

Департамент образования и науки города Москвы  
Государственное автономное образовательное учреждение  
высшего образования города Москвы  
«Московский городской педагогический университет»  
Институт цифрового образования  
Департамент информатики, управления и технологий

ДИСЦИПЛИНА:

Распределенные системы

**Лабораторная работа 5**

Выполнил: Бойко К. К., группа: АДЭУ-221

Преподаватель: Босенко Т.М.

Москва

2024

1.1. Обновить пакеты системы: `sudo apt-get update`

```
8 kB]
Get:39 http://security.ubuntu.com/ubuntu noble-security/main amd64 Components [7,240 B]
Get:40 http://security.ubuntu.com/ubuntu noble-security/main Icons (48x48) [11.3 kB]
Get:41 http://security.ubuntu.com/ubuntu noble-security/main Icons (64x64) [17.1 kB]
Get:42 http://security.ubuntu.com/ubuntu noble-security/main amd64 c-n-f Metadata [5,892 B]
Get:43 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Packages [394 kB]
Get:44 http://security.ubuntu.com/ubuntu noble-security/restricted Translation-en [75.7 kB]
Get:45 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Components [212 B]
Get:46 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 c-n-f Metadata [424 B]
Get:47 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Packages [557 kB]
Get:48 http://security.ubuntu.com/ubuntu noble-security/universe Translation-en [149 kB]
Get:49 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [51.9 kB]
Get:50 http://security.ubuntu.com/ubuntu noble-security/universe Icons (48x48) [47.0 kB]
Get:51 http://security.ubuntu.com/ubuntu noble-security/universe Icons (64x64) [73.4 kB]
Get:52 http://security.ubuntu.com/ubuntu noble-security/universe amd64 c-n-f Metadata [13.5 kB]
Get:53 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Packages [12.2 kB]
Get:54 http://security.ubuntu.com/ubuntu noble-security/multiverse Translation-en [2,940 B]
Get:55 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [212 B]
Get:56 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 c-n-f Metadata [356 B]
Get:57 https://dl.google.com/linux/chrome/deb stable/main amd64 Packages [1,224 B]
Fetched 5,539 kB in 5s (1,164 kB/s)
Reading package lists... Done
devops@devopsvm:~$
```

port MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

## 1.2. Установите Python 3 и необходимые библиотеки:

```
devops@devopsvm:~$ sudo apt-get install -y python3 python3-pip
sudo apt-get install -y python3-flask python3-cryptography
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
python3 is already the newest version (3.12.3-0ubuntu2).
The following packages will be upgraded:
  python3-pip python3-pip-whl
2 upgraded, 0 newly installed, 0 to remove and 232 not upgraded.
Need to get 3,020 kB of archives.
After this operation, 0 B of additional disk space will be used.
Get:1 http://ru.archive.ubuntu.com/ubuntu noble-updates/universe amd64 python3-pip all 24.0+dfsg-1ubuntu1.1 [1,317 kB]
Get:2 http://ru.archive.ubuntu.com/ubuntu noble-updates/universe amd64 python3-pip-whl all 24.0+dfsg-1ubuntu1.1 [1,703 kB]
Fetched 3,020 kB in 0s (6,841 kB/s)
(Reading database ... 194556 files and directories currently installed.)
Preparing to unpack .../python3-pip_24.0+dfsg-1ubuntu1.1_all.deb ...
Unpacking python3-pip (24.0+dfsg-1ubuntu1.1) over (24.0+dfsg-1ubuntu1) ...
Preparing to unpack .../python3-pip-whl_24.0+dfsg-1ubuntu1.1_all.deb ...
Unpacking python3-pip-whl (24.0+dfsg-1ubuntu1.1) over (24.0+dfsg-1ubuntu1) ...
Setting up python3-pip-whl (24.0+dfsg-1ubuntu1.1) ...
Setting up python3-pip (24.0+dfsg-1ubuntu1.1) ...
Processing triggers for man-db (2.12.0-4build2) ...
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
python3-flask is already the newest version (3.0.2-1ubuntu1).
python3-cryptography is already the newest version (41.0.7-4ubuntu0.1).
python3-cryptography set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 232 not upgraded.
devops@devopsvm:~$
```

port MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

## 1.3. Установите OpenSSL: sudo apt-get install -y openssl

```
devops@devopsvm:~$ sudo apt-get install -y openssl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
openssl is already the newest version (3.0.13-0ubuntu3.4).
openssl set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 232 not upgraded.
devops@devopsvm:~$
```

## Шаг 2. Создание сертификатов X.509 для аутентификации

2.1. Сгенерируйте корневой сертификат центра сертификации (CA): `openssl req -x509 -newkey rsa:4096 -keyout ca_key.pem -out ca_cert.pem -days 365 -nodes` При генерации сертификата CA будет предложено ввести некоторую информацию (страна, город и т.д.). Заполните её по вашему усмотрению.

```
devops@devopsvm:~$ openssl req -x509 -newkey rsa:4096 -keyout ca_key.pem -out ca_cert.pem -days 365 -nodes
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:RU
State or Province Name (full name) [Some-State]:Moscow oblast
Locality Name (eg, city) []:Moscow
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Boiko_adeu221
Organizational Unit Name (eg, section) []:mgpu
Common Name (e.g. server FQDN or YOUR name) []:Konstantin
Email Address []:kokabokin@gmail.com
devops@devopsvm:~$
```

Port MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

2.2. Создайте сертификаты для клиентских и серверных узлов: Аналогично, при генерации сертификатов для клиента и сервера будет предложено ввести некоторую информацию. Заполните её по вашему усмотрению.

```
Email Address []:kokabokin@gmail.com
devops@devopsvm:~$ ^C
devops@devopsvm:~$ openssl req -newkey rsa:4096 -keyout server_key.pem -out server_req.pem -nodes
openssl x509 -req -in server_req.pem -CA ca_cert.pem -CAkey ca_key.pem -
CAcreateserial -out server_cert.pem -days 365
openssl req -newkey rsa:4096 -keyout client_key.pem -out client_req.pem -nodes
openssl x509 -req -in client_req.pem -CA ca_cert.pem -CAkey ca_key.pem -
CAcreateserial -out client_cert.pem -days 365
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:RU
State or Province Name (full name) [Some-State]:Moscow oblast
Locality Name (eg, city) []:Moscow
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Boiko_adeu
Organizational Unit Name (eg, section) []:mgpu
Common Name (e.g. server FQDN or YOUR name) []:Konstantin
Email Address []:kokabokin@gmail.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:boiko
An optional company name []:Boiko
Certificate request self-signature ok
subject=C = RU, ST = Moscow oblast, L = Moscow, O = Boiko_adeu, OU = mgpu, CN = Konstantin, emailAddress = kokabokin@gmail.com
-----BEGIN CERTIFICATE-----
MIIFsJCCA5oCFwS0rehRJNSip9aT1tStU3wknU/MA0GCSqGSIb3DQEBCwUAMIGW
M0swCQYDVQ0GEwJSVTEwMBQGA1UECAwMTW9zY293IG91b6GFzdDEPM0GA1UEBwwG
-----
