Network Operating Systems and Services

Computer Systems (CS2000)

Trimester 2 2020

Important Information

- Practical Assessment (Linux, Windows, Scripting) 80%
  - Lab Assessment 1 (Linux) 30%
  - Lab Assessment 2 (Windows) 20%
  - Bash Assessment 30%
- Assignment 20%

The Network Operating System

- The complexity involved in the control of communications on a computer network should be obvious.
  - Differing protocols
  - Various operating systems
  - Hardware
  - and many other factors
- What is the means of accomplishing this task?
  - The network operating system (NOS).

NOS

- What defines a NOS?
  - *Distinction between a stand-alone OS and NOS is not always obvious*
  - *NOS may provide:*
    - Printer sharing
    - File sharing
    - Database sharing
    - Application sharing
    - Network Security
    - Data backup and replication services
    - Remote access services
    - Fault tolerance
- *Stability!!*

Example NOSs (NOSes??)

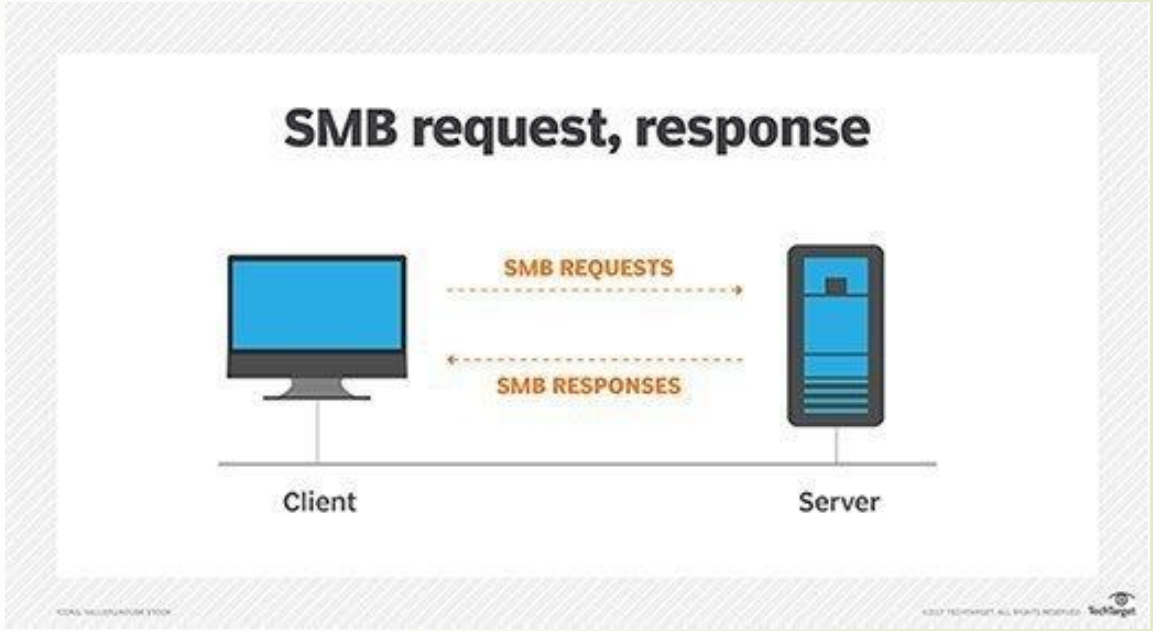
- Novell Open Enterprise Server
  - *Netware – now Suse Linux based*
- Linux-based – Fedora, Debian, Ubuntu ….
- Mac OSX server/Windows Server
  - *Some multipurpose operating systems such as OSX, Windows Server 2003/2008/2012/2016/2019 come with capabilities that enable them to be classified as a NOS.*

The NOS

- Client-Server Network Operating Systems
  - *On a client-server network*
    - The network operating system is installed and runs on a computer called the network server.
    - The server must be a specific type of computer.
    - A client-server operating system is responsible for coordinating the use of all resources and services available from the server

The NOS (cont.)

- The client part is any other network device or process that makes requests to use server resources and services.
  - *Network users at workstations request the use of services and resources through client software and talk to the operating system in the server by means of a common protocol.*



The NOS (cont.)

- *Users "log in" to the network server from the workstation*
- *A user enters a login command and gives his or her username and password.*
- *If the username and password are valid, the server "authenticates" the user and allows him or her access to all network services and resources to which he or she has been granted rights.*
- *As long as the user has correct network rights, the client-server operating system provides the services or resources requested by the distributed applications running in workstations.*

The NOS (cont.)

