How to set up, connect to, and put VNC on a CU Boulder Cloud Instance

Step 1.

Go to https://cloud.cs.colorado.edu/auth/login/?next=/ and login with your CU Identikey

Step 2.
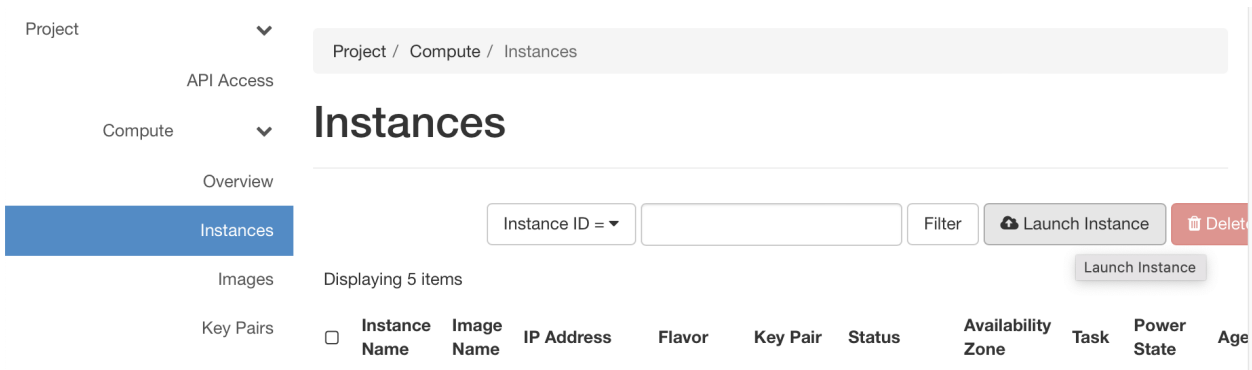
Set your project in the top bar to **Information Science - Recsys**

Step 3.
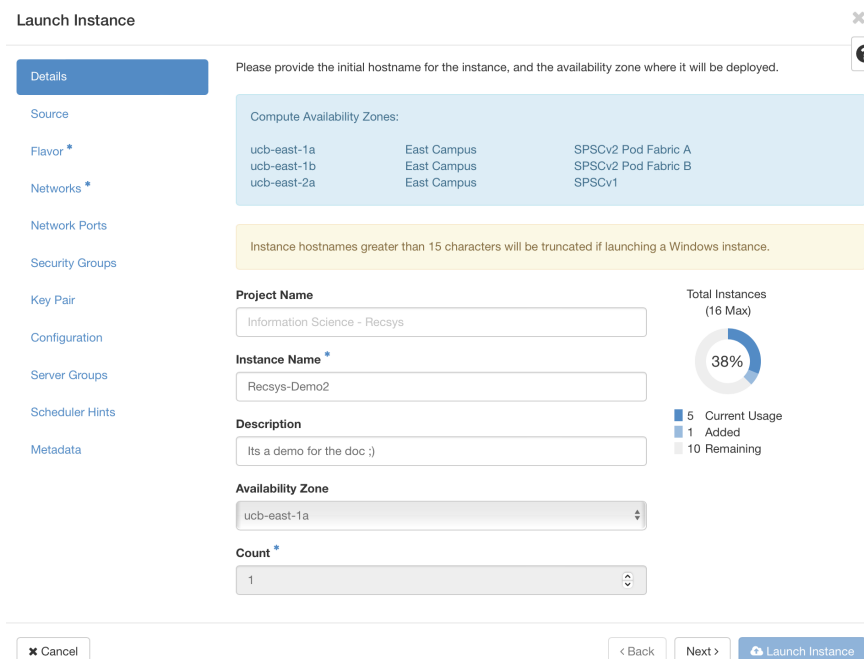
Go to the instances page and launch an instance (Compute -> Instances -> Launch Instance)

Step 4.0.

On the first page of the Launch Instance Menu Set a name and a description for your instance then click Next.

Step 4.1

On the next page set the OS or Source for your instance. Find the Distro you want, then click the up arrow on the right side to select the image you want to boot from. The boot volume size will be set automatically for the distro you choose. Then click next

Launch Instance   ✖

                                                                                          ❓

Details	Instance source is the template used to create an instance. You can use an image, a snapshot of an instance (image snapshot), a volume or a volume snapshot (if enabled). You can also choose to use persistent storage by creating a new volume.
Source	
Flavor *	
Networks *	
Network Ports	
Security Groups	
Key Pair	
Configuration	
Server Groups	
Scheduler Hints	
Metadata	

> The boot volume type will be created as "ceph-gp2" (General Purpose SSD). It is recommended to add additional volumes rather than expanding the boot volume. Additional volumes can be added in the Volumes menu and attached to the instance.

Select Boot Source

Image ⬍

Create New Volume

Yes | No

Volume Size (GB) *

4 ⬍

Delete Volume on Instance Delete

Yes | No

Allocated

Displaying 1 item

Name	Updated	Size	Type	Visibility	
> Ubuntu 21.04	9/18/21 1:41 AM	553.69 MB	QCOW2	Public	⬇

Displaying 1 item

⌄ Available 13   Select one

🔍 Click here for filters or full text search.   ✖

Displaying 13 items

Name	Updated	Size	Type	Visibility	
> CentOS 7	9/18/21 1:40 AM	847.81 MB	QCOW2	Public	⬆
> CentOS 8 Stream	9/18/21 1:40 AM	1.26 GB	QCOW2	Public	⬆
> Debian 10	9/18/21 1:40 AM	222.25 MB	QCOW2	Public	⬆
> Debian 11	9/18/21 1:40 AM	241.31 MB	QCOW2	Public	⬆

Step 4.2

On the next page select the instance flavor or size you need. Do be mindful of over provisioning for your needs as there are limited CPU and RAM available. Use the up arrow on the right side

to select the instance flavor you need. Then click next.

Launch Instance ✕

❓

Flavors manage the sizing for the compute, memory and storage capacity of the instance.

