

**РОССИЙСКОЙ ФЕДЕРАЦИИ**  
**Федеральное государственное автономное**  
**образовательное учреждение высшего образования**  
**«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций**  
**«Работа в Docker с сетью контейнеров и томами»**

**Отчет по лабораторной работе**  
**по дисциплине «Анализ данных»**

Выполнил студент группы ИВТ-б-о-21-1

Уланбекова Айканыш Уланбековна

«20» ноября 2023г.

Подпись студента \_\_\_\_\_

Работа защищена «    » \_\_\_\_\_ 20\_\_ г.

Проверил Воронкин Р.А. \_\_\_\_\_  
(подпись)

Ставрополь 2023

**Цель работы:** научиться использовать Docker для управления томами и сетями.

### Порядок выполнения работы:

#### Задача 1: Создание пользовательской сети:

Создайте пользовательскую сеть в Docker с именем "my\_custom\_network". Запустите два контейнера, присоединенных к этой сети, например, с использованием образов Nginx и PostgreSQL. Убедитесь, что они могут взаимодействовать друг с другом.

```
PS C:\Users\User> docker run -d --name my_nginx --network my_custom_network nginx
9b1531f42018f2c38f45721b4be154d4e28e72fd6714e12f614ca1560dd9d029
PS C:\Users\User> docker run -d --name my_postgres --network my_custom_network -e POSTGRES_PASSWORD=postgres postgres
7e3bfd1ca6f78f3dfa6e4451a0fe758f78374b991315665b647acf287f3b0655
PS C:\Users\User> docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
7e3bfd1ca6f7	postgres	"docker-entrypoint.s..."	4 seconds ago	Up 3 seconds	5432/tcp	my_postgres
9b1531f42018	nginx	"/docker-entrypoint...."	12 seconds ago	Up 10 seconds	80/tcp	my_nginx

```
PS C:\Users\User>
```

Рисунок 1 – Создание сети

#### Задача 2: Передача данных через тома:

Создайте Docker-контейнер с использованием тома. Запишите данные в том из одного контейнера, а затем прочитайте их из другого контейнера, используя тот же том. Обеспечьте, чтобы данные сохранялись после перезапуска контейнеров.

```
PS C:\Users\User> docker run -d --name my_nginx -v my_custom_volume:/my_volume nginx
139c1e505f02b86221618cf553d4b5f08f39a0efa3d29287c5b761281e143558
PS C:\Users\User> docker run -d --name my_postgres -e POSTGRES_PASSWORD=postgres -v my_custom_volume:/my_volume postgres
e80c0ee019fa57b4eef130d4a11bf44ff186a9094b77311a446871554c20d816
PS C:\Users\User> docker exec -it my_postgres bash
root@e80c0ee019fa:/# cd /my_volume/
root@e80c0ee019fa:/my_volume# echo "Ulanbekova" > test.txt
root@e80c0ee019fa:/my_volume#
exit
PS C:\Users\User> docker exec -it my_nginx bash
root@139c1e505f02:/# cat /my_volume/test.txt
Ulanbekova
root@139c1e505f02:/#
```

Рисунок 2 – Передача данных через том

#### Задача 3: Создание сети overlay для распределенного приложения:

Используйте Docker Swarm или Kubernetes (в зависимости от предпочтений) для создания кластера. Создайте overlay-сеть и запустите несколько контейнеров, которые могут взаимодействовать через эту сеть.

```

PS C:\Users\User> docker swarm init
Error response from daemon: This node is already part of a swarm. Use "docker swarm leave" to leave this swarm and join another one.
PS C:\Users\User> docker network create -d overlay --attachable my_custom_network
otrjnqy2kpm1wyb2m0zb0z5ws
PS C:\Users\User> docker run -d --name my_postgres -e POSTGRES_PASSWORD=postgres --network my_custom_network postgres
e103759dd5c9b764b12468a0cadededf073ae9206fa6c3430736c59ac3608f6d
PS C:\Users\User> docker run -d --name my_nginx --network my_custom_network nginx
a4a955ab0649319494687223adf94fb50346d7a5fd9754f80a5024ca7823cc0

```

Рисунок 3 – Создание сети overlay

#### Задача 4: Связь контейнеров по IP-адресу:

Запустите два контейнера и присвойте им IP-адреса из одной пользовательской сети. Обеспечьте взаимодействие между контейнерами по их IP-адресам.

```

PS C:\Users\User> docker inspect -f '{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}' my_nginx
10.0.3.4
PS C:\Users\User> docker exec -it my_web_app sh
/ # ping 10.0.3.4
PING 10.0.3.4 (10.0.3.4): 56 data bytes
64 bytes from 10.0.3.4: seq=0 ttl=64 time=4.433 ms
64 bytes from 10.0.3.4: seq=1 ttl=64 time=0.076 ms
64 bytes from 10.0.3.4: seq=2 ttl=64 time=0.076 ms
64 bytes from 10.0.3.4: seq=3 ttl=64 time=0.101 ms
64 bytes from 10.0.3.4: seq=4 ttl=64 time=0.078 ms

```

Рисунок 4 - Связь контейнеров по IP-адресу

#### Задача 5: Использование ссылок для связи контейнеров:

Используя устаревшую опцию `--link`, создайте два контейнера (например, с Nginx и MySQL) и свяжите их между собой. Убедитесь, что контейнер с Nginx может успешно обращаться к контейнеру с MySQL через имя контейнера, указанное при использовании опции `--link`.

