



life.augmented

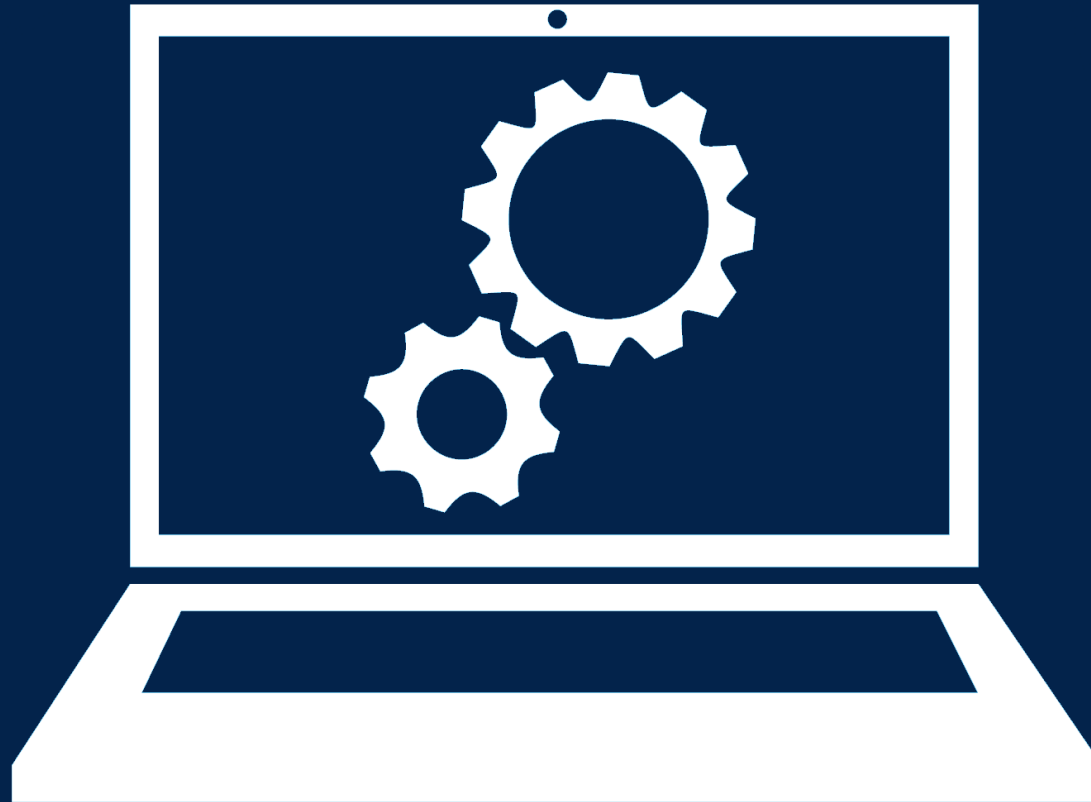
Training environment

Johnson Zhang

March 2021

- 1 PC prerequisites
- 2 Training materials
- 3 Linux virtual machine

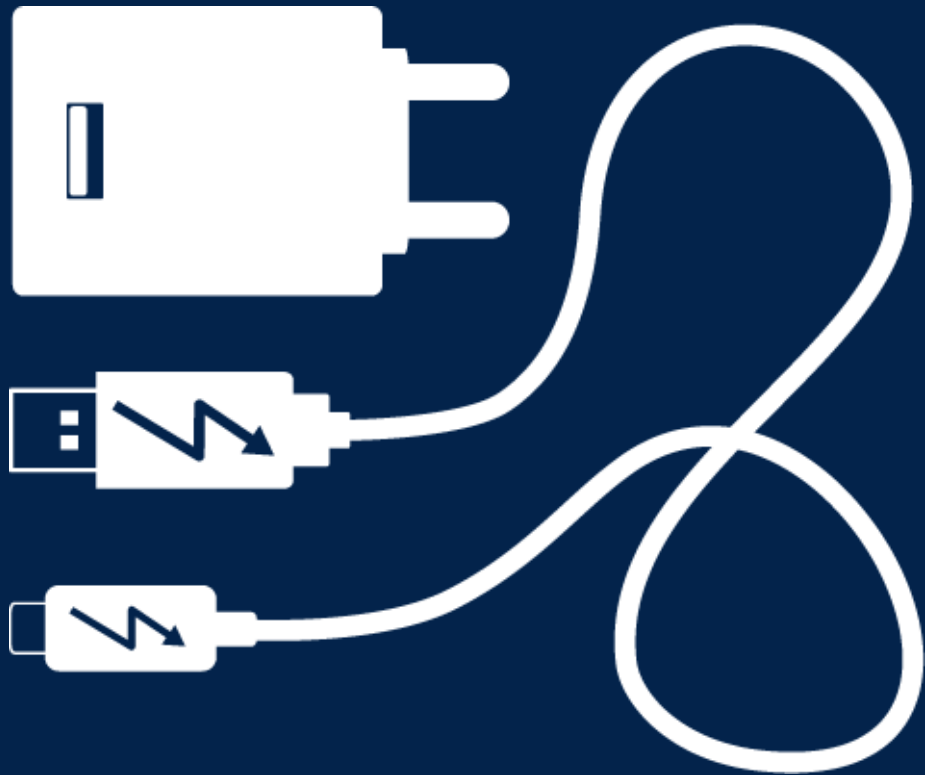
PC prerequisites



Recommended PC configurations

Hardware item	Minimal validated configuration	Comments / Recommendations	
CPU	core i5-2540M @ 2.6 GHz 2 cores (4 threads) 3-Mbyte cache	64-bit instruction set is mandatory 8 cores/threads or more is a good configuration for <i>Developer Package</i> and <i>Distribution Package</i> .	
RAM	8 Gbytes	16 Gbytes or more are recommended especially for <i>Virtual machine</i> setup, <i>Developer Package</i> and <i>Distribution Package</i> .	
Hard drive	320 Gbytes	1 Tbytes is the best suited configuration for <i>Distribution Package</i>	
USB ports	3xUSB3.0	Can be insteaded with a USB hub with independent power supply	
Host environment	Starter Package	Developer Package	Distribution Package
Windows® 10 (64 bits)	Ubuntu virtual machine	Ubuntu virtual machine	Ubuntu virtual machine

Training materials



Training materials

- For the training materials has been distributed :

