# Practical work 4

https://github.com/heig-vd-dai-course

Markdown · PDF

L. Delafontaine and H. Louis, with the help of GitHub Copilot.

# Table of contents

# Introduction

Web applications are accessible from anywhere in the world. You can access them from your computer, your smartphone, your tablet, etc. using a web browser, mobile application or even a command line tool.

In this practical work, you will create a web application that uses the HTTP protocol.

The web application will be defined by an application protocol interface (API) and will be deployed on a virtual machine and accessible from the Internet using a domain name.

You can interact with the web application using your web browser and/or a command line tool such as curl or you can develop your own application client.

The API will be defined by you.

Feel free to be creative and to create a web application that you would like to use. It can be a web application that you would like to use in your daily life or a web application that you would like to use for your studies. For example, you can choose to create a web application that allows you to manage your shopping list, a web application that allows you to manage your tasks, a web application that allows you to manage your notes, etc. If you do not have any idea, come to see us and we can give you.

Multiple groups can choose the same application protocol and you can share your methodology but please do not copy/paste code from other groups.

# Objectives

- Dialog with the IT department of the HEIG-VD to get a virtual machine on our cloud infrastructure
- Access the virtual machine with SSH
- Install Docker and Docker Compose on the virtual machine
- Define some Docker Compose files to run the web application with a reverse proxy (Traefik)
- Deploy the simple CRUD API on the virtual machine
- Access the CRUD API from a domain name

# Group composition

As the end of the semester is approaching, we know you have a lot of work to do for other courses.

As this practical work is more complex than the previous ones, you can work in groups between two and four students. You can choose your partners. If you do not have a partner or a group, we will assign you one.

To announce your group, create a new GitHub Discussion at https://github.com/orgs/heig-vd-dai-course/discussions with the following information:

- **Title**: DAI 2023-2024 - Practical work 4 - First name Last name member 1, First name Last name member 2, First name Last name member 3 and First name Last name member 4
- **Category**: Show and tell
- **Description**: A quick description of what you will achieve during this practical work

The teaching staff might ask you to change the scope of your practical work if it is too complex or too simple.

**Please do it a soon as possible, even if you do not have a clear idea yet as it will help us to plan the practical work review.**

# Guidelines

In order for this practical work to be successful, you must follow the following steps:

1. Obtain a virtual machine
2. Install and configure the virtual machine
3. Obtain a domain name
4. Run the secure Traefik and whoami examples
5. Develop the web application

Steps 1, 3 can be done in parallel before steps 2 and 4. Step 5 can be started right away.

## Obtain a virtual machine

### Order the virtual machine

You must send an email to the IT department of the HEIG-VD to get a virtual machine on our cloud infrastructure.

Define a person in the group that will be the contact person for the virtual machine. This person will be responsible for the virtual machine and will be contacted by the IT department if there is a problem with the virtual machine.

The contact email is: mehdi.salhi@hes-so.ch.

You have a template email below. Please take some time to understand what information you need to provide and why.

Replace the `<Prénom Nom de l'étudiant-e>` and `<Date de début de mise en service>` placeholders with the correct information.

For the `<Date de début de mise en service>` placeholder, set it for the end of the week. The IT department will need some time to create the virtual machine.

For the `<Date de fin de service>` placeholder, set it for the end of the semester. The virtual machine will be deleted at the end of the semester.

**Please do it as soon as possible!**

```
Sujet : [DAI] Machine virtuelle pour le cours DAI 2023-2024

Bonjour,

Je souhaite obtenir une machine virtuelle avec les spécifications suivantes dans le cadre
du cours DAI :

- Personne de contact / responsable : <Prénom Nom de l'étudiant-e>
- Date de mise en service souhaitée : <Date de début de mise en service>
- Date de fin de service souhaitée : <Date de fin de service>
- OS / distribution : Ubuntu 22.04
- CPU : 1vCPU
- RAM : 2GB
- Stockage : 50GB
- Ports :
  - Réseau interne (avec le VPN) : 22 (SSH)
  - Réseau externe (depuis Internet) : 80 (HTTP), 443 (HTTPS)

Merci d'avance et bonne journée,
<Prénom Nom de l'étudiant-e>
```

## Wait for the virtual machine

The IT department will send you an email with the information to access the virtual machine with the following information:

- The internal IP address of the virtual machine (to access it with SSH)
- The external IP address of the virtual machine (to access it from the Internet)
- The username to access the virtual machine
- The password to access the virtual machine

# Install and configure the virtual machine

## Access the virtual machine

Once the virtual machine is created, you can access it using its internal IP with SSH as seen in the SSH and SCP chapter.

