

Creating Docker Container for ACL Tool

This document would guide you to setup docker on your windows, linux or Mac OS, with our ACL Tool and its dependencies.

We are going to use standard **Ubuntu:20.04**

Create a DOCKERFILE (Windows):

Open a notepad, please put these contents in the file and save without any extension.

```
-----
FROM ubuntu:20.04
MAINTAINER Siddharth Joshi
ARG DEBIAN_FRONTEND=noninteractive
ENV TZ=Europe/Dublin
RUN ln -snf /usr/share/zoneinfo/$TZ /etc/localtime && echo $TZ > /etc/timezone
RUN apt update && \
apt install -y net-tools mtr htop strace && \
apt install -y python3 && \
apt install -y vim python3-pip && \
python3 -m pip install cryptography paramiko netmiko ipaddress napalm pyntc pytest
capiroca && \
apt-get install -y build-essential libssl-dev libffi-dev software-properties-common git && \
git clone https://github.com/1982League/FYP-2022.git && \
git clone https://github.com/networktocode/ntc-templates.git && \
git clone https://github.com/google/capiroca.git && \
cd FYP-2022 && \
./setup.py && \
python3 -m pip install -r requirements.txt && \
apt-get clean && \
-----
```

Open a Command Prompt from the windows, go to the same directory as the **DOCKERFILE** file is created on.

Click on start button and, type **cmd** (Command Prompt).

Once you are on the command prompt, make sure the DOCKERFILE is there, by checking with **dir** command.

```

16/05/2022  14:13  <DIR>      .
16/05/2022  14:13  <DIR>      ..
13/07/2022  17:25                759 Dockerfile
          1 File(s)                759 bytes
          2 Dir(s)  194,507,190,272 bytes free

```

Figure 1 DOCKERFILE

Once confirmed the file is there!! type **docker build.** on the command prompt, (.) denotes the docker build will be in the current directory.

```

Desktop\docker\network automation>docker build .
[+] Building 288.5s (8/8) FINISHED
=> [internal] load build definition from Dockerfile                                0.1s
=> => transferring dockerfile: 811B                                              0.0s
=> [internal] load .dockerignore                                                  0.1s
=> => transferring context: 2B                                                  0.0s
=> [internal] load metadata for docker.io/library/ubuntu:20.04                 1.9s
=> [auth] library/ubuntu:pull token for registry-1.docker.io                   0.0s
=> [1/3] FROM docker.io/library/ubuntu:20.04@sha256:fd92c36d3cb9b1d027c4d2a72c6bf0125da82425fc2ca37c414d4f0180180dc19  0.0s
=> CACHED [2/3] RUN ln -snf /usr/share/zoneinfo/Europe/Dublin /etc/localtime && echo Europe/Dublin > /etc/timezone  0.0s
=> [3/3] RUN apt update && apt install -y net-tools mtr htop strace && apt install -y python3 && apt install -y vim python3-pip && py 279.4s
=> exporting to image                                                            7.0s
=> => exporting layers                                                            6.9s
=> => writing image sha256:20a702f66752a56d530be00d28991b78bd444b0099f22867582a919a3c185844 0.0s

```

Figure 2 Successful Docker Build

Docker Image: **20a702f66752a56d530be00d28991b78bd444b0099f22867582a919a3c185844**

To list out the docker process – **docker ps -a**

```

C:\docker\network automation>docker run -it
20a702f66752a56d530be00d28991b78bd444b0099f22867582a919a3c18584
4 bash

root@88afc8eb5841:/#

```

```

root@88afc8eb5841: /FYP-2022# ls
AUTHORS  README.md  aclgen.py  net_module.py  requirements.txt  telnet_check.py  validate_policy.py
DOCKERFILE  __pycache__  find_target.py  nplm_net_module.py  serial_conn_module.py  telnet_module.py  validate_services.py
LICENSE  acl_tool.py  net_devices.py  policies.json  setup.py  validate_addresses.py
root@88afc8eb5841: /FYP-2022#

```

Figure 3 ACL Tool Docker Container

This container can be setup on Linux, Windows or MAC OS.