

Video Transcoder Deployment Guide

Video Transcoder Deployment Guide

1) Environment Requirements

- Ubuntu 24.04+ (recommended)
- Node.js 20+ (tested with 25)
- npm
- PostgreSQL 16+
- Redis 7+
- FFmpeg 6+
- Nginx
- Certbot (optional but recommended for SSL)
- PM2

2) Project Structure

- `/var/www/video-transcoder``
- `app/backend`` (API, queue worker, watcher)
- `app/frontend`` (dashboard static build)
- `data/upload`` (input files)
- `data/transcode`` (working files)
- `data/final`` (finished outputs)
- `.env`` (all configurable values)

3) Install Dependencies

```
``bash
sudo apt-get update
sudo apt-get install -y postgresql postgresql-contrib redis-server ffmpeg nginx certbot python3-certbot-nginx
sudo npm install -g pm2
``
```

4) Database Setup

```
``bash
sudo -u postgres psql -c "CREATE USER video_transcoder_user WITH PASSWORD 'CHANGE_ME';"
sudo -u postgres psql -c "CREATE DATABASE video_transcoder OWNER video_transcoder_user;"
``
```

5) Configure Environment

Copy and edit:

```
``bash
cp /var/www/video-transcoder/.env.example /var/www/video-transcoder/.env
nano /var/www/video-transcoder/.env
``
```

Important `.env`` values:

- `UPLOAD_DIR``
- `TRANSCODE_DIR``
- `FINAL_DIR``
- `DATABASE_URL``
- `REDIS_URL``
- `BACKEND_PORT``
- `FRONTEND_ORIGIN``
- `FFMPEG_PATH``

6) Install Node Packages and Build

```
``bash
cd /var/www/video-transcoder/app/backend
npm install
npx prisma generate
set -a && source /var/www/video-transcoder/.env && set +a
npx prisma db push
npm run build

cd /var/www/video-transcoder/app/frontend
npm install
npm run build
``
```

7) PM2 Process (Backend)

```
``bash
cd /var/www/video-transcoder
pm2 start ecosystem.config.js
pm2 save
pm2 startup systemd -u root --hp /root
``
```

8) Nginx Configuration

Create file `/etc/nginx/sites-available/uploader.softifysolutions.net``:

```
``nginx
server {
    listen 80;
    server_name uploader.softifysolutions.net;

    location /api/ {
        proxy_pass http://127.0.0.1:4000/;
        proxy_http_version 1.1;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
    }

    location /socket.io/ {
        proxy_pass http://127.0.0.1:4000/socket.io/;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection "upgrade";
        proxy_set_header Host $host;
    }

    root /var/www/video-transcoder/app/frontend/dist;
    index index.html;

    location / {
        try_files $uri $uri/ /index.html;
    }
}
```

```
}  
...
```

Enable and reload:

```
```bash  
sudo ln -s /etc/nginx/sites-available/uploader.softifysolutions.net /etc/nginx/sites-enabled/
sudo /usr/sbin/nginx -t
sudo systemctl restart nginx
...
```

## 9) SSL Certificate

```
```bash  
sudo PATH=$PATH:/usr/sbin certbot --nginx -d uploader.softifysolutions.net --non-interactive --agree-tos -m admin@softif  
ysolutions.net --redirect  
...
```

10) Functional Test

1. Put a test video in `UPLOAD_DIR`.
2. Check dashboard on domain.
3. Verify output appears in `FINAL_DIR`.
4. Verify `UPLOAD_DIR` and `TRANSCODE_DIR` are cleaned after success.

API quick checks:

```
```bash  
curl http://127.0.0.1:4000/health
curl http://127.0.0.1:4000/jobs
...
```

## 11) Upgrade / Redeploy

```
```bash  
cd /var/www/video-transcoder/app/backend && npm install && npm run build  
cd /var/www/video-transcoder/app/frontend && npm install && npm run build  
pm2 restart video-transcoder-backend  
...
```

12) Notes

- Queue is single-concurrency (one transcode at a time).
- All runtime settings are in `.env` for easy migration to new servers.
- Backup PostgreSQL database and `.env` before major updates.