Chapter 3 Installing the Server

In this chapter, you will:

♦ Understand the installation process

♦ **Download and Install Windows Server 2012** (acquire

    License Key) **and Ubuntu 14** on Virtual Machines

♦ Understand Windows Server 2012 Server Manager,

    basic navigation & commands

♦ Understand Ubuntu v14 basic navigation & commands

♦ Examine basic Linux commands

♦ Configure TCP/IP and set the Server Computernames

UNDERSTANDING THE INSTALLATION PROCESS

During the installation process, the Setup program will ask about your system and its configuration. You should know the answers to these questions *before* you start the installation. Because many of these questions are common to all operating systems, understanding these similarities will help you to install any operating system.

Partitioning the Hard Disk

You need to understand how you will **partition** the hard disk. A partition is a logical division of the hard disk. When you install operating systems, you typically create a number of partitions. Although the concept of partitioning is the same across all operating systems, Windows and Linux use partitions in different ways.

Windows Partitioning

On a Windows computer, you can create a **primary partition**, which is a drive or part of a drive where the system starts its boot process, and an **extended partition**, where more drives can be added as needed. Usually, the operating system boots from the C: drive. When configuring a server, you should isolate the system software from other software and data files on this drive by keeping the files in separate partitions; doing so improves security and system performance.

If hackers access a partition through your Web site, they can likely access data, which is stored on the Web partition that doesn't contain the operating system. They may find it difficult to move from that drive to the one with the operating system, where they could cause more damage. If you install system, software, and data files on the same drive and then hackers or users store so many files on the system that they fill the drive, the operating system will stop functioning. If you store programs and data on a separate

partition and fill up that partition, however, only the programs are affected; the operating system keeps running so that you can clean up the disk.

On a Windows production server, the primary drive partition is C:. The next drive is given the next drive letter available. Typically the D: drive is the CD-ROM drive, the E: drive is the next disk partition, etc.. Allow plenty of space for the E: drive because you will often add temporary files, log files, and other files over time.

Linux Partitioning

The concept of dividing the hard disk into more than one partition in Linux is the same as it is for Windows. You should also keep the Linux operating system separate from applications. However, unlike Windows, Linux does not assign drive letters to disks or partitions. Instead, the drive begins at the root, which is designated with a forward slash (/). Two partitions are required for Linux—the root partition and the swap partition. Linux uses the swap partition to swap instructions and data in and out of RAM when a program requires more space than is available in RAM. Linux can create a number of default partitions for you. Table 3-1 identifies default partitions.

Table 3-1   Linux default partitions

Name	Description
/ (root)	The root partition is the top of the directory structure. It does not contain many files when you first install Linux. If you create directories at the root, it will use up this space.
swap	Linux stores **virtual memory** in the swap partition. The processor uses virtual memory when it does not have enough RAM. A rule of thumb is that the swap partition should be at least the size of your physical memory. By default, Red Hat makes the swap partition double the size of your physical memory. Unlike the other partitions, swap is not part of the directory structure, which means that you cannot explicitly store files on this partition. Rather, the operating system uses it behind the scenes.

Here are the default folders created with the Ubuntu installation.

/usr	Linux stores the files and programs that are shared among all users in the usr partition. Because this partition can contain the most files, it should be the largest. By default, it takes up about 70 percent of the disk after Linux allocates space for the other partitions.
/home	Users create their directories in the home partition. Depending on how many users are on the system, this partition can be quite large. By default, it takes up about 30 percent of the disk after Linux allocates space for the other partitions. On a Web server, however, the home partition can be smaller because users should not store files in the home directory, unless the user is an ISP with user Web sites associated with the home directories.
/var	The var partition contains your Web site, FTP site, log files, and e-mail spool files. Log files and e-mail files can take up a lot of room on a server, so you should increase this partition to at least a few gigabytes. You can take some space away from either /home or /usr to do so.

Services like DNS and Apache and their configuration files are stored in folders under /etc. User files including your DNS zone files and web pages and scripts will be stored in folders under /var. Please explore Ubuntu's file structure with the gedit utility (Linux's equivalent to Windows Explorer & Notepad).

Naming Computers

The primary purpose of naming a computer in either Windows or Linux is to distinguish it from other computers on your LAN. You can name a computer as you install it. Recognize also that you can share that name and its aliases with the rest of the network, even the Internet.

You must give your Windows computer a name that is unique throughout your network. To make it easy to recognize and remember, keep the name simple and descriptive, such as web1. **We'll use the server naming convention W12-fml and U14-fml where W12 is Windows Server 2012, U14 is Ubuntu v 14 and f, m & l are your first, middle and last initials.** The name you assign is used on the LAN for communication; it is not related to how the computer is recognized on the Internet. For example, you might name the computer web1, but to a web user it may be *www.technowidgets.com*. To an e-mail user, it may be technowidgets.com as in *info@technowidgets.com*.

In Linux, computer names are usually referred to as host names. You can create a network with Linux computers that share the same host name with no problems, because in Linux you communicate with other computers by IP address. Of course, it is helpful in general to have unique names associated with each IP address, so it is still a good idea to have unique names that are also descriptive. (The method used to associate names and IP addresses is a separate issue discussed later.)

Licensing the Software

A software license describes how you can use the software. Software licenses differ among the operating systems. Microsoft users must verify that they are compliant with licensing by activating the software once it is installed. Licensing is not a concern for Linux, because copies of the operating system are free, although packaging, support, and special additions may cost extra.

When you purchase a Windows server operating system, each computer that connects to a LAN must have a **client access license** (**CAL**). Two types of licenses for Windows servers exist: per server and per

seat. You use a **per processor license** when you have a single server and not all client computers need to connect to it at the same time. You use **per seat licenses** in network environments with multiple servers. Per seat licensing means that each client computer has its own license and can connect to as many servers as you have. Because most networked environments have multiple servers, per seat licensing is the most common type. Connecting to a Web-based application, whether it is on a LAN or WAN, does not require a CAL, but it may require another type of license such as a per processor license. Typically, however, all computers in a LAN have a CAL.

Linux takes a different licensing approach. Linux is produced under the **GNU general public license**. GNU, pronounced "guh-NEW," is a recursive acronym for "GNU's Not UNIX." According to the GNU general public license, "the GNU general public license is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users." Some companies bundle Linux with extra software for specific purposes and charge more for that. For many environments, a personal version of Linux is satisfactory for basic Web connectivity and networking. The personal version is used in this class. You can obtain a professional version of Linux which has more installation options and better support and is more appropriate for mission-critical systems.

Identifying Devices

The most significant improvement in the installation procedures in servers has been the increased ability of the operating system to correctly identify devices. With Plug and Play devices the operating system automatically configures a device such as a NIC as it is being installed. Windows and Linux support Plug and Play and recognize a wide variety of devices.

Problems can arise when you try to use old specialized devices, such as RAID or older NICs and video adapters. Purchase both the server and the operating system from a top vendor to increase the probability that they will work together.

Selecting a File System

A file system determines the way in which files are organized on a computer disk; in Windows, the file system also determines the characteristics of a file. With both Windows and Linux, you can choose from more than one type of file system. In the Windows operating systems, you can use New Technology file system (NTFS) or the new Windows Server 2012 file system offering **Resilient File System**(REFS). With Linux, ext4 is the current choice, although you can use others, such as ext3.

The **FAT** file system for Windows is based on the original DOS operating system from 1981. For server installations, FAT is obsolete because it does not provide adequate file system security. In the past, one advantage of using the FAT file system was that if the server did not boot correctly, you could boot from a DOS disk and correct the problem by copying or editing files. Now a number of utilities allow you to boot from the floppy disk drive, so you no longer need to use the FAT file system to enjoy this advantage.

The **NTFS** was first introduced with Windows NT. It offers better performance with large hard disks than does FAT, better error correction, and, most important, built-in file system security. The security in NTFS controls file access for all users. In a Web server, everyone who receives a Web page is associated with a user name. The user associated with the Web server can access only those files that are part of the Web site, not the whole hard disk. This restriction greatly improves security.

In Linux the file systems have the same security capabilities. The current file system **ext4** is the fourth extended filesystem. It includes a journaling file system and other improvements listed at https://en.wikipedia.org/wiki/Ext4

Verifying Installation Requirements

The final preparation step before installing an operating system on your server is to verify both the minimum and the recommended installation requirements. The software manufacturer typically sets the minimum requirements needed to allow the software to function. However, most Web server administrators require more than the minimum hardware resources to handle Web traffic efficiently. Linux is less stringent in its minimum requirements. In fact, a special-purpose version of Linux can be embedded on a chip with very little RAM.

