Setting up Ubuntu

Starting from scratch, what we need to kick off is an email, together with its password.

SAAS

Open web browser, login the following, click forget password if necessary :

- login microsoft outlook
- login microsoft team
- login freshteam
- login gitlab

Fail to resolve domain name ygit.yubo.local, so scp file from another machine 10.250.6.80.

```
>> scp dick@10.250.6.80:/etc/hosts .
>> sudo vim /etc/hosts
```

nvim

```
>> sudo add-apt-repository ppa:neovim-ppa/stable
```
(cannot get latest version without this line)

```
>> sudo apt-get update
>> sudo apt install neovim
```

then install nvim plugin manager, please refer to `https://github.com/junegunn/vim-plug` :

```
>> sh -c 'curl -fLo "${XDG_DATA_HOME:-$HOME/.local/share}"/nvim/site/autoload/plug.vim --create-dirs \
   https://raw.githubusercontent.com/junegunn/vim-plug/master/plug.vim'

>> cd .config
>> mkdir nvim
>> cd nvim
>> scp dick@10.250.6.80:~/.config/nvim/* .
>> nvim ~/.config/nvim/init.vim
```

`:PlugInstall` (install plugin inside nvim)

gcc and cmake

```
>> sudo add-apt-repository ppa:ubuntu-toolchain-r/test
>> sudo apt-get update
>> sudo apt install gcc-10 g++-10
>> sudo apt install gcc-11 g++-11
>> gcc --version
```

However this version is old version, we need to reset symlink.

```
>> which gcc
>> cd /usr/bin
>> ll gcc*
>> sudo ln -sf gcc-11 gcc
>> sudo ln -sf g++-11 g++
>> sudo snap install cmake --classic
>> cmake --version    (3.20.5)
```

Install other development tools as well :

```
>> sudo snap install valgrind
>> valgrind --version (3.13.0)
```

Git and Gitlab

```
>> sudo apt install git
>> git --version
>> git clone git@ygit.yubo.local:developer/any_repository.git
```

However it failed, as we need to setup ssh key for target host : `ygit.yubo.local`. Here are the three steps.

Step 1. Generate public and private key

```
>> cd ~/.ssh
>> ssh-keygen --help
>> ssh-keygen -b 4096 -t rsa -C "gitlab_server"
>> mv id_rsa.pub my_gitlab_server.pub
>> mv id_rsa     my_gitlab_server
```

Step 2. Register public key with Gitlab

```
>> cat id_gitlab_server.pub
```

then copy and paste to gitlab personal account

Step 3. Register private key with local ssh folder

```
>> cd ~/.ssh
>> scp dick@10.250.6.80:~/.ssh/config .
>> nvim ~/.ssh/config
```

Put the following inside `~/.ssh/config`

```
Host my_github_server
    HostName github.com
    PreferredAuthentications publickey
    IdentityFile ~/.ssh/my_gitlab_server
    ServerAliveInterval 15
    ServerAliveCountMax 1
```

Now Git works :

```
>> git clone git@ygit.yubo.local:developer/any_repository.git
```

Docker

Curl is needed for Docker :

```
>> sudo snap install curl
>> sudo snap install docker
```

The above installation of docker is not good. Please refer to

```
https://docs.docker.com/engine/install/ubuntu
https://docs.docker.com/engine/install/linux-postinstall
```

Try another way :

```
>> curl -fsSL https://get.docker.com -o get-docker.sh
>> sh ./get-docker.sh
>> sudo groupadd docker
>> sudo usermod -aG docker dick.chow
```

YLibrary

```
>> cd ~/dev
>> git clone git@ygit.yubo.local:dick/Ylibrary.git
>> git clone git@ygit.yubo.local:dick/hk-options.git
>> cd YLibrary
>> git fetch --all (you have all remote branches, but no local branches, you need to create them as follows)
>> git checkout -b develop remote/develop
>> git checkout -b feature remote/feature
>> git submodule update --init --recursive

Then copy scripts that are not in git repo
>> scp -r dick@10.250.6.80:~/dev/hk-options/scripts
```

We need to copy the build scripts, which are not in git. Lets check where the scripts are ...

```
>> scp -r dick@10.250.6.80:~/dev/YLibrary/scripts .
>> nvim
F3
```