P6.1 SSH instructions

Part 1: Configuration modifications

To copy the nanorc file in to home directory I use command cp to copy it and place it in my home directory as a hidden file. The exact command was { cp /etc/nanorc /home/ron/.nanorc }. To check if the file has been copied command { ls -a }.
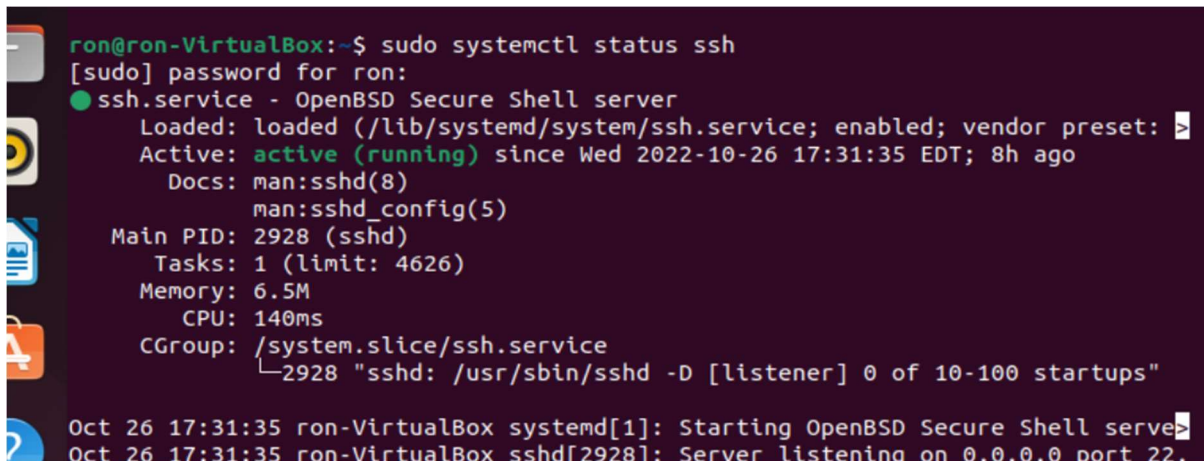
```
ron@ron-VirtualBox:~$ cp /etc/nanorc /home/ron/.nanorc
ron@ron-VirtualBox:~$ ls
copy_test   Downloads   kansio        Music      Public      Templates
Desktop     fbar.c      Linux_course  my_test    snap        Videos
Documents   foo.c       linux_pop     Pictures   sudo_test
ron@ron-VirtualBox:~$ ls -a
.                copy_test   kansio        .nanorc                    sudo_test
..               Desktop     .lesshst      Pictures                   Templates
.bash_history    Documents   Linux_course  .profile                   Videos
.bash_logout     Downloads   linux_pop     Public
.bashrc          fbar.c      .local        snap
.cache           foo.c       Music         .ssh
.config          .gnupg      my_test       .sudo_as_admin_successful
ron@ron-VirtualBox:~$
```

To open and examine our new nanorc file command nano .nanorc opens the new file. Where we can see many comments with ether one or two hashtags. Now the difference between these comment ways do not differ other than it makes it more readable. One hashtag is good for commenting the function itself and two hashtags for the description of said function. For good habits its recommended to add the date and modifier of set feature so it can be changed later with confidence and ease.

```
  GNU nano 6.2                          .nanorc *                              S
71 # set indicator
72
73 ## Scroll the buffer contents per half-screen instead of per line.
74 #ron 26.10.2022
75 #set jumpyscrolling
76   set jumpyscrolling
77
78 ## Display line numbers to the left (and any anchors in the margin).
79 #ron 26.10.2022
80 #set linenumbers
81 set linenumbers
82
83 ## Enable vim-style lock-files.  This is just to let a vim user know you
84 ## are editing a file [s]he is trying to edit and vice versa.  There are
85 ## no plans to implement vim-style undo state in these files.
86 #ron 26.10.2022 felt like it
87 #set locking
88 set locking
```
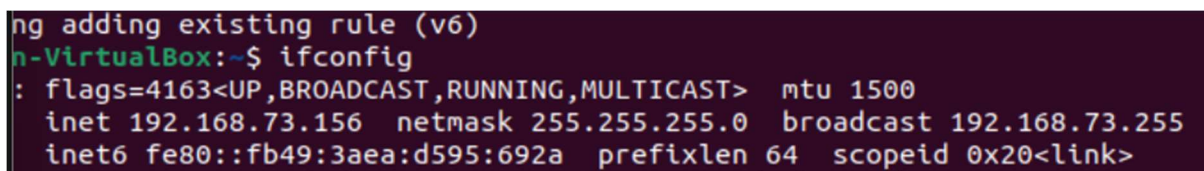
Part 2: SSH-server