- Details
- Source
- **Flavor**
- Networks *
- Network Ports
- Security Groups
- Key Pair
- Configuration
- Server Groups
- Scheduler Hints
- Metadata

A quick summary of compute generations at east campus are below:

Generation 2	Westmere	Xeon(R) X5660	DDR3
Generation 3	Ivy Bridge	Xeon(R) E5-2667 v2	DDR3
Generation 4	Haswell	Xeon(R) E5-2630 v3	DDR4
Generation 5	Cascade Lake	Xeon(R) Gold 6226R	DDR4

Allocated

Name	VCPUS	RAM	Public	
❯ m3.xlarge	4	16 GB	No	⬇

⌄ Available ⑤   Select one

🔍 Click here for filters or full text search. ✕

Name	VCPUS	RAM	Public	
❯ m4.nano	2	512 MB	No	⬆
❯ m4.micro	2	1 GB	No	⬆
❯ m4.small	2	2 GB	No	⬆
❯ m4.medium	2	4 GB	No	⬆
❯ m3.large	2	8 GB	No	⬆

✖ Cancel   ‹ Back   Next ›   ☁ Launch Instance

Step 4.3

Now we need to set up networking for our instance. There are two choices, one will allow access from any public IP, and the other will allow for access from within the CU network. This includes the VPN which is the route we will be using for this demo. Use the up arrow to select a projectnet ID between 1820 & 2564.

Launch Instance ✖

Networks provide the communication channels for instances in the cloud.

Details

Source

A floating IP must be attached after the instance is created to be reachable outside of the cloud environment.

Flavor

Networks

If attaching a UCB public/internet floating IP is desired, select a projectnet with an ID **above 26128**.
If attaching a UCB internal floating IP is desired, select select a projectnet with an ID **between 1820 and 2564**.

Network Ports

Security Groups

Key Pair

∨ Allocated ①   Select networks from those listed below.

		Network	Subnets Associated	Shared	Admin State	Status	
⇕ 1	›	projectnet1834-private	projectnet1834-private-subnet	No	Up	Active	⬇

Configuration

Server Groups

∨ Available ①   Select at least one network

Scheduler Hints

| 🔍 | Click here for filters or full text search. | | | | | ✖ |

Metadata

	Network	Subnets Associated	Shared	Admin State	Status	
›	projectnet26134-private	projectnet26134-private-subnet	No	Up	Active	⬆

✖ Cancel   **‹ Back** **Next ›** **☁ Launch Instance**

Step 4.4
Next skip down to the Security group section. Use the Up Arrow to add the ICMP-Restricted and SSH-Restricted security groups which will allow SSH and Ping from the VPN as well as other CU private networks. Add these 2 groups and click next.

Launch Instance ✕

Details

Source

Flavor

Networks

Network Ports

Security Groups

Key Pair

Configuration

Server Groups

Scheduler Hints

Metadata

Select the security groups to launch the instance in.

The default rule permits all egress. All ingress traffic is blocked unless security groups are allocated below. Rules with the "restricted" postfix requires the UCB VPN or UCB VPN Limited.

If an inbound rule from the internet is not listed below in the default set; it is not open on the UCB Border Firewall. The Firewall Exception Form will first need to be completed with OIS with the specific external floating IP. A custom security group can then be created and allocated to instances.

✔ Allocated ③

Displaying 3 items

| Name	Description ⬍	
❯ default	Default security group	⬇
❯ icmp-restricted	Allow ingress icmp from cu-vpn-priv, cu-sslvpn, and cu-sn-spsc-cscldmgt	⬇
❯ ssh-restricted	Allow ingress tcp/22 from cu-vpn-priv, cu-sslvpn, and cu-sn-spsc-cscldmgt	⬇

Displaying 3 items

✔ Available ⑩

Select one or more

🔍 Click here for filters or full text search.   ✖

Displaying 10 items

Name	Description	
❯ icmp	Allow ingress icmp from anywhere	⬆
❯ https	Allow ingress tcp/443 from anywhere	⬆
❯ powershell-restricted	Allow ingress PSRemoting from cu-vpn-priv, cu-sslvpn, and cu-sn-spsc-cscldmgt	⬆

Step 4.5.1

Use the create Key Pair Option on this page to create an SSH Key. Once you have created it, copy the private key to the clipboard. This is your only chance to get this private key and without it you will not be able to access the instance.
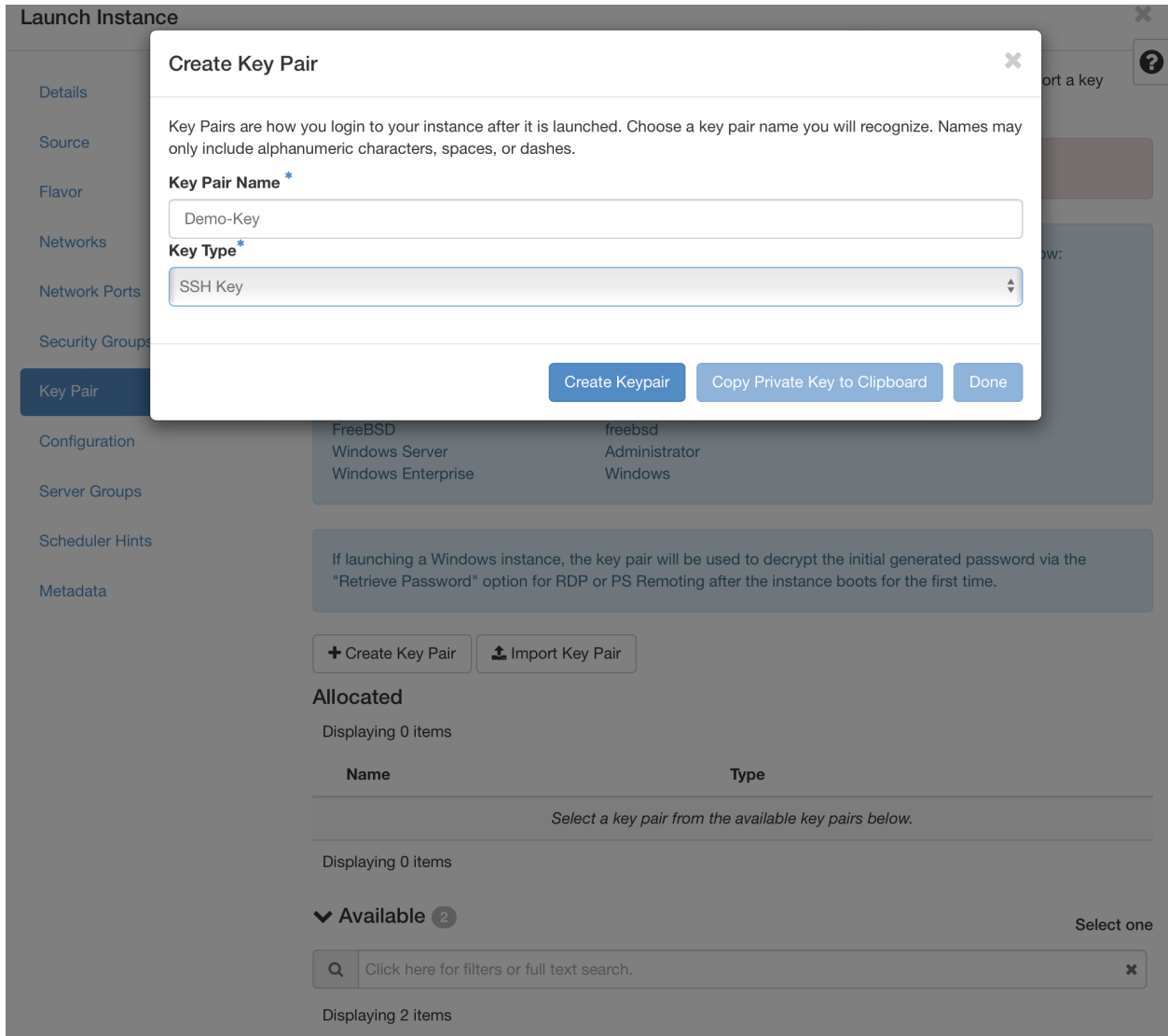
Step 4.5.2
Now we need to save the text from this key to a file we can use later to connect to the instance. Open a terminal window and type the following command

```
vim ~/.ssh/your_key_name
```

