

## UNIT 4

# Installing Ubuntu Server and network host configuration

**Objectives:** After the end of the activity, the students will be able to:

- Learn the step by step instruction on Ubuntu Server Installation
- First login to ubuntu server
- Check and Test the Configuration

### Procedure:

After the Ubuntu Server Installation, perform the following instruction

## 1. First Login

---

Now Login on the shell (or remotely by SSH) on the server as user "administrator". The username may differ if you have chosen a different name during setup.

```
ubuntu 16.04 LTS server1 tty1
administrator:
Password:
Welcome to Ubuntu 16.04 LTS (GNU/Linux 4.4.0-21-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

6 packages can be updated.
0 updates are security updates.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

administrator@server1:~$ _
```

## 2. Get root Privileges

---

After the reboot, you can log in with your previously created username (e.g. *user*). Because we must run all the steps from this tutorial with root privileges, we can either prepend all commands in this tutorial with the string *sudo*, or we become root right now by typing:

```
sudo -s
```

(You can as well enable the root login by running)

```
sudo passwd root
```

And giving root a password. You can then directly log in as root, but this is frowned upon by the Ubuntu developers and community for various reasons.

## 3. Install the SSH Server (Optional)

---

If you did not select to install the OpenSSH server during the system installation above, you could do it now:

```
apt-get install ssh openssh-server
```

From now on you can use an SSH client such as [PuTTY](#) and connect from your workstation to your Ubuntu 16.04 (Xenial Xerus) server.

## 4. Install a shell based editor (Optional)

---

Here we will install two text based editors. The Nano editor is easier to use for newbies while others prefer the traditional vi/vim editor. The default vi program has some strange behavior on Ubuntu and Debian; to fix this, we install *vim-nox*:

```
apt-get -y install nano vim-nox
```

## 5. Configure the Network

---

Because the Ubuntu installer has configured our system to get its network settings via DHCP, we have to change that now because a server should have a static IP address. If you want to keep the DHCP based network configuration, then skip this chapter. Edit */etc/network/interfaces* and adjust it to your needs (in this example setup I will use the IP address *192.168.1.100* and the DNS servers *8.8.4.4*, *8.8.8.8* starting with Ubuntu 12.04, you cannot edit */etc/resolv.conf* directly anymore, but have to specify your nameservers in your network configuration - see for more details):

```
man resolvconf
```

Open the network configuration file with nano:

```
nano /etc/network/interfaces
```

The server is using DHCP right after the install; the interfaces file will look like this:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto ens33
iface ens33 inet dhcp
```

To use a static IP address 10.0.4.136, I will change the file so that it looks like this afterward:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto ens33
iface ens33 inet static
    address 10.0.4.136
    netmask 255.255.255.0
    network 10.0.4.0
    broadcast 10.0.4.255
    gateway 10.0.4.1
```

```
dns-nameservers 8.8.8.8 8.8.4.4
```

Then restart your network to apply the changes:

```
Sudo service networking restart
```

Or

```
Sudo /etc/init.d/networking restart
```

Then edit `/etc/hosts`.

```
nano /etc/hosts
```

Make it look like this:

```
127.0.0.1 localhost
10.0.4.136 tvetinstitute.local server1

# The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Now, we will change the hostname of our machine as follows:

```
echo server1 > /etc/hostname
service hostname start
```

Afterward, run:

```
hostname
hostname -f
```

The first command returns the short hostname while the second command shows the fully qualified domain name (fqdn):

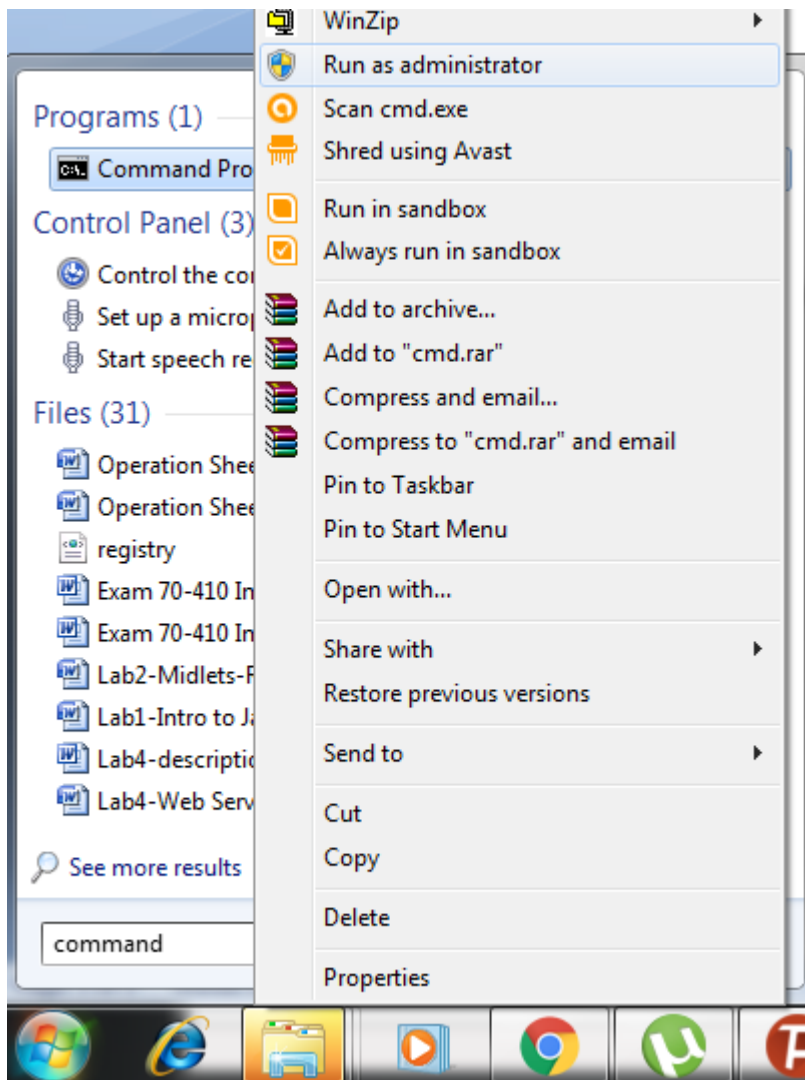
```
root@server1:/home/administrator# hostname
server1
root@server1:/home/administrator# hostname -f
server1.example.com
root@server1:/home/administrator#
```

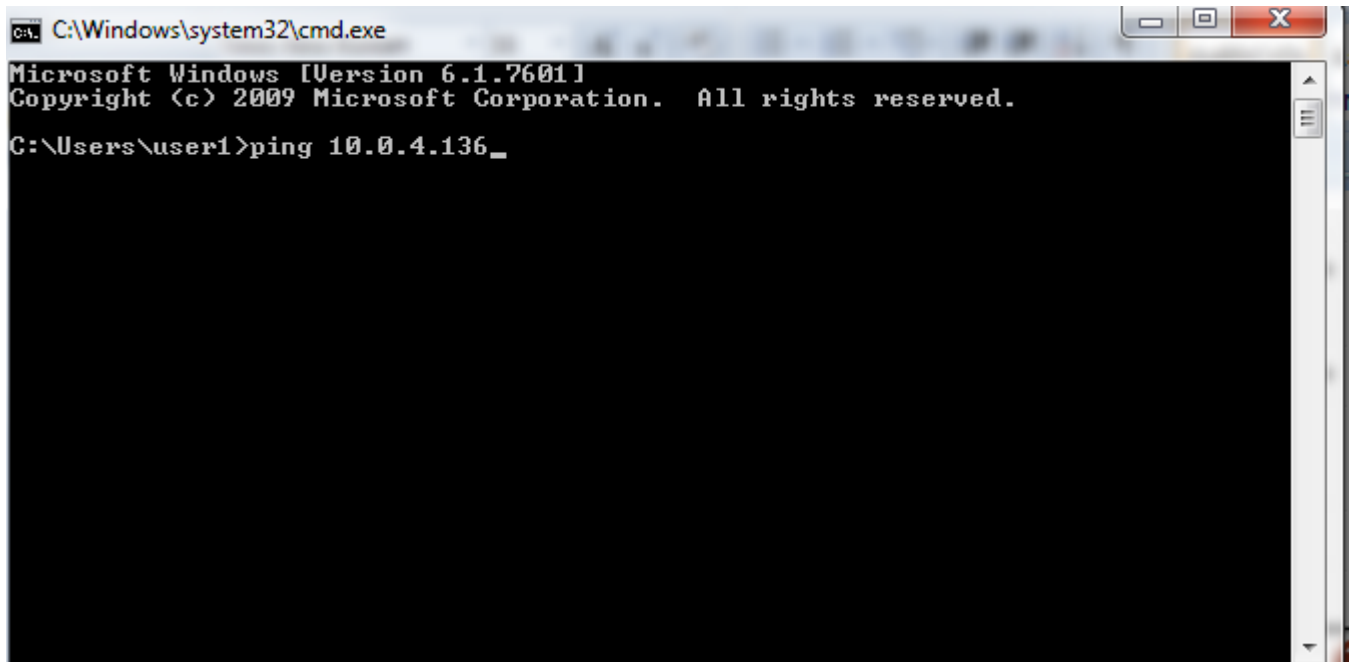
Congratulations! Now we have a basic Ubuntu 16.04 server setup that provides a solid basis for all kind of Ubuntu Server setups.

## 6. Test the network connection

---

**Start up Windows 7 workstation / client computer. Open command prompt and write the command “ping 10.0.4.36”**



A screenshot of a Windows command prompt window. The title bar shows the path 'C:\Windows\system32\cmd.exe'. The window content displays the following text: 'Microsoft Windows [Version 6.1.7601] Copyright (c) 2009 Microsoft Corporation. All rights reserved. C:\Users\user1>ping 10.0.4.136\_'. The prompt is followed by an underscore character, indicating the command is still being processed or the output is being delayed.

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.
C:\Users\user1>ping 10.0.4.136_
```

**6. If responding or replying, you have finished setting up ubuntu server**

---