- 1 DK1 STM32MP15x board

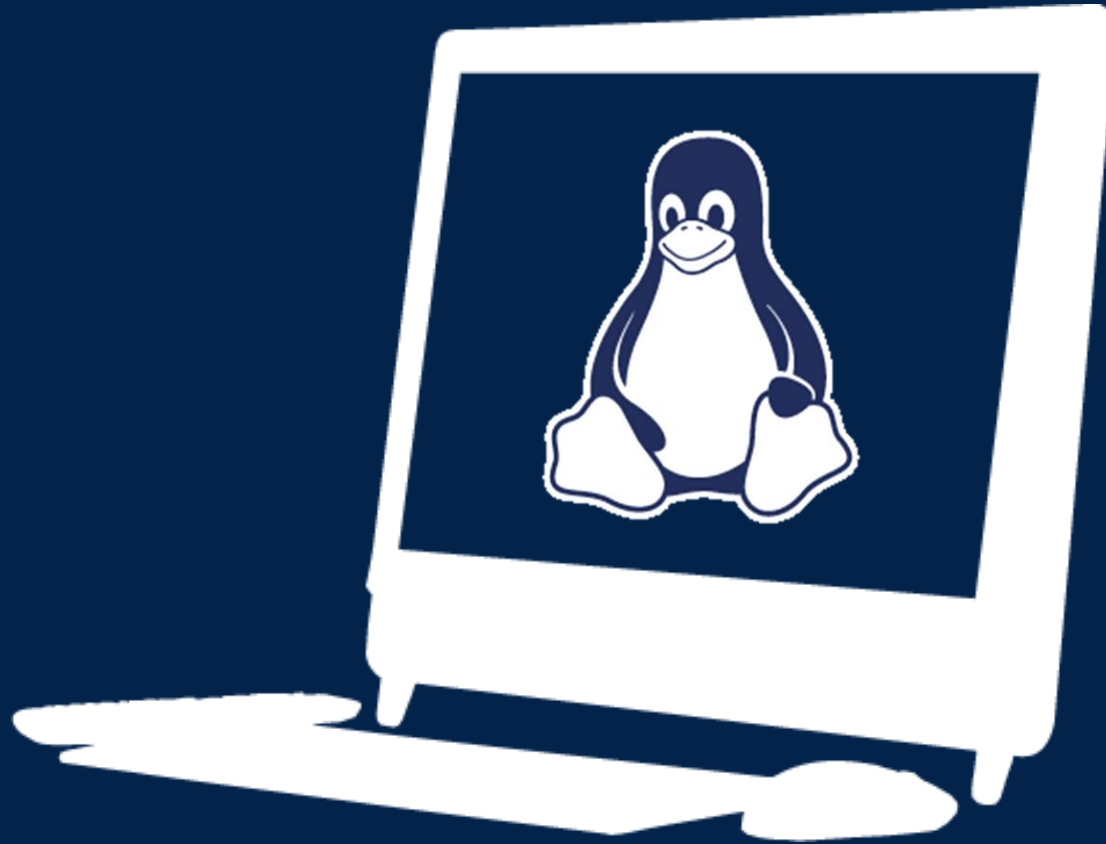


- 1 USB key with the Linux environment ready to use.
- 2 FDCAN transceivers for the developer package hands on.
- Some wires to connect the boards and the FDCAN
- USB cable :
 - 1 micro-usb to type-A to connect on ST-Link and console
 - 1 type-C to type-C to power supply(independent power adapter is better)
 - 1 type-C to type-A for the USB communication with laptop
- **YOU HAVE TO GET BACK all the materials at the end of this training**