Once VIM is open press the "i" key to enter insert mode and paste the data for your private key. Next press "Escape" to exit insert mode and return to system mode. Then type ":wq" to write the file and quit vim.

```
R0XGoWLETfsY454XX0+Wu03eG5idab0QJp5CrDqkMbSuiMNUArGCGWnoHD+zg1ga
NJW9IZfwfvKOyQ3o8XOinXXRXxo+r8UBqkBBwQIDAQABAoIBAEDjmEejsaiktrS+
oUYxCb4GSU9SvTO6jVc/7jtNa5g9R7Er6ZurrT9ZzfFsZ0ZAH4ya2quZ4Cm8NwSB
ZSdsurqVMwGRLWJAzscQjg3oo2G+5jjULChRjJfDrSLv0VAYtcYjw3XEhUUlrmRR
j2N47jop0kxHIrh2fO/7IzE27ziIF5FNRqCXCbeTP+6l5qYoQiPqidC+owe+2ydO
1LrCPDdzHasCA9qJHL95d344ePCW0avel3HYBWJRj7VEmXfUCy66ZRIQP08EM6Gk
AzpAh/7nflL264XQpyTtMbCBxtjMlx16kHaS/TRKhnsBvRKq5NUk6RZyam2mKFS/
N345yQECgYEA4iR+/wzhaheQD61JNjmnsh4euPc4klJglYz/1CE2vCr93HsVf3dm
iEDfVTcfXvoWvKGl6ZnidZxoXjn9AjB5mIhGLokjQvUBkd5NtfBf99eeUlryURZ3
5wZquiRbkGRWv1gi3605O7FELdM4YLcaYs70ZjSnqwNaY3cBrYEKb00CgYEAy/zN
ZoVHENhQEmwAylFY6FzL8PNwrd+2x3u+5tf2fobNbl9AHFcFLkItmUIMhzUHlKq4
wh+YhVJ72h1915AcTjabgLTMhaMq45AJxHLyxQW7w5SThToaV2dQxpf9medBH3ML
E3cu0YtRj5WYmGoIcXsUhCPHBj/MzvRMez1USkUCgYEAx1W0O812xLWQXsE4T3vy
vaafHhuM2i2SxUBMIQhF3wdmTfqQsJLm5iOIuOyl0mpPsGYsgQUMAO+lb6ldw5hH
FpTIiCyJ0HN2tsVlaoI/lY+SFrXj8mYDuriFSx7RNy266V8fHlrqoBtZJauRfm7n
60DRXWX8hVmnSxWTfQTJ1eECgYEAjliyaPlWVy3r6n8NqiFMHpIrCtCGb/8uSFRV
KjfIXhDpqDy1u1EKX3Z05X7OyUc7LAZaebSctWSS4ukzmMqKxhWbPBxuoE71rekk
t77TJKPUnC3BJ/YczaNeHceQg+Z5bl3Ue9fuhvFmkAD/hFRx4ebWsUNaLkU1QbH+
8P0iJdUCgYAAuKkniNjS1iV6vMj5ND2u5njqIF8208N5U+4h8m7FF4YJr9Dxucwq
YcHoWTNiv4Pbn0hyqfxv3C4WPmk4zNaaoYt+IM7UCA4hVZ0V5VNU3+LU8P5Wh6c9
KZf0eSCys7bI1X4marTjlnL2aW1T+bNRF5I75vSH2Ye0GZ7Fn4kFNQ==
-----END RSA PRIVATE KEY-----
```

:wq

Step 4.5.3
Next we need to set the proper permissions for the Private key file. Do this by running the following command in your terminal.

```
sudo chmod 400 ~/.ssh/your_private_key
```

Step 4.6
Once you have saved your private key, you can use the launch instance button to launch the instance, this will take 1 to 2 minutes to complete.

Create Key Pair   **Import Key Pair**

Allocated

Displaying 1 item

Name	Type	
> Demo-Key	ssh	⬇

Displaying 1 item

∨ Available ②

Select one

🔍 Click here for filters or full text search.   ✕

Displaying 2 items

Name	Type	
> recsys-demo-key	ssh	⬆
> RecSYS-Linux-LJ-4	ssh	⬆

Displaying 2 items

‹ Back   **Next ›**   **☁ Launch Instance**

Now we need to set a Floating IP to allow the instance to talk to the greater CU network including the VPN, The IP we assign will also be the one we use when SSH-ing to the instance