Question A: To enable SSH service we first update our machine by using command { sudo apt update } and then istall the SSH server program by command { sudo apt install openssh-server }. After the download you can verify that the SSH is working by giving command { sudo systemctl status ssh } and it should look like this.

To open the SSH port from being blocked by firewall use command { sudo ufw allow ssh }. Now we are set to connect to the virtualmachine. First we find out our machines we have to know the ip address of the virtual machine. Command { ifconfig } does the trick.

Now we connect by going on our host machine to putty by writing on hostname ron@ipaddress and with successful connection putty should look like the image below.

Question B: The configuration file of shh can be found in /etc/ssh/sshd_config. To showcase our skills we will create a banner. First we want to copy the configuration file so if something were to go south we can restart and pretend it never happened. To copy config file to home directory command { cp /etc/ssh/sshd_config /etc/ssh/sshd_config.backup }

Then modify the /etc/ssh/sshd_config file and enable banner and give it a path. I decided to copy an ascii art fom the web. Then restart the server so our modifications can be set with command { sudo service ssh restart }. Then we connect to our local ssh to see the result and our beautiful cat has appeared.

```
ron@ron-VirtualBox:~$ cp /etc/ssh/sshd_config ~/sshd_config.backup
ron@ron-VirtualBox:~$ ls
copy_test   Downloads  kansio       Music     Public              sudo_test
Desktop     fbar.c     Linux_course my_test   snap                Templates
Documents   foo.c      linux_pop    Pictures  sshd_config.backup  Videos
ron@ron-VirtualBox:~$ sudo nano /etc/ssh/sshd_config
ron@ron-VirtualBox:~$ sudo nano /etc/ssh/banner
ron@ron-VirtualBox:~$ sudo service ssh restart
ron@ron-VirtualBox:~$ ssh ron@localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ED25519 key fingerprint is SHA256:l+/IIUzBPIrfR7+ndmZ/9lKliOGGsXuKv8vaktMWpjM.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'localhost' (ED25519) to the list of known hosts.
  /\ ___ /\
 (  o   o  )
  \  >#<  /
  /       \
 /         \      ^
|           |    //
 \         /    //
  ///  ///   --
ron@localhost's password:
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-52-generic x86_64)
```

Allow one user to login remotely

Well the title says it all, we start by creating a new user that will have the rights to access our ssh server remotely. Command { sudo adduser ssh-user } will create our new user.
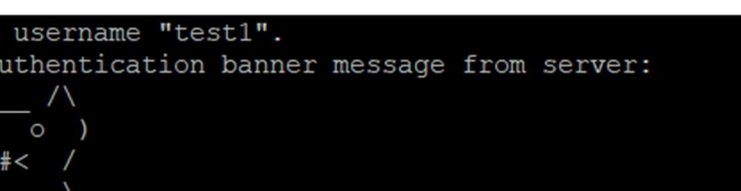
```
ron@ron-VirtualBox:~$ sudo adduser ssh-user
[sudo] password for ron:
Adding user `ssh-user' ...
Adding new group `ssh-user' (1004) ...
Adding new user `ssh-user' (1003) with group `ssh-user' ...
Creating home directory `/home/ssh-user' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for ssh-user
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] y
```