To install your operating systems, you will start your 'Create a new VM' and browse to the respective Windows and Ubuntu .iso files. The minimum hardware requirement for Windows is: CPU – 1.4GHz, RAM – 512MB, Disk 32GB. However adequate performance will require a faster CPU and more RAM. We will fine tune the CPU & RAM on your VMs in this Chapter's Hands-On Projects. The disk requirement for your Windows VM won't exceed 25GB. Your Ubuntu VM has a much smaller footprint. The disk requirement for your Ubuntu VM shouldn't exceed 9GB.

Installing Windows Server 2012 R2 on a Virtual Machine

Activity 3-1. Install Windows Server 2012 R2 on a VM

To install the Windows Server 2012 operating system
1. Start VMware Player (or VMware Workstation) and 'Create a new VM'
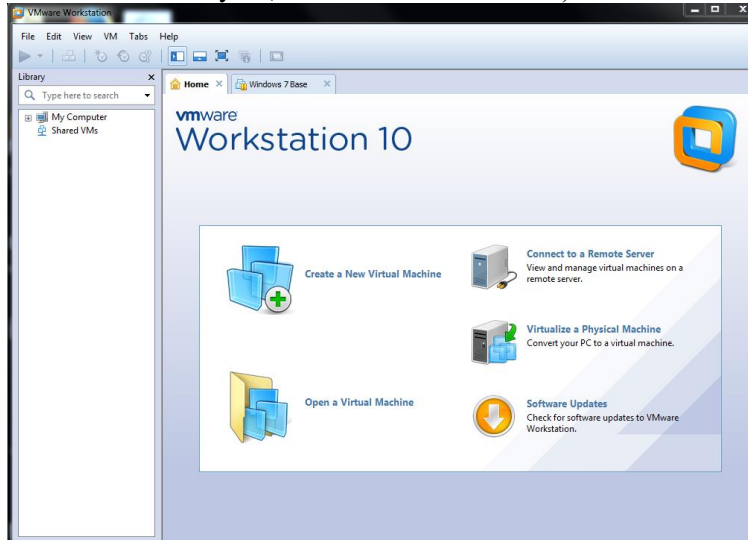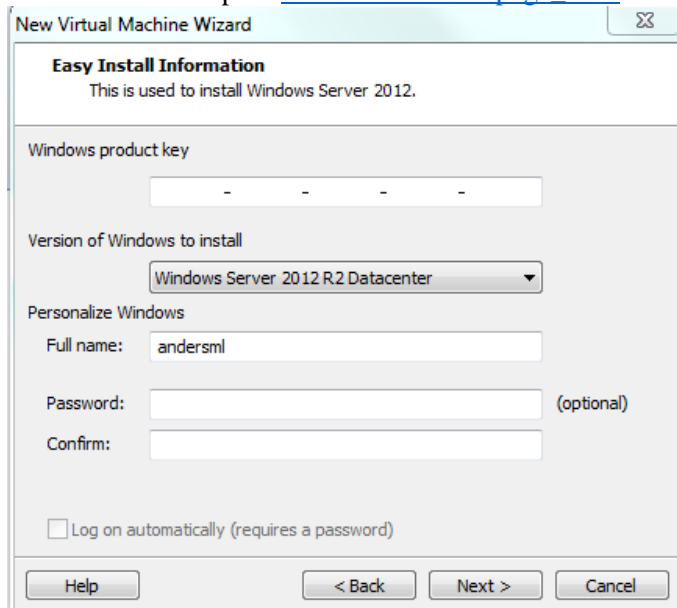
2. Select 'Installer disc image file' and browse to the Windows .iso file
    en_windows_server_2012_r2_x64_dvd_2707946.iso
    from DreamSpark www.cis.lsc.edu/?page_id=2
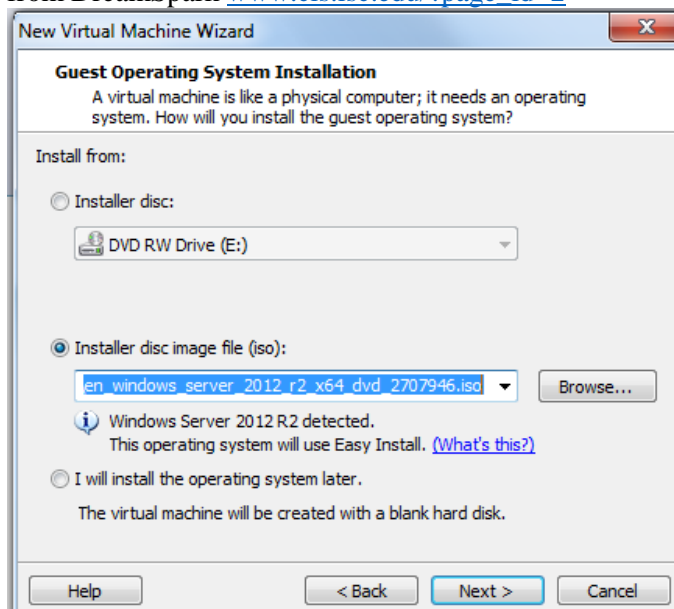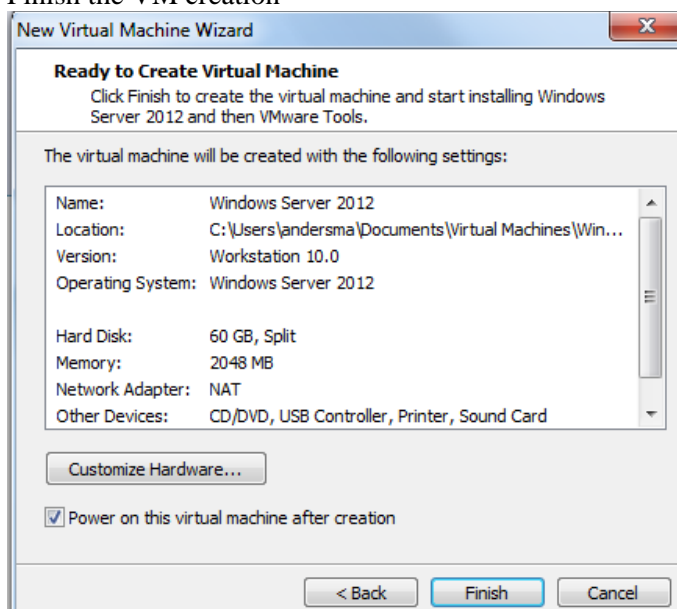
3. Enter your Windows Server license Key
also from DreamSpark www.cis.lsc.edu/?page_id=2
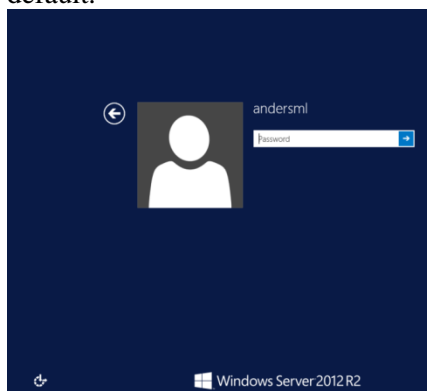

4. Finish the VM creation

5. Choose Windows Server 2012 R2 Standard (Server with a GUI)
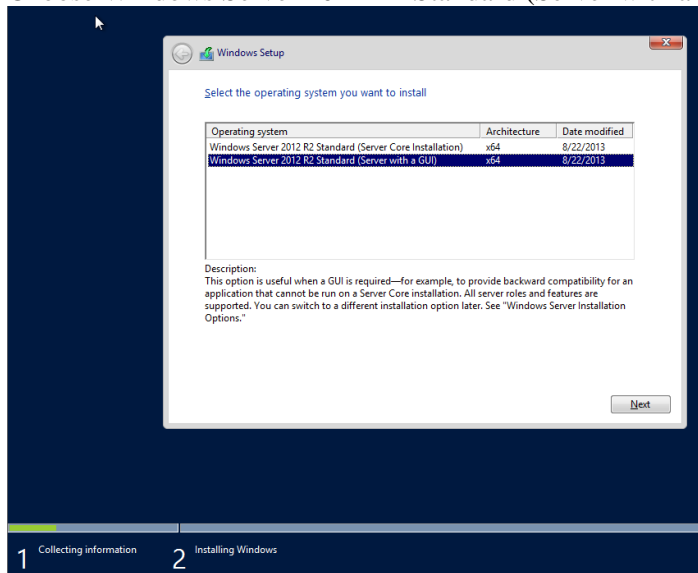

6. Choose next to continue the installation
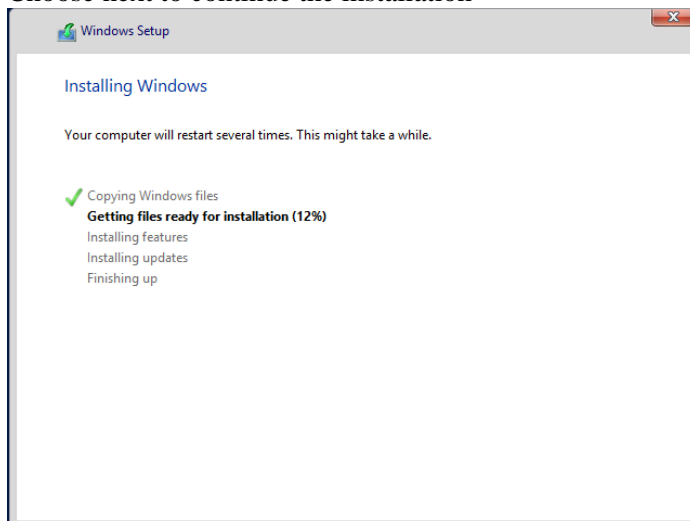
7. The installation should take 10-30 minutes depending on your equipment. When the installation completes, **make a screenshot of your login screen** and login. Server Manager comes up by default.

