates
[johnmurray@student-555426 steve_project]$ touch test
[johnmurray@student-555426 steve_project]$ ls
project_desc  test
[johnmurray@student-555426 steve_project]$ rm test
[johnmurray@student-555426 steve_project]$ ls
project_desc
[johnmurray@student-555426 steve_project]$ _
```

Laura Lance SFTP via Windows UI using WINSCP

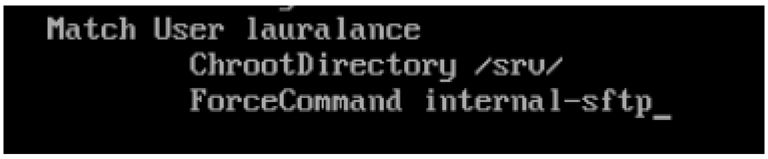


```
C:\Users\James\Documents\

Name                      Size  Type              Changed
..                              Parent directory  14/04/2020 11:09:09
Black Desert                    File folder       06/09/2019 12:37:16
Custom Office Templ...          File folder       05/08/2019 14:10:08
Downloads                       File folder       31/10/2019 09:50:04
Gaming Keyboard                 File folder       04/08/2019 20:49:56
GitHub                          File folder       04/04/2020 10:36:10
IISExpress                      File folder       03/08/2019 23:13:03
KillHouseGames                  File folder       03/08/2019 23:02:07
Klei                            File folder       06/08/2019 16:54:55
League of Legends               File folder       13/01/2020 18:32:49
MAGIX                           File folder       13/04/2020 14:37:56
MAGIX Downloads                 File folder       13/04/2020 14:13:59
My Games                        File folder       03/08/2019 23:02:34
My Web Sites                    File folder       03/08/2019 23:13:03
Overwatch                       File folder       16/08/2019 21:43:20
Processing                      File folder       08/11/2019 15:10:28
Python Scripts                  File folder       11/11/2019 14:40:26
Test                            File folder       11/02/2020 15:43:29
The Witcher 3                   File folder       03/08/2019 23:01:15
Visual Studio 2019              File folder       04/08/2019 00:39:25
WindowsPowerShell               File folder       18/12/2019 13:44:28

0 B of 0 B in 0 of 20                                   4 hidden
```

```
/srv/http/

Name          Size  Changed              Rights      Owner
..                  16/04/2020 11:04:26  rwxr-xr-x   root
marketing     1 KB  16/04/2020 11:15:50  rw-r--r--   root

0 B of 31 B in 0 of 1
```

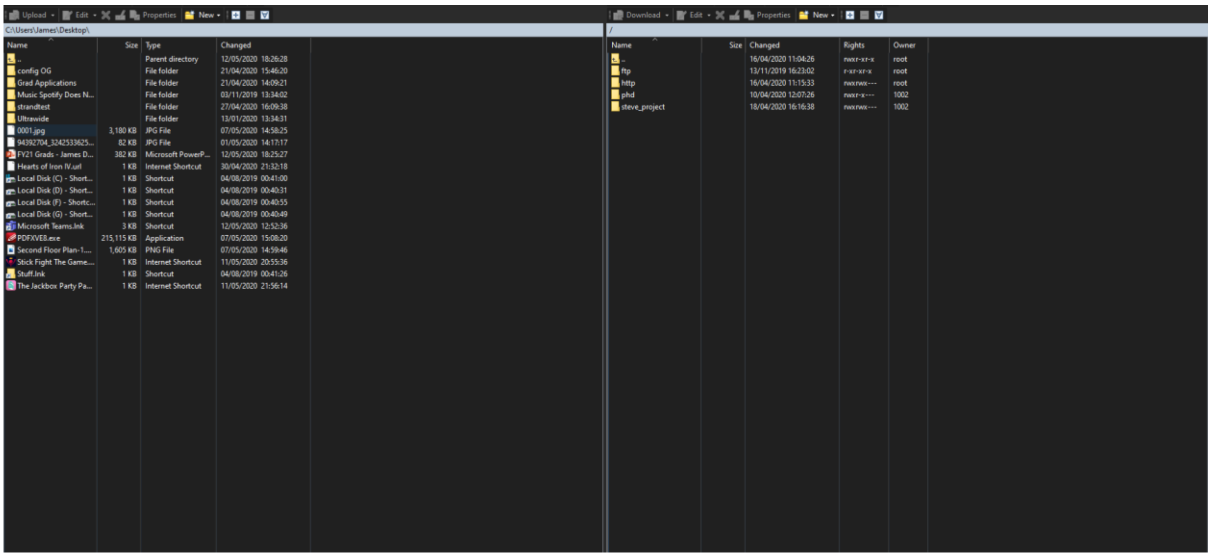
Jailed Laura Lance SFTP

Jailed code



```
Match User lauralance
        ChrootDirectory /srv/
        ForceCommand internal-sftp_
```

Jailed Screenshot



```
C:\Users\James\Desktop\

Name                       Size       Type               Changed
..                                    Parent directory   12/05/2020 18:26:28
config OG                             File folder        21/04/2020 15:46:20
Grad Applications                     File folder        21/04/2020 14:09:21
Music Spotify Does N...               File folder        03/11/2019 13:34:02
strandtest                            File folder        27/04/2020 16:09:38
Ultrawide                             File folder        13/01/2020 13:34:31
0001.jpg                   3,180 KB   JPG File           07/05/2020 14:58:25
94392704_3242533625...        82 KB   JPG File           01/05/2020 14:17:17
FY21 Grads - James D...      382 KB   Microsoft PowerP...  12/05/2020 18:25:27
Hearts of Iron IV.url          1 KB   Internet Shortcut  30/04/2020 21:32:18
Local Disk (C) - Short...      1 KB   Shortcut           04/08/2019 00:41:00
Local Disk (D) - Short...      1 KB   Shortcut           04/08/2019 00:40:31
Local Disk (F) - Shortc...     1 KB   Shortcut           04/08/2019 00:40:55
Local Disk (G) - Short...      1 KB   Shortcut           04/08/2019 00:40:49
Microsoft Teams.lnk            3 KB   Shortcut           12/05/2020 12:52:36
PDFXVE8.exe              215,115 KB   Application        07/05/2020 15:08:20
Second Floor Plan-1....    1,605 KB   PNG File           07/05/2020 14:59:46
Stick Fight The Game....       1 KB   Internet Shortcut  11/05/2020 20:55:36
Stuff.lnk                      1 KB   Shortcut           04/08/2019 00:41:26
The Jackbox Party Pa...        1 KB   Internet Shortcut  11/05/2020 21:56:14
```

```
/

Name            Size  Changed              Rights      Owner
..                    16/04/2020 11:04:26  rwxr-xr-x   root
ftp                   13/11/2019 16:23:02  r-xr-xr-x   root
http                  16/04/2020 11:15:33  rwxrwx---   root
phd                   10/04/2020 12:07:26  rwxr-x---   1002
steve_project         18/04/2020 16:16:38  rwxrwx---   1002
```