Port MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net
```

```

-----BEGIN CERTIFICATE-----
MIIFsjCCA5oCFuB0Jb34u48y3BhsAcc51T5UvLlMA0GCSqGSIb3DQEBCAUAMIGW
M0swCQYDVQ0GEwJSVTEwMB0GA1UECAwNTW9zY293IG91bGFzDDEPMA0GA1UEBnwG
TW9zY293MRYwFAYDVoQKDA1Cb21rb19hZGVV1Mj1xM00wCwYDVQ0LDartZ3B1MR1w
EQYDVQ0QDApLb25zdGFGdGUuMSwIwIAYJKoZIhvcNAQkBFhNrb2thYm9raW5AZ21h
aWwuuY29tMB4XD0tMT0MEwMTA2NDUu1N0tXD0tMT0MTwMTA2NDUu1N0wZMxZCAJBGNV
BAYTA1JlWmRYwFAYDVoQIDA1Nb3Njb3cgb2J5XN0M0q8wDQYDVQHDAZNB3Njb3c
EzARBGNVBA0McKjvaWtvX2FKZUxDTALBgNVBAsMBGincHxUeXARBGNVBAMMKctv
bnN0YV9w4w4xIjAgBgkqhkiG9w0BCQEWZ2tva2F1b2tpbkBnbWVpbC5jb2wggIi
MA0GCSqGSIb3DQEBAAUAAICDwAwggIKaoICAQCA3G1j0Cj9tqfj+6xmw00BMkRi
Pj3oBgmaAvZTDPHInghqWSSfTmzwk6d5ZmwhEvIIXq6oaPpkUij5LzmjPZGMSF
09cq6Uj9yitxRXvSWYqLD0H8VXjIA47xsGTTxw0s2JIdZeRsjoq5VnSYCS/6Mk+g
MwMLE+3/vJxPNZqAK548KhqMRHYJgijw44Yz25B0YhroFen58Gf0ut29GtBxTqy
4R2eKJse0FT/k048wshKjnwKQ0zMRQWZ5eIQZIsrqgKoVX3M8cNcmq/jNmQv4/
nKhJMNJeh7bncE+dHTEXSfjNirItv+4LMPZ1GzvFSkaaEZR3Smjfu62MocCM8Jy
Rw0aaAeh++3YFX7IMumCj4j9AmL+qKkeblX0veE1XN10Hkjq1dWoTN2hbsM++C8
Pr4514C4Gj2VwEqEJ/LfLEgbMYqIw98N0gfzItSQK6AEj9GJDdhptIa8IRdKI
E5YfJ1ry0tTHqPu/0S6gaF4grdgzL2ye/1eKG0/LFF5iF2mFY3avDfh9BCRIDxq
DCqr4SP3d06/zRPHd9auFe314wBxD3INmtvcKJkK7H/0aA4LEvmRXK5EwLH2az1C
x/ZC/NZ1Agd0S5GoMCs0Ld5emZ8WR4Tz49p5C3hZL/0aJqYpnWuXRj1Xkf2AHkHM
ogVLrIt5xfsgj0go0IDA0BMA0GCSqGSIb3DQEBCAUAAICACXFA6B7IJV4/Pf
L9LY+puGdKs7qW76uPKS6MHIwXgc6bmdpLlbA9AHZRJ7ED53Sghc0Zyt9e2f9Pwt
SyYuBX0j90RfoIoEvFXL59hgYKIrt0Rq1khiCLwt72Cv0ByeR7RcyIw8Adg1NcLw
30tLwWcLU9zuJ4Tz4u9BjFdwBWLyaI0975Rad0Xfw9Q91ncZCIAGf/PoAY7n3WC
8tXDwvzcVw+u0b0bkZzWZgKsjZPi4/+CJtdq3Mw0FMECwBdGexPaIqdKn084m
jdt8eVukoP+7THdXrWfaxXcuFFTZ7Z0kpKkjZuS1EKq9o3yc8ue3RtSn3ExgbFU
326PqbhaHAlVtLnXFtYFPKS0GtYf0fsf+rAXmLYtLTkbqZRmnA0g0Cb46Hq+mp
YiNZ+c77mFwjpSg0Dcps+Ca7eC2HCnGvdytmUhl/tN7mH+HC+xcIKJSSZdr1b3UJe
eFVHDGZGyERwK+0xKNwCCLZ03xJNcMBJ83X5CB087Yp060GvB9qCwNAk6HTTWoR
Ds6WuWIo7YxyQBAi8ZsEWXth0MqbPVDzHV/4nf0JG1o37z8Bf3PH0T2ZS5L8bDa
RP25eb04EwxYoPZCLWL1wDp85NKZ2E1a1R+6M1yn65gzT1nsMR6zRT1z3dvnUJ
uFpi0LLfwi1LhVhKvKpDUEc0WkCgQ==
-----END CERTIFICATE-----
CAcreateserial: command not found
devops@devopsvm:~$ ^C
devops@devopsvm:~$ openssl x509 -req -in server_req.pem -CA ca_cert.pem -CAkey ca_key.pem -CAcreateserial -out server_cert.pem -days 365
Certificate request self-signature ok
subject=C = RU, ST = Moscow oblast, L = Moscow, O = Boiko adeu, OU = mgpu, CN = Konstantin, emailAddress = kokabokin@gmail.com
devops@devopsvm:~$ openssl x509 -req -in client_req.pem -CA ca_cert.pem -CAkey ca_key.pem -CAcreateserial -out client_cert.pem -days 365
Certificate request self-signature ok
subject=C = RU, ST = Moscow oblast, L = Moscow, O = Boiko_adeu, OU = mgpu, CN = Konstantin, emailAddress = kokabokin@gmail.com
devops@devopsvm:~$