If no floating IP's are available we can allocate a new one to the project from the network section. Set the Pool to Scinet-Internal, and then leave all other options blank and click Allocate IP.
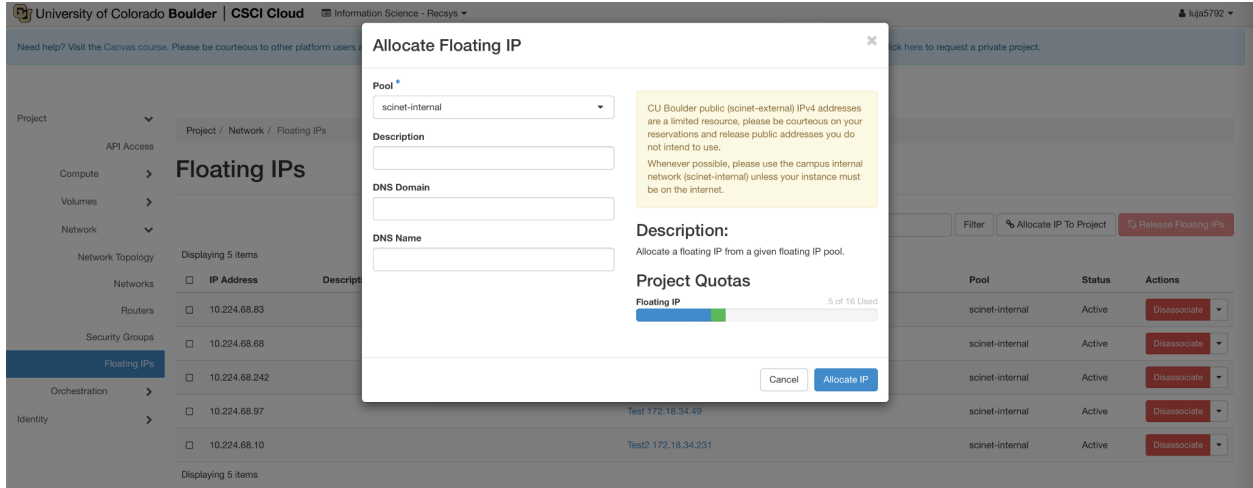
Step 4.7

The default size of the boot drive will be too small for use to install a GUI without filling the disk to 100% and rendering the system unusable. So we are going to extend the volume that was automatically made and attached for us when we created the instance. Go to Volumes -> Volumes and find the volume attached to your instance from the list.
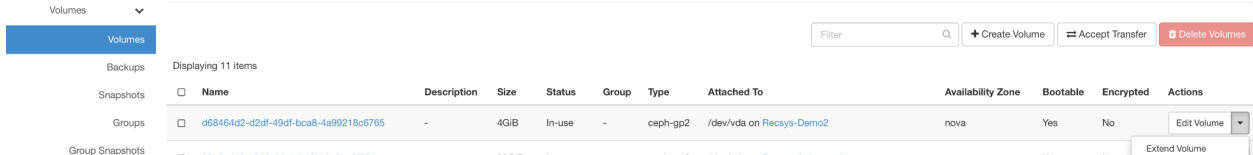
Once you find the instance, select extend volume from the far right drop down in the menu and set the size of the Volume to a more reasonable size like 16GB, 32GB, 64GB.
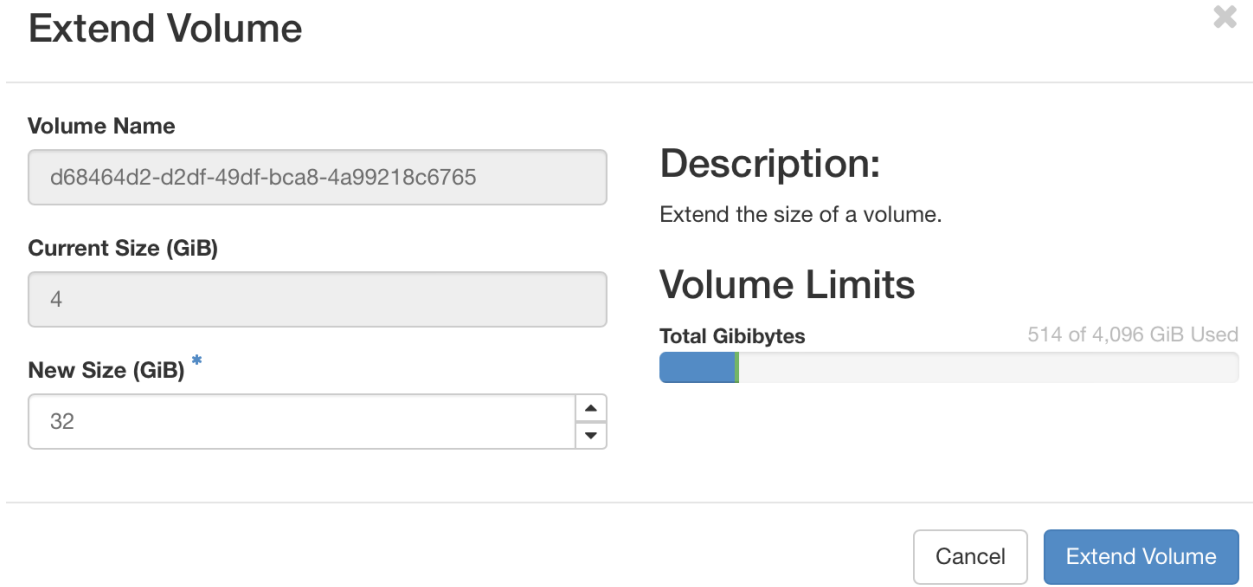
Once this is complete we can move on to connecting to the instance.

Step 5.

Open a terminal and ssh to the box with the following command. This assumes Ubuntu as the target Distro, others may have other user names. The IP and Private Key Name will also change for your instance.
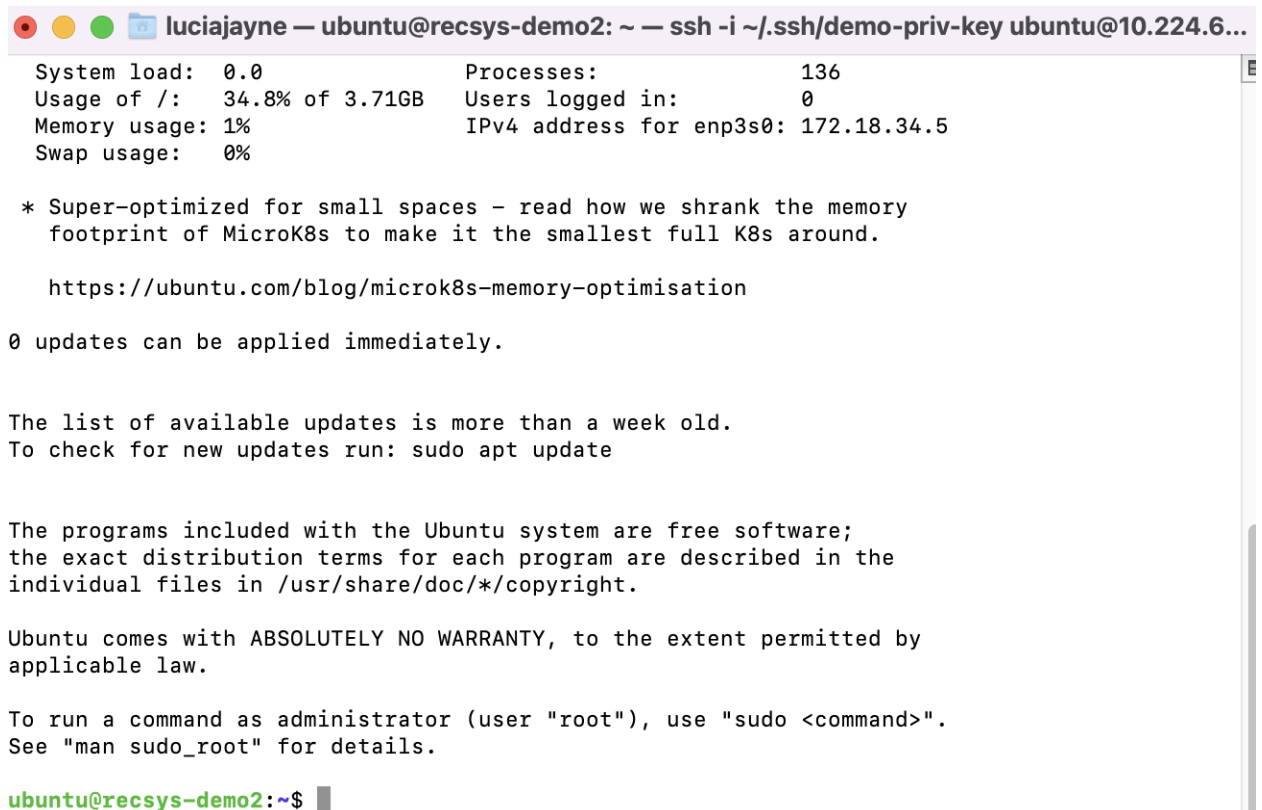
Open a terminal and connect to the VPN before proceeding.

