

Deploying Installation Tools with Docker Compose

Prerequisites

Before we begin, ensure you have the following installed on your system:

1. **Docker** and **Docker Compose**: Installation instructions can be found [here](#).
2. **Git**: Installation instructions can be found [here](#).

Step 1: Logging into Docker

To access a private Docker registry, you'll need to authenticate with your Docker credentials.

1. Log in to Docker:

```
1 | docker login
```

Follow the prompts to enter your Docker username and password.

Step 2: Create a Directory for Deployment

Create a directory where you will download and store the Docker Compose file.

1. Open a terminal.
2. Create a new directory:

```
1 | mkdir -p /app/liberty-admin  
2 | cd /app/liberty-admin
```

Step 3: Download the Docker Compose File

Next, download the Docker Compose file from the provided URL.

1. Using `curl`:

```
1 | curl -L -o docker-compose.yml https://github.com/fblettner/liberty-  
public/blob/main/release/latest/liberty-admin.yml
```

2. Alternatively, using `wget`:

```
1 | wget -O docker-compose.yml https://github.com/fblettner/liberty-  
public/blob/main/release/latest/liberty-admin.yml
```

Step 4: Deploy the Docker Container using Docker Compose

Once you have the `docker-compose.yml` file downloaded into your `liberty-admin` directory, use Docker Compose to deploy the container.

1. In the terminal, navigate to the `liberty-admin` directory (if not already there):

```
1 | cd /app/liberty-admin
```

2. Deploy the Docker container:

```
1 | docker-compose up -d
```

This command will pull the necessary images from the registry (if they are not already available locally) and start the containers in detached mode.

Step 5: Verify the Deployment

To ensure the deployment is successful, you can check the status of the containers.

1. List the running containers:

```
1 | docker ps
```

You should see the following containers running as defined in the `docker-compose.yml` file:

- **traefik:** This service is managing routing and load balancing, and exposes several endpoints for web (port 3000), websecure (port 3443), dashboard (port 8080), and database (port 5432).
- **portainer:** This service provides a UI for managing Docker environments, accessible via paths prefixed with `/portainer`.
- **error-pages:** This service handles error pages and is available to respond to general HTTP requests.

Summary of Commands

```
1 | # Log in to Docker
2 | docker login
3 |
4 | # Create and navigate to the admin directory
5 | mkdir -p /app/liberty-admin
6 | cd /app/liberty-admin
7 |
8 | # Download the Docker Compose file
9 | curl -L -o docker-compose.yml https://raw.githubusercontent.com/fblettner/liberty-
10 | public/release/latest/liberty-admin.yml
11 | # or using wget
12 | wget -O docker-compose.yml https://raw.githubusercontent.com/fblettner/liberty-
13 | public/release/latest/liberty-admin.yml
14 |
    | # Deploy the Docker container
```

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

Accessing Services

After deployment, you can access the services with the following URLs:

- **Traefik Dashboard:** Accessible at `http:// <your_server_ip> :8080/dashboard/` (authentication may be required).
- **Portainer:** Accessible at `http:// <your_server_ip> :3000/portainer` or `https:// <your_server_ip> :3443/portainer`.

Replace `<your_server_ip>` with the IP address or hostname of your server. Feel free to reach out if you have any further questions or run into any issues!