45HomeLab
A Division of 45Drives

HL4 - HOMELAB SERVER
USER MANUAL

REVISION DETAILS

Version	Description
1.0	Initial published version

Contents

45HomeLab
A Division of 45Drives

45HomeLab
A Division of 45Drives

45HomeLab
A Division of 45Drives

45HomeLab
A Division of 45Drives

45HomeLab
A Division of 45Drives

GETTING STARTED

45HomeLab
A Division of 45Drives

Welcome to 45HomeLab community. We are so glad you chose us.

ABOUT US

Serving our customers well is at the core of everything we do at 45HomeLab. In an industry where technical support is strictly timed, automated and impersonal we are real people solving real storage problems. Our storage solutions are non-proprietary, giving you the freedom to run any software you choose.

OUR GOAL

Provide you with the best storage solution for your data needs - not the most expensive one. Today we provide the most affordable storage solutions in the industry.

OUR MISSION

To provide affordable open-source storage solutions while staying true to our community roots by giving back to the open-source community that we rely on.

OPEN DESIGN

Unlike mainstream data storage providers 45HomeLab maintains an open design and ongoing relationships with the open-source community.

WHY 45HOMELAB HL4

At 45HomeLab we know home labbers have a strong vision of the infrastructure they want, and how to build and configure it. A key pillar of a great home lab, is a sufficient storage that is customized and configured how you need it.

The problem is home storage offerings today are under powered and have locked down software, while enterprise solutions are just too big and expensive. We understand how frustrating this is, which is why we've created the 45HomeLab product line. It is big, strong, fast, while also being open and flexible so you can easily modify, upgrade, and repair it, all at a price that makes sense for a power home lab user.

Our new HL4 is a 4-bay server that was designed to provide the power and storage needed for a great home lab.

HARDWARE

COMPONENTS AND SPECIFICATIONS

Chassis Dimensions (LxWxH): 9 3/4" x 5/16" x 8 1/2"
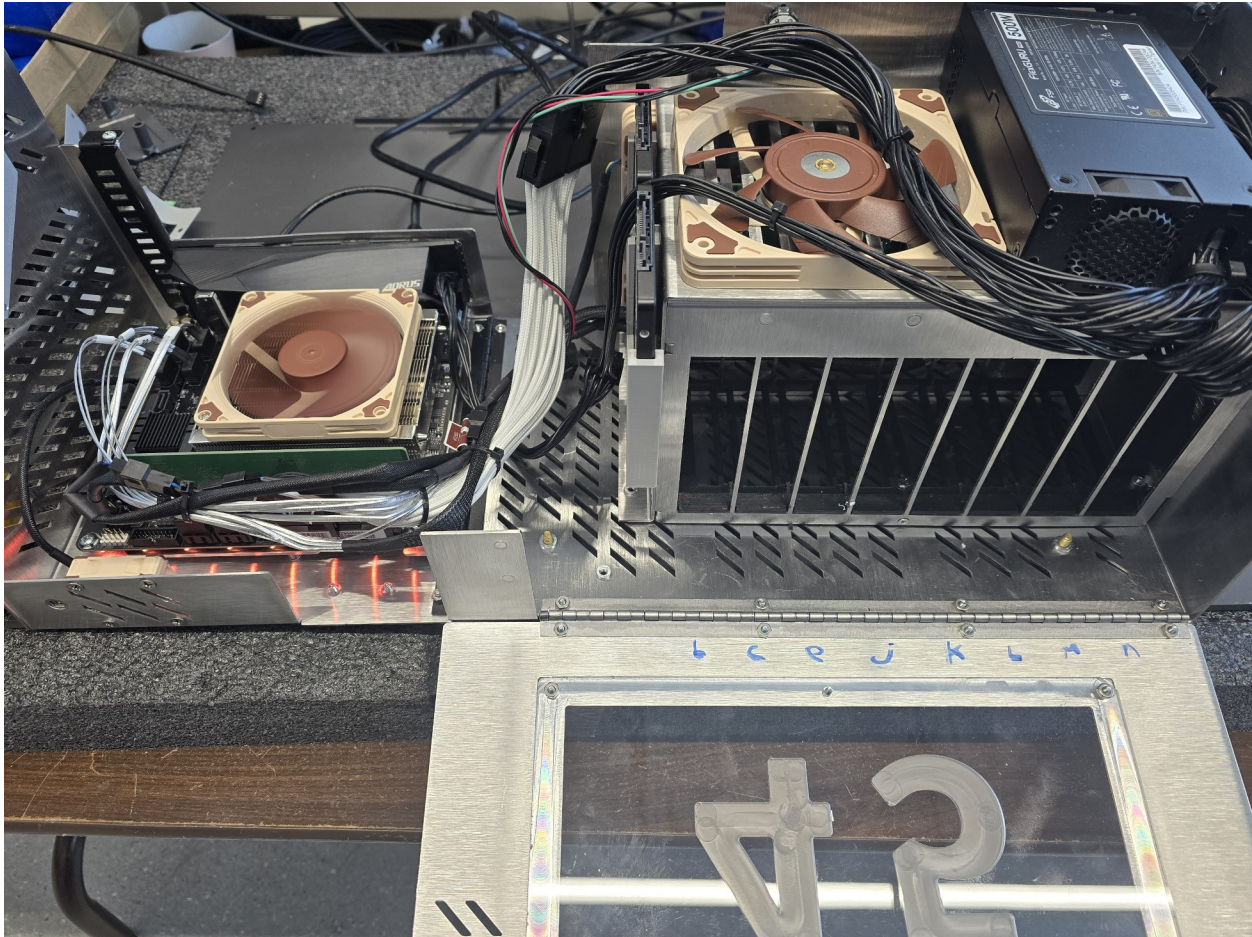
Below are the components that will be in your HL4 if you have not requested for any modification in the order the below components will be present by default. Your CPU, motherboard, RAM, Boot drives etc. could change based on your customization.

45HomeLab
A Division of 45Drives

ID	Component	Model - specs
1	Backplane	HL – 8 Drive PCB Backplane
2	Boot Storage	Optional: 1TB/500GB/250GB - Gen 4x4 NVMe M.2
3	Cables	miniSAS HD 0.6m
4	Cables	MiniSAS SFF-8087 to miniSAS HD SFF-8643 50cm
5	Chassis	HL4 Chassis
6	CPU	Optional: Ryzen 5 5500GT or Ryzen 7 5700G
7	Fans	NF-A12x15 PWM 120mmx15mm Noctua x1 NF-A4x10 FLX 40mmx10 Noctua x1 NH-L9a-AM4 Noctua CPU Cooler
8	Heatsink	B8 2U Passive Cooler
9	Motherboard	GIGABYTE B550I AORUS PRO AX 1.0
11	Networking	
12	Networking	2.5 GbE LAN
13	NVME Riser Card	AOC-SLG3-2M2 – 2 Position M.2 NVMe Add on Card (Optional)
14	Power Supply	FLEX GURU 500W 80 Plus Gold Certified Efficiency ≥90%
15	RAM	16GB DDR4 UDIMM
17	RAM	32GB DDR4 UDIMM
18	RAM	64GB DDR4 UDIMM

INITIAL SETUP

INSERTING THE STORAGE DRIVES

⊕ Once your unit is unboxed or in its final destination you can now start installing the storage drives.

45HomeLab
A Division of 45Drives

⊕ It is best practice to install the drives from starting from index 1-1 and sequentially incrementing in order.

⊕ The drives should fit snug into the slots. Make sure the back of the drive is facing the right when placing in the slot

⊕ After all the drives are inserted into the slots, you can close up the unit and move on to hooking up the needed cables to your unit.

CABLE SETUP

At this point, you should have the unit unboxed and storage drives installed. The next step is to connect the cables needed to connect and configure the unit.

45HomeLab
A Division of 45Drives

Fig:

Back panel slots

No	Component
1	2.5G port
2	BIOS USB 3.2 slot
3	BIOS Q-Flash Plus button
4	USB slot
5	USB slot
6	USB slot
7	USB slot
8	Type C slot
9	HDMI port
10	HDMI port
11	Display (DP) port
12	WIFI ports
13	MIC audio port
14	LINE OUT audio port
15	LINE IN audio port
16	Power cable slot
17	Power switch
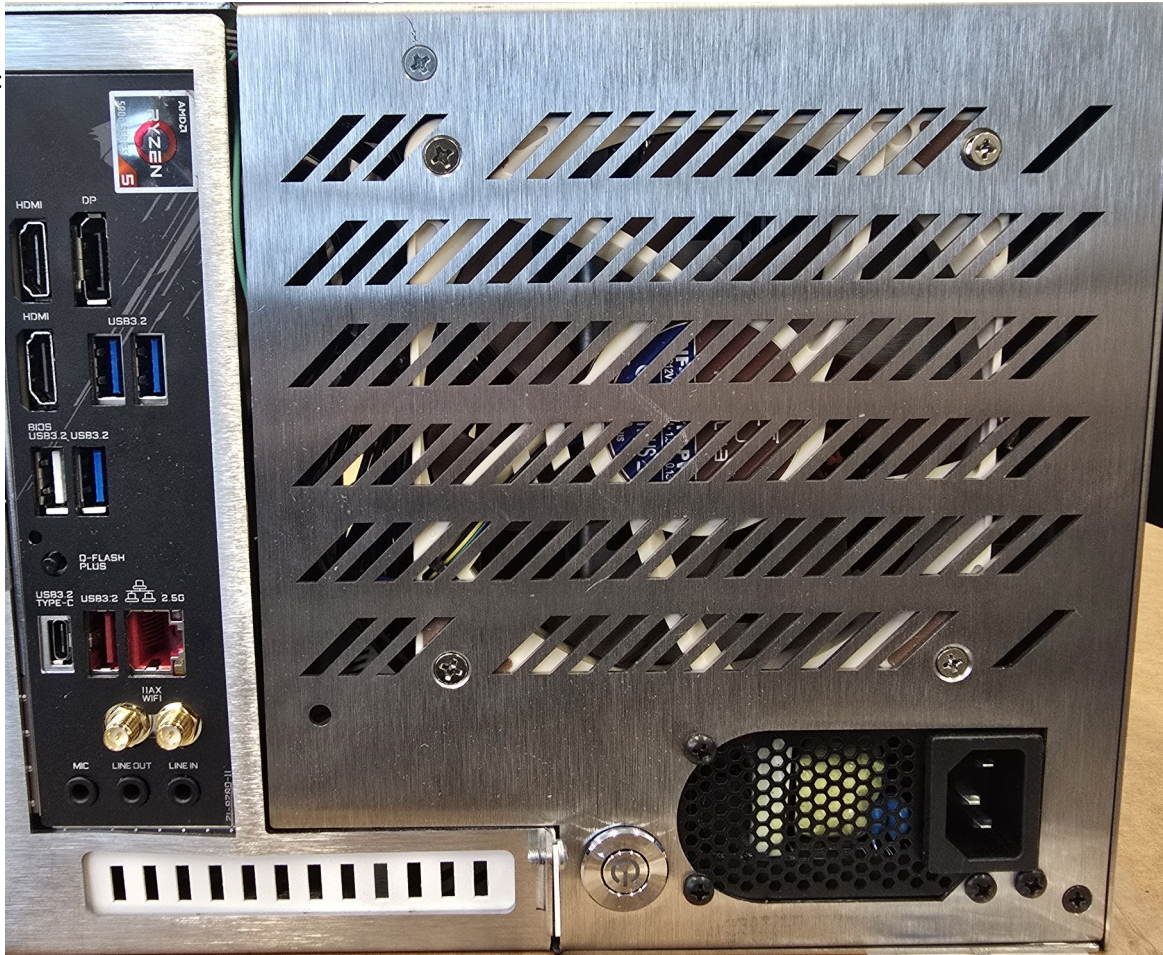
Table: Back panel slot details

45HomeLab
A Division of 45Drives

- You can connect a 1GB LAN cable from your router or switch as per your network setup. For the initial connection to your unit, you can plug in the other end of the Ethernet/ LAN cable to provide internet to the HL4 and where we will be accessing the data from the slot labeled 1 in the back on the unit.
- A HMDI or Display Port monitor will also be connected initially so you can see the see the desktop/Graphical User Interface screen when the unit first turns on. You can connect the monitor to the HMDI ports (9)/(10) or Display Port (11).
- If you want to configure the unit locally, a USB keyboard will also be beneficial. You can use any of the USB slots labelled from 4 to 7 in fig above.
- The last step would be to plug in the power cable. Slot labelled 16.
- After that you can turn on the power switch and the press the round blue power button to turn on.

Power requirements

- The PSU we provide with the appropriate options is a FLEX GURU 500W 80 Plus Gold Certified Efficiency ≥90%
- If you are sourcing your own power supply, you can use this as a minimum guideline. It is especially important to ensure you have at least 20A of 5v power.
- In addition, if you plan on using a graphics card, or anything that will increase power draw, you may require a more powerful supply.

45HomeLab
A Division of 45Drives

45HOMELAB HL 4 INITIAL CABLING SETUP DIAGRAM

45HomeLab
A Division of 45Drives
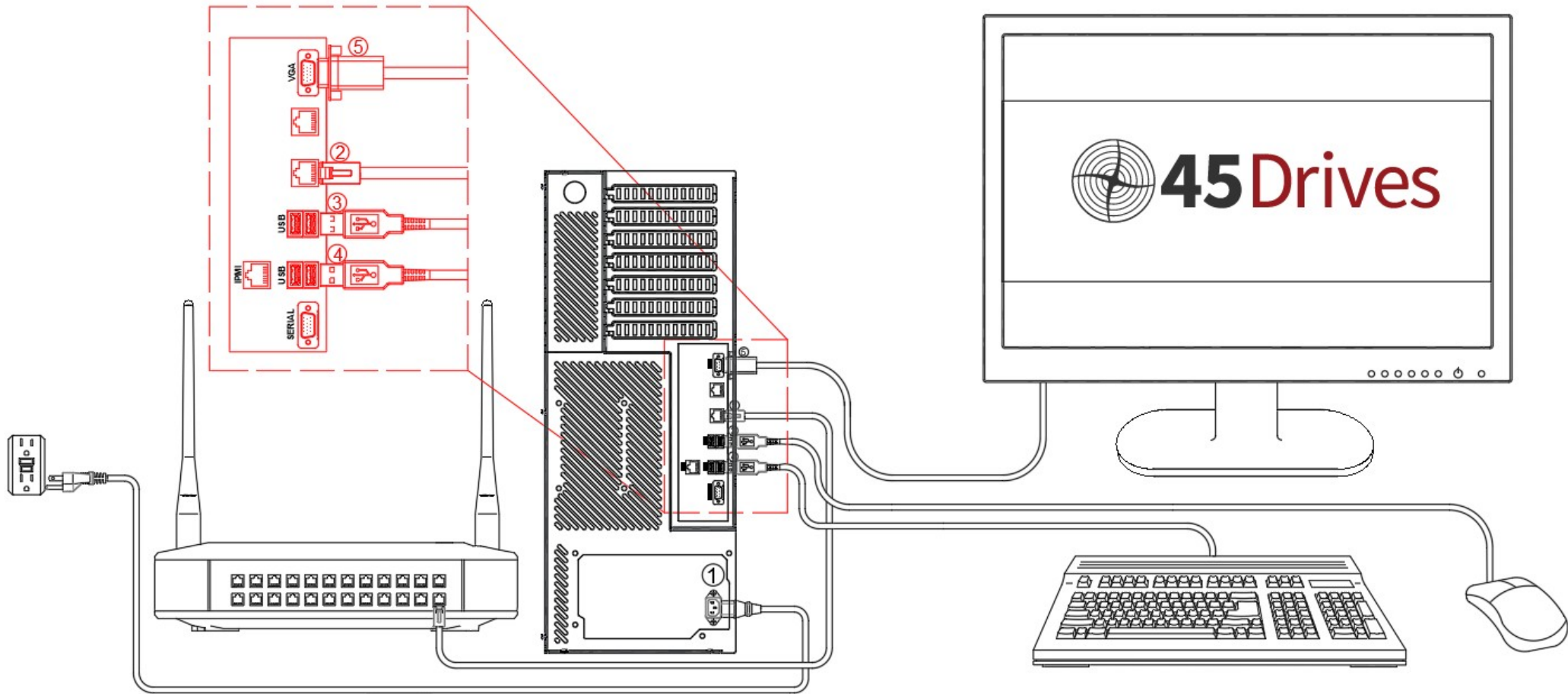
INITIAL CONFIGURATION /SETUP

Now you have your HL4 all powered up and ready to be configured. Before we proceed you need to have the below steps completed.

1. The power cables have been connected.
2. The 2.5G slot should have a LAN cable plugged in.

NETWORK SETUP

USING THE DESKTOP UI

- ⊕ Login to the desktop UI using the 45drives user and 45Dr!ves as the password.
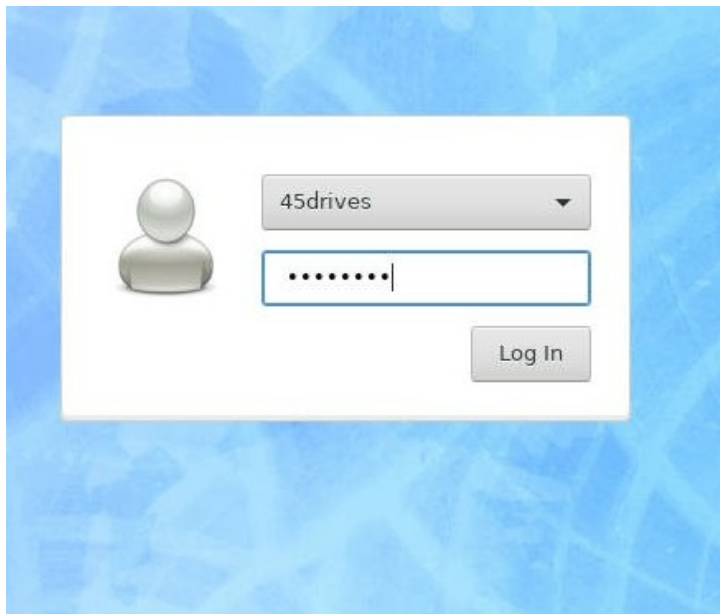
- ⊕ Once logged in we can check if our interfaces are up
- ⊕ Open a terminal and type the command **ip a** to check if your interface is up. You should be able to see that a broadcast carrier is available like below.

45HomeLab
A Division of 45Drives

```
root@iaustine:~# ip a
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: eno1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 100
    link/ether 3c:ec:ef:79:9c:20 brd ff:ff:ff:ff:ff:ff
    inet 192.168.250.100/16 brd 192.168.255.255 scope global noprefixroute eno1
       valid_lft forever preferred_lft forever
    inet6 fe80::52b6:addd:a436:4f1c/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: eno2: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN group default qlen 1000
    link/ether 3c:ec:ef:79:9c:21 brd ff:ff:ff:ff:ff:ff
4: ens7f0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN group default qlen 1000
    link/ether a0:36:9f:b1:ba:90 brd ff:ff:ff:ff:ff:ff
5: ens7f1: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN group default qlen 1000
    link/ether a0:36:9f:b1:ba:92 brd ff:ff:ff:ff:ff:ff
6: bond0: <NO-CARRIER,BROADCAST,MULTICAST,MASTER,UP> mtu 1500 qdisc noqueue state DOWN group default qlen 1000
    link/ether 10:10:10:10:10:10 brd ff:ff:ff:ff:ff:ff
    inet 192.168.168.192/16 brd 192.168.255.255 scope global noprefixroute bond0
       valid_lft forever preferred_lft forever
7: virbr0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default qlen 1000
    link/ether 52:54:00:40:97:fe brd ff:ff:ff:ff:ff:ff
    inet 192.168.122.1/24 brd 192.168.122.255 scope global virbr0
       valid_lft forever preferred_lft forever
8: virbr0-nic: <BROADCAST,MULTICAST> mtu 1500 qdisc fq_codel master virbr0 state DOWN group default qlen 1000
    link/ether 52:54:00:40:97:fe brd ff:ff:ff:ff:ff:ff
root@iaustine:~#
```

carrier/connection detected

No carrier/connection detected

- ⊕ Go to the applications in the top left corner. Go to settings -> Advanced network Configurations
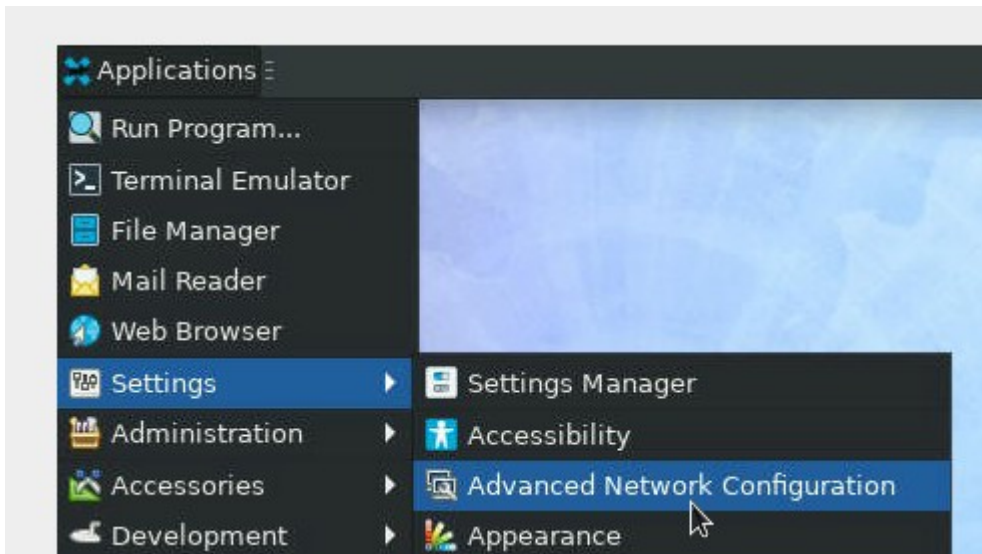
- ⊕ You should be able to see your network connections/ interfaces there
- ⊕ It will either show the interface name or might say like wired connection1, 2 etc
- ⊕ You can double click on it to check the actual interface name
- ⊕ If the interface is not showing you can click on the plus sign and add a new ethernet connection.

