Installing frigate from scratch

Some context:

This document walks through installing frigate from scratch. The document is written for a technical enthusiast that has at least a casual knowledge of linux. Others are welcome to follow along the best they can.

- I'd recommend installing frigate on your HA environment and skip the following complicated install process if you are running home assistant (HA) on something more substantial than a raspberry PI
    - Some caveats:
        - HA's media storage is limited to what's physically connected to the HA instance. As of this writing, there is no reasonable way to extend the media storage of HA to a network storage device. A multi-camera frigate environment, recording continuously, will fill a 1Tb disk up in less then a week. Then, HA will fail with a full disk situation and, well, you'll be forced to learn HA's restore process.
        - The HA hardware should be intel based (haven't tried AMD) to provide reasonable performance. I love the raspberry PI and variants but they are too weak to be useful with frigate.
- If your HA environment is **not** suitable, and for decent performance, reliable operation and expandable storage, its best to install frigate standalone on intel based hardware. You'll be able to use network attached storage that can be quite large and use relatively dedicated hardware (more on that later). When you're done installing frigate, you'll have an environment that invites other possible applications if you have any enthusiasm left.

When completed with this install guide, you will have:
A standalone frigate environment running in its own server. Under the covers, frigate runs on docker and is managed through portainer. You'll be able to run headless (no display) and remotely change the frigate config file through you nas share. Frigate can also be managed through the portainer web gui.

As an example, I am running a 7 camera system, 2 reolinks and 5 wyze cameras successfully with solid object detection. The nvr is integrated into home assistant through the frigate proxy and does things such as alerting me if my old dog has fallen into the pool! It can be done!

Author: GR

From scratch instructions:

Hardware:

While a wide variety of intel based hardware will work, I used the following that I put together 10 years ago and left for dead in the closet:

- Intel ASUS P8h77-M motherboard with an intel i5-3450, 12GB memory, 1TB samsung 850 EVO SSD

  - Note that lessor memory and disk space will work, this is just what I have. If you lack hardware, check out refurbished Dell or HP micro business computers on eBay or Amazon that cost around $200 – but try to get as recent an intel cpu generation as your budget allows. Also be sure there's a slot to plugin your coral chip.

  - If you can afford it, using a multi-TB ssd will avoid the need to use NAS storage and simplify your environment. A 4TB samsung disk is about $380 on amazon.

- Coral chip – this is a must or don't bother, sorry. They are hard to come by right now so good luck. I have a mini pci-e version plugged into a pcie adapter that's plugged into my motherboard.

  -
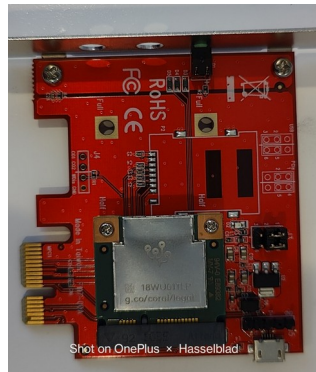
Installing frigate from scratch

Flow:

Set aside the better part of a day to complete the install.

Here's the flow:

1. Install ubuntu

2. Persistently mount NAS storage (if you have it)

3. Install Docker

4. Install Portainer

5. Install coral drivers

6. Configure frigate.yml

7. Create frigate container

Install ubuntu

You will need:

- Another computer and at least a 8g usb stick.

- The target computer all assembled and ready to go with coral, disk, memory, any graphics card, external nas up and running and a hardwired network connection.

On the other computer, navigate to:

- Install Ubuntu desktop, https://ubuntu.com/tutorials/install-ubuntu-desktop#1-overview

- ***Carefully follow the directions.*** Note: You'll need to install balenaEcther to create the bootable usb install stick. The ubuntu install instructions show you how. Be patient while things progress.

    ○ Some notes:

        ▪ Pick the most recent ubuntu. as of this writing it is 22.04.1 LTS.

        ▪ If lucky, your ubuntu pc will boot straight-up from the usb stick. If not, you'll need to get into the bios at boot time and figure out how to force the pc to boot from the stick.

        ▪ If you have multiple graphic cards, as I do, the pc may boot to a blank screen. Alternate where you are plugging in the monitor and/or hunt around in the bios setting to force only one graphics card to be active.

