Sprawozdanie

Programowanie aplikacji w chmurze obliczeniowej

Laboratorium 5

Wykorzystanie magazynów przechowywania

danych w środowisku Docker

Łukasz Oleksiuk

Grupa: 6.6

Index: 097690

Zadanie 5.1. Podstawowa obsługa wolumenów

Utworzenie własnego wolumenu o nazwie RedisRob:

```
student@vhost1:~/docker_lab4$ sudo  docker volume create RedisRob
[sudo] password for student:
RedisRob
```

Uruchomienie serwera Redis z wykorzystaniem wolumenu RedisRob:

```
student@vhost1:~/docker_lab4$ sudo docker run -d --name redis_server -v RedisRob:/data redis:latest
Unable to find image 'redis:latest' locally
latest: Pulling from library/redis
8a1e25ce7c4f: Already exists
8ab039a68e51: Pull complete
2b12a49dcfb9: Pull complete
cdf9868f47ac: Pull complete
e73ea5d3136b: Pull complete
890ad32c613f: Pull complete
4f4fb700ef54: Pull complete
ba517b76f92b: Pull complete
Digest: sha256:3134997edb04277814aa51a4175a588d45eb4299272f8eff2307bbf8b39e4d43
Status: Downloaded newer image for redis:latest
4b8d7adcc45f6f755b146eb58df5347597d3189182affc165923cec63ec0b116
```

Potwierdzenie poprawności konfiguracji:

```
student@vhost1:~/docker_lab4$ sudo docker inspect redis_server
[
    {
        "Id": "4b8d7adcc45f6f755b146eb58df5347597d3189182affc165923cec63ec0b116",
        "Created": "2024-04-08T23:55:40.330235853Z",
        "Path": "docker-entrypoint.sh",
        "Args": [
            "redis-server"
        ],
        "State": {
            "Status": "running",
            "Running": true,
            "Paused": false,
            "Restarting": false,
            "OOMKilled": false,
            "Dead": false,
            "Pid": 43466,
            "ExitCode": 0,
            "Error": "",
            "StartedAt": "2024-04-08T23:55:40.723062956Z",
            "FinishedAt": "0001-01-01T00:00:00Z"
        },
        "Image": "sha256:170a1e90f8436daa6778aeea3926e716928826c215ca23a8dfd8055f663f9428",
        "ResolvConfPath": "/var/lib/docker/containers/4b8d7adcc45f6f755b146eb58df5347597d318
        "HostnamePath": "/var/lib/docker/containers/4b8d7adcc45f6f755b146eb58df5347597d3189
        "HostsPath": "/var/lib/docker/containers/4b8d7adcc45f6f755b146eb58df5347597d3189182
        "LogPath": "/var/lib/docker/containers/4b8d7adcc45f6f755b146eb58df5347597d3189182af
        "Name": "/redis_server",
        "RestartCount": 0,
        "Driver": "overlay2",
        "Platform": "linux",
        "MountLabel": "",
        "ProcessLabel": "",
        "AppArmorProfile": "docker-default",
        "ExecIDs": null,
        "HostConfig": {
            "Binds": [
                "RedisRob:/data"
            ],
            "ContainerIDFile": "",
            "LogConfig": {
                "Type": "json-file",
                "Config": {}
            },
            "NetworkMode": "default",
            "PortBindings": {},
            "RestartPolicy": {
                "Name": "no",
                "MaximumRetryCount": 0
            },
```