```

### 3.2. Создайте файл client.py для реализации клиентской части:

#### Шаг 4. Создайте client.py для шифрования данных:

```
GNU nano 7.2 client1.py
import requests
import ssl
from cryptography.x509 import load_pem_x509_certificate
from cryptography.hazmat.backends import default_backend

def make_request():
    # Prepare the data to be sent in the request
    data = {
        "certificate": open('client_cert.pem', 'r').read(),
        "data": "some_data"
    }

    # Set up verify and cert parameters for SSL/TLS
    s = requests.Session()
    s.verify = 'ca_cert.pem' # Path to the CA certificate for verifying the server
    s.cert = ('client_cert.pem', 'client_key.pem') # Client certificate and key

    try:
        # Make the HTTPS request without 'context'
        response = s.post('https://localhost:8000/api/data', json=data)

        if response.status_code == 200:
            print(response.json())
        else:
            print(f"Error: {response.status_code} - {response.text}")

    except requests.exceptions.SSLError as ssl_error:
        print(f"SSL error: {ssl_error}")
    except requests.exceptions.RequestException as req_error:
        print(f"Request error: {req_error}")

if __name__ == '__main__':
    make_request()
```

#### Шаг 5. Настройка отказоустойчивости 5.1. Создайте файл coordinator.py для реализации координатора распределенной системы:

```
GNU nano 7.2 coordinator.py *
from flask import Flask, request
import requests

app = Flask(__name__)

# List of server URLs to forward requests to
server_urls = ['https://127.0.0.1:5000', 'https://10.193.21.172:5000'] # Update with your server IPs

@app.route('/api/data', methods=['POST'])
def handle_request():
    data = request.get_json() # Get the JSON data from the incoming request

    for url in server_urls:
        try:
            # Forward the request to the server
            response = requests.post(f"{url}/api/data", json=data, verify=False) # Set verify=False for testing purposes

            if response.status_code == 200:
                return response.json() # Return the response from the server if successful
        except requests.exceptions.RequestException as e:
            print(f"Error connecting to {url}: {e}") # Log the error for debugging

    return {'error': 'All servers are down'}, 503 # Return an error if all servers fail

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=8000) # Run the Flask app on all interfaces
```

Создадим самоподписанные сертификаты. Откройте терминал и перейдите в директорию, где будут находиться ваши файлы: `cd /home/devops/Downloads/lw_05` Сгенерируйте приватный ключ для

