Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
Высшего образования
Факультет Программной Инженерии и Компьютерной Техники

Лабораторная работа 1 по Облачным и туманным вычислениям

Группа: P3416
Выполнил:
Сиразетдинов А.Н.
Проверил:
Перл И. А.

Г. Санкт-Петербург
2026

Оглавление

Выполнение

Приложение

Напишем простое flask приложение

```python
from flask import Flask
import redis
import os

app = Flask(__name__)

redis_host = os.getenv("REDIS_HOST", "localhost")
r = redis.Redis(host=redis_host, port=6379,
decode_responses=True)
@app.route("/")
def hello():
    count = r.incr("hits")
    return f"<h1>Количество посещений: {count}</h1>"

if __name__ == "__main__":
    app.run(host="0.0.0.0", port=8080)
```

Dockerfile и docker-compose.yml

```
FROM python:3.10.19-slim-trixie

WORKDIR /app

COPY requirements.txt .
RUN pip install --no-cache-dir -r requirements.txt

COPY app.py .

EXPOSE 5000

CMD ["python", "app.py"]
```

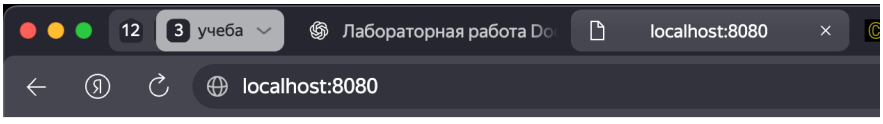
```
version: "3.9"

services:
  frontend:
    build: .
    ports:
      - "8080:8080"
    environment:
      - REDIS_HOST=redis
    depends_on:
      - redis

  redis:
    image: redis:7-
alpiperts:
      - "6379:6379"
```

Запустим командой docker compose up

```
Terminal  Local
redis-1  | 1:M 17 Jan 2026 21:58:10.030 * Server initialized
redis-1  | 1:M 17 Jan 2026 21:58:10.030 * Loading RDB produced by version 7.4.7
redis-1  | 1:M 17 Jan 2026 21:58:10.030 * RDB age 12 seconds
redis-1  | 1:M 17 Jan 2026 21:58:10.030 * RDB memory usage when created 0.98 Mb
redis-1  | 1:M 17 Jan 2026 21:58:10.030 * Done loading RDB, keys loaded: 1, keys expired: 0.
redis-1  | 1:M 17 Jan 2026 21:58:10.030 * DB loaded from disk: 0.000 seconds
redis-1  | 1:M 17 Jan 2026 21:58:10.030 * Ready to accept connections tcp
frontend-1  |  * Serving Flask app 'app'
frontend-1  |  * Debug mode: off
frontend-1  | WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server inste
frontend-1  |  * Running on all addresses (0.0.0.0)
frontend-1  |  * Running on http://127.0.0.1:5000
frontend-1  |  * Running on http://10.89.15.5:5000
frontend-1  | Press CTRL+C to quit
```



Количество посещений: 6

Инициализация swarm

```
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Post "http://%2Fvar%2Frun%2Fdocker.sock/
azat222@compute-vm-2-2-40-ssd-1759532797367:~/clouds$ sudo docker swarm init
Swarm initialized: current node (iole3bzruga8a7hooehjtkiaj) is now a manager.

To add a worker to this swarm, run the following command:

    docker swarm join --token SWMTKN-1-66htbqw186w17h10ddwxvj8oxsqlkimper09mk3p8ih8fk7v2i-8b8wk3uptin2u3f63uudwdsxj 10.128.0.12:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.
azat222@compute-vm-2-2-40-ssd-1759532797367:~/clouds$ docker node ls
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v
azat222@compute-vm-2-2-40-ssd-1759532797367:~/clouds$ sudo docker node ls
ID                            HOSTNAME                                STATUS    AVAILABILITY   MANAGER STATUS   ENGINE VERSION
iole3bzruga8a7hooehjtkiaj *   compute-vm-2-2-40-ssd-1759532797367   Ready     Active         Leader           28.5.2
azat222@compute-vm-2-2-40-ssd-1759532797367:~/clouds$
```

Соберем образ

sudo docker build -t flask-counter:1.0 .