8. We'll do an introduction to Server Manager in the next Activity. For now add the feature PowerShell by choosing the pulldown menu **Manage** and click **Add Roles and Features.**
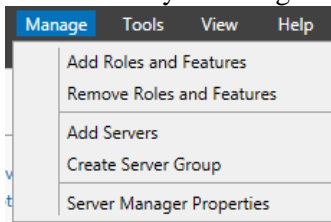
9. Click **Next** twice, click **Features**, scroll down and select **Windows PowerShell** and click **Install:**
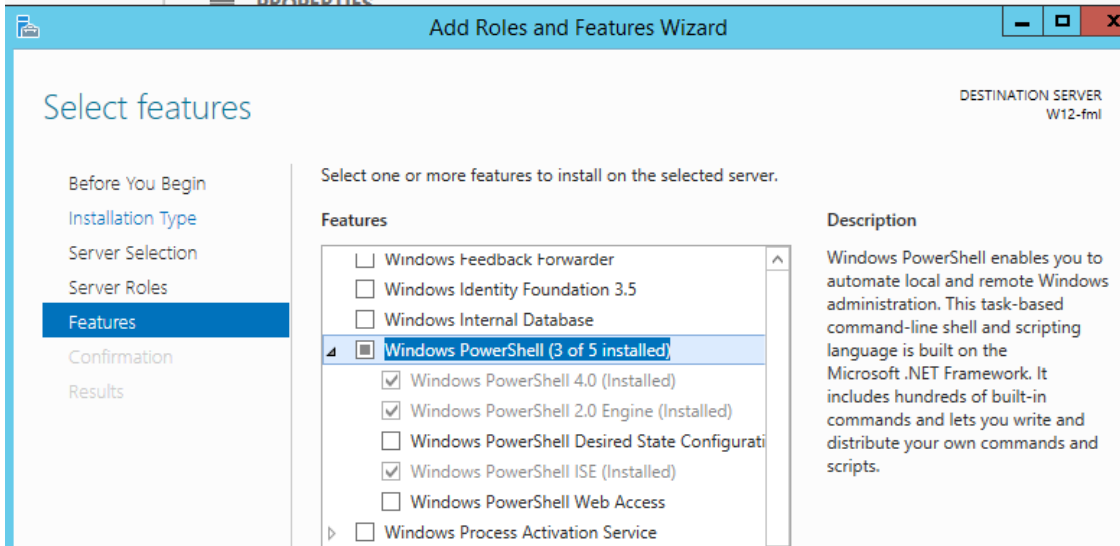
10. Next we'll run ipconfig in PowerShell to view the IP number assigned to us dynamically by VM. Run PowerShell
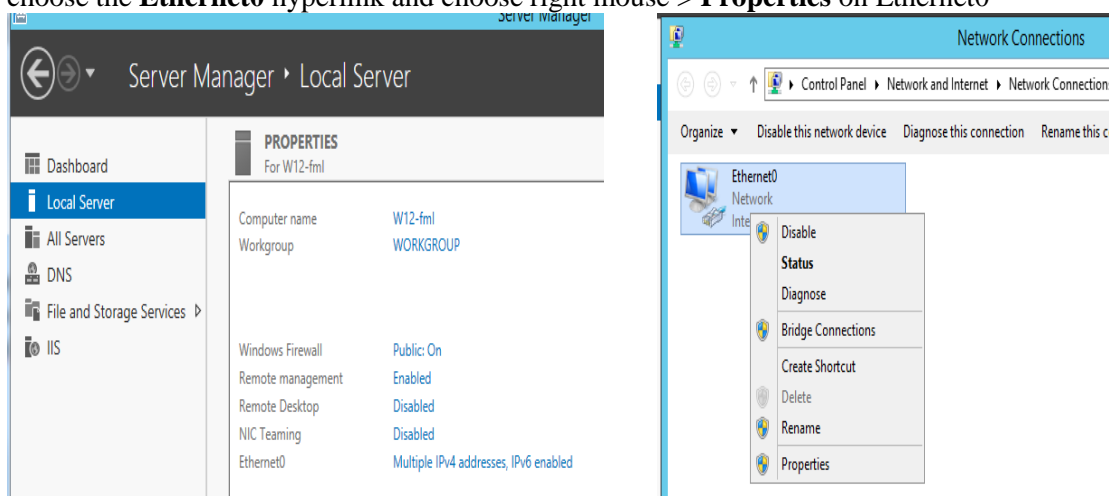

and type: **ipconfig /all**

11. Write down your IPv4 Address and go back to **Server Manager**. From your **Local Server** choose the **Ethernet0** hyperlink and choose right mouse > **Properties** on Ethernet0

12. Choose **TCP/IPv4 Properties** and assign a Static IP address and DNS addresses
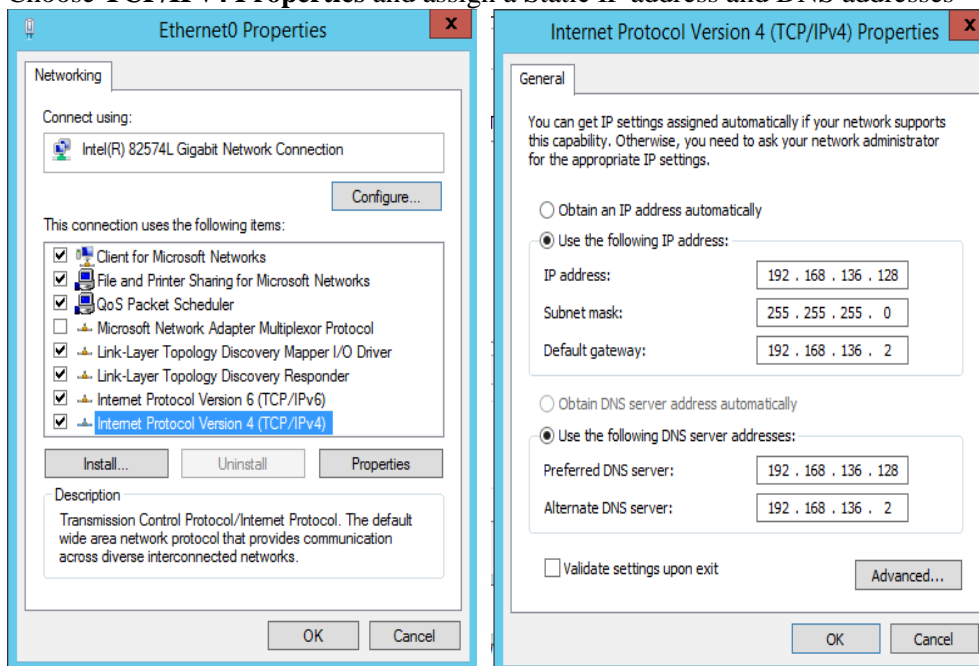
To communicate on a network, you need a network address, a unique host address, a gateway address for the device that connects your network to other networks, and DNS Server address that serves your client computer. The subnet mask defines the network address (in this example 192.168.136) and the host address (in this example .128). The VM above uses .2 as the last octet of the default gateway address. Notice that your Preferred DNS server is yourself (.128) because in the next chapter we setup DNS on this server. Notice that we use the default gateway (.2) for the Alternate DNS server. The VM's DNS server is the host computer which means we're passing DNS requests through to the host computer. We could also have used Google's public DNS server 8.8.8.8. Click **OK** to save this information.

