Dockpeek

Docker Monitoring Tool

- Installing Dockpeek

Installing Dockpeek

Here is the guide to install and standup Dockpeek tool that will monitor all docker and docker images on a server and remote servers with additional agent install

Github Repository

Install Dockpeek

1. SSH to Overseer machine, or get in trough Portainer/ Other GUI Compose Interface
2. Run following docker compose

```
services:
  dockpeek:
    container_name: dockpeek
    image: ghcr.io/dockpeek/dockpeek:latest
    environment:
      - SECRET_KEY=my_secret_key #change this
      - USERNAME=admin #change this
      - PASSWORD=admin #change this
    ports:
      - "3420:8000" #change 3420 as necessary for your setup. do not change 8000.
    volumes:
      - /var/run/docker.sock:/var/run/docker.sock #this is how we connect to the docker-
socket.
    restart: unless-stopped #this is fine.
```

Adjust environment info for secure login

Install Dockpeek Agents for Remote Clients

If you have multiple server running dockers on a local network

1. Run following docker compose for installing agents

```
### Don't change anything here. Just deploy it as it is on the node/server you want to monitor
services:
  dockpeek-socket-proxy:
    image: lscr.io/linuxserver/socket-proxy:latest
    container_name: dockpeek-socket-proxy
    environment:
      - CONTAINERS=1
      - IMAGES=1
      - PING=1
      - VERSION=1
      - INFO=1
      - POST=1       # <-- This is needed for "Check for updates" operations
    volumes:
      - /var/run/docker.sock:/var/run/docker.sock:ro
    read_only: true
    tmpfs:
      - /run
    ports:
      - "2375:2375"
    restart: unless-stopped
```

2. Once all agents are installed adjust main Dockpeek compose file by adding additional variables under environment

```
services:
  dockpeek:
    container_name: dockpeek
    image: ghcr.io/dockpeek/dockpeek:latest
    environment:
      - SECRET_KEY=my_secret_key #change this
      - USERNAME=admin #change this
      - PASSWORD=admin #change this

      # Docker Host 1 (This is our local server)
      - DOCKER_HOST_1_URL=unix:///var/run/docker.sock #this connects to the local docker socket.
      - DOCKER_HOST_1_NAME=LocalHost #this is the identifier that will show in the dashboard.
      - DOCKER_HOST_1_PUBLIC_HOSTNAME=192.168.0.18 #this is the IP address of the server (no http/https).
```

```
      # Docker Host 2 (This is a remote server on our local network)
      - DOCKER_HOST_2_URL=tcp://192.168.0.107:2375 #change the IP address to your remote server's IP address. Don't change anything else.
      - DOCKER_HOST_2_NAME=OpenCloud #this is the identifier that will show in the dashboard.
      - DOCKER_HOST_2_PUBLIC_HOSTNAME=192.168.0.107 #this is the IP address of the server (no http/https).

      # Docker Host 3 (This is a remote server on our local network)
      - DOCKER_HOST_3_URL=tcp://192.168.0.108:2375 #change the IP address to your remote server's IP address. Don't change anything else.
      - DOCKER_HOST_3_NAME=ClosedCloud #this is the identifier that will show in the dashboard.
      - DOCKER_HOST_3_PUBLIC_HOSTNAME=192.168.0.108 #this is the IP address of the server (no http/https).

      # Keep adding more hosts as necessary. Be sure to increase the number of the Docker Host.

    ports:
      - "3420:8000" #change 3420 as necessary for your setup. do not change 8000.
    volumes:
      - /var/run/docker.sock:/var/run/docker.sock #this is how we connect to the docker-socket.
    restart: unless-stopped #this is fine.
```