

Learning Journal for:

General Description:

This learning journal 1 covers all the Network Servers lectures and labs that is required to be completed. Specifically covering the commands and processes that are needed in order to complete the required tasks.

Intent:

This Journal has intent to create a learning portfolio that allows me to remember the step and information needed to complete the process. Due to the tasks being extensive and complicated, a detailed journal will need to be made to ensure that I could complete the tasks by referring to this journal. It will include all the content learnt throughout the semester from week 1 to week 5 for 31338 Network Servers.

Week 1

Lecture 1 Introduction and System Startup

Introduction

- The role of the system administrator
Used to maintain up-to-date knowledge of issues (e.g., backing up & restoring file, securing system, configuring and installing server, maintaining accounts)
- Boot processes
 1. The server **firmware** starts, to quick check the hardware.
 2. The **boot loader** choose which kernel to run.
 3. The **kernel** program loads into memory; prepares the system
 4. The **initialization** process starts the necessary background programs
- Init/systemd
Init/systemd is the first process to run in a UNIX system (PID=1)
Sys V init processes instructions **from /etc/inittab;** typically starts system daemons (background processes) & gettys (terminals for login activity)
- Systemd
Systemctl utility to manage system services
Command = disable, enable, restart, start, status, stop, reload, is-active, is-enabled, is-failed
Unit-Name: firewalld, ntpd, dhcpcd
- Service Unit Files **.service** (information used when a service started)
systemctl list-unit-files (display unit file name and state)
systemctl cat xxx.service (display more information)
- Target Unit Files **.target** (services to start at system boot time)
systemctl list-units (confirm running units)
systemctl get-default (graphical.target)
systemctl cat graphical.target

- **COMMAND** = get-default, set-default, isolate (jump)
 - poweroff.target (power offs the system)
 - rescue.target (system recovery)
 - multi-user.target (multi-user mode)
 - emergency.target (emergency system recovery)
 - graphical.target (full graphical mode)
 - reboot.target (shuts down and boots the system again)
- Runlevels and services

COMMAND	
- grep :initdefault: /etc/inittab	- telinit q
- runlevel (See current runlevel)	- service cups start
- telinit 4 (Modify runlevel)	- service httpd start
- init 4 (Modify runlevel)	
- Log files

Kernel ring buffer	
- dmesg less (write kernel messages in <i>/var/log/dmesg</i> or <i>/var/log/boot.log</i>)	
- journalctl (display logs by system)	
Main system log file – <i>/var/log/messages</i>	
Other log files - <i>/var/log</i>	

Linux Certifications

- LPI C: Linux Professional Institute (LPIC -1, -2, -3)
- RHCSA: Red Hat Certified System Administrator

Windows Certifications

- MCS A, MCS, MCS D and MCSE retried
- Roles: Developer, Administrator, Solution Architect, etc.

Lab 1a VMware Lab familiarization

Logging in as Root and Shutting Down your Linux Image

Log in as the root user (superuser). For username: “**root**” and password: “**student123!**”
Never create a **simple** password.



Attempt several commands based on **Unix commands** from Web Systems.

```
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
        loop txqueuelen 1000 (Local Loopback)
        RX packets 340 bytes 29500 (28.8 KiB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 340 bytes 29500 (28.8 KiB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

virbr0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 192.168.122.1 netmask 255.255.255.0 broadcast 192.168.122.255
        ether 52:54:00:56:eb:e4 txqueuelen 1000 (Ethernet)
        RX packets 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Checking the network by typing *ifconfig*.

What do you see? Can you determine if there is a default tcip address?

Yes, there are 2 interface which are loopback and virbr0. The default ip address for lo is 127.0.0.1 with netmask 255.0.0.0. While for the virbr0 interface the default ip address is 192.168.122.1 with netmask 255.255.255.0

How about the ls command?

```
[root@localhost ~]# ls
anaconda-ks.cfg  Desktop   Downloads      Music      Public     Videos
Backup          Documents initial-setup-ks.cfg Pictures  Templates
[root@localhost ~]# pwd
/root
```

The *ls* command shows the list of files or directory in this Linux.

Logging on as root

Run the privileged commands by using *sudo* command or the “superuser” command *su*.

```
[root@localhost ~]# su
```

While trying to use the *su* command, there is no change on prompt since we already in the root. Usually, it will prompt for the root password if changes happen.

Reboot the system.

There are various commands to do this, but the simplest under these circumstances is *reboot*.

```
[root@localhost ~]# reboot
```

Why would a system admin not normally just reboot?

The system has the potential to corrupt the data and potentially damage the hardware. It also doesn't work when logged into your favorite remote server.

Then when you are ready, shut down your virtual machine with:

shutdown -h now

```
[root@localhost ~]# shutdown -h now
```

Why do you think it is important to learn to shutdown from the command line?

Knowing how to do things on the command-line allows you to be as stack agnostic as possible. You can switch easily between hosting providers and always have the same consistent set of tools at your disposal.

Lab 1b System Startup, Runlevels and Log Files

Task 1: Boot Single-User Shell only, Using the Boot Loader

There are 4 processes to be followed to boot in single user mode:

1. The GRUB2 boot loader shows the kernels then press the key 'E' to modify the commands.

```
CentOS Stream (4.18.0-408.e18.x86_64) 8
CentOS Linux (4.18.0-348.7.1.e18_5.x86_64) 8
```

2. Look for the word "*ro*" and replace it with: "*rw init=/sysroot/bin/sh*"

```
linux ($root)/vmlinuz-4.18.0-193.6.3.e18_2.x86_64 root=/dev/mapper/cl-root rw \
init=/sysroot/bin/sh resume=/dev/mapper/cl-swap rd.lvm.lv=cl/root rd.lvm.lv=cl\
```

3. Press *Ctrl-X* to boot the VM with the new parameters.

```
Generating "/run/initramfs/rdsosreport.txt"

Entering emergency mode. Exit the shell to continue.
Type "journalctl" to view system logs.
You might want to save "/run/initramfs/rdsosreport.txt" to a USB stick or /boot
after mounting them and attach it to a bug report.

:/# ls
bin  dracut-state.sh  etc  kernel  lib64  root  sbin  sys  tmp  var
```

VM will boot with a root (single-user) shell that doesn't require a password. Single-user boot is basically used for system maintenance, when need to be sure that no-one else is using the system. It is also the way to do password recovery on a Linux system using GRUB2.

If the attacker has physical access, they able do the same thing. However, there is also a 'rescue mode' that gives a single-user environment without require the root password.

Task 2: Explore and modify system startup scripts

There are different modes the system can boot in such as single-user mode is one; multi-user mode is second; graphical mode is a third. These different modes called "runlevels" or now called as "targets".

```
[root@localhost ~]# runlevel
N 5
```

Try to check the **runlevel** and it showed that it is in "N 5".

Use the command **systemctl get-default** to check the current default mode. Or change it with **systemctl set-default**.

```
[root@localhost ~]# systemctl get-default
graphical.target
```

The main targets are:

- **emergency.target** – emergency system recovery
- **rescue.target** – rescue mode for system recovery
- **multi-user.target** – multi-user mode
- **graphical.target** – full graphical mode

Move to multi-user mode with command: **systemctl isolate multi-user.target**

```
[root@localhost ~]# systemctl isolate multi-user.target
```

```

CentOS Stream 8
Kernel 4.18.0-193.6.3.el8_2.x86_64 on an x86_64

Activate the web console with: systemctl enable --now cockpit.socket

localhost login: root
Password:
Last login: Mon Aug 29 00:59:51 on tty2
[root@localhost ~]# runlevel
5 3
[root@localhost ~]# systemctl is-system-running
degraded
[root@localhost ~]# systemctl is-active
Too few arguments.
[root@localhost ~]# systemctl is-active multi-user.target
active
[root@localhost ~]# 

```

Go back to graphical mode with: *systemctl isolate graphical.target*

```

[root@localhost ~]# systemctl isolate graphical.target
[root@localhost ~]# runlevel
3 5
[root@localhost ~]# systemctl is-active graphical.target
active
[root@localhost ~]#

```

Try change to emergency mode with: *systemctl isolate emergency.target*

```

[root@localhost ~]# systemctl isolate emergency.target
You are in emergency mode. After logging in, type "journalctl -xb" to view
system logs, "systemctl reboot" to reboot, "systemctl default" or "exit"
to boot into default mode.
Give root password for maintenance
(or press Control-D to continue):
[root@localhost ~]# runlevel
N N
[root@localhost ~]# systemctl is-active emergency.target
active
[root@localhost ~]#

```

Try change to rescue mode with: *systemctl isolate rescue.target*

```

[root@localhost ~]# systemctl isolate rescue.target
You are in rescue mode. After logging in, type "journalctl -xb" to view
system logs, "systemctl reboot" to reboot, "systemctl default" or "exit"
to boot into default mode.
Give root password for maintenance
(or press Control-D to continue):
[root@localhost ~]# runlevel
5 1
[root@localhost ~]# systemctl is-active rescue.target
active
[root@localhost ~]#

```

systemctl used to manages the startup process. Explore **services** by command: *systemctl list-unit-files*

UNIT FILE	STATE
proc-sys-fs-binfmt_misc.automount	static
-.mount	generated
boot.mount	generated
dev-hugepages.mount	static
dev-mqueue.mount	static
proc-fs-nfsd.mount	static
proc-sys-fs-binfmt_misc.mount	static
run-vmblock\x2dfuse.mount	enabled
sys-fs-fuse-connections.mount	static
sys-kernel-config.mount	static
sys-kernel-debug.mount	static
tmp.mount	disabled
var-lib-machines.mount	static
var-lib-nfs-rpc_pipefs.mount	static
cups.path	enabled
ostree-finalize-staged.path	disabled
systemd-ask-password-console.path	static
systemd-ask-password-plymouth.path	static
systemd-ask-password-wall.path	static
session-6.scope	transient
abrt-ccpp.service	enabled
abrt-journal-core.service	disabled

lines 1-23

It shows a list of all units can be targets or services. For services, it will show whether it's enabled or disabled

Is the “sshd” service enabled or disabled by default? What about the “httpd” service?

sshd.service	enabled
sshd@.service	static

The sshd service is **enabled** by **default**.

httpd.service	disabled
httpd@.service	disabled

The httpd service is **disabled** by **default**.

Another way to check whether is to use: *systemctl is-enabled sshd*

Enabled/disabled means whether the service will/won't start automatically. See whether a service is active:
systemctl is-active sshd

```
[root@localhost ~]# systemctl is-enabled sshd
enabled
[root@localhost ~]# systemctl is-active sshd
active
[root@localhost ~]#
```

The system is **enabled** and **active**.

Modify the activity by:

```
systemctl start sshd
systemctl stop sshd
systemctl enable sshd
systemctl disable sshd
```

```
[root@localhost ~]# systemctl stop sshd
[root@localhost ~]# systemctl is-enabled sshd
enabled
[root@localhost ~]# systemctl is-active sshd
inactive
[root@localhost ~]# systemctl disable sshd
Removed /etc/systemd/system/multi-user.target.wants/sshd.service.
[root@localhost ~]# systemctl is-enabled sshd
disabled
```

Task 3: Examine system log information

Use the *dmesg* command to view the contents of the kernel ring buffer.

What kinds of information do you see? What about if you run the command *journalctl --dmesg*?

```
[root@localhost ~]# dmesg
[    0.000000] Linux version 4.18.0-193.6.3.el8_2.x86_64 (mockbuilder@kbuilder.bsys.centos.org) (gcc version 8.3.1 20191121 (Red Hat 8.3.1-5) (GCC)) #1 SMP Wed Jun 10 11:09:32 UTC 2020
[    0.000000] Command line: BOOT_IMAGE=(hd0,msdos1)/vmlinuz-4.18.0-193.6.3.el8_2.x86_64 root=/dev/mapper/cl-root ro resume=/dev/mapper/cl-swap rd.lvm.lv=cl/root rd.lvm.lv=cl/swap rhgb quiet
[    0.000000] [Firmware Bug]: TSC doesn't count with P0 frequency!
[    0.000000] x86/fpu: Supporting XSAVE feature 0x001: 'x87 floating point registers'
[    0.000000] x86/fpu: Supporting XSAVE feature 0x002: 'SSE registers'
[    0.000000] x86/fpu: Supporting XSAVE feature 0x004: 'AVX registers'
[    0.000000] x86/fpu: xstate_offset[2]: 576, xstate_sizes[2]: 256
[    0.000000] x86/fpu: Enabled xstate features 0x7, context size is 832 bytes, using 'compacted' format.
[    0.000000] BIOS-provided physical RAM map:
[    0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009ebff] usable
[    0.000000] BIOS-e820: [mem 0x000000000009ec00-0x000000000009ffff] reserved
[    0.000000] BIOS-e820: [mem 0x00000000000dc000-0x00000000000fffff] reserved
[    0.000000] BIOS-e820: [mem 0x000000000100000-0x0000000007fedffff] usable
[    0.000000] BIOS-e820: [mem 0x0000000007fee000-0x0000000007fefeffff] ACPI NVS
[    0.000000] BIOS-e820: [mem 0x0000000007fefff000-0x0000000007fefffff] ACPI NVS
[    0.000000] BIOS-e820: [mem 0x0000000007ff00000-0x0000000007fffffff] usable
```

The *dmesg* displays kernel-related messages from the kernel ring buffer. The ring buffer stores many information and messages from kernel modules that take place during system startup.

journctl -dmesg

```
[root@localhost ~]# journalctl --dmesg
-- Logs begin at Mon 2022-08-29 01:09:06 AEST, end at Mon 2022-08-29 01:35:56 A>
Aug 29 01:09:06 localhost.localdomain kernel: Linux version 4.18.0-193.6.3.el8_5
Aug 29 01:09:06 localhost.localdomain kernel: Command line: BOOT_IMAGE=(hd0,msd)
Aug 29 01:09:06 localhost.localdomain kernel: [Firmware Bug]: TSC doesn't count
Aug 29 01:09:06 localhost.localdomain kernel: x86/fpu: Supporting XSAVE feature
Aug 29 01:09:06 localhost.localdomain kernel: x86/fpu: Supporting XSAVE feature
Aug 29 01:09:06 localhost.localdomain kernel: x86/fpu: Supporting XSAVE feature
Aug 29 01:09:06 localhost.localdomain kernel: x86/fpu: xstate offset[2]: 576, >
Aug 29 01:09:06 localhost.localdomain kernel: x86/fpu: Enabled xstate features >
Aug 29 01:09:06 localhost.localdomain kernel: BIOS-provided physical RAM map:
```

Journalctl is used to filter the dmesg command.

Also examine the file */var/log/messages*. *What kind of messages do you find in this file?*

```
[root@localhost ~]# head /var/log/messages
Aug 25 11:24:01 localhost systemd[1]: Starting dnf makecache...
Aug 25 11:24:02 localhost dnf[3234]: CentOS Stream 8 - AppStream
  0.0 B/s |  0 B   00:00
Aug 25 11:24:02 localhost dnf[3234]: Errors during downloading metadata for repository 'appstream':
Aug 25 11:24:02 localhost dnf[3234]: - Curl error (6): Couldn't resolve host name for http://mirrorlist.centos.org/?release=8-stream&arch=x86_64&repo=AppStream
&infra=stock [Could not resolve host: mirrorlist.centos.org]
Aug 25 11:24:02 localhost dnf[3234]: Error: Failed to download metadata for repo 'appstream': Cannot prepare internal mirrorlist: Curl error (6): Couldn't resolve host name for http://mirrorlist.centos.org/?release=8-stream&arch=x86_64&repo=AppStream&infra=stock [Could not resolve host: mirrorlist.centos.org]
```

It shows more complete message than the dmesg which only show the contents of the kernel ring buffer.

What about */var/log/secure*?