45HomeLab
A Division of 45Drives
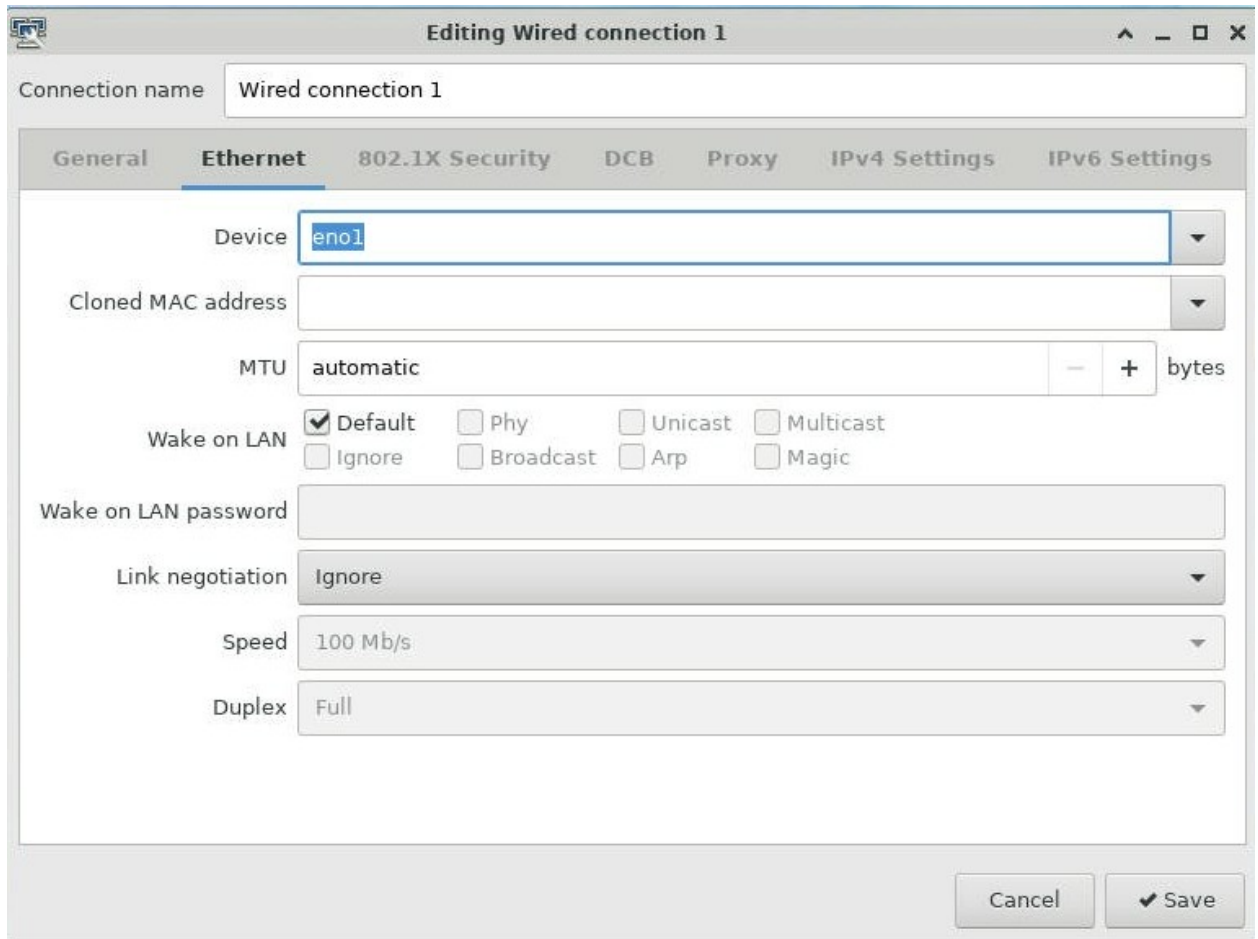
Edit the interface and go to the IPV4 settings section to set the IP. Change the method to manual for setting the static IP and click on add to enter the IP details.

45HomeLab
A Division of 45Drives
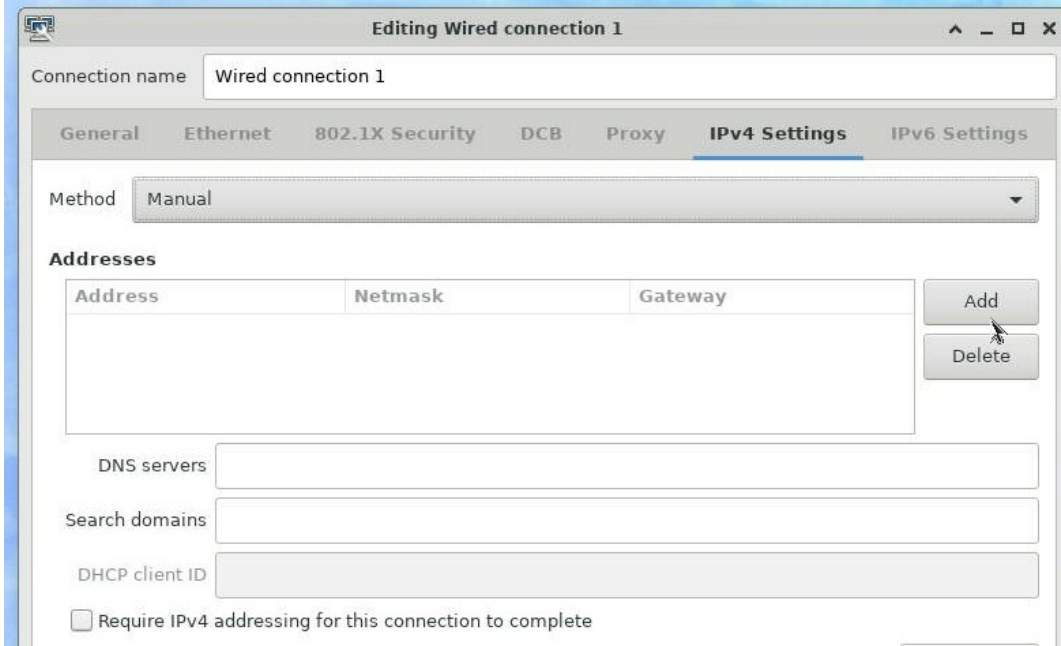
Once the details have been entered click save

45HomeLab
A Division of 45Drives
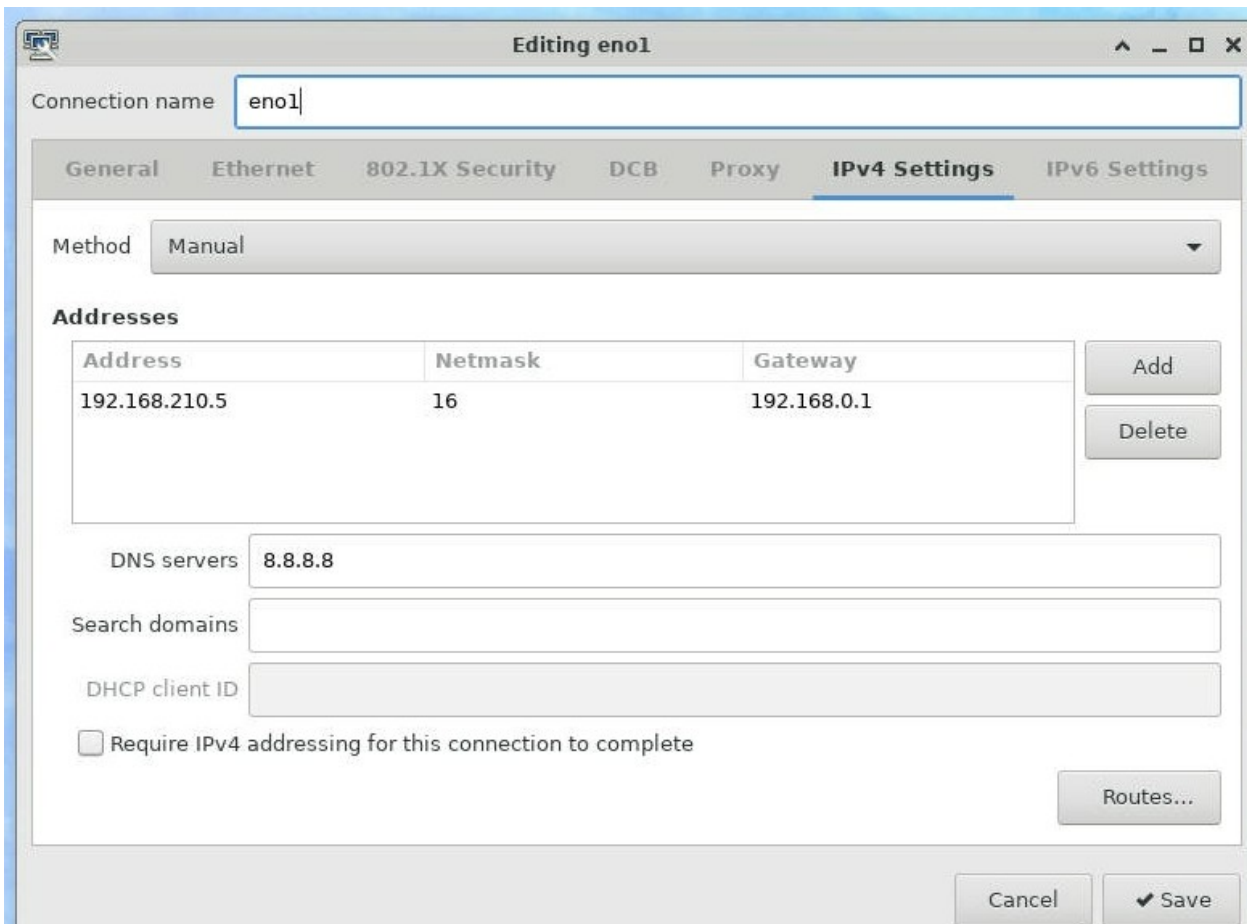
After this you can start setting password for your root user using below command in the terminal and set the password.
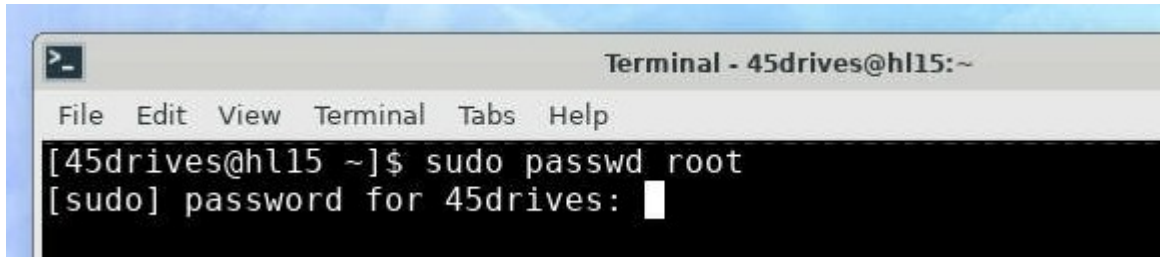
SETTING THE NETWORKING USING NMTUI

This is just an addition option to set up network other than using the UI. You can ignore this if you have already set the IP using the UI and move to the Houston login and config section.

⊕ You can run "ip -c a" to show your interfaces with colored IP addresses. This helps to distinguish what you're looking at.
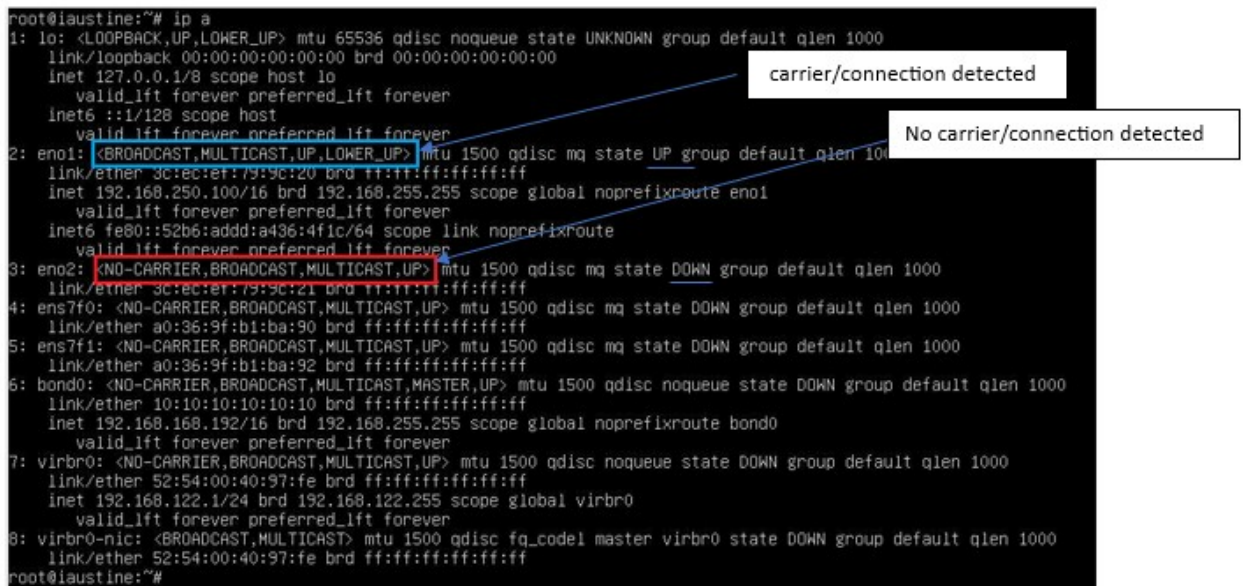
Fig – showing connection detected on the network interfaces

⊕ So, you can see in the above diagram that the interface eno1 has carrier detected which means there is a LAN cable connected to NIC slot.
⊕ Similarly, eno2 has no carrier which means its not connected. If it is connected and still not detected it will need further troubleshooting.
⊕ Once you have your interface connected with the carrier detected next step is to set a static ip for your HL4.

45HomeLab
A Division of 45Drives

⊕ We will be using this IP going forward to access the HL4.

Setting Static IP

⊕ Run the command "nmtui" to access NMTUI, network manager.
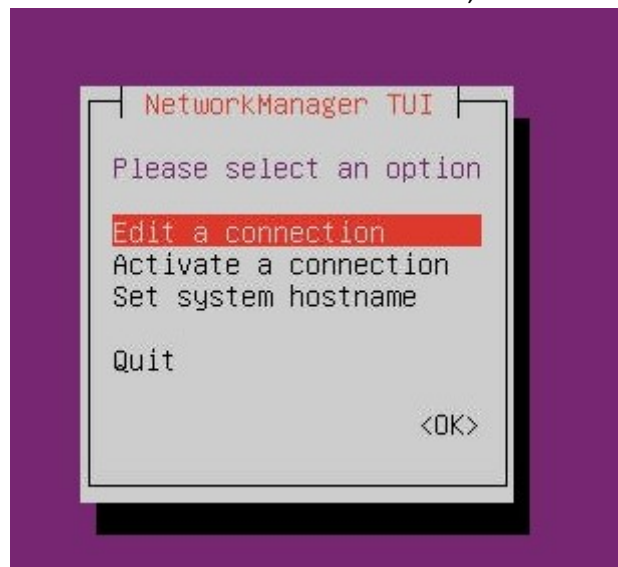
Fig: NMTUI

⊕ First, we will set a static IP. Click Edit a connection.
⊕ Select the interface that we want to set the static IP. This example it is eno1. Select that interface and click on edit.

45HomeLab
A Division of 45Drives
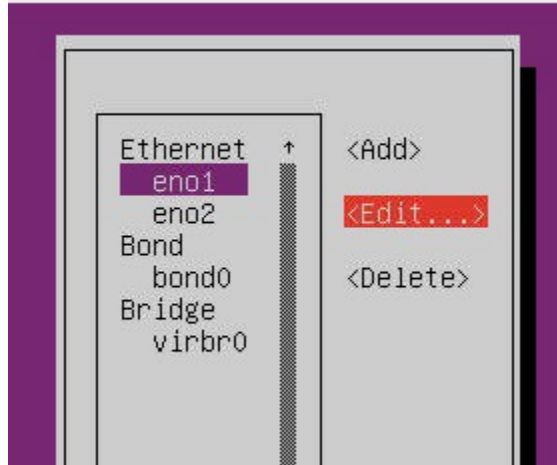
⊕ In case the interface names show up as wired connection 1,2 etc you can select that and confirm what the actual interface name is by checking device name like below screenshot.
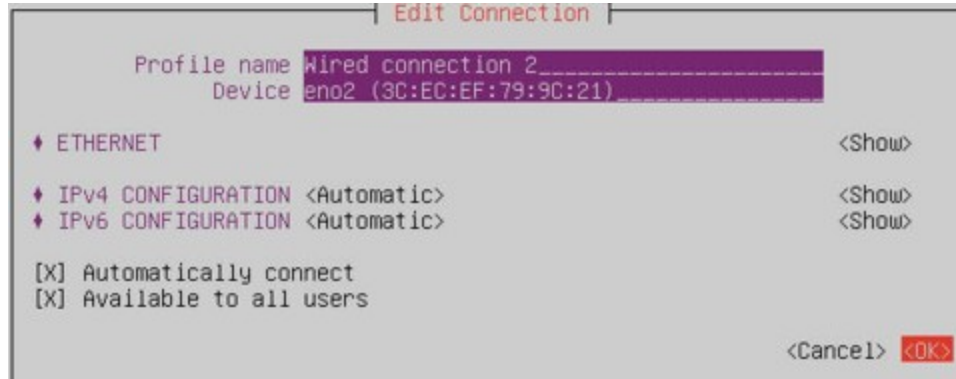
⊕ Navigate to IPv4 configuration and change automatic to manual and then click show.

45HomeLab
A Division of 45Drives
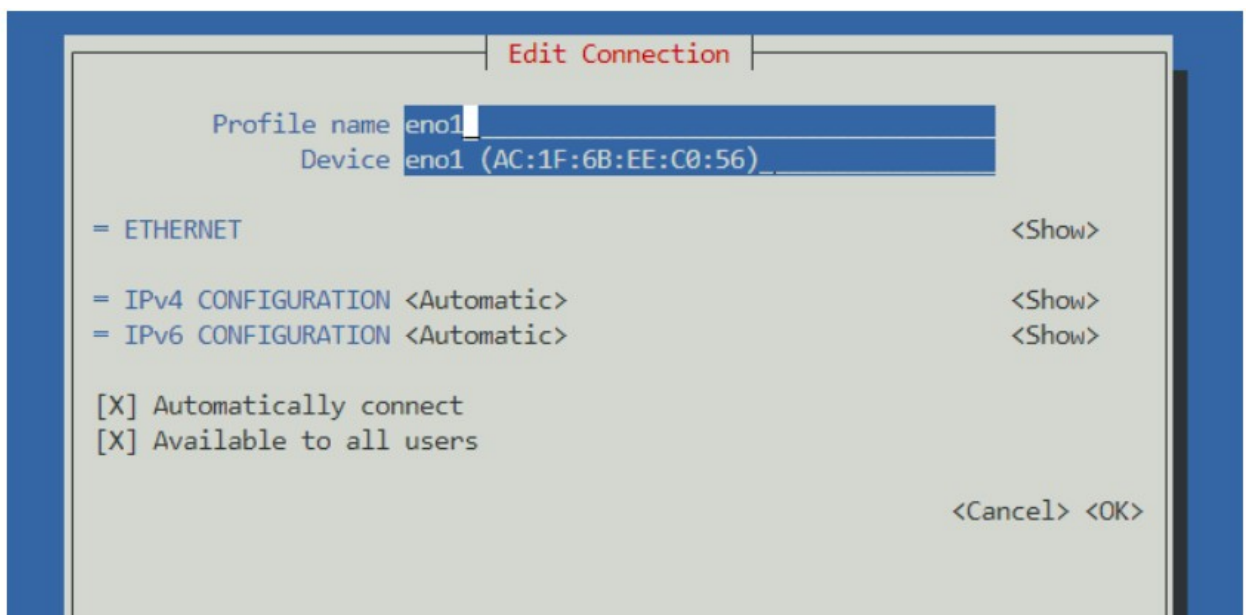
⊕ Here we can enter our static IP information. Be sure to remember to add your subnet after your IP address. (Most common would be /24, which correlates to a subnet mask of 255.255.255.0) If you will be joining a domain, you can set your HL4's DNS. Click OK. (Even if you aren't joining a domain, if you want to be able to resolve internet addresses via names you will need to give a DNS whether its a public one or an internal one. Because were setting a static IP, if you don't include a DNS at all you won't be able to download packages or anything because it needs DNS.)
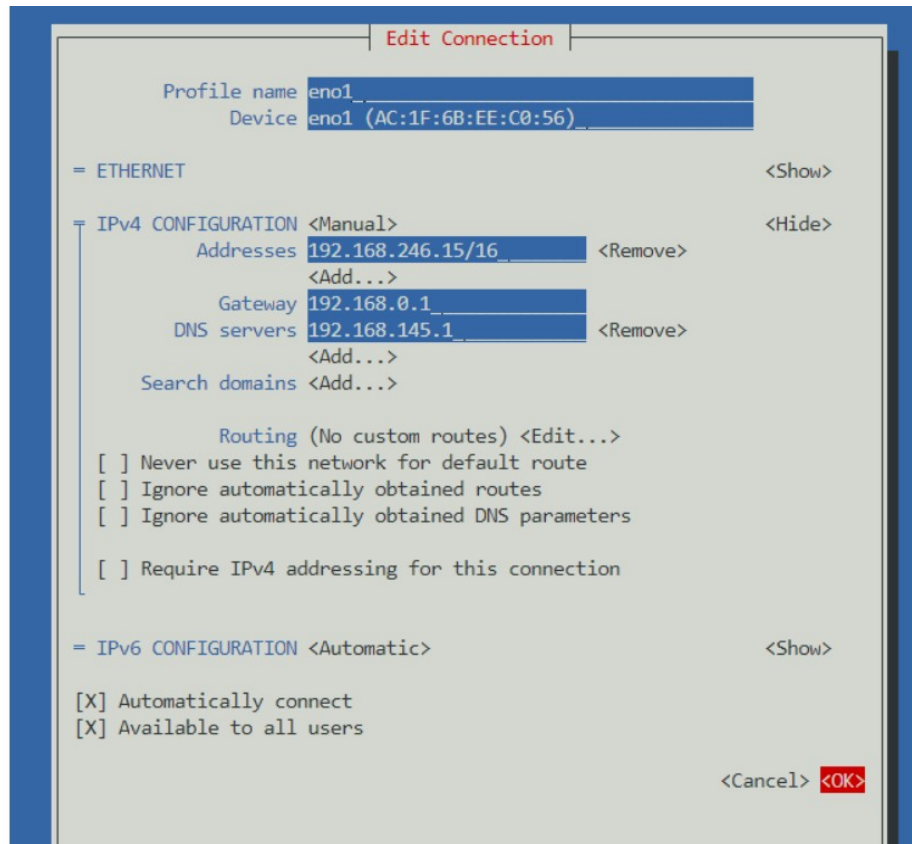
Fig – Setting static IP

⊕ Next, click Back and then click on Activate a connection.

45HomeLab
A Division of 45Drives

⊕ You will need to activate that interface .
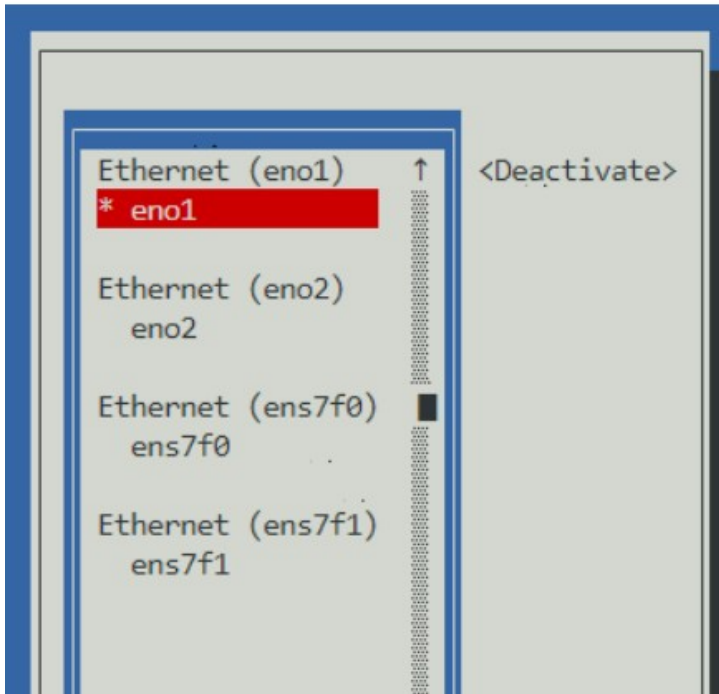
⊕ Deactivate and reactivate the interface. This resets the connection and ensures proper communication. (If you as SSH'd over the interface that you are working on, deactivating will kick you out of your SSH session, so you will need to do this when you are physically at the HL4.

Setting a Bond

⊕ Click Edit a connection, and click Add.
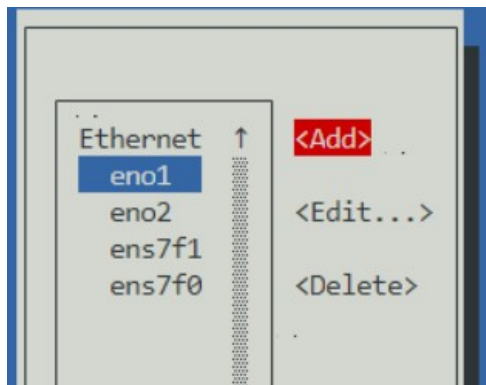
Fig – Add Bond

⊕ For the type of connection, select Bond.

