

DevOps Exercises

Session 2: Managing Flask Applications with systemd

Student Name: Mehdi Daneshvar

Date: August 15, 2024

Running Multiple Instances of a Flask Application with systemd

/opt/myapp/app.py

```
from flask import Flask, jsonify
import sys
app=Flask(__name__)
@app.route('/api/hello')
def hello():
    response={'message': 'Hello, this is your Flask API!'}
    return jsonify(response)
if __name__ == '__main__':
    port=int(sys.argv[1] if len(sys.argv)>1 else 5000)
    app.run(host='0.0.0.0',port=port)
```

This file defines a basic Flask application that listens on a specified port and responds with a JSON message.

/etc/systemd/system/myapp@.service

```
[Unit]
Description=my flask app
After=network.target

[Service]
Restart=always
User=myapp
Group=myapp
WorkingDirectory=/opt/myapp
ExecStart=/usr/bin/python3 app.py %i

[Install]
WantedBy=default.target
```

This systemd unit file allows running multiple instances of the Flask application. `%i` in `ExecStart` is a placeholder for the instance identifier, typically used to pass the port number.

Steps to Run Multiple Instances

1. Create the `myapp` User

Ensure the `myapp` user exists to run the service securely:

```
sudo useradd -r -s /bin/false myapp
```

2. Start Services

Run multiple instances of the application by specifying different ports:

```
systemctl start myapp@8080
systemctl start myapp@8081
systemctl start myapp@8082
systemctl start myapp@8083
```

3. Enable Services at Boot

Ensure the services start automatically after a reboot:

```
systemctl enable myapp@8080
systemctl enable myapp@8081
systemctl enable myapp@8082
systemctl enable myapp@8083
```

4. Check Service Status

Verify that the services are running properly:

```
systemctl status myapp@8080
systemctl status myapp@8081
systemctl status myapp@8082
systemctl status myapp@8083
```