```

PS C:\Users\User> docker run -d --name my_mysql -e MYSQL_ROOT_PASSWORD=mysql mysql
5ff4f8f065eb067272f978f081b23d95ac5b56973e8f70b00263360e42e71b16
PS C:\Users\User> docker run -d --name my_nginx --link my_mysql:database_mysql nginx
11a8f53409374d95feade91c7df7f18deae2d25fb012bf8e7fa94607adfca5c6
PS C:\Users\User> docker ps -a
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS                               NAMES
11a8f5340937   nginx     "/docker-entrypoint...." 11 seconds ago Up 5 seconds   80/tcp                               my_nginx
5ff4f8f065eb   mysql     "docker-entrypoint.s..." About a minute ago Up About a minute   3306/tcp, 33060/tcp                my_mysql
PS C:\Users\User> docker exec -it my_nginx bash
root@11a8f5340937:/# apt-get update -y && apt-get install -y iputils-ping > /dev/null
Get:1 http://deb.debian.org/debian bookworm InRelease [151 kB]
Get:2 http://deb.debian.org/debian bookworm-updates InRelease [52.1 kB]
Get:3 http://deb.debian.org/debian-security bookworm-security InRelease [48.0 kB]
Get:4 http://deb.debian.org/debian bookworm/main amd64 Packages [8787 kB]
Get:5 http://deb.debian.org/debian bookworm-updates/main amd64 Packages [11.3 kB]
Get:6 http://deb.debian.org/debian-security bookworm-security/main amd64 Packages [128 kB]
Fetched 9177 kB in 10s (903 kB/s)
Reading package lists... Done
debconf: delaying package configuration, since apt-utils is not installed
root@11a8f5340937:/# ping database_mysql
PING database_mysql (172.17.0.2) 56(84) bytes of data.
64 bytes from database_mysql (172.17.0.2): icmp_seq=1 ttl=64 time=5.07 ms
64 bytes from database_mysql (172.17.0.2): icmp_seq=2 ttl=64 time=0.078 ms
64 bytes from database_mysql (172.17.0.2): icmp_seq=3 ttl=64 time=0.151 ms
64 bytes from database_mysql (172.17.0.2): icmp_seq=4 ttl=64 time=0.081 ms

```

Рисунок 5 – Связь контейнеров с помощи ссылки

#### Индивидуальное задание

Создать контейнер `my_mongo` с образом `mongolight`, таблицу в базе данных и несколько записей в ней. Сделать возможным доступ к созданной таблице в новом контейнере `my_mongo_2`, а также выполнить к ней запрос после удаления контейнера `my_mongo` и образа `mongolight`.

Задание выполнено при использовании тома `my_mongo_volume`. Данные mongo хранит в директории `/data/db`.

```
PS C:\Users\User> docker pull alinab/mongolight
Using default tag: latest
latest: Pulling from alinab/mongolight
2f12ed5a7535: Pull complete
be0a6512e13a: Pull complete
7d09b8656141: Pull complete
Digest: sha256:83d5c8de2c017483194d58e1cb18888acfd0749bf81a96f929754363fee5d21a
Status: Downloaded newer image for alinab/mongolight:latest
docker.io/alinab/mongolight:latest

What's Next?
 1. Sign in to your Docker account → docker login
 2. View a summary of image vulnerabilities and recommendations → docker scout quickview alinab/mongolight
PS C:\Users\User> docker run -d --name my_mongo -v my_mongo_volume:/data/db alinab/mongolight
59828e1776b3c5519d74708aa7f03bc305b1df9f497f5ecceda37854988e2c3c
PS C:\Users\User> docker run -it my_mongo mongo
Unable to find image 'my_mongo:latest' locally
docker: Error response from daemon: pull access denied for my_mongo, repository does not exist or may require 'docker login': denied: requested access to the resource is denied.
See 'docker run --help'.
PS C:\Users\User> docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                NAMES
59828e1776b3   alinab/mongolight  "/root/run.sh mongod"    About a minute ago    Up About a minute    27017/tcp, 28017/tcp    my_mongo
PS C:\Users\User> docker exec -it my_mongo mongo
MongoDB shell version: 3.2.10
connecting to: test
welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
  http://docs.mongodb.org/
Questions? Try the support group
  http://groups.google.com/group/mongodb-user
Server has startup warnings:
2023-12-21T11:48:43.298+0000 I CONTROL [initandlisten]
2023-12-21T11:48:43.298+0000 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/enabled is 'always'.
2023-12-21T11:48:43.298+0000 I CONTROL [initandlisten] **           We suggest setting it to 'never'
2023-12-21T11:48:43.298+0000 I CONTROL [initandlisten]
> use mydb
switched to db mydb
> db.createCollection("test")
{ "ok" : 1 }
> db.test.insertMany([ { lastname: "Ulanbekova" }, { lastname: "Voronkin" } ])
{
  "acknowledged" : true,
  "insertedIds" : [

```

Рисунок 6 – Работа с первым контейнером