```
[root@localhost ~]# cat /var/log/secure
Aug 25 13:07:33 localhost gdm-password][5577]: gkr-pam: unlocked login keyring
Aug 25 13:09:52 localhost groupadd[5697]: group added to /etc/group: name=dhcpd,
  GID=177
Aug 25 13:09:52 localhost groupadd[5697]: group added to /etc/gshadow: name=dhcpd
Aug 25 13:09:52 localhost groupadd[5697]: new group: name=dhcpd, GID=177
Aug 25 13:09:53 localhost useradd[5709]: new user: name=dhcpd, UID=177, GID=177,
  home=/, shell=/sbin/nologin
Aug 25 13:50:45 localhost gdm-password][39718]: gkr-pam: unlocked login keyring
Aug 28 23:41:49 localhost polkitd[1097]: Loading rules from directory /etc/polki
```

The command shows only for the hidden security message and no message from the kernel.

journctl COMMAND:

journctl --since "10 minutes ago"

```
[root@localhost ~]# journalctl --since "10 minutes ago"
-- Logs begin at Mon 2022-08-29 01:09:06 AEST, end at Mon 2022-08-29 01:43:38 A>
Aug 29 01:35:56 localhost.localdomain org.gnome.Terminal.desktop[4903]: # watch>
Aug 29 01:35:56 localhost.localdomain org.gnome.Terminal.desktop[4903]: # watch>
Aug 29 01:35:56 localhost.localdomain org.gnome.Terminal.desktop[4903]: # watch>
```

journctl --since "2020-01-01" --until "yesterday"

```
[root@localhost ~]# journalctl --since "2020-01-01" --until "yesterday"
-- Logs begin at Mon 2022-08-29 01:09:06 AEST, end at Mon 2022-08-29 01:47:07 A>
```

journctl -u multi-user.target

```
[root@localhost ~]# journalctl -u multi-user.target
-- Logs begin at Mon 2022-08-29 01:09:06 AEST, end at Mon 2022-08-29 01:47:08 A>
Aug 29 01:09:22 localhost.localdomain systemd[1]: Reached target Multi-User >
Aug 29 01:16:34 localhost.localdomain systemd[1]: Stopped target Multi-User >
Aug 29 01:21:31 localhost.localdomain systemd[1]: Reached target Multi-User >
lines 1-4/4 (END)
```

journctl -u sshd.service

```
[root@localhost ~]# journalctl -u sshd.service
-- Logs begin at Mon 2022-08-29 01:09:06 AEST, end at Mon 2022-08-29 01:49:18 A>
Aug 29 01:09:14 localhost.localdomain sshd[1317]: Starting OpenSSH server dae>
Aug 29 01:09:14 localhost.localdomain sshd[1317]: Server listening on 0.0.0.>
Aug 29 01:09:14 localhost.localdomain sshd[1317]: Server listening on :: por>
Aug 29 01:09:14 localhost.localdomain systemd[1]: Started OpenSSH server dae>
Aug 29 01:16:34 localhost.localdomain sshd[1]: Stopping OpenSSH server dae>
```

journctl -u sshd.service --since "10 minutes ago"

```
[root@localhost ~]# journalctl -u sshd.service --since "10 minutes ago"
-- Logs begin at Mon 2022-08-29 01:09:06 AEST, end at Mon 2022-08-29 01:49:18 A>
-- No entries --
lines 1-2/2 (END)
```

journctl -p err

```
[root@localhost ~]# journalctl -p err
-- Logs begin at Mon 2022-08-29 01:09:06 AEST, end at Mon 2022-08-29 01:49:15>
Aug 29 01:09:06 localhost.localdomain kernel: Detected CPU family 17h model >
Aug 29 01:09:06 localhost.localdomain kernel: Warning: AMD Processor - this >
Aug 29 01:09:06 localhost.localdomain kernel: [Firmware Bug]: cpu 0, try to >
Aug 29 01:09:06 localhost.localdomain kernel: [Firmware Bug]: cpu 1, try to >
```

journctl -p warning

```
[root@localhost ~]# journalctl -p warning
-- Logs begin at Mon 2022-08-29 01:09:06 AEST, end at Mon 2022-08-29 01:51:30>
Aug 29 01:09:06 localhost.localdomain kernel: [Firmware Bug]: TSC doesn't co>
Aug 29 01:09:06 localhost.localdomain kernel: Detected CPU family 17h model >
Aug 29 01:09:06 localhost.localdomain kernel: Warning: AMD Processor - this >
Aug 29 01:09:06 localhost.localdomain kernel: unchecked MSR access error: RD>
```

journctl -p err --since "1 hour ago"

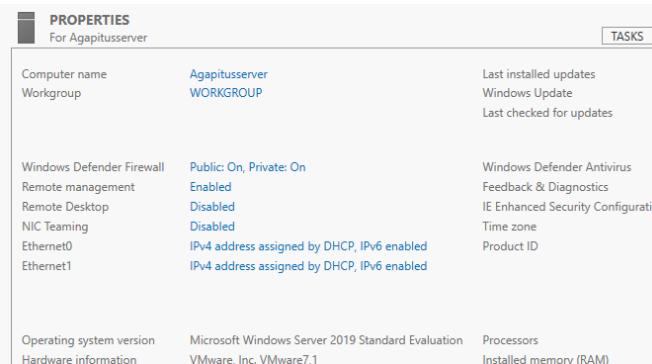
```
[root@localhost ~]# journalctl -p err --since "1 hour ago"
-- Logs begin at Mon 2022-08-29 01:09:06 AEST, end at Mon 2022-08-29 01:51:30>
Aug 29 01:09:06 localhost.localdomain kernel: Detected CPU family 17h model >
Aug 29 01:09:06 localhost.localdomain kernel: Warning: AMD Processor - this >
Aug 29 01:09:06 localhost.localdomain kernel: [Firmware Bug]: cpu 0, try to >
```

Lab 1c Windows Server 2019

Task 1: Startup

Check:

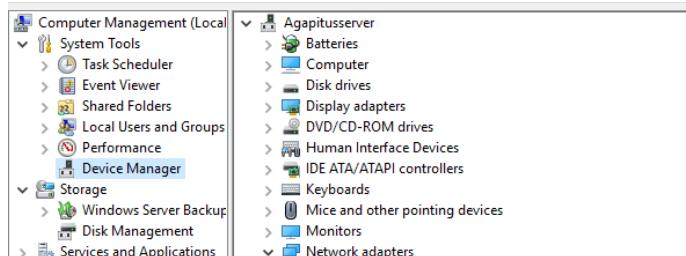
1. Time zone is set to (UTC+10:00) Canberra, Melbourne
2. Ethernet0 and Ethernet1 should be IPv4 address assigned by DHCP, IPv6 enabled.
3. Change the computer name to something more reasonable
4. Note the section on updates to see when updates were last installed/checked.



Last installed updates	22/07/2020 10:19 AM
Windows Update	Download updates only, using Windows Update
Last checked for updates	4/09/2022 1:28 AM
Windows Defender Antivirus	Real-Time Protection: On
Feedback & Diagnostics	Settings
IE Enhanced Security Configuration	On
Time zone	(UTC+10:00) Canberra, Melbourne, Sydney
Product ID	Not activated

Task 2: Server Management

Explore the device manager. **What hardware is installed on this machine?**



The installed hardware is various from computer, disk drives, display adapters, etc.

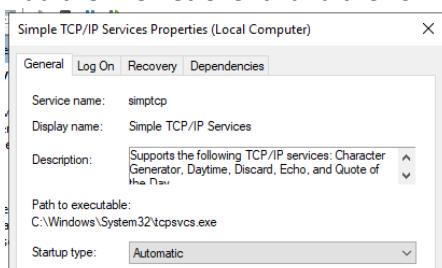
What is the network address (IP address) that connects to the internet?

```
Ethernet adapter Ethernet0:  
  Connection-specific DNS Suffix . : localdomain  
  Link-local IPv6 Address . . . . . : fe80::35f3:64ed:c2eb:fb89%7  
  IPv4 Address . . . . . : 192.168.209.133  
  Subnet Mask . . . . . : 255.255.255.0  
  Default Gateway . . . . . : 192.168.209.2  
  
Ethernet adapter Ethernet1:  
  Connection-specific DNS Suffix . : online.uts.edu.au  
  Link-local IPv6 Address . . . . . : fe80::95cd:dd8:942:5387%16  
  Autoconfiguration IPv4 Address . . : 169.254.83.135  
  Subnet Mask . . . . . : 255.255.0.0  
  Default Gateway . . . . . :
```

The ip address for eth0 is 192.168.209.133 with subnet 255.255.255.0 and gateway 192.168.209.2.

While for eth1 is 169.254.83.135 with subnet 255.255.0.0.

Add the “Telnet Client” and the “Simple TCP/IP services” features.



Test by starting a Command Prompt and typing: **telnet localhost 13; telnet localhost 17**

```
"We want a few mad people now. See where the sane ones have landed us!"  
George Bernard Shaw (1856-1950)
```

```
Connection to host lost.
```

Try using the above commands to your server's IP address. **Can you do this? Why do think this is so?**
No, cause the firewall block the connection. It needs to be turned off in order to connect the telnet.

Task 3: Command line

View the services: **net start**

```
C:\Users\Administrator>net start  
These Windows services are started:  
  
AppX Deployment Service (AppXSVC)  
Background Intelligent Transfer Service  
Background Tasks Infrastructure Service  
Base Filtering Engine
```

Is Simple TCP/IP Services running? Stop them using the command: **net stop "Simple TCP/IP Service"**

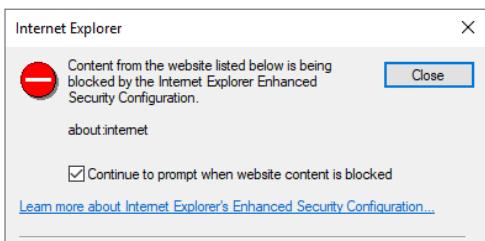
```
C:\Users\Administrator>  
C:\Users\Administrator>net stop "Simple TCP/IP Services"  
The Simple TCP/IP Services service is stopping.  
The Simple TCP/IP Services service was stopped successfully.
```

Allow the “quote of the day” though the firewall: **netsh advfirewall firewall add rule name="TCP Port 17" dir=in action=allow protocol=TCP localport=17**

```
C:\Users\Administrator>  
C:\Users\Administrator>netsh advfirewall firewall add rule name="TCP Port 17" dir=in action=allow protocol=TCP localport=17  
Ok.
```

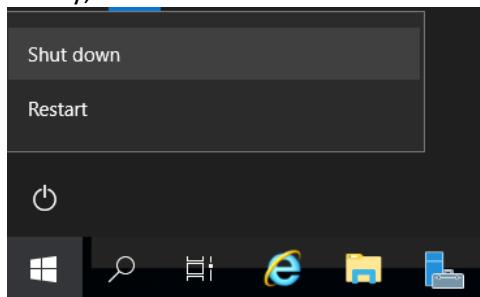
Task 4: Documentation

The Server Manager provides links to Microsoft documentation



Task 5: Shutdown

Finally, shutdown VM via the Start Menu.



Week 2

Lecture 2 Installation & Configuration

Documentation

- **man command**
man passwd
man 5 passwd

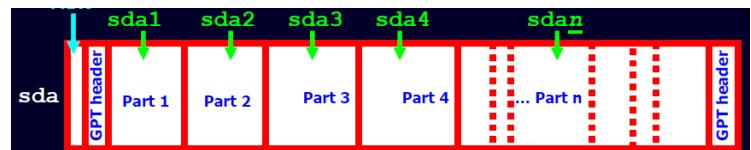
Configuration

- Linux manual
Config file **/etc/man_db.conf**
Location **MANDATORY_MANPATH/usr/local/share/man**
Package documentation **/usr/share/doc**
- Kernel & modules
Older kernels were monolithic while current still monolithic but support loadable modules
uname -a (print system information)
lsmod (print contents of **/proc/modules**)
modinfo <modulename> (shows specific module)
modprobe fat (add and remove module from Linux Kernel)
rmmmod fat (remove modules)

Disk Partitions

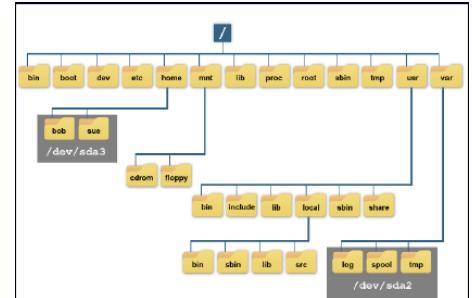
- Auto partition
3 partitions - /boot, /, swap
Root Partition managed by **LVM** (Logical Volume Manager)

- Guide Partition Table
Modern Config: UEFI based machines



Filesystems

- A filesystem is a data structure used for storing files on a disk partition
 - Directory structure indexing performance and file volume
 - btrfs: newer, high-performance, up to 16 EiB, support RAID
 - alternatives = ReiserFS, JFS, XFS
- Filesystems Hierarchy Standard (**FHS**)
The process of placing a filesystem under a mount point is called mounting.
- Creating partitions and filesystems
Create/edit partition table **COMMAND**
 - fdisk (MBR only)
 - gdisk (support GPT index method)
 - parted (GNU program)
- Maintaining filesystem
 - df (display disk usage by partition)
 - du (display disk usage by directory)
- Mounting filesystems filesystem
Mounting: **mount /media/cdrom**
Unmounting: **umount /media/cdrom**



Yum – Red Hat

- **Yum** = Yellowdog Updater
Tool for keeping system's RPM packages up to date
e.g., **yum install mysql; yum update; yum upgrade**

apt-get - Ubuntu

- Apt-get = Advanced Packaging Tool "get" utility
Tool for keeping system's Debian packages up to date
e.g., **apt-get install mysql; apt-get update; apt-get upgrade**

Viewing Processes & Job Control

- View running processes:
ps: Support Unix-style option e.g., **ps; ps -ef; top**
- Job Control **COMMAND**:
Jobs (show jobs)
Ctrl-C (permanently kill current job)
Ctrl-X (permanently stop current job)

Process Priorities

- **Priorities**
High priority processes likely to get more CPU time
Different Unixes differ in numeric ranges and algorithms used to implement priorities

- **Nice Value**
High nice value means low priority (lets others go first)
Nice Value range from -20 to +19
Negative nice values have highest priority
Only root can increase a process priority (decrease niceness)
- **Creating new processes**
Nice myprog (no nice value)
Nice -12 myprog (runs with nice +12)
Nice -n -12 myprog (runs with nice -12)
- **Change priority of running process**
Renice -20 12345 (set PID 12345 to nice -20)
Renice 19 -u chen anup (nice 19 for chen & anup's processes)

Lab 2a System Documentation

Task 1: Using and configuring man pages

Start terminal, run the command '**mandb -cqs &**' in the background.

```
[root@localhost ~]# mandb -cqs &
[1] 4744
```

From another terminal, run the following 2 commands: **man 1 passwd**; **man 5 passwd**

What is the difference between the two?

```
PASSWD(1)                               User utilities
NAME
passwd - update user's authentication tokens
```

man 1 passwd command shows the User Utilities, which used to update user's authentication tokens

```
PASSWD(5)                               Linux Programmer's Manual
NAME
passwd - password file
```

man 5 passwd shows the linux programmer's manual, which is a command for a password file.

Now run the command:

man passwd

Which one is shown (section 1 or section 5)?

Section 1. It is because the command shows the password according to the index number.

Look in the file **/etc/man_db.conf**. See the SECTION entry then see at the MANDATORY_MANPATH.

```
# particular order. Sections with extensions should usually be adjacent to
# their main section (e.g. "1 lmh 8 ...").
#
SECTION      1 1p 8 2 3 3p 4 5 6 7 9 0p n l p o 1x 2x 3x 4x 5x 6x 7x 8x
#
#-----
# Range of terminal widths permitted when displaying cat pages. If the
# terminal falls outside this range, cat pages will not be created (if
# missing) or displayed.
#
```

The section **verifies** the behavior of the 'man passwd' command.

```
[root@localhost ~]# cat /etc/man_db.conf | grep MANDATORY_MANPATH
# MANDATORY_MANPATH                         manpath element
#MANDATORY_MANPATH                          /usr/src/pvm3/man
MANDATORY_MANPATH                          /usr/man
MANDATORY_MANPATH                          /usr/share/man
MANDATORY_MANPATH                          /usr/local/share/man
[root@localhost ~]#
```

The **MANDATORY_MANPATH** command shows which directories the man command will look for man pages.

Now run the following two commands and compare them: `man grep`; `info grep`

```
GREP(1)          General Commands Manual          GREP(1)

NAME
    grep, egrep, fgrep - print lines matching a pattern

SYNOPSIS
    grep [OPTIONS] PATTERN [FILE...]
    grep [OPTIONS] -e PATTERN ... [FILE...]
    grep [OPTIONS] -f FILE ... [FILE...]
```

`man grep` command used to show the **General Command Manual** which has the synopsis, description, option, and many more.

```
Next: Introduction, Up: (dir)
grep
*****
'grep' prints lines that contain a match for a pattern.

This manual is for version 3.1 of GNU Grep.
```

`info grep` command used to show the **introduction** which also contain the hierarchy, hyperlink, command-line option, and many more.

Are they basically the same, or are they quite different?

Both '`man grep`' and '`info grep`' command is actually quite similar. However, sometimes some commands do not have the manual page, so we could use the info command.

Checking the mandb command whether it is finished by running "`jobs`".

```
[root@localhost ~]# jobs
[1]-  Done                  mandb -cqs
[2]+  Done                  mandb -cqs
[root@localhost ~]#
```

Mandb command has done running.

Then, try after the above process finished running, continue by running: `whatis passwd`; `apropos passwd`

```
[root@localhost ~]# whatis passwd
openssl-passwd (lssl) - compute password hashes
passwd (1)           - update user's authentication tokens
passwd (5)           - password file
```

The `whatis` command display only one-line manual page descriptions of any name matched.