Example output:

```
root@linux:/opt# systemctl start myapp@8080
root@linux:/opt# systemctl status myapp@8080
● myapp@8080.service - my flask app
   Loaded: loaded (/etc/systemd/system/myapp@.service; disabled; vendor preset: enabled)
   Active: active (running) since Mon 2024-12-09 17:37:36 UTC; 2min 19s ago
     Main PID: 8216 (python3)
       Tasks: 1 (limit: 4522)
      Memory: 18.1M
     CGroup: /system.slice/system-myapp.slice/myapp@8080.service
            └─8216 /usr/bin/python3 app.py 8080

Dec 09 17:37:36 linux systemd[1]: Started my flask app.
Dec 09 17:37:36 linux python3[8216]: * Serving Flask app 'app'
Dec 09 17:37:36 linux python3[8216]: * Debug mode: off
Dec 09 17:37:36 linux python3[8216]: WARNING: This is a development server. Do not use it
Dec 09 17:37:36 linux python3[8216]: * Running on all addresses (0.0.0.0)
Dec 09 17:37:36 linux python3[8216]: * Running on http://127.0.0.1:8080
Dec 09 17:37:36 linux python3[8216]: * Running on http://192.168.40.20:8080
Dec 09 17:37:36 linux python3[8216]: Press CTRL+C to quit
Dec 09 17:38:19 linux python3[8216]: 192.168.60.3 - - [09/Dec/2024 17:38:19] "GET /api/he
lines 1-18/18 (END)

root@linux:/opt#

root@linux:~# systemctl start myapp@8082
root@linux:~# systemctl status myapp@8082
● myapp@8082.service - my flask app
   Loaded: loaded (/etc/systemd/system/myapp@.service; disabled; vendor preset: enabled)
   Active: active (running) since Mon 2024-12-09 17:37:42 UTC; 3min 53s ago
     Main PID: 8236 (python3)
       Tasks: 1 (limit: 4522)
      Memory: 18.1M
     CGroup: /system.slice/system-myapp.slice/myapp@8082.service
            └─8236 /usr/bin/python3 app.py 8082

Dec 09 17:37:42 linux systemd[1]: Started my flask app.
Dec 09 17:37:42 linux python3[8236]: * Serving Flask app 'app'
Dec 09 17:37:42 linux python3[8236]: * Debug mode: off
Dec 09 17:37:42 linux python3[8236]: WARNING: This is a development server. Do not use it
Dec 09 17:37:42 linux python3[8236]: * Running on all addresses (0.0.0.0)
Dec 09 17:37:42 linux python3[8236]: * Running on http://127.0.0.1:8082
Dec 09 17:37:42 linux python3[8236]: * Running on http://192.168.40.20:8082
Dec 09 17:37:42 linux python3[8236]: Press CTRL+C to quit
Dec 09 17:38:27 linux python3[8236]: 192.168.60.3 - - [09/Dec/2024 17:38:27] "GET /api/he
lines 1-18/18 (END)

root@linux:~# systemctl start myapp@8081
root@linux:~# systemctl status myapp@8081
● myapp@8081.service - my flask app
   Loaded: loaded (/etc/systemd/system/myapp@.service; disabled; vendor preset: enabled)
   Active: active (running) since Mon 2024-12-09 17:37:38 UTC; 4min 58s ago
     Main PID: 8232 (python3)
       Tasks: 1 (limit: 4522)
      Memory: 18.1M
     CGroup: /system.slice/system-myapp.slice/myapp@8081.service
            └─8232 /usr/bin/python3 app.py 8081

Dec 09 17:37:38 linux systemd[1]: Started my flask app.
Dec 09 17:37:39 linux python3[8232]: * Serving Flask app 'app'
Dec 09 17:37:39 linux python3[8232]: * Debug mode: off
Dec 09 17:37:39 linux python3[8232]: WARNING: This is a development server. Do not use it
Dec 09 17:37:39 linux python3[8232]: * Running on all addresses (0.0.0.0)
Dec 09 17:37:39 linux python3[8232]: * Running on http://127.0.0.1:8081
Dec 09 17:37:39 linux python3[8232]: * Running on http://192.168.40.20:8081
Dec 09 17:37:39 linux python3[8232]: Press CTRL+C to quit
Dec 09 17:38:23 linux python3[8232]: 192.168.60.3 - - [09/Dec/2024 17:38:23] "GET /api/he
lines 1-18/18 (END)

root@linux:~# systemctl start myapp@8083
root@linux:~# systemctl status myapp@8083
● myapp@8083.service - my flask app
   Loaded: loaded (/etc/systemd/system/myapp@.service; disabled; vendor preset: enabled)
   Active: active (running) since Mon 2024-12-09 17:41:57 UTC; 7s ago
     Main PID: 8353 (python3)
       Tasks: 1 (limit: 4522)
      Memory: 18.0M
     CGroup: /system.slice/system-myapp.slice/myapp@8083.service
            └─8353 /usr/bin/python3 app.py 8083

Dec 09 17:41:57 linux systemd[1]: Started my flask app.
Dec 09 17:41:57 linux python3[8353]: * Serving Flask app 'app'
Dec 09 17:41:57 linux python3[8353]: * Debug mode: off
Dec 09 17:41:57 linux python3[8353]: WARNING: This is a development server. Do not use it
Dec 09 17:41:57 linux python3[8353]: * Running on all addresses (0.0.0.0)
Dec 09 17:41:57 linux python3[8353]: * Running on http://127.0.0.1:8083
Dec 09 17:41:57 linux python3[8353]: * Running on http://192.168.40.20:8083
Dec 09 17:41:57 linux python3[8353]: Press CTRL+C to quit
lines 1-17/17 (END)
```

Verify Running Services

- Use **top** to check running processes under the **myapp** user:

```
top -u myapp
```

