Setup

Frigate NVR — Docker Setup Guide

Frigate is a self-hosted NVR (Network Video Recorder) with real-time object detection, face recognition, license plate recognition, and semantic search. This guide covers deploying Frigate via Docker Compose with go2rtc stream rebroadcasting, SMB-mounted NAS storage for footage, and role-based camera access control.

Prerequisites

- Docker and Docker Compose installed on the host
- A dedicated storage path for footage (local or NAS mount — see SMB section below)
- IP cameras with RTSP streams accessible from the host
- Host machine on the same network/VLAN as the cameras

Directory Structure

```
~/frigate/
├── docker-compose.yml
└── config/
    └── config.yml
```

SMB NAS Mount (fstab)

Footage is stored on a NAS via SMB share, mounted at boot using `/etc/fstab`. Add the following entry to `/etc/fstab`:

```
//YOUR_NAS_IP/share_name  /homedata/cameras  cifs  credentials=/etc/smb-
credentials,uid=1000,gid=1000,_netdev,auto  0  0
```

Create the credentials file:

bash

```
sudo nano /etc/smb-credentials
```

```
username=YOUR_SMB_USER
password=YOUR_SMB_PASSWORD
```

bash

```
sudo chmod 600 /etc/smb-credentials
sudo mkdir -p /mnt/nvr/cameras
sudo mount -a
```

Verify the mount is working before starting Frigate:

bash

```
df -h | grep cameras
```

Docker Compose

yaml

```
services:
  frigate:
    container_name: frigate
    privileged: true
    restart: unless-stopped
    image: ghcr.io/blakeblackshear/frigate:stable
    shm_size: "64mb"
    volumes:
      - /etc/localtime:/etc/localtime:ro
      - ./config:/config
      - /mnt/nvr/cameras:/media/frigate   # NAS SMB mount
      - type: tmpfs
        target: /tmp/cache
        tmpfs:
```

```
          size: 1000000000                # 1GB tmpfs for decode cache
    ports:
      - "5000:5000"      # Web UI
      - "8971:8971"      # go2rtc web UI
      - "8554:8554"      # RTSP restream feeds
      - "8555:8555/tcp"  # WebRTC
      - "8555:8555/udp"  # WebRTC
    environment:
      - FRIGATE_RTSP_PASSWORD=YOUR_RTSP_PASSWORD
```

> **Note:** `privileged: true` is required for hardware acceleration access on most setups. If you're not using a Coral TPU or specific GPU passthrough, you can try dropping this and mounting only the devices you need.

---

config.yml

yaml

```
version: 0.17-0

go2rtc:
  webrtc:
    listen: :8555
    candidates:
      - YOUR_HOST_IP:8555   # LAN IP of the Frigate host
      - stun:8555
  streams:
    camera_name_1:
      - rtsp://USER:PASS@CAMERA_IP:554/stream_path
    camera_name_2:
      - rtsp://USER:PASS@CAMERA_IP:554/stream_path
    # Add additional streams as needed

mqtt:
  enabled: false   # Enable and configure if using Home Assistant or MQTT alerts

# ------------------------------------------------------
```

```
# Recording
# ------------------------------------------------------
record:
  enabled: true
  continuous:
    days: 1          # 1 day of continuous footage retention
  detections:
    retain:
      days: 7
      mode: motion   # Keep only clips with motion
  alerts:
    retain:
      days: 14
      mode: motion   # Keep alert clips longer

# ------------------------------------------------------
# Auth & Role-Based Camera Access
# ------------------------------------------------------
auth:
  enabled: true
  roles:
    username:
      - camera_name_1
      - camera_name_2
      # List only the cameras this user should have access to

# ------------------------------------------------------
# FFmpeg Global Settings
# ------------------------------------------------------
ffmpeg:
  input_args: preset-rtsp-restream

# ------------------------------------------------------
# Object Detection
# ------------------------------------------------------
objects:
  track:
    - person
    - dog
    - cat
```

```
    - car
  filters:
    person:
      threshold: 0.7   # Confidence threshold — tune to reduce false positives

# ------------------------------------------------------
# AI Features
# ------------------------------------------------------
semantic_search:
  enabled: true
  model_size: small   # Options: small, large

face_recognition:
  enabled: true
  model_size: small

lpr:
  enabled: true        # License plate recognition

classification:
  bird:
    enabled: true

# ------------------------------------------------------
# Notifications
# ------------------------------------------------------
notifications:
  enabled: true
  email: YOUR_EMAIL@domain.com

# ------------------------------------------------------
# Cameras
# ------------------------------------------------------
cameras:
  camera_name_1:
    enabled: true
    ffmpeg:
      inputs:
        - path: rtsp://127.0.0.1:8554/camera_name_1?video=copy
          input_args: preset-rtsp-restream
```

```
        roles:
          - detect
          - record
          - audio
  detect:
    enabled: true
    width: 896
    height: 512
    fps: 8
  notifications:
    enabled: true

camera_name_2:
  enabled: true
  ffmpeg:
    inputs:
      # Use separate high-res and sub-stream inputs when available
      - path: rtsp://127.0.0.1:8554/camera_name_2_main
        input_args: preset-rtsp-restream
        roles:
          - record
          - audio
      - path: rtsp://127.0.0.1:8554/camera_name_2_sub
        input_args: preset-rtsp-restream
        roles:
          - detect
  live:
    height: 360
  detect:
    enabled: true
    width: 1280
    height: 360
    fps: 8
  notifications:
    enabled: true

# Camera with ONVIF PTZ support
ptz_camera:
  enabled: true
  ffmpeg:
```

```
      inputs:
        - path: rtsp://127.0.0.1:8554/ptz_camera?video=copy
          input_args: preset-rtsp-restream
          roles:
            - detect
            - record
            - audio
    detect:
      enabled: true
      width: 896
      height: 512
      fps: 8
    onvif:
      host: CAMERA_IP
      port: 80
      user: YOUR_USER
      password: YOUR_PASSWORD
    notifications:
      enabled: true
```

Starting Frigate

bash

```
cd ~/frigate
docker compose up -d
```

Check logs:

bash

```
docker logs -f frigate
```

Accessing the UI

Service	URL
Frigate Web UI	`http://HOST_IP:5000`

Service	URL
go2rtc Web UI	`http://HOST_IP:8971`
RTSP Restreams	`rtsp://HOST_IP:8554/camera_name`

Revision #1
Created 2026-05-14 14:45:58 UTC by Dale
Updated 2026-05-14 14:47:21 UTC by Dale