**1-Virtual Machine Setup**

If you want to know what is a virtual machine?

<https://en.wikipedia.org/wiki/Virtual_machine>

Read the explanation of how to create and configure a virtual machine with **VMware player**

In **section 4 in Wiki page PC Pre-requisite** below

<https://wiki.st.com/stm32mpu/wiki/PC_prerequisites>

You will have to :

1/ Install VM Player **14.1.5** for windows

Please use version **14** (and not latest) as ST has license for this version.

<https://my.vmware.com/en/web/vmware/free#desktop_end_user_computing/vmware_workstation_player/15_0>



2/ Get initial Ubuntu 16.04.5 Xenial ubuntu image that will be run by VM player

From [Osboxes\_1604564.7z](file://PRGCWD0579.prg.st.com/TOMAS_Share/MPU/Reference_VM_Images/Osboxes_1604564.7z)

We got it from <https://www.osboxes.org/ubuntu/> but now

Ubuntu 16.04.5 Xenial *is not any more available*

3/ Create a Virtual machine with the ubuntu image downloaded from osboxes.org

Click on the link just below and carefully follow the document VMwarePlayerHelp.pdf

<https://wiki.st.com/stm32mpu/nsfr_img_auth.php/2/24/VMwarePlayerHelp.pdf>

also following document is useful to install VMware tools

<https://wiki.st.com/stm32mpu/nsfr_img_auth.php/4/49/PreRequisite-Vmware-tools.pdf>

|  |
| --- |
| **Tips:** In case of having error:  E: Could not get lock /var/lib/dpkg/lock - open (11: Resource temporarily unavailable)  then  sudo rm /var/lib/dpkg/lock |

**2-Ubuntu configuration to compile the stm32mp1 software**

To Configure the VM machine for compilation of the STM32MP1 embedded software **distribution package.**

To be able to compile the ST distribution a set of libraries have to be added to the default Ubuntu 16.04 machine provide by osboxes.org. Also proxy, git and github account access need to be configured. This is explained in next chapter.

Note that when this configuration is done you will be able to handle all the 3 packages (starter, development,distribution). You have all.

For this configuration in the page <https://wiki.st.com/stm32mpu/wiki/PC_prerequisites>

We follow all the steps (summarized here)

* Section 3 Linux PC
* Section3.1

Only check http Proxy config is ok : first command (wget google …) to check the proxy is opened

Only check git proxy config is ok last command (with git ls-remote) command

(the reset is done by stenv.sh)

* Section 3.2 to install extra package
* Section 3.3 additional configuration
* Section 3.4 for git config and check

Warning: with Ubuntu machine on a VM , FIRST install some tools git repo and java (for cubeMx):

|  |
| --- |
| sudo apt update  sudo apt-get install git repo  sudo apt-get install meld minicom p7zip-full |
| sudo apt-get install default-jre |

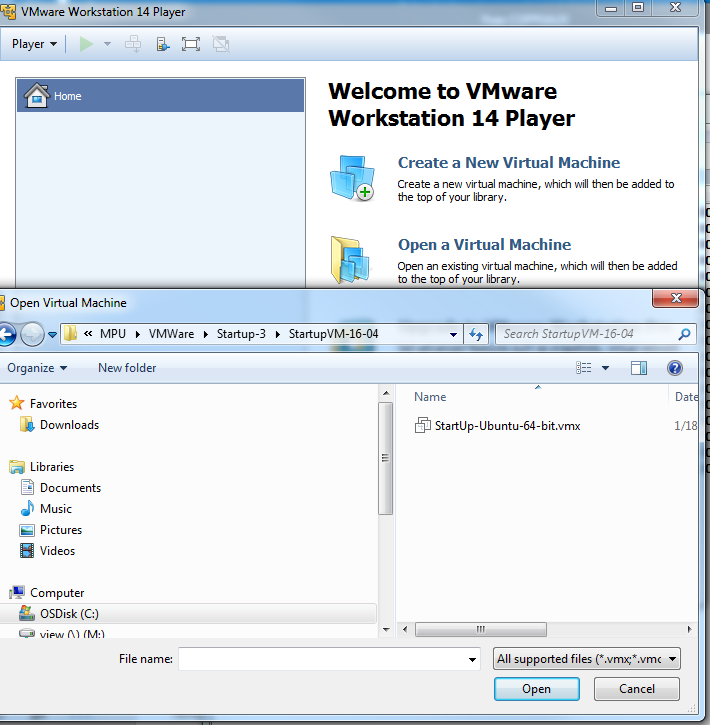
**3-Use the share Virtual Machine image provided for the session**

