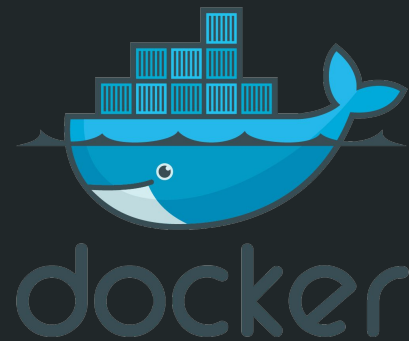
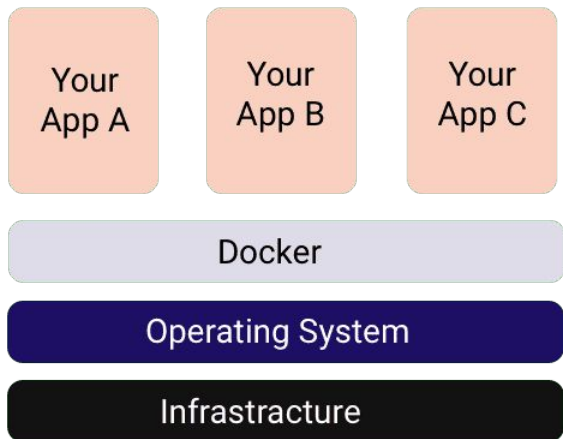


Docker

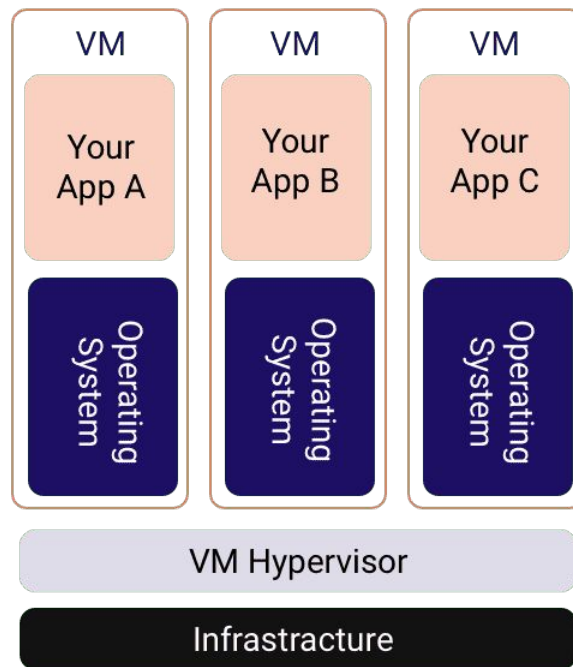


Prob 40 min 🙌

Containers, not a VM, but seems like one



Docker Containers



Virtual Machines

Why Docker

Dependencies

Avoid "Program need python 3.10 not 3.14"

Portability

No flags to set, windows, macOS, linux

Isolation

Run insecure or temporary programs

Port mappings

Avoid "Error: Port Address already in use"

Compose files

They are great

How to make a container

Write a Dockerfile - A file called Dockerfile with no extension

Use `docker build` to build a docker Image

An image is a read-only template for container with files and configs

Use `docker run` to start the container

Let's do it ->



Install docker

Windows / MacOS -> Docker desktop

Good GUI and easy installation

<https://docs.docker.com/get-started/get-docker/>

Linux:

Check package manager

Install docker engine: <https://docs.docker.com/engine/install/>

(Docker desktop for linux)

(Podman is also cool)

Build a docker image

FROM python:3.13

WORKDIR /app

COPY index.html ./index.html

CMD ["python", "-m", "http.server", "80"]



**The world's largest container
registry**

Build a docker image

```
echo "Hello docker" > index.html
```

```
docker build -t my-image .
```

```
> docker build -t my-image .  
[+] Building 0.2s (8/8) FINISHED  
=> [internal] load build definition from Dockerfile  
=> => transferring dockerfile: 159B  
=> [internal] load metadata for docker.io/library/python:3.13  
=> [internal] load .dockerignore  
=> => transferring context: 2B  
=> [1/3] FROM docker.io/library/python:3.13@sha256:0dcc3be031e886a41feafca89f9  
=> [internal] load build context  
=> => transferring context: 31B  
=> CACHED [2/3] WORKDIR /usr/local/app  
=> CACHED [3/3] COPY index.html ./index.html  
=> exporting to image  
=> => exporting layers  
=> => writing image sha256:0cefde62b3d8a681988ccd1f98cbb4f55a52cb5cbba47d65bdd  
=> => naming to docker.io/library/my-image
```

More dockerfile options

FROM python:3.13

WORKDIR /usr/local/app

Install the application dependencies

COPY requirements.txt .

