

## Documentation

### Initialize nginx web server from a Docker container with Ansible

To initialize a nginx web server from a Docker container on the dedicated server [hwsrv-745125.hostwindsdns.com](https://hwsrv-745125.hostwindsdns.com) with the files in this archive, follows these steps:

1. *[Only if you have connected via ssh to the remote server [hwsrv-745125.hostwindsdns.com](https://hwsrv-745125.hostwindsdns.com) before and the server has been restarted for maintenance reasons]* Remove the "known\_host" file containing all existing ssh host keys by inputting in a terminal

```
$ rm $HOME/.ssh/known_hosts
```

This step may be necessary because step 4 will crash due to ssh keys mismatch if the ssh keys of the server have changed from your last login due to maintenance or other contingencies.

2. Extract the files from the .zip archive to a desired folder, for example in \$HOME/Ansible\_project. All file inside the archive are needed for a correct execution.
3. Open a terminal and go to the directory containing the unzipped files with

```
$ cd $HOME/Ansible_project
```
4. Deploy the docker container that host an html page on the server<sup>1</sup> [hwsrv-745125.hostwindsdns.com](https://hwsrv-745125.hostwindsdns.com) with the "main.yaml" playbook by inputting in the terminal

```
$ ansible-playbook --vault-password-file=vault_pass.txt main.yaml
```

The playbook instructs the server to

- install python, the docker-py package and the docker command line on the server.
- copy the dockerfile "Dockerfile" and the html page "simple\_html\_page.html" to the server.
- stop and prune all existing containers in the server to free port 80.
- build a docker image using information from the "Dockerfile" file. The Docker image comprises of two layers. The first one installs all system updates and nginx. The second layer initialize nginx and replace the default nginx html page with "simple\_html\_page.html".
- deploy the docker image as container.

### Connect to the hosted html page

To connect to the nginx web hosted page and check that everything is running, open a terminal and input

```
$ curl hwsrv-745125.hostwindsdns.com:80
```

The output should be:

```
$ This file has been created on 04/06/20
```

Alternatively, you can copy and paste [hwsrv-745125.hostwindsdns.com:80](https://hwsrv-745125.hostwindsdns.com:80) in the URL address of your preferite browser. This should redirect to a page containing only the string which is outputted by curl. Note however, that sometimes curl and browsers redirect to different pages. Specifically, curl redirect to the correct page, whereas browsers redirect to the default nginx page. This should arise from some mismatch in file ownership that I was not able to solve in the given time frame. As a consequence, curl should be the preferred tools to connect to the page hosted by nginx.

---

<sup>1</sup> The ssh password to access the server is encrypted with ansible\_vault. The associated vault password is stored in "vault\_pass.txt". Automatic password decryption is warranted by the local configurations in "ansible.cfg" and "inventory.yaml", and by the --vault-password-file=vault\_pass.txt tag.