You will have to be connected to the VPN. It is mandatory.

## Update the virtual machine and install apache2-utils

Once you have access to the virtual machine, you can update it with the following commands:

```
# Update the list of packages
sudo apt update

# Upgrade the packages
sudo apt upgrade
```

This will ensure that the virtual machine is up-to-date.

Install apache2-utils with the following command:

```
# Install apache2-utils
sudo apt install apache2-utils
```

This will ensure that you can use the htpasswd command line tool to create users for the Traefik dashboard in a future step.

## Install Docker and Docker Compose on the virtual machine

Install Docker and Docker Compose as seen in the Docker and Docker Compose chapter.

As the virtual machine is running Linux, follow the instructions for Linux. Do not forget the post-installation steps. This will ensure that Docker and Docker Compose start automatically when the virtual machine is rebooted and that you can use Docker without the need of using sudo (= admin) each time.

Check that you can run the hello-world Docker image as seen in the Docker and Docker Compose chapter to ensure you can use Docker on the virtual machine without the need of sudo.

### Clone the example repository on the virtual machine

Clone the heig-vd-dai-course/heig-vd-dai-course-code-examples repository to get the code examples but this time, **clone it using HTTPS**.

This will ensure that you can clone the repository without the need of an SSH configuration (useful when you are not on your own computer).

### Try to access the whoami applications from the Internet

Warning

This step might fail, it is normal. You will have to fix it in the next step.

Start the whoami-on-ports-80-and-443 example available in the 24-practical-work-3 directory.

This should start two whoami containers on ports 80 and 443 on the virtual machine.

You should notice that the application is not accessible from the Internet (yet).

This is because the virtual machine is protected by a firewall. You will have to open the ports on the virtual machine to allow the traffic to go through.

### Open the ports on the virtual machine

You can open the port of the virtual machine using UFW with the following command:

```
# Open the HTTP protocol on the virtual machine
sudo ufw allow http

# Open the HTTPS protocol on the virtual machine
sudo ufw allow https
```

### Check out everything is now working

Using the external IP address of the virtual machine, try to access the whoami applications from the Internet again.

This time, you should be able to access the application from the Internet!

You have confirmed that you can access the application(s) deployed on the virtual machine from the Internet.

You can now stop the application on the virtual machine with `docker compose down`.

## Obtain a domain name

If you already own a domain name, you can use it for the purpose of this practical work.

If you do not have a domain or you do not want to use your own domain, you will acquire a free domain name for the purpose of this practical work.

You can register a free domain name at https://desec.io/. Unlike other providers, this provider allows you to create 15 domain names per account with sub-domains for free. In addition, the provider is open-source and does not bind you to some shady terms and conditions.

Create an account on https://desec.io/ and register a domain name. Please use the option **Register a new domain under dedyn.io (dynDNS)** to register a domain name under the `dedyn.io` domain.

The domain name can be anything you want. It does not have to be related to the practical work nor this course and you can use it for other purposes in the future as well.

## Run the secure Traefik and whoami examples

Note

You might need to finish the Web infrastructures chapter before this step.

Start the `traefik-secure` and `whoami-with-traefik-secure` examples in the 24-practical-work-3 directory.

Note

You can run the advanced Traefik and whoami examples if you want. They are more complex but they are more versatile and easier to maintain.

They are not required for this practical work.

If you are able to access the Traefik dashboard and the whoami application from the Internet, you have successfully configured Traefik to use HTTPS with a domain name.

You now have all the required elements to deploy and run your web application on the Internet!

# Grading criteria

- 0 point - The work is not done
- 0.1 point - The work is insufficient
- 0.2 point - The work is done

Maximum grade: 25 points * 0.2 + 1 = 6

## Category 1 - Git, GitHub and Markdown

If your repository is private, you must add us as collaborators to your repository!

| # | Criterion | Points |
|---|---|---|
| 1 | The README is well structured and explains what the web application is for with its documented application protocol interface (API) | 0.2 |
| 2 | The README explains the group composition and roles for each member | 0.2 |
| 3 | The README explains how to install and configure the server with each step | 0.2 |
| 4 | The README explains how to deploy, run and access the web application | 0.2 |
| 5 | The README explains how to configure the DNS zone to access your web application | 0.2 |
| 6 | The README explains how to build and publish the web application with Docker | 0.2 |
| 7 | The README explains how to interact with the web application with examples and outputs using curl | 0.2 |
| 8 | The repository is cloned using Git (HTTPS) on the server to run the web application | 0.2 |