RUN pip install --no-cache-dir -r requirements.txt

Copy in the source code

COPY src ./src

EXPOSE 8080

Setup an app user so the container doesn't run as the root user

RUN useradd app

USER app

CMD ["uvicorn", "app.main:app", "--host", "0.0.0.0", "--host", "0.0.0.0", "--port", "8080"]

Run a container

```
> docker run ubuntu
```

Docker run flags

```
> docker run -it ubuntu
```

-v Volume

-p Port

-it Interactive

--rm Remove

-e Environment variable

Other docker commands

- docker ps** Show docker containers, -a to see all, even stopped containers
- docker kill [container id]** Kill a docker container
- docker rm** Remove a docker container, and any files not in a volume
- docker exec** Execute a command inside the container, fx -it /bin/bash
- docker images** Manage the downloaded docker images
- docker system prune** Remove unused docker data, free up space

Docker compose

compose.yml (docker-compose.yml)

services:

web:

build: .

ports:

- "80:80"

environment:

- hello=world

docker compose up (use -d to detach/run in background)

Making CTF challenges

Our current challenge infra

1 dual-core hostinger VPS with docker

Static vs non-static challenges

Every non-static challenge has a single compose file

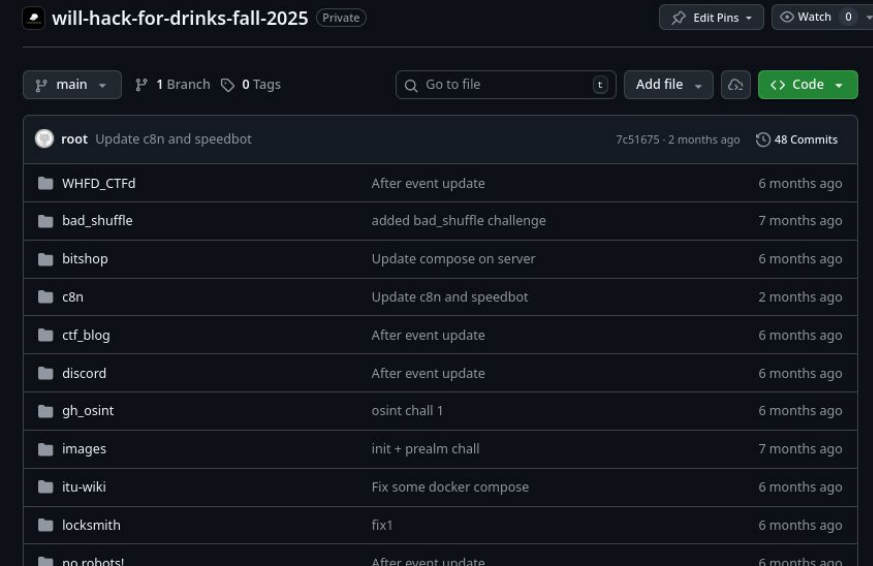
Wrap your challenge in a container

Expose ports and flag through compose.yml

<https://itunderground.dk/blog/making-challenges>

Github

We have a github org



The screenshot shows a GitHub repository page for 'will-hack-for-drinks-fall-2025'. The repository is private and has 1 branch and 0 tags. The main branch is selected. The repository contains several folders, each with a commit message and a timestamp. The folders are: root, WHFD_CTFd, bad_shuffle, bitshop, c8n, ctf_blog, discord, gh_osint, images, itu-wiki, locksmith, and no robots!.

Folder	Commit Message	Timestamp
root	Update c8n and speedbot	7:51675 · 2 months ago
WHFD_CTFd	After event update	6 months ago
bad_shuffle	added bad_shuffle challenge	7 months ago
bitshop	Update compose on server	6 months ago
c8n	Update c8n and speedbot	2 months ago
ctf_blog	After event update	6 months ago
discord	After event update	6 months ago
gh_osint	osint chall 1	6 months ago
images	init + prealm chall	7 months ago
itu-wiki	Fix some docker compose	6 months ago
locksmith	fix1	6 months ago
no robots!	After event update	6 months ago

Domains







itunderground.dk

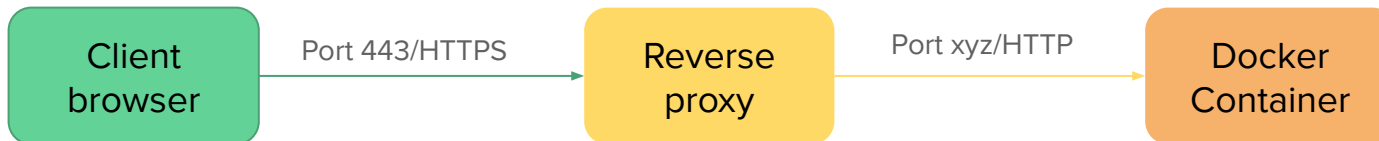
Reverse proxy (docker network)

We use subdomains for challs

Wildcard certificates (HTTPS)

Only for http challenges

SOURCE	DESTINATION	SSL	ACCESS	STATUS
 analyst.chall.itunderground.dk Created: 2nd April 2025	http://analyst:80	Let's Encrypt	Public	● Online
 apichess.chall.itunderground.dk Created: 13th March 2025	http://chess:3000	Let's Encrypt	Public	● Online
 bitshop.whfd.itunderground.dk Created: 26th October 2025	http://bitshop:80	Let's Encrypt	Public	● Online
 blog.whfd.itunderground.dk Created: 26th October 2025	http://ctf_blog:3000	Let's Encrypt	Public	● Online
 chess.chall.itunderground.dk Created: 13th March 2025	http://chess:80	Let's Encrypt	Public	● Online
 ctf.itunderground.dk Created: 5th March 2023	http://CTFD:8000	Let's Encrypt	Public	● Online



Let's make a challenge

<https://github.com/ITUnderground/challenge-templates>