Next we are going to configure the file so our new user can have rights to access the ssh server. We give the following command { sudo nano /etc/ssh/sshd_config }

```
ron@ron-VirtualBox:~$ sudo nano /etc/ssh/sshd_config
ron@ron-VirtualBox:~$ sudo nano /etc/ssh/sshd_config
[sudo] password for ron:
```

At the bottom of the file we add a few lines.

```
#ron 27.10.2022 allow ssh-user to login remotely
PasswordAuthentication yes
AllowUsers        ssh-user
```

And the we restart so the changes can be applied

```
ron@ron-VirtualBox:~$ sudo nano /etc/ssh/sshd_config
ron@ron-VirtualBox:~$ sudo systemctl restart sshd
[sudo] password for ron:
ron@ron-VirtualBox:~$
```

To test that it works I tried to log in as test1 to my ssh server

```
192.168.1.28 - PuTTY
Using username "test1".
Pre-authentication banner message from server:
    /\ ___ /\
   (  o   o  )
    \   >#<   /
    /         \
   /           \        ^
  | |          |      //
  |  \        /     //
  |   /// ///   --
End of banner message from server
test1@192.168.1.28's password:
Access denied
test1@192.168.1.28's password:
```

Success user test1 is not allowed to the server. Let's try if ssh-user is working correctly?

```
 End of banner message from server
 ssh-user@192.168.1.28's password:
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-52-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:      https://landscape.canonical.com
 * Support:         https://ubuntu.com/advantage

28 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

*** System restart required ***

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

ssh-user@ron-VirtualBox:~$
```

Move server to a different port number

The default portnumber for ssh servers is 22 to make it more secure we want to move that port number to 40400. Command { sudo nano /etc/ssh/ssh_config } we can modify the file as in the picture below. Then restart


```
#Port 22
#ron 29.10.2022 chande port to 40400
Port 40400
```

We can test if it works by trying to login with the default port 22(left) and the new port 40400(right) to connect to this new port with linux we must give the parameter -p so to take connection it should be written { ssh ron@ipaddress -p 40400 }


```
192.168.1.30 - PuTTY

Using username "ron".
Pre-authentication banner message from server:
      /\   /\
       )  ( ')
      (  /  )
 ron    \(__)|
End of banner message from server
ron@192.168.1.30's password:
Access denied
```



```
ssh-user@ron-VirtualBox: ~
Using username "ssh-user".
Pre-authentication banner message from server:
      /\   /\
       )  ( ')
      (  /  )
 ron    \(__)|
End of banner message from server
ssh-user@192.168.1.30's password:
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-52-generic x86_6

 * Documentation:  https://help.ubuntu.com
 * Management:      https://landscape.canonical.com
 * Support:         https://ubuntu.com/advantage

0 updates can be applied immediately.

Last login: Sat Oct 29 17:58:51 2022 from 192.168.1.11
ssh-user@ron-VirtualBox:~$
```

X11 Forwarding

Now we will configure x11 forwarding on startup. Form our sshd_config file we enable x11forwarding.

We also enable it from client computers ssh_config file.

After this not much has changed but we need to add a new parameter -X to our command. So the following terminal should go like this.

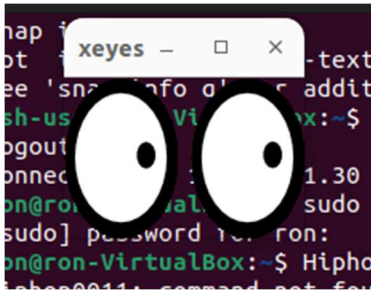When I wrote xeyes a pair of eyes that followed the mouse appeared so that tells us connection is working. Could not get firefox running.