```
[root@localhost ~]# apropos passwd
chpasswd (8)        - update group passwords in batch mode
chpasswd (8)        - update passwords in batch mode
fgetpwent_r (3)    - get passwd file entry reentrantly
getpwent_r (3)    - get passwd file entry reentrantly
```

The `apropos` command will search for the manual page names and description for instance of keyword.

Which one returns more results? Why?

The `apropos` command results more complete than the `whatis` command since it searches for the manual page names and description for instance of keyword.

Task 2: Finding installed package documentation

Installed package documentation is in the directory `/usr/share/doc`.

This documentation only for the sed editor. View the contents of the FAQ file with: `zless sedfaq.txt.gz`

The `zless` command will uncompress the `.gz` file and pipe it to the `less` pager.

```
Archive-Name: editor-faq/sed
Posting-Frequency: irregular
Last-modified: 10 March 2003
Version: 015
URL: http://sed.sourceforge.net/sedfaq.html
Maintainer: Eric Pement (pemente@northpark.edu)

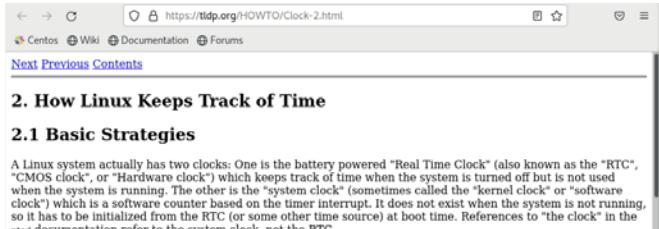
THE SED FAQ
Frequently Asked Questions about
sed, the stream editor
```

The content of the FAQ files viewed using `zless` command.

Task 3: Finding documentation on the Internet

[www.tldp.org](https://tldp.org) (The Linux Documentation Project) is a platform for up-to-date Linux documentation.

Find a HOWTO document about the Linux Clock



A screenshot of a web browser window displaying the URL <https://tldp.org/HOWTO/Clock-2.html>. The page content is titled "2. How Linux Keeps Track of Time" and "2.1 Basic Strategies". It contains text explaining that a Linux system has two clocks: the Real Time Clock (RTC) and the system clock. A note at the bottom states: "and documentation refer to the system clock, not the RTC".

<https://tldp.org/HOWTO/Clock-2.html>

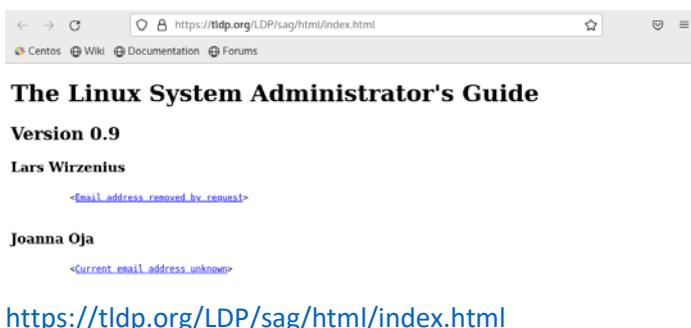
Find the main Linux FAQ



A screenshot of a web browser window displaying the URL <https://tldp.org/docs.html#faq>. The page features the "The Linux Documentation Project" logo with a penguin icon. The navigation menu includes links for home, HOWTOs, Guides, FAQs, man pages, Linux Gazette, and LinuxFocus. A blue header bar says "Linux Documentation Project Works". Below the header, a note states: "The Linux Documentation Project (LDP) is working on developing good, reliable documentation for the Linux operating system."

<https://tldp.org/docs.html#faq>

Find the Linux System Administrator's Guide (or SAG for short)



A screenshot of a web browser window displaying the URL <https://tldp.org/LDP/sag/html/index.html>. The page title is "The Linux System Administrator's Guide Version 0.9". It lists the authors: Lars Wirzenius (<email address removed by request>) and Joanna Oja (<Current email address unknown>).

<https://tldp.org/LDP/sag/html/index.html>

Lab 2b System Updates

Task 1: Enable Linux Networking

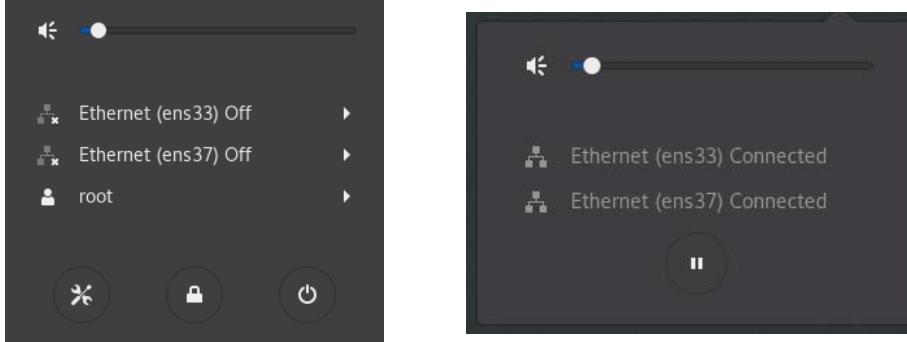
Start the *NetworkManager* service and enable it using command: `systemctl start NetworkManager.service`.

Afterwards, run `systemctl enable NetworkManager.service` to enable the *NetworkManager*.service

```
[root@localhost ~]# systemctl start NetworkManager
[root@localhost ~]# systemctl is-active NetworkManager
active
[root@localhost ~]# systemctl enable NetworkManager
```

Setting up the Network Manager and check by using `systemctl is-active NetworkManager.service`.

Once done, click on the top right of the screen to turn on Ethernet “ens33”.



The ethernet is already connected to the network.

Verify that your network is working by pinging a known address.

```
[root@localhost ~]# ping www.uts.edu.au -c 3
PING uts.edu.au (54.79.20.73) 56(84) bytes of data.
64 bytes from ec2-54-79-20-73.ap-southeast-2.compute.amazonaws.com (54.79.20.73)
: icmp_seq=1 ttl=128 time=5.81 ms
64 bytes from ec2-54-79-20-73.ap-southeast-2.compute.amazonaws.com (54.79.20.73)
: icmp_seq=2 ttl=128 time=4.55 ms
64 bytes from ec2-54-79-20-73.ap-southeast-2.compute.amazonaws.com (54.79.20.73)
: icmp_seq=3 ttl=128 time=3.96 ms
```

Successfully ping www.uts.edu.au

Task 2: Updates for Linux

By default, Redhat derive Linux with the **Gnome Software Manager**. This is in a package called ‘**PackageKit**’.

To start the **Gnome Software Manager**, run the command: **gnome-software**

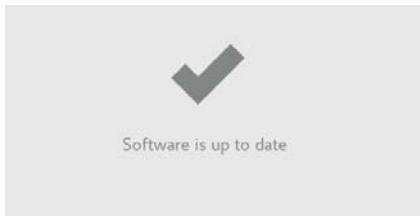
The other ways to start it is by using the icon that looks like a shopping bag:



The interface of the gnome software manager.

Find the tab called **Updates** and use the **Refresh icon** to check for new updates.

Be **EXTREMELY** careful of any changes since it could lead to operational issues.



The software is up to date.

Alternative installation method

Use the **yum** command to update packages. This is often a safer way of doing updates. Yum **COMMANDS** include:

yum check-update
yum search XXXXX
yum install XXXXX

yum update XXXXX
yum remove XXXXX

Yum automatically determines the dependencies and can warn you.

```
[root@localhost ~]# yum check-update
Last metadata expiration check: 1:24:13 ago on Mon 29 Aug 2022 23:25:37 AEST.
[root@localhost ~]# yum check-update
Last metadata expiration check: 1:24:49 ago on Mon 29 Aug 2022 23:25:37 AEST.
[root@localhost ~]#
```

Yum is up to date.

Task 3: Updates for Windows

Windows manages updates within Server Manager on **Local Server**, there are information about the installed updates.

PROPERTIES	Last installed updates	22/07/2020 10:19 AM
	Windows Update	Download updates only, using Windows Update
	Last checked for updates	22/07/2020 11:05 AM
For Agapitusserver	Last installed updates Windows Update Last checked for updates	22/07/2020 10:19 AM Download updates only, using Windows Update Today at 2:19 AM

Shown in the Server Manager that the Windows is already up-to-date.

To run updates, there are **two options** could be used, which are by using the Server Manager, or through the settings area and search for **Windows Update** in the Update & Security section.

Other way to check for and install updates in Windows is from the **command line**. It could be done both from the **Command Prompt** and **Windows PowerShell** which provides a module for working with Windows update from the command line.

Open Windows PowerShell, install the module by running command: *Install-Module PSWindowsUpdate*

It will probably warn you about installing other dependent software – choose ‘Yes’ if asked.

```
PS C:\Users\Administrator> Install-Module PSWindowsUpdate
NuGet provider is required to continue
PowerShellGet requires NuGet provider version '2.8.5.201' or newer to interact with NuGet-based repositories. The NuGet provider must be available in 'C:\Program Files\PackageManagement\ProviderAssemblies' or 'C:\Users\Administrator\AppData\Local\PackageManagement\ProviderAssemblies'. You can also install the NuGet provider by running 'Install-PackageProvider -Name NuGet -MinimumVersion 2.8.5.201 -Force'. Do you want PowerShellGet to install and import the NuGet provider now?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): Y
```

The module is then already installed using Windows PowerShell.

Once installed, run the PowerShell commands: *Get-WindowsUpdate* to Check for updates.

```
PS C:\Users\Administrator> Get-WindowsUpdate
ComputerName Status KB Size Title
----- -- -- --
AGAPITUSS... -D---- KB4535680 47KB Security Update for Windows Server 2019 for x64-based Systems (KB4535680)
AGAPITUSS... -D---- KB4577586 21KB Update for Removal of Adobe Flash Player for Windows Server 2019 for x64-ba...
AGAPITUSS... ----- KB5011267 75MB 2022-02 Cumulative Update Preview for .NET Framework 3.5, 4.7.2 and 4.8 for...
```

Checking Windows Update on PowerShell.

Then use *Get-WULastResults* to show results of the last time updates were checked.

```
PS C:\Users\Administrator> Get-WULastResults
ComputerName LastSearchSuccessDate LastInstallationSuccessDate
----- -- -- --
AGAPITUSERVER 29/08/2022 2:53:38 PM 22/07/2020 12:19:04 AM
```

Shown the result of last time updates.

Use ***Get-WUHistory*** to show history of updates installed.

ComputerName	Operationname	Result	Date	Title
AGAPITUSS...	Installation	Succeeded	29/08/2022 2:29:...	Update for Microsoft Defender Antivirus antimalware plat...
AGAPITUSS...	Installation	Succeeded	29/08/2022 2:19:...	Security Intelligence Update for Microsoft Defender Antiv...
AGAPITUSS...	Installation	Aborted	18/08/2022 12:04...	Security Intelligence Update for Microsoft Defender Antiv...
AGAPITUSS...	Installation	Succeeded	18/08/2022 12:04...	Update for Microsoft Defender Antivirus antimalware plat...
AGAPITUSS...	Installation	Succeeded	13/08/2022 12:11...	Security Intelligence Update for Microsoft Defender Antiv...

Show the history of updates installed.

Check the Status by: ***Get-WURebootStatus***

ComputerName	RebootRequired	RebootScheduled
AGAPITUSSERVER		False

Checking whether the system need reboot.

Does the system need rebooting to install most recent updates?

The system for the above updates doesn't need any rebooting system.

Lab 2c Manage Processes

Task 1: Viewing Process Information in Linux

Run the command in Linux: ***ps***

Which processes does it show by default? Now re-run the command with the -ef options. What does it show?

	PID	TTY	TIME	CMD
[root@localhost ~]#	2776	pts/0	00:00:00	bash
	2810	pts/0	00:00:00	ps

The ***ps*** command show the current processes by shell. By default, the processes are ***bash*** and ***ps***.

	UID	PID	PPID	C	STIME	TTY	TIME	CMD
[root@localhost ~]#	root	1	0	1	02:36 ?		00:00:02	/usr/lib/systemd/systemd --s
	root	2	0	0	02:36 ?		00:00:00	[kthreadd]
	root	3	2	0	02:36 ?		00:00:00	[rcu_gp]
	root	4	2	0	02:36 ?		00:00:00	[rcu_par_gp]
	root	5	2	0	02:36 ?		00:00:00	[kworker/0:0-cgroup_pidlist_

The ***ps -ef*** command print all the ***running processes*** on the ***terminal*** along with some other information. It includes the UID, PID, PPID, C, STIME, TTY, TIME, and CMD.

Then, do it with the ***-ef*** options and also ***--forest***. ***What does it show?***

	UID	PID	PPID	C	STIME	TTY	TIME	CMD
[root@localhost ~]#	root	2	0	0	02:36 ?		00:00:00	[kthreadd]
	root	3	2	0	02:36 ?		00:00:00	_ [rcu_gp]
	root	4	2	0	02:36 ?		00:00:00	_ [rcu_par_gp]
	root	6	2	0	02:36 ?		00:00:00	_ [kworker/0:0H-events_hig
	root	8	2	0	02:36 ?		00:00:00	_ [kworker/u256:0-writebac

The ***-- forest*** command used to list processes in a ***tree format*** (the default is by the PID order). The content is still the same, just different format of the display.

Now use the ***top*** command to see a ***continually updated list*** of running processes.

Which process(es) are often near the top of the list? How much physical memory is installed in your VM? How much of the physical memory is currently being used? How much swap space is there, and how much are being used?

```

top - 02:51:49 up 15 min, 1 user, load average: 1.58, 2.76, 1.78
Tasks: 253 total, 2 running, 250 sleeping, 0 stopped, 1 zombie
%Cpu(s): 47.8 us, 5.4 sy, 0.0 ni, 39.6 id, 0.4 wa, 6.3 hi, 0.5 si, 0.0 st
MiB Mem : 1945.3 total, 393.5 free, 1148.5 used, 403.3 buff/cache
MiB Swap: 2048.0 total, 1508.0 free, 540.0 used. 591.8 avail Mem

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
3134 root 20 0 3587292 385180 109044 S 53.8 19.3 4:36.02 Web Con+
2864 root 20 0 3468308 293428 95368 S 30.6 14.7 3:01.30 firefox
2291 root 20 0 3471364 124268 32604 S 17.3 6.2 2:37.16 gnome-s+
2334 root 20 0 282408 20504 16576 S 1.7 1.0 0:14.27 Xwayland

```

The **top two processes** that often on the top of the list are **Web Content** and **Firefox**.

The total physical memory installed in virtual machine is 1945.3 mb, while currently being used for 1148.5 mb. The total swap space in VM is 2048.0 mb, while currently being used for 540.0 mb.

Task 2: Viewing and Changing Process Priorities in Linux

Run the command: `dd if=/dev/zero; of=/dev/null`

This will constantly read input from **zero** device and writes it to **null** device (bit bucket).

```
[root@localhost ~]# dd if=/dev/zero of=/dev/null
```

Running the **dd** command.

Open another terminal and run the **top** command. See the **dd** process at the top list, and “NI” value label.
What is the default nice value of the dd process?

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
3673	root	20	0	7360	1016	948	R	57.1	0.1	1:12.50	dd
2864	root	20	0	3466904	299356	92124	S	56.1	15.0	6:01.12	firefox
3134	root	20	0	3579888	397156	102668	R	30.6	19.9	8:24.51	Web Con+

The **dd** process could be seen at the top of the list. It has the **default NI value of 0**.

To exit from top, press “q” and then use the **kill command** to kill the dd process by specifying its process ID.
What appears on the screen?

```
[root@localhost ~]# dd if=/dev/zero of=/dev/null
Terminated
[root@localhost ~]#
```

It shows that the **dd** process is **TERMINATED**.

Now re-run the **dd** command, but with a nice value: `nice -n 15 dd if=/dev/zero of=/dev/null`

Run top in the other console and check the nice value

```
[root@localhost ~]# nice -n 15 dd if=/dev/zero of=/dev/null
```

Running the **dd** command with **nice value of 15**.

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
3977	root	35	15	7360	944	876	R	80.1	0.0	0:19.93	dd
3134	root	20	0	3575964	411100	103512	R	38.2	20.6	12:05.47	Web Content
2864	root	20	0	3466120	299976	91852	S	18.9	15.1	11:05.02	firefox

The **dd** process has **15 NI value**.

Start another terminal and run the command: `renice -20 XXXX`

Replace the XXXXX with the PID number of the **dd** process. **What does it show? Does this mean the process has high priority or low priority? Try running some other commands like ls or rpm -qa. What do you notice?**

```
[root@localhost ~]# renice -20 3977
3977 (process ID) old priority 15, new priority -20
[root@localhost ~]#
```

Changing the **dd** value of NI to **-20**.

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
3977	root	0	-20	7360	944	876	R	96.0	0.0	1:44.75	dd
3134	root	20	0	3574116	424096	99600	S	32.7	21.3	12:47.60	Web Content
2291	root	20	0	3494988	193380	66992	S	0.7	9.7	6:29.81	gnome-shell

The NI value **changed to -20**.

This mean that the **process has high priority**. It could be seen by the **CPU usage** that is **above 90%** and the **speed** when running other command.

Now change the process' priority again with: **renice 19 XXXXX**

What does this do to the process, and to your system overall when you run other commands in third console window?