сервера: `openssl genrsa -out server_key.pem 2048` Эта команда создаст файл `server_key.pem` с  
приватным ключом длиной 2048 бит. Создайте сертификат X.509 для сервера, используя  
сгенерированный ключ: `openssl req -new -x509 -key server_key.pem -out server_cert.pem -days 365`  
Команда создаст самоподписанный сертификат `server_cert.pem`, действительный в течение 365 дней.  
Во время выполнения команды вас попросят ввести некоторую информацию о сертификате, такую  
как страна, название организации и т.д. После выполнения этих двух команд в рабочей директории  
`/home/devops/Downloads/lw_05` должны появиться два новых файла: `server_key.pem` `server_cert.pem`

```
devops@devopsvm:~$ cd /home/devops/Downloads/lw_05
-bash: cd: /home/devops/Downloads/lw_05: No such file or directory
devops@devopsvm:~$ mkdir cd /home/devops/Downloads/lw_05
devops@devopsvm:~$ cd /home/devops/Downloads/lw_05
devops@devopsvm:~/Downloads/lw_05$ openssl genrsa -out server_key.pem 2048
devops@devopsvm:~/Downloads/lw_05$ openssl req -new -x509 -key server_key.pem -out server_cert.pem -days 365
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:RU
State or Province Name (full name) [Some-State]:Moscow oblast
Locality Name (eg, city) []:Moscow
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Boiko_adeu
Organizational Unit Name (eg, section) []:mgpu
Common Name (e.g. server FQDN or YOUR name) []:Konstantin
Email Address []:kokabokin@gmail.com
devops@devopsvm:~/Downloads/lw_05$ ls -a
.  ..  server_cert.pem  server_key.pem
devops@devopsvm:~/Downloads/lw_05$
```

Сгенерируйте приватный ключ для центра сертификации: `openssl genrsa -out ca_key.pem 2048`  
Команда создаст файл `ca_key.pem` с приватным ключом центра сертификации длиной 2048 бит.  
Создайте самоподписанный сертификат центра сертификации: `openssl req -new -x509 -key  
ca_key.pem -out ca_cert.pem -days 3650`

```
devops@devopsvm:~$ cd /home/devops/Downloads/lw_05
devops@devopsvm:~/Downloads/lw_05$ openssl genrsa -out server_key.pem 2048
devops@devopsvm:~/Downloads/lw_05$ openssl req -new -x509 -key server_key.pem -out server_cert.pem -days 365
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:RU
State or Province Name (full name) [Some-State]:Moscow oblast
Locality Name (eg, city) []:Moscow
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Boiko_adeu
Organizational Unit Name (eg, section) []:mgpu
Common Name (e.g. server FQDN or YOUR name) []:Konstantin
Email Address []:kokabokin@gmail.com
devops@devopsvm:~/Downloads/lw_05$ ls -a
.  ..  server_cert.pem  server_key.pem
devops@devopsvm:~/Downloads/lw_05$ openssl genrsa -out ca_key.pem 2048
devops@devopsvm:~/Downloads/lw_05$ openssl req -new -x509 -key ca_key.pem -out ca_cert.pem -days 3650
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:RU
State or Province Name (full name) [Some-State]:Moscow oblast
Locality Name (eg, city) []:Moscow
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Boiko_adeu
Organizational Unit Name (eg, section) []:mgpu
Common Name (e.g. server FQDN or YOUR name) []:Konstantin
Email Address []:kokabokin@gmail.com
devops@devopsvm:~/Downloads/lw_05$
```

5.2. Запустить `server.py` на нескольких машинах (или на одном при тестировании) для обеспечения  
отказоустойчивости.



```
devops@devopsvm:~$ python3 server.py
Зашифрованные данные для хранения: b'\xeb\xe2\xa7L\xff\xe3\xf8\x15\xaa\xe6\xbe-i\x9e\xb7\x9b\x0f\xd7\x88\xe9\xf0\x17\xf0\x0bc\xbcA'\xd4\xca\x9191i\x99?\x87;\xe0L\xb2\x98\x00\xc92\x8e-&'
Сервер готов к приему данных...
```