Step 6.

Connect to the instance with the following command

```
ssh -i ~/.ssh/your_key_file ubuntu@flo.tin.g.ip (10.224.68.X)
```

You will be prompted to accept the fingerprint for the remote system, type yes then you will arrive at a terminal on the remote system.

```
luciajayne — ubuntu@recsys-demo2: ~ — ssh -i ~/.ssh/demo-priv-key ubuntu@10.224.6...

  System load:   0.0              Processes:                136
  Usage of /:    34.8% of 3.71GB  Users logged in:          0
  Memory usage:  1%               IPv4 address for enp3s0:  172.18.34.5
  Swap usage:    0%

 * Super-optimized for small spaces — read how we shrank the memory
   footprint of MicroK8s to make it the smallest full K8s around.

   https://ubuntu.com/blog/microk8s-memory-optimisation

0 updates can be applied immediately.


The list of available updates is more than a week old.
To check for new updates run: sudo apt update


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.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@recsys-demo2:~$
```

Step 7.

Once you are there, run the following command to check how much space there is on your volumes

```
df -h
```

```
[ubuntu@recsys-demo2:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
tmpfs           1.6G  1.1M  1.6G   1% /run
/dev/vda1       3.8G  1.4G  2.4G  36% /
tmpfs           7.9G     0  7.9G   0% /dev/shm
tmpfs           5.0M     0  5.0M   0% /run/lock
tmpfs           4.0M     0  4.0M   0% /sys/fs/cgroup
/dev/vda15      105M  5.2M  100M   5% /boot/efi
tmpfs           1.6G  4.0K  1.6G   1% /run/user/1000
ubuntu@recsys-demo2:~$
```

Since the volume size is still 4GB rather than the lager size we expected we will need to reboot the machine for it to expand its boot disk. Do this by running the following command

```
sudo reboot
```

Wait 1 to 2 minutes for the remote machine to finish rebooting then run the SSH command again to reconnect.

```
ssh -i ~/.ssh/your_key_file ubuntu@flo.tin.g.ip (10.224.68.X)
```

Step 8.
Once you are connected, we are going to update the package manager before we start installing new tools. Do this by running the following commands

```
sudo apt update
sudo apt upgrade
```

Press "Y" to agree to install the new packages then wait for them to finish, this will take 2 to 3 minutes. If you see a message about updating the kernel, accept the update, then accept the default services to be restarted on the next page. Once the upgrade is complete reboot the system to allow the kernel to fully update before proceeding. Then reconnect.

Step 9.
Next we are going to install the tools for the VNC server, this will allow us to have access to a GUI on the remote box. Do this by running the following command on the remote machine.

```
sudo apt install tigervnc-standalone-server tigervnc-xorg-extension tigervnc-viewer
```

```
[ubuntu@recsys-demo2:~$ sudo apt install tigervnc-standalone-server tigervnc-xorg-extension tigervnc-viewer
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
  net-tools
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  cpp cpp-10 fontconfig-config fonts-dejavu-core gcc-10-base libdrm-amdgpu1 libdrm-intel1 libdrm-nouveau2
  libdrm-radeon1 libegl-mesa0 libegl1 libepoxy0 libfile-readbackwards-perl libfltk-images1.3 libfltk1.3
  libfontconfig1 libfontenc1 libgbm1 libgl1 libgl1-mesa-dri libglapi-mesa libglvnd0 libglx-mesa0 libglx0 libice6
  libisl23 libjpeg-turbo8 libjpeg8 libllvm12 libmpc3 libpciaccess0 libpixman-1-0 libsensors-config libsensors5 libsm6
  libvulkan1 libwayland-client0 libwayland-server0 libx11-xcb1 libxaw7 libxcb-dri2-0 libxcb-dri3-0 libxcb-glx0
  libxcb-present0 libxcb-randr0 libxcb-shm0 libxcb-sync1 libxcb-xfixes0 libxcursor1 libxfixes3 libxfont2 libxft2
  libxi6 libxinerama1 libxkbfile1 libxmu6 libxpm4 libxrandr2 libxrender1 libxshmfence1 libxt6 libxxf86vm1
  mesa-vulkan-drivers tigervnc-common x11-common x11-xkb-utils x11-xserver-utils xfonts-base xfonts-encodings
  xfonts-utils xserver-common xserver-xorg-core
Suggested packages:
  cpp-doc gcc-10-locales lm-sensors xfonts-100dpi | xfonts-75dpi xfonts-scalable nickle cairo-5c xorg-docs-core
The following NEW packages will be installed:
  cpp cpp-10 fontconfig-config fonts-dejavu-core gcc-10-base libdrm-amdgpu1 libdrm-intel1 libdrm-nouveau2
  libdrm-radeon1 libegl-mesa0 libegl1 libepoxy0 libfile-readbackwards-perl libfltk-images1.3 libfltk1.3
  libfontconfig1 libfontenc1 libgbm1 libgl1 libgl1-mesa-dri libglapi-mesa libglvnd0 libglx-mesa0 libglx0 libice6
  libisl23 libjpeg-turbo8 libjpeg8 libllvm12 libmpc3 libpciaccess0 libpixman-1-0 libsensors-config libsensors5 libsm6
  libvulkan1 libwayland-client0 libwayland-server0 libx11-xcb1 libxaw7 libxcb-dri2-0 libxcb-dri3-0 libxcb-glx0
  libxcb-present0 libxcb-randr0 libxcb-shm0 libxcb-sync1 libxcb-xfixes0 libxcursor1 libxfixes3 libxfont2 libxft2
  libxi6 libxinerama1 libxkbfile1 libxmu6 libxpm4 libxrandr2 libxrender1 libxshmfence1 libxt6 libxxf86vm1
  mesa-vulkan-drivers tigervnc-common tigervnc-standalone-server tigervnc-viewer tigervnc-xorg-extension x11-common
  x11-xkb-utils x11-xserver-utils xfonts-base xfonts-encodings xfonts-utils xserver-common xserver-xorg-core
0 upgraded, 75 newly installed, 0 to remove and 0 not upgraded.
Need to get 57.0 MB of archives.
After this operation, 213 MB of additional disk space will be used.
Do you want to continue? [Y/n] █
```