- The operating system manages various server resources:
  - *Hardware such as hard disks, RAM, printers, modems etc.*
  - *The network file system is also a server resource.*
  - *Coordinating file access and file sharing (including file and record locking)*
  - *Managing server memory*
  - *Managing data security*
  - *Scheduling tasks for processing*
  - *Coordinating printer access*
  - *Managing internetwork communications.*
  - *Among the most important functions performed by a client-server operating system are ensuring the reliability of data stored on the server and managing server security.*

The NOS (cont.)

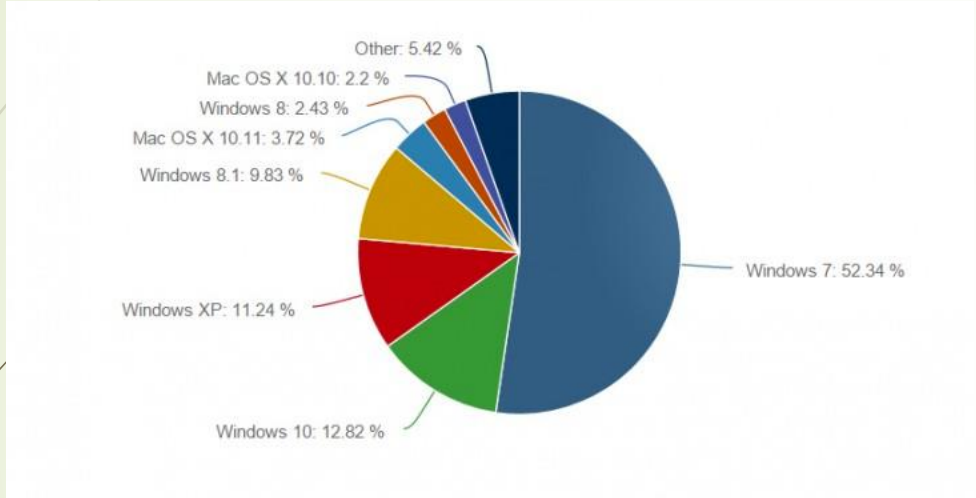
- Different NOSs provide varying levels of services.
- It is critical that the correct NOS is chosen for the network requirements.
  - *Novell : Excels at File and print serving and very large networks.*
    - Note that Netware is the NOS, NDS is a different animal.
  - *UNIX/Linux : Efficient, suited to Internet based networks, computationally intensive small workgroups.*
  - *Windows Server : Smaller Domains, relatively simple administration.*
    - UNIX/Linux (natively) lacks this functionality completely.
    - LDAP can overcome some shortfalls
  - *NDS is available for Windows and most UNIX varieties.*