We prepared a Virtual machine image for VM player where Ubuntu 16.04 is installed and configured to compile stm32 mp1 software (config steps of chapter 1 & 2 above are done).

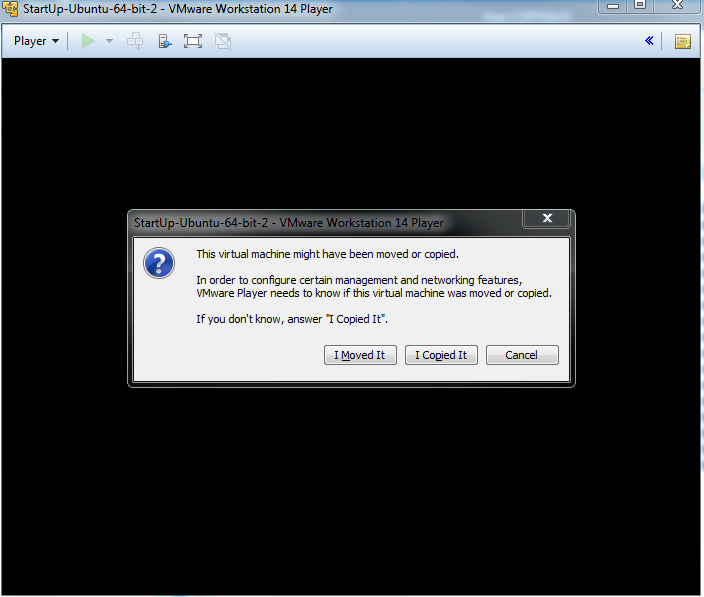
To start this virtual machine

**Open a Virtual Machine**

**Select the .vmx file**



**When starting the existing virtual machine for first time you may need reply to question with “I copied It”**



**The virtual machine appear in the list select it and right click to reach “settings…” menu**

**Configure git and network proxy with own accounts (this is done once)**

3 files contains the passwords

windows login and passwd

$HOME/.git-proxy/.git\_corkscrew\_auth   
$HOME/.git-proxy.auth

github login and passwd

$HOME/.netrc

**Check http proxy is opened**

In a new terminal window:

wget -q www.google.com && echo "Internet access over HTTP/HTTPS is OK !" || echo "No internet access over HTTP/HTTPS ! You may need to set up a proxy."

**Check git proxy is opened**

In a terminal window:

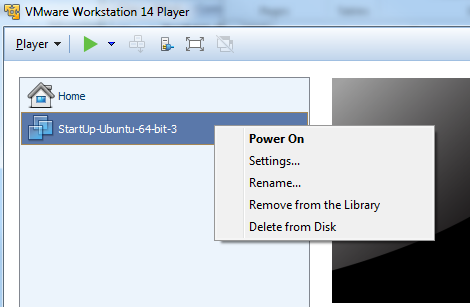
git ls-remote <git://git.openembedded.org/openembedded-core> > /dev/null && echo OK || echo KO

WARNING!!!Do not Forget to clear/erase the passwords in case you would share VM.

