# Lab 01 : Recap Lab

Kiss that VMWARE console goodbye and re-familiarize yourself with some basic stuff

1. Working in the VMware console blows. Resolution issues, copy / paste oddities. It’s much better to connect to your machine using the SSH server you installed while setting up your VM. Lets make it easy on ourselves from now on

Look at the screenshot you took after finishing your setup. It shows your Prime’s IP ADDRESS. Now, edit “a” special file (remember OSC ?) on your main host machine and add an alias for that ip, called **Debian-YourFamilyNameWithoutSpaces-YourFirstNameWithoutSpaces-Prime**. (tip : you may have to use an elevated editor to modify this special file)

What is the name of that “special” file ? (QUIZ)

Hosts file

1. Now let’s try this out ! Open a new console on your main host machine and type ssh yourfamilynameyourfirstname@Debian-YourFamilyNameWithoutSpaces-YourFirstNameWithoutSpaces-Prime.

Accept the fingerprint it shows you and connect. For the rest of the labs, always connect using ssh this way. Only when specifically asked for (or when ssh won’t budge for some reason, fall back to the vmware console)

Your prompts looks something like this : 

What does the $ stand for ? (QUIZ)

Promp in bash

1. What command do you use to see your current username ? (QUIZ)

whoami

1. How do you get more information about the use of ANY command ? Apply that to the ls command to find out how to **never** apply color to files located in the root folder (QUIZ)

man

Ls / –color=never

1. You’ll need to become root for the next part. Use the complete *su –login* statement so you’ll become root in a true root login shell

Once you are root, type in *apt install tmux*. Once it is installed, leave the root shell by typing exit. Make sure you have a maximized (or at least pretty large) window for your ssh session, because we’ll be splitting it up next.

TMUX (terminal multiplexer) allows you to keep any session(s) available even if you disconnect from the server (as long as your server isn’t shutdown or you explicitedly terminate a session. Use the helpful command we mentioned above to find out how to (QUIZ)

1. Start a named session called Linux\_Lab\_01 from the terminal
2. Split your tmux session window horizontally (=create a new pane) while in a tmux session
3. Split your tmux session window vertically (=create a new pane) while in a tmux session
4. Switch focus between panes while in a tmux session.
5. Detach from your named session Linux\_Lab\_01 while in a tmux session
6. Re-attach to your existing named session Linux\_Lab\_01 from the terminal
7. List all tmux sessions in the terminal
8. Destroy the named session Linux\_Lab\_01 from the terminal (don’t actually execute this, keep the named session alive till the end of the lab)
9. You should now have 4 terminals running in the same tmux session Linux\_Lab\_01 session. Make sure to connect to the tmux session if you hadn’t in the previous question. You can now continue the lab by using any and all panes your created before.
10. More or less ? use one (and the pipe | symbol) to paginate up and down through a long list of all files in the /etc folder

Ls -R | more

1. Get a long list of all files recursive from the root folder. Pause it. Resume it. Stop it. Bring it back to the foreground. Stop it again. Destroy it without bringing it to the foreground.

Ls / -R

Crtl+z > bg=resume background > crtl+C > fg=resume foreground

1. Give a command that shows the content of the /etc/hosts file (QUIZ)

cat /etc/hosts

1. Use the echo command to show this on the session input window (including brackets) :

(Hello World!)

echo \(Hello World\!\)

1. Will sudo (remember OSC?) work out of the box in a vanilla debian setup like we just did ?

No, the user needs to be added to the sudoers file

1. Show the content of the file where the hashed passwords are stored.

sudo cat /etc/shadow

1. Create a folder called funbits in your normal user’s homedir

Mkdir funbits

1. Create an empty file called joy in your home dir without using an editor

Touch joy

1. Use vi to edit the content of joy. Enter these lines :

Happiness

Sparkle

Sugar

Sunshine

1. Was the vi you used the REAL vi ? what was the real underlying command ? where did it reside ?

The name "vi" is derived from the shortest unambiguous abbreviation for the ex command visual , which switches the ex line editor to its full-screen mode.

1. Move the joy to funbits

Mv joy funbits/

1. Show the first two lines of funbits/joy

cat funbits/joy | head -n 2 or head -n 2 funtbits/joy

1. Show the last line of funbits/joy

cat funbits/joy | tail -n 2 or tail -n 2 funtbits/joy

1. Show all lines in funbits/joy containing the letter “r” and their line number

grep -n r joy

1. Create a hard link called funbits/hard and a soft link called funbits/soft to funbits/joy

What’s difference between the two ?

Ln funbits/joy funbits/hard

Ln -s funbits/joy funbits/hard

You can think a hard link as an additional name for an existing file. Hard links are associating two or more file names with the same inode . You can create one or more hard links for a single file. Hard links cannot be created for directories and files on a different filesystem or partition.

Soft links. A soft link is something like a shortcut in Windows. It is an indirect pointer to a file or directory. Unlike a hard link, a symbolic link can point to a file or a directory on a different filesystem or partition.

1. Show the file permission (and only that!) in rwx format of funbits/joy

-rw-r--r--

1. What is 664 in rwx format ?

-rw-rw-r--

1. Change funbits/joy so that only root will be able to read it (you may need several steps)

chmod a-r funbits/joy

1. Use a statement to find out which shells debian supports out of the box

cat /etc/shells

1. Start a dash shell. How do you go back to your previous shell ?

Bash > exit

1. How can you make variables be persistent in subshells ?

export MY\_VAR=200

1. What is the command to find out much free space (in K) you have left on your partitions ?

Df -hk

1. Use a statement to find out what processes the current user has started

ps -u $USER

1. Show current memory and cpu usage

top

1. Use a single line statement that shows the line from the /etc/passwd file that contains information about the current user (QUIZ)

grep $(whoami) /etc/passwd

grep $(id -u) /etc/passwd

grep $USER /etc/passwd

1. Power down your linux instance