Now test your Static IP from PowerShell by successfully pinging Google's DNS server and making a screenshot.

ping 8.8.8.8

```
Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 43ms, Maximum = 121ms, Average = 80ms
PS C:\Users\andersml> ping 192.168.136.128

Pinging 192.168.136.128 with 32 bytes of data:
Reply from 192.168.136.128: bytes=32 time<1ms TTL=128
Reply from 192.168.136.128: bytes=32 time<1ms TTL=128
Reply from 192.168.136.128: bytes=32 time<1ms TTL=128
Reply from 192.168.136.128: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.136.128:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Users\andersml>
```

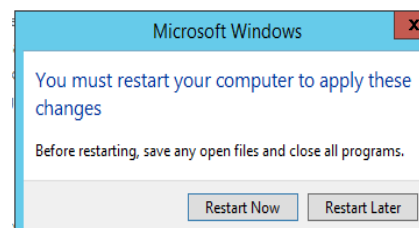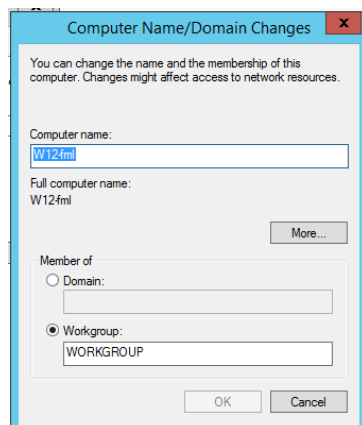13. Next go to Server Manager > Local Server and select the **Computer Name** tab. Click the **Change** button.
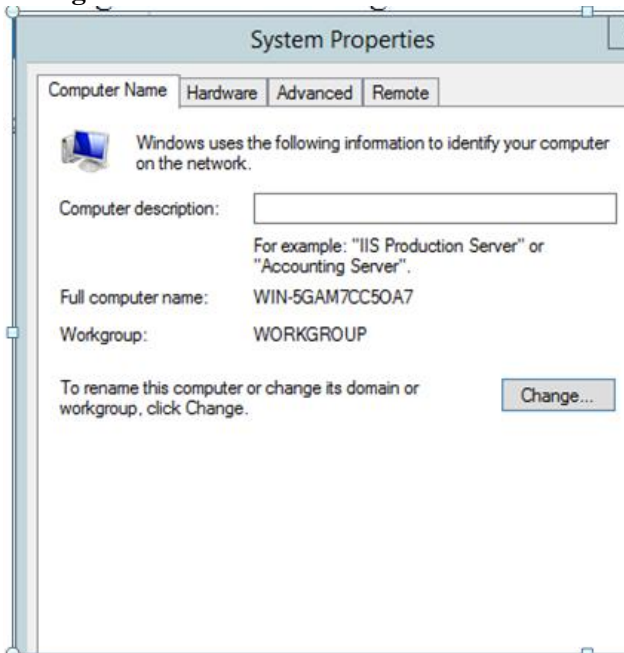
14. Enter a new computer name with your first, middle and last initials (ex: W12tlg)**,** click **OK** and **Restart Now**.

Activity 3-2. Windows Server 2012 R2 Server Manager and basic commands & navigation

When you start your Windows Server 2012 VM, it launches the Server Manager Dashboard.

.
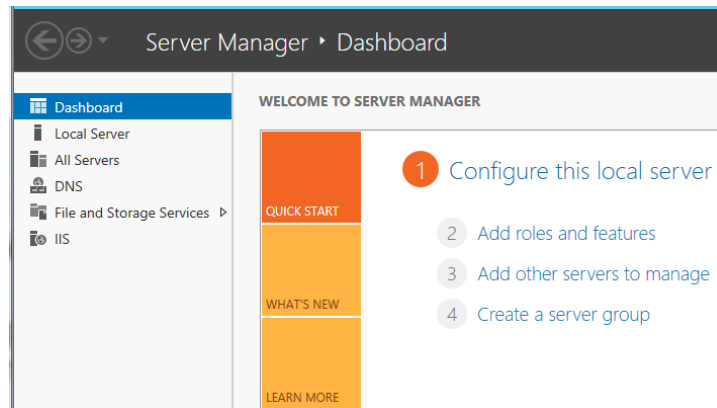
Server Manager uses this task bar icon

1. Choose **Local Server**. You've already setup your static IP Address and Computer name in your Local Server properties. Examine other Local Server properties including Processors, RAM and Total disk space. Make sure the 'IE Enhanced Security Config' is turned off.

2. Note that the right hand pulldown menu '**Manage**' is where you'll add Roles and Features in upcoming chapters.
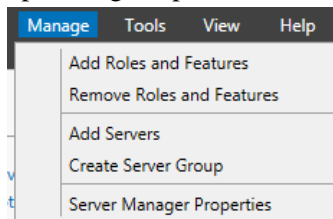
3. Click on the right hand pull-down menu '**Tasks**'. This lets you shut down your server, access PowerShell and access to other services. Run **PowerShell** and make a screenshot of the command **'whoami'**. Noticed that I have already installed the DNS and IIS Roles.
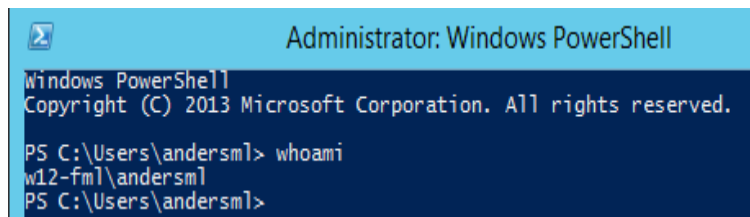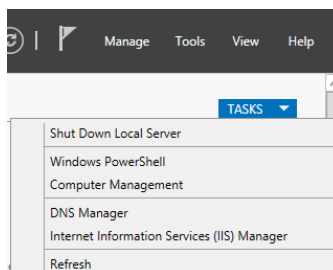
4. Click on right hand pull-own menu '**Tools**'. This gives you access to Administrative Tools. Run **Resource Monitor**, expand the **CPU** and **Memory** detail and take a screenshot.
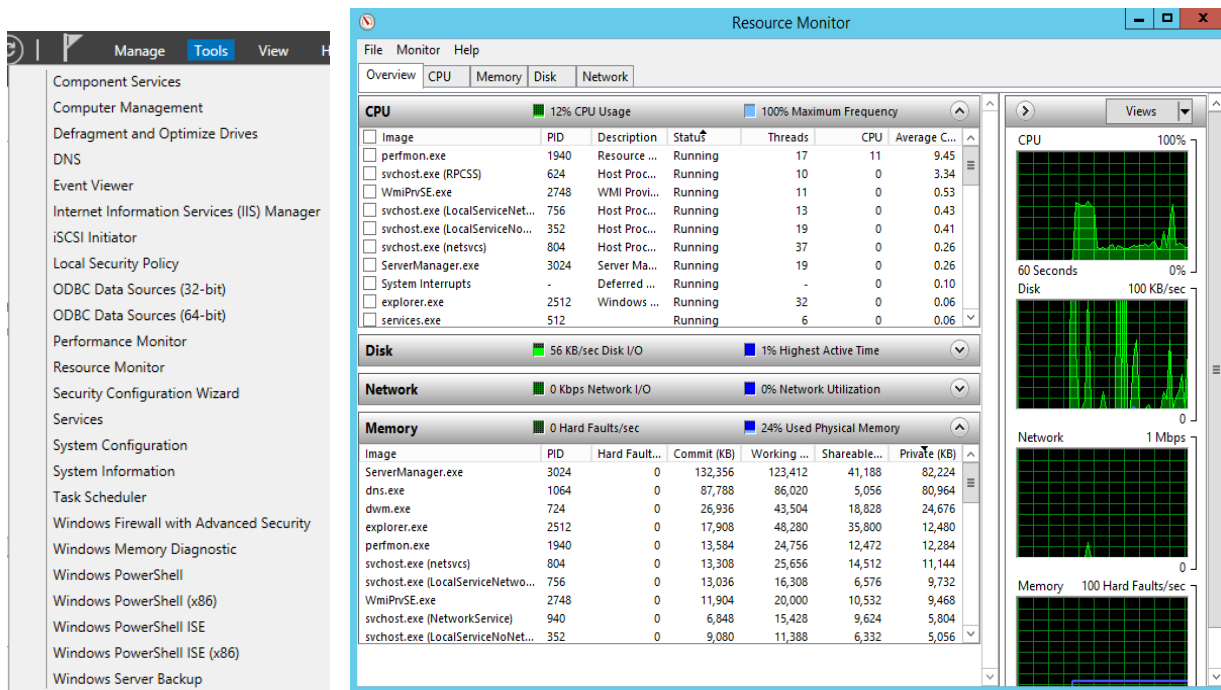
5. Minimize Server Manager and then click your start button multiple times. Notice that it toggles you back and forth between the tiled interface and the Desktop.
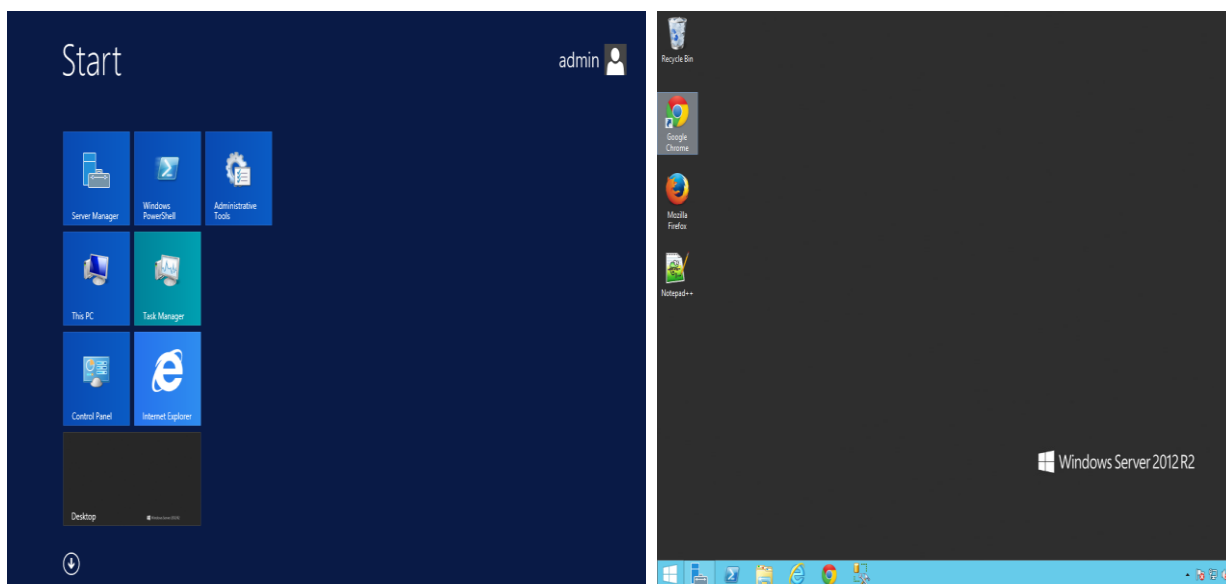
Activity 3-3. Install Ubuntu 14 on a VM