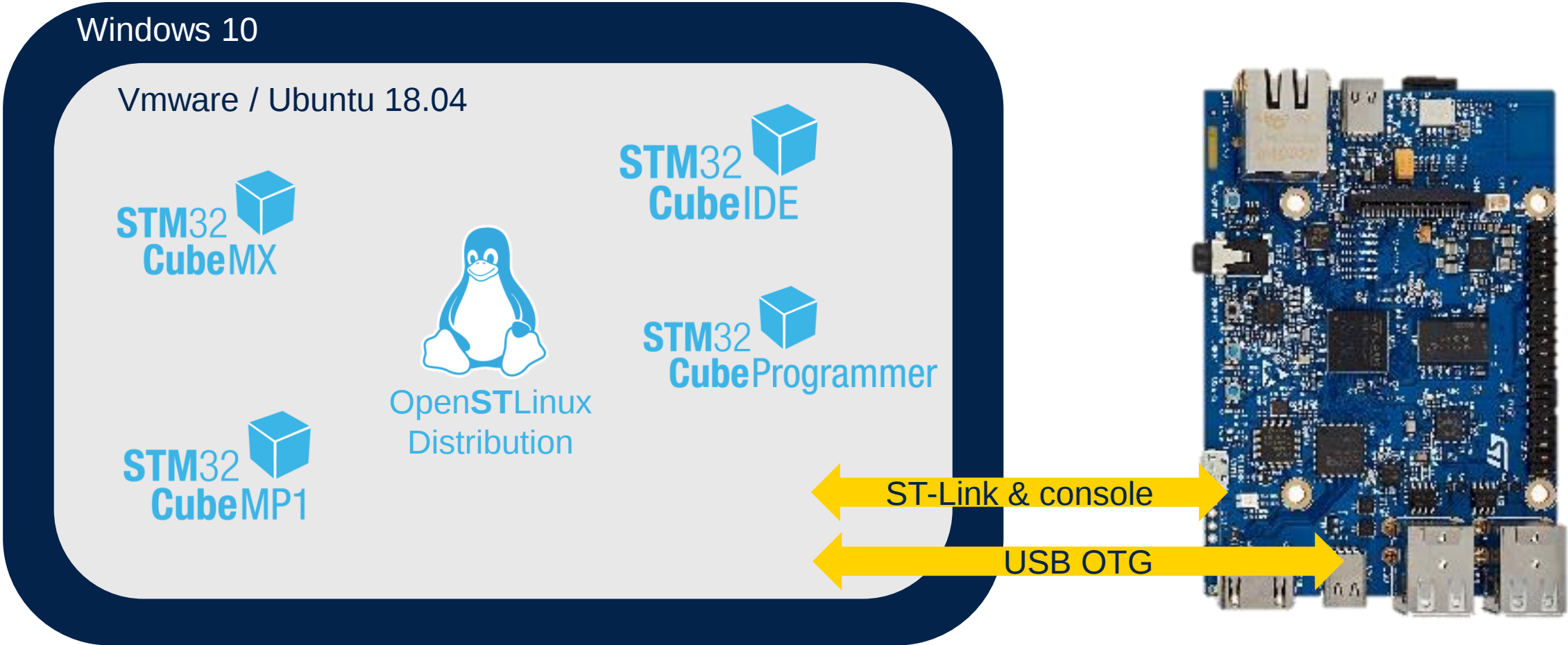
USB key content

- Tool to run virtual machine : vmware player 16
- Virtual machine : Ubuntu 18.04
- ST Tools (already installed in the virtual image):
 - STM32CubeProgrammer v2.6.0
 - STM32CubeMX v6.1.1
 - STM32CubeIDE v1.5.0
- Source code of OpenSTLinux and Cube Firmware needed on the STM32MP1 are installed in the virtual image.
- STM32MP1 documentation as these slides, the datasheet, the erratasheet, the reference manual and the DK1 schematics.

