

# Hướng Dẫn Cài GitLab, Docker Registry, GitLab Runner Trên Docker

## Giới Thiệu

Tài liệu này hướng dẫn chi tiết cài đặt **GitLab**, **Docker Registry**, và **GitLab Runner** trên Docker, kèm các bước khắc phục lỗi mạng (TLS handshake timeout, network bridge, domain mapping, v.v.).

## Cấu Trúc Docker Compose Chuẩn

```
services:
  mysql:
    image: mysql:8.0
    environment:
      MYSQL_ROOT_PASSWORD: root
      MYSQL_DATABASE: flask_login_demo
      MYSQL_USER: kheuser
      MYSQL_PASSWORD: 123
    ports:
      - "3306:3306"
    network_mode: bridge
    healthcheck:
      test: ["CMD", "mysqladmin", "ping", "-h", "localhost", "-uroot", "-proot"]
      interval: 10s
      timeout: 5s
      retries: 5

  redis:
    image: redis:7.4
    ports:
      - "6379:6379"
    healthcheck:
      test: ["CMD", "redis-cli", "ping"]
      interval: 5s
      timeout: 3s
      retries: 5
    network_mode: bridge

  web1:
    image: lamluongkhe/flask-redis-mysql:1.0
    environment:
      MYSQL_HOST: 192.168.0.25
      MYSQL_DATABASE: flask_login_demo
      MYSQL_USER: kheuser
      MYSQL_PASSWORD: 123
```

```
REDIS_HOST: 192.168.0.25
REDIS_PORT: 6379
REDIS_DB: 0
ports:
  - "5001:5001"
depends_on:
  mysql:
    condition: service_healthy
  redis:
    condition: service_healthy
network_mode: bridge
```

---

## Cài Đặt Docker Registry

```
docker run -d
  -p 5000:5000
  --name registry
  --restart=always
  -v /opt/registry/data:/var/lib/registry
  registry:2
```

Truy cập để kiểm tra: [http://localhost:5000/v2/\\_catalog](http://localhost:5000/v2/_catalog)

---

## Kéo & Push Image Lên Registry Local

```
# Kéo image từ Docker Hub
sudo docker pull lamluongkhe/web:1.0

# Tag lại cho registry local
sudo docker tag lamluongkhe/web:1.0 192.168.0.25:5000/lamluongkhe/web:1.0

# Push lên local registry
sudo docker push 192.168.0.25:5000/lamluongkhe/web:1.0
```

---

## Cài Đặt GitLab Runner

```
docker run -d
  --name gitlab-runner
  --restart always
  -v /srv/gitlab-runner/config:/etc/gitlab-runner
  -v /var/run/docker.sock:/var/run/docker.sock
  gitlab/gitlab-runner:latest
```

---

## Cách Xử Lý Khi Runner Không Ping Được 8.8.8.8 Hoặc Lỗi TLS

### Cách Xử Lý

#### Bước 1 — Kiểm Tra Mạng


```
docker exec -it gitlab-runner ping -c 3 8.8.8.8
```

Nếu không ping được:

```
systemctl restart docker
docker network rm bridge
docker network create bridge
```

#### Bước 2 — Sửa File `/etc/hosts` Trong Container

```
docker exec -it gitlab-runner bash
echo "192.168.1.252 gitlab.local.com" >> /etc/hosts
ping -c 3 gitlab.local.com
```

 **IP** là IP của host hoặc container GitLab.

#### Bước 3 — Đăng Ký Lại Runner

```
sudo docker exec -it gitlab-runner gitlab-runner register
```

- **URL:** `http://gitlab.local.com/` - **Token:** Lấy trong *GitLab* → *Admin* → *CI/CD* → *Runners* - **Executor:** `docker`  
- **Default image:** `alpine:latest`

Sau khi đăng ký thành công:

```
docker restart gitlab-runner
```

 Runner sẽ hiện trong **GitLab** → **Admin** → **Runners**.


---

## Kiểm Tra Các Lỗi Thường Gặp

Lỗi	Nguyên nhân	Cách khắc phục
<code>TLS handshake timeout</code>	Mạng bridge hoặc DNS Docker lỗi	Restart Docker, xóa và tạo lại network bridge
<code>invalid config for network bridge</code>	Compose khai báo alias sai	Xóa alias trong network, để <code>network_mode: bridge</code>
<code>manifest unknown</code>	Image không tồn tại	Kiểm tra tag trên Docker Hub trước khi pull
<code>UNAUTHORIZED</code>	Thiếu token registry	Login và push lại: <code>docker login localhost:5000</code>

## Tóm Tắt

- Docker Registry giúp lưu image nội bộ.
- Runner cần domain có trong `/etc/hosts` để kết nối với GitLab.
- Lỗi mạng, TLS, hoặc bridge thường xảy ra do Docker DNS.
- Luôn tạo network bridge mới nếu container không truy cập ngoại mạng.

 Tài liệu hoàn chỉnh — có thể xuất ra PDF!