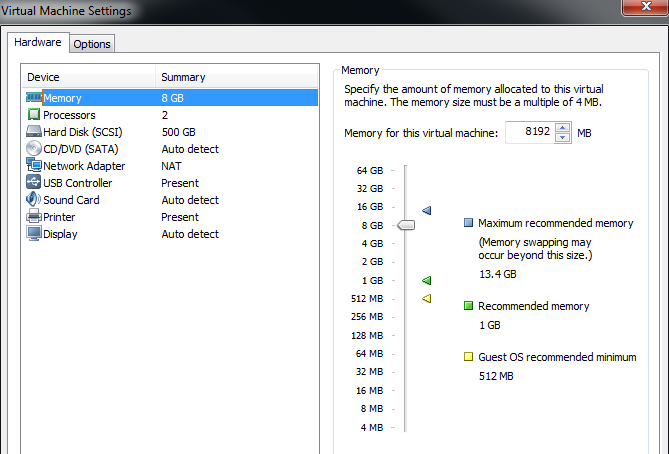
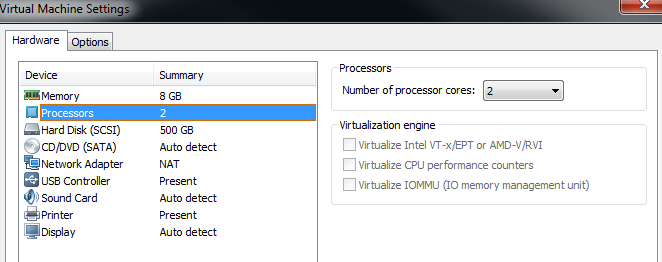
**4-Virtual Machine Configuration**

At any time you can change some setting of your virtual machine:

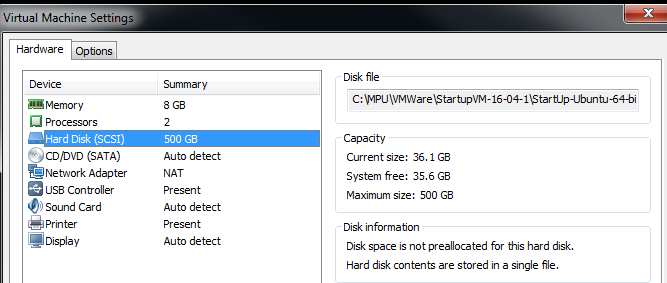
**In Settings menu below , following parameters may need to be configured**



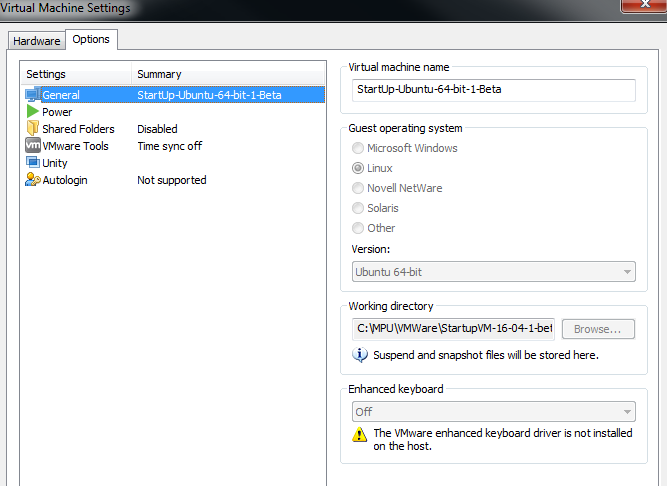
**Half PC resources given to the VM machine for the following :**

* **Memory Size (according to PC RAM available, optimized is to configure 8 Giga for PC with 16Giga of RAM)**
  + 
* **Number of CPU can be tuned depending on PC**
  + 