45HomeLab
A Division of 45Drives
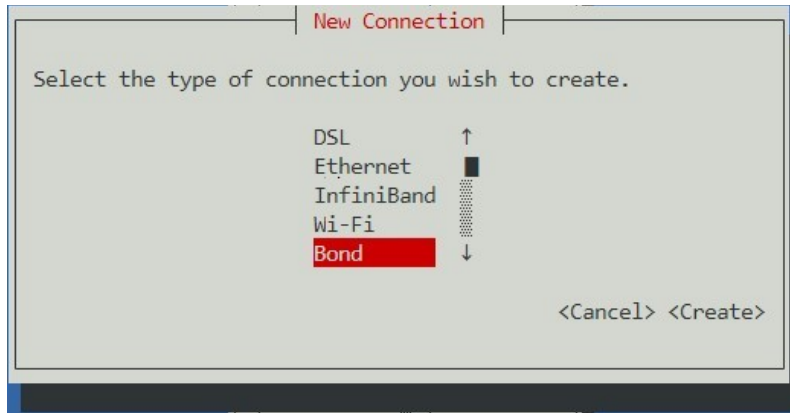
⊕ Enter your Profile name and Device name. For this example, we have used Bond1. Under Slaves, click Add and select Ethernet.
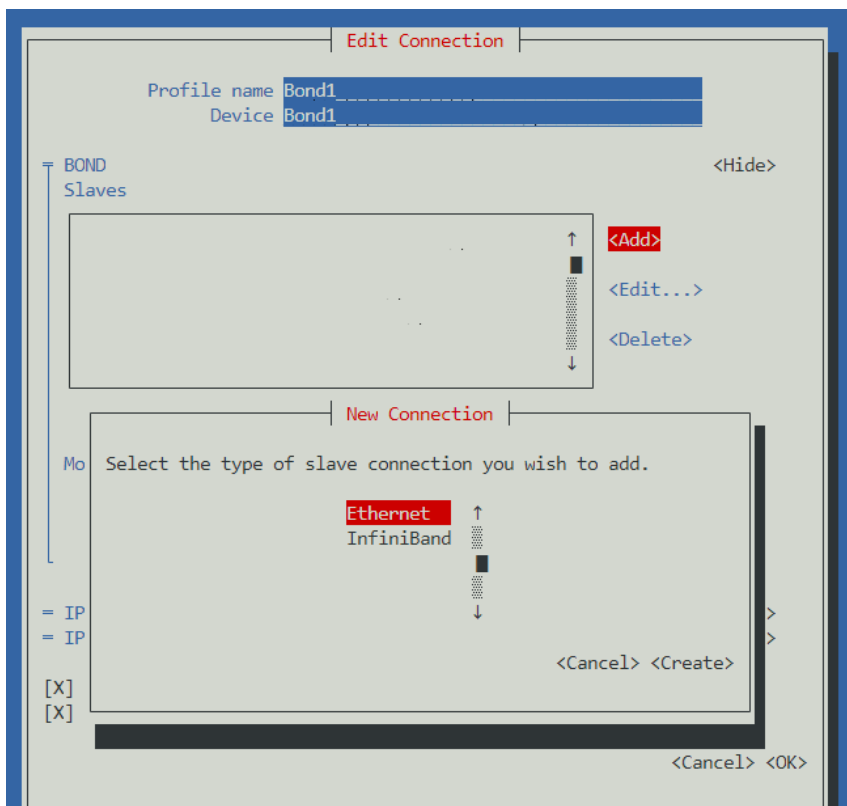
⊕ For this example, we will be using the NIC ports. Enter your interface name under Profile name and Device, and click OK.

⊕ Do the same for ens7f1. Your Slaves table should look like this.

45HomeLab
A Division of 45Drives

```
┌─ Edit Connection ─────────────────────────────────┐

              Profile name  Bond1_____
                    Device  Bond1_____

   ⊤ BOND                                              <Hide>
     Slaves
     ┌────────────────────────────────────────┬─┐
     │  ensf71                                 │↑│   <Add>
     │  ens7f0                                 │▓│
     │                                         │▓│   <Edit...>
     │                                         │▓│
     │                                         │█│   <Delete>
     │                                         │↓│
     └────────────────────────────────────────┴─┘
                      Mode  <Round-robin>
           Link monitoring  <MII (recommended)>
      Monitoring frequency  100_____  ms
             Link up delay  0_____   ms
           Link down delay  0_____   ms
         Cloned MAC address

   = IPv4 CONFIGURATION    <Automatic>                <Show>
   = IPv6 CONFIGURATION    <Automatic>                <Show>

   [X] Automatically connect
   [X] Available to all users

                                              <Cancel> <OK>
└───────────────────────────────────────────────────┘
```
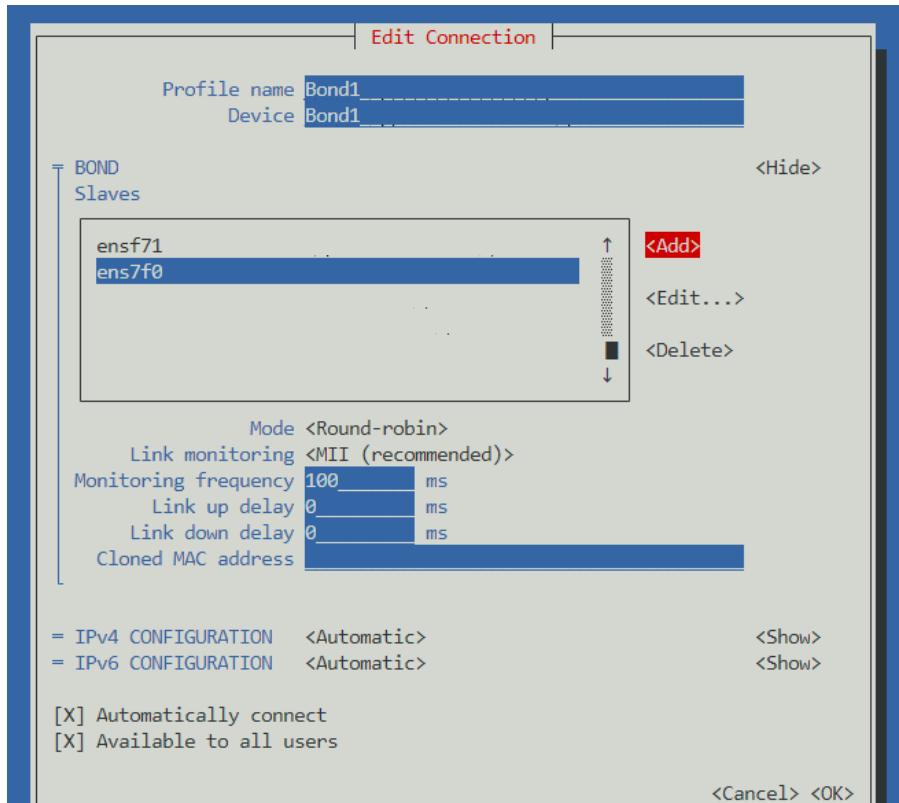
⊕ Add your IP address info for your bond, and click OK.

45HomeLab
A Division of 45Drives

```
                           ┤ Edit Connection ├                              ↑
│
│              . . . . . . . . . . . . . . . . . . . .      <Add>
│              . . . . . . . . . .                 . .      <Edit...>
│
│                                                          <Delete>
│
│                  . .               . .
│             Mode <802.3ad>
│    Link monitoring <MII (recommended)>
│ Monitoring frequency 100_____  ms
│       Link up delay 0_____  ms
│     Link down delay 0_____  ms
│    Cloned MAC address [_____]
│
│
│ ═ IPv4 CONFIGURATION   <Manual>                          <Hide>
│           Addresses 192.168.246.1/16_____  <Remove>
│                     <Add...>
│             Gateway 192.168.0.1_____
│         DNS servers 8.8.8.8_____  <Remove>
│                     <Add...>
│       Search domains <Add...>
│
│             . . . . . . . . . . . . . . . .
│             Routing (No custom routes) <Edit...>
│ [ ] Never use this network for default route
│ [ ] Ignore automatically obtained routes
│ [ ] Ignore automatically obtained DNS parameters
│
│ [ ] Require IPv4 addressing for this connection
│
│
│ ═ IPv6 CONFIGURATION   <Automatic>                       <Show>
│
│ [X] Automatically connect
│ [X] Available to all users
│
│                                              <Cancel> <OK>
│                                                                          ↓
```

⊕ Here we can see our bond has been created.

45HomeLab
A Division of 45Drives
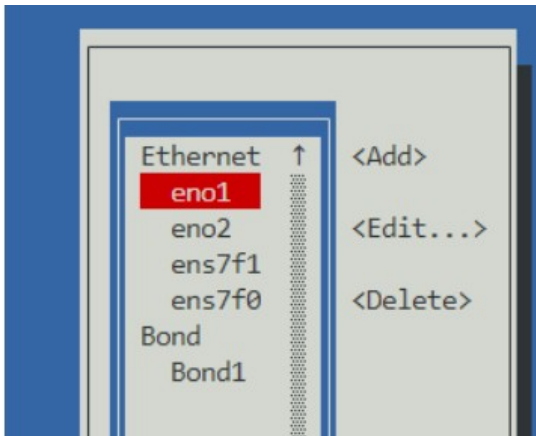
- Click Back, and go to Activate a connection. Highlight Bond1 and Deactivate it.
- Now activate it. Make sure the two slave ports are deactivated.
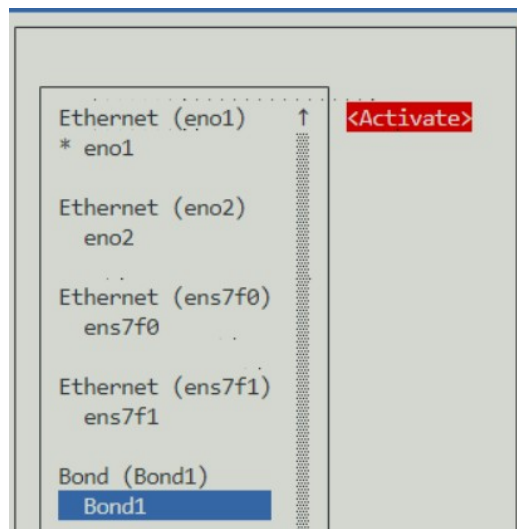
- Now, run another "ip a" command. Your bond is now listed.

45HomeLab
A Division of 45Drives

```
[root@krocky45d ~]# ip -c a
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: eno1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether ac:1f:6b:ee:c0:56 brd ff:ff:ff:ff:ff:ff
    inet 192.168.246.15/16 brd 192.168.255.255 scope global noprefixroute eno1
       valid_lft forever preferred_lft forever
    inet6 fe80::ae1f:6bff:feee:c056/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: eno2: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN group default qlen 1000
    link/ether ac:1f:6b:ee:c0:57 brd ff:ff:ff:ff:ff:ff
4: ens7f0: <NO-CARRIER,BROADCAST,MULTICAST,SLAVE,UP> mtu 1500 qdisc mq master Bond1 state DOWN group default qlen 1000
    link/ether a0:36:9f:2f:ff:40 brd ff:ff:ff:ff:ff:ff
5: ens7f1: <NO-CARRIER,BROADCAST,MULTICAST,SLAVE,UP> mtu 1500 qdisc mq master Bond1 state DOWN group default qlen 1000
    link/ether a0:36:9f:2f:ff:40 brd ff:ff:ff:ff:ff:ff permaddr a0:36:9f:2f:ff:42
9: Bond1: <NO-CARRIER,BROADCAST,MULTICAST,MASTER,UP> mtu 1500 qdisc noqueue state DOWN group default qlen 1000
    link/ether a0:36:9f:2f:ff:40 brd ff:ff:ff:ff:ff:ff
    inet 192.168.246.1/16 brd 192.168.255.255 scope global noprefixroute Bond1
       valid_lft forever preferred_lft forever
[root@krocky45d ~]#
```

HOUSTON LOGIN AND CONFIG

You should already have the Houston modules installed in your HL4. You can access Houston UI at https://SERVER-IP:9090 .

If the Houston UI is not loading do check if the cockpit service is running and enable it using below command

You can check using **systemctl status cockpit** and if its not active enable it using below commnd

systemctl enable --now cockpit.socket

You can use the root account you created and the password or the 45drives user as well to login. Root account would have the elevated privileges.

Rocky Linux

User name

Password

▶ Other options

Log in

Server: homelabs

Log in with your server user account.

HOUSTON
COMMAND CENTER

45HomeLab
A Division of 45Drives
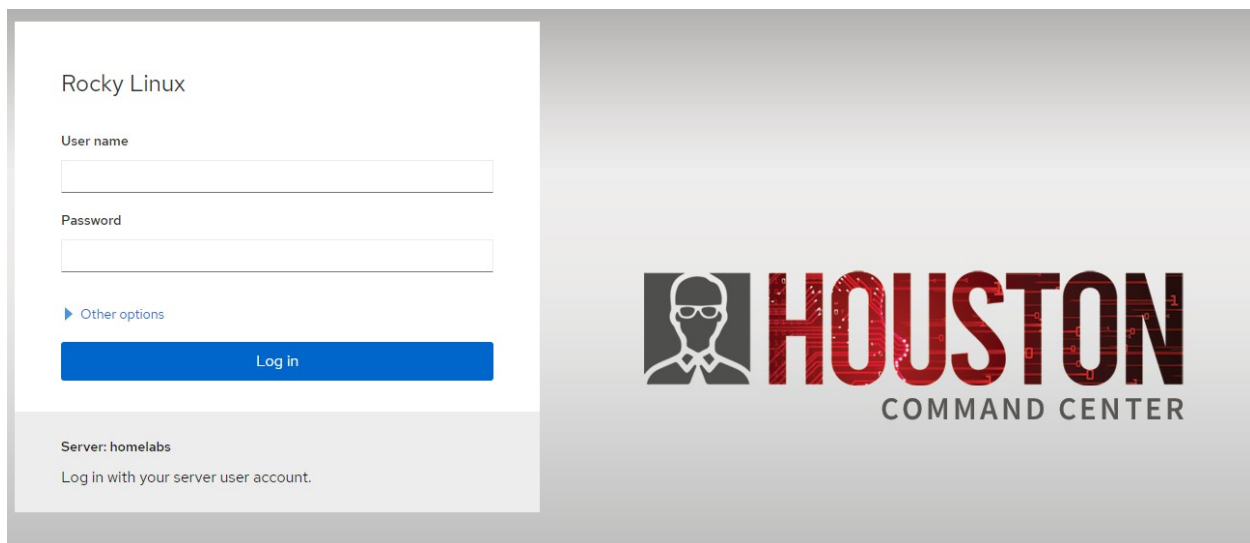
CREATING ZPOOL

- Once we're logged into Houston, you should see a list of modules. You will see the ZFS tab. Click on this to continue.
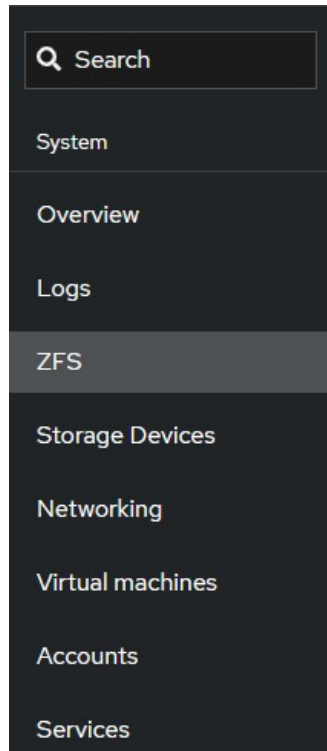
- On the top right of the ZFS page you will notice Storage Pools. Click on this to begin creating your first storage pool. When we create a pool with ZFS, we select the drives we want to be part of the pool. In our case all 15 drives

- Provide a name for your new storage pool, in this test environment we have named the pool tank in this example.
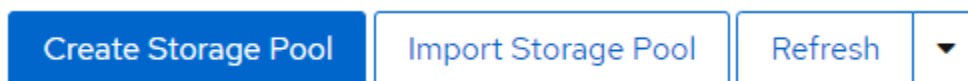- Select which raid you would like to use under "Virtual Device ". We usually recommend RAIDZ1 which is equivalent to RAID 5 and has one drive redundancy. You may use RAIDZ2 for 2 drive redundancy but will not have as much available data to utilize as the system can only hold 4 drives.
- Then select your hard drives and click on "Create" to finish. Ensure you uncheck Disks WWN, and then select "Device Alias" so the pool is created with the Disk Aliasing to easily identify drives (i.e 1-1, 1-2, etc.)

We do not recommend setting "refreservation" below 10%. With this disabled we can write to the ZFS Pool/Dataset and fill it entirely, at which point the pool may be unusable and data could be lost.

45HomeLab
A Division of 45Drives

Create Storage Pool

Name ⓘ	tank
Virtual Device	RaidZ2 ▾
Disks WWN	◯
Disks Identifier	Device Alias ▾

Disks ⓘ

☐ **12 TB ST12000NM0007-2A (5000c500b17dc72b)**
1-1
Physical Sector Size: 4 KiB

☐ **10 TB WDC_WD101KRYZ-01 (5000cca273d9632b)**
2-5
Physical Sector Size: 4 KiB

☐ **10 TB WDC_WD101KRYZ-01 (5000cca273d971ef)**
/dev/sdn ⚠
Physical Sector Size: 4 KiB

One or more disks is missing a Device Alias identifier.

One or more disks used to be a member of a Storage Pool. ⚠

Sector Size	4 KiB ▾
Record Size	128 KiB ▾
Deduplication	Off ▾
Refreservation	———————●————— 10 %

Options

✓◯ LZ4 compression

✓◯ Automatically expand storage pool when larger devices are added

◯ Automatically replace devices

◯ Automatic TRIM

◯ SELinux contexts for Samba

◯ Forcefully create storage pool

Cancel   **Create**

CREATING ZFS FILESYSTEM/DATASET

⊕ With your new storage pool created, we can now create some datasets to share out. You will see Create Filesystem. Click on this to continue.

45HomeLab
A Division of 45Drives

Storage Pools							
Name	Health	Size	Allocated	Free	Fragmentation	Usage	
∨ tank	⊘ ONLINE	81.86 TiB	4.14 MiB	81.86 TiB	0 %		⋮

File Systems   Snapshots   Status

7/12/2021, 9:36:17 AM   3

Name	Available	Used	Snapshots	Refreservation	Record Size	Compression	Deduplication	Share	Mounted	Replication	
tank	60.41 TiB	2.29 MiB	0 B	0 B	128 KiB	LZ4	Off	Off	Yes	No	⋮

- In the new window that appears, give your new filesystem a name. We can leave all of the options at the default settings. In this example, we have used nfs.
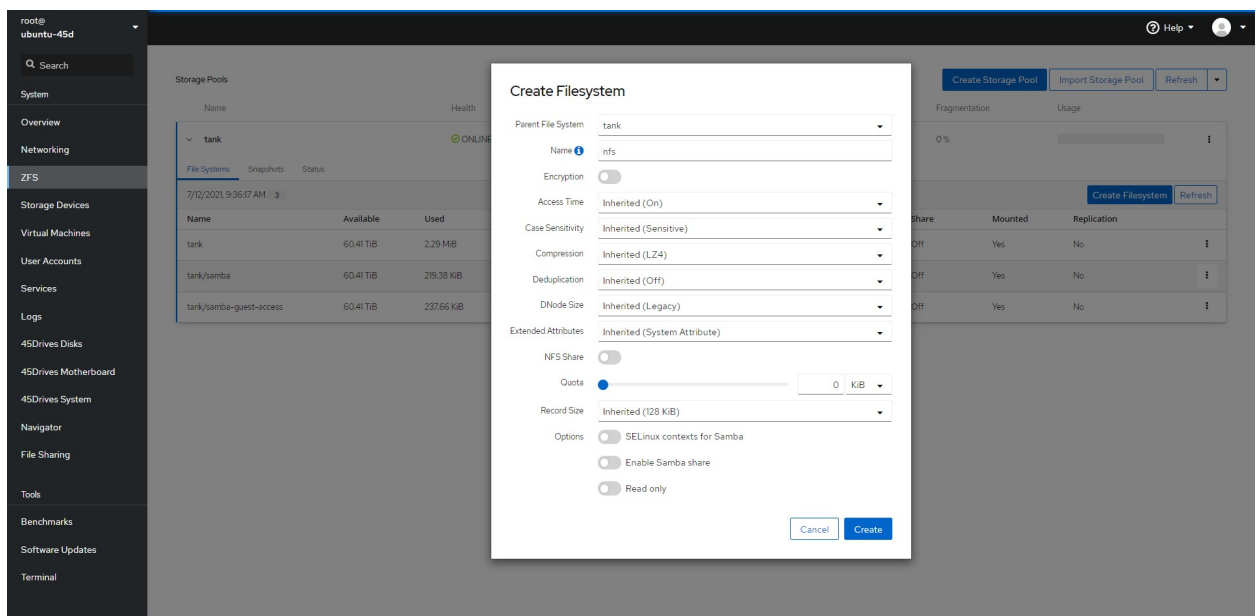
- If you wish you can change those other parameters as per your preference and even set quota as well.
- Once you click on create you will have your dataset created.

- Now you will be able to see your ZFS pool, datasets, and drives within the File Systems Tab, and Status Tab within the ZFS module.
- If you are to run zpool status command within the Terminal you should see the output of the ZFS Pool, it's VDEVs, and Disks.

45HomeLab
A Division of 45Drives

⊕ You can create as many datasets you want based on how many shares you are planning to create.

If you are unable to create the pool, ensure the drives you are using are free of any partitions.

Ensure the disks you are using to create the pool are of the same size

CREATING YOUR SHARES

Once you have your pool and dataset created next step is to create network shares. You can either create SAMBA/NFS shares.

Samba/SMB shares

Points to note

- Make sure SMB Services Running and Enabled
- SMB Ports Open on Firewall (133/tcp, 445/tcp and 137/udp, 138/udp)

⊕ In Houston UI, navigate to the File Sharing tab. And click on the Samba tab, if not selected.

45HomeLab
A Division of 45Drives
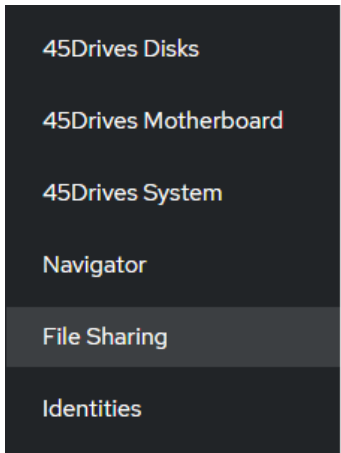
✛ Once here, we can begin configuring our SMB Shares
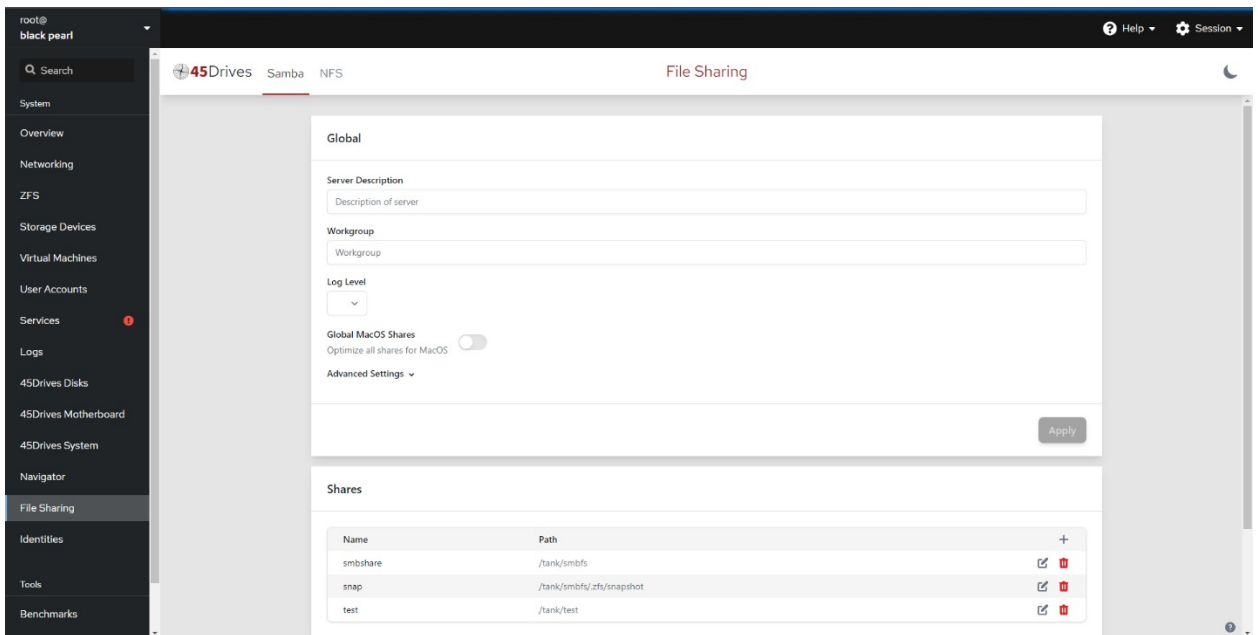
✛ First, we can add any options to the Global SMB configuration.
✛ Here we can change the Server Description, the Workgroup, Log Level, and add any additional parameters to the SMB configuration in the Advanced Settings box by clicking the down arrow. For example, here we've added a few parameters to help with MacOS performance on an SMB share.