To install the Ubuntu 14 operating system

1. Start VMware Player (or VMware Workstation) and 'Create a new VM'
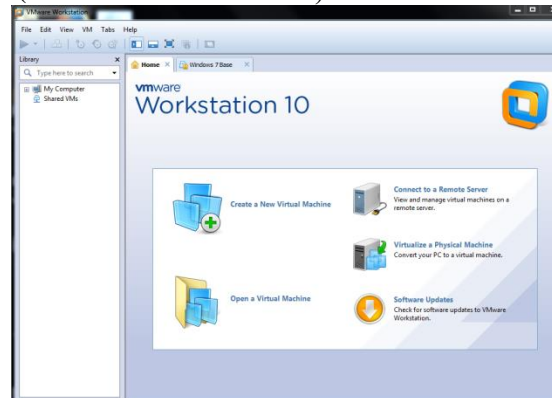
2. Select 'Installer disc image file' and browse to the Ubuntu v14 .iso file
   ubuntu-14.04-server-amd64.iso found at http://releases.ubuntu.com/14.04/
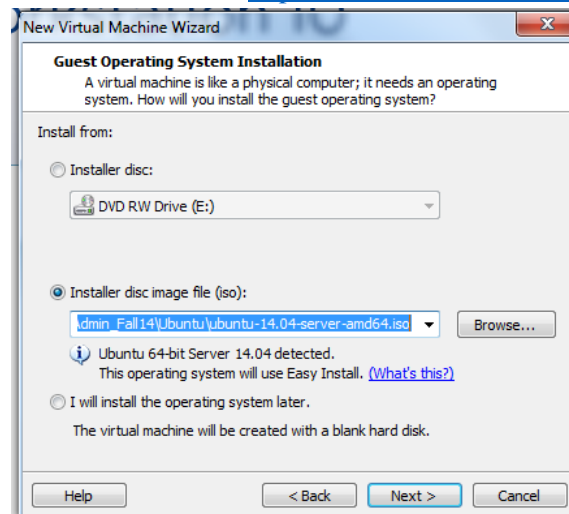
3. Next setup your UserName & Password, VMachinename & Location and Max disk size.
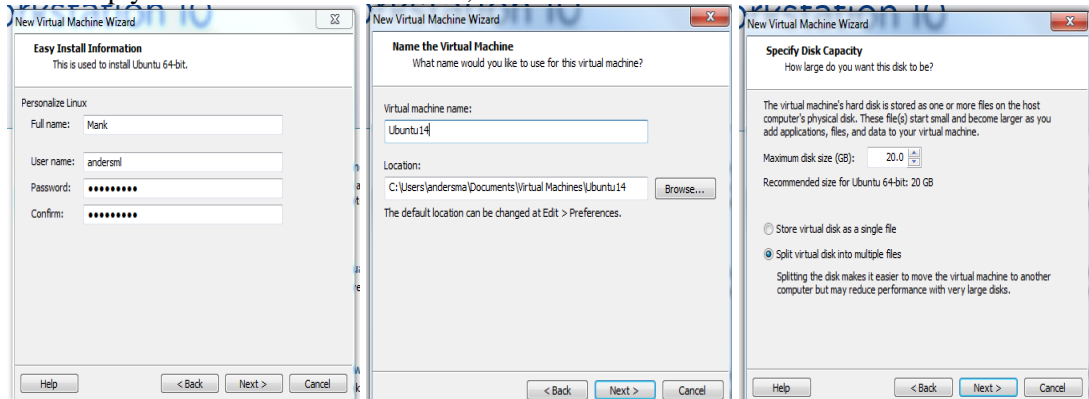
4. Login to your Ubuntu Server from the command prompt

```
Ubuntu 14.04 LTS ubuntu tty1

ubuntu login: andersml
Password:
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

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

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.

andersml@ubuntu:~$ _
```

5. Next perform updates and upgrades as follows .. (upgrade takes a while):
   a. **sudo apt-get update**
   b. **sudo apt-get upgrade**

```
andersml@ubuntu:~$ sudo apt-get update_

Get:25 http://us.archive.ubuntu.com precise-updates/main Translation-en [305 kB]
Hit http://security.ubuntu.com precise-security/main Translation-en
Hit http://security.ubuntu.com precise-security/multiverse Translation-en
Hit http://security.ubuntu.com precise-security/restricted Translation-en
Hit http://security.ubuntu.com precise-security/universe Translation-en
Hit http://us.archive.ubuntu.com precise-updates/multiverse Translation-en
Get:26 http://us.archive.ubuntu.com precise-updates/restricted Translation-en [2
,562 B]
Hit http://us.archive.ubuntu.com precise-updates/universe Translation-en
Hit http://us.archive.ubuntu.com precise-backports/main Translation-en
Hit http://us.archive.ubuntu.com precise-backports/multiverse Translation-en
Hit http://us.archive.ubuntu.com precise-backports/restricted Translation-en
Hit http://us.archive.ubuntu.com precise-backports/universe Translation-en
Fetched 2,407 kB in 6s (394 kB/s)
Reading package lists... Done
andersml@ubuntu:~$ _
```

```
andersml@ubuntu:~$ sudo apt-get upgrade_

Setting up perl (5.14.2-6ubuntu2.3) ...
Setting up apache2.2-common (2.2.22-1ubuntu1.4) ...
Setting up apache2-mpm-prefork (2.2.22-1ubuntu1.4) ...
 * Starting web server apache2
apache2: Could not reliably determine the server's fully qualified domain name,
using 127.0.1.1 for ServerName
                                                                          [ OK ]
Setting up apache2 (2.2.22-1ubuntu1.4) ...
Setting up libapache2-mod-php5 (5.3.10-1ubuntu3.7) ...
 * Reloading web server config apache2
apache2: Could not reliably determine the server's fully qualified domain name,
using 127.0.1.1 for ServerName
                                                                          [ OK ]
Processing triggers for libc-bin ...
ldconfig deferred processing now taking place
Processing triggers for initramfs-tools ...
update-initramfs: Generating /boot/initrd.img-3.5.0-23-generic
andersml@ubuntu:~$ _
```