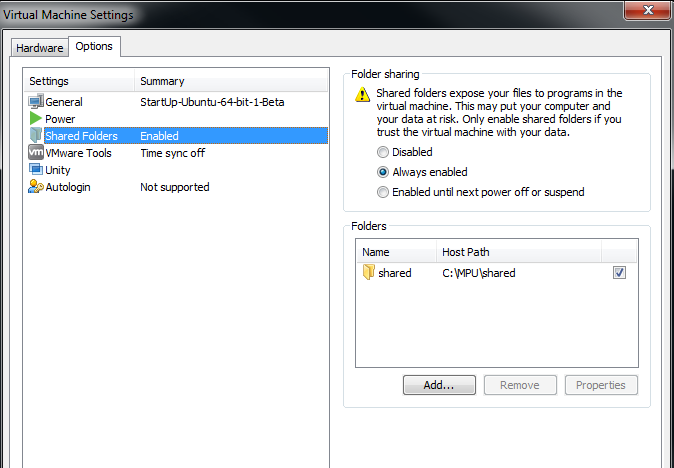
**Hard Drive location of the VM machine (contains the Virtual machine) on your Window PC**



**Virtual Machine Name and Working directory (where the VM player configuration files are stored)**



**Shared directory between Windows and VM machine**



**After, when the WM machine will be ON**

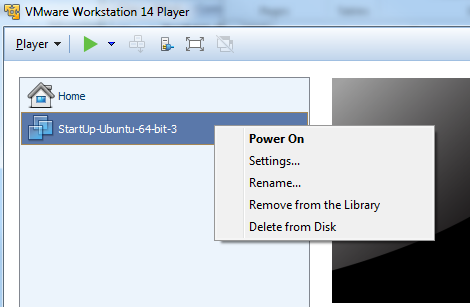
**You will be able to see the content of windows shared folder**

**under the VM machine in /mnt/hgs/shared**

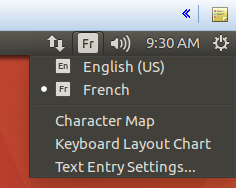
**(shared = the name of the folder on the windows)**