Press "Y" to install then allow it to complete.

Step 10.
Next we will install the desktop environment, for this example we are using Gnome which is the default for Ubuntu Desktop, however since this is a CLI only server install it does not have this installed. We are going to install it using the following command.

```
sudo apt install ubuntu-gnome-desktop
```

```
ubuntu@recsys-demo2:~$ sudo apt install ubuntu-gnome-desktop
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
  net-tools
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  acl acpi-support acpid adwaita-icon-theme aisleriot alsa-base alsa-utils anacron apg app-install-data-partner
  apport-gtk appstream apt-config-icons apt-config-icons-hidpi aptdaemon aptdaemon-data apturl apturl-common aspell
  aspell-en at-spi2-core avahi-autoipd avahi-daemon avahi-utils baobab bluez bluez-cups bluez-obexd branding-ubuntu
  brltty bubblewrap bzip2 cheese cheese-common colord colord-data cracklib-runtime cups cups-browsed cups-bsd
  cups-client cups-common cups-core-drivers cups-daemon cups-filters cups-filters-core-drivers cups-ipp-utils
  cups-pk-helper cups-ppdc cups-server-common dbus-user-session dbus-x11 dc dconf-cli dconf-gsettings-backend
  dconf-service deja-dup desktop-file-utils dictionaries-common dmz-cursor-theme dns-root-data dnsmasq-base
  docbook-xml duplicity emacsen-common enchant-2 eog espeak-ng-data evince evince-common evolution-data-server
  evolution-data-server-common file-roller firefox fontconfig fonts-beng fonts-beng-extra fonts-deva fonts-deva-extra
  fonts-droid-fallback fonts-freefont-ttf fonts-gargi fonts-gubbi fonts-gujr fonts-gujr-extra fonts-guru
  fonts-guru-extra fonts-indic fonts-kacst fonts-kacst-one fonts-kalapi fonts-khmeros-core fonts-knda fonts-lao
  fonts-liberation fonts-liberation2 fonts-lklug-sinhala fonts-lohit-beng-assamese fonts-lohit-beng-bengali
  fonts-lohit-deva fonts-lohit-gujr fonts-lohit-guru fonts-lohit-knda fonts-lohit-mlym fonts-lohit-orya
  fonts-lohit-taml fonts-lohit-taml-classical fonts-lohit-telu fonts-mlym fonts-nakula fonts-navilu fonts-noto-cjk
  fonts-noto-color-emoji fonts-noto-mono fonts-opensymbol fonts-orya fonts-orya-extra fonts-pagul fonts-sahadeva
  fonts-samyak-deva fonts-samyak-gujr fonts-samyak-mlym fonts-samyak-taml fonts-sarai fonts-sil-abyssinica
  fonts-sil-padauk fonts-smc fonts-smc-anjalioldlipi fonts-smc-chilanka fonts-smc-dyuthi fonts-smc-gayathri
  fonts-smc-karumbi fonts-smc-keraleeyam fonts-smc-manjari fonts-smc-meera fonts-smc-rachana
  fonts-smc-raghumalayalamsans fonts-smc-suruma fonts-smc-uroob fonts-taml fonts-telu fonts-telu-extra
  fonts-teluguvijayam fonts-thai-tlwg fonts-tibetan-machine fonts-tlwg-garuda fonts-tlwg-garuda-ttf
  fonts-tlwg-kinnari fonts-tlwg-kinnari-ttf fonts-tlwg-laksaman fonts-tlwg-laksaman-ttf fonts-tlwg-loma
  fonts-tlwg-loma-ttf fonts-tlwg-mono fonts-tlwg-mono-ttf fonts-tlwg-norasi fonts-tlwg-norasi-ttf fonts-tlwg-purisa
  fonts-tlwg-purisa-ttf fonts-tlwg-sawasdee fonts-tlwg-sawasdee-ttf fonts-tlwg-typewriter fonts-tlwg-typewriter-ttf
  fonts-tlwg-typist fonts-tlwg-typist-ttf fonts-tlwg-typo fonts-tlwg-typo-ttf fonts-tlwg-umpush fonts-tlwg-umpush-ttf
  fonts-tlwg-waree fonts-tlwg-waree-ttf fonts-ubuntu fonts-urw-base35 fonts-yrsa-rasa foomatic-db-compressed-ppds
  fprintd gamemode gamemode-daemon gcr gdb gdm3 gedit gedit-common genisoimage geoclue-2.0 ghostscript ghostscript-x
  gir1.2-accountsservice-1.0 gir1.2-atk-1.0 gir1.2-atspi-2.0 gir1.2-clutter-1.0 gir1.2-clutter-gst-3.0
```

This is a large package (2GB of downloaded data) and will take about 5 min to download and install. Press "Y" to accept the installation and grab a cup of tea.

Once the installation has finished we have a few more small things to take care of before we can access the desktop on this instance.

