

Отчёт по практической работе №2

Основы работы с технологиями контейнеризации и ботами Telegram

Цель работы:

Создание сервера с постоянно работающим Telegram ботом

Ход работы:

Для начала откроем PowerShell и подключимся к серверу шлюзу и через него к рабочему серверу

```
PS C:\Users\user> ssh student@193.124.118.93
The authenticity of host '193.124.118.93 (193.124.118.93)' can't be established.
ED25519 key fingerprint is SHA256:9YfF0RJ043svji7MPRcbpG4Jp3k2f7tVmDEavUEQwYQ.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '193.124.118.93' (ED25519) to the list of known hosts.
student@193.124.118.93's password:
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.2.0-1015-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Mon Mar 10 13:26:01 2025 from 85.249.163.217
student@ruvds-x7i06:~$ ssh student@10.8.0.5
student@10.8.0.5's password:
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-131-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

12 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

New release '22.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Your Hardware Enablement Stack (HWE) is supported until April 2025.
*** System restart required ***
Last login: Mon Mar 10 16:26:10 2025 from 10.8.0.1
```

Рис 2.1

Создадим директорию по номеру зачётной книжки и перейдём в неё.

Установим Python необходимой версии и создадим окружение

```
student@user-IPMSB-H61:~$ mkdir 220803148 && cd 220803148
student@user-IPMSB-H61:~/220803148$ python3.10
Python 3.10.16 (main, Dec 4 2024, 08:53:37) [GCC 9.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> exit()
student@user-IPMSB-H61:~/220803148$ python3.10 -m venv env
student@user-IPMSB-H61:~/220803148$ source env/bin/activate
(env) student@user-IPMSB-H61:~/220803148$ pip install telepot==12.7
```

Рис 2.2.

Создадим bot.py и запишем в него код из методички параллельно создадим самого бота в телеграмм и получим токен, после чего запустим его.

```
(env) student@user-IPMSB-H61:~/220803148$ nano bot.py
(env) student@user-IPMSB-H61:~/220803148$ nano bot.py
(env) student@user-IPMSB-H61:~/220803148$ python bot.py
File "/home/student/220803148/bot.py", line 9
    bot.sendMessage(chat_id, 'Oks')
    ^
IndentationError: expected an indented block after 'if' statement on line 8
(env) student@user-IPMSB-H61:~/220803148$ nano bot.py
(env) student@user-IPMSB-H61:~/220803148$ python bot.py
Traceback (most recent call last):
  File "/home/student/220803148/bot.py", line 8, in <module>
    if command == '/command1':
NameError: name 'command' is not defined
(env) student@user-IPMSB-H61:~/220803148$ nano bot.py
(env) student@user-IPMSB-H61:~/220803148$ (env) student@user-IPMSB-H61:~/220803148$ python bot.py
I am listening ...
Got command: /start
From : 985955237
Got command: /command1
From : 985955237
Got command: /command2
From : 985955237
```

Рис 2.3.

Зайдём в телеграмм найдём бота и протестируем его.



Рис 2.4.