45HomeLab
A Division of 45Drives
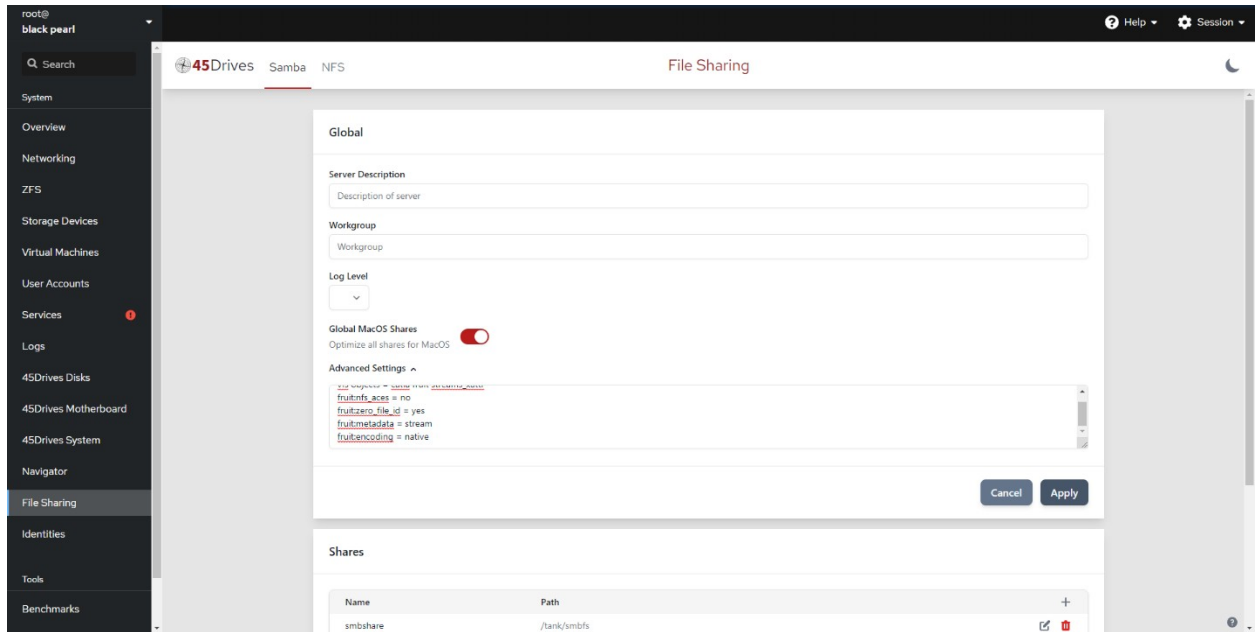
⊕ Next, we can create an SMB share, we can scroll down to Shares and select the Plus button to begin.

⊕ Here we can set the SMB Share Name, a Description, the Path to be shared out. It will be usually /pool_name/dataset_name (you can now create the path here instead of ZFS)

⊕ if we are domain joined, we can select to use Windows ACLs.

⊕ We can specify any Valid Users and Valid Groups, allow Guest Access, make the SMB share Read Only, make it Browsable, and enable Windows ACLs, as well as some other options.

Adding options to the SMB share

⊕ We can also add any additional options in the Advanced Settings box via the drop-down arrow.

⊕ In the Advanced Settings box, we entered "inherit permissions = yes" ourselves. This is to make sure that the permissions are inherited from the parent folder.

⊕ There are a few preselected options we can enter into the Advanced Settings box by selecting one of the buttons below: Shadow Copy, MacOS Share, Audit Logs.

⊕ Here we can see we've created a share called "samba" with a description of "smb-share". It is pathed to our ZFS dataset at "/tank/samba". We have added the "administrator" to Valid Groups, and left the share Browsable.

45HomeLab
A Division of 45Drives
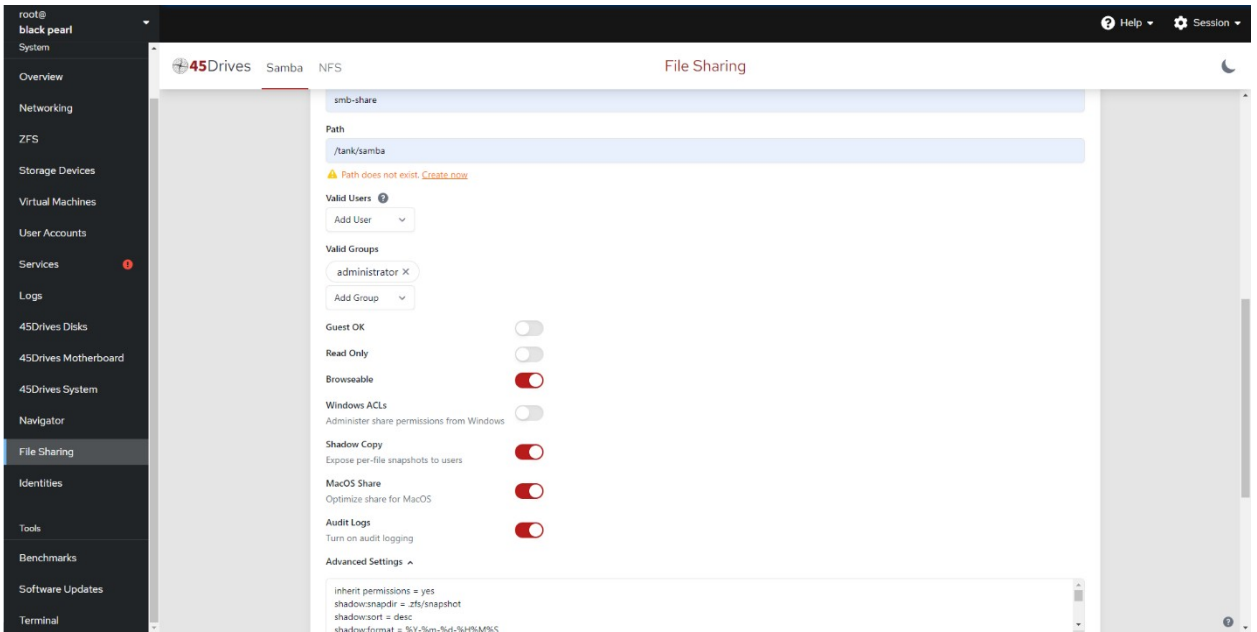
⊕ Here we can see our SMB share is created.

✓

Verification
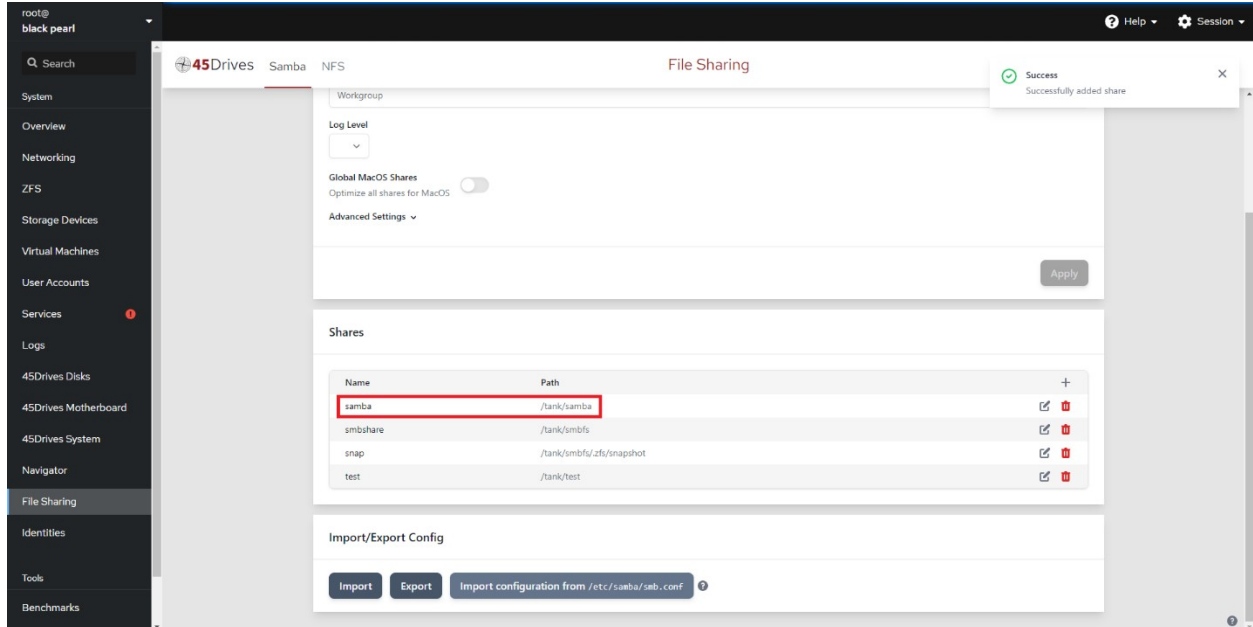
- If you were to run "**testparm -s**" on the command line you will see your samba share has been added and configured properly in its own section.

45HomeLab
A Division of 45Drives

Setting up Local User Access SMB Shares

This is to create local users and groups and add them to provide access to your samba share.

- Go to the identities tab in Houston
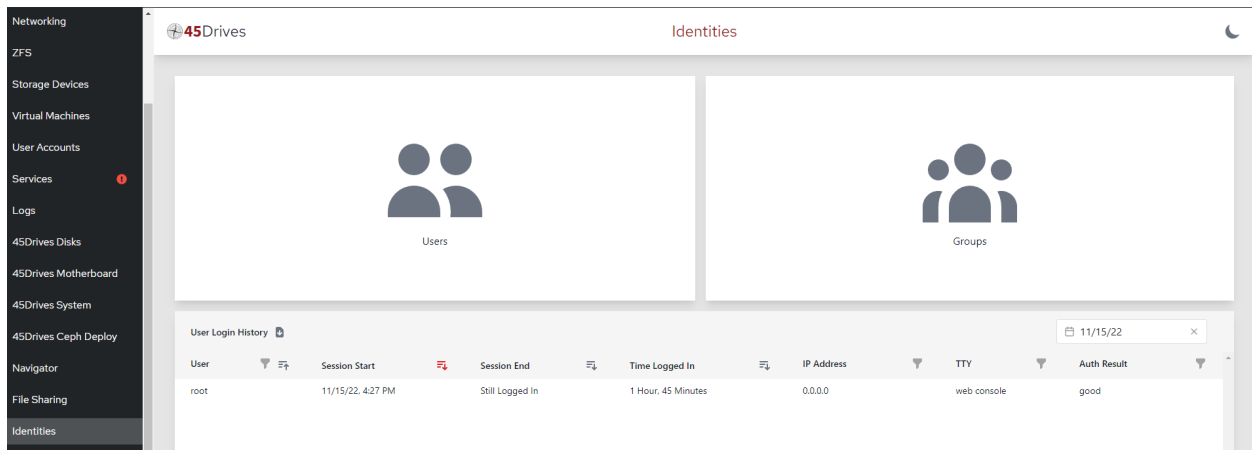
- Click the 'Users' section, this will open a page with a list of users. Select "New User".
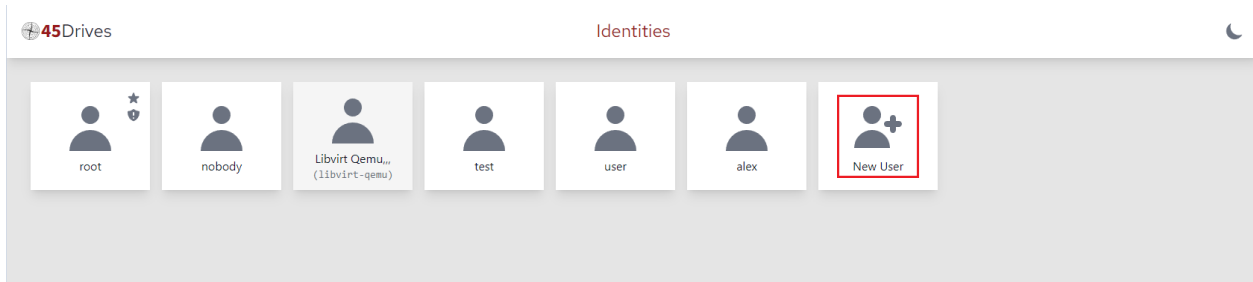
⊕ Fill in the Username, Full Name/Description. Then click "Apply". It is also possible to assign this new user to a group at the bottom of the prompt. If applicable it can be done here, or a user can be assigned at a later time.

⊕ A prompt will come up to set the users password. This is a local Linux password, it can be used to SSH into the machine, or to authenticate to Houston.

45HomeLab
A Division of 45Drives

Set login password for smbuser

The password should satisfy the following requirements:
- one lowercase letter ✓
- one uppercase letter ✓
- one number ✓
- one special character ✓
- 8 characters long ✓

No Password **Apply**

⊕ The user generation will complete successfully. The user should now be able to authenticate to Houston or SSH with that password.

⊕ The below sections will detail assigning a user to a specific group, and setting an SMB password if applicable.

Assigning a Linux Group to a User

⊕ In the "Users" section the new user should be seen. Click on the user to assign a group.

45Drives   Identities

root   nobody   Libvirt Qemu,,,   test   user   alex   Samba User   New User
                (libvirt-qemu)   (smbuser)

⊕ Click on the "+" under the Groups section.

Groups	+
smbuser (primary group)	

45HomeLab
A Division of 45Drives

⊕ A dropdown will pop up and you can now select a group to assign to the user.
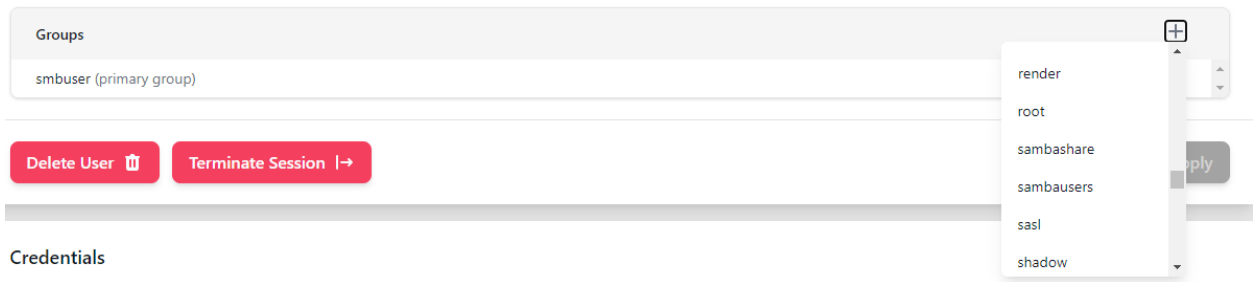
⊕ The newly added group should be displayed under the Groups section. Click "Apply" to save changes.
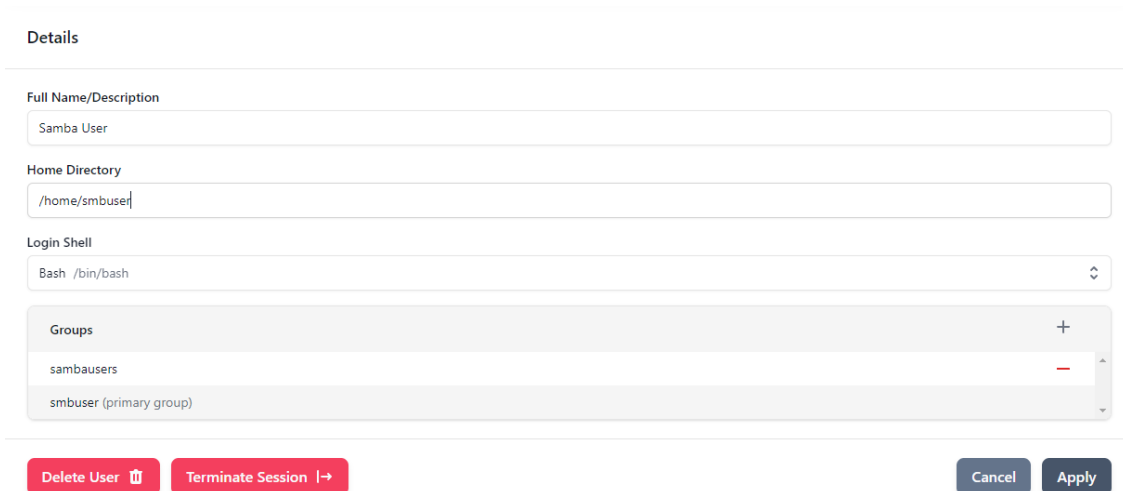
Configuring a Samba Password for a User

⊕ Select the user you wish to assign a Samba password. The Samba password will need to be set to allow users to authenticate to Samba shares. It can be the same as the Linux password, but does not have to be. Click on "Set Samba Password". Enter a password and click "Apply".

45HomeLab
A Division of 45Drives

Samba User (login=smbuser,uid=1003,gid=1004)

Credentials

User Login

[Change Account Password]   [Lock Account Password 🔓]   [Edit Password Expiry]

ⓘ Password never expires.

Samba

[Set Samba Password]

SSH

[Generate SSH Key Pair]   [Test Passwordless SSH]

Authorized SSH Access Keys	+
No keys. Click '+' to add one.	

Set Samba password for smbuser

🔒 ⦁⦁⦁⦁⦁⦁⦁⦁⦁⦁⦁ 👁̶

🔒 ⦁⦁⦁⦁⦁⦁⦁⦁⦁⦁⦁ 👁̶

The password should satisfy the following requirements:
- one lowercase letter ✓
- one uppercase letter ✓
- one number ✓
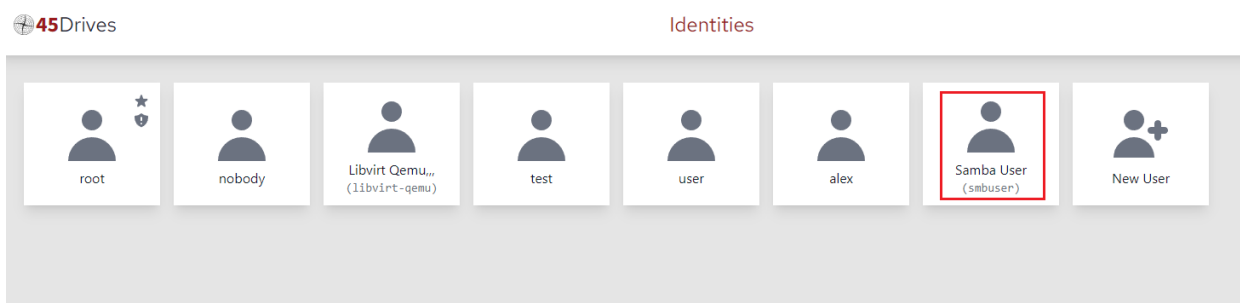- one special character ✓
- 8 characters long ✓
- different from user name ✓
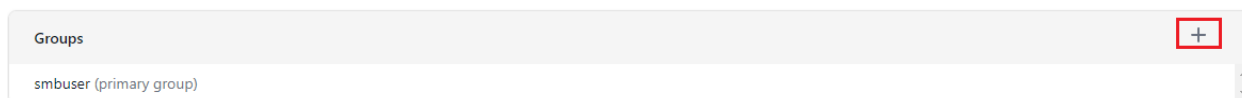
[Cancel] [Apply]

45HomeLab
A Division of 45Drives

Set SMB permissions using local users and group

⊕ Click on the edit button of the share you want to add the user to grant the permissions.

⊕ Click on edit permissions

⊕ Select the owner and the group for the share and set the permissions as well as per your preference.

45HomeLab
A Division of 45Drives

Share Directory Permissions

	Read	Write	Execute
Owner	☑	☑	☑
Group	☑	☑	☑
Other	☑	☐	☑
Mode		rwxrwxr-x (775)	

Owner

admin ⌄

Group

smbgroup ⌄

Cancel   Apply

⊕ You can also select valid users and groups if you want to provide access to multiple of them.

Valid Users ❓

Add User ⌄

Valid Groups

Add Group ⌄

⊕ Now you can verify by connecting the share.

Connecting to SMB Share on Windows and MacOS

Connecting to SMB Share on Windows

⊕ On a Windows client, go to This PC in File Explorer.
⊕ Right click and select Map network drive.

45HomeLab
A Division of 45Drives
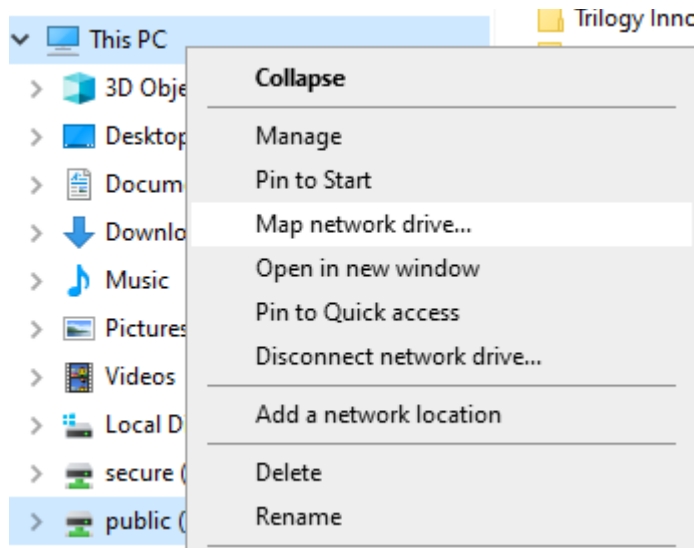
⊕ In this menu, enter the server IP address in the following format **\\SERVER-IP\SHARE-NAME** and select **Connect using different credentials**.