Zadanie 5.2. Zaawansowana obsługa wolumenów

Budowanie kontenerów z plików Dockerfile

```
student@vhost1:~/docker_lab5$ sudo docker build -t first-server -f Dockerfile1 .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
            Install the buildx component to build images with BuildKit:
            https://docs.docker.com/go/buildx/

Sending build context to Docker daemon   5.12kB
Step 1/3 : FROM httpd:latest
 ---> 147ddc9b1d39
Step 2/3 : COPY index.html /var/www/html/
 ---> 92fbd933e5f8
Step 3/3 : EXPOSE 80
 ---> Running in 637cbae68ad4
Removing intermediate container 637cbae68ad4
 ---> 17012f0f963d
Successfully built 17012f0f963d
Successfully tagged first-server:latest
```

```
student@vhost1:~/docker_lab5$ sudo docker build -t second-server -f Dockerfile2 .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
            Install the buildx component to build images with BuildKit:
            https://docs.docker.com/go/buildx/

Sending build context to Docker daemon   5.12kB
Step 1/3 : FROM httpd:latest
 ---> 147ddc9b1d39
Step 2/3 : COPY --from=first-server /var/www/html/ /var/www/html/
 ---> cbf323cc004e
Step 3/3 : EXPOSE 80
 ---> Running in 36e52f0c4c1e
Removing intermediate container 36e52f0c4c1e
 ---> 983b942fe9a5
Successfully built 983b942fe9a5
Successfully tagged second-server:latest
```

```
student@vhost1:~/docker_lab5$ sudo docker build -t third-server -f Dockerfile3 .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
            Install the buildx component to build images with BuildKit:
            https://docs.docker.com/go/buildx/

Sending build context to Docker daemon   5.12kB
Step 1/3 : FROM httpd:latest
 ---> 147ddc9b1d39
Step 2/3 : COPY --from=first-server /var/www/html/ /var/www/html/
 ---> Using cache
 ---> cbf323cc004e
Step 3/3 : EXPOSE 80
 ---> Using cache
 ---> 983b942fe9a5
Successfully built 983b942fe9a5
Successfully tagged third-server:latest
```

Uruchomienie kontenerów

```
student@vhost1:~/docker_lab5$ sudo docker run -d -v volume_name:/var/www/html/ -p 8090:80 --name first-container first-server
17ae28837de6198b1b216bfd55d0b81fbc6baebf10b710a909acbfb0f80adf60
```

```
student@vhost1:~/docker_lab5$ sudo docker run -d -v volume_name:/var/www/html/ -p 8091:80 --name second-container second-server
e501f0ba8408973ad774b2771c9f1520741d61ceb6fa2902b3ad878f82325264
```

```
student@vhost1:~/docker_lab5$ sudo docker run -d -v volume_name:/var/www/html/ -p 8092:80 --name third-container third-server
6712631165e5005c7ea164c0845af5ec9661224b4fd21d2ebdaf6fd42970acb9
```

Stworzone pliki


Dockerfile1


Dockerfile2


Dockerfile3


index.html

Zawartość stworzonych plików Dockerfile

*Dockerfile1

```
1 FROM httpd:latest
2
3 COPY index.html /var/www/html/
4
5 EXPOSE 80
6
```

*Dockerfile2 *Dockerfile1

```
1 FROM httpd:latest
2
3 COPY --from=first-server /var/www/html/ /var/www/html/
4
5 EXPOSE 80
6
```

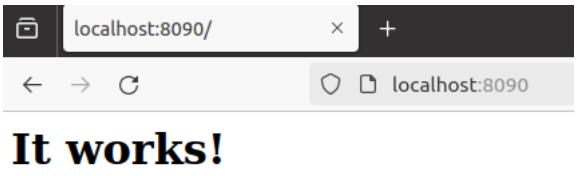
*Dockerfile3 *Dockerfile2

```
1 FROM httpd:latest
2
3 COPY --from=first-server /var/www/html/ /var/www/html/
4
5 EXPOSE 80
6
```

Plik index.html

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7     <title>Simple HTML Page</title>
8 </head>
9 <body>
10     <h1>Hello, world!</h1>
11     <p>This is a simple HTML page served by Apache server in a Docker container.</p>
12 </body>
13 </html>
```

Widok strony na serwerze



Wnioski:

Laboratorium pozawala zapoznać się z konteneryzacją aplikacji webowych i podstawowym zarządzaniem woluminami w środowisku Docker. Sposób wykorzystania woluminów w tym ćwiczeniu umożliwia ograniczenie redundancji danych.