После того что мы увидим что бот работает создадим docker

```
(env) student@user-IPMSB-H61:~/220803148$ deactivate
deactivate: command not found
(env) student@user-IPMSB-H61:~/220803148$ deactivate
student@user-IPMSB-H61:~/220803148$ nano requirements.txt
student@user-IPMSB-H61:~/220803148$ nano Dockerfile
student@user-IPMSB-H61:~/220803148$ docker build -t 220803148
ERROR: docker: 'docker buildx build' requires 1 argument

Usage:  docker buildx build [OPTIONS] PATH | URL | -

Run 'docker buildx build --help' for more information
[+] Building 11.1s (13/13) FINISHED                                docker:default
=> [internal] load build definition from Dockerfile                0.7s
=> => transferring dockerfile: 205B                                  0.0s
=> [internal] load metadata for docker.io/library/python:3.10      2.3s
=> [internal] load metadata for docker.io/library/python:3.10-slim 2.2s
=> [internal] load -dockerignore                                    0.7s
=> => transferring context: 2B                                         0.0s
=> [builder 1/3] FROM docker.io/library/python:3.10@sha256:c79cd7b345644821bdee8cd8d8e314450aac30ea0f7000f7a7152 0.0s
=> [internal] load build context                                    0.7s
=> => transferring context: 505B                                       0.0s
=> [stage-1 1/4] FROM docker.io/library/python:3.10-slim@sha256:f000fc3f447306d9be2ae53dc7ab447fe9b33113af309125 0.0s
=> CACHED [stage-1 2/4] WORKDIR /code                               0.0s
=> CACHED [builder 2/3] COPY requirements.txt .                     0.0s
=> CACHED [builder 3/3] RUN pip install --user -r requirements.txt  0.0s
=> CACHED [stage-1 3/4] COPY --from=builder /root/.local /root/.local 0.0s
=> [stage-1 4/4] COPY ./bot.py .                                     1.7s
=> exporting to image                                              1.0s
=> => exporting layers                                                1.2s
=> => writing image sha256:f0b7f0daec3f003c5c77a4f0071a10048100f001c125f014a00f11104ac16a0 0.1s
=> => pushing to docker.io/library/220803148                         0.1s
```

Рис 2.5.

Запустим docker образ и получаем container ID

```
student@user-IPMSB-H61:~/220803148$ docker run -d --restart=always 220803148
5feab6795db1749ce619abb30793513244116f79fd5daa7ac0c43d9dd839acf3
student@user-IPMSB-H61:~/220803148$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED
5feab6795db1   220803148     "python -u ./bot.py"    About a minute ago
```

Рис 2.6.

Далее сохраним архив docker образа на ПК. Здесь удобно использовать команду `pwd` которая выведет нам путь в текущую папку.

```
student@ruvds-x7106:~$ ssh student@10.8.0.5
student@10.8.0.5's password:
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-131-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

12 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

New release '22.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Your Hardware Enablement Stack (HWE) is supported until April 2025.
*** System restart required ***
Last login: Mon Mar 10 21:43:55 2025 from 10.8.0.1
student@user-IPMSB-H61:~$ cd 220803148
student@user-IPMSB-H61:~/220803148$ ls
bot.py Dockerfile docker_image_220803148.tar env requirements.txt
student@user-IPMSB-H61:~/220803148$ pwd
/home/student/220803148
student@user-IPMSB-H61:~/220803148$ exit
logout
Connection to 10.8.0.5 closed.
student@ruvds-x7106:~$ scp student@10.8.0.5:/home/student/220803148/docker_image_220803148.tar .
student@10.8.0.5's password:
docker_image_220803148.tar                                100% 138MB  5.8MB/s  00:23
student@ruvds-x7106:~$ pwd
/home/student
student@ruvds-x7106:~$ exit
logout
Connection to 193.124.118.93 closed.
PS C:\Users\user> scp student@193.124.118.93:/home/student/docker_image_220803148.tar .
student@193.124.118.93's password:
docker_image_220803148.tar                                100% 138MB  1.2MB/s  02:00
PS C:\Users\user> ls
```

Рис 2.7.

Откроем архив для просмотра скачанных файлов

Имя	Размер	Сжат	Тип	Изменён	CRC32
..			Папка с файлами		
blobs	144 782 655	144 782 655	Папка с файлами	10.03.2025 21:24	
index.json	366	366	JSON File	01.01.1970 3:00	
manifest.json	2 296	2 296	JSON File	01.01.1970 3:00	
oci-layout	31	31	Файл	01.01.1970 3:00	
repositories	92	92	Файл	01.01.1970 3:00	

Рис 2.8.

Вывод:

В ходе выполненной работы мы создали постоянно работающего Телеграмм бота и сохранили его в docker container, также получили опыт взаимодействия с серверами через Power Shell.