```
3977 (process ID) old priority 19, new priority 20
[root@localhost ~]# renice 19 3977
3977 (process ID) old priority -20, new priority 19
[root@localhost ~]#
```

Changing the **dd** value of NI to 19.

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2864	root	20	0	3531204	302580	89952	S	77.1	15.2	13:30.61	firefox
3134	root	20	0	3578076	382212	99988	S	44.9	19.2	15:18.74	Web Content
3977	root	39	19	7360	944	876	R	29.2	0.0	8:03.15	dd

The NI value **changed to 19**.

This effect to the **speed** when running other command which is **faster than the value of -20**.

Task 3: Job Control in Linux

Suspend the command by pressing Ctrl-Z. **What message is shown when you press Ctrl-Z?** Try the command: **jobs**

```
^Z
[1]+  Stopped                  nice -n 15 dd if=/dev/zero of=/dev/null
[root@localhost ~]# jobs
[1]+  Stopped                  nice -n 15 dd if=/dev/zero of=/dev/null
```

The **Ctrl-Z** command **stopped** the process as same as the jobs results since it stopped.

Make the **dd** process continue running in the background: **bg 1**

Then run the **jobs** command again – **what does it say the status of the dd process is this time?**

```
[root@localhost ~]# bg 1
[1]+ nice -n 15 dd if=/dev/zero of=/dev/null &
[root@localhost ~]# jobs
[1]+ Running                   nice -n 15 dd if=/dev/zero of=/dev/null &
```

After running the **bg 1** command, the jobs now able to running again.

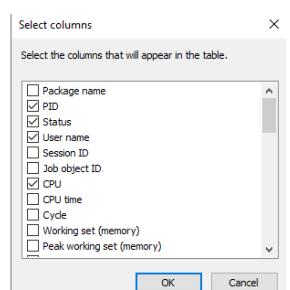
Bring the **dd** command back to the foreground by typing: **fg 1**

Anything typed is just echoed to the screen, and there is no response from the shell.

Finally, end the **dd** process by pressing **Ctrl-C**. This should **terminate** the process.

```
[root@localhost ~]# fg 1
nice -n 15 dd if=/dev/zero of=/dev/null
^C192504805+0 records in
192504804+0 records out
98562459648 bytes (99 GB, 92 GiB) copied, 245.761 s, 401 MB/s
```

Bring back the **dd** command, then **end** the process by pressing **Ctrl-C**.



Task 4: Viewing Processes in Windows

There are **two ways** to open task manager which by right-click in the Windows Taskbar and choose **Task Manager** or by using the key-combination **Ctrl-Shift-Esc**. Click “Show More” -> “More Details” -> “Details” -> “Select Columns”.

Other way to process information is from the **command-line** in Windows, using **PowerShell**. Open a PowerShell window and type: ***Get-Process***

PS C:\Users\Administrator> Get-Process						
V	Handles	NPM(K)	PM(K)	WS(K)	CPU(s)	Id SI ProcessName
C	243	14	4096	17192	0.05	3052 1 backgroundTaskHost
C	255	14	4240	17624	0.08	5484 1 conhost
C	426	18	2236	5376	0.20	400 0 csrss
C	301	18	2188	5320	0.33	516 1 csrss

Get-Process command to get the **process information**.

There is another command which works in both PowerShell and a simple command prompt: ***Tasklist***

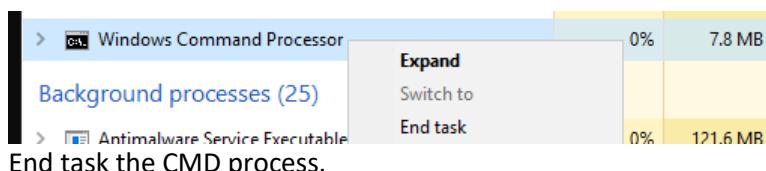
PS C:\Users\Administrator> tasklist				
Image Name	PID	Session Name	Session#	Mem Usage
System Idle Process	0	Services	0	8 K
System	4	Services	0	160 K
Registry	88	Services	0	71,536 K

Tasklist command to see the **list of tasks**.

Task 5: Killing a Process in Windows

Open a Command Prompt to try killing this process several ways.

Open the Task Manager, search for the command prompt process (**Windows Command Processor**). Right-click on the process in the list and choose “**End task**”.



End task the CMD process.

Now open another Command Prompt window. It will be a different process with a different process ID (PID).

Using PowerShell, find the process and kill the process using it's ID in PowerShell, type: ***Stop-Process -ID XXXX*** (XXXX is the PID number).

PS C:\Users\Administrator> ***Stop-Process -ID 4204***

Stopping the CMD process by using **PowerShell**.

Use ***taskkill /PID XXXX*** where XXXX is the PID number.

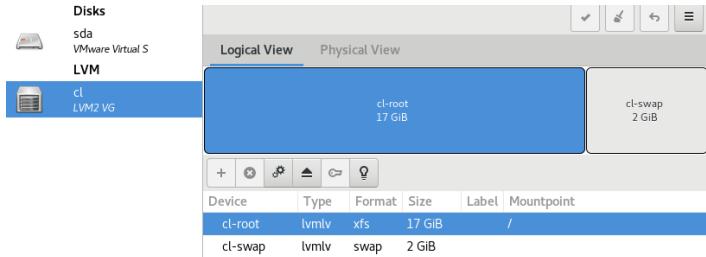
```
C:\Users\Administrator>taskkill /PID 1060
SUCCESS: Sent termination signal to the process with PID 1060.
```

End process the application using **CMD**.

Lab 2d Disk Partitioning

Task 1: Linux Disk Partitioning

By default, Redhat and Centos derivative Linux operating systems first create a disk partition called **/boot**, just for the **bootloader files**. Then for the remaining disk space, they use the **Logical Volume Manager** to create 2 logical volume groups called **cl-root** and **cl-swap**.



This allows the system to expand the size of the root drive as needed.

Check the current layout by: **mount**

Filter it using **grep** to only show the lines that start with **/dev**: **mount | grep '^/dev/'**

```
[root@localhost ~]# mount | grep '^/dev/'
/dev/mapper/cl-root on / type xfs (rw,relatime,attr2,inode64,logbufs=8,logbsize=32k,noquota)
/dev/sda1 on /boot type ext4 (rw,relatime)
[root@localhost ~]#
```

Display the current layout and filter it using **grep** command.

This display the device and where it is mounted on the file system hierarchy. Run the command: **df**

```
[root@localhost ~]# df
Filesystem      1K-blocks    Used Available Use% Mounted on
devtmpfs          966272      0   966272   0% /dev
tmpfs            996012      0   996012   0% /dev/shm
tmpfs            996012   9780   986232   1% /run
```

df command to show the **device** and **where it is mounted**.

This shows the amount of free space in the mounted file system. You can use: **parted /dev/sda print**

```
[root@localhost ~]# parted /dev/sda print
Model: VMware, VMware Virtual S (scsi)
Disk /dev/sda: 21.5GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start   End     Size    Type      File system  Flags
 1      1049kB  1075MB  1074MB  primary   ext4        boot
 2      1075MB  21.5GB   20.4GB  primary           lvm

[root@localhost ~]#
```

Displaying **current partitions**.

This should display your current partitions. To see what's inside the logical volume, use: **lvs**.

```
[root@localhost ~]# lvs
  LV   VG Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  root cl -wi-ao---- <17.00g
  swap cl -wi-ao----  2.00g
[root@localhost ~]#
```

lvs command to see **inside the logical volume**.

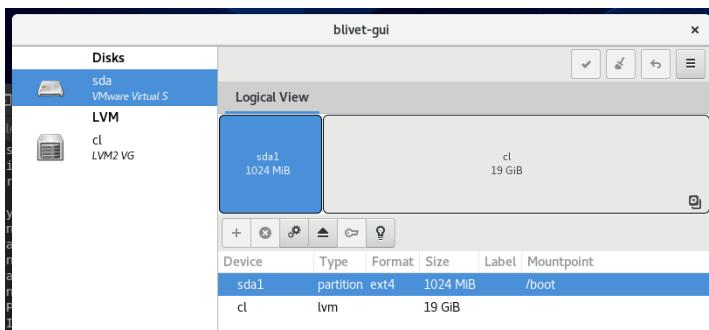
To view more detail of the **swap partition** use command: **Swapon**

```
[root@localhost ~]# swapon
NAME      TYPE      SIZE    USED  PRI0
/dev/dm-1 partition  2G  238.3M   -2
[root@localhost ~]#
```

Swapon command to see the **details of the swap partition**.

GUI interface

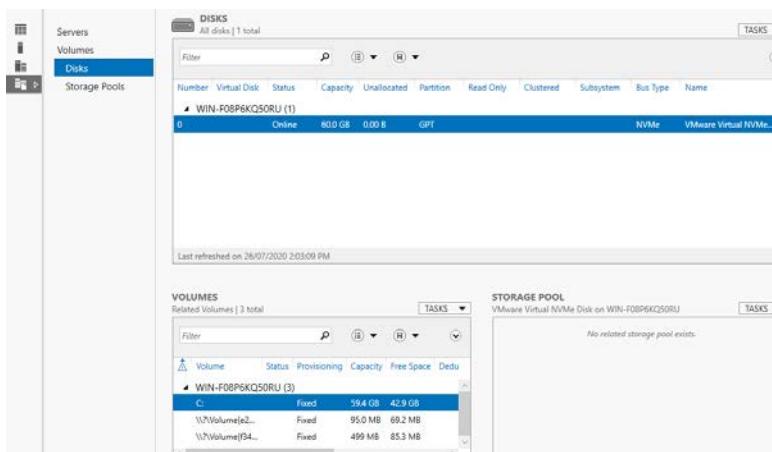
Run ***blivet-gui*** and see a graphical view of the disk layout.



Blivet-gui graphical view.

Task 2: Windows Server Disk Management

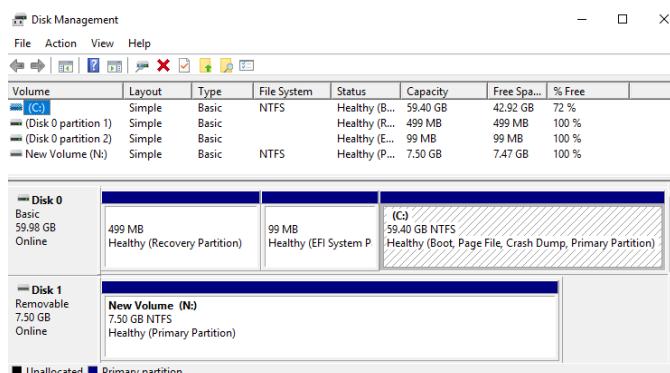
Server Manager manage ‘File and Storage Services’ on the left-hand menu “Disks”.



It shows that our system has a 60GB disk, with three partitions/volumes. The main volume is the one labelled “C:” which is the main OS hard drive.

Managing disks on non-server Windows systems

To manage disks and volumes on a Windows desktop machine, go to **Settings**, go for “disk”, find the **Windows disk management tool**.



Week 3

Lecture 3 Networking and Services

Network

- Layers
 - Physical Layer (Hardware)
 - Link Layer (Hardware)
 - Network Layer (IP & ICMP)
- Address
 - Link Layer (MAC Addresses)
 - Network Layer (4-byte IP Addresses; 16-bit segment IP v6 Addresses)
 - Private IP addresses (freely used in private networks)
 - Public IP addresses (Authority delegated to regional centers)
 - Network Address Translation (NAT) (Packet with private IP processed by NAT)
- Configuration
 - Legacy:** ifconfig; route; hostname
 - IP Utility:** set IP address (ip addr); link down/up (ip link); route, default gateway (ip route)
 - NetworkManager (NM):** nmcli; systemctl
 - Config Files:** /etc/sysconfig/network; /etc/resolv.conf; /etc/sysconfig/network-scripts/ifcfg-xxx;
 - Tools:** nm-connection-editor; nmtui; nmcli;

Hostname

System Time

- Three classes:
 - Static:** stored in /etc/hostname
 - Transient:** dynamic by kernel; hostname hostname.example.com
 - Pretty:** free-form UTF-8
- Unix Time = GMT / UTC
 - COMMAND:** date; date -u
 - Timedatectl** tool to set time/zone/system

NTP Time

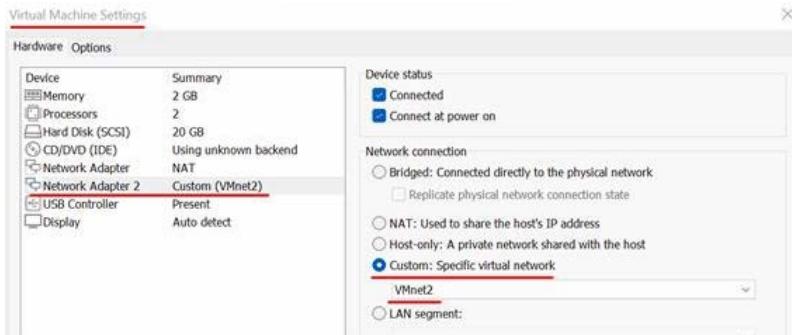
- Configure **server side**
Systemctl start ntpd
- Synchronize **client side**
Ntpdate time.uts.edu.au
- **Chrony** daemon
Faster and keep accurate time: /etc/chrony.conf

Lab 3a Static Networking

Task 1: Querying your Network Configuration

Use **vmnet2** as the new private network.

Go to “**Settings**” change Network Adapter to “**Custom**” using VMnet2.



Enter the default values here for the record:

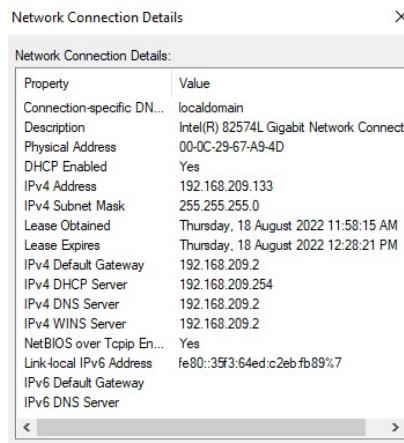
Linux: ens33		Windows: Ethernet0	
DNS suffix	localdomain	DNS suffix	localdomain
Inet address	192.168.209.128	Inet address	192.168.209.133
Subnet mask	255.255.255.0	Subnet mask	255.255.255.0
Default route	192.168.209.2	Default route	192.168.209.2

```
[root@localhost ~]# systemctl is-active NetworkManager
active
[root@localhost ~]# systemctl start NetworkManager
[root@localhost ~]# systemctl enable NetworkManager
[root@localhost ~]# ifconfig ens33
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      ether 00:0c:29:60:cf:eb txqueuelen 1000  (Ethernet)
      RX packets 0 bytes 0 (0.0 B)
      RX errors 0 dropped 0 overruns 0 frame 0
      TX packets 0 bytes 0 (0.0 B)
      TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@localhost ~]#
[root@localhost ~]# ifconfig ens33
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 192.168.209.128 netmask 255.255.255.0 broadcast 192.168.209.255
          inet6 fe80::dc4:b003:eda6:2840 prefixlen 64 scopeid 0x20<link>
            ether 00:0c:29:60:cf:eb txqueuelen 1000  (Ethernet)
            RX packets 28 bytes 8000 (7.8 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 53 bytes 6201 (6.0 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@localhost ~]#
```

```
[root@localhost ~]#
[root@localhost ~]# route
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref    Use Iface
default         gateway        0.0.0.0       UG    100    0        0 ens33
192.168.122.0  0.0.0.0       255.255.255.0 U      0    0        0 virbr0
192.168.209.0  0.0.0.0       255.255.255.0 U      100   0        0 ens33
[root@localhost ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref    Use Iface
0.0.0.0         192.168.209.2  0.0.0.0       UG    100    0        0 ens33
192.168.122.0  0.0.0.0       255.255.255.0 U      0    0        0 virbr0
192.168.209.0  0.0.0.0       255.255.255.0 U      100   0        0 ens33
[root@localhost ~]#
```



The shown values for the record.

Check the status of all network interfaces on: `ifconfig -a`.

Global network configuration parameters are in `/etc/sysconfig/network`.

Interface-specific network parameters are in `/etc/sysconfig/network-scripts/ifcfg-ens33`.

