Hackfest Development Environment Set Up

Gerardo García (Telefónica, OSM TSC chair)
Pedro Escaleira (IT Aveiro, Expert)

OSM#15

12/06/2023
The following instructions are intended to be generic and applicable to all setups.

For every challenge, you will have dedicated instructions on how to setup your development environment.
General aspects

- Recommended editor: VS Code
- Development can be local or in a remote machine (with VS Code Remote Explorer extension)
- Other requirements: git, Docker, kubectl, openstack-client
Prepare environment to develop in Ubuntu22.04 (1/4)

# Install Python 3.10
sudo apt-get update
sudo apt-get install python3 python3-pip python3-dev
python3 --version       # Python 3.10.X expected as output

# Install git
sudo apt-get install git
Prepare environment to develop in Ubuntu22.04 (2/4)

# Install VSCode (https://code.visualstudio.com/docs/setup/linux)
sudo apt-get install wget gpg
wget -qO- https://packages.microsoft.com/keys/microsoft.asc | gpg --dearmor > packages.microsoft.gpg
sudo install -D -o root -g root -m 644 packages.microsoft.gpg
 /etc/apt/keyrings/packages.microsoft.gpg
sudo sh -c 'echo "deb [arch=amd64,arm64,armhf signed-by=/etc/apt/keyrings/packages.microsoft.gpg]
https://packages.microsoft.com/repos/code stable main" >
 /etc/apt/sources.list.d/vscode.list'
rm -f packages.microsoft.gpg
sudo apt install apt-transport-https
sudo apt update
sudo apt install code # or code-insiders
# Install Docker (https://docs.docker.com/engine/install/ubuntu/ and https://docs.docker.com/engine/install/linux-postinstall/)

```
sudo apt-get update
sudo apt-get install ca-certificates curl gnupg
sudo install -m 0755 -d /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
sudo chmod a+r /etc/apt/keyrings/docker.gpg
echo "deb [arch="$(dpkg --print-architecture)" signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu "$(. /etc/os-release && echo "$VERSION_CODENAME")" stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

```
sudo apt-get update
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
```

```
sudo groupadd docker
sudo usermod -aG docker $USER
newgrp docker
```

# logout and login
Prepare environment to develop in Ubuntu22.04 (4/4)

# Install Openstack CLI (https://docs.openstack.org/newton/user-guide/common/cli-install-openstack-command-line-clients.html)
sudo apt install python-dev python-pip
pip install python-openstackclient

# Install kubectl (https://kubernetes.io/docs/tasks/tools/install-kubectl-linux/)
sudo apt-get update
sudo apt-get install -y ca-certificates curl
sudo curl -fsSLo /etc/apt/keyrings/kubernetes-archive-keyring.gpg
https://packages.cloud.google.com/apt/doc/apt-key.gpg; echo "deb [signed-by=/etc/apt/keyrings/kubernetes-archive-keyring.gpg] https://apt.kubernetes.io/kubernetes-xenial main" | sudo tee /etc/apt/sources.list.d/kubernetes.list'
echo "deb [signed-by=/etc/apt/keyrings/kubernetes-archive-keyring.gpg] https://apt.kubernetes.io/kubernetes-xenial main" | sudo tee /etc/apt/sources.list.d/kubernetes.list
sudo apt-get update
sudo apt-get install -y kubectl
Prepare environment to develop in Windows (1/2)

- Recommended setup:
  - Install Docker Desktop: [https://docs.docker.com/desktop/install/windows-install/](https://docs.docker.com/desktop/install/windows-install/)
  - Install VSCode: [https://code.visualstudio.com/](https://code.visualstudio.com/)
    - Install extensions related to Python
    - Install other extensions depending on the development environment: WSL, Remote Explorer
Prepare environment to develop in Windows (2/2)

- For WSL, you can open Windows CMD and run the following commands:
  - List current installed distributions
    ```bash
    wsl -l -v
    ```
  - List available distros
    ```bash
    wsl -l -o
    ```
  - Install Ubuntu (latest stable version)
    ```bash
    wsl --install
    ```
  - Install a specific distro
    ```bash
    wsl --install -d Ubuntu-22.04
    ```

- Once you have WSL running, you can follow the steps for Linux installation, but avoiding the installation of Docker.
Use VS Code Remote Explorer Set Up (1/3)

VS Code Remote Explorer Set Up (2/3)

- You may develop directly on your attributed VM, using VS Code.

Add New SSH Host...
Configure SSH Hosts...

Select SSH configuration file to update

/home/escaleira/.ssh/config
/etc/ssh/ssh_config
Settings specify a custom configuration file
Help about SSH configuration files
VS Code Remote Explorer Set Up (3/3)
Thank You!