### 5.3. Запустить coordinator.py на отдельной машине.

```
devops@devopsvm:~$ python3 coordinator.py
* Serving Flask app 'coordinator'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:8000
* Running on http://10.193.21.172:8000
Press CTRL+C to quit
```

Теперь server\_urls содержит адреса ваших реальных серверов, запущенных на 127.0.0.1:5000 и 192.168.0.137:5000. Ваш прокси-сервер будет пытаться отправить запросы на эти адреса, пока не получит успешный ответ. Обратите внимание, что ваш прокси-сервер будет работать на адресе 0.0.0.0:8000, а не на тех же адресах, что и реальные серверы. Это позволит клиентам обращаться к вашему прокси-серверу, а он, в свою очередь, будет перенаправлять запросы на соответствующие реальные серверы.

5.4. Клиенты будут отправлять запросы на coordinator.py, который будет распределять их между работающими серверами.

3. Запустите client.py и убедитесь, что клиент успешно аутентифицируется, данные шифруются и отправляются на сервер.

```
devops@devopsvm:~/Downloads/lw_05$ python3 client1.py
SSL error: HTTPSConnectionPool(host='localhost', port=8000): Max retries exceeded with url: /api/data (Caused by SSLError(SSLError(1, '[SSL: WRONG_VERSION_NUMBER] wrong version number (_ssl.c:1000)'))))
devops@devopsvm:~/Downloads/lw_05$
```

Подключение через http:

```
devops@devopsvm:~/Downloads/lw_05$ nano client2.py
devops@devopsvm:~/Downloads/lw_05$ python3 client2.py
Ошибка: 503 - {"error": "All servers are down"}
devops@devopsvm:~/Downloads/lw_05$
```

```
warnings.warn(
Error connecting to https://10.193.21.172:5000: HTTPSConnectionPool(host='10.193.21.172', port=5000): Max retries exceeded with url: /api/data (Caused by SSLError(SSLEOFError(8, 'EOF occurred in violation of protocol (_ssl.c:2406)'))))
127.0.0.1 - - [01/Nov/2024 11:44:59] "POST /api/data HTTP/1.1" 503 -
```

Дерево проекта:

```
Fetches 47.1 KB in 0s (440 KB/s)
Selecting previously unselected package tree.
(Reading database ... 194556 files and directories currently installed.)
Preparing to unpack .../tree_2.1.1-2ubuntu3_amd64.deb ...
Unpacking tree (2.1.1-2ubuntu3) ...
Setting up tree (2.1.1-2ubuntu3) ...
Processing triggers for man-db (2.12.0-4build2) ...
devops@devopsvm:~/Downloads/lw_05$ tree
.
├── ca_cert.pem
├── ca_key.pem
├── client1.py
├── client2.py
├── client_cert.pem
├── client_key.pem
├── coordinator.py
├── server1.py
├── server_cert.pem
└── server_key.pem

1 directory, 10 files
devops@devopsvm:~/Downloads/lw_05$
```

port MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

## Индивидуальное задание 25 вариант

Реализация механизма динамического масштабирования

- Добавьте в coordinator.py возможность динамически запускать и останавливать серверные экземпляры в зависимости от нагрузки.
- Обновите server.py, чтобы он мог регистрироваться и отключаться от координатора.

## Тестирование регистрации

```
devops@devopsvm:~/Downloads/lw_05$ curl -X POST http://localhost:8000/api/data -H "Content-Type: application/json" -d '{"data": "test data"}'
```

Press CTRL+C to quit

```
127.0.0.1 - - [01/Nov/2024 12:15:24] "POST /register HTTP/1.1" 200 -
Ошибка подключения к http://localhost:5000: HTTPConnectionPool(host='localhost', port=5000): Max retries exceeded with url: /api/data (Caused by NewConnectionError(<curl>lib3_connection.HTTPConnection object at 0x793d1078c1a0s: 'Failed to establish a new connection: [Errno 111] Connection refused'))
```