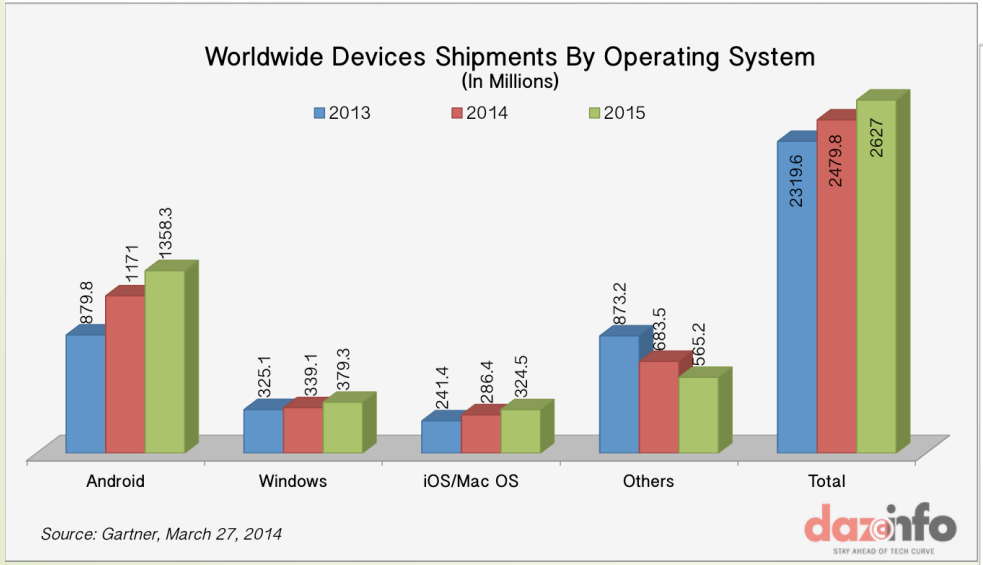
Desktop Operating Systems

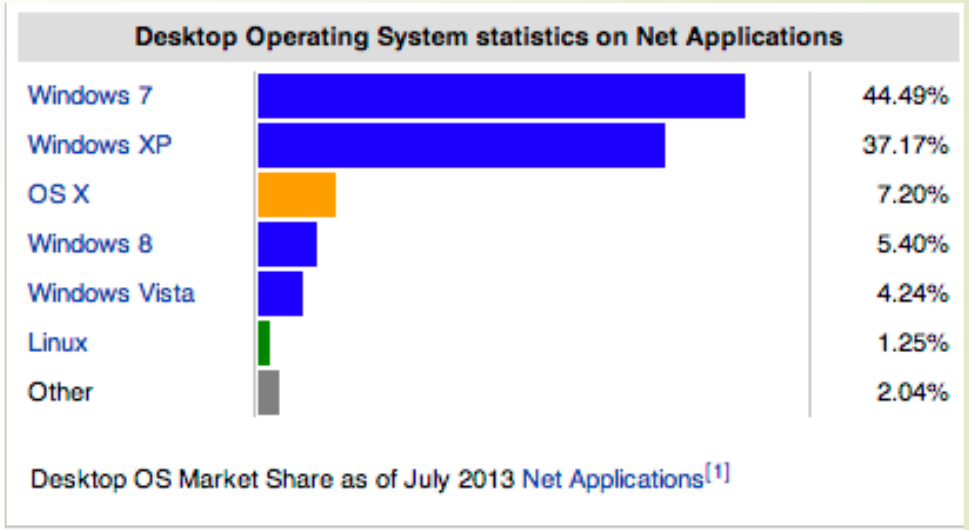
- Each workstation on the network must have software that manages its own resources.
  - *Known as the desktop operating system*
- Enables a workstation to:
  - *Access files from its own local hard disks*
  - *Displaying information on its video display*
  - *Coordinating local printing*
- Commonly used desktop operating systems
  - *Windows XP/7/8/10, UNIX, Linux, Apple OSX.*
  - *Each of the different desktop operating systems has advantages and disadvantages.*

