Table of Contents

TTTE LINUX GUIDE

1. INTRODUCTION

1.1. Purpose

The purpose of this guide is to provide a generic guide for working on Linux and with emphasis on the handlings and operations on the command line.

1.2. Audience

Should you need a structured way for installing, configuring and operating a Linux machine, containing the basics this is your guide.

1.3. Set basic regional settings

Choose the regional settings for your region, choose the keyboard language you are familiar with

1.4. Configure disk partitioning

Partitition the disk so that under the root partition you would have at least 25 GB of space - basically this will save you from a lot of troubles ones your root partition is filled up with all the binaries you are going to install - remember disk space is cheap, your work time not.

1.4.1. Check the block devices

To check the block devices issue the following command:

```
sudo lsblk

NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda      8:0    0   35G  0 disk
├─sda1   8:1    0 26.3G  0 part /
├─sda2   8:2    0  286M  0 part /boot
├─sda3   8:3    0    1K  0 part
└─sda5   8:5    0  8.5G  0 part /opt
sr0     11:0    1 1024M  0 rom
```

1.4.2. List the partitions

To list the partitions issue the following command:

```
sudo fdisk -l
Disk /dev/sda: 35 GiB, 37580963840 bytes, 73400320 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xe37db9c5

Device     Boot    Start      End  Sectors  Size Id Type
/dev/sda1  *        2048 55048191 55046144 26.3G 83 Linux
```

```
/dev/sda2      55048192 55633919   585728  286M 83 Linux
/dev/sda3      55635966 73398271 17762306  8.5G  5 Extended
/dev/sda5      55635968 73398271 17762304  8.5G 83 Linux
```

1.5. Enable sudo for the devops user

The devops user in this case referes to your personal Linux username on the System.

```
sudo cp -v /etc/sudoers /etc/sudoers.`date +%Y%m%d_%H%M%S`
# add the appuser to the sudoers group
sudo echo 'appuser  ALL=(ALL) NOPASSWD: ALL' >> /etc/sudoers
```

1.6. Configure ssh

Set-up public private key authentication.

```
# create pub priv keys on server
# START copy
ssh-keygen -t rsa
# Hit enter twice
# copy the rsa pub key to the ssh server
scp ~/.ssh/id_rsa.pub $ssh_user@$ssh_server:/home/$ssh_user/
# STOP copy
# now go on the server
ssh $ssh_user@$ssh_server

# START copy
cat id_rsa.pub >> ~/.ssh/authorized_keys
cat ~/.ssh/authorized_keys
chmod -v 0700 ~/.ssh
chmod -v 0600 ~/.ssh/authorized_keys
chmod -v 0600 ~/.ssh/id_rsa
chmod -v 0644 ~/.ssh/id_rsa.pub
find ~/.ssh -exec stat -c "%U:%G %a %n" {} \;
rm -fv ~/id_rsa.pub
exit
# and verify that you can go on the server without having to type a pass
ssh $ssh_user@$ssh_server
```

1.7. Change the hosts file

Change the hosts file according to the networking requirements.

```
sudo cp -v /etc/hosts /etc/hosts..`date +%Y%m%d_%H%M%S`
sudo vim /etc/hosts
```

1.8. Bash profile configurations

This step is optional,as it's purpose is to configure your bash profile in a way that enables quick configuration settings transfer betweeen the different hosts.

```
mkdir -p ~/"$USER"-confs; cd ~/"$USER"-confs/


 # clone the repo to see the stuff
```

```
git clone git://github.com/YordanGeorgiev/ysg-confs.git .

# check the files
ls -al

# generate the command for every run-time
while read -r f ; do echo cp -v $f ~/$(`echo basename $f`|perl -ne "s/host-name/"`hostname -s`"/g;print") ; \
done < <(find . -maxdepth 1 -type f  -name '.*')

echo 'source ~/.bash_opts.'`hostname -s` >> ~/.bashrc
```

1.9. Configure vim settings

Of course you have your own vim settings you can skip this section, which is eitherway optional - vim is however one of the most powerful editors on the planet , plus it works via .ssh on the terminal - meaning that once you learn some basics of it you will be able to code everywhere ...

```
cd ~/"$USER"-confs/

cp -vr ./.vim ~/
cp -v ./.vimrc ~/
```

2. PROVISIONING

2.1. Provisioning of users and groups

Let's assume that the application(s) you are going to install are going to be installed and run under a separate Linux account, which will belonng to a separate Linux group.

2.1.1. Add the appication group

Add the application group as shown in the command bellow (the reason for using such a large number is due to the fact that lower values are usually used by System and commercial software):

```
export group=appgroup
export gid=10001
sudo groupadd -g "$gid" "$group"
sudo cat /etc/group | grep --color "$group"
```

2.1.2. Add the appication user

Add the application by running the commands bellow (feel free to use different values)

```
export user=appuser
export uid=10001
export home_dir=/home/$user
export desc="the application user of the appgroup group"
#how-to add an user
sudo useradd --uid "$uid" --home-dir "$home_dir" --gid "$group" \
--create-home --shell /bin/bash "$user" \
--comment "$desc"
sudo cat /etc/passwd | grep --color "$user"
```