## Category 2 - Docker and Docker Compose

| # | Criterion | Points |
|---|---|---|
| 9 | Docker and Docker Compose are correctly installed on the server | 0.2 |
| 10 | The Docker applications (Traefik + API) are split into multiple directories | 0.2 |
| 11 | The web application runs with Docker and Docker Compose on the server | 0.2 |

## Category 3 - SSH and SCP

| # | Criterion | Points |
|---|---|---|
| 12 | You can access the server without a password using a SSH key | 0.2 |

## Category 4 - HTTP and curl

| # | Criterion | Points |
|---|---|---|
| 13 | The web application makes usage of at least the following HTTP verbs: GET, POST, PATCH/PUT and DELETE | 0.2 |
| 14 | The web application can be used using a web browser, a command line tool such as curl or a custom application client | 0.2 |

## Category 5 - Web infrastructures

| # | Criterion | Points |
|---|---|---|
| 15 | At least Traefik and your web application are deployed on the server | 0.2 |
| 16 | The applications are accessible using a domain name and/or subdomain names | 0.2 |
| 17 | The applications use automatic HTTPS/TLS certificate generations with Let's Encrypt | 0.2 |
| 18 | The domain names are correctly configured in the DNS zone | 0.2 |

## Category 6 - Presentation and questions

| # | Criterion | Points |
|---|---|---|
| 19 | The presentation is clear and well prepared | 0.2 |
| 20 | Everyone speaks during the presentation, and the presentation lasts the time allowed | 0.2 |
| 21 | The presentation presents the access to the server and how to clone and run the web application | 0.2 |
| 22 | The presentation presents the configuration of the DNS zone with its (sub)domain names | 0.2 |
| 23 | The presentation presents the web application | 0.2 |
| 24 | A demo of the web application is made using the deployed version of the application (with curl and/or the browser) | 0.2 |
| 25 | The answers to the questions are correct | 0.2 |

# Constraints

- The web application must be written in Java, compatible with Java 17
- The web application must be built using Maven
- You must use one or more of the Java classes seen in the course
- Your application must be slightly more complex and different than the ones presented during the course
- The web application can only use the HTTP protocol
- You must build and release the application using Docker and Docker Compose
- You must use Traefik as the reverse proxy with a domain name to access the web application
- You must use Docker and Docker Compose to run the web application

# Remarks

Remember the KISS principle: Keep It Simple, Silly! Sometimes it is better to use a simple solution than a complex one.

If your implementation is too complex, we might penalize you.

If elements that are supposed to be acquired through the course or previous practical works are omitted, forgotten or poorly implemented, we might penalize you.

If you have a domain name, you can use it. If you do not have a domain name, you will obtain a free domain name for the purpose of this practical work.

You can protect the `main` branch of your repository to prevent any push on it and force signed commits from team members. This will force all team members to use signed pull requests to merge your work.

# Submission

Your work is due on the **day of the presentations** (the *"Practical work review"* sessions in the official planning available at https://github.com/orgs/heig-vd-dai-course/projects) at **13:15**.

Any commit after the deadline will not be taken into account. Each day of delay will result in a penalty of -1 point on the final grade.

You must create a new GitHub Discussion at https://github.com/orgs/heig-vd-dai-course/discussions with the following information:

- **Title**: DAI 2023-2024 - Practical work 4 - @member1 and @member2
- **Category**: Show and tell
- **Description**: The link to your repository as well as the latest commit hash of your work before submission

# Grades and feedback

Grades will be entered into GAPS, followed by an email with the feedback.

The evaluation will use exactly the same grading grid as shown in the course material.

Each criterion will be accompanied by a comment explaining the points obtained, a general comment on your work and the final grade.

If you have any questions about the evaluation, you can contact us!

# Finished? Was it easy? Was it hard?

Can you let us know what was easy and what was difficult for you during this practical work?

This will help us to improve the course and adapt the content to your needs. If we notice some difficulties, we will come back to you to help you.

➡ GitHub Discussions

You can use reactions to express your opinion on a comment!

# Sources

- Main illustration by Lāsma Artmane on Unsplash