Example output:

```
mehdi — mehdi@linux: ~ — ssh mehdi@192.168.40.20 — 80x24
top - 17:46:24 up 1:03, 2 users, load average: 0.00, 0.00, 0.00
Tasks: 129 total, 1 running, 128 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3859.8 total, 2351.8 free, 471.6 used, 1036.5 buff/cache
MiB Swap: 3882.0 total, 3882.0 free, 0.0 used. 3131.3 avail Mem

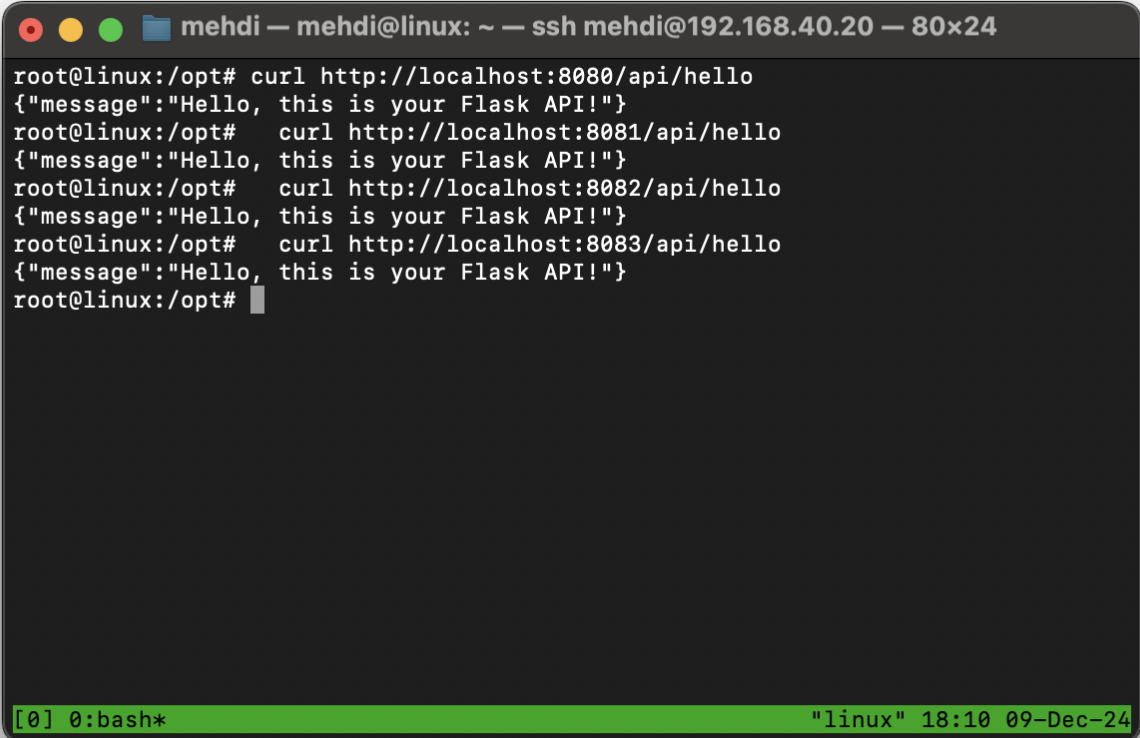
  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM    TIME+  COMMAND
  8216 myapp    20   0 108388  27452  9692  S   0.0   0.7   0:00.50 python3
  8232 myapp    20   0 108388  27656  9792  S   0.0   0.7   0:00.48 python3
  8236 myapp    20   0 108388  27528  9760  S   0.0   0.7   0:00.46 python3
  8353 myapp    20   0 108388  27672  9808  S   0.0   0.7   0:00.46 python3

[0] 0:top*                                     "linux" 17:46 09-Dec-24
```

- Test the API for each instance:

```
curl http://localhost:8080/api/hello
curl http://localhost:8081/api/hello
curl http://localhost:8082/api/hello
curl http://localhost:8083/api/hello
```

Example output:



```
mehdi — mehdi@linux: ~ — ssh mehdi@192.168.40.20 — 80x24
root@linux:/opt# curl http://localhost:8080/api/hello
{"message":"Hello, this is your Flask API!"}
root@linux:/opt# curl http://localhost:8081/api/hello
{"message":"Hello, this is your Flask API!"}
root@linux:/opt# curl http://localhost:8082/api/hello
{"message":"Hello, this is your Flask API!"}
root@linux:/opt# curl http://localhost:8083/api/hello
{"message":"Hello, this is your Flask API!"}
root@linux:/opt#
```

[0] 0: bash* "linux" 18:10 09-Dec-24