Linux virtual machine

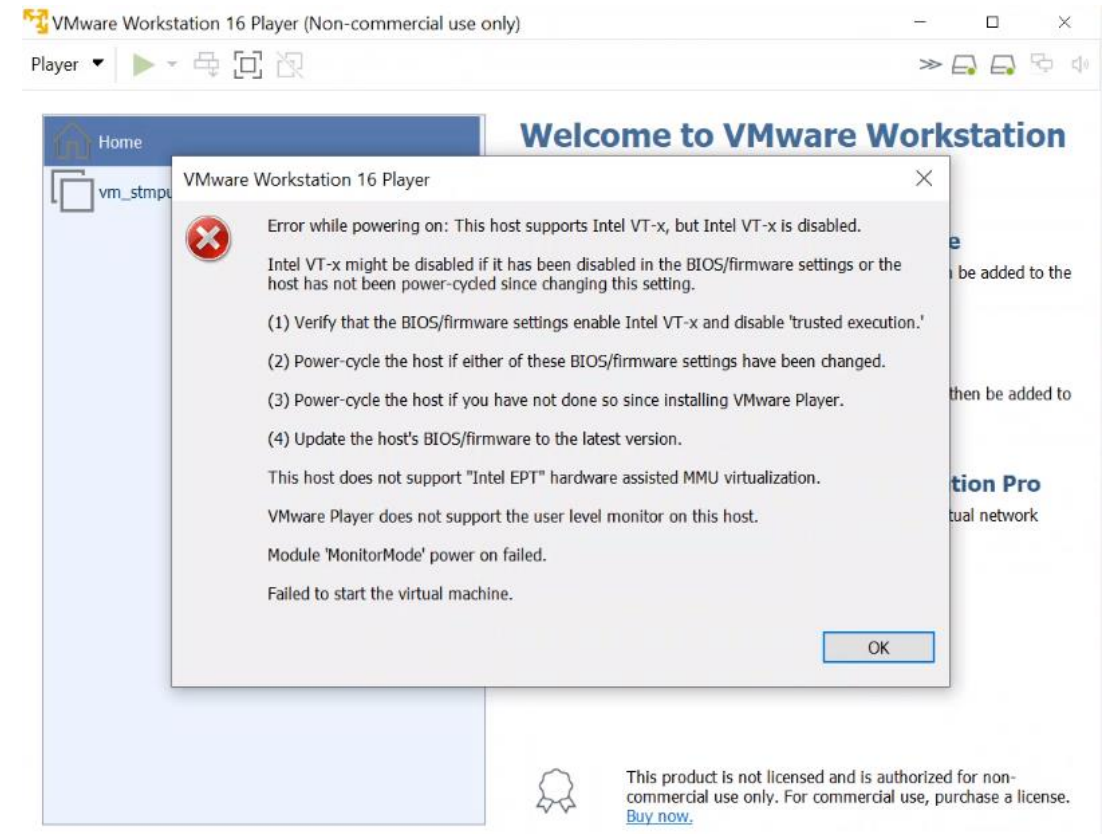


Software Environment Execution



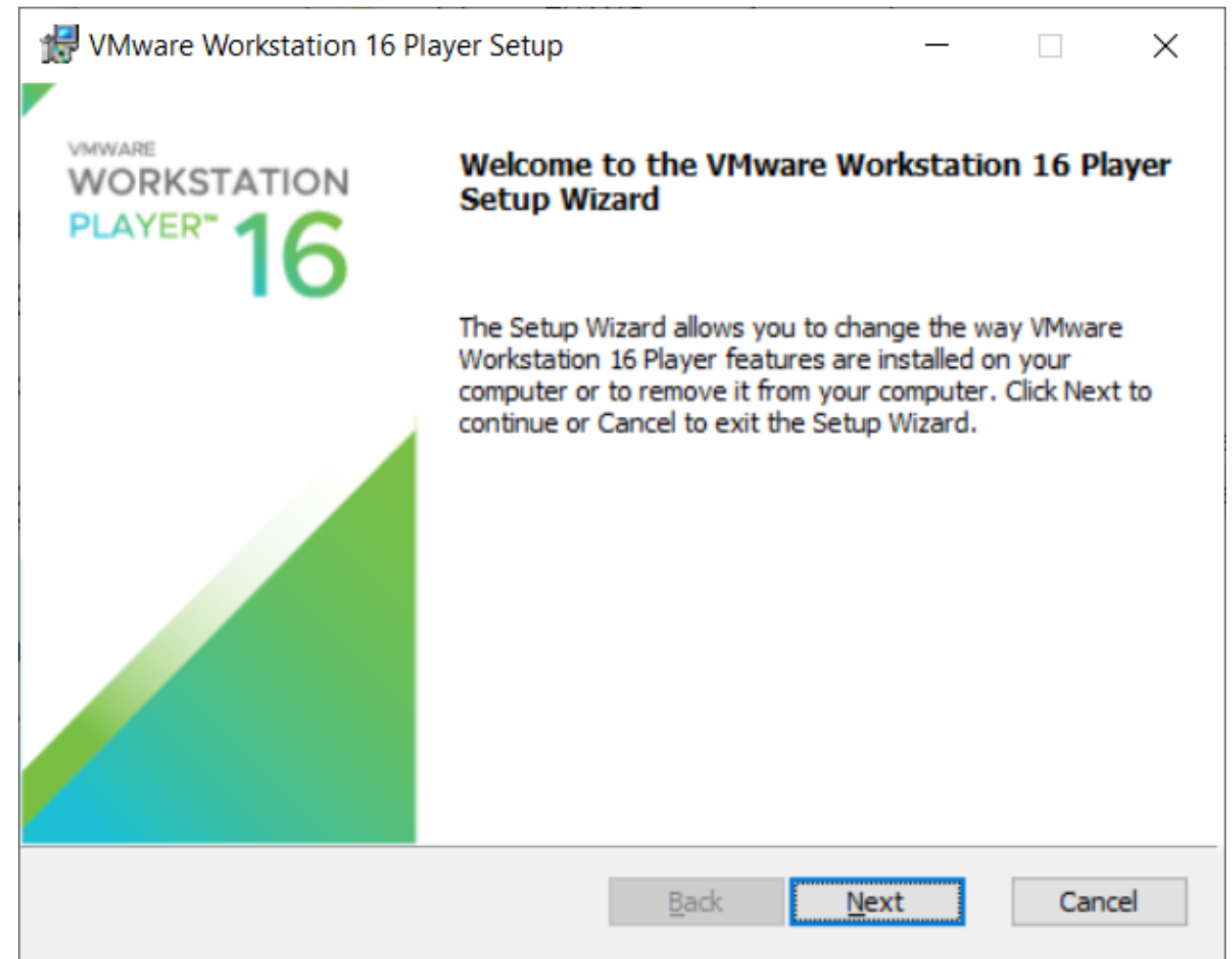
Ubuntu 18.04 virtual machine

- Prerequisites:
 - We use vmware player to run the Ubuntu 18.04 virtual machine for the training, pls enable Intel Virtual IO in BIOS before vmware installation, or you may meet below error:



Vmware player installation