6. Next install Ubuntu's GUI (this takes a while)
   sudo apt-get install ubuntu-desktop

```
andersml@ubuntu:~$ sudo apt-get install ubuntu-desktop_

INFO:softwarecenter.db.pkginfo_impl.aptcache:aptcache.open()
Software catalog update was successful.
Setting up ubuntu-artwork (57) ...
Setting up ubuntu-desktop (1.267.1) ...
Setting up indicator-printers (0.1.6-0ubuntu1) ...
Processing triggers for libreoffice-common ...
Setting up libreoffice-core (1:3.5.7-0ubuntu4) ...
Setting up libreoffice-base-core (1:3.5.7-0ubuntu4) ...
Setting up libreoffice-calc (1:3.5.7-0ubuntu4) ...
Setting up libreoffice-draw (1:3.5.7-0ubuntu4) ...
Setting up python-uno (1:3.5.7-0ubuntu4) ...
Setting up libreoffice-gtk (1:3.5.7-0ubuntu4) ...
Setting up libreoffice-gnome (1:3.5.7-0ubuntu4) ...
Setting up libreoffice-writer (1:3.5.7-0ubuntu4) ...
Setting up libreoffice-help-en-us (1:3.5.7-0ubuntu4) ...
Setting up libreoffice-impress (1:3.5.7-0ubuntu4) ...
Setting up libreoffice-math (1:3.5.7-0ubuntu4) ...
Processing triggers for libreoffice-common ...
Setting up libreoffice-emailmerge (1:3.5.7-0ubuntu4) ...
Processing triggers for libc-bin ...
ldconfig deferred processing now taking place
Processing triggers for libgdk-pixbuf2.0-0 ...
Processing triggers for initramfs-tools ...
update-initramfs: Generating /boot/initrd.img-3.5.0-23-generic
andersml@ubuntu:~$ _
```

7. Reboot and login to Ubuntu Server from the desktop & take a screenshot of the desktop.
sudo reboot

Search for **terminal** and drag & drop it on the left navigation taskbar.

8. Change the computer name from ubuntu to U14-fml. To interrogate your hostname, type:
sudo hostname

```
andersml@ubuntu:~$ sudo hostname
[sudo] password for andersml:
ubuntu
```

9. Change your hostname from ubuntu to u14-fml (where fml is your first, middle & last initials) with these commands
    a. **sudo nano /etc/hostname**

```
  GNU nano 2.2.6              File: /etc/hostname

u14-fml
```

    b. **sudo nano /etc/hosts**

```
  GNU nano 2.2.6                File: /etc/hosts

127.0.0.1       localhost
127.0.1.1       u14-fml

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

    c. **sudo reboot**

10. Verify your changes hostname and **take a screenshot**
sudo hostname

```
  andersml@u14-fml: ~
andersml@u14-fml:~$ sudo hostname
[sudo] password for andersml:
u14-fml
```

Next we need to create a static IP address. Start with an ifconfig to determine your VM's dhcp assigned IP address (after "inet addr:").

```
andersml@ubuntu:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:5f:c7:fa
          inet addr:192.168.136.129  Bcast:192.168.136.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe5f:c7fa/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1660 errors:0 dropped:0 overruns:0 frame:0
          TX packets:913 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1714121 (1.7 MB)  TX bytes:94483 (94.4 KB)
```

11. Copy the example below from your text and paste it into the interfaces file. You must have VMware Tools installed for copy/paste between host and VM to work. Use SHIFT+INS to paste text into the nano editor. Notice the changes, including iface for eth0 to static (# is the comment character), **use your address from ipconfig, netmask is unchanged, network, broadcast, and gateway use the third octet from your address**. Use this nano syntax: **sudo nano /etc/network/interfaces. The first dns-nameserver address is your address.**

#This file describes the network interfaces available on your system	
# and how to activate them. For more information, see interfaces(5).	
# The loopback network interface	
auto lo	
iface lo inet loopback	
# The primary network interface	
auto eth0	
#iface eth0 inet dhcp	
iface eth0 inet static	
address	192.168.136.129
netmask	255.255.255.0
network	192.168.136.0
broadcast	192.168.136.255
gateway	192.168.136.2
dns-nameservers	192.168.136.129 8.8.8.8

Edit the content of the interfaces file using the content in your clipboard and type **CTRL+X** to save the file.

12. Next uninstall the network-manager and reboot
    a. **sudo apt-get remove network-manager**
    b. **sudo reboot**

13. Finally verify that your static address is active by pinging both your static address and Google's public DNS 8.8.8.8. Break with CTRL+C and take a screenshot of your successful ping.

```
andersml@ubuntu:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=128 time=42.8 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=128 time=42.4 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 42.473/42.677/42.882/0.290 ms
andersml@ubuntu:~$ ping 192.168.136.129
PING 192.168.136.129 (192.168.136.129) 56(84) bytes of data.
64 bytes from 192.168.136.129: icmp_seq=1 ttl=64 time=0.040 ms
64 bytes from 192.168.136.129: icmp_seq=2 ttl=64 time=0.042 ms
64 bytes from 192.168.136.129: icmp_seq=3 ttl=64 time=0.040 ms
```

Activity 3-4. Ubuntu 14 basic commands & navigation

Summary of Linux commands used in Chapter 3 Activities

Command	Description .. source http://ss64.com/bash/
apt-get	Search for and install software packages (Debian/Ubuntu)
sudo	Gives root privileges from terminal (superuser)
gksu	GUI form of sudo
nano	Terminal editor
gedit	GUI editor with folder navigation
cat	Concatenate and print (display) the content of files
hostname	Print or set system name
reboot	Reboot the system
ifconfig	Configure a network interface
ping	Test a network connection

The Ubuntu GUI has its task bar on the left which includes from top to bottom Search, gedit, terminal, Firefox, Libre's Write Calc & Impress, Ubuntu Software Center, Amazon, System Settings & the floppy disk. You'll be impressed when you try these out. Notice Ubuntu's pull down menu on the right including System Settings and Shut Down. Notice that in the gedit example below, the close min & max icons and the app menu in the app's top left. Also notice that the root folders in gedit are under 'Computer'. In this class we edit files in /etc and /var so spend some time looking at their subfolders.
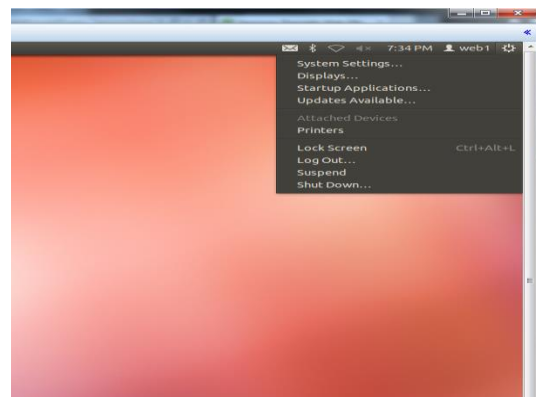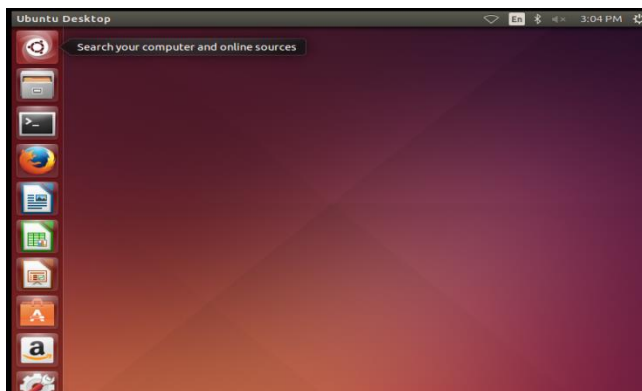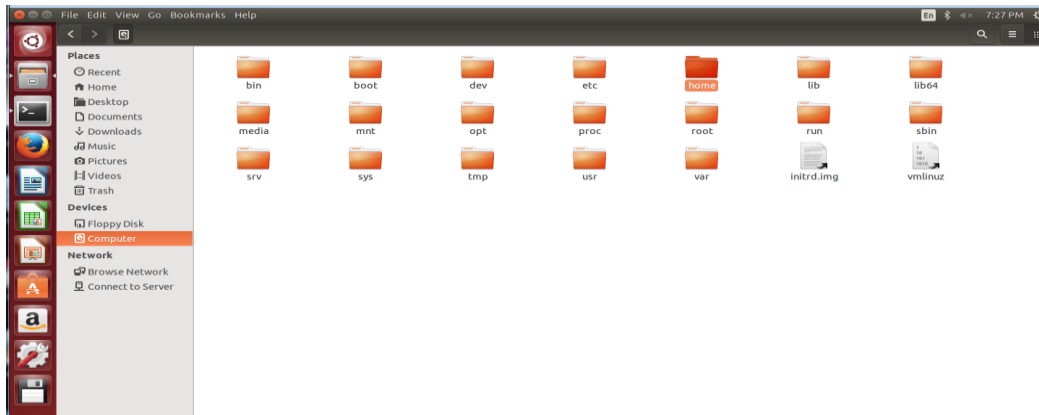
Here are the steps for editing the interfaces file using **GUI tool gedit** instead of the **terminal prompt tool nano**.

1. Install gksu
   sudo apt-get install gksu

2. Start gedit with gksu and open /etc/network/interfaces
   gksu gedit
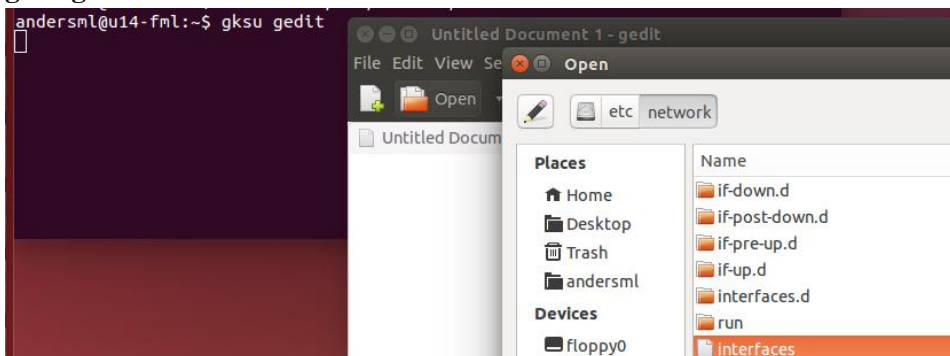
3. Write the comment on the last line: **# I edited interfaces with gedit!** and save the file. Submit this screenshot.

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

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0

#iface eth0 inet dhcp
iface eth0 inet static
address                         192.168.136.129
netmask                         255.255.255.0
network                         192.168.136.0
broadcast                       192.168.136.255
gateway                         192.168.136.2
dns-nameservers                 192.168.136.129  8.8.8.8

# I edited interfaces with gedit!
```