⊕ Enter the login to connect to the SMB share.
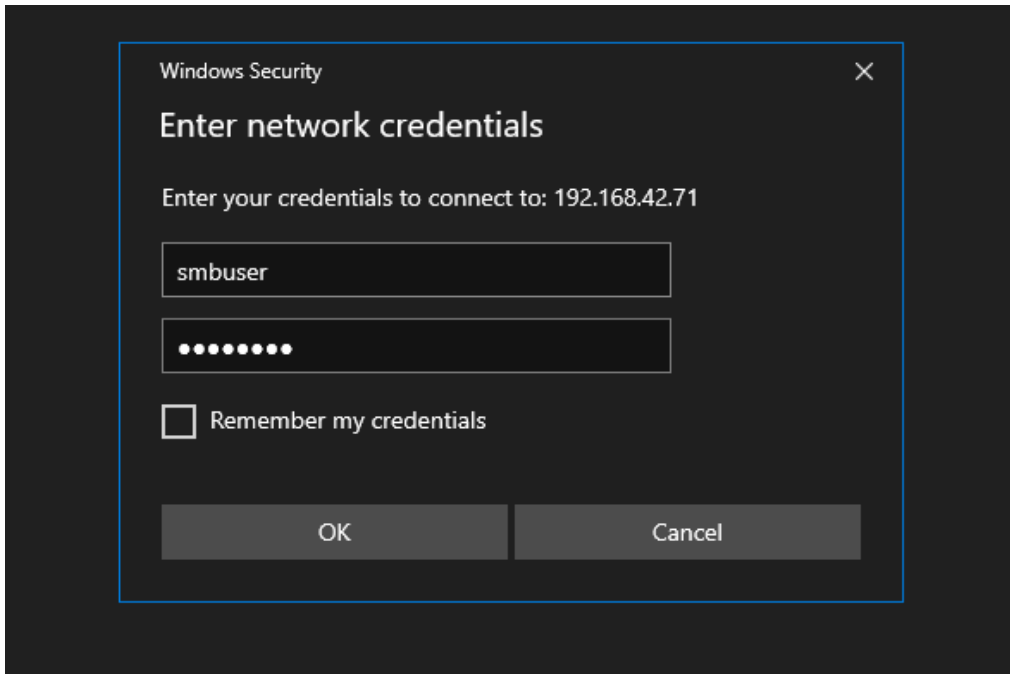
45HomeLab
A Division of 45Drives

Here we can see our SMB share is connected, and we can create a folder.

45HomeLab
A Division of 45Drives

Connecting to SMB Share on MacOS

⊕   At the toolbar on the top of the screen, press Go and then "Connect to Server…".

⊕   In the address bar, enter **smb://SERVER-IP/SHARE** and click **Connect**. We can also select the plus at the bottom to save this information if we need to reconnect.

45HomeLab
A Division of 45Drives
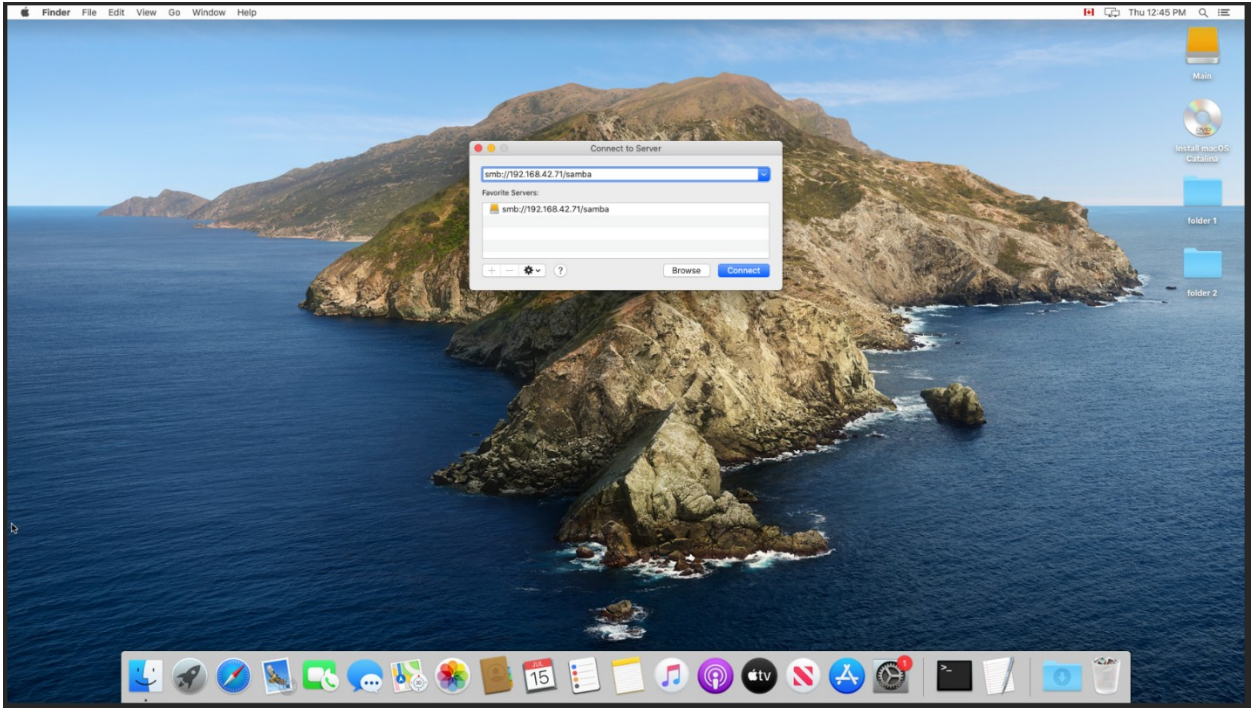
⊕ Enter the login to connect to the SMB share.

⊕ Here we can see our SMB share is connected, and we can create a folder.

45HomeLab
A Division of 45Drives
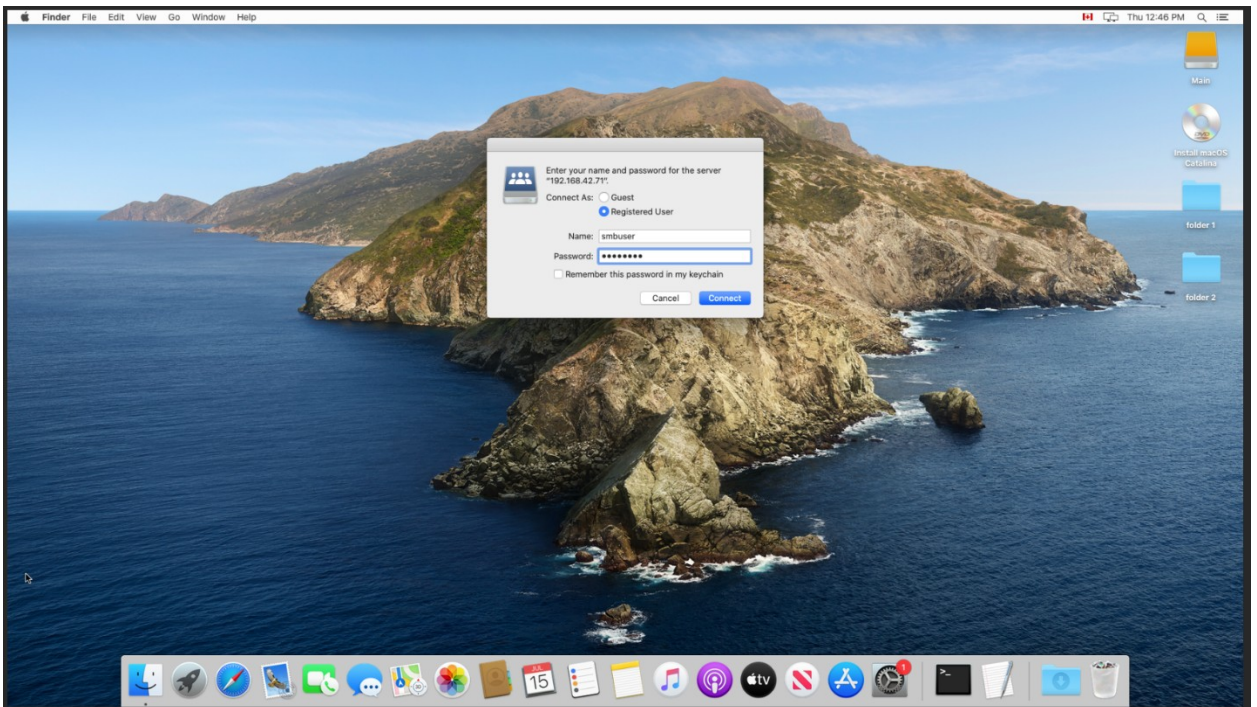
Managing NFS in Houston UI

- NFS Packages Installed
- NFS Services Running and Enabled
- NFS Ports Open on Firewall (2049/tcp, and 2049/udp if NFSv3)

⊕ In Houston UI, navigate to the File Sharing tab.

⊕ Once in that page, we can select the NFS Tab to begin configuring our NFS shares.
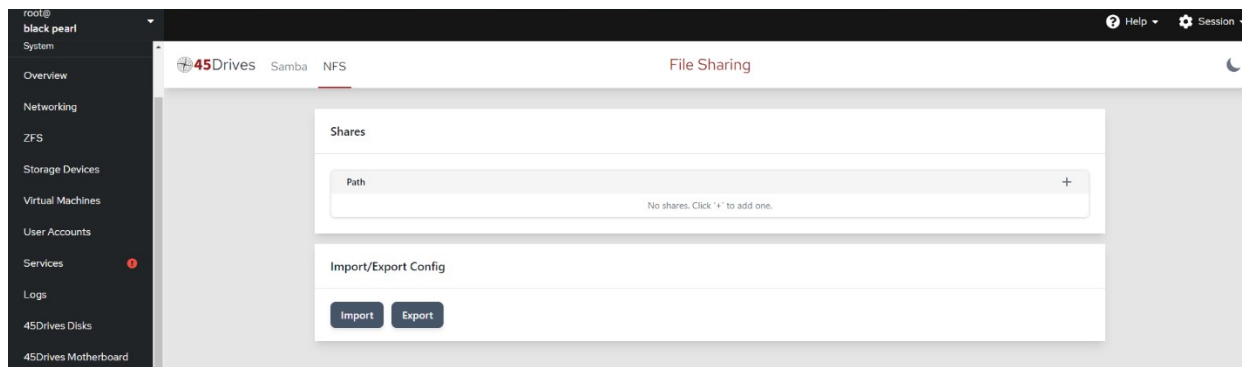
⊕ We first begin to create our NFS export by selecting the Plus Icon on the right.

⊕ We can define the path to be shared out. It would be usually /pool_name/dataset_name

45HomeLab
A Division of 45Drives

- select the range of IPs for client access, and add any additional NFS options to our share.
- If we leave Client IP empty, it will default to everyone, and if we leave Options empty, it will default to 'rw,sync,no_subtree_check'
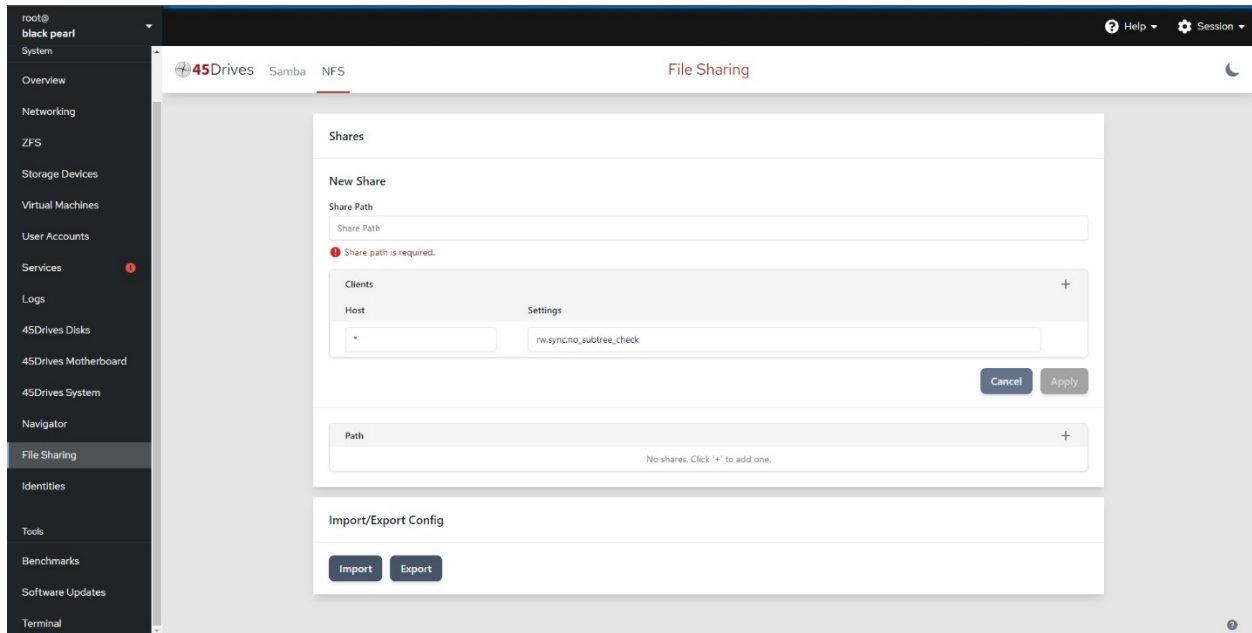
- Here we create an NFS share, with a path to our NFS dataset at /tank/nfsfs. We have set the Client IP to be available to **192.168.*.*.** For settings you can give "**rw,sync,no_subtree_check,no_root_squash**"

45HomeLab
A Division of 45Drives
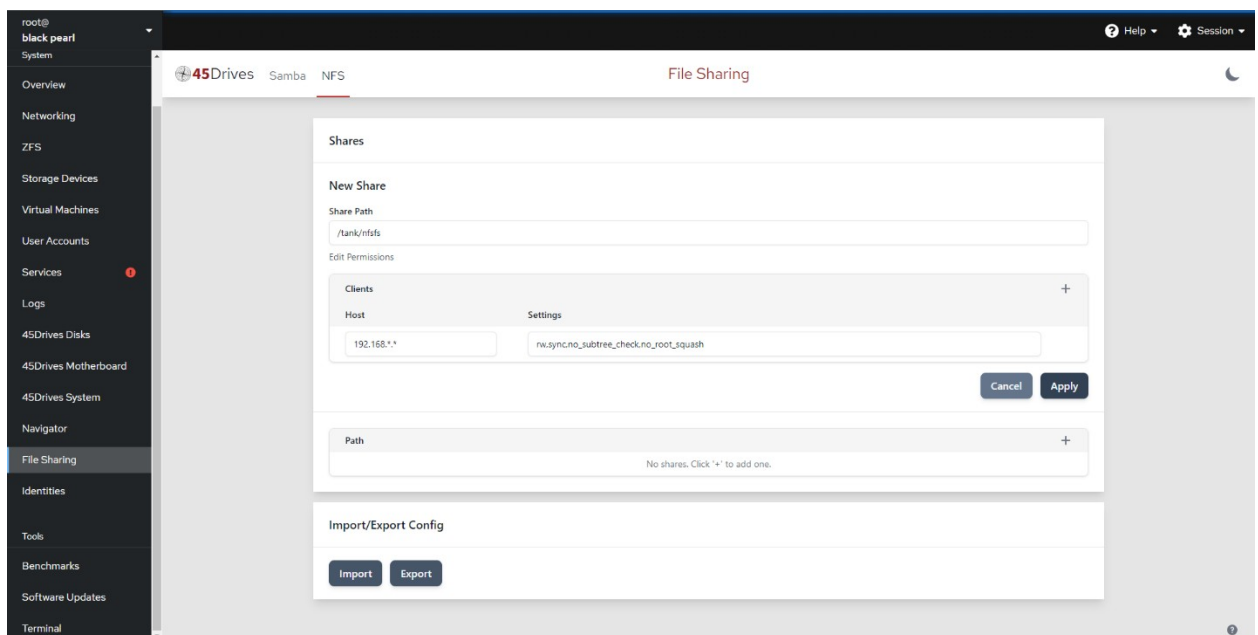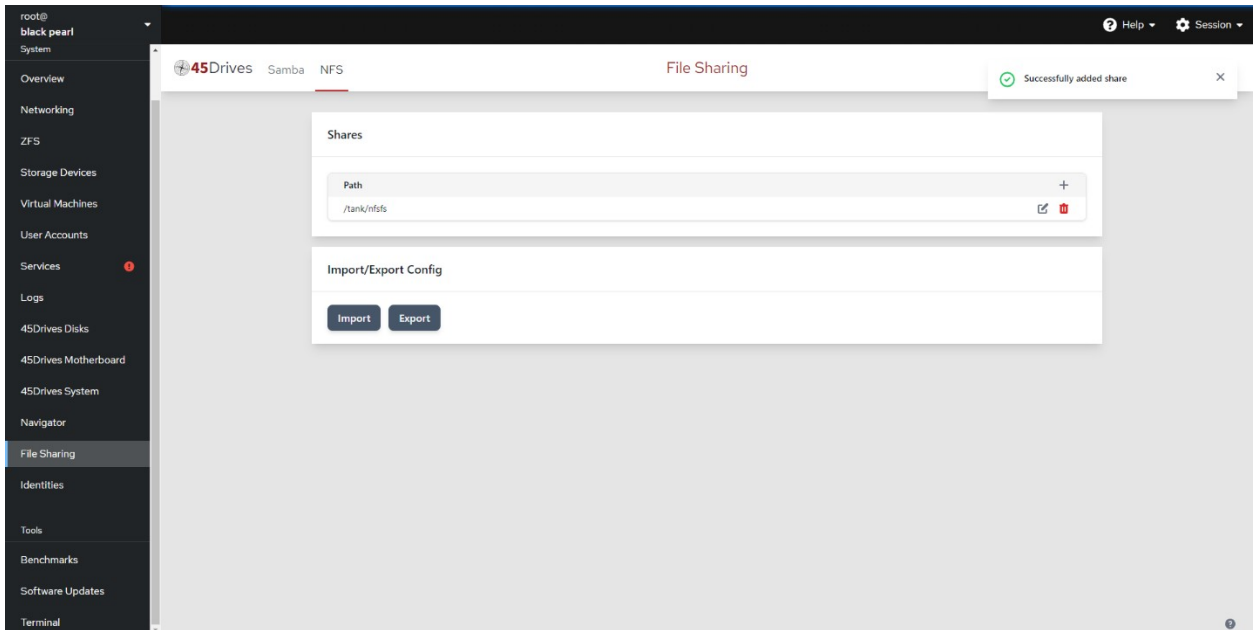
🔷 If you haven't opened the firewall ports then follow the below steps
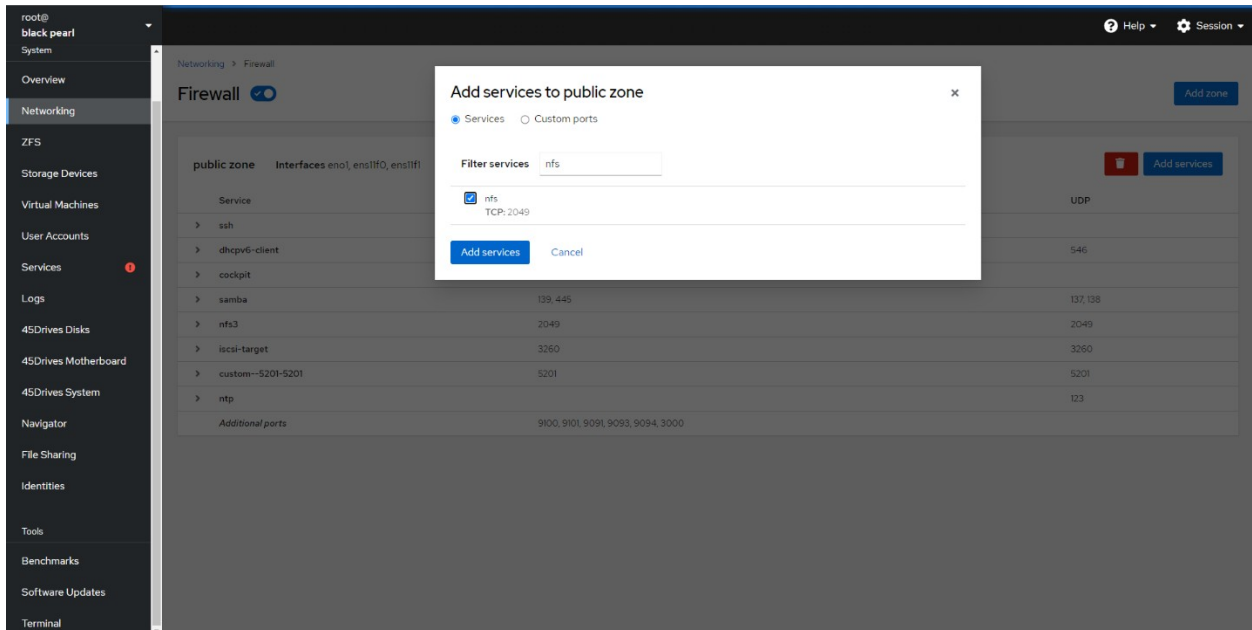
Mounting NFS Share to Linux Client

Here we will discuss on the process of mounting an NFS share to a Linux client and to mount on reboot.

🔷 Go the terminal and run the command below to install the nfs package

45HomeLab
A Division of 45Drives

install nfs-utils

⊕ To mount an NFS share, first create a directory to mount it to.

<div align="center">

`mkdir /mnt/(mount_point)`

</div>

eg :

```
mkdir /mnt/nfs_share
```

⊕ Now use this command to mount it to the share. Edit the fields for your specific case, i.e. Server IP, Pool Name, and Share Name. See example below.

mount -t nfs {ServerIP}:/(pool_name)/(nfs_share_name) /mnt/(mount_point)

```
mount -t nfs 192.168.35.39:/tank/nfs_sharetest /mnt/nfs_share
```

Add Mount on Reboot

⊕ To allow the share to mount on reboot, you will need to edit the fstab. You can use your preferred text editor, here we have used vim.

  vim /etc/fstab

⊕ Add the mount point in the format see below.

{ServerIP}:/(pool_name)/(share_name) /mnt/(mount_point) nfs defaults,_netdev 0 0

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
/dev/disk/by-id/md-uuid-5bef0df4:f65beb63:7f835222:41ec8940 none swap sw 0 0
# / was on /dev/md125p1 during curtin installation
/dev/disk/by-id/md-uuid-2a10c4dd:c02a5098:0ee79f38:2078a857-part1 / ext4 defaults 0 0
# /boot was on /dev/md/swap during curtin installation
/dev/disk/by-id/md-uuid-5fbc9b25:25381cd8:2b507526:40ea9196 /boot ext4 defaults 0 0
/swap.img       none    swap    sw      0       0
/dev/RBD-VOLGROUP/RBD-LVM       /mnt/rbd        xfs     defaults,_netdev 0 0
192.168.35.39:/tank/nfs_sharetest /mnt/nfs_share        nfs     defaults        0 0
```

Verify Share Mounted

⊕ To ensure the share has mounted, you can run the command **df**.

45HomeLab
A Division of 45Drives

```
root@ubuntu-45d:~# df
Filesystem                        1K-blocks      Used   Available Use% Mounted on
udev                               65907972         0    65907972   0% /dev
tmpfs                              13190628      4004    13186624   1% /run
/dev/md126p1                      117313456  16268152    95043028  15% /
tmpfs                              65953124         0    65953124   0% /dev/shm
tmpfs                                  5120         0        5120   0% /run/lock
tmpfs                              65953124         0    65953124   0% /sys/fs/cgroup
/dev/md125                          4057792    221160     3610792   6% /boot
tank                            36654107136       256 36654106880   1% /tank
tank/samba-windows-acl          29137914624       256 29137914368   1% /tank/samba-wind
tank/nfs                        29137914624       256 29137914368   1% /tank/nfs
tank/samba                      29137914624       256 29137914368   1% /tank/samba
tank/smb-windows                29137914624       256 29137914368   1% /tank/smb-window
tmpfs                              13190624         0    13190624   0% /run/user/0
192.168.35.39:/tank/nfs_sharetest  2798961664       0  2798961664   0% /mnt/nfs_share
```

Verify Mount on Reboot

- ⊕ To ensure the share will mount on reboot after editing the /etc/fstab, unmount the share.

  umount /mnt/nfs_share

- ⊕ Remount the share using the following command, this command will mount all shares in the /etc/fstab file.

  mount -a

- ⊕ Run the command **df** again to ensure the share was mounted.

45HomeLab
A Division of 45Drives

SETUP ISCSI STORAGE

- Install necessary packages for iSCSI

$$\textbf{dnf -y install targetcli}$$

- Create a ZFS dataset with your naming scheme of choice, in this case we'll use "images".
- Multiple iSCSI LUNs can be stored in this singular dataset or new datasets can be created for each iSCSI LUN.

45HomeLab
A Division of 45Drives

Create Filesystem

Parent File System	tank
Name ⓘ	images
Encryption	(toggle off)
Access Time	Inherited (On)
Case Sensitivity	Inherited (Sensitive)
Compression	Inherited (LZ4)
Deduplication	Inherited (Off)
DNode Size	Inherited (Legacy)
Extended Attributes	Inherited (System Attribute)
NFS Share	(toggle off)
Quota	0 KiB
Record Size	Inherited (128 KiB)
Options	◯ SELinux contexts for Samba
	◯ Enable Samba share
	◯ Read only

Cancel   Create

Configuring iSCSI Target

✦   Before using targetcli, we'll need to create a sparse image file. Creating large files in targetcli can be a very time-consuming process.