- Follow below steps to install vmware player:
 - execute VMware-player-16.0.0-16894299.exe in “USBKey\vmwareplayer_installer\”.
 - Follow the GUI guide to install VirtualMachine, click the "Next" button one by one.



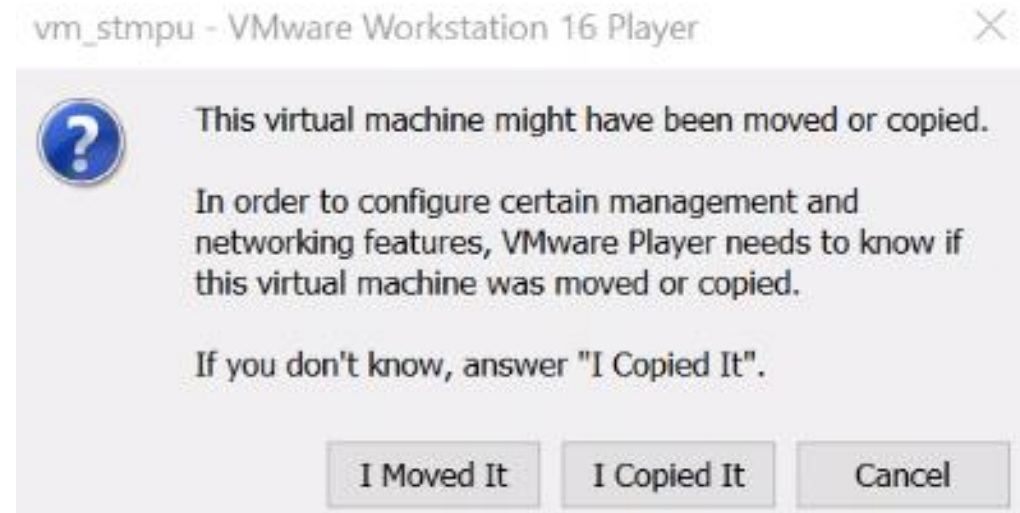
Vmware player installation

- Use the default selection(Use VMware Workstation 16 Player for free for non-commercial use) in the following screen and click “Next” directly:



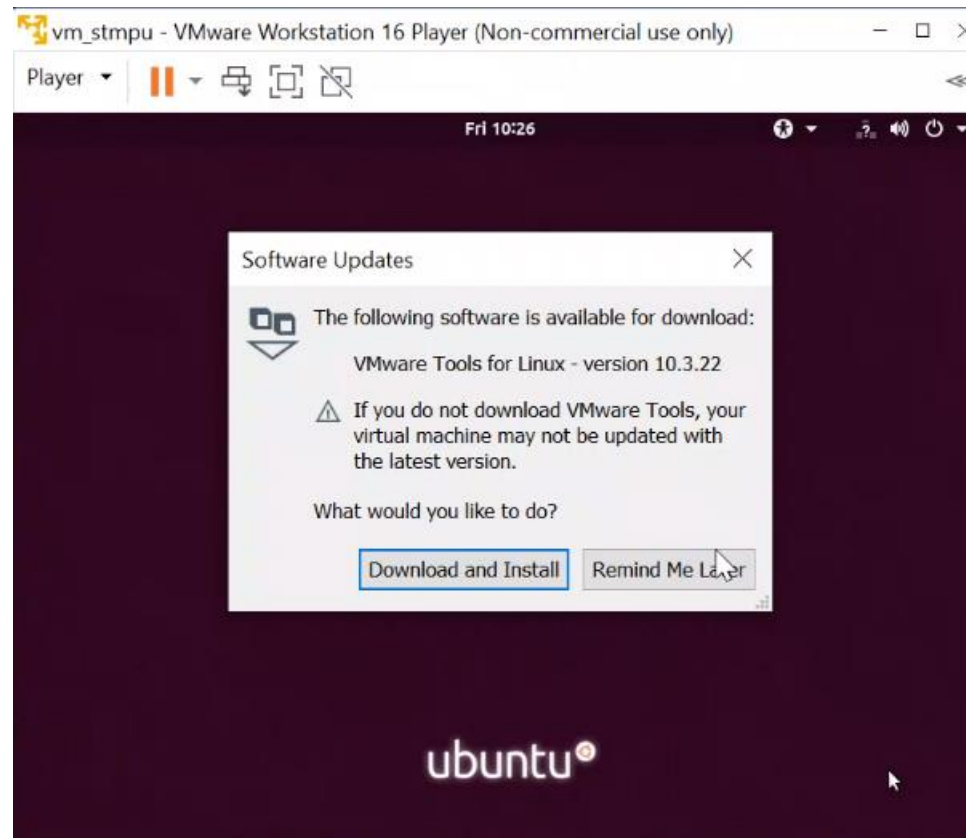
Ubuntu 18.04 virtual machine

- To start Ubuntu 18.04 image in vmware:
 - Double click the vmx file in the USBKey:
 - USBKey\vm_stmpu\vm_stmpu.vmx
- Choose “I moved it” if you meet this prompt:



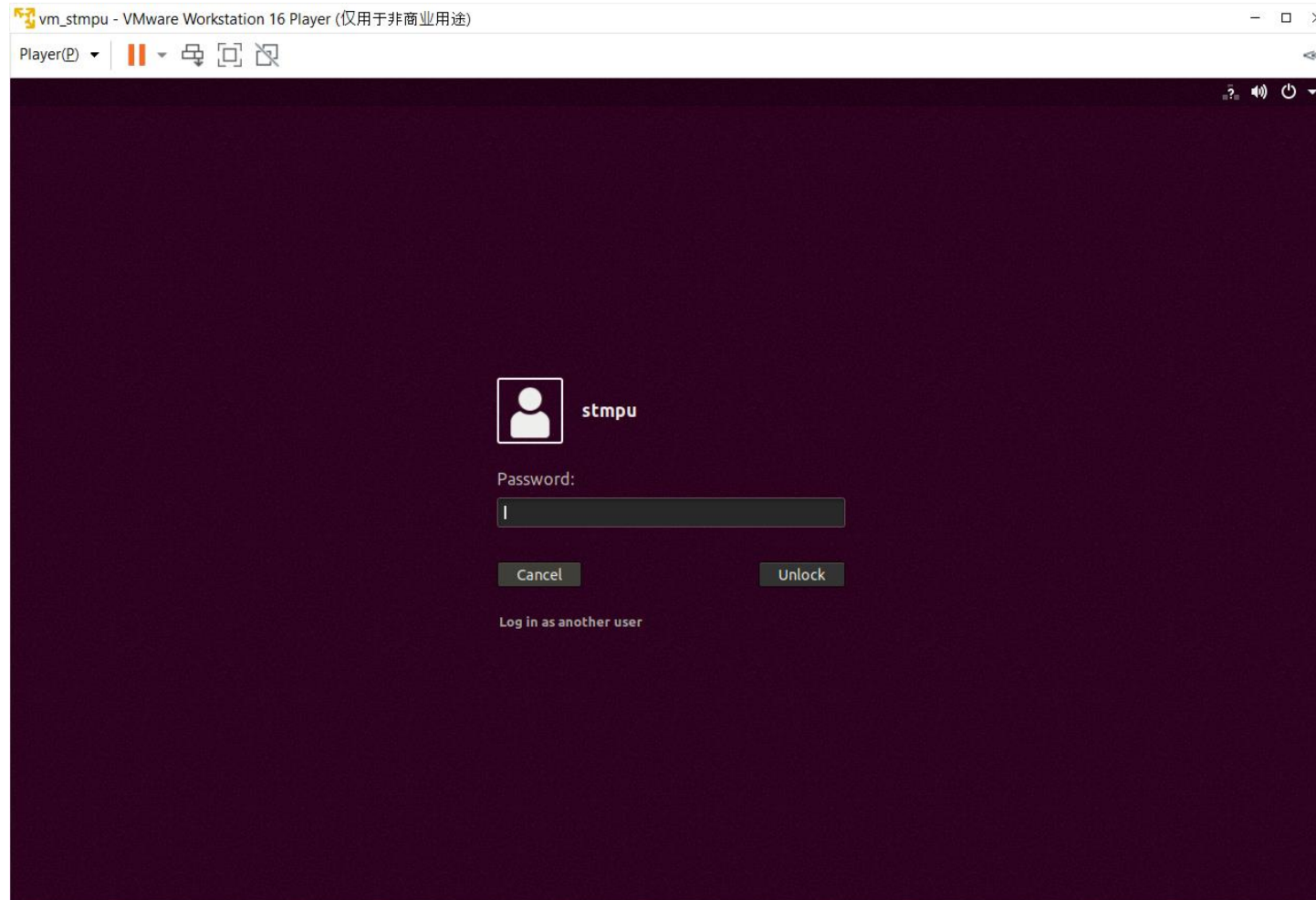
Ubuntu 18.04 virtual machine

- Choose “Remind Me Later” if it ask you to install VMware Tools:



Ubuntu 18.04 virtual machine

- Both the login name and password are "stmpu":



Environment Check

- All packages for STOpenLinux V2.1.0 are already included in the virtual machine:
 - Starter package:
 - /work/starter
 - Developer package:
 - /work/developer
 - Distribution package:
 - /work/distribution

```
stmpu@stmpu: /work$  
├── developer  
│   ├── arm-trusted-firmware  
│   ├── build  
│   ├── linux  
│   ├── optee_os  
│   └── u-boot  
├── distribution  
│   ├── build-openstlinuxweston-stm32mp1  
│   ├── layers  
│   └── wget-log  
└── starter  
    └── stm32mp1-openstlinux-5-4-dunfell-mp1-20-11-12
```

Alias commands

- There are some build-in commands for you to build the packages.
- You can use command “alias” to check all the alias commands.

```
stmpu@stmpu:/work$ setdevenv
stmpu@stmpu:/work$ maketfa
maketfa          maketfaev1emmc      maketfaev1nand    maketfaev1nor     maketfaev1sd      maketfaev1spinand
stmpu@stmpu:/work$ makeuboot
makeuboot        makeubootconfig  makeubootev1
stmpu@stmpu:/work$ makekernel
makekernel       makekernelconfig makekernelinstall makekernelmodule
stmpu@stmpu:/work$ makeoptee
makeopteedk2     makeopteedk2m4  makeopteeev1
```

Thank you