Here is a good youtube tutorial on static IP addresses assignment in Ubuntu

www.youtube.com/watch?v=Uabfu_RJLyg

AN INTRODUCTION TO LINUX COMMANDS

If this occasion marks your first exposure to Linux, you need to know enough commands to perform basic functions such as editing files and finding directories. When you use Linux as a Web server, you edit only a few files.

The shell interface is very similar to the command-line interface in Windows. However, some notable differences exist:

- Linux has no drive letters; in Windows, different partitions have different drive letters, such as C: and D:.
- In Linux, the root of the disk is /. Even though you may create different partitions for /home and /usr, all of them still start at /.
- In Linux, you use the forward slash (/) instead of the backward slash (\) that you use in Windows.

The Linux file system is organized in a hierarchy that is similar to the Windows file hierarchy. Shell commands typically require a reference to the file system, so it is a good idea to understand how you refer to files and directories. For example, suppose you need to edit, move, or copy a file called products.html, which is found in /var/www/research. The exact location of this file is called the **path**. The complete path to the previous file, including the file itself, is /var/www/research/products.html.

You can use this path with any command that requires a reference to the file. The "/" at the beginning of the path means to start at the root of the drive. If you do not have a "/" at the beginning, the path begins at the current position in the directory structure. For example, if you are working in the /var/www directory and you want to reference the products.html file, you would type research/products.html.

You can also use two dots in your command (..) to move up one directory in the structure. For example, if you were working in the /var/www/html directory and you wanted to reference the products.html file, you could type ../research/products.html.

Armed with this introduction to paths, you are ready to learn some useful commands:

- *ls*—List the contents of a directory. You can type ls -a to list hidden files and ls -l to see all the characteristics of a file, including the file permissions. You can use the asterisk (*) wildcard character to represent one or more characters. For example, to display all .conf files, you would type ls *.conf. You can also combine options, as in ls -a -1*.conf.

- *cd*—Change the directory. If you need to perform a number of operations in a single directory, such as editing multiple files, it might be easier to move to that directory and then start the editor. For example, to reach the www directory, which is below /var, you would type cd /var/www.To move to the html directory, which is below /var/www, you could type cd /var/www/html or cd html.

- *mkdir*—Make (create) a directory. For example, you may need to create a directory in your Web site for HTML files. To create a directory called secure in the /var/www/html directory, you could type mkdir /var/www/html/secure. If you were already in the /var/www/html directory, you could simply type mkdir secure to create the directory.

- *rmdir*—Remove a directory. If you created a directory called secure with the previous command and then decided to remove it, you could type rmdir secure, assuming that you were already in /var/www/html.

▪ *mv*—Move and/or rename a file. To move a file called ftpaccess from /etc to /var/ftp, you could type mv /etc/ftpaccess /var/ftp/ftpaccess. If you were in the /var/www/html directory and just finished editing a file called info.tml, you could rename it by typing mv info.tml info.html.

▪ *cp*—Copy a file. This command is like the mv command, except that it does not delete the original copy. For example, to copy a file called ftpaccess from /etc to /var/ftp, you could type cp /etc/ftpaccess /var/ftp/ftpaccess. The cp command is useful when you want to make a backup file before you start editing a configuration file. Assuming that you were in the /etc directory, you could make a backup copy of ftpaccess before editing it by typing cp ftpaccess ftpaccess.backup.To copy /etc/ftpaccess to the floppy disk, you would type cp /etc/ftpaccess /mnt/floppy/ftpaccess.

▪ *locate*—Find a file if you forget where it is located. For example, to find a file called httpd.conf, you would type locate httpd.conf.

▪ *gedit*—Edit a file. This command is used in the Ubuntu GUI environment and assumes that the Ubuntu Desktop has been installed. It allows you to edit an existing file or create a file to edit. To edit /etc/ftpaccess, for example, you would type sudo gedit /etc/ftpaccess. If the ftpaccess file did not exist, this command would create it.

Many other shell commands exist, but the preceding list covers the most important ones. Other commands with special purposes, such as monitoring programs that are running, are covered in a later chapter.

CHAPTER SUMMARY
❏ Naming computers is similar in both Windows and Linux. It is a good idea to keep the names simple and descriptive. Although Linux allows two computers to have the same name, it is nevertheless a good practice to use unique names.
❏ Windows and Linux are licensed in different ways. Windows product musts be activated by Microsoft after installation to avoid piracy constraints. Linux is based on the GNU general public license, so the product is free.
❏ Installing Microsoft operating systems has become easier with each new product. VMWare has also streamlined its Linux installation process.
❏ Windows Server Manager provides a one-stop management interface with tools for querying the server status and changing roles and features.
❏ In Linux, you use basic commands such as ls to list the contents of a directory, mkdir to create a directory, rmdir to remove a directory, mv to move or rename a file, cp to copy a file, cd to change directories, and locate to find files.
❏ Before installing a DNS server, you need to create a static IP Address.

Complete Activities 3.1 through 3.4

Activity 3.1 Windows Server 2012 Install

Submit screenshot of the completed installation (step 7), the Static IP assignment (step 10 - ping) and the Computer name (step 12 - Server Manager, Local Server property).

Activity 3.2 Windows Server 2012 basic commands and navigation.

Submit screenshot of 'whoami' (step 3) and Resource Monitor (step 4).

Activity 3.3 Ubuntu 14 Install

Submit screenshot of the completed installation (step 7), the Computer name (step 10 - sudo hostname) and the Static IP assignment (step 14 - ping.