Author: GR

Installing frigate from scratch

- At "erase disk and install ubuntu", select advanced options and LVM. Encryption seems like overkill for in-home use but go ahead if you want.

- Help the installer along and when finally done (maybe 20 minutes) login to the offered services and any wrap-up as instructed by the install process.

You hopefully will now have a functioning ubuntu install from which to proceed.

Author: GR

Persistently mount NAS storage

If you plan to use your networked attached storage to store frigate video, then follow this step, if not move on.

- Be aware that a usable frigate system with 6 or 7 4k cameras storing continuous streams will quickly eat up TB's of storage in a few days.

- One alternative would be to use a large SSD (e.g., 4TB) from the start - if you have the budget - and skip the NAS part altogether.

Instructions:

There are many authors providing advice on how to persistently (over boots) mount NAS. They are all daunting if you've never done it before - so just carefully follow the instructions :-}

1. Create the cifs (smb) share on your nas device – I use a 10 year old Netgear ReadyNas that works fine.

   - Here's what I did, as an example, creating a share anticipating future expansion into other use cases:
     - mynas – hubfiles – frigate
     - or in windows: [\\mynas\hubfiles\frigate](\\mynas\hubfiles\frigate)
     - on windows use "add network location" to be sure the share is working

2. To mount on ubuntu, I'd recommend follow the directions from here:

   Mount Windows Shares Permanently

   I mounted my share at \mnt\hubfiles

   On reboot, your should see your share present in ubuntu's file manager. Check it by creating a file in it and seeing if it shows up on the nas.

Author: GR

Install Docker

Frigate runs as a container in docker. If you're unfamiliar with docker, try watching youtube videos and reading the plethora of info on the web. It is endless. But, take comfort that only rudimentary docker understanding is need to get frigate up and running.

Instructions:

Navigate to:

- [Install Docker Engine on Ubuntu](#)

- Carefully follow the instructions. Run the hello-world test as instructed at the end to be sure docker is operational. If not, you'll need to stop and figure out what's wrong.

Installing frigate from scratch

Install Portainer

Portainer provides a remote web ui to manager the docker environment. Hardcore commands for docker are not needed and it makes docker accessible to the more casual but still well informed user.

- Using portainer will circumvent the need to login into ubuntu or otherwize use ssh or a remote desktop whenever the frigate container needs modifying.
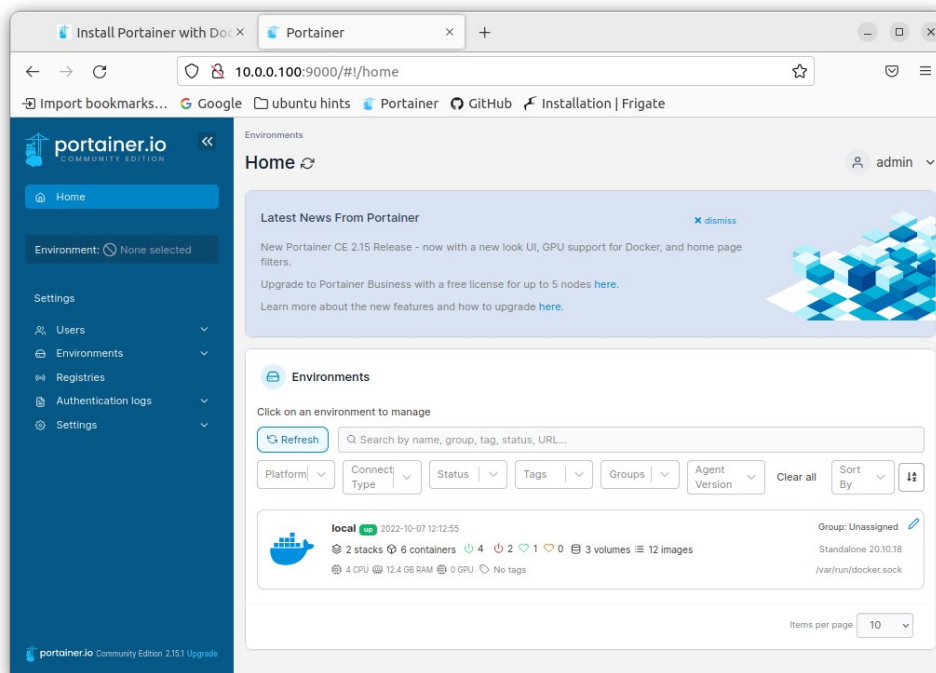
Instructions:

Navigate to:

- [Install Portainer with Docker on Linux](#)

- Carefully follow the instructions. Note that the latest version does not make portainer available on http, only https. For ease of use, I'd recommend adding "-p 9000:9000" into the docker run command to get http access.

