Docker Compose for multi-container applications, Docker security best practices

Docker Compose is a tool for defining and running multi-container Docker applications. With Compose, user use a YAML file to configure their application's services. Then, with a single command, we create and start all the services from our configuration.

Step 1: Define the Application Services

Create a file named docker-compose.yml in your project directory and define your application services.

```
version: '3.8'
services:
    web:
        image: nginx:latest
        ports:
            - "80:80"
        volumes:
            - ./html:/usr/share/nginx/html
        depends_on:
            - db
    db:
        image: postgres:latest
        environment:
            POSTGRES_USER: user
            POSTGRES_PASSWORD: password
            POSTGRES_ DB:database
        volumes:
            - db_ data:/var/lib/postgresql/data

volumes:
    db_data:
```

Step 2: Run the Application

Run the following command in the directory containing your docker-compose.yml file to start your application.

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

In such a way, we have create a docker-compose.yml file that will manage our multiple container in out application.

Docker Security Best Practices

Securing your Docker environment is crucial to protect your applications and data.

1. Use Minimal Base Images

Use minimal base images to reduce the attack surface. For example, use alpine instead of ubuntu when possible.

```
FROM alpine:latest
```

2. Keep Images Up-to-Date

Regularly update your base images and dependencies to include the latest security patches.

3. Run Containers as Non-Root Users

Avoid running containers as the root user to minimize the risk of privilege escalation.

```
RUN useradd -m myuser
USER myuser
```

4. Use Docker Content Trust

Enable Docker Content Trust to ensure that images are signed and verified.

```
export DOCKER_CONTENT_TRUST=1
```