```
[root@localhost ~]# cat /etc/sysconfig/network
# Created by anaconda
[root@localhost ~]# cat /etc/sysconfig/network-scripts/ifcfg-ens33
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=dhcp
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens33
UUID=9a5b99db-9450-44c5-aece-fbfb20f28e7d
DEVICE=ens33
ONBOOT=yes
[root@localhost ~]#
```

The virtual network between VM and the host workstation is set to **192.168.3.0/24**. The IP addresses from 192.168.3.128 to 192.168.3.254 are allocated by **DHCP** to both the Ethernet 0 interfaces on the Linux and Windows VM.

Task 2: Designing the Network

Set up our own private network on the 2nd network adapter via the VMnet2 virtual switch. Try using these configurations:

1. Our network will be in the subject **10.0.2.0/24**
2. We will have to eventually cope with **200 machines**
3. Our **gateway** will be the **Linux machine**
4. Pick an appropriate IP address/netmask/gateway for the Linux and Windows servers

Calculate the appropriate subnet and host values here:

Linux: ens37		Windows: Local Area Connection 2	
Inet address	10.0.2.1	Inet address	10.0.2.2
Subnet mask	255.255.255.0	Subnet mask	255.255.255.0
Default route	10.0.2.1	Default route	10.0.2.1

Task 3: Designing the Network Configuration

3a: Linux Server

Initially, VM is configured to use **DHCP networking**. Use **ifconfig** manually configure the parameters for Ethernet 1 (ens37). E.g., **ifconfig ens37 [your-ip-addr] netmask [your-netmask]**.

```
simulated command is: ifconfig
[root@localhost ~]# ifconfig ens37 10.0.2.1 netmask 255.255.255.0
[root@localhost ~]# ifconfig ens37
ens37: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 10.0.2.1 netmask 255.255.255.0 broadcast 10.0.2.255
                ether 00:0c:29:69:cf:f5 txqueuelen 1000 (Ethernet)
                RX packets 23 bytes 7866 (7.6 KiB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 11 bytes 1095 (1.0 KiB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Configure the network parameters for **ens37**.

View the routing table using **route** then add a default route using: **route add default gw [gateway-ip-addr]**.

Test the statically configured network by: **ping [gateway-ip-addr]**.

```
[root@localhost ~]# route add default gw 10.0.2.1
[root@localhost ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref  Use Iface
0.0.0.0         10.0.2.1       0.0.0.0        UG   0      0      0 ens37
0.0.0.0         192.168.209.2  0.0.0.0        UG   100    0      0 ens33
10.0.2.0        0.0.0.0        255.255.255.0  U     0      0      0 ens37
192.168.122.0   0.0.0.0        255.255.255.0  U     0      0      0 virbr0
192.168.209.0   0.0.0.0        255.255.255.0  U     100    0      0 ens33
[root@localhost ~]# ping 10.0.2.1
PING 10.0.2.1 (10.0.2.1) 56(84) bytes of data.
64 bytes from 10.0.2.1: icmp_seq=1 ttl=64 time=0.610 ms
64 bytes from 10.0.2.1: icmp_seq=2 ttl=64 time=0.157 ms
64 bytes from 10.0.2.1: icmp_seq=3 ttl=64 time=0.180 ms
^C
--- 10.0.2.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2033ms
rtt min/avg/max/mdev = 0.157/0.315/0.610/0.209 ms
```

Add default gateway and try to ping the gateway address.

3b: Windows Server

On Windows Server, use the GUI to configure our 2nd network card:

- Server manager --- Local Server --- Ethernet1
- Right-click on the interface (Ethernet1) --- Properties
- Choose “Internet Protocol Version 4 (TCP/IPv4)” --- Properties
- Then choose the “Use the following IP Address” & enter the values you calculated.
-

Confirm that the configuration is correct?

```
Ethernet adapter Ethernet1:
Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . : fe80::95cd:dd8:942:5387%16
IPv4 Address . . . . . : 10.0.2.2
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 10.0.2.1

C:\Users\Administrator>ping 10.0.2.1

Pinging 10.0.2.1 with 32 bytes of data:
Reply from 10.0.2.1: bytes=32 time<1ms TTL=64
Reply from 10.0.2.1: bytes=32 time<1ms TTL=64
Reply from 10.0.2.1: bytes=32 time=1ms TTL=64
```

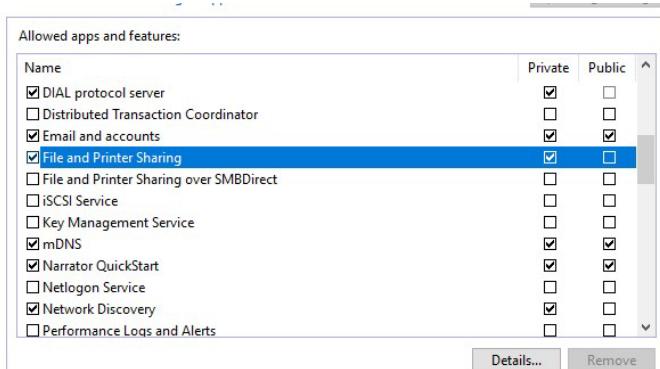
Configure the IP in windows server. The configuration is **confirmed**

Task 4: Setting up the Firewall

4a: Windows Server

There are 2 ways to modify the Windows firewall:

1. Click Start --- Control Panel --- Network and Internet --- System and Security - Windows Defender Firewall --- Allow an app through firewall --- File and Print sharing --- enable checkbox(es) (private and public)
2. Server manager --- Tools --- Windows Defender Firewall with Advanced Security --- Inbound rules--- File and Printer sharing (Echo request ICMPv4 IN, domain....) --- enable for each profile



Setting up the File and Printer Sharing.

```
[root@localhost ~]# ping 10.0.2.2
PING 10.0.2.2 (10.0.2.2) 56(84) bytes of data.
64 bytes from 10.0.2.2: icmp_seq=1 ttl=128 time=0.951 ms
64 bytes from 10.0.2.2: icmp_seq=2 ttl=128 time=0.790 ms
64 bytes from 10.0.2.2: icmp_seq=3 ttl=128 time=0.475 ms
64 bytes from 10.0.2.2: icmp_seq=4 ttl=128 time=0.571 ms
64 bytes from 10.0.2.2: icmp_seq=5 ttl=128 time=0.525 ms
^C
--- 10.0.2.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4069ms
rtt min/avg/max/mdev = 0.475/0.662/0.951/0.181 ms
[root@localhost ~]#
```

Check that the ping is **working**.

4b: Linux

Different versions of Linux can use different firewall packages. The package used by Centos is '**firewalld**'. Verify that firewalld is active and enabled with **systemctl** status **firewalld**

```
[root@localhost ~]# systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor pre
   Active: active (running) since Thu 2022-08-18 11:49:35 AEST; 44min ago
     Docs: man:firewalld(1)
 Main PID: 1242 (firewalld)
   Tasks: 2 (limit: 12077)
  Memory: 31.0M
    CGroup: /system.slice/firewalld.service
           └─1242 /usr/libexec/platform-python -s /usr/sbin/firewalld --nofork -->
```

Verifying that firewall is **active** and **enabled**.

Install the **firewall-config** package.

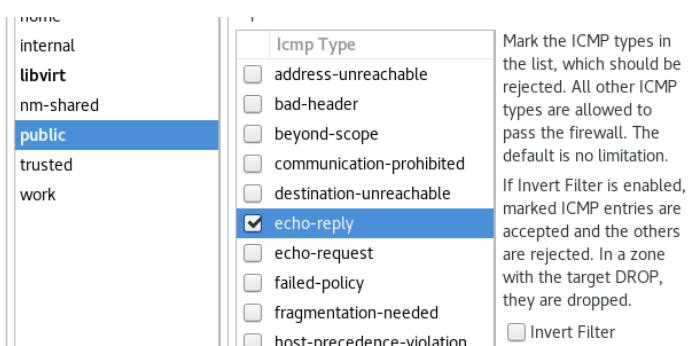
```
firewall-config-0.9.3-13.el8.noarch.rpm      1.0 MB/s | 161 kB     00:00
-----
Total                                         139 kB/s | 161 kB     00:01
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      :                                                 1/1
  Installing    : firewall-config-0.9.3-13.el8.noarch          1/1
  Running scriptlet: firewall-config-0.9.3-13.el8.noarch          1/1
  Verifying     : firewall-config-0.9.3-13.el8.noarch          1/1
Installed products updated.

Installed:
  firewall-config-0.9.3-13.el8.noarch

Complete!
```

The **firewall-config** package is **installed**.

Turn on the checkbox for **echo-reply**, Linux should **REJECT** the incoming pings.



Turning on the echo-reply.

Check that you now can't Ping from Windows.

```
C:\Users\Administrator>ping 10.0.2.1
Pinging 10.0.2.1 with 32 bytes of data:
Request timed out.
Request timed out.

Ping statistics for 10.0.2.1:
    Packets: Sent = 2, Received = 0, Lost = 2 (
Control-C
^C
C:\Users\Administrator>
```

Ping didn't work from Windows.

Task 5: Modifying Linux Network Configuration Files

The **statically configured Linux** network setup will **vanish** on reboot and be **replaced**. On Centos, the two main configuration files to configure static networking are

/etc/sysconfig/network and */etc/sysconfig/network-scripts/ifcfg-ensXX*

First, edit *ifcfg-ens33* (Eth0). **BOOTPROTO=dhcp**, the interface will be dynamically assigned an IP address by a DHCP server. There **shouldn't** be an **IPADDR** or **NETMASK** entry in this file. **DEFROUTE=yes** and **ONBOOT=yes**.

```
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=dhcp
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens33
UUID=9a5b99db-9450-44c5-aece-fbfb20f28e7d
DEVICE=ens33
ONBOOT=yes
```

Editing *the ifcfg-ens33* (Eth0).

Then edit *ifcfg-ens37* (or create it if it doesn't exist by copying ifcfg-ens33):

- DEVICE=ens37
- NAME=ens37
- Delete UUID line
- BOOTPROTO=none
- Set the IPADDR and NETMASK to the same values
- DEFROUTE=no
- ONBOOT=yes

```
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=none
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens37
DEVICE=ens37
ONBOOT=yes
IPADDR=10.0.2.1
NETMASK=255.255.255.0
```

Editing the *ifcfg-ens37* (Eth1).

Use the *ifdown* and *ifup* scripts in the network-scripts directory. Bring the interface down first, and then up:

ifdown ens37; ifup ens37

To **reload/restart** all networking, you can run: *systemctl restart NetworkManager.service*

Another way is to use the NetworkManager command *nmcli i*. Try the following commands and compare them with ifconfig/ifdown/ifup.

- *nmcli con show ens37*
- *nmcli con down ens37*
- *nmcli con up ens37*

A fourth way is to use the command *nmtui*.

```
[root@localhost ~]# nmcli con up ens37
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/4)
[root@localhost ~]# ifup ens37
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/5)
[root@localhost ~]# ifconfig ens37
ens37: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 10.0.2.1 netmask 255.255.255.0 broadcast 10.0.2.255
        inet6 fe80::dbc6:3151:621:ed84 prefixlen 64 scopeid 0x20<link>
          ether 00:0c:29:69:cff:f5 txqueuelen 1000 (Ethernet)
```

Activating *ens37* (Eth1).

```
[root@localhost ~]# nmcli con down ens37
Connection 'ens37' successfully deactivated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/7)
[root@localhost ~]# ifconfig ens37
ens37: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      ether 00:0c:29:69:cff:f5 txqueuelen 1000 (Ethernet)
        RX packets 656 bytes 63018 (61.5 KiB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 267 bytes 32203 (31.4 KiB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Deactivating *ens37* (Eth1).

Lab 3b Establishing and maintaining time and date settings

Task 1: Set the Time Zone

Use the *date* command to identify the current UTC and local times. If the local time zone is incorrectly identified, change */etc/localtime* under the directory */usr/share/zoneinfo/*.

Try setting your system to a different time zone to verify that you can.

```
[root@localhost ~]# date
Thu Aug 18 12:04:02 AEST 2022
[root@localhost ~]# date -u
Thu Aug 18 02:04:04 UTC 2022
[root@localhost ~]#
[root@localhost ~]# mv /etc/localtime /etc/localtime.bak
mv: overwrite '/etc/localtime.bak'? y
[root@localhost ~]#
[root@localhost ~]# ln -s -f /usr/share/zoneinfo/Australia/Perth /etc/localtime
bash: ln-s: command not found...
[root@localhost ~]# ln -s -f /usr/share/zoneinfo/Australia/Perth /etc/localtime
[root@localhost ~]# date
Thu Aug 18 10:05:02 AWST 2022
[root@localhost ~]# ln -s -f /usr/share/zoneinfo/Australia/Sydney /etc/localtime

[root@localhost ~]# date
Thu Aug 18 12:05:12 AEST 2022
[root@localhost ~]#
```

Identifying and setting system time zone.

Task 2: Maintain Correct Time with Chrony

In the latest versions of Centos, **chrony** is the system that maintains network time. Older systems use **ntpd**. Chrony consists of both a server (daemon) and a client, called **chronyd** and **chronyc** respectively.

See whether **chronyd** is already running using the command **ps -ef | grep chronyd**. Then see whether **chronyd** is known by **systemctl status chronyd**.

```
[root@localhost ~]# systemctl status chronyd
● chronyd.service - NTP client/server
  Loaded: loaded (/usr/lib/systemd/system/chronyd.service; enabled; vendor pre>
  Active: active (running) since Fri 2022-09-02 22:14:16 AEST; 2 days ago
    Docs: man:chronyd(8)
          man:chrony.conf(5)
 Main PID: 1176 (chronyd)
   Tasks: 1 (limit: 12077)
  Memory: 1.4M
   CGroup: /system.slice/chronyd.service
           └─1176 /usr/sbin/chronyd
```

Chronyd is known by system

Use **systemctl** to start and enable chronyd.

```
[root@localhost ~]# ps -ef | grep chronyd
root      3409     3280  0 21:08 pts/0    00:00:00 grep --color=auto chronyd
[root@localhost ~]# systemctl is-active chronyd
inactive
[root@localhost ~]# systemctl enable chronyd
Created symlink /etc/systemd/system/multi-user.target.wants/chronyd.service → /u
sr/lib/systemd/system/chronyd.service.
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# systemctl start chronyd
[root@localhost ~]# ps -ef | grep chronyd
chrony    3485      1  0 21:09 ?        00:00:00 /usr/sbin/chronyd
root      3498     3280  0 21:10 pts/0    00:00:00 grep --color=auto chronyd
```

Starting and enabling the chronyd.

Verify that chrony is working by:

- **chronyc sources**
- **chronyc tracking**
- **chronyc add server time.uts.edu.au**

```
[root@localhost ~]#
[root@localhost ~]# chronyc sources
MS Name/IP Address          Stratum Poll Reach LastRx Last sample
=====
^+ ec2-13-55-50-68.ap-south>    3   6   17   2   -3208us[-1935us] +/-  74ms
^* thomas-avatar.bnrla        2   6   33   0   +2330us[+3603us] +/-  47ms
^+ 202-142-132-8.cae84.mel>  2   6   17   2   +6832us[+8105us] +/-  57ms
^+ super-quarter.bnrla       2   6   17   2   -87us[+1186us]  +/-  60ms
[root@localhost ~]#
[root@localhost ~]# chronyc tracking
Reference ID : 70052214 (thomas-avatar.bnrla)
Stratum      : 3
Ref time (UTC): Thu Aug 18 01:34:45 2022
System time  : 0.000000959 seconds slow of NTP time
Last offset  : +0.001273137 seconds
RMS offset   : 0.001273137 seconds
Frequency    : 0.000 ppm slow
Residual freq: +180.327 ppm
Skew         : 1000000.000 ppm
Root delay   : 0.013250327 seconds
Root dispersion: 24.318338394 seconds
Update interval: 2.1 seconds
Leap status   : Normal
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# chronyc add server time.uts.edu.au
200 OK
[root@localhost ~]# chronyc sources
MS Name/IP Address          Stratum Poll Reach LastRx Last sample
=====
^+ ec2-13-55-50-68.ap-south>    3   6   17   42  -3208us[-1935us] +/-  74ms
^* thomas-avatar.bnrla        2   6   33   40  +2330us[+3603us] +/-  47ms
^+ 202-142-132-8.cae84.mel>  2   6   17   42  +6832us[+8105us] +/-  57ms
^+ super-quarter.bnrla       2   6   17   42  -87us[+1186us]  +/-  60ms
^? B1L2-c7206.gw.uts.edu.au  2   6   1    5   -1355us[-1355us] +/-  18ms
[root@localhost ~]#
```

Verified that Chrony is working.

Test that it will synchronize by changing the system date: **date 123123591999.00**

```
[root@localhost ~]# date
Thu Aug 18 11:36:59 AEST 2022
[root@localhost ~]# date 123123591999.00
Fri Dec 31 23:59:00 AEDT 1999
[root@localhost ~]# date
Fri Dec 31 23:59:01 AEDT 1999
[root@localhost ~]# date
Fri Dec 31 23:59:26 AEDT 1999
[root@localhost ~]#
```

Changing the current date to 1999 and check the changes.

Does it reset to the current date/time? Why or why not?

The date **doesn't reset** since it was already changed before.

Restart chronyd, using systemctl. Then run the date command a few times. **What happens now? Why?**

```
[root@localhost ~]# date
Fri Dec 31 23:59:26 AEDT 1999
[root@localhost ~]#
[root@localhost ~]# systemctl restart chronyd
[root@localhost ~]#
[root@localhost ~]# date
Sat Jan  1 00:00:06 AEDT 2000
[root@localhost ~]# date
Thu Aug 18 11:36:59 AEST 2022
```

Restarting the chronyd. The date is **reset back** to the beginning. It happens because the date is **connected to the system**, so it will be reset.

Look at the configuration file **/etc/chrony.conf**.