  Something like this:

  > docker run -d -p 9000:9000 -p 8000:8000 -p 9443:9443 --name portainer
  > ……
  - Access portainer on your local network like this: http://myubuntucomputer_ip:9000

After creating your password, you should see something like this:

Author: GR

Install Coral drivers

The coral tensor gpu is vital for object detection. Without it, object detection is not practicable on a generic cpu. Drivers must be installed to make it available to frigate.

Instructions:

- **Coral:** Install the coral drivers on the ubuntu pc. Mini pcie coral will require this step. Not sure about the usb version so do it anyway, it won't hurt anything.

  - Navigate to:

    [Install the PCIe driver and Edge TPU runtime](#)

    Once completed the coral unit should be discoverable by frigate

- **Nvidia:** The nvidia support, assuming you have it, will be handled later when the frigate container is created. It is claimed to reduce the workload on the cpu whilst delivering video streams.

    You can check here: [Supported Nvidia GPUs for decoding](#) , to see if your nvidia gpu is supported. Apparently my 10 year old one is not, bummer.

Installing frigate from scratch

Create a starter config

Before launching the frigate container in portainer, a config file is needed.

Instructions:

In my case, my config.yml is located at mnt/hubfiles/frigate/frigate.yml. Note the name change to frigate.yml – more to come. See below for how that location is configured. Place either the sample below or your own concoction into the appropriate location to be read when the frigate container fires up. Note that if the config.yml is somehow wrong on container launch, frigate will report the error and then quit - then debugging ensues...

Sample starter config.yml – modify for your own setup

```
mqtt:
#point to your host
  host: 10.0.0.xx
  user: mqtt_user
  password: mqtt_pswd

database:
  path: /db/frigate.db

detectors:
  coral1:
    type: edgetpu
    device: pci:0

rtmp:
  enabled: false

birdseye:
  width: 1920
  height: 1080
  mode: continuous

cameras:
#-------------------------------------------------------------------------------------------------
  1_driveway:
    ffmpeg:
      hwaccel_args:
        - -hwaccel
        - vaapi
        - -hwaccel_device
        - /dev/dri/renderD128
        - -hwaccel_output_format
  #       - yuv420p
        - -c:v h264_cuvid
      output_args:
        # modification turns on audio recording using aac encoding
        record: -f segment -segment_time 10 -segment_format mp4 -reset_timestamps 1 -strftime 1 -c:v
copy -c:a aac
      inputs:
      #adjust for you camera, this is a reolink
        - path: rtsp://admin:pswd@10.0.0.xx:554/h264Preview_01_main
          roles:
            - record
      # by default, the detect stream is used for the live view in the browser
        - path: rtsp://admin:pswd@10.0.0.xx:554/h264Preview_01_sub
          roles:
            - detect
  # Optional: Adjust sort order of cameras in the UI. Larger numbers come later
```

Author: GR

Installing frigate from scratch

```yaml
ui:
  # By default the cameras are sorted alphabetically.
  order: 0
  # Optional: Whether or not to show the camera in the Frigate UI
  dashboard: True
detect:
  width: 2560
  height: 1920
  fps: 10
live:
  height: 2560
  quality: 8
objects:
  track:
    - person
    - dog
    - bicycle
    - car
```

Author: GR

Create the frigate stack

Now the fun begins and this is a part of the process that can cause much frustration.

You will be creating a "stack" in portainer and then deploying that stack to launch frigate.

Instructions:

- Launch portainer in a browser somewhere – could be on the ubuntu pc or a windows PC.

  Web address will generally be http://your_ubuntu_ip:9000, in my case it is http://10.0.0.100:9000,

  - Login to portainer. In the left pane click on "Stacks".