Step 11

First we need to set a VNC password for access to the machine. Use the following command to set your VNC password, when prompted to set a view only password respond "n" as we do not need view only access to this machine

```
vncpasswd
```

```
ubuntu@recsys-demo2:~$ vncpasswd
Password:
Verify:
Would you like to enter a view-only password (y/n)? n
ubuntu@recsys-demo2:~$
```

Step 12

Now we are going to set a password for the Ubuntu user, this will allow us to login to the desktop through the user login window, as the system has no way of doing Pub Key Auth when logging in to the desktop over VNC. Set the password for the Ubuntu user with the following command.

```
sudo passwd ubuntu
```

```
ubuntu@recsys-demo2:~$ sudo passwd ubuntu
New password:

Retype new password:

passwd: password updated successfully
ubuntu@recsys-demo2:~$
```
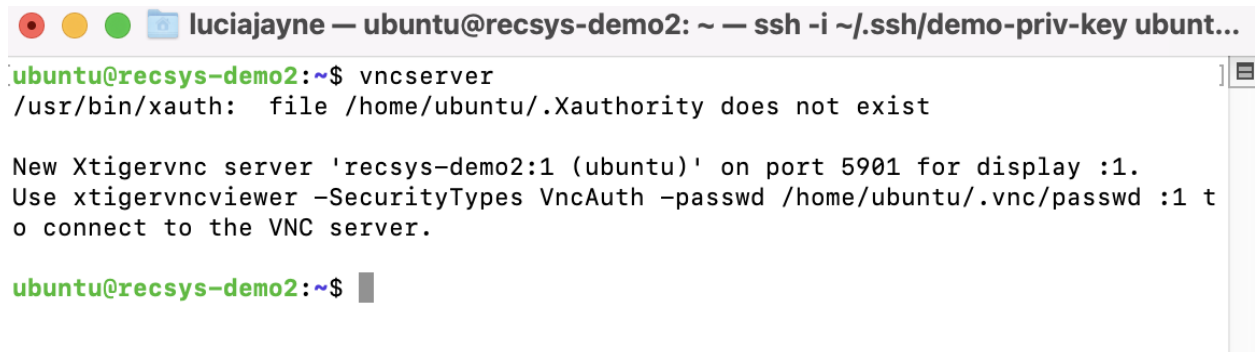
Step 13

Now that we have the passwords for our Ubuntu user and VNC set we need to start the VNC Service on the remote machine. Do this by running the following command

```
vncserver
```
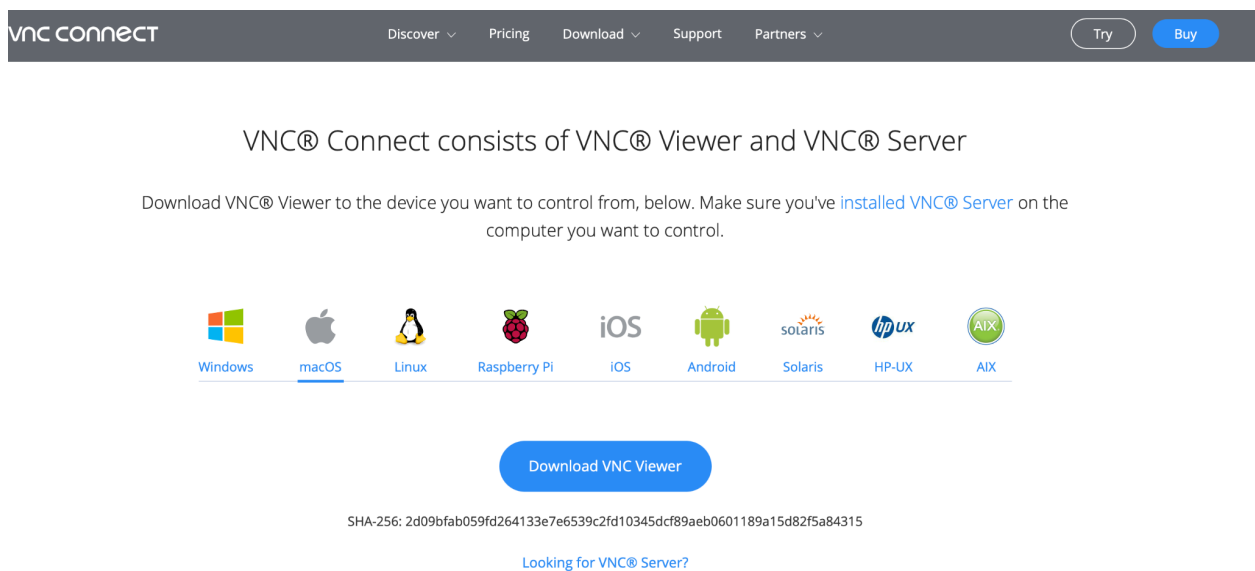

You should not receive any error messages, however if you do, try rebooting the machine as we have made a lot of changes this session. Additionally if there is an error related a VNC service already running you can terminate all running vnc sessions with `vncserver -kill :*` .

Once the service is started we need to move back to our local machine to take care of accessing the remote instance.

Step 14

Download and install VNC Viewer from Real VNC at This Link

Step 15

Once you have installed VNC Viewer open a new tab in your terminal and run the following command to establish an SSH tunnel between the localhost IP on your local machine and the remote box on port 5901 which is used for VNC sessions.

```
ssh -i ~/.ssh/your_private_key -L 5901:127.0.0.1:5901 -N -f -l ubuntu 10.224.68.X
```
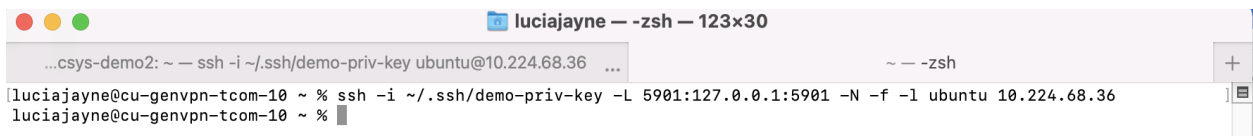

Once the tunnel is established we can use VNC viewer to connect to the graphical desktop on the remote host. Open VNC viewer and connect to the following IP and Port number