**Activity 3.4 Ubuntu 14 basic commands and navigation**

Submit screenshot of your gedited interfaces (step 3)

Hands On Projects
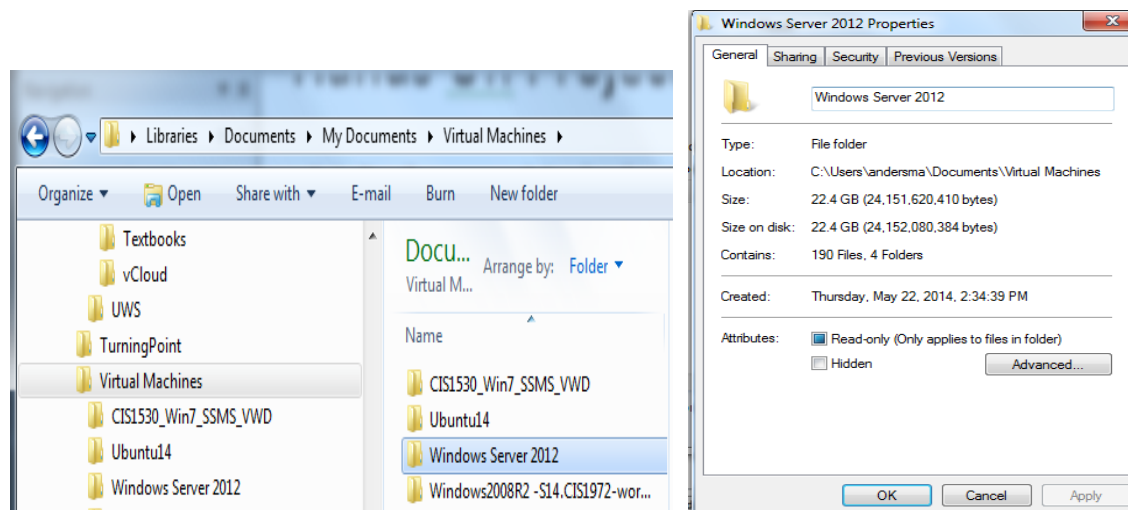
3.1 Performance Monitoring W2012 and backup.

Resource Monitor (resmon) measures the four most important dimensions of performance: CPU, Disk, Network and Memory. Resmon lets you manage your machine performance by monitoring tasks such that CPU & Memory don't exceed 70% on either your host or guest system. Also with resmon performance stats, you can use the Virtual Machine Settings to balance CPU and Memory resources between your host and guest systems.

1. Run Resource Monitor (resmon) on your host and on your Windows Server guest and make note of the CPU usage & Physical Memory used.
2. Go to your Virtual Machine Settings, fine tune your VM and screenshot your settings.
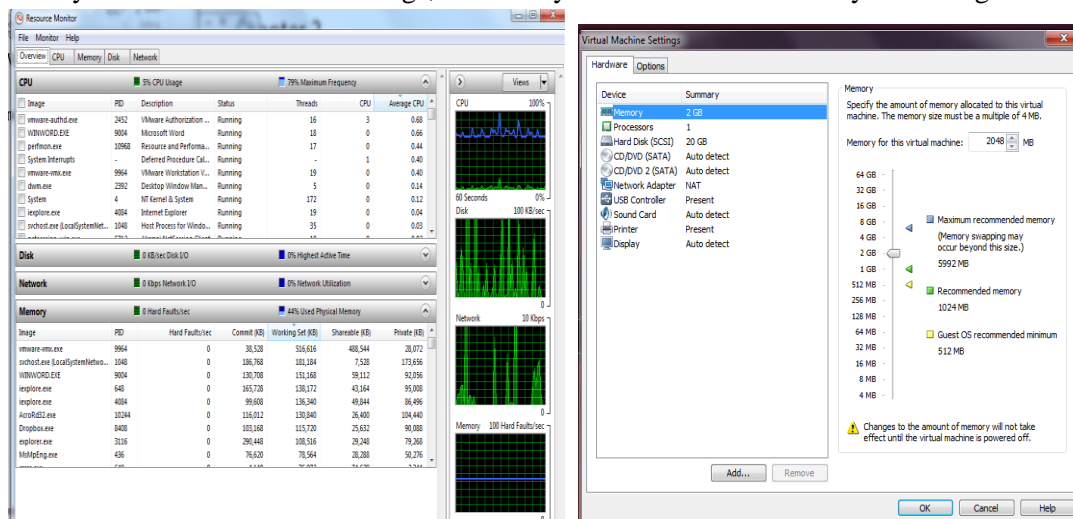
By default your virtual machine (VM) can be found in \My Documents\Virtual Machines. If you back up your VM (the entire folder) at the end of each chapter, you'll never risk losing more than one chapter of work.

3. Backup your Windows Server VM and make a screenshot of the properties of your backup folder.

3.2 Performance Monitoring Ubuntu v14 and backup

1. Run resmon (or use the performance tab on Task Manager) on your host machine and make note of the CPU & RAM usage.
2. Run gnome-system-monitor in your Ubuntu VM and make note of the CPU & RAM usage
3. Go to your Virtual Machine Settings, fine tune your VM and make a screenshot of your settings
4. Backup your Ubuntu VM and make a screenshot of the properties of your backup folder.
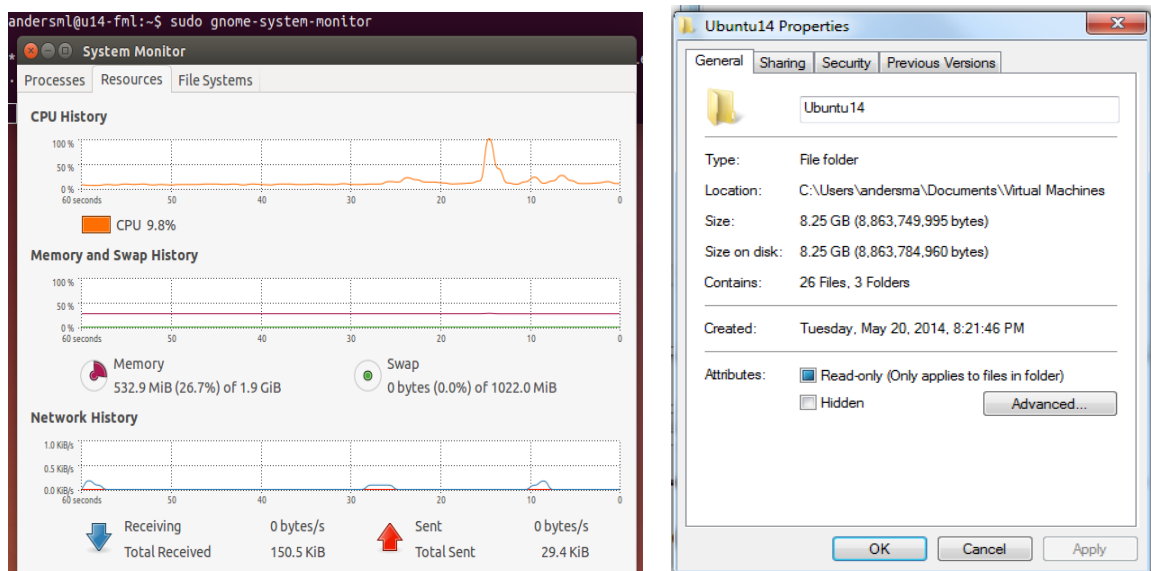
You'll start experiencing performance degradation when either the CPU or Memory usage is greater than 70%. Throughout the remainder of the course, you'll use these resource monitoring techniques to manage Web Server performance on your VM.

3.3 Linux Command Line Utilities

Start the terminal command and try the following commands. See your worksheet for screenshot instructions.

- **pwd** - displays your current working directory (default directory)
- **cd ../../usr/bin** - changes your current working directory to /usr/bin
- **ls -** (stands for list) displays the contents of your current working directory
- **ls -l** - (that's a lower case L at the end) displays a long listing of the current directory. Note that lines beginning with a 'd' are directories and lines beginning with a '-' are files.
- **cd** - changes your current working directory back to your home directory
- **mkdir testdir** - creates a directory named 'testdir' in the current directory
- **ls -l** - note that the 'testdir' directory is now created in your home directory
- **touch newfile** - creates a new, empty file named 'newfile'
- **ls -l** - note that the 'newfile' file is now created in your home directory
- **rmdir testdir** - removes the directory named 'testdir' (how can you look to see if it's really gone?
- **rm newfile -** removes the file 'newfile' (how can you look to see if it's really gone?)
- **touch file1 file2 file3** - creates three new files named 'file1' file2' and 'file3'
- **touch file10 file20 file30** - create three files named 'file10' file20' file30'
- **ls file?** - list all files starting with the word file and any single letter following
- **ls file*** - list all files starting with the word file and any number of letters following (note: the ? and * characters are called 'wild cards' because they do pattern matching)
- **ls file1*** - explain the results of this command
- **ls file1?** - Explain the results of this command
- **mv file10 file100** - What does this command do? How can you verify the results?
- **rm file1?** - Explain the results of this command
- **rm file*** - Explain the results of this command

Go to D2L **Discussions** to respond to Ch 3 Technical Tips and **Quizzes** to complete the Ch 3 Review Questions. You have three attempts at the review questions and your score is the average of all three.