```
cd /tank/images
dd if=/dev/zero of=lun0 bs=1 count=0 seek=1T
```

root@rocky-45d:/tank/images

```
[root@rocky-45d ~]# cd /tank/images
[root@rocky-45d images]# dd if=/dev/zero of=lun0 bs=1 count=0 seek=1T
0+0 records in
0+0 records out
0 bytes copied, 8.9868e-05 s, 0.0 kB/s
[root@rocky-45d images]# ls
lun0
[root@rocky-45d images]#
```

45HomeLab
A Division of 45Drives

- The above will create a 1 TB sparse image file named "lun0"
- Now we'll launch targetcli and create a fileio backstore with the sparse image we've created.

```
targetcli
cd backstores/fileio
create lun0 /tank/images/lun0
```

```
[root@rocky-45d images]# targetcli
Warning: Could not load preferences file /root/.targetcli/prefs.bin.
targetcli shell version 2.1.53
Copyright 2011-2013 by Datera, Inc and others.
For help on commands, type 'help'.

/> cd backstores/fileio
/backstores/fileio> create lun0 /tank/images/lun0
Created fileio lun0 with size 1099511627776
/backstores/fileio>
```

- Verify the block device has been created in targetcli

```
/backstores/fileio> ls
o- fileio ................................................................ [Storage Objects: 1]
  o- lun0 ................................................. [/tank/images/lun0 (1.0TiB) write-back activated]
    o- alua ................................................................ [ALUA Groups: 1]
      o- default_tg_pt_gp ........................................... [ALUA state: Active/optimized]
/backstores/fileio>
```

Create an iSCSI Target

- Create the iSCSI target by selecting a name
- Naming scheme: iqn.YYYY-MM.REVERSE.DOMAIN.NAME:unique
- For this example, we'll be using "iqn.2021-11.fourfive.lab:test.environment"

```
cd /iscsi
create iqn.2021-11.fourfive.lab:test.environment
```

```
/backstores/fileio> cd /iscsi
/iscsi> create iqn.2021-11.fourfive.lab:test.environment
Created target iqn.2021-11.fourfive.lab:test.environment.
Created TPG 1.
Global pref auto_add_default_portal=true
Created default portal listening on all IPs (0.0.0.0), port 3260.
```

Create iSCSI LUNs

45HomeLab
A Division of 45Drives

cd iqn.2021-11.fourfive.lab:test.environment/tpg1/luns
create /backstores/fileio/lun0

```
/iscsi> cd iqn.2021-11.fourfive.lab:test.environment/tpg1/
/iscsi/iqn.20...ironment/tpg1> cd luns
/iscsi/iqn.20...ent/tpg1/luns> create /backstores/fileio/lun0
Created LUN 0.
```

Configure ACL (Initiator IQN)

- ⊕ When creating the ACL, it needs to be the name of the initiator IQN name of the client that will be connecting to the LUN.
- ⊕ In this example, the IQN name of the client will be "iqn.2019-05.com.archie:client"

```
/iscsi> cd iqn.2021-11.fourfive.lab:test.environment/tpg1/acls
/iscsi/iqn.20...ent/tpg1/acls> create iqn.2019-05.com.archie:client
Created Node ACL for iqn.2019-05.com.archie:client
Created mapped LUN 0.
```

Configure Authentication

- ⊕ We'll need to configure the userid and password when connecting to the LUN from the client. *This step is optional.*
- ⊕ Choose any name and password for userid and password

set auth userid=username
set auth password=password

```
/iscsi/iqn.20...ent/tpg1/acls> cd iqn.2019-05.com.archie:client/
/iscsi/iqn.20...archie:client> set auth userid=username
Parameter userid is now 'username'.
/iscsi/iqn.20...archie:client> set auth password=password
Parameter password is now 'password'.
/iscsi/iqn.20...archie:client>
```

Exit and Save the Configuration

```
/iscsi/iqn.20...archie:client> exit
Global pref auto_save_on_exit=true
Configuration saved to /etc/target/saveconfig.json
```

Open Firewall for iSCSI and enable the iSCSI service

45HomeLab
A Division of 45Drives

- In the Houston UI, navigate to Networking > Firewall > Edit Zones and Rules
- Add "iscsi-target" to the services
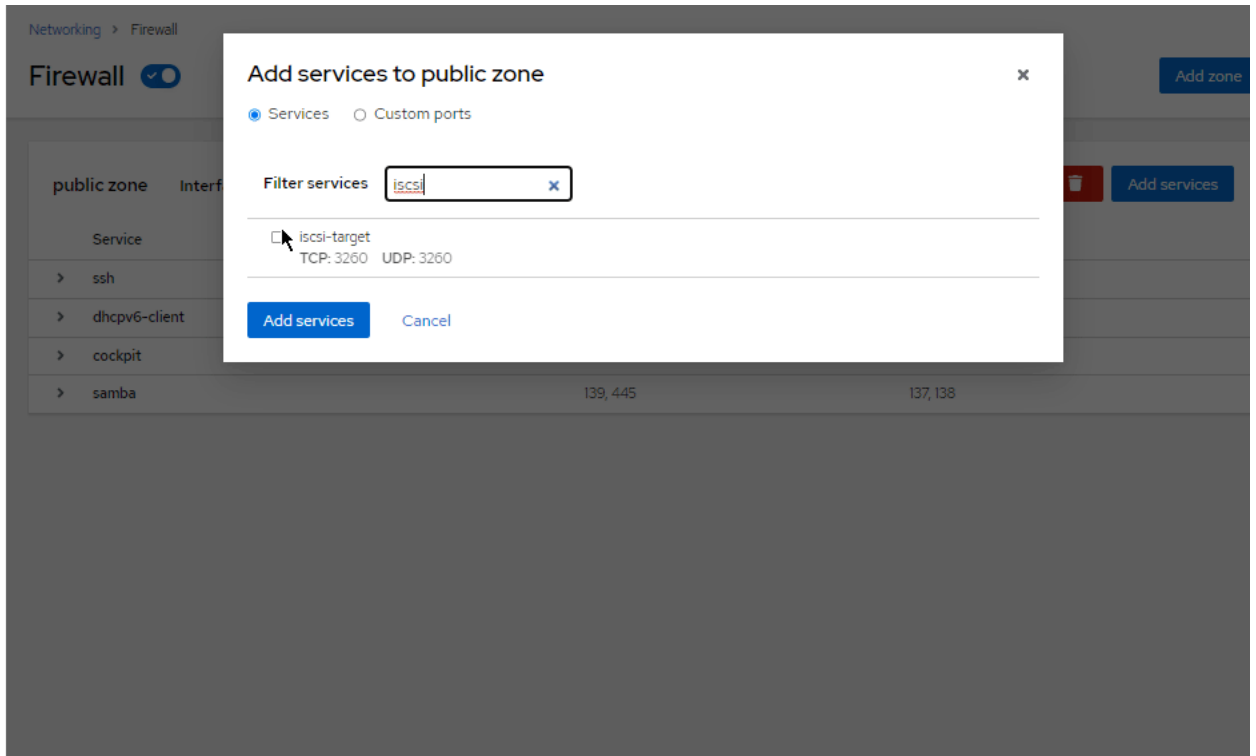
- In the Houston service tab, search for "target" and enable the Targetcli daemon

45HomeLab
A Division of 45Drives
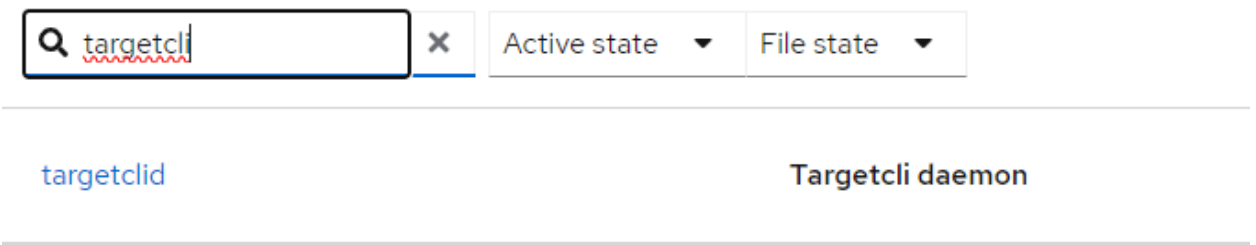
Targetcli daemon

Start and enable

Status	ⓘ Disabled
Path	/lib/systemd/system/targetclid.service
Triggered by	targetclid.socket

❯ Show more relationships

Verification /Connecting to the LUN

- ⊕ On a Windows client, open "iSCSI Initiator"
- ⊕ Enter the target IP address into the 'target' field

45HomeLab
A Division of 45Drives

⊕ Click 'quick connect'

45HomeLab
A Division of 45Drives

Quick Connect

Targets that are available for connection at the IP address or DNS name that you provided are listed below. If multiple targets are available, you need to connect to each target individually.

Connections made here will be added to the list of Favorite Targets and an attempt to restore them will be made every time this computer restarts.

Discovered targets

Name	Status
iqn.2021-11.fourfive.lab:test.environment	Connected

Progress report

Immediate Login Succeeded.Persistent Login failed, target will not be available after system reboot

Connect   Done

In Disk Management, there should now be a new disk that is unallocated and ready for formatting.

45HomeLab
A Division of 45Drives
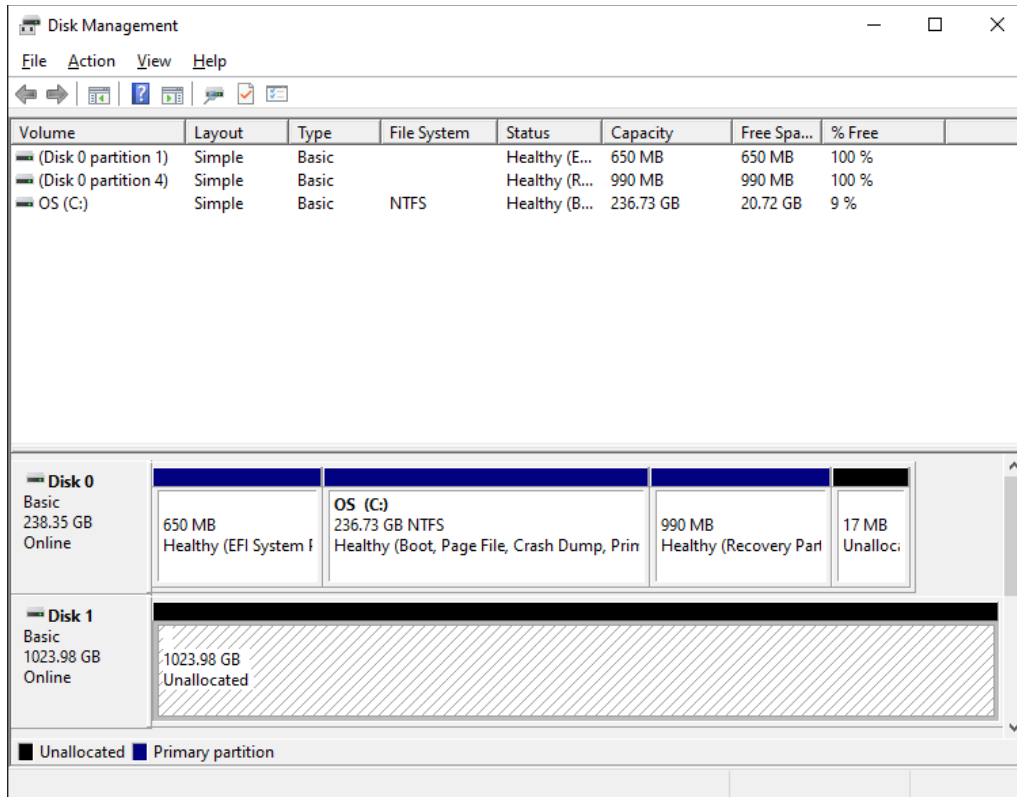
AUTOMATED ZFS REPLICATION/SNAPSHOTS IN HOUSTON UI

- ⊕ Make sure to install mbuffer and remove the old auto snapshot service on all Storinators you wish to use snapshots to use with znapzend.

> dnf install mbuffer

> systemctl restart znapzend

- ⊕ In the Houston UI, go to the ZFS tab.

45HomeLab
A Division of 45Drives
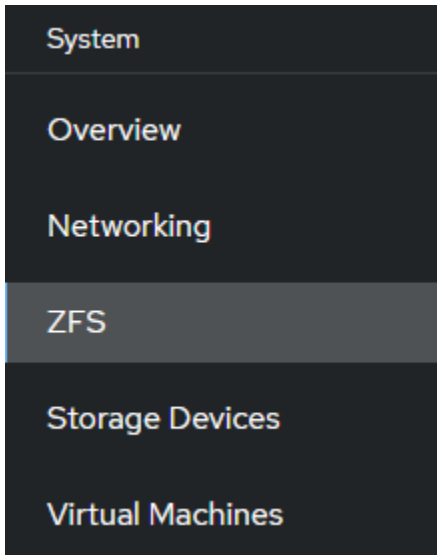
- Select the Dataset you wish to backup. Click the drop down on the left and select "Configure Replication Task

tank	52.13 TiB	70.52 MiB	127.83 KiB	0 B	128 KiB	LZ4	Off	Off	Yes	No	⋮
tank/backup	52.13 TiB	170.44 KiB	0 B	0 B	128 KiB	LZ4	Off	Off	Yes	No	⋮
tank/nfs	52.13 TiB	170.44 KiB	0 B	0 B	128 KiB	LZ4	Off	Off	Yes		Configure File System
tank/samba	52.13 TiB	291.16 KiB	99.42 KiB	0 B	128 KiB	LZ4	Off	Off	Yes		Edit Permissions
tank/samba-test	52.13 TiB	170.44 KiB	0 B	0 B	128 KiB	LZ4	Off	Off	Yes		Rename File System
tank/samba-windows-acl	52.13 TiB	170.44 KiB	0 B	0 B	128 KiB	LZ4	Off	Off	Yes		Unmount File System
											Destroy File System
											Configure Replication Task
											Enable Samba Share

- The screenshot below details a task that takes a snapshot once daily, and retains the snapshots for 1 month. This can be customized to your use case.
- Also, you can add multiple rules by clicking the +, for example setup below is for every hour for 7 days, every 4 hours for 30 days, and every 90 days for a year.

45HomeLab
A Division of 45Drives

Configure Replication Task

Recursive	☑
Destination	☐
mBuffer Size	1
mBuffer Unit	G

Source Plans ⊞

Retention Time	7
Retention Time Unit	Day
Interval Time	1
Interval Time Unit	Hour

Retention Time	30
Retention Time Unit	Day
Interval Time	4
Interval Time Unit	Hour

Retention Time	1
Retention Time Unit	Year
Interval Time	90
Interval Time Unit	Day

Cancel   Configure

Points to note

Make sure to restart the 'znapzend' service after any change has been made to snapshot tasks

znapzend   ZnapZend – ZFS Backup System

⊕ To ensure the snapshots are being created you can go to the Snapshots section of the ZFS to see all snapshots that were created.

45HomeLab
A Division of 45Drives

SOFTWARE

45HomeLab comes with Rocky as the default OS and Houston UI for the server management. Apart from that you can install other software stacks as well as per your preference. We have included some of them for your reference.

PORTAINER

With the help of Portainer, you can easily interact with containerized programs, monitor your Docker installation, and set up new stacks. To centralize your container administration around a single application, a single Portainer instance may link to numerous Docker hosts.

Make sure docker is installed before you proceed. If not installed you can refer
https://docs.rockylinux.org/gemstones/docker/

- ⊕ Create the docker volume
  ```
  docker volume create portainer_data
  ```
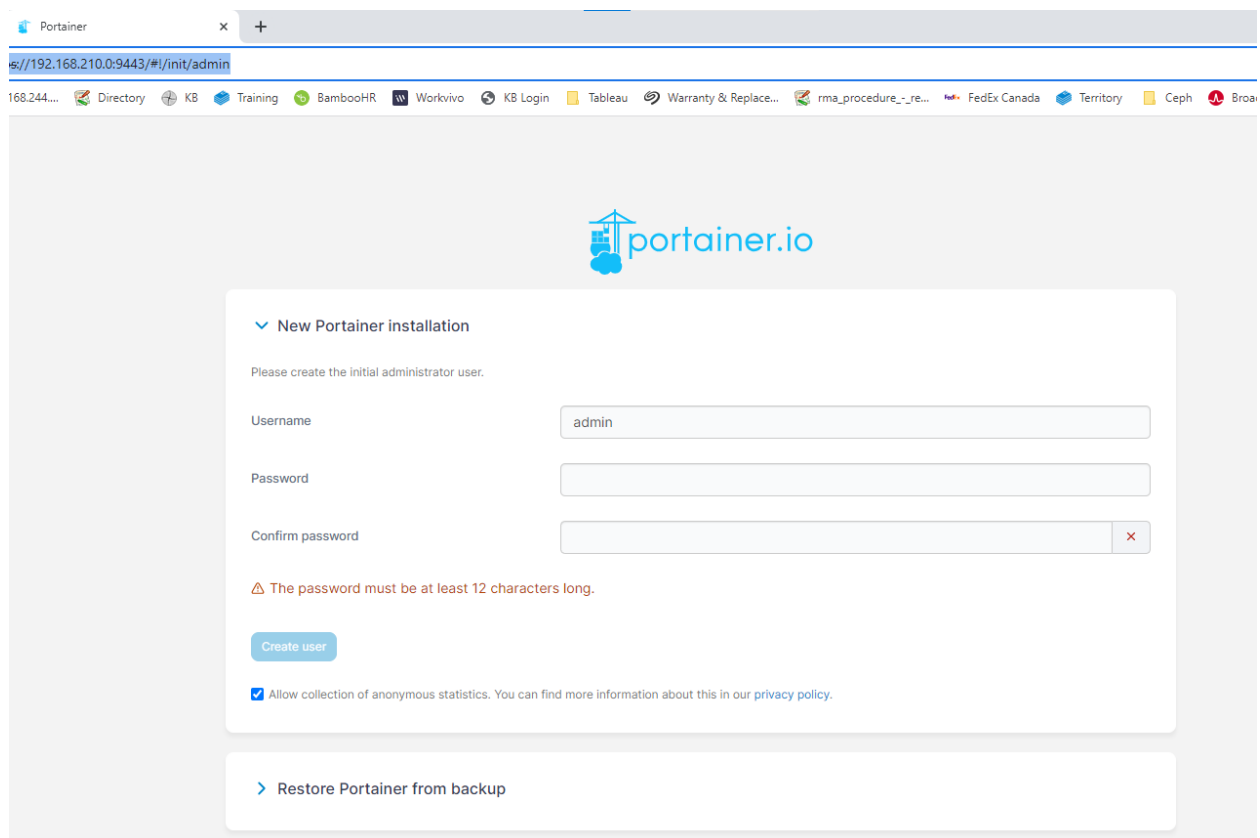
```
[root@storinator ~]# docker volume create portainer_data
Emulate Docker CLI using podman. Create /etc/containers/nodocker to quiet msg.
portainer_data
[root@storinator ~]#
```

- ⊕ docker run -d -p 8000:8000 -p 9443:9443 --name portainer --
  restart=always -v /var/run/docker.sock:/var/run/docker.sock -v
  portainer_data:/data portainer/portainer-ce:latest

45HomeLab
A Division of 45Drives

```
[root@HL15 ~]# docker run -d -p 8000:8000 -p 9443:9443 --name portainer --restart=always -v /var/run/docker.sock:/var/run/docker.sock -v portainer_data:/data portainer/portainer-ce:latest
Emulate Docker CLI using podman. Create /etc/containers/nodocker to quiet msg.
? Please select an image:
    registry.access.redhat.com/portainer/portainer-ce:latest
    registry.redhat.io/portainer/portainer-ce:latest
  ▸ docker.io/portainer/portainer-ce:latest
```

```
[root@HL15 run]# docker run -d -p 8000:8000 -p 9443:9443 --name portainer --restart=always -v /var/run/docker.sock:/var/run/docker.sock -v portainer_data:/data portainer/portainer-ce:latest
Unable to find image 'portainer/portainer-ce:latest' locally
latest: Pulling from portainer/portainer-ce
795a208431d7: Pull complete
4f272ca3dde3: Pull complete
5171176db7f2: Pull complete
52e9438966a5: Pull complete
43d4775415ac: Pull complete
c1cad9f5200f: Pull complete
27d6dca9cab4: Pull complete
231d7e50ef35: Pull complete
589f2af34593: Pull complete
5fc2ddaa6f07: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:f29cbc7b26ebd701b1fe92b4df42edea350e871372a6296a1fa16ba999481fb2
Status: Downloaded newer image for portainer/portainer-ce:latest
f786478bed70e71711e9c1540c15a94980cc4af401bd8ca9f1be857dde24f156
[root@HL15 run]#
```

⊕ The portainer will be on port 9443 so access it using https://serverip:9443

⊕ Make sure the port is open in firewall.

⊕ Create the admin user

⊕ You can click on getting started.

⊕ You can check more on the documentation at https://docs.portainer.io/

45HomeLab
A Division of 45Drives

⊕ You should have portainer local environment like below

Now you have your portainer ready to deploy other software stacks.

45HomeLab
A Division of 45Drives

NEXTCLOUD AND NPM ON PORTAINER

- ⊕ You will need portainer to be installed and configured first before installing NextCloud.
- ⊕ In portainer navigate to "Images" and select Build a New Image:

- ⊕ Name the new image "45cloud" and paste the contents of
  https://github.com/45Drives/scripts/blob/main/nextcloud/Dockerfile

45HomeLab
A Division of 45Drives

Then click "Build the image", this will start the build process, wait until this completes.

Then navigate to the "Stacks" tab, and click "Add Stack"

Name the stack nextcloud-nginx and paste the following into the text box:

If you already have Nginx Proxy Manager installed remove those NPM sections from the text below when pasting into the web editor.

45HomeLab
A Division of 45Drives

- Paste the below contents on the web editor. You can copy it from the link below so that .yml file formatting is not lost https://scripts.45homelab.com/docker-compose/nextcloud-npm/docker-compose.yml

45HomeLab
A Division of 45Drives

```yaml
version: "3"
volumes:
  nextcloud-data:
  nextcloud-db:
  npm-data:
  npm-ssl:
  npm-db:
networks:
  frontend:
  backend:
services:
  nextcloud-app:
    image: 45cloud:latest
    restart: always
    volumes:
      - nextcloud-data:/var/www/html
    environment:
      - MYSQL_PASSWORD=thisisjustatest
      - MYSQL_DATABASE=nextcloud
      - MYSQL_USER=nextcloud
      - MYSQL_HOST=nextcloud-db
      - PHP_UPLOAD_LIMIT=200G
    networks:
      - frontend
      - backend
  nextcloud-db:
    image: mariadb:10.5
    restart: always
    command: --transaction-isolation=READ-COMMITTED --binlog-format=ROW
    volumes:
      - nextcloud-db:/var/lib/mysql
    environment:
      - MYSQL_ROOT_PASSWORD=thisisjustatest
      - MYSQL_PASSWORD=thisisjustatest
      - MYSQL_DATABASE=nextcloud
      - MYSQL_USER=nextcloud
      - OVERWRITEPROTOCOL=https
    networks:
      - backend
  npm-app:
    image: jc21/nginx-proxy-manager:2.9.19
    restart: always
    ports:
      - "80:80"
      - "81:81"
      - "443:443"
      - "8900:8900"
      - "32400:32400"
      - "2283:2283"
    environment:
      - DB_MYSQL_HOST=npm-db
      - DB_MYSQL_PORT=3306
      - DB_MYSQL_USER=npm
      - DB_MYSQL_PASSWORD=thisisjustatest
      - DB_MYSQL_NAME=npm
    volumes:
      - npm-data:/data
      - npm-ssl:/etc/letsencrypt
    networks:
      - frontend
      - backend
```