127.0.0.1:5901

You will be prompted to enter the password for VNC that you set earlier. Enter that password and you will arrive at the desktop
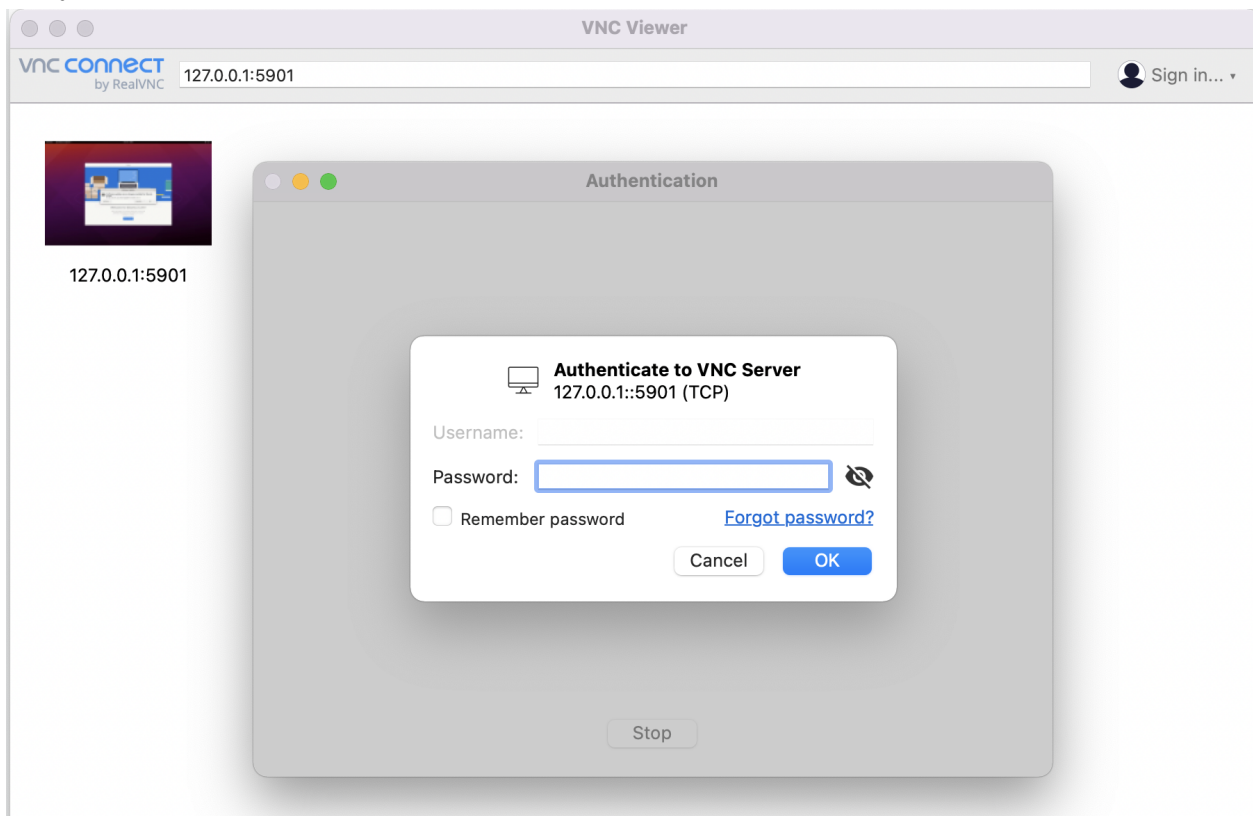
Complete the initial setup and you will have access to the desktop. You may not be prompted for the password you set for the ubuntu user on the first login, however you will need it for future sessions.

If the interface is very low detail or very slow to respond, use the gear icon in the top bar in the VNC window to change the detail level.
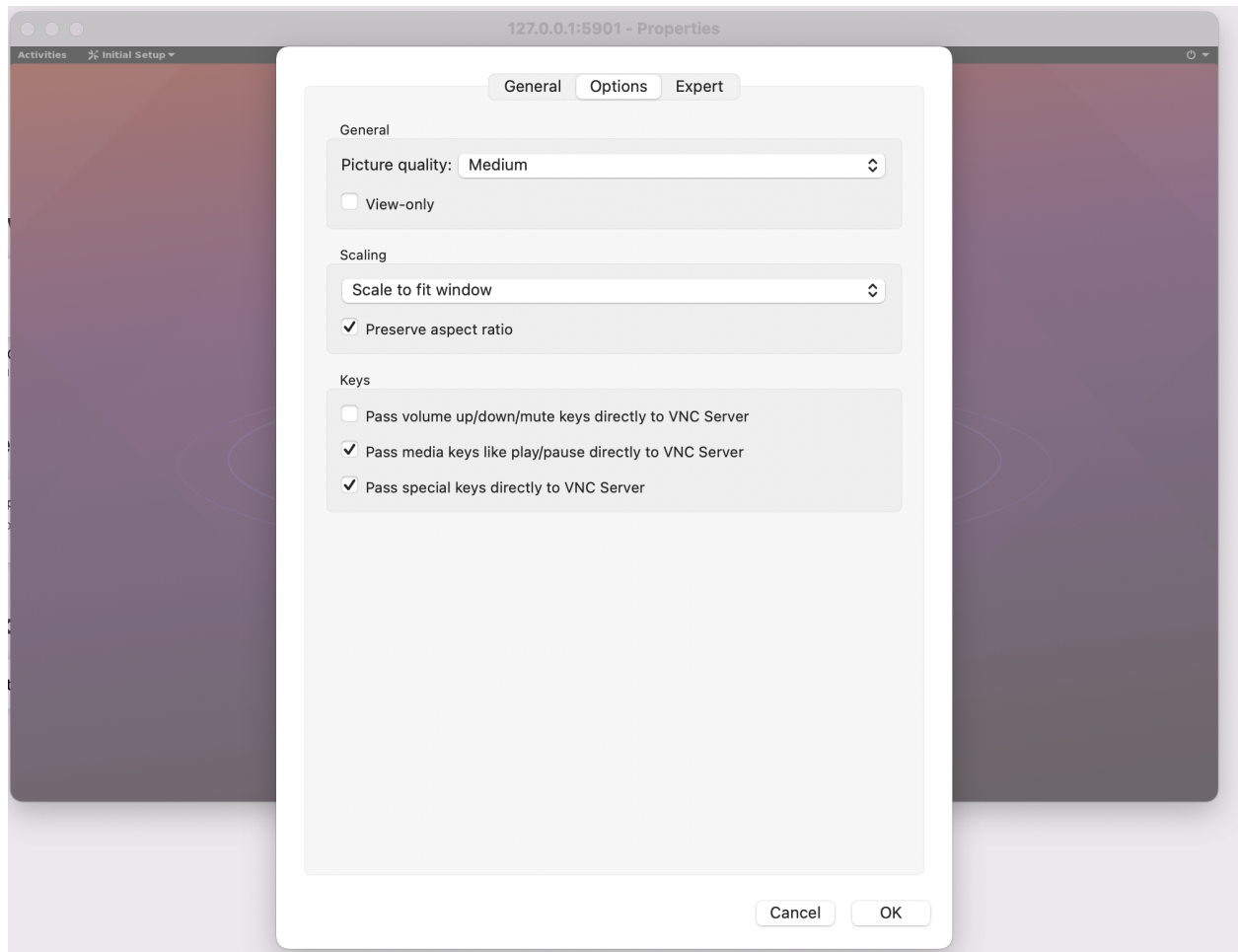
You are now ready to use your remote instance as if it was any other desktop, congratulations and have fun.