```
GNU nano 2.9.8                               chrony.conf
# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (http://www.pool.ntp.org/join.html).
pool 2.centos.pool.ntp.org iburst

# Record the rate at which the system clock gains/losses time.
driftfile /var/lib/chrony/drift

# Allow the system clock to be stepped in the first three updates
# if its offset is larger than 1 second.
makestep 1.0 3

# Enable kernel synchronization of the real-time clock (RTC).
rtcsync

# Enable hardware timestamping on all interfaces that support it.
#hwtimestamp *

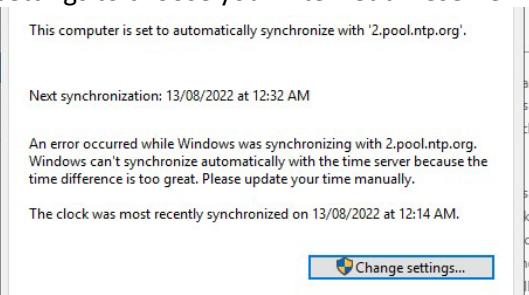
# Increase the minimum number of selectable sources required to adjust
# the system clock.
#minsources 2

# Allow NTP client access from local network.
#allow 192.168.0.0/16
```

Configuration file of **/etc/chrony.conf**.

Task 3: Windows NTP

Windows can also **automatically update** time and date via **NTP**. In Server Manager / Local Server, change the settings to choose your internet timeserver. The **default** is **time.microsoft.com** → change to: **2.pool.ntp.org**



Try changing the current time to **2.pool.ntp.org**.

Week 4

Lecture 4 Advanced Networking (Dynamic Networking)

DHCP

- Introduction
 - DHCP reduce network complexity and administration
 - Automatically obtain network config from a DHCP server
 - User UDP port 67/68
 - Uses subnet FF:FF:FF:FF
- Address Allocation
 - Dynamic IP Config
 - Allocate “lease” IP address from a ‘pool’ of addresses
 - Could reserve IP address for a specific host

- Operations

DHCP Client

- Broadcasts **DHCPDISCOVER** UDP packet on local subnet
- Chooses lease & broadcasts **DHCPOFFER** packet
- Reply with **DCHPDECLINE** if duplicate
- Reply with **DCHPRELEASE** when finished

DHCP Server

- Broadcasts **DHCPOFFER** with lease info
- Replies with **DCHPPACK**, **DCHPNACK**



- Scope & Reservation

Scope: range of IP addresses for leasing

Reservation: fixed of IP addresses for specific DHCP client



- Standard Operation

Packages and daemon: *dhcp-server*; *dhclient*; *dhcpd*

Global option: */etc/dhcp/dhcpd.conf*

Individual subnet config: */etc/dhcp/dhcpd.conf*

```
subnet 10.5.5.0 netmask 255.255.255.224 {  
    range 10.5.5.26 10.5.5.30;  
    option domain-name-servers ns1.internal.example.org;  
    option domain-name "internal.example.org";  
    option routers 10.5.5.1;  
    option broadcast-address 10.5.5.31;  
    default-lease-time 600;  
    max-lease-time 7200;  
}  
  
host fantasia {  
    hardware ethernet 08:00:07:26:c0:a5;  
    fixed-address 10.5.5.10;  
    option domain-name-servers 8.8.8.8;  
}
```

- Network Configuration

FILES

- */etc/sysconfig/network-scripts/ifcfg-ens33*
- */etc/hostname*
- */etc/resolv.conf*

IFCFG FILE

- Vim */etc/sysconfig/network-scripts/ifcfg-ens33*
- *nmcli c reload ens37*
- *nmcli c up ens3*

Lab 4a Configuring a DHCP Client

Task 1: Linux Command-Line Configuration

Test network connectivity using *ping* and *ssh*. Check MV's IP address with *ifconfig* or *nmcli*.

```
[root@localhost ~]# ifconfig ens33
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.209.128 netmask 255.255.255.0 broadcast 192.168.209.255
          inet6 fe80::dc4:b003:eda6:2840 prefixlen 64 scopeid 0x20<link>
            ether 00:0c:29:69:cf:eb txqueuelen 1000 (Ethernet)
              RX packets 12332 bytes 17675734 (16.8 MiB)
              RX errors 0 dropped 0 overruns 0 frame 0
              TX packets 1348 bytes 91504 (89.3 KiB)
              TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@localhost ~]# ping www.google.com -c3
PING www.google.com (142.250.66.228) 56(84) bytes of data.
64 bytes from syd15s15-in-f4.1e100.net (142.250.66.228): icmp_seq=1 ttl=128 time=195 ms
64 bytes from syd15s15-in-f4.1e100.net (142.250.66.228): icmp_seq=2 ttl=128 time=127 ms
64 bytes from syd15s15-in-f4.1e100.net (142.250.66.228): icmp_seq=3 ttl=128 time=231 ms

--- www.google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 127.243/184.501/230.819/42.986 ms
[root@localhost ~]#
```

Network is connected.

Verify the configuration files so that it will automatically send a DHCP query to retrieve network settings. Change **BOOTPROTO** to "dhcp" in the interface-specific configuration file in */etc/sysconfig/network-scripts/ifcfg-ens**

```
[root@localhost ~]# cat /etc/sysconfig/network-scripts/ifcfg-ens33
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=dhcp
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens33
UUID=9a5b99db-9450-44c5-aece-fbfb20f28e7d
DEVICE=ens33
```

Configuring the interface-specific configuration file.

Reboot VM and test the network that the IP address appears to have been **dynamically allocated**. It could be done by just ifdown and ifup, but rebooting ensures all caches, etc are cleared.

```
[root@localhost ~]# ifconfig ens33
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.209.128 netmask 255.255.255.0 broadcast 192.168.209.255
          inet6 fe80::dc4:b003:eda6:2840 prefixlen 64 scopeid 0x20<link>
            ether 00:0c:29:69:cf:eb txqueuelen 1000 (Ethernet)
              RX packets 12499 bytes 17691583 (16.8 MiB)
              RX errors 0 dropped 0 overruns 0 frame 0
              TX packets 1509 bytes 106774 (104.2 KiB)
              TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@localhost ~]# nmcli con down ens33
Connection 'ens33' successfully deactivated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/2)
[root@localhost ~]#
```

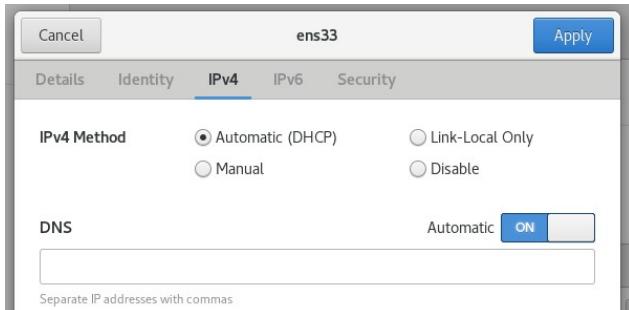


```
Aug 25 11:50:09 localhost journal[2359]: Failed to StopUnit service: GDBus.Error :org.freedesktop.systemd1.NoSuchUnit: Unit gnome-user-share-webdav.service not loaded.
Aug 25 11:50:09 localhost journal[2359]: Failed to StopUnit service: GDBus.Error :org.freedesktop.systemd1.NoSuchUnit: Unit rygel.service not loaded.
Aug 25 11:50:09 localhost chrony[1151]: Source 129.250.35.251 offline
Aug 25 11:50:09 localhost chrony[1151]: Source 220.158.215.21 offline
Aug 25 11:50:09 localhost chrony[1151]: Source 129.250.35.250 offline
Aug 25 11:50:09 localhost chrony[1151]: Can't synchronise: no selectable source
s
Aug 25 11:50:09 localhost chrony[1151]: Source 139.99.222.72 offline
Aug 25 11:50:19 localhost systemd[1]: NetworkManager-dispatcher.service: Succeed
[root@localhost ~]# dhclient ens33
[root@localhost ~]#
Aug 25 11:55:22 localhost dhclient[4592]: DHCPRELEASE on ens33 to 192.168.209.25
4 port 67 (xid=0x44be4907)
Aug 25 11:55:22 localhost avahi-daemon[1138]: Withdrawing address record for 192
.168.209.128 on ens33.
Aug 25 11:55:22 localhost avahi-daemon[1138]: Leaving mDNS multicast group on in
terface ens33.IPv4 with address 192.168.209.128.
```

The update configuration is verified.

Task 2: Linux GUI Configuration

Configure using GUI via the Settings icon → Network menu item, or running either *gnome-control-center* or *nm-connection-editor*



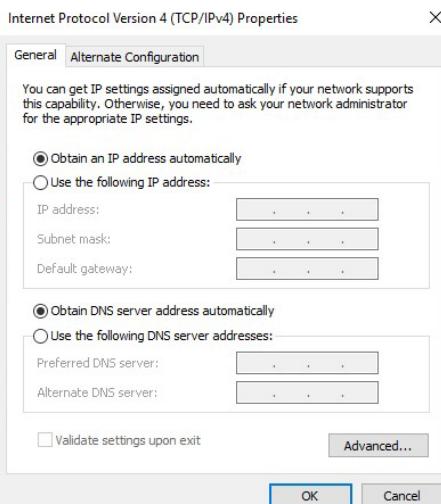
Change the configuration using Linux GUI configuration

Task 3: Windows configuration

On Windows Server, it is useful to know the command-line versions to create scripts to automate management of servers:

1. Server Manager → View Network Connections
2. Choose the appropriate adapter & select Properties
3. In the Items listing, choose Internet Protocol v4 (TCP/IPv4) and choose Properties
4. Set the “Obtain an IP address automatically” radio button

Investigate by using command: *netsh*



Changing the DHCP configuration in Windows server.

SUBCOMMANDS:

Show configuration: *netsh interface ip show config*

Set interface to use DHCP: *netsh interface ip set address "Local Area Connection" dhcp*

Set interface to use static IP address: *netsh interface ip set address name="Local Area Connection" static 10.0.2.11 255.255.255.0 10.0.2.1*

Explore other settings of netsh which is: `netsh dump`

```
C:\Users\Administrator>netsh interface ip show config

Configuration for interface "Ethernet0"
  DHCP enabled: Yes
  IP Address: 192.168.209.133
  Subnet Prefix: 192.168.209.0/24 (mask 255.255.255.0)
  Default Gateway: 192.168.209.2
  Gateway Metric: 0
  InterfaceMetric: 25
  DNS servers configured through DHCP: 192.168.209.2
  Register with which suffix: Primary only
  WINS servers configured through DHCP: 192.168.209.2

Configuration for interface "Ethernet1"
  DHCP enabled: No
  IP Address: 10.0.2.2
  Subnet Prefix: 10.0.2.0/24 (mask 255.255.255.0)
  Default Gateway: 10.0.2.1
  Gateway Metric: 256
  InterfaceMetric: 25
  Statically Configured DNS Servers: None
  Register with which suffix: Primary only
  Statically Configured WINS Servers: None

Configuration for interface "Loopback Pseudo-Interface 1"
  DHCP enabled: No
  IP Address: 127.0.0.1
  Subnet Prefix: 127.0.0.0/8 (mask 255.0.0.0)
  InterfaceMetric: 75
  Statically Configured DNS Servers: None
  Register with which suffix: None
  Statically Configured WINS Servers: None
```

Verifying that the configuration **successfully changed**.

Lab 4b Configuring a DHCP Server on Linux

Task 1: Install DHCP Server on Linux

Install the standard dhcp-server on Linux by using yum.

```
Transaction test succeeded.
Running transaction
  Preparing : 1/1
  Running scriptlet: dhcp-server-12:4.3.6-40.el8.x86_64 1/1
  Installing : dhcp-server-12:4.3.6-40.el8.x86_64 1/1
  Running scriptlet: dhcp-server-12:4.3.6-40.el8.x86_64 1/1
  Verifying : dhcp-server-12:4.3.6-40.el8.x86_64 1/1
Installed products updated.

Installed:
  dhcp-server-12:4.3.6-40.el8.x86_64

Complete!
```

Standard dhcp-server is **installed**.

Task 2: Configure DHCP server on Linux

Designing a subnet based on:

Subnet:	10.0.2.0/24
Gateway:	10.0.2.1
DNS:	10.0.2.1
Reserved space for	10.0.0.2 → 10.0.2.127
Dynamically allocated space for workstations	10.0.2.128 → 10.0.2.254

Sample file of dhcpd.conf provided at: `/usr/share/doc/dhcp-server/dhcpd.conf.example`
Then use vim to change the file at: `/etc/dhcp/dhcpd.conf file`

Parameters to use:

- Set **default-lease-time** to 60 and **max-lease-time** to 600 (10 minutes)
- Create a subnet range 10.0.2.129-10.0.2.254
- Options routers 10.0.2.1; domain-name-servers 10.0.2.1, domain-name whatever.localdomain

```

subnet 10.0.2.0 netmask 255.255.255.0 {
    range 10.0.2.128 10.0.2.254;
    option domain-name-servers 10.0.2.1;
    option domain-name "online.uts.edu.au";
    option routers 10.0.2.1;
    default-lease-time 60;
    max-lease-time 600;
}

```

Editing the real */etc/dhcp/dhcpd.conf file*

Then, make change for the DHCP server on ens37

```

TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=none
IPADDR=10.0.2.1
PREFIX=24
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens37
UUID=a8b75431-25ce-4970-8065-cec287043c2d
DEVICE=ens37
ONBOOT=yes

```

The DHCP on ens37 is **settled up**.

Task 3: Start and Monitor the DHCP Server on Linux

Start the dhcpd service using *systemctl* and watch the DHCP by using *-f* option on the *tail* command to see the written entries on system log file. *tail -f /var/log/messages*

Confirm that the DHCP is on the correct interface by seeing the log file. Another way to see DHCP database: */var/lib/dhcpd/dhcpd.leases*

```

root@localhost:~#
File Edit View Search Terminal Help
[root@localhost ~]# systemctl start dhcpcd
[root@localhost ~]# systemctl restart dhcpcd
[root@localhost ~]# systemctl restart dhcpcd
[root@localhost ~]# vim /etc/dhcp/dhcpd.conf
[root@localhost ~]# cat /var/lib/dhcpd/dhcpd.leases
# The format of this file is documented in the dhcpcd.leases(5) manual page.
# This lease file was written by isc-dhcp-4.3.6

# authoring-byte-order entry is generated, DO NOT DELETE
authoring-byte-order little-endian;

server-duid "\000\001\000\001*\231\252G\000\014)i\317\365";

```

```

dhcpcd
Aug 25 13:41:10 localhost dhcpcd[39559]:  in your dhcpd.conf file for the network segment
Aug 25 13:41:10 localhost dhcpcd[39559]:  to which interface virbr0 is attached.
Aug 25 13:41:10 localhost dhcpcd[39559]:
Aug 25 13:41:10 localhost dhcpcd[39559]: Listening on LPF/ens37/00:0c:29:69:cf:f5
Aug 25 13:41:10 localhost dhcpcd[39559]: Sending on   LPF/ens37/00:0c:29:69:cf:f5
Aug 25 13:41:10 localhost dhcpcd[39559]: 10.0.2.0/24

```

DHCP server on ens37 is **verified**.

Task 4: Set Up Windows DHCP Client & Test the DHCP Server

Way to **set up** DHCP on Windows: Network Connections → Ethernet1 → Properties → Internet Protocol Version 4 (TCP/IPv4) → Properties

Then after changes, refresh it.

