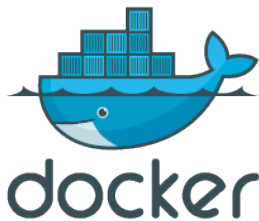# Step-by-Step Setup Guide for Databag with Traefik

This guide will walk you through the setup process of deploying Databag, a decentralized and lightweight messaging system, in combination with Traefik as a reverse proxy using Docker Swarm. The configuration ensures that Databag runs efficiently across your Swarm nodes with persistent data storage.

Step 1: Set Up Data Directory in GlusterFS

To ensure that Databag's data is available across all Docker Swarm nodes, you will need to create a shared directory using GlusterFS. This directory will store the application's data.

Run the following command:

```
mkdir /mnt/glustermount/data/databagchat_data
```

This directory will be used to store Databag's data, making it accessible across all nodes in the swarm.

Step 2: Create a `docker-compose.yaml` File

Next, you need to create a `docker-compose.yaml` file that defines the Databag service and its configuration. This file will also include Traefik as the reverse proxy to handle incoming HTTP/HTTPS traffic.

1. **Create the file**:
   Navigate to the directory where you want to store your Docker Compose file, and create it with the following command:

   ```
   nano docker-compose.yaml
   ```

2. **Edit the Admin Password**:
   Replace `${DATABAG_PASSWORD}` with a secure password or define it as an environment variable for added security. This password will be used to manage your Databag instance.

Here's the Docker Compose file structure:

```
version: "3.8"

services:
  databag:
    image: balzack/databag:0.1.17
    environment:
      # The ADMIN environment variable sets the admin password for the Databag application.
      # Replace ${DATABAG_PASSWORD} with your actual password or securely manage it via a secrets manager.
      - ADMIN=${DATABAG_PASSWORD}
    volumes:
      # The following volume ensures persistent storage for Databag's data across Swarm nodes.
      - /mnt/glustermount/data/databagchat_data:/var/lib/databag
    networks:
```

```
      - management_net
    deploy:
      # Deploy in "replicated" mode ensures the service runs across the number of replicas specified below.
      # In this case, the replicas are set to 1, meaning only one instance of the Databag service will run.
      mode: replicated
      replicas: 1
      labels:
        - 'traefik.enable=true'
        - 'traefik.http.routers.databag.rule=Host(`databag.domain.tld`)'
        - 'traefik.http.routers.databag.entrypoints=websecure'
        - 'traefik.http.routers.databag.tls.certresolver=leresolver'
        - 'traefik.http.services.databag.loadbalancer.server.port=7000'
        - 'traefik.docker.network=management_net'


  networks:
    management_net:
      external: true
```

Optional: Set Additional Environment Variables

Here are some additional environment variables that can be set for further customization:

- `ADMIN` : Sets the admin password for the Databag application. Replace `${DATABAG_PASSWORD}` with a strong, secure password.
- `DEV` : Set this to `1` if you want to launch the server manually for development purposes.

> IF YOU WANT TO USE AUDIO/VIDEO CALL FUNCTION, GO VISIT **THIS ARTICLE**.

Step 3: Deploy the Stack in Docker Swarm

Once the Docker Compose file is configured, you can deploy the stack to Docker Swarm. This will ensure that Databag and Traefik are running across your nodes.

Deploy the stack using the following command:

```
docker stack deploy -c docker-compose.yaml databag
```

This command will start the Databag service with the Traefik reverse proxy, ensuring that traffic is correctly routed to your Databag instance.

You can now access your Databag Server underhttps://databag.domain.tld/ and Login as Admin to configure more details.

Conclusion

By following this guide, you have successfully set up Databag with Traefik in a Docker Swarm environment. The use of GlusterFS ensures persistent storage across your nodes, while Traefik manages traffic to your Databag service. You can now manage your decentralized chat service with a secure setup, fully capable of scaling across multiple nodes in your Swarm cluster.