  - In the upper right click on "add stack"

  - Enter a name for the stack, e.g., frigate

  - Copy and paste the code below into the portainer stack editor and modify the directories to match yours.

    - As an example, note my customizations for the directory locations and shm-size:

      remember: in the earlier step I mounted my nas on these directories

      mnt/hubfiles/frigate/frigate.yml – I rename config.yml to frigate.yml and place it on the nas so that I can easily access it from windows using something like notepad++.

      mnt/hubfiles/frigate – this is where the video clips and recordings are stored which is located on my nas device

      home/user/frigate – the database is located on the ubuntu machine for performance. Running the database off the nas will diminish performance. "user" is your user name for ubuntu

      shm-size: "256mb" - To avoid buffer issues I needed to increase the shm size from 64mb for my 7 camera setup. I suggest you do the same just cause.

  - Make sure there are no errors reported inside the editor in red type. You may get an error like:

    name already used – this may be from a previous attempted install. In portainer, switch over to the container tab and delete any unwanted duplicate containers and start the stack creation process again.

Installing frigate from scratch

- At the bottom of the stack editor page hit "deploy the stack" and, if all is well, the container will start up.

Sample code to copy into the portainer stack editor, modify as needed for your mnt points:

```
version: "3.9"
services:
  frigate:
    container_name: frigate
    privileged: true # this may not be necessary for all setups
    restart: unless-stopped
    image: blakeblackshear/frigate:stable
    deploy:     # <------------- Add this section for nvidia
      resources:
        reservations:
          devices:
            - driver: nvidia
              device_ids: ['0'] # this is only needed when using multiple GPUs
              capabilities: [gpu]
    shm_size: "256mb" # update for your cameras based on calculation
    devices:
      - /dev/bus/usb:/dev/bus/usb # passes the USB Coral, needs to be modified for other versions
      - /dev/apex_0:/dev/apex_0 # passes a PCIe Coral
      - /dev/dri/renderD128 # for intel hwaccel, needs to be updated for your hardware
    volumes:
      - /etc/localtime:/etc/localtime:ro
      - /mnt/hubfiles/frigate/frigate.yml:/config/config.yml:ro  # modify to match your setup
      - /mnt/hubfiles/frigate:/media/frigate                     # modify to match your setup
      - /home/user/frigate:/db                                   # modify to match your setup
      - type: tmpfs # Optional: 1GB of memory, reduces SSD/SD Card wear
        target: /tmp/cache
        tmpfs:
          size: 1000000000
    ports:
      - "5000:5000"
#      - "1935:1935" # RTMP feeds – comment out if not needed , it may conflict with other containers
    environment:
      FRIGATE_RTSP_PASSWORD: "password"  #modify to whatever if using rtsp
```

Author: GR

Installing frigate from scratch

Hopefully you are up and running!

Once the container is running, you can inspect the frigate log by selecting "containers" and then the log icon under "quick actions" for the frigate container.

You may see weird errors, but ensure that frigate is not exiting, otherwise debug is necessary, and it can get frustrating. Make sure the frigate.yml is ok and the directory mappings are correct.

Go to a browser and navigate to: http://ubuntu_ip:5000. In my case http://10.0.0.100:5000. The frigate ui should launch and you are up and running. From this point, you can play with the frigate,yml to adjust your frigate nvr behavior with your cameras

> **weird errors** – if the frigate container won't start, you could have port conflicts or other weirdness. One thing to try is to login to the ubuntu machine and do some docker commands to stop and delete containers that are unwanted as well as the frigate container.
>
> > Useful docker commands:
> >
> > - **list running containers:** *sudo docker ps*
> > - **stop a container:** *sudo docker stop container_name*
> > - **remove a container:** *sudo dock rm container_name*
> > - *reboot the server:* might help…
>
> Try again to deploy the stack. If it still won't start then things get difficult but the first thing to try is to, assuming the container got created:
>
> - in portainer, select the container, frigate
> - in the upper right select "duplicate/edit"
> - at the bottom of the page select "Env"
> - delete the environment variable "PATH" from the list. There are reports of the path being incorrectly overridden.
> - If problems persist look for help in the frigate discussion area.

Hopefully this guide got you up and running with frigate. There is still much tweaking of your nvr to do, but at least you've got something to work with!

Author: GR

Installing frigate from scratch

Author: GR