```
npm-db:
    image: jc21/mariadb-aria:latest
    restart: always
    environment:
      - MYSQL_ROOT_PASSWORD=thisisjustatest
      - MYSQL_DATABASE=npm
      - MYSQL_USER=npm
      - MYSQL_PASSWORD=thisisjustatest
    volumes:
      - npm-db:/var/lib/mysql
    networks:
      - backend
```

- ⊕ Click Deploy Stack and wait for it to finish
- ⊕ At this point all configuration steps are identical to:
- ⊕ https://knowledgebase.45drives.com/kb/kb450427-nextcloud-with-nginx-proxy-manager-on-ubuntu-20-04/
- ⊕ Nginx Proxy Managers webUI will be available at port :81 with admin@example.com and changeme credentials.
- ⊕ When creating your proxy host use the name of the container found in the containers tab of portainer.

45HomeLab
A Division of 45Drives

INSTALLING NGINX PROXY MANAGER(NPM) ON PORTAINER

If you need NextCloud to be installed as well, please follow the steps on the NextCloud Section that will handle the deployment of both NextCloud and Nginx proxy manager.

If you have already installed NextCloud using the steps above. You can ignore this separate installation as Nginx is already installed with NextCloud install above.

If not, below would be the steps to install NPM to set up proxy for custom ports for various applications.

- ⊕ Go to your portainer home and select your environment.
- ⊕ Click on stacks and add a new stack.
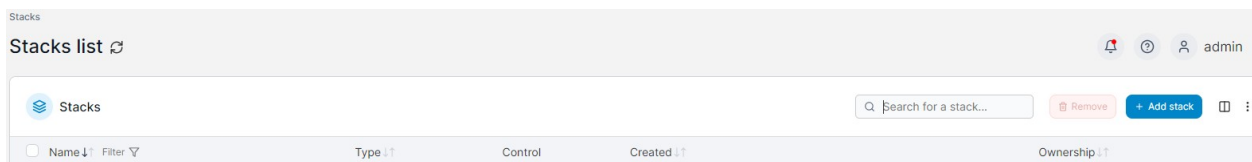
- ⊕ Give a name for your stack. For example, we can give npm.
- ⊕ Paste the below contents on the web editor. You can copy it from the link below so that .yml file formatting is not lost.

  https://scripts.45homelab.com/docker-compose/nginx-proxy-manager/docker-compose.yml

45HomeLab
A Division of 45Drives

```
version: '3'
volumes:
  npm-data:
  npm-ssl:
  npm-db:
networks:
  frontend:
  backend:
services:
  npm-app:
    image: jc21/nginx-proxy-manager:2.9.19
    restart: always
    ports:
      - "80:80"
      - "81:81"
      - "443:443"
    environment:
      - DB_MYSQL_HOST=npm-db
      - DB_MYSQL_PORT=3306
      - DB_MYSQL_USER=npm
      - DB_MYSQL_PASSWORD=thisisjustatest
      - DB_MYSQL_NAME=npm
    volumes:
      - npm-data:/data
      - npm-ssl:/etc/letsencrypt
    networks:
      - frontend
      - backend
  npm-db:
    image: jc21/mariadb-aria:latest
    restart: always
    environment:
    - MYSQL_ROOT_PASSWORD=thisisjustatest
    - MYSQL_DATABASE=npm
    - MYSQL_USER=npm
    - MYSQL_PASSWORD=thisisjustatest
    volumes:
      - npm-db:/var/lib/mysql
    networks:
      - backend
```

⊕ After that click on deploy stack and wait for the deployment to be completed.

45HomeLab
A Division of 45Drives

Actions

Deploy the stack

⊕ Nginx Proxy Managers webUI will be available at port :81 http://serverip:81 with admin@example.com and changeme credentials.

NGINX
PROXY MANAGER
v2.9.19

Login to your account

Email address

Email address

Password

Password

Sign in

⊕ Once in set the admin account email id and password

⊕ After that you can start creating your proxies.

45HomeLab
A Division of 45Drives

CONFIGURING PLEX PORTAINER

⊕ Go to portainer and create a new volume called plex_config

⊕ Then go to create a container name as you want example "plexmedia"
⊕ For the image give **linuxserver/plex:latest**

45HomeLab
A Division of 45Drives

Containers > Add container

Create container

Name

plexmedia

Image configuration

Registry

Docker Hub (anonymous)

Image*

docker.io   linuxserver/plex:latest

⊕ Advanced mode

Always pull the image ⑦

- ⊕ Under volumes click on map additional volume and create a volume with container path of /config to the volume you created in step 1

- ⊕ Click on map additional volume and create a second bind volume with container path to /media. This name can change to whatever and bind it to the host path that they want to store media such as a zpool, etc. (here we can used our path in the zpool where our storage is).

- ⊕ Under network select the nginx proxy frontend network

45HomeLab
A Division of 45Drives

⊕ Under ENV click on add environment variable and throw in PGID 1000 and PUID 1000. Here 1000 is the ID of the user that we are given ownership of the content, so add the ID accordingly.

⊕ Select a restart policy of unless stopped

⊕ deploy the container

⊕ You should be able to see that the container is deployed and running.

45HomeLab
A Division of 45Drives

⊕ Nginx Proxy Managers web UI will be available at port :81 with admin@example.com and 'change me' credentials.

⊕ When creating your proxy host use the name of the container found in the containers tab of portainer.

⊕ Create a proxy host with domain name IP:PORT or DNS name if they are port forwarding or have a hostname select scheme (http/https), forward hostname is the name of the container and forward port by default would be 32400. Then under advanced custom nginx config put in "listen 32400;"

⊕ Click on proxy hosts

⊕ Add proxy host

45HomeLab
A Division of 45Drives
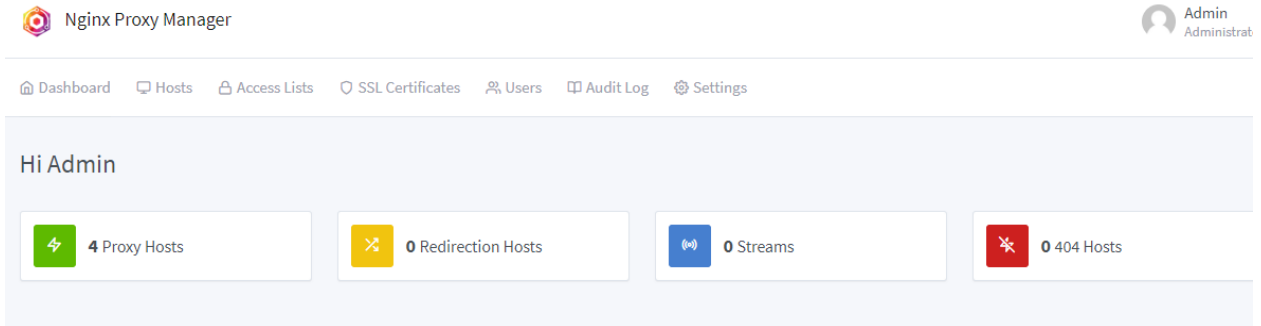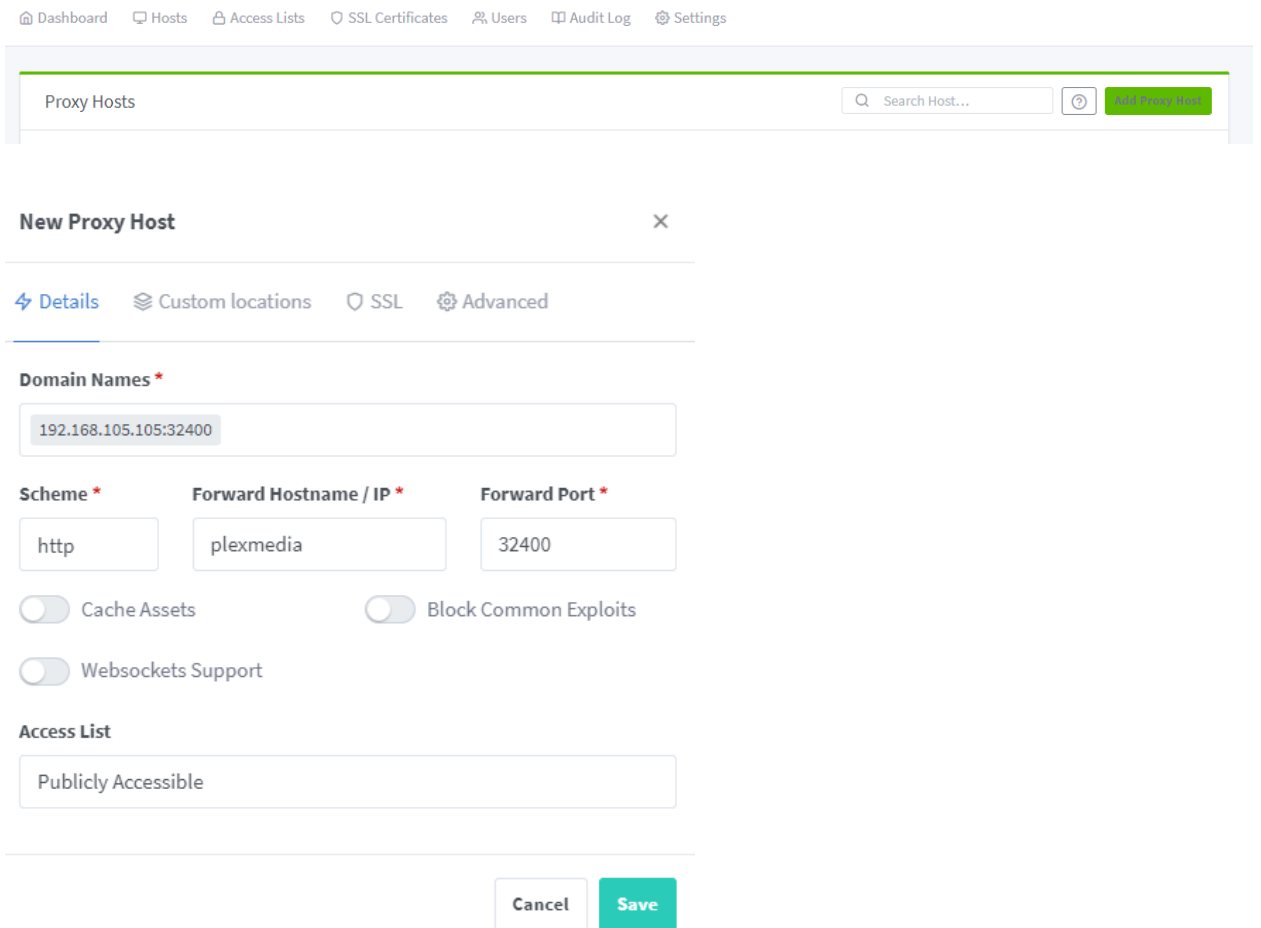
⊕ In the advanced give the below and save

Edit Proxy Host ✕

⚡ Details   ⊗ Custom locations   ○ SSL   ⚙ Advanced

These proxy details are available as nginx variables:

- $server Forward Hostname / IP
- $port Forward Port
- $forward_scheme Scheme

Custom Nginx Configuration

```
listen 32400;
```

⚠ Please note, that any add_header or set_header directives added here will not be
used by nginx. You will have to add a custom location '/' and add the header in the
custom config there.

Cancel   **Save**

⊕ Now you can access your plex using http://serverip:32400/web/index.html#!/
⊕ Make sure the port is open in firewall.
⊕ Sign in or create a plex account

45HomeLab
A Division of 45Drives

Plex Web

would like to sign in to your Plex account

!

This application is at **192.168.105.105** and is not hosted by Plex. Continue only if you recognize this server and wish to grant access.

G   **Continue with Google**

f   **Continue with Facebook**

     Continue with Apple

― or ―

Email or Username

Password   Forgot?

Sign In

Need an account? Press the Google, Facebook, or Apple buttons above, or **sign up with email**

45HomeLab
A Division of 45Drives

How Plex Works

Plex Media Server runs on the computer where you keep your media

Plex scans your media, automatically organizes it, and makes it beautiful

Play your media on any screen with your favorite Plex app

Got It!

⊕ After that you can set up the library as per your wish

45HomeLab
A Division of 45Drives
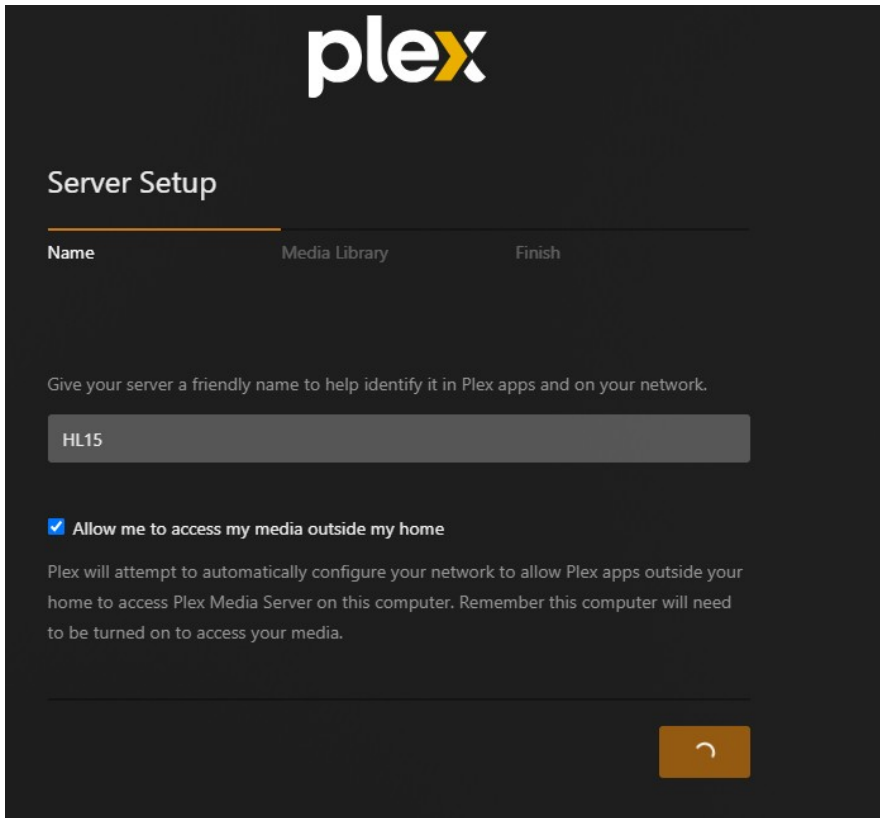
⊕ Click on add libraries
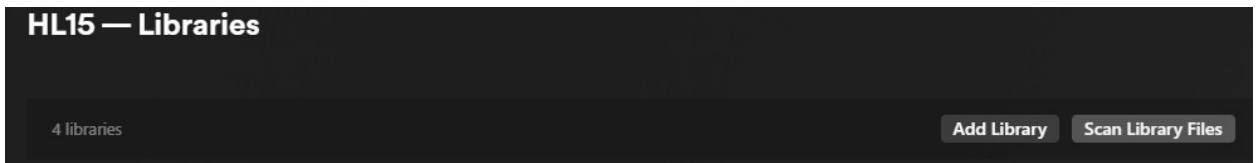
⊕ You can select the library type and then select the media folder that we had linked to our storage in HL4

45HomeLab
A Division of 45Drives
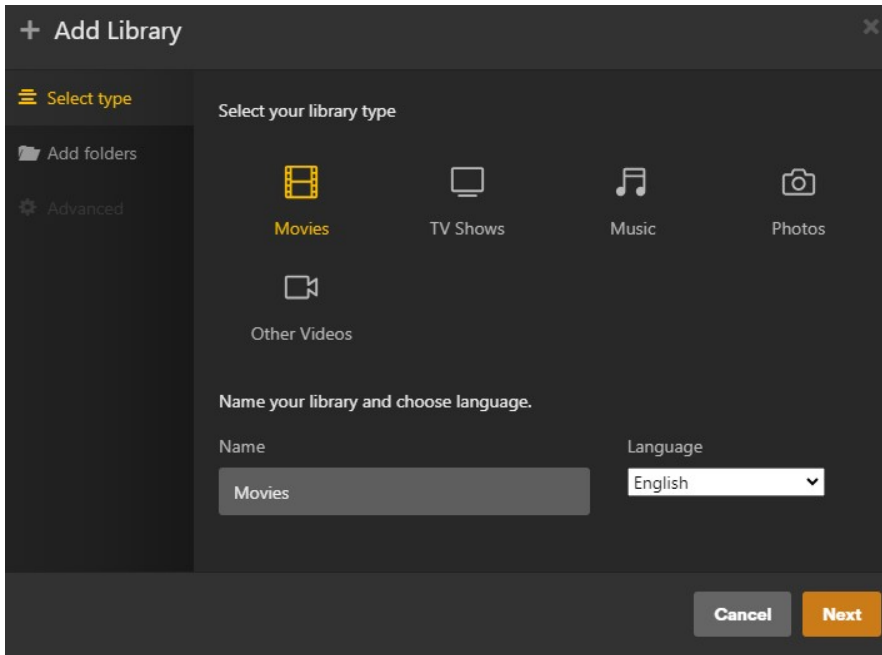
⊕ After that select the folder, we are selecting media as we had given that during our container creation.

45HomeLab
A Division of 45Drives
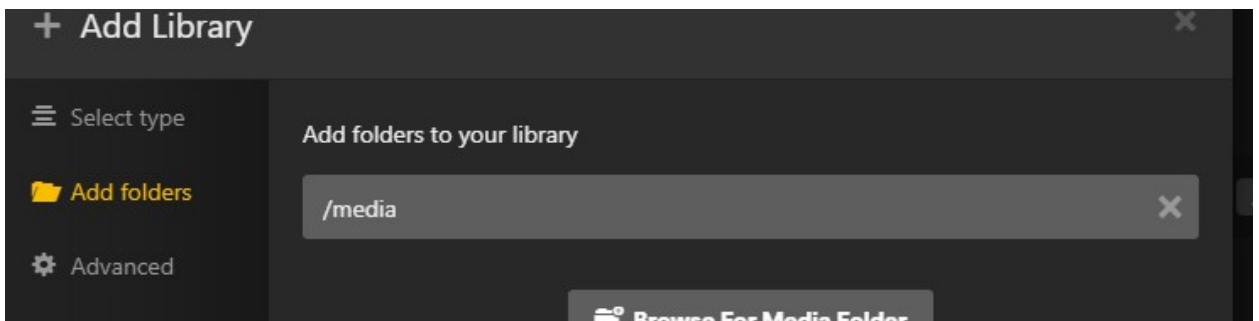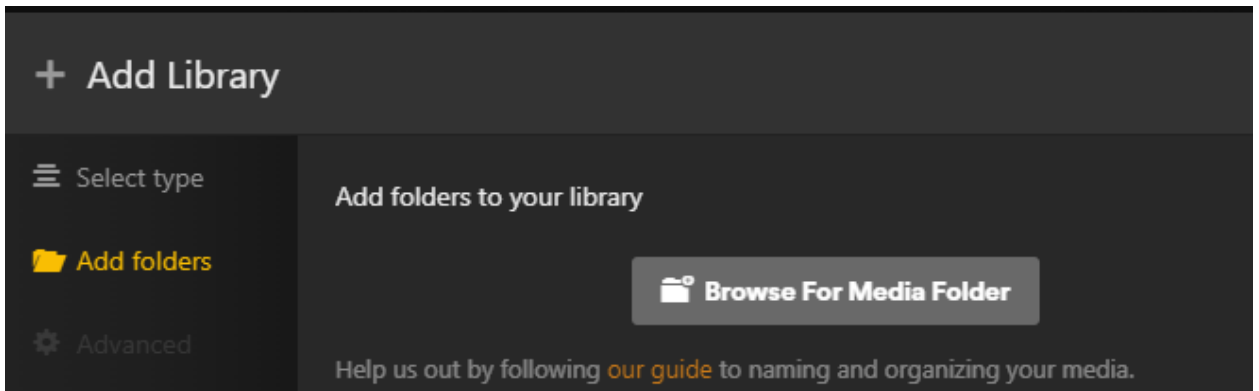
- Then you can scan the library for the existing files
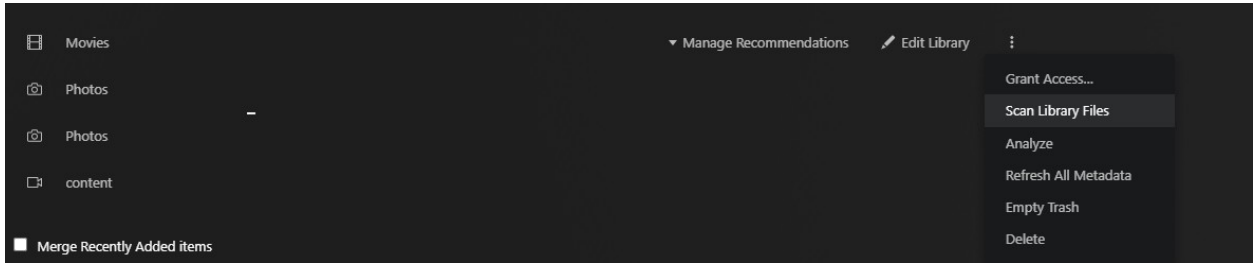
- You will be able to see the videos that are there in your server in the dataset or path you had given.

You can check https://support.plex.tv/articles/200264746-quick-start-step-by-step-guides/ for more guides and info.

45HomeLab
A Division of 45Drives
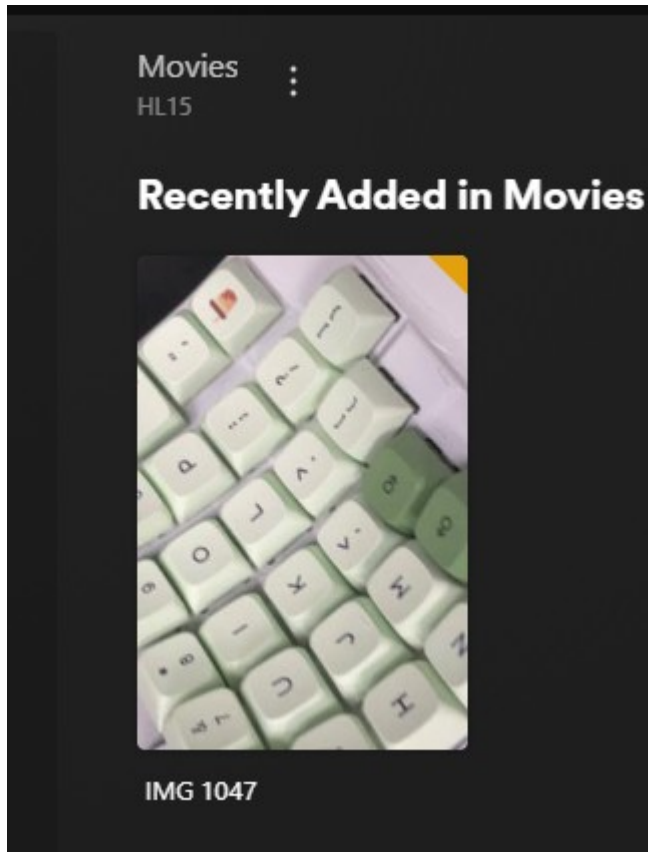
IMMICH- self-hosted backup solution for photos and videos

- Go to "Stacks" in the left sidebar.
- Click on "Add stack".
- Give the stack a name (i.e. Immich), and select "Web Editor" as the build method.
- Copy the content of the docker-compose.yml file (github repo - https://github.com/immich-app/immich/releases/latest/download/docker-compose.yml)