Password less login

For a password less login we need to generate a public key to the server. To generate one we give command ssh-keygen.


```
ron@ron-VirtualBox:/$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ron/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ron/.ssh/id_rsa
Your public key has been saved in /home/ron/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:dvNH+HPUT5s7tB0Owe4TR8NvGPagQlhUfbTqGn2d7bE ron@ron-VirtualBox
The key's randomart image is:
```

After generating the key it must be copied to the servers authorized keys with command { ssh-copy-id -f -p 40400 ssh-user@192.168.1.30 }


```
ron@ron-VirtualBox:~$ ssh-copy-id -f -p 40400 ssh-user@192.168.1.30
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/ron/.ssh/i
d_rsa.pub"
      /\   /\
      )  ( ')
     (  /  )
ron    \(__)|
ssh-user@192.168.1.30's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh -p '40400' 'ssh-user@192.168.1.3
0'"
and check to make sure that only the key(s) you wanted were added.

ron@ron-VirtualBox:~$
```

Log on successful

Question D: password less login means that the server has a public key that allows the user to log to the server without asking for the password because the key has already been authorized.

```
ron@ron-VirtualBox:~$ ssh ssh-user@192.168.1.30 -p 40400
      /\   /\
       ) ( ')
      (  /  )
ron    \(__)|
Warning: No xauth data; using fake authentication data for X11 forwarding.
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-52-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

4 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

*** System restart required ***
Last login: Sun Nov  6 20:57:55 2022 from 192.168.1.30
ssh-user@ron-VirtualBox:~$
```

Ssh config file

For an even faster log to our ssh server we create a configuration file in our .ssh file.
Command { touch ~/.ssh/config} will create the file and nano to the same path we can apply
the following configurations.

```
  ⌐+⌐                              ron@ron-VirtualBox: ~              Q    ≡

    GNU nano 6.2                   /home/ron/.ssh/config *
Host mymachine
     HostName 192.168.1.30
     User ssh-user
     Port 40400
```

To make it work we type ssh and mymachine and it logs on automatically

```
ron@ron-VirtualBox:~$ ssh mymachine
      /\   /\
       ) ( ')
      (  /  )
ron    \(__)|
Warning: No xauth data; using fake authentication data for X11 forwarding
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-52-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

4 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

*** System restart required ***
Last login: Sun Nov  6 21:46:03 2022 from 192.168.1.30
ssh-user@ron-VirtualBox:~$
```

Ssh tunnel

```
connection to shell.puv.fi closed.
ron@ron-VirtualBox:~$ ssh e2101495@shell.puv.fi -L 2020:omega.cc.puv.fi:80
e2101495@shell.puv.fi's password:
Warning: No xauth data; using fake authentication data for X11 forwarding.
Linux shell.vamk.fi 4.9.0-19-686-pae #1 SMP Debian 4.9.320-2 (2022-06-30) i686

Welcome to shell.vamk.fi.

The character encoding is now standard UTF-8.

If you use PuTTY use the latest version, or if you really must use a version
older than 0.63: go to Window/Translation and change to "UTF-8".

Last login: Sun Nov  6 22:05:30 2022 from 83.245.133.206
e2101495@shell:~$
```

Firewall does not give permission to connect but connection to shell.puv.fi was successful

Firewall

Enable firewall with { sudo ufw enable }

```
ron@ron-VirtualBox:~$ sudo ufw enable
[sudo] password for ron:
Firewall is active and enabled on system startup
ron@ron-VirtualBox:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

To                         Action      From
--                         ------      ----
22/tcp                     ALLOW IN    Anywhere
22/tcp (v6)                ALLOW IN    Anywhere (v6)

ron@ron-VirtualBox:~$
```

Allow access on port 40400

```
ron@ron-VirtualBox:~$ sudo ufw allow 40400
Rule added
Rule added (v6)
ron@ron-VirtualBox:~$
```

To see the changes command { sudo iptables -L }

```
Chain ufw-user-input (1 references)
target       prot opt source              destination
ACCEPT       tcp  --  anywhere            anywhere            tcp dpt:ssh
ACCEPT       tcp  --  anywhere            anywhere            tcp dpt:40400
ACCEPT       udp  --  anywhere            anywhere            udp dpt:40400
```