```

Ethernet adapter Ethernet0:

Connection-specific DNS Suffix . : localdomain
Link-local IPv6 Address . . . . . : fe80::35f3:64ed:c2eb:fb89%7
IPv4 Address . . . . . : 192.168.209.133
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.209.2

Ethernet adapter Ethernet1:

Connection-specific DNS Suffix . : online.uts.edu.au
Link-local IPv6 Address . . . . . : fe80::95cd:dd8:942:5387%16
IPv4 Address . . . . . : 10.0.2.128
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 10.0.2.1

C:\Users\Administrator>

```

DHCP client on Windows is **set up**.

On the Linux DHCP server, check the content of:

```
client-hostname "Agapitusserver";
}
lease 10.0.2.128 {
    starts 4 2022/08/25 03:51:34;
    ends 4 2022/08/25 03:56:34;
    cltt 4 2022/08/25 03:51:34;
    binding state active;
    next binding state free;
    rewind binding state free;
    hardware ethernet 00:0c:29:67:a9:57;
    uid "\001\000\014)\025IW";
    set vendor-class-identifier = "MSFT 5.0";
    client-hostname "Agapitusserver";
}
[root@localhost ~]#
```

Check [/var/lib/dhcpd/dhcpd.leases](#)

View the dhcpd.leases about the windows machine.

Verify the MAC address on ens37 using command:

```
[root@localhost ~]#
[root@localhost ~]# arp -i ens37
Address      HWtype  HWaddress          Flags Mask   Iface
10.0.2.2     ether    00:0c:29:67:a9:57  C      ens37
10.0.2.128   ether    00:0c:29:67:a9:57  C      ens37
[root@localhost ~]#
```

The MAC address on ens37 is **confirmed**.

See the **log file** for DHCP messages for the message:

- **DHCPDISCOVER**
- **DHCPOFFER**
- **DHCPREQUEST**
- **DHCPACK**

```
Aug 25 13:44:36 localhost journal[2402]: unable to get EDID for xrandr-Virtual-1: una
ble to get EDID for output
Aug 25 13:46:33 localhost dhcpd[39559]: DHCPDISCOVER from 00:0c:29:67:a9:57 via ens37
Aug 25 13:46:34 localhost dhcpd[39559]: DHCPOFFER on 10.0.2.128 to 00:0c:29:67:a9:57
(Agapitusserver) via ens37
Aug 25 13:46:34 localhost dhcpd[39559]: DHCPREQUEST for 10.0.2.128 (10.0.2.1) from 00
:0c:29:67:a9:57 (Agapitusserver) via ens37
Aug 25 13:46:34 localhost dhcpd[39559]: DHCPACK on 10.0.2.128 to 00:0c:29:67:a9:57 (A
gapitusserver) via ens37
Aug 25 13:49:04 localhost dhcpd[39559]: DHCPREQUEST for 10.0.2.128 from 00:0c:29:67:a
9:57 (Agapitusserver) via ens37
Aug 25 13:49:04 localhost dhcpd[39559]: DHCPACK on 10.0.2.128 to 00:0c:29:67:a9:57 (A
gapitusserver) via ens37
Aug 25 13:56:40 localhost dbus-daemon[1126]: [system] Activating via systemd: service
```

Confirm the request response sequence form from the log file.

Task 5: Set Up Reserved Addresses

Find out the Windows Server's MAC address by searching it in [/var/lib/dhcpd/dhcpd.leases](#)
Then edit the DHCP configuration by adding a host entry

```
subnet 10.0.2.0 netmask 255.255.255.0{
    range 10.0.2.128 10.0.2.254;
    option domain-name-servers 10.0.2.1;
    option domain-name "online.uts.edu.au";
    option routers 10.0.2.1;
    default-lease-time 60;
    max-lease-time 600;
}
host win {
    hardware ethernet 00:0c:29:67:a9:57;
    fixed-address 10.0.2.20;
}
```

Host entry of Windows is **added**.

Use **systemctl restart dhcpd** to restart DHCP server.
Test Windows Server client and see the new static address.

```
Ethernet adapter Ethernet0:
Connection-specific DNS Suffix . : localdomain
Link-local IPv6 Address . . . . . : fe80::ec26:b2a6:9935:666c%7
IPv4 Address . . . . . : 192.168.37.138
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.37.2

Ethernet adapter Ethernet1:
Connection-specific DNS Suffix . : online.uts.edu.au
Link-local IPv6 Address . . . . . : fe80::e02c:5917:4c19:ddea%16
IPv4 Address . . . . . : 10.0.2.20
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 10.0.2.1
```

Checking the new static address on Windows.

On the Linux machine, check the arp: ***arp -i ens37***

What do you notice? What happens to the old lease?

```
dhcpd[36344]: DHCPDISCOVER from 00:0c:29:67:a9:57 via ens37
dhcpd[36344]: DHCPOFFER on 10.0.2.20 to 00:0c:29:67:a9:57 via ens37
dhcpd[36344]: uid lease 10.0.2.128 for client 00:0c:29:67:a9:57 is duplicate on 10
dhcpd[36344]: DHCPREQUEST for 10.0.2.20 (10.0.2.1) from 00:0c:29:67:a9:57 via ens37
dhcpd[36344]: DHCPACK on 10.0.2.20 to 00:0c:29:67:a9:57 via ens37
```

Checking message on Linux logs.

```
[root@ffff ~]# arp -i ens37
Address          HWtype  HWaddress          Flags Mask   Ifa
10.0.2.20        ether    00:0c:29:67:a9:57  C      ens
10.0.2.128       ether    00:0c:29:67:a9:57  C      ens
```

Checking the arp message.

Since it is a **fixed address**, the **old lease will not be recorded**

Week 5

Lecture 5 A Users, Groups, Directories

User

- Manage User Accounts
 - Users contain UID and a username
 - Each user joined for a group(s), which has each GID
- Policies
 - Allocating UID
 - Groups that contain a user
 - Generating a usernames
 - Security features
- COMMAND
 - /etc/passwd (globally readable)
 - /etc/shadow (only readable for root user)
 - /etc/group (the availability of groups)

Group

- Policies
 - Not able to read other files
 - Able to share files if on the same projects
- Add/Modify/Delete User
 - *useradd [options] username*
 - *usermod [options] username*
 - *userdel [options] username*
- Add/Modify/Delete Groups
 - *groupadd [options] username*
 - *groupmod [options] username*
 - *groupdel [options] username*

- Manage accounts
 - **COMMANDS:** `passwd -1` and `passwd -u`
 - *To lock a password, put!! Or * in the password field*
 - *To lock a username, set shell to /sbin/nologin or equivalent*
 -
- System Issues
 - Terminal login
 - Local login (TTY): /etc/issue
 - Network login (PTS): /etc/issue.net
 - Message of the Day (TTY/PTS): /etc/motd

Lab 5a Managing Users and Groups

Task 1: Creating users with command-line tools

Using the command-line tools (`useradd`, etc), create a username peter with the default uid and gid. Peter's full name is "Peter Griffin" and his preferred login shell is zsh (the Z shell). Set a password for Peter!

```
[root@localhost ~]# useradd -c "Peter Griffin" -s /bin/zsh peter
[root@localhost ~]#
[root@localhost ~]# passwd peter
Changing password for user peter.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[root@localhost ~]#
```

A peter username with preferred login shell zsh and add a password is **created**.

Edit the `etc/skel` to add README file.

```
passwd: all authentication tokens updated successfully.
[root@localhost ~]# ls -la /etc/skel/
total 32
drwxr-xr-x.  3 root root   92 Aug 13 08:37 .
drwxr-xr-x. 152 root root 12288 Sep  1 10:58 ..
-rw-r--r--.  1 root root   18 Apr  8 09:05 .bash_logout
-rw-r--r--.  1 root root  141 Apr  8 09:05 .bash_profile
-rw-r--r--.  1 root root   376 Apr  8 09:05 .bashrc
drwxr-xr-x.  4 root root   39 Jan 20 2022 .mozilla
-rw-r--r--.  1 root root  658 Jun  9 19:12 .zshrc
[root@localhost ~]# touch /etc/skel/README
[root@localhost ~]# ls -la /etc/skel/
total 32
drwxr-xr-x.  3 root root  106 Sep  1 11:03 .
drwxr-xr-x. 152 root root 12288 Sep  1 10:58 ..
-rw-r--r--.  1 root root   18 Apr  8 09:05 .bash_logout
-rw-r--r--.  1 root root  141 Apr  8 09:05 .bash_profile
-rw-r--r--.  1 root root   376 Apr  8 09:05 .bashrc
drwxr-xr-x.  4 root root   39 Jan 20 2022 .mozilla
-rw-r--r--.  1 root root    0 Sep  1 11:03 README
-rw-r--r--.  1 root root  658 Jun  9 19:12 .zshrc
[root@localhost ~]#
```

README file is **added** into the skeleton directory.

Create a second user with username stewie with the default uid and gid. Stewie's full name is "Stewie Griffin", and his preferred login shell is bash. Set a password for Stewie.

```
[root@localhost ~]# useradd -c "Stewie Griffin" -s /bin/bash stewie
[root@localhost ~]# passwd stewie
Changing password for user stewie.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[root@localhost ~]#
```

A stewie username with preferred login shell bash and add a password is **created**.

```
[root@localhost ~]# ls -la /home/stewie
total 16
drwx----- 3 stewie stewie 106 Sep  1 11:05 .
drwxr-xr-x. 4 root   root   33 Sep  1 11:05 ..
-rw-r--r--  1 stewie stewie 18 Apr  8 09:05 .bash_logout
-rw-r--r--  1 stewie stewie 141 Apr  8 09:05 .bash_profile
-rw-r--r--  1 stewie stewie 376 Apr  8 09:05 .bashrc
drwxr-xr-x  4 stewie stewie 39 Jan 20 2022 .mozilla
-rw-r--r--  1 stewie stewie  0 Sep  1 11:03 README
-rw-r--r--  1 stewie stewie 658 Jun  9 19:12 .zshrc
```

README file is **added** to the stewie directory.

Create a third user with username brian with uid 200 and make Brian's default group to be the group called "users". Brian's full name is "Brian Griffin", and his preferred login shell is bash. Set a password for Brian.

```
[root@localhost ~]# useradd -c "Brian Griffin" -u 200 -g users -s /bin/bash brian
[root@localhost ~]# passwd brian
Changing password for user brian.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[root@localhost ~]#
```

A username brian with uid 200, default group "users", and preferred login shell bash is **created**.

Test the users by log on into each account.

```
[root@localhost ~]# ssh peter@localhost
peter@localhost's password:
Activate the web console with: systemctl enable --now cockpit.socket

id
[peter@localhost]~% id
uid=1000(peter) gid=1000(peter) groups=1000(peter)
[peter@localhost]~%
```

Logging in as peter and checking the details.

```
[root@localhost ~]# ssh stewie@localhost
stewie@localhost's password:
Activate the web console with: systemctl enable --now cockpit.socket

[stewie@localhost ~]$ id
uid=1001(stewie) gid=1001(stewie) groups=1001(stewie)
[stewie@localhost ~]$
```

Logging in as stewie and checking the details.

```
[root@localhost ~]# ssh brian@localhost
The authenticity of host 'localhost (::1)' can't be established.
ECDSA key fingerprint is SHA256:p3h5a0wPvn5kt5KGa8J/X4j5ibRmwCnmQih0dhnvTg0.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
brian@localhost's password:
Activate the web console with: systemctl enable --now cockpit.socket

[brian@localhost ~]$ id
uid=200(brian) gid=100(users) groups=100(users)
[brian@localhost ~]$
```

Logging in as brian and checking the details.

Task 2: Configuring groups and assigning users to groups

Log in as Stewie or Brian and verify their group membership by running the “id” command. Create a new file (for example, by using the touch command). Who is the group owner?

What is the mechanism by which “newgrp” works? Use the ps command to see how many shells you have running now. What would be the most sensible way to reverse the effects of newgrp?

```
[root@localhost ~]# groupadd family
[root@localhost ~]# usermod -G family stewie
[root@localhost ~]# usermod -G family brian
[root@localhost ~]# tail /etc/group
postdrop:x:90:
postfix:x:89:
rngd:x:974:
dovecot:x:97:
dovenull:x:973:
tcpdump:x:72:
dhcpcd:x:177:
peter:x:1000:
stewie:x:1001:
family:x:1002:stewie,brian
[root@localhost ~]#
```

Creating a group called family and adding stewie and brian into it. Then checking the details.

```
[brian@localhost ~]$ id
uid=200(brian) gid=100(users) groups=100(users),1002(family)
[brian@localhost ~]$ touch a
[brian@localhost ~]$ ls -la a
-rw-r--r-- 1 brian users 0 Sep 1 11:43 a
```

Logging in as brian and checking the group's membership. Then creating 'a' file and the access into the file. The **group owner** is **users** since users is the **primary group**.

```
[brian@localhost ~]$ ps
 PID TTY          TIME CMD
 5963 pts/2    00:00:00 bash
 6027 pts/2    00:00:00 ps
```

Running the ps command and it shown the **PID** of the **bash** is **5963**.

```
[brian@localhost ~]$ newgrp family
[brian@localhost ~]$ touch b
[brian@localhost ~]$ ls -la b
-rw-r--r-- 1 brian family 0 Sep 1 11:46 b
[brian@localhost ~]$ id
uid=200(brian) gid=1002(family) groups=1002(family),100(users)
```

Changing the current group to family and by seeing from the id command, it showed that the **family group** is the **primary group**. Then creating 'b' file and verified that the **ownership is taken by family group**.

The mechanism by newgrp command is the command used to change the current group ID during a login session. If the optional - flag is given, the user's environment will be reinitialized as though the user had logged in, otherwise the current environment, including current working directory, remains unchanged.

The ps command shows 2 shells which both of them created after having a change to the primary group.

The way to reverse the effects of newgrp, we could use the - flag in the newgrp command. By using this flag, the environment will be reinitialized as though the user had logged in, otherwise the current environment, including current working directory, remains unchanged.

The different between the primary group and other group is that the primary group is the one that's recorded in the /etc/passwd file, configured when an account is set up. While the secondary groups are those that users might be added to once they already have accounts.

Task 3: Modifying user account settings

Edit Peter's account to be expired in 5 days.

```
[root@localhost ~]# usermod -e 2022-09-06 peter
[root@localhost ~]# tail /etc/shadow
avahi:!::18462::::::
postfix:!::18462::::::
rngd:!::18462::::::
dovecot:!::18462::::::
dovenull:!::18462::::::
tcpdump:!::18462::::::
dhcpcd:!::19229::::::
peter:$6$XnX4Igp4FTIJ1xrZ$IBquuFHHpKClfsguBkUNC6/kKpGS1ukfqqLV5nRRsQxbp62uoUddEbrkANTv
AcQvd8upqqdSwuI138bNdXvs1:19236:0:99999:7::19241:
stewie:$6$Eb6Fd0x45PPqewlh$B4Z9MdpoGuw9QTjVgAXQD/jt4208H8DBPzJxPB56CxpU18Bih44ztiXY80M
brian:$6$L1.18ksp5CB4ZX0m$Xe0xiDusv6i9HOrifHkZzhfRHw7DiYozAGtJwTuluL6pEDezhV3yYEIzJFI
CmGzoQRjE88W6gz1rE6lNtdRa/:19236:0:99999:7:::
[root@localhost ~]#
```

Modifying peter account so it would expire in 5 days. Checking the change in */etc/shadow*.

Check the aging parameters of Peter's password with: *chage -l peter*

```
[root@localhost ~]# chage -l peter
Last password change : Sep 01, 2022
Password expires      : never
Password inactive     : never
Account expires        : Sep 06, 2022
Minimum number of days between password change : 0
Maximum number of days between password change : 99999
Number of days of warning before password expires : 7
[root@localhost ~]#
```

Shown that peter account will expires in 5 days.

Set Stewie's account so that he is required to change his password at least every 5 days.

```
[root@localhost ~]# chage -M 5 stewie
[root@localhost ~]# tail /etc/shadow
avahi:!::18462::::::
postfix:!::18462::::::
rngd:!::18462::::::
dovecot:!::18462::::::
dovenull:!::18462::::::
tcpdump:!::18462::::::
dhcpcd:!::19229::::::
peter:$6$XnX4Igp4FTIJ1xrZ$IBquuFHHpKClfsguBkUNC6/kKpGS1ukfqqLV5nRRsQxbp62uoUddEbrkANTv
AcQvd8upqqdSwuI138bNdXvs1:19236:0:99999:7::19241:
stewie:$6$Eb6Fd0x45PPqewlh$B4Z9MdpoGuw9QTjVgAXQD/jt4208H8DBPzJxPB56CxpU18Bih44ztiXY80M
brian:$6$L1.18ksp5CB4ZX0m$Xe0xiDusv6i9HOrifHkZzhfRHw7DiYozAGtJwTuluL6pEDezhV3yYEIzJFI
CmGzoQRjE88W6gz1rE6lNtdRa/:19236:0:99999:7:::
[root@localhost ~]#
```

Stewie's account is modified.

Lock Brian's account: *usermod -L*

```
[root@localhost ~]# usermod -L brian
[root@localhost ~]# tail /etc/shadow
avahi:!::18462::::::
postfix:!::18462::::::
rngd:!::18462::::::
dovecot:!::18462::::::
dovenull:!::18462::::::
tcpdump:!::18462::::::
dhcpcd:!::19229::::::
peter:$6$XnX4Igp4FTIJ1xrZ$IBquuFHHpKClfsguBkUNC6/kKpGS1ukfqqLV5nRRsQxbp62uoUddEbrkANTv
AcQvd8upqqdSwuI138bNdXvs1:19236:0:99999:7::19241:
stewie:$6$Eb6Fd0x45PPqewlh$B4Z9MdpoGuw9QTjVgAXQD/jt4208H8DBPzJxPB56CxpU18Bih44ztiXY80M
brian:$6$L1.18ksp5CB4ZX0m$Xe0xiDusv6i9HOrifHkZzhfRHw7DiYozAGtJwTuluL6pEDezhV3yYEIzJFI
CmGzoQRjE88W6gz1rE6lNtdRa/:19236:0:99999:7:::
[root@localhost ~]#
```

Brian's account is locked.

lock Brian's password by adding a !! or * in front of the password: **vipw -s**

```
brian:!$6$Li..
```

Adding ! in front of the password to unlock the account.

Modify Brian's shell to a **/sbin/nologin** shell

```
stewie:x:1001:1001:Stewie Griffin:/home/stewie:/bin/bash
brian:x:200:100:Brian Griffin:/home/brian:/sbin/nologin
```

Shell is **changed** to a **/sbin/nologin**.

Task 4: Creating a User Without Using Command-Line Tools

Create a fourth user with username lois. Lois's full name is "Lois Griffin", and she should be allocated the next uid and gid in sequence, as Linux would have done. Lois prefers the bash shell.

chown the skeleton files and home directory.