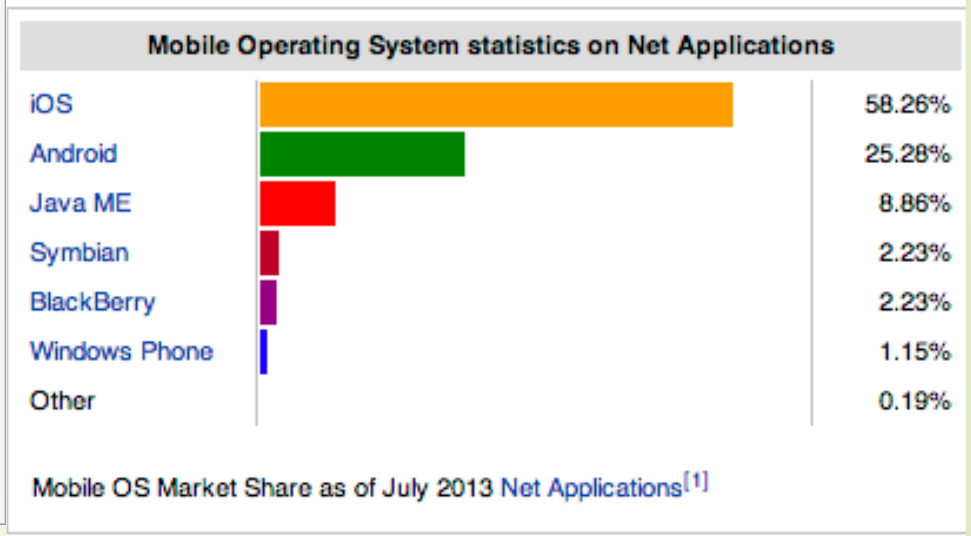
OS'es don't go away

- **16.9% of PCs still run XP (2015)**

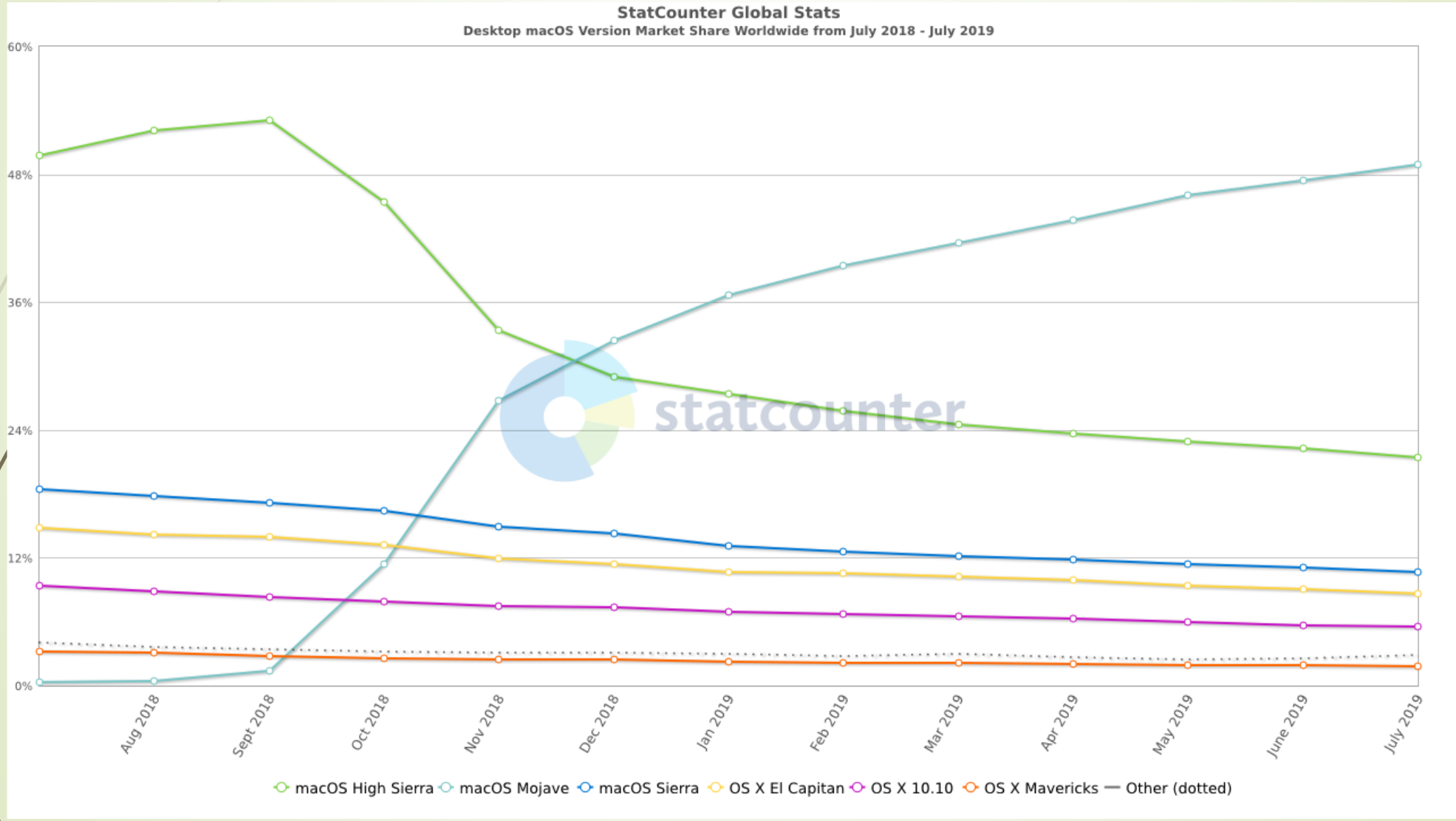








Some a little different



Desktop Operating Systems

- *Historically, they are not compatible with each other.*
- *Linux and OSX have changed this in recent years*
- This brings us to another important function of a network operating system
  - *it should be able to interconnect all the commonly used desktop operating systems to ensure that all network users have access to the computer that they are most familiar with and that is best suited to the job they need to do.*

Novell

- Software elements of Novell
  - *The NOS (Netware/OES)*
    - Runs the server.
  - *Client software*
    - Runs on each workstation
  - *Novell Directory Services (NDS) – now called eDirectory*
    - Tracks all network entities.
  - *Service software*
    - Provides services to the network e.g. printer queues.

Novell (cont.)

- *NetWare utilities*
  - Allows users and administrators to work with different aspects of the network.
- NOS replaces the regular operating system.
  - *Replaces DOS (MS-DOS, PCDOS etc.) once it loads i.e. uses DOS to boot into NetWare.*
  - *Netware 6.x comes with caldera's PC-DOS.*

Reliability

Server Operating System

