Exercise 1

Hardware

- Processor: AMD Ryzen 7 4800H with Radeon Graphics (2.90 GHz)
- RAM: 8,00 GB (7,42 GB usable)
- Disk: 1TB SSD

Operating System

- OS Name: Microsoft Windows 11 Home Single Language
- OS Version: 10.0.26100 N/A Build 26100
- OS Manufacturer: Microsoft Corporation
- OS Configuration: Standalone Workstation
- OS Build Type: Multiprocessor Free

Docker and Docker Compose versions

- Docker version 28.4.0
- Docker Compose version v2.39.2-desktop.1

Diagram

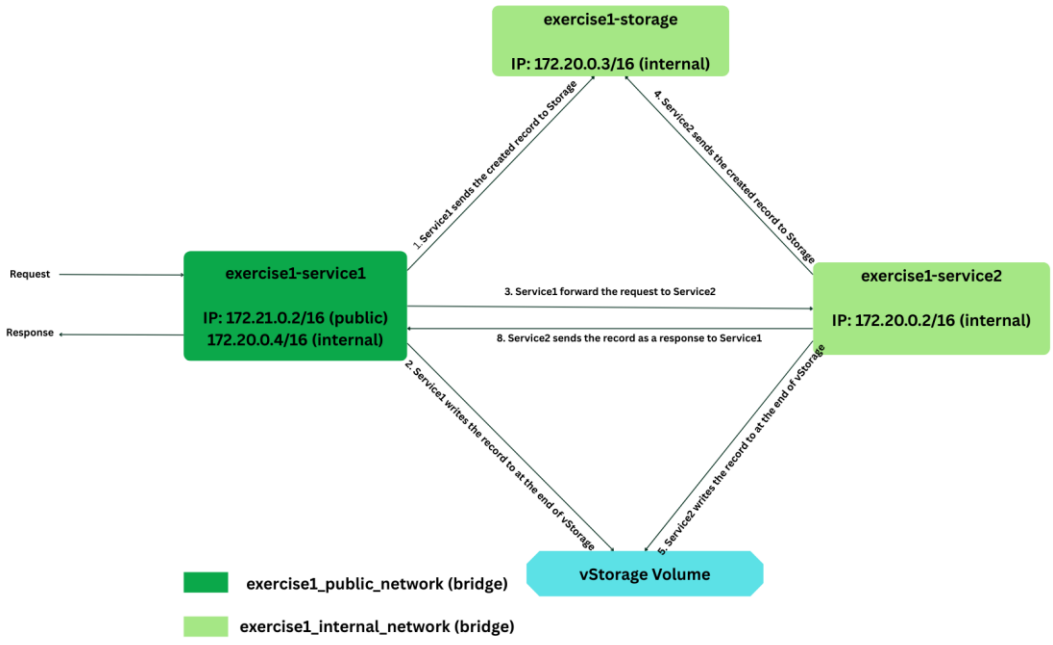

Fig: A diagram showing the services, network and storage

Analysis status records

URL: curl http://localhost:8199/log

Response:

```
Timestamp1: uptime 6.61 hours, free disk in root: 968149 Mbytes
Timestamp2: uptime 6.61 hours, free disk in root: 968149 Mbytes
Timestamp1: uptime 6.61 hours, free disk in root: 968149 Mbytes
Timestamp2: uptime 6.61 hours, free disk in root: 968149 Mbytes
```

a) Uptime:

- Reads **/proc/uptime** from the container.
- Get system uptime in seconds, then converts seconds to hours.
- Correctly measuring container uptime if Docker running. However, this uptime does not reflect total system uptime outside the container.

b) Disc space:

- Get free space of the root of the container's filesystem (/), which is usually a virtual overlay filesystem managed by Docker, then converts bytes to Mbytes.
- For better result, I could mount a host path as a volume, then check free space of the volume.

Analysis persistent storage

Feature	Host-mounted directory (./vStorage:/app/vStorage)	Named Docker volume (logs:/app/Storage/logs)
Ease of inspection	Yes, directly on host	No, need to use Docker commands
Persistence	Yes, host device	Yes, survives container removal
Security	Lower, file can be modified easily	Higher, isolated volume
Usage	Development	Production

Instructions for cleaning up storage

URL: **curl -X 'DELETE' 'http://localhost:8199/deleteLog'**

Response:

```
true
```

Hitting this endpoint deletes both:
./vStorage:/app/vStorage (host-mounted directory)
logs:/app/Storage/logs (named Docker volume)

What was difficult?

- Start the Service1 API and listen for HTTP requests on port 8199 on all available network interfaces (0.0.0.0).
- /proc/uptime exists only on Linux. Docker Desktop on Windows requires additional WSL ubuntu distro and could not be tested without using Docker.

Main problems and Solution

- **No response from Service1**: solved by binding ASP.NET Kestrel to 0.0.0.0 (webBuilder.UseUrls("http://0.0.0.0:8199")) in program.cs file.
- **C# JSON handling**: dynamic vs type (bool, string) mismatch, solved by using ToObject<T>().
- **Storage volume accessibility**: chose host bind mount for inspecting vStorage file easily.

Instructions

git clone-b exercise1 https://github.com/Samith-Pantho/DevOps-Course.git

cd DevOps-Course

cd exercise1

docker compose up-d –build

curl http://localhost:8199/status **(For checking Service1 and Service2 status)**

curl http://localhost:8199/log **(For checking Storage service data)**

curl-X 'DELETE' 'http://localhost:8199/deleteLog' **(For removing both Storage service data and vStorage data)**

docker-compose down