```
PS C:\Users\User> docker stop my_mongo
my_mongo
PS C:\Users\User> docker rm my_mongo
my_mongo
PS C:\Users\User> docker rmi alinab/mongolight
Untagged: alinab/mongolight:latest
Untagged: alinab/mongolight@sha256:83d5c8de2c017483194d58e1cb18888acfd0749bf81a96f929754363fee5d21a
Deleted: sha256:ce632008396ba4aacf4837c13dfe6e9ae1a0e5b6ffbb5403e318a8239f22b0
Deleted: sha256:d6af8dee3678d7bc96b8557ad8a2d88194c5e1e390a1d1f3a88f2a01b7fdcadf
Deleted: sha256:6eb2193f99478fcc2a7f8149e47d6094a646706fc157343d1481f5f66cf69d57
Deleted: sha256:6f4ada5745cd4e3d641a47c1795aebad45449ee2131cc801b59271e25051fdb3
PS C:\Users\User> docker run -d --name my_mongo_2 -v my_mongo_volume:/data/db alinab/mongolight
latest: Pulling from alinab/mongolight
2f12ed5a7535: Pull complete
be0a6512e13a: Pull complete
7d09b8656141: Pull complete
Digest: sha256:83d5c8de2c017483194d58e1cb18888acfd0749bf81a96f929754363fee5d21a
Status: Downloaded newer image for alinab/mongolight:latest
603d9f8f19a0a0c0d5e575fdf8e5f2d00a9a076ac569efd914b70d49251173aa
PS C:\Users\User> docker exec -it my_mongo_2 mongo
MongoDB shell version: 3.2.10
connecting to: test
welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
  http://docs.mongodb.org/
Questions? Try the support group
  http://groups.google.com/group/mongodb-user
Server has startup warnings:
2023-12-21T11:51:44.355+0000 I CONTROL [initandlisten]
2023-12-21T11:51:44.355+0000 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/enabled is 'always'.
2023-12-21T11:51:44.356+0000 I CONTROL [initandlisten] **           We suggest setting it to 'never'
2023-12-21T11:51:44.356+0000 I CONTROL [initandlisten]
> use mydb
switched to db mydb
> db.test.find()
{ "_id" : ObjectId("6584268b07c54da6411d81d4"), "lastname" : "Ulanbekova" }
{ "_id" : ObjectId("6584268b07c54da6411d81d5"), "lastname" : "Voronkin" }
>
```

Рисунок 7 - Работа со вторым контейнером

## **Контрольные вопросы:**

### **1. Как создать новый том в Docker?**

`docker volume create`

### **2. Как удалить существующий том в Docker?**

`docker volume rm`

### **3. Как просмотреть список всех созданных томов в Docker?**

`docker volume ls`

### **4. Как создать том с определенным именем?**

`docker volume create my_volume`

### **5. Как присоединить том к контейнеру при его запуске?**

`docker run -v /путь/на/хосте:/путь/в/контейнере -d image_name`

### **6. Как просмотреть подробную информацию о конкретном томе в Docker?**

`docker volume inspect my_volume`

### **7. Как создать новую сеть в Docker?**

`docker network create my_custom_network`

### **8. Как удалить существующую сеть в Docker?**

`docker network rm my_custom_network`

### **9. Как просмотреть список всех созданных сетей в Docker?**

`docker network ls`

### **10. Как создать пользовательскую сеть с определенным именем?**

`docker network create my_custom_network`

### **11. Как присоединить контейнер к пользовательской сети при его запуске?**

`docker run --network=my_custom_network -d nginx`

### **12. Как просмотреть подробную информацию о конкретной сети в Docker?**

`docker network inspect my_network`

### **13. Как указать определенную сеть при запуске контейнера с использованием `docker run` ?**

```
docker run --network=my_custom_network -d nginx
```

**14. Какие сети будут доступны по умолчанию для контейнера, если не указана конкретная сеть?**

bridge, host и none.

**15. Как присоединить контейнер к нескольким сетям сразу при его запуске?**

```
docker run --network=my_custom_network -d nginx
```

**16. Как просмотреть список сетей, доступных на хосте Docker?**

```
docker network ls
```

**17. Как создать контейнер, подключенный к сети "bridge"?**

```
docker run --network=bridge -d nginx
```

**18. Как создать контейнер, подключенный к сети "host"?**

```
docker run --network=host -d nginx
```

**Вывод:** были изучены способы использования Docker для управления томами и сетями.