Добавим образ в compose файл

```yaml
version: "3.9"

services:
  frontend:
    build: .
    image: flask-counter:1.0
    ports:
      - "8080:8080"
    environment:
      - REDIS_HOST=redis
    depends_on:
      - redis

  redis:
    image: redis:7-alpine
    ports:
      - "6379:6379"
```

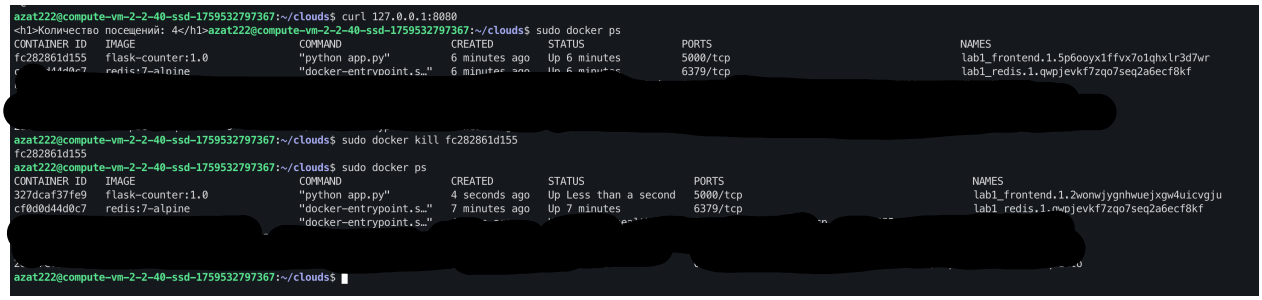
Запустим контейнеры

sudo docker stack deploy -c docker-compose.yml lab1

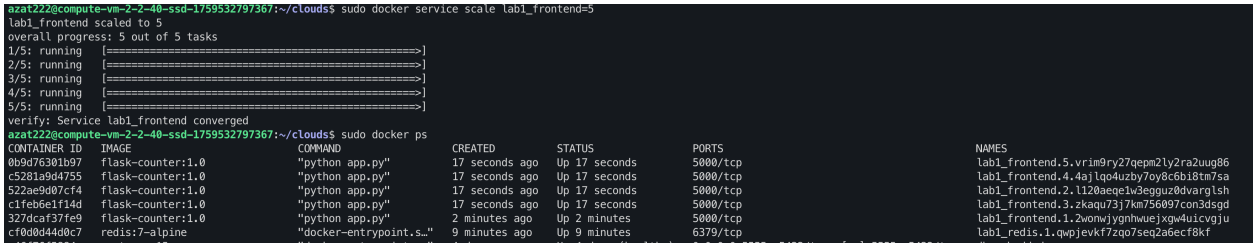
```
azat222@compute-vm-2-2-40-ssd-1759532797367:~/clouds$ sudo docker service ls
ID             NAME            MODE         REPLICAS   IMAGE              PORTS
afd983eweyhd   lab1_frontend   replicated   1/1        flask-counter:1.0   *:8080->8080/tcp
2vx2eyha0rgp   lab1_redis      replicated   1/1        redis:7-alpine      *:6379->6379/tcp
azat222@compute-vm-2-2-40-ssd-1759532797367:~/clouds$ sudo docker service ps lab1_frontend
ID             NAME              IMAGE              NODE                                  DESIRED STATE   CURRENT STATE               ERROR     PORTS
hhf73j0so245   lab1_frontend.1   flask-counter:1.0   compute-vm-2-2-40-ssd-1759532797367   Running         Running about a minute ago
azat222@compute-vm-2-2-40-ssd-1759532797367:~/clouds$
```

```
azat222@compute-vm-2-2-40-ssd-1759532797367:~/clouds$ curl 127.0.0.1:8080
<h1>Количество посещений: 1</h1>azat222@compute-vm-2-2-40-ssd-1759532797367:~/clouds$ curl 127.0.0.1:8080
<h1>Количество посещений: 2</h1>azat222@compute-vm-2-2-40-ssd-1759532797367:~/clouds$ curl 127.0.0.1:8080
```

Self-healing



Маситабирование



Вывод

В лабораторной работе я написал простое Flask приложение и познакомился с docker swarm