**5-Using Virtual Machine**

**Start virtual machine: select ” Power On”**



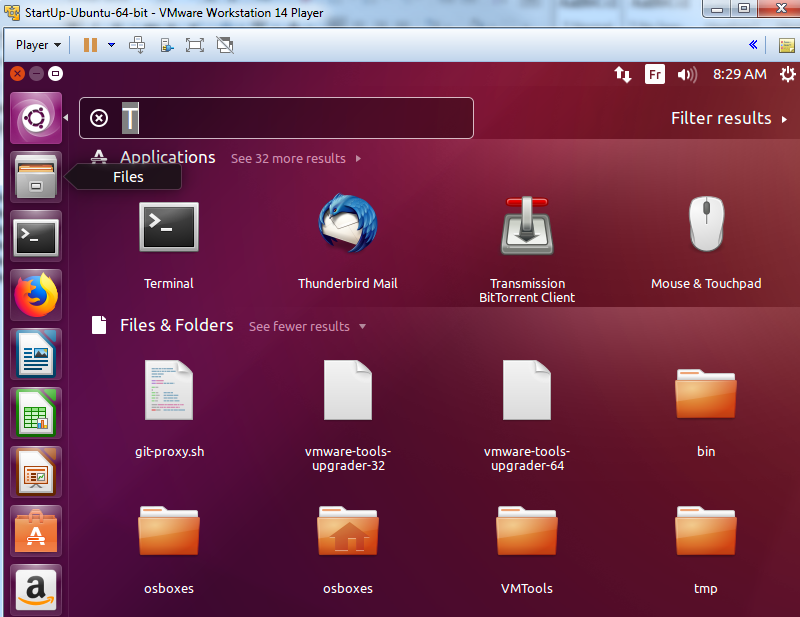
**The console and logging in**

* + **Keyboard selection**
  + 

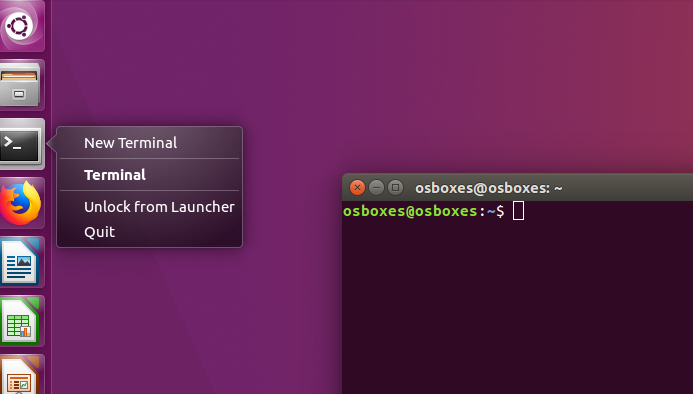


* + **Login : osboxes.org**
  + **Passwd: osboxes.org**

**After login: launch Terminal**



**Terminal is a command line shell window**



**password can be modified with linux command**

|  |
| --- |
| **sudo passwd** |

**The desktop environment hits:**

VM tools allow user to drag and drop files from window file manager to VM machine desktop

Or use copy past from Windows and VM machine

The tool are already installed. But if you want to update.

**Player Menu**

**Manage**

**VM Tools**

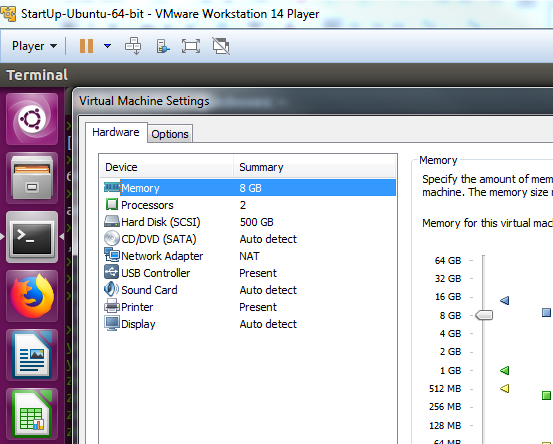
**Re-Install VMware tools**

**Virtual Machine Settings can be accessed in the VM**

**Player Menu**

**Manage**

**Virtual Machine settings**

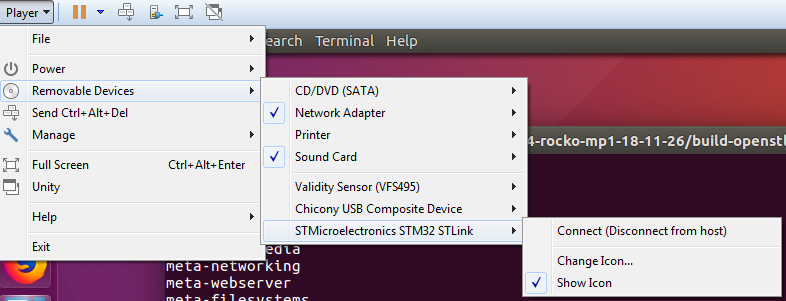


**Removable device**

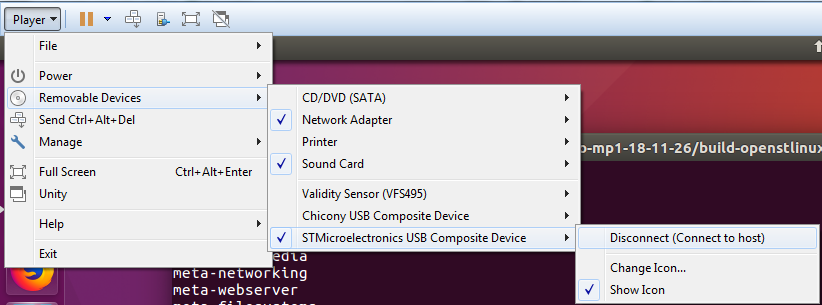
**connect/disconnect removable devices**