```
brian:x:200:100:Brian Griffin:/home/brian:/bin/bash
lois:x:1003:1003:Lois Griffin:/home/lois:/bin/bash
```

4th user with username lois is **created**.

```
lois:x:1003:
```

Lois group file is **set up**.

```
[root@localhost ~]# passwd lois
Changing password for user lois.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[root@localhost ~]#
```

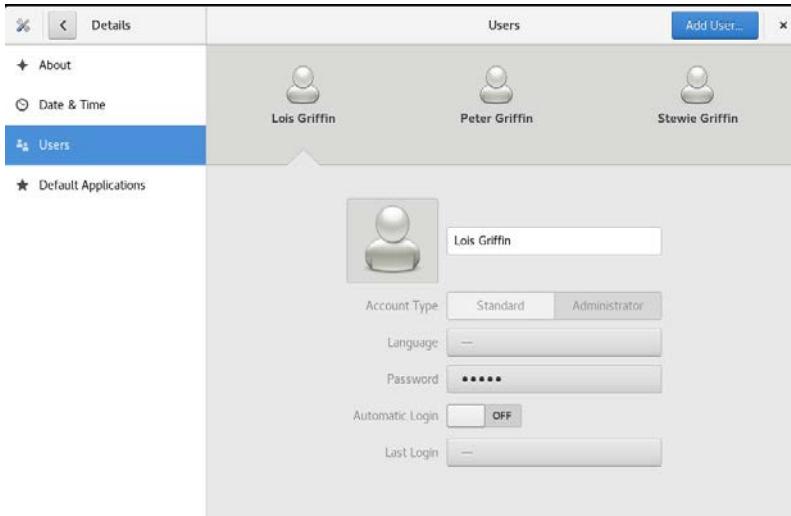
Password is **created** for lois.

```
[root@localhost ~]# cp -av /etc/skel /home/lois
'/etc/skel' -> '/home/lois'
'/etc/skel/.bashrc' -> '/home/lois/.bashrc'
'/etc/skel/README' -> '/home/lois/README'
'/etc/skel/.bash_logout' -> '/home/lois/.bash_logout'
'/etc/skel/.bash_profile' -> '/home/lois/.bash_profile'
'/etc/skel/.zshrc' -> '/home/lois/.zshrc'
'/etc/skel/.mozilla' -> '/home/lois/.mozilla'
'/etc/skel/.mozilla/plugins' -> '/home/lois/.mozilla/plugins'
'/etc/skel/.mozilla/extensions' -> '/home/lois/.mozilla/extensions'
[root@localhost ~]# chown -R lois:lois /home/lois
[root@localhost ~]# chmod 700 /home/lois
[root@localhost ~]# ll -d /home/lois
drwx----- 3 lois lois 106 Sep 1 11:03 /home/lois
[root@localhost ~]#
```

Home directory (with skeleton files) is **created**, and chown her home directory and skeleton files.

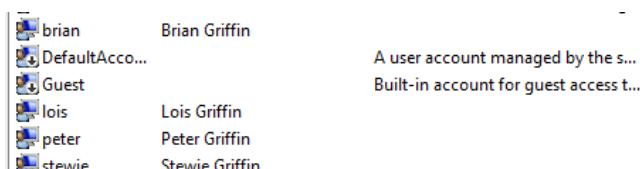
Task 5: Using the GUI

Verify the configuration of lois though the GUI via Settings → Details → Users.

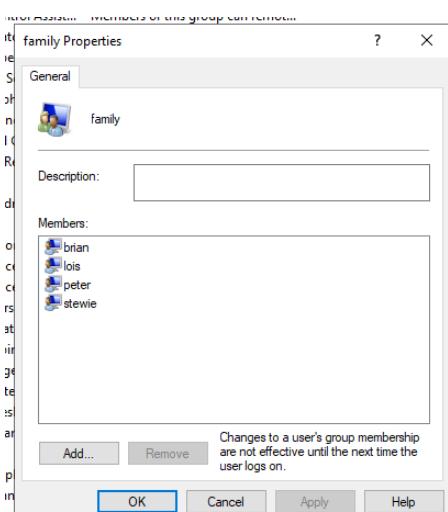


Task 6: Using Windows Server GUI to add users and groups

Start with Server Manager → Local Server → “Computer Management” → Local Users and Groups → Users → New User. Add the created users on Linux then add and assign them as members of the group.



Users is created on windows.



Member is assigned on each group.

Task 7: Using Windows Server command line to add users and groups

Some PowerShell commands:

New-LocalUser -Name "joe" -FullName "Joe Swanson"

```
PS C:\Users\Administrator> New-LocalUser -Name "joe" -FullName "Joe Swanson" -NoPassword
Name Enabled Description
---- ----- -----
joe True
```

Get-LocalUser

```
PS C:\Users\Administrator> Get-LocalUser
Name Enabled Description
---- ----- -----
Administrator True Built-in account for administering the computer/domain
brian True
DefaultAccount False A user account managed by the system.
Guest False Built-in account for guest access to the computer/domain
joe True
lois True
peter True
stewie True
WDAGUtilityAccount False A user account managed and used by the system for Windows Defender Application Guard
```

Disable-LocalUser joe

```
PS C:\Users\Administrator> Disable-LocalUser joe
PS C:\Users\Administrator> Get-LocalUser
Name Enabled Description
---- ----- -----
Administrator True Built-in account for administering the computer/domain
brian True
DefaultAccount False A user account managed by the system.
Guest False Built-in account for guest access to the computer/domain
joe False
```

Enable-LocalUser joe

```
PS C:\Users\Administrator> Enable-LocalUser joe
PS C:\Users\Administrator> Get-LocalUser
Name Enabled Description
---- ----- -----
Administrator True Built-in account for administering the computer/domain
brian True
DefaultAccount False A user account managed by the system.
Guest False Built-in account for guest access to the computer/domain
joe True
```

net user bonnie /add /fullname: "Bonnie Swanson"

```
C:\Users\Administrator>net user bonnie /add /fullname:"Bonnie Swanson"
The command completed successfully.
```

net user

```
C:\Users\Administrator>net user
User accounts for \\SOPUTERANTO

-----
Administrator bonnie brian
DefaultAccount Guest joe
lois peter stewie
WDAGUtilityAccount

The command completed successfully.
```

net user bonnie

```
C:\Users\Administrator>net user bonnie
User name bonnie
Full Name Bonnie Swanson
Comment
User's comment
Country/region code 000 (System Default)
Account active Yes
Account expires Never
```

wmic useraccount where "name='bonnie'"

```
C:\Users\Administrator>wmic useraccount where "name='bonnie'"
AccountType Caption Description Disabled Domain FullName InstallDate LocalAccount Loc
kout Name PasswordChangeable PasswordExpires PasswordRequired SID
SIDType Status
512 SOPUTERANTO\bonnie FALSE SOPUTERANTO Bonnie Swanson TRUE FAL
SE bonnie TRUE TRUE TRUE S-1-5-21-1116545310-2023033734-25960337-1015
1 OK
```

Lab 5b Superuser Powers

Task 1: Executing Root Commands Safely via Sudo in Linux

From root, execute the command *vi sudo (/etc/sudoers file)*

What commented-out user permissions are there?

```
## Allow root to run any commands anywhere
root    ALL=(ALL)      ALL

peter  ALL=/sbin/ifconfig
```

Granting 'peter' user to allow root.

```
[root@localhost ~]# su - peter
[peter@localhost]~% id
uid=1000(peter) gid=1000(peter) groups=1000(peter)
[peter@localhost]~% sudo ifconfig ens33 down

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

 #1) Respect the privacy of others.
 #2) Think before you type.
 #3) With great power comes great responsibility.

[sudo] password for peter:
```

Disconnecting ens33 by user 'peter'.

If an administrator wanted to grant all users the ability to disconnect or reconnect the ens33 network port, how should this be written to visudo?

```
## Allow root to run any commands anywhere
root    ALL=(ALL)      ALL

peter  ALL=/sbin/ifconfig
%family ALL=/sbin/ifconfig
```

To give grant all users the ability to change the ens33 network, is by adding the family group to visudo configuration.

Try run: *sudo su -*

What does it do? Is this a good thing?

```
U [lois@localhost ~]$ sudo su -
[root@localhost ~]#
+-----+
+ They must know the root pass +
```

Sudo su command used to **switch to root**. This could be good or even bad command depends on the wanted setting. In some case it could be good since not all the user could run the 'sudo su-' command, by doing this, it could give footprint to which user is changing the configuration on the root.

Task 2: Executing Admin Commands in Windows Server

Logged in as a normal user then execute cmd as the Administrator: *runas /user:Administrator cmd.exe*

```
LAWRENCE
Administrator: Command Prompt
Microsoft Windows [Version 10.0.17763.1369]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>runas /user:Administrator cmd.exe
Enter the password for Administrator:
Attempting to start cmd.exe as user "SOPUTERANTO\Administrator" ...

Administrator: cmd.exe (running as SOPUTERANTO\Administrator)
Microsoft Windows [Version 10.0.17763.1369]
(c) 2018 Microsoft Corporation. All rights reserved.
```

Trying to run **CMD as administrator** using command line.

Task 3: Notifying users on system issues or computer usage policies (Linux – the old way)

Put in a message on `/etc/motd` (message-of-the-day). **When is this message displayed?**

`ssh your_username@localhost`

Is the `/etc/motd` file displayed for ssh sessions?

```
[root@localhost ~]# ssh peter@localhost
peter@localhost's password:
This is MOTD
!!!!!!!
D:
Activate the web console with: systemctl enable --now cockpit.socket
```

The MOTD message is **displayed** as soon as **logged in** to the user.

Press Ctrl-Alt-F3 and log in using the text login approach. **Is `/etc/motd` displayed? When?**

```
CentOS Stream 8
Kernel 4.18.0-408.el8.x86_64 on an x86_64

Activate the web console with: systemctl enable --now cockpit.socket

localhost login: peter
Password:
Last login: Fri Sep  2 22:18:55 from ::1
This is MOTD
!!!!!!!
n:
```

The MOTD message is **displayed** as soon as **logged in** to the user.

Go to graphical screen by press Ctrl-Alt-F1.

Add welcome message inside file `/etc/issue`; `/etc/motd`; and `/etc/issue.net`!

Use the **banner** option in `/etc/ssh/sshd_config` to show `/etc/issue.net`

```
# no default banner path
#Banner none
Banner /etc/issue
Banner /etc/issue.net
```

Adding the `/etc/issue` to the banner so it will appear at the login prompt.

```
[root@localhost ~]# ssh peter@localhost
\S
Kernel \r on an \m
WELCOME!!!
WE MISS U <3
peter@localhost's password:
This is MOTD
!!!!!!!
.D:
Activate the web console with: systemctl enable --now cockpit.socket
```

The **banner** of `/etc/issue` **appeared** at the login prompt.

However, the banner only **appeared** on the **Graphical login**.

Task 4: Notifying Users on System Issues (Linux – the Graphical Way)

Create a new file `/etc/dconf/profile/gdm` with the contents:

```
user-db:user
system-db:gdm
file-db:/usr/share/gdm/greeter-dconf-defaults
```

```
File Edit View Search Terminal Help
user-db:user
system-db:gdm
file-db:/usr/share/gdm/greeter-dconf-defaults
```

Create second new file `/etc/dconf/db/gdm.d/01-banner-message` with the contents:

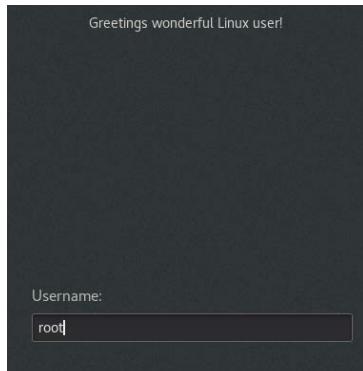
```
[org/gnome/login-screen]
banner-message-enable=true
banner-message-text='Greetings wonderful Linux user!'
```

```
[org/gnome/login-screen]
banner-message-enable=true
banner-message-text='Greetings wonderful Linux user!'
```

Update the dconf database by: `dconf update`

```
[root@localhost ~]# vim /etc/dconf/db/gdm.d/01-banner-message
[root@localhost ~]# dconf update
[root@localhost ~]#
```

Use `systemctl restart gdm` to restart the gdm service

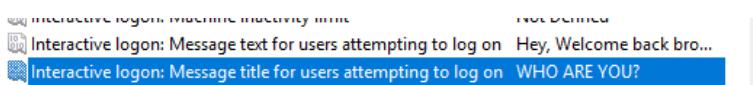


The message **appears** in the **login prompt**.

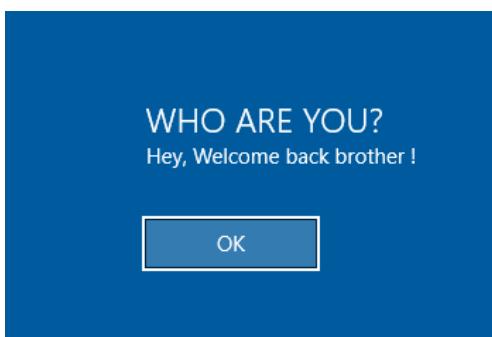
Task 5: Notifying Users on System Issues (Windows Server)

Add a message-of-the-day by:

1. Go to Server Manager → Tools → Local Security Policy → Security Options → Interactive logon
2. Edit these two entries:
 - Interactive logon: Message text for users attempting to log on
 - Interactive logon: Message title for users attempting to log on



3. Update system policies by running: `gpupdate /force`
4. Log out and log back in again.



The message **appears** **before** the **login prompt**.

Reflection:

On the first weeks of learning, the materials is still basic and not difficult at all. Especially for the lab1a which only repeating all the knowledge from Web Server Subject. It works as well for the lab b and c which are not very technical but still can learn many things such as the target mode and how to start and enable a system. First week went very smooth and has no struggle on it.

Going to the second week, the materials is reasonable but it contains a lot from lab2 a to d. I actually got my struggle at very first command which is -cqs command. I read this command and try several times until I find out that the pdf type wrong command, which should be -cqs while it typed as -cgs. I solved this by searching in google about the command. The rest of the materials went well since I already know most of it about the documentation and system. However, I still get new things about the Nice Value. It is the new things I learn from Week 2.

The third week I learnt to statically set the network on both Linux and windows. It's a bit familiar for me looking for the command for both OS, however learning this static network is not hard but a bit tricky. I got several times error while running the `nmcli up ens eth`. This issue I got since I put the wrong message on the `ifcfg-ens`. While for the windows, I got no issue on it.

Afterwards, I learn about the Dynamic Network on Week 4. This week is quite similar to the third week. By doing DHCP Network, I could understand better about the last week material about the static network. It really challenges me on reading the log files, however, I could manage to read it after watch the recording video that give me more insight.

On week 5, the materials are about users and grouping. This is actually not a new thing for me, however, I never try to create a new user or new group, so this week is a good chance for me to learn deeper about it. Creating and assigning user to the group is okay but it is not that simple. The other new thing I love to learn is about how to lock the user on Linux and how I can add banner message on both Linux and Windows prompt.

Reference List

- Bosko Marijan 2022, *How to Use the dmesg Linux Command*, viewed 1 September 2022, <<https://www.digitalocean.com/community/tutorials/workflow-command-line-basics-shutdown-reboot>>
- Joshtronic 2019, *Command-line Basics: Shutdown and Reboot*, viewed 30 August 2022, <<https://www.digitalocean.com/community/tutorials/workflow-command-line-basics-shutdown-reboot>>
- Karim Buzdar 2022, *Linux Command ps –ef*, viewed 2 September 2022, <<https://linuxhint.com/ps-minus-ef/>>
- LDP Author 2005, *How Linux Keeps Track of Time*, viewed 25 August 2022, <<https://tldp.org/HOWTO/Clock-2.html>>
- LDP Author 2005, *Linux Documentation Project Works*, viewed 25 August 2022, <<https://tldp.org/docs.html#faq>>
- LDP Author 2005, *The Linux System Administrator's Guide*, viewed 25 August 2022, <<https://tldp.org/LDP/sag/html/index.html>>