```
version: "3.8"


services:

  immich-server:

    container_name: immich_server

    image: ghcr.io/immich-app/immich-server:${IMMICH_VERSION:-release}

    command: ["start.sh", "immich"]

    volumes:

      - ${UPLOAD_LOCATION}:/usr/src/app/upload

      - /etc/timezone:/etc/timezone:ro

      - /etc/localtime:/etc/localtime:ro

    env_file:
```

45HomeLab
A Division of 45Drives

```yaml
      - stack.env
    depends_on:
      - redis
      - database
      - typesense
    restart: always


immich-microservices:
    container_name: immich_microservices
    image: ghcr.io/immich-app/immich-server:${IMMICH_VERSION:-release}
    # extends:
    #    file: hwaccel.yml
    #    service: hwaccel
    command: ["start.sh", "microservices"]
    volumes:
      - ${UPLOAD_LOCATION}:/usr/src/app/upload
      - /etc/timezone:/etc/timezone:ro
      - /etc/localtime:/etc/localtime:ro
    env_file:
      - stack.env
    depends_on:
      - redis
      - database
      - typesense
    restart: always


immich-machine-learning:
    container_name: immich_machine_learning
    image: ghcr.io/immich-app/immich-machine-learning:${IMMICH_VERSION:-release}
    volumes:
      - model-cache:/cache
    env_file:
      - stack.env
```

45HomeLab
A Division of 45Drives

```
    restart: always


  immich-web:

    container_name: immich_web

    image: ghcr.io/immich-app/immich-web:${IMMICH_VERSION:-release}

    env_file:

      - stack.env

    restart: always


  typesense:

    container_name: immich_typesense

    image:
typesense/typesense:0.24.1@sha256:9bcff2b829f12074426ca044b56160ca9d777a0c488303469143dd9f8259
d4dd

    environment:

      - TYPESENSE_API_KEY=${TYPESENSE_API_KEY}

      - TYPESENSE_DATA_DIR=/data

      # remove this to get debug messages

      - GLOG_minloglevel=1

    volumes:

      - tsdata:/data

    restart: always


  redis:

    container_name: immich_redis

    image: redis:6.2-
alpine@sha256:70a7a5b641117670beae0d80658430853896b5ef269ccf00d1827427e3263fa3

    restart: always


  database:

    container_name: immich_postgres

    image: postgres:14-
alpine@sha256:28407a9961e76f2d285dc6991e8e48893503cc3836a4755bbc2d40bcc272a441

    env_file:

      - stack.env
```

45HomeLab
A Division of 45Drives

```
    environment:

      POSTGRES_PASSWORD: ${DB_PASSWORD}

      POSTGRES_USER: ${DB_USERNAME}

      POSTGRES_DB: ${DB_DATABASE_NAME}

    volumes:

      - pgdata:/var/lib/postgresql/data

    restart: always


  immich-proxy:

    container_name: immich_proxy

    image: ghcr.io/immich-app/immich-proxy:${IMMICH_VERSION:-release}

    environment:

      # Make sure these values get passed through from the env file

      - IMMICH_SERVER_URL

      - IMMICH_WEB_URL

    ports:

      - 2283:8080

    depends_on:

      - immich-server

      - immich-web

    restart: always


volumes:

  pgdata:

  model-cache:

  tsdata:
```

⊕ Click on "Advanced Mode" in the Environment Variables section.

Environment variables

These values will be used as substitutions in the stack file. To reference the .env file in your compose file, use 'stack.env'

✎ Advanced mode
ⓘ Switch to advanced mode to copy & paste multiple variables

45HomeLab
A Division of 45Drives

- Copy the content of the example.env from below and paste into the editor. (github repo - https://github.com/immich-app/immich/releases/latest/download/example.env)

```
# You can find documentation for all the supported env variables at
https://immich.app/docs/install/environment-variables

# The location where your uploaded files are stored

UPLOAD_LOCATION=./library

# The Immich version to use. You can pin this to a specific version like "v1.71.0"

IMMICH_VERSION=release

# Connection secrets for postgres and typesense. You should change these to random passwords

TYPESENSE_API_KEY=some-random-text

DB_PASSWORD=postgres

# The values below this line do not need to be changed

################################################################################

DB_HOSTNAME=immich_postgres

DB_USERNAME=postgres

DB_DATABASE_NAME=immich

REDIS_HOSTNAME=immich_redis
```

- Switch back to "Simple Mode".
- Populate custom database information if necessary.
- Populate UPLOAD_LOCATION with your preferred location for storing backup assets.
- Click on "Deploy the stack".

Actions

Deploy the stack

- Immich will be accessible via http://serverip:2283/
- Make sure the port is open in firewall.

Post installation you can you can use the guide below to set up
https://documentation.immich.app/docs/install/post-install

45HomeLab
A Division of 45Drives

Immich user guide

Uploading Pictures

- ⊕ For First-Time Users

- ⊕ On the main home page, you'll see a noticeable box labeled "CLICK TO UPLOAD YOUR FIRST PHOTO." Click on it to begin the upload process.

- ⊕ For Users with Existing Uploads:

- ⊕ If you already have pictures uploaded, simply click the "Upload" option located in the top-right corner of the screen. This will allow you to select images, videos, or other media from your local computer. **YOU MUST BE IN THE PHOTOS TAB FOR THE UPLOAD BUTTON TO APPEAR**

Explore Tab

- ⊕ Within the Explore tab, you have access to a range of tools that make searching and viewing your photos a breeze. Leveraging the capabilities of ChatGPT, facial recognition, and geolocation technology, finding your pictures has never been more convenient.

- ⊕ You can also narrow down your search by categories such as Favorites, Recently Added, Videos, Motion Pictures, and Panorama Photos, making it

Map Tab

- ⊕ In the Map tab, you can harness the geolocation data of your photos to visualize them on a world map. This feature allows you to see exactly where each of your photos was taken, providing a unique and interactive way to explore your memories.

- ⊕ Inside Map Settings, you have the capability to adjust the map's visual mode (dark or light), decide whether to display only your favorite photos on the map, and specify a date range for the displayed photos. These options enable you to tailor your map experience to your preferences.

Sharing Tab

- ⊕ Within the Sharing tab, you have the capability to create shared albums with other IMMICH members or share album links with individuals who don't have IMMICH accounts. This feature simplifies the process of sharing your albums with others, regardless if they use IMMICH or not.

45HomeLab
A Division of 45Drives

Library

Favorites Tab

- In the Favorites tab, you can conveniently locate all the photos you've marked as your favorites. This tab serves as a dedicated space where your cherished photos are readily accessible.

Albums Tab

- In the Albums tab, you can seamlessly create and organize photos into distinct albums. This feature allows you to categorize your photos and easily access them based on the specific albums they belong to.

Archive Tab

- In the Albums tab serves as a storage space for your photos, keeping them separate from the main photo view. This feature is useful for safely tucking away photos that you may not want readily accessible or visible in the main photos tab or to other users.

Mobile App

Settings

In the Albums tab serves as a storage space for your photos, keeping them separate from the main photo view. This feature is useful for safely tucking away photos that you may not want readily accessible or visible in the main photos tab

Administration

When logged into IMMICH as an administrative account in the WebUI, you will notice an "Administration" button located in the top-left corner. Clicking this button will navigate you to the administration window, where you can access and configure all the administrative settings for IMMICH.

Users

Within the "User" tab, you have the ability to both view existing users and manage them by adding or removing users as needed. To make changes to a user's settings, simply locate the user and click on the blue pencil icon situated to the right of their entry. This enables you to edit various aspects of their profile, such as their email address, name, storage label, external path, and even reset their password.

Furthermore, each user account has its dedicated personal settings accessible by clicking on your name or icon located in the top right corner. Within this section, you can perform various actions, including

45HomeLab
A Division of 45Drives

managing your account details, handling API keys, authorizing devices, managing memories, overseeing password settings for your account, and configuring sharing preferences.

Jobs

When in the 'Jobs' tab, you can conveniently monitor all currently active tasks, such as generating thumbnails, extracting metadata, sidecar metadata, tagging objects, encoding clips, recognizing faces, transcoding videos, and handling storage template migration jobs, which can be initiated from the bottom of the page. Furthermore, you have the option to control the concurrency settings by navigating to the top right corner of the page, where you will find a blue box labeled 'Manage Concurrency".

Settings

Inside the settings tab, we have a number of options to choose from. Starting with Job Settings - These settings are the same settings that are found in the jobs tab in the "manage concurrency" button.

- Machine Learning – In this section, you can configure Machine Learning settings, including the option to enable or disable features such as Image Tagging, Smart Search, and Facial Recognition.
- Map Settings – n this section, you have the flexibility to enable or disable the map features, and you can also modify the tile URL if needed.
- OAuth Authentication – In this section, you have the ability to control the login settings using OAuth. You can manage various parameters such as the Issuer URL, Client ID, Client Secret, Scope, Storage Label Claim, Button Text, as well as options like Auto Register, Auto Launch, and Mobile Redirect URL override
- Password Authentication – In this section, you can toggle the option to log in using a username and password, allowing you to either enable or disable this feature.
- Storage Template – In this section, you can customize how your images and videos are saved, as well as configure the desired file structure.
- Thumbnail Settings – Here, you have the option to modify the resolution of both small and large thumbnails, as well as fine-tune the quality percentage. Additionally, there's a setting available for those who prefer the Wide Gamut display.
- Video Transcoding Settings – This tab is for the bit more advanced users. Here we manage the resolutions and encoding information of the video files. We can find things such as:
  - Constant Rate Factor (-crf)
  - Present (-present)
  - Audio Codec
  - Video Codec
  - Target Resolution
  - Max Bitrate
  - Threads
  - Transcode Policy
  - Tone-Mapping

45HomeLab
A Division of 45Drives

- Two-Pass Encoding

This is also two subdirectories inside the Video Transcoding settings being

- Hardware Acceleration (Experimental)
- And Advanced in side advanced are settings most users will not need to change

Server Status

In the Server Status tab, you can access information like the total number of photos and videos stored, as well as the amount of storage used. Additionally, you can view detailed user usage data, which provides insights into the number of photo and video uploads for each individual user.

You can also check the server status, version, and storage usage from any page or tab by looking in the bottom left corner of the WebUI interface.

CLI Commands

You can find the CLI and bulk upload commands / guides here

https://immich.app/docs/features/bulk-upload

45HomeLab
A Division of 45Drives

HOME ASSISTANT

An Open-source home automation that puts local control and privacy first. Powered by a worldwide community of tinkerers and DIY enthusiasts.

Method 1- Portainer deployment

This is the bare minimum to get the container up and running, to access the Home Assistant web UI.

1. Go to your portainer UI —> stacks —> "Add Stack" —> Web editor —> Paste your docker-compose.yml

```
version: "3.9"
services:
  homeassistant:
    image: homeassistant/home-assistant:stable
    container_name: homeassistant
    network_mode: host
    environment:
      - TZ=America/Halifax
    volumes:
    - /opt/docker/stack/home_assistant/config:/config
restart: unless-stopped
```

The only things that need to change in this file are the time zone (TZ) and the volume path.

45HomeLab
A Division of 45Drives

2. Click "Deploy the stack".
3. Navigate to http://IP:8123
4. Make sure the port is open in firewall.
5. Follow guided [Home Assistant Web UI setup](#).

Method 2- Deploying in Rocky Terminal

Step 1: Create a Docker Compose file

⊕ Create a directory for your Home Assistant configuration, and inside this directory, create a docker-compose.yml file

```
mkdir -p /opt/docker/stack/home_assistant
cd /opt/docker/stack/home_assistant
vim docker-compose.yml
```

⊕ In the docker-compose.yml file, paste the following content:

```
version: "3.9"
services:
  homeassistant:
    image: homeassistant/home-assistant:stable
    container_name: homeassistant
    network_mode: host
    environment:
      - TZ=America/Halifax #Replace with your timezone
    volumes:
      - /opt/docker/stack/home_assistant/config:/config
    restart: unless-stopped
```

⊕ Save the file and exit the editor.

Step 2: Start the Service

Run Docker Compose from the same directory as your docker-compose.yml file:

```
docker-compose up -d
```

45HomeLab
A Division of 45Drives

This command will pull the necessary image and start the container in the background.

Step 3: Accessing Home Assistant

Since you're using network_mode: host , you can access the Home Assistant UI by navigating to http://<host-ip>:8123 or http://localhost:8123 if you're on the host machine, replacing <host-ip> with the actual IP address of your host.

Additional Notes:

- Ensure that no other services are using port 8123 on your host.
- If you face any issues accessing Home Assistant, consider checking firewall rules or network policies that might be blocking the port.

WIREGUARD- fast, modern, and secure VPN tunnel

- Go to "Stacks" in the left sidebar.
- Click on "Add stack".
- Give the stack a name (i.e. wireguard), and select "Web Editor" as the build method.
- Copy the content of the docker-compose.yml file
- You will need to set a secure password in the section in the .yml (WGUI_PASSWORD) file if you are planning to do port forwarding.

```
version: "3"
services:
  wireguard:
    image: linuxserver/wireguard:latest
    container_name: wireguard
    cap_add:
```

45HomeLab
A Division of 45Drives

```
    - NET_ADMIN
  volumes:
    - ./config:/config
  ports:
    - "5000:5000"
    - "51820:51820/udp"

wireguard-ui:
  image: ngoduykhanh/wireguard-ui:latest
  container_name: wireguard-ui
  depends_on:
    - wireguard
  cap_add:
    - NET_ADMIN
  network_mode: service:wireguard
  environment:
    - SENDGRID_API_KEY
    - EMAIL_FROM_ADDRESS
    - EMAIL_FROM_NAME
    - SESSION_SECRET
    - WGUI_USERNAME=admin
    - WGUI_PASSWORD=kGMrU6S7(+`Ah93ENLK><8
    - WG_CONF_TEMPLATE
    - WGUI_MANAGE_START=true
    - WGUI_MANAGE_RESTART=true
  logging:
    driver: json-file
    options:
      max-size: 50m
  volumes:
    - ./db:/app/db
    - ./config:/etc/wireguard
```

⊕ Click on "Deploy the stack".
⊕ If you want to do port forwarding after that you could follow that steps as well.

45HomeLab
A Division of 45Drives

FRIGATE- open-source NVR built around real-time AI

FRIGATE

Environment:

- ⊕ rocky 8
- ⊕ docker files in default location /var/lib/docker
- ⊕ Example containers location & structure

```
opt
├── docker
│   └── stack
│       ├── frigate
│       │   ├── config
│       │   │   └── config.yml
│       │   └── docker-compose.yml
```

docker-compose.yml

```
version: "3.9"
services:
  frigate:
    container_name: frigate
    image: ghcr.io/blakeblackshear/frigate:stable
    privileged: true # This may not be necessary for all setups
    restart: unless-stopped
    shm_size: "64mb" # Update for your cameras based on calculation above
    ports:
      - "5000:5000" # WebUI
      - "8554:8554" # RTSP feeds
      - "8555:8555/tcp" # WebRTC over TCP
      - "8555:8555/udp" # WebRTC over UDP
    environment:
      FRIGATE_RTSP_PASSWORD: "password"
    volumes:
      - /etc/localtime:/etc/localtime:ro
```

45HomeLab

```
      - /opt/docker/stack/frigate/config:/config # Mounting config directory
      - frigate_media:/media/frigate
      - type: tmpfs
```

Note: Ensure to use the image in the docker-compose.yml searching will often provide results for the depreciateds images.

Explanation of docker-compose.yml

1. **version: "3.9"**

Defines the version of Docker Compose file syntax. 3.9 is a specific version of the

Docker Compose file format.

2. **services:**

Defines the services (containers) to be created.

2.1 **frigate:**

Defines the name of the service (container) to be created.

2.2 **container_name: frigate**

Specifies the name of the container that will be created.

2.3 **image: ghcr.io/blakeblackshear/frigate:stable**

Specifies the Docker image to be used for this service, pointing to Frigate's stable release.

2.4 **privileged: true**

Allows the service to access the host's devices and possibly other privileged

functionalities.

2.5 **restart: unless-stopped**

Ensures the container will restart automatically unless explicitly stopped by the user.

2.6 **shm_size: "64mb"**

Allocates shared memory for the container, useful when your application has specific

memory requirements.

2.7 **ports:**

Maps the container's ports to the host's ports.

2.8 **environment:**

Sets environment variables within the container.

2.9 **volumes:**

45HomeLab
A Division of 45Drives

Mounts host paths or named volumes to paths inside the container.

- /etc/localtime:/etc/localtime:ro binds the host's timezone settings to the container in read-only mode.
- /opt/docker/stack/frigate/config:/config binds the configuration directory from the host to the container.
- frigate_media:/media/frigate creates a named volume for storing media.
- type: tmpfs creates a temporary filesystem in memory, providing high-speed read/write access.

3. **volumes**:

Defines named volumes used by services.

3.1 **frigate_media:**

This is a named volume declaration, which creates persistent storage independent of

the container lifecycle, used here for storing media.

Additional Notes:

- FRIGATE_RTSP_PASSWORD: "password" is setting an environment variable with the password for RTSP access. This will be found in your cameras configuration settings
- "5000:5000" makes the web UI of Frigate accessible on port 5000 of the host machine.
- "8554:8554" is for RTSP feeds, "8555:8555/tcp" and "8555:8555/udp" are for WebRTC over TCP and UDP, respectively.
- Each line in this file serves to configure the behavior, access, or setup of the Docker container or its interaction with the host system or other containers.

Config.yml

The config.yml file is used to specify configuration settings for Frigate. Frigate needs this file to know how to connect to your cameras, how to process the video streams, how to interact with other services, and many other settings.

Ensure you have this file created & populated with the correct path in your docker-compose.yml before running the container.

```
cameras:
front_door: # This is just an example name for your camera
  ffmpeg:
    inputs:
      - path: rtsp://your_camera_ip:554/your_stream_path
        roles:
            - detect
# Other camera settings go here
```

1. **mqtt**:

This denotes the configuration section related to MQTT, a lightweight messaging protocol typically used in IoT setups.

2. **enabled: False**

This line is configuring whether or not to enable MQTT. Here, it is set to False , meaning MQTT is disabled.

3. **cameras**:

This is the start of the section where you define the configurations for each of your cameras.

4. **name_of_your_camera:**

This is where you name your camera; it is just a placeholder, so you should replace it with a name that makes sense for your setup (e.g., front_door).

5-7. **ffmpeg: ... roles: - detect**

These lines are defining how Frigate should use FFmpeg to interact with the camera. FFmpeg is a multimedia framework used to handle video, audio, and other multimedia files and streams. Here, it is specified to use the camera for detection purposes.

8. **inputs:**

This denotes the start of a list of input streams from the camera that Frigate should use.

9. **path: rtsp://IP_OF_CAMERA:554/rtsp**

This line is specifying the RTSP (Real-Time Streaming Protocol) path to your camera stream. You should replace IP_OF_CAMERA with the actual IP of your camera, and the rest of the path may vary based on the camera model and manufacturer.

10. **roles**:

This specifies the roles that this input stream will be used for.

11. **detect**

This specifies that the input stream should be used for detection purposes.

12. **detect**:

This is the start of the section where you define the detection settings for this camera.

45HomeLab
A Division of 45Drives

Why is this file needed?

This file is crucial because it allows you to tailor Frigate to your specific needs and hardware. By providing this file, you are informing Frigate about your camera(s), their properties, locations, and how you want Frigate to process their streams, enabling Frigate to function correctly according to your use case. Without this configuration file, Frigate would not know how to interact with your camera(s) or how to process their streams.

TROUBLESHOOTING

HL4 WONT POWER ON

- ⊕ Try powering the HL4 using the switch
- ⊕ Check if you are seeing lights on the motherboard and the PSU
- ⊕ Check if the fans are spinning
- ⊕ If fans are spinning and you can see the lights on the motherboard and PSU it could be one of the hardware components
- ⊕ If you are not getting any video output it is most likely a motherboard issue or a CPU issue.
- ⊕ Replace the faulty component and you should be able to power up the HL4.

HOUSTON UI IS NOT ACCESSIBLE

If your Houston UI is not loading.

- ⊕ First thing would be to check if the server IP is accessible.
- ⊕ Open a terminal or command prompt and try pinging the server IP.

45HomeLab
A Division of 45Drives

- If you get a response then your IP / network settings is fine.
- Either ssh into the server or use the Graphical User Interface.
- Once you have connected to the server via terminal through SSH or Graphical User Interface, check if the cockpit service is running
- **systemctl status cockpit**

```
[root@homelabs ~]# systemctl status cockpit
● cockpit.service - Cockpit Web Service
   Loaded: loaded (/usr/lib/systemd/system/cockpit.service; static; vendor preset: disabled)
   Active: active (running) since Tue 2023-09-26 11:09:05 EDT; 2h 58min ago
     Docs: man:cockpit-ws(8)
  Process: 66010 ExecStartPre=/usr/libexec/cockpit-certificate-ensure (code=exited, status=0/SUCCESS)
 Main PID: 66013 (cockpit-tls)
    Tasks: 2 (limit: 408093)
   Memory: 5.4M
   CGroup: /system.slice/cockpit.service
           └─66013 /usr/libexec/cockpit-tls

Sep 26 11:09:05 homelabs systemd[1]: Starting Cockpit Web Service...
Sep 26 11:09:05 homelabs systemd[1]: Started Cockpit Web Service.
```

- If the status is inactive then restart the service using systemctl restart cockpit
- Your Houston interface should load fine after that.
- If the you were not getting ping response during our troubleshooting steps, then you need to check your networking
- You can use the NMTUI interface and check the interface that has the IP set and check if it up and active.
- If all good check if your device is in the correct network.

DRIVES ARE MISSING IN MY ZPOOL

- Try checking which are the drives in your pool that are missing
- You can either use the zpool status or check in the UI using ZFS tab
- Check the server logs using dmesg -T to see if it has any drives drop warning.
- Try re-seating the drive and see if it is getting picked up.
- If the drive is still not detected try swapping it with a drive in another slot and see if it is getting detected.
- If the drives is still not detected it is most likely a failed drive.
- But if the drive gets detected on a different slot. It could be a bad slot.

ZPOOL IS IN A DEGRADED STATE

- Try checking which are the drives in your pool that are missing
- You can either use the zpool status or check in the UI using ZFS tab
- Check the server logs using dmesg -T to see if it has any drives drop warning.

45HomeLab
A Division of 45Drives

- Check if it is an actual failed drive by running SMART diagnostics on the drive
- Check if you are noticing any uncorrectable or offline sectors if yes then it is most likely a failed drive and needs replacement.
- If the drive is good you would need to troubleshoot if it is the underlying hardware such as backplane/cables etc.

SAMBA SHARES ARE NOT ACCESSIBLE TO MOUNT

- Check if you have samba ports added to the firewall.

- Check if the samba services is running
- Check if you have set a separate samba password for the user
- Make sure you are accessing the share using the right password.

GETTING ACCESS DENIED WHEN ACCESSING THE FILES IN THE SHARE

- Make sure you are having the appropriate permission to access the files.
- Check the permission granted and see if the user getting access denied has access.
- It is most likely that he is not having access or is having access to just the top-level folder and not the child folders and files.
- Modify the permission as per your preference and you should be able to access the files.

HOW DO I UPDATE MY HL4

- You can update the HL4 by using the software updates tab in the Houston UI

45HomeLab
A Division of 45Drives

45DRIVES DISK MODULE IS NOT WORKING

- If the 45drives disks module does not show up and does not detect the drives you can perform device mapping manually using **sudo dalias -t HL4**
- Once the command finishes, reload the 45drives disks module and it should show the drives.

SYSTEM WOULD NOT BOOT INTO THE OS

- Try to go the boot menu by pressing F11 during the reboot.
- Once you are in the boot menu try manually select the boot drive and boot into it.
- If that works check if the boot order is set properly.
- Press Delete key to go to BIOS and check the boot order.

45HomeLab
A Division of 45Drives