**(usb, stlink, dfu, usb over ethernet)**

**Connect (Disconnect from host) means device is currently connected to VM Linux machine**



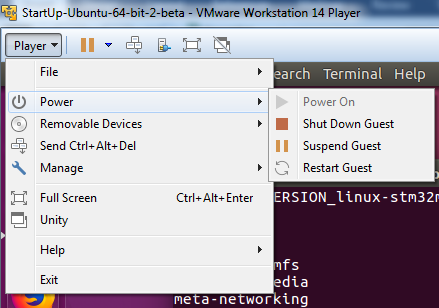
**Disconnect (Connect to host) means device is currently connected to windows**



**Power**

**Shut Down (close linux host completely)**

**or Suspend (keep the context of linux host)**



**!!Be careful to let time the Virtual Machine completes the power off**

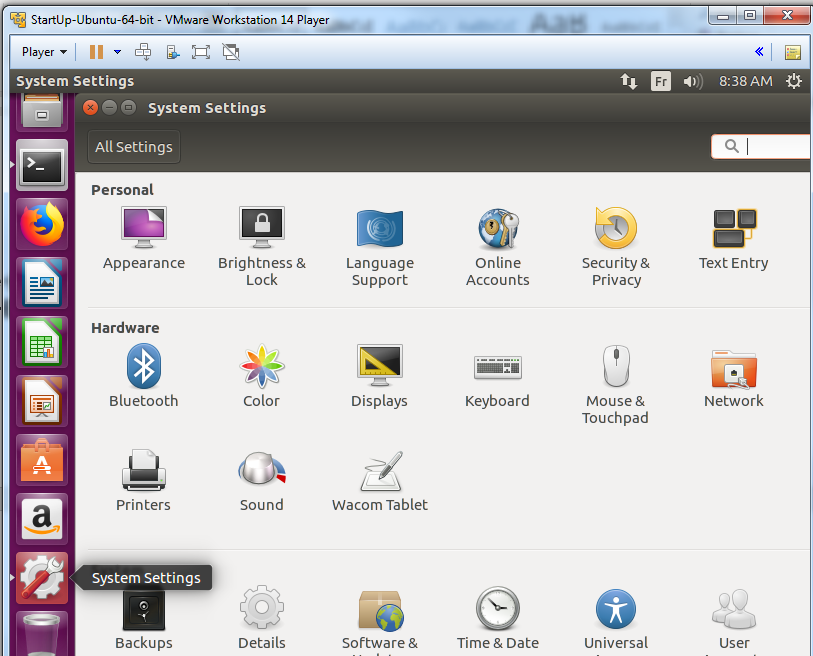
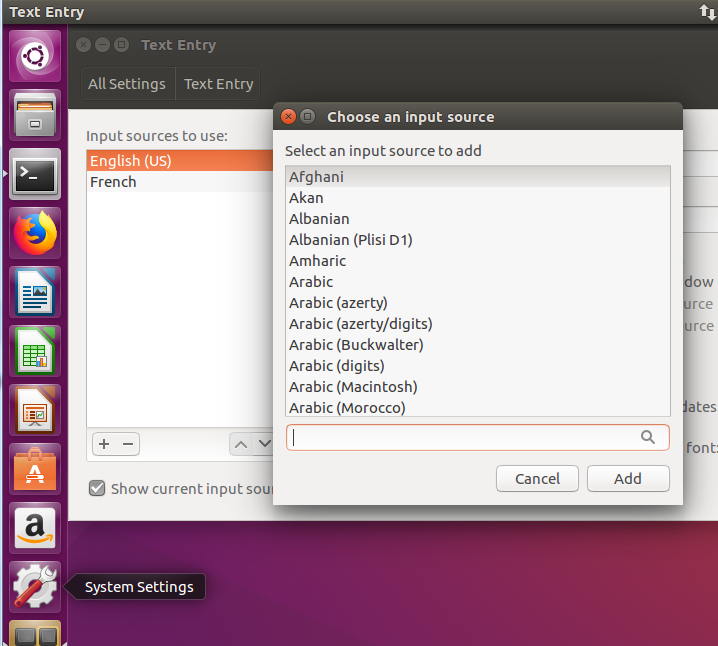
**Otherwise issue can happen at restart**

**Exit (=shutdown)**

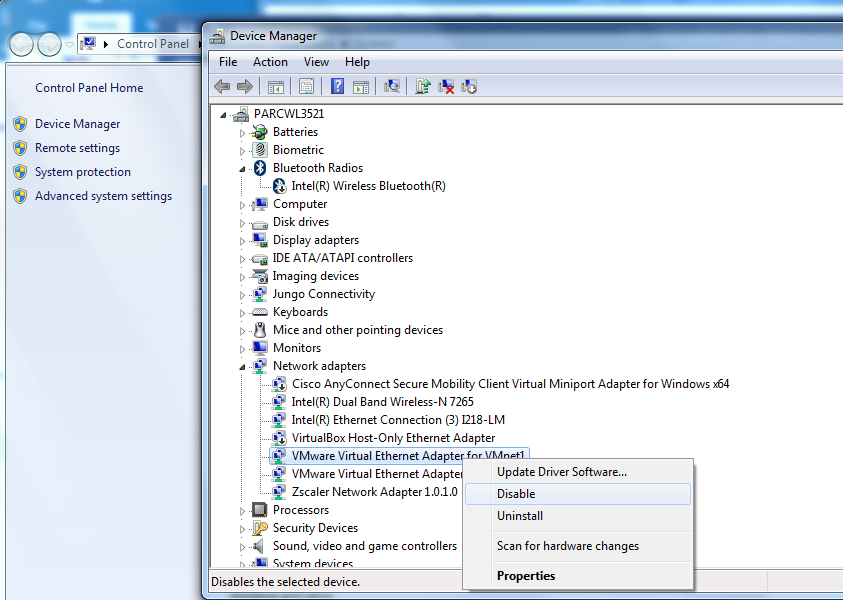
**System settings**

**Add a language (for keyboard for example)**

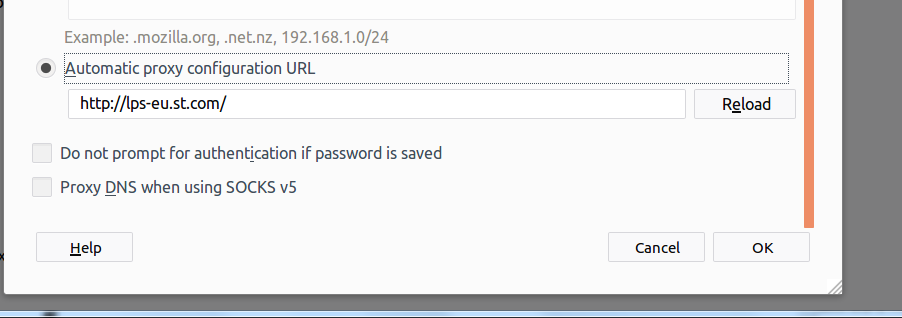
**Use Text Entry**

**If using wifi internet connection may not work any more on windows PC, you need to switch between enable and disable in VM ware network adapter**



**Configure firefox in Preferences->Network Proxy Settings**



When browsing you may need to accept the first security exceptions to login

**6-Basic Linux Lab**

**Linux manual (man <command>)**

|  |
| --- |
| **man ls** |

**List home directory contents**

|  |
| --- |
| **ls $HOME** |

**Create directory contents**

|  |
| --- |
| **mkdir tst\_dir** |

**Change directory**

**relative path**

|  |
| --- |
| **cd tst\_dir**  **cd ..** |

**absolute path**

|  |
| --- |
| **cd $HOME/tst\_dir** |

**current directory**

|  |
| --- |
| **pwd** |

**Create a file**

|  |
| --- |
| **touch test1.txt** |

**Write in file**

|  |
| --- |
| **echo abcd > test1.txt**  **echo efgh >> test1.txt**  **echo abgh > test2.txt** |

**Copy file**

|  |
| --- |
| **cp test1.txt test3.txt** |

**Move file**

|  |
| --- |
| **mv test2.txt text4.txt** |

**Diff between files**

|  |
| --- |
| **diff test1.txt text4.txt** |

**Diff between directories**

|  |
| --- |
| **meld <dir1> <dir2>** |

**List new directory contents**

|  |
| --- |
| **ls \*test\***  **ls $HOME/tst\_dir/\***  **ls .**  **ls ..** |

**Symbolic links**

|  |
| --- |
| **ln –s test3.txt test5.txt**  **ls \*test\*** |

**Display file contents**

|  |
| --- |
| **cat test1.txt** |

**Use an editor**

|  |
| --- |
| **gedit test1.txt**  **vi test1.txt**  **emacs test1.txt** |

**Search a text recursively**

|  |
| --- |
| **grep –R ab $HOME/tst\_dir** |

**Find location of all files with name containing test**

|  |
| --- |
| **find $HOME/tst\_dir –name \*test\* -print** |

**Remove file**

|  |
| --- |
| **rm tst\_dir/test1.txt** |

**Remove directory (use –f option to force remove)**

|  |
| --- |
| **rmdir tst\_dir** |

**Path variable**

**display $PATH variable**

|  |
| --- |
| **echo $PATH** |

**add directory to $PATH variable**

|  |
| --- |
| **export $PATH=$PATH:$HOME/tst\_dir**  **echo $PATH** |

**locate path of a binary**

|  |
| --- |
| **which xterm** |

**File system permissions**

three groups of permissions on UNIX type systems:

“user”, “group” and “others

Three kind of permission

“read”, “write” or “executable

List the permissions on a file

|  |
| --- |
| **ls –l test1.txt** |

Change the permission chmod

(u: user, g: group, o: others)

(r: read, w: write, x: execute)

+ add

-remove

Remove write permission to user

|  |
| --- |
| chmod u–w toto |

Add write permission to user

|  |
| --- |
| chmod u+w toto |

Long format chmod

4+2+1= 7 (rwx)

4+2 = 6 (rw-)

4 (r--)

UserGroupOthers

Example give recursively read access to user only

|  |
| --- |
| **chmod 400 –R $HOME/tst\_dir** |

User=4=rwx

Group=0=---

Others=0=--

**Execute with root permissions**

|  |
| --- |
| **whoami**  **su –l osboxes**  **sudo ls** |

**Command history**

|  |
| --- |
| *history* |

**Recall a previous command (number 15)**

|  |
| --- |
| **!15** |

**Non-interactive text editors**

|  |
| --- |
| sed –e ‘s/*REGEX*/*REPLACEMENT*/g’ < *INFILE* > *OUTFILE*  awk –e ‘{ print $2 }’ < *INFILE*  cut –d: -f1 < /etc/passwd |

**Redirect** stdout and stderr

|  |
| --- |
| *command* 2>&1 >file1 |

**Pipelines**

|  |
| --- |
| *command1* | *command2* |

The output (stdout) from *command1* is used as the input (stdin) of *command2*

***Process and jobs***

|  |
| --- |
| *ps-ef*  *kill -9 pid*  *ps –ef*  *top* |

***Disk space***

|  |
| --- |
| *du*  *df* |

***Sort by directory size***

|  |
| --- |
| **du -k | sort** |

***Ubuntu Packages***

*Install new package*

|  |
| --- |
| *sudo apt-get install* |

*Remove package*

|  |
| --- |
| *sudo apt-get remove* |

*List installed package version*

|  |
| --- |
| *sudo apt list --installed* |

**7 Tips**

**Use tabulation for completion (for instance when parsing directories)**

**Use short cuts to kill a program**

**CTRL+C to kill process**

**CTRL+Z to suspend process**

**Use ‘&’ to launch a program in background (or CTRL+Z and bg)**

|  |
| --- |
| **gedit test1.txt &** |

**or**

|  |
| --- |
| **gedit test1.txt** |

**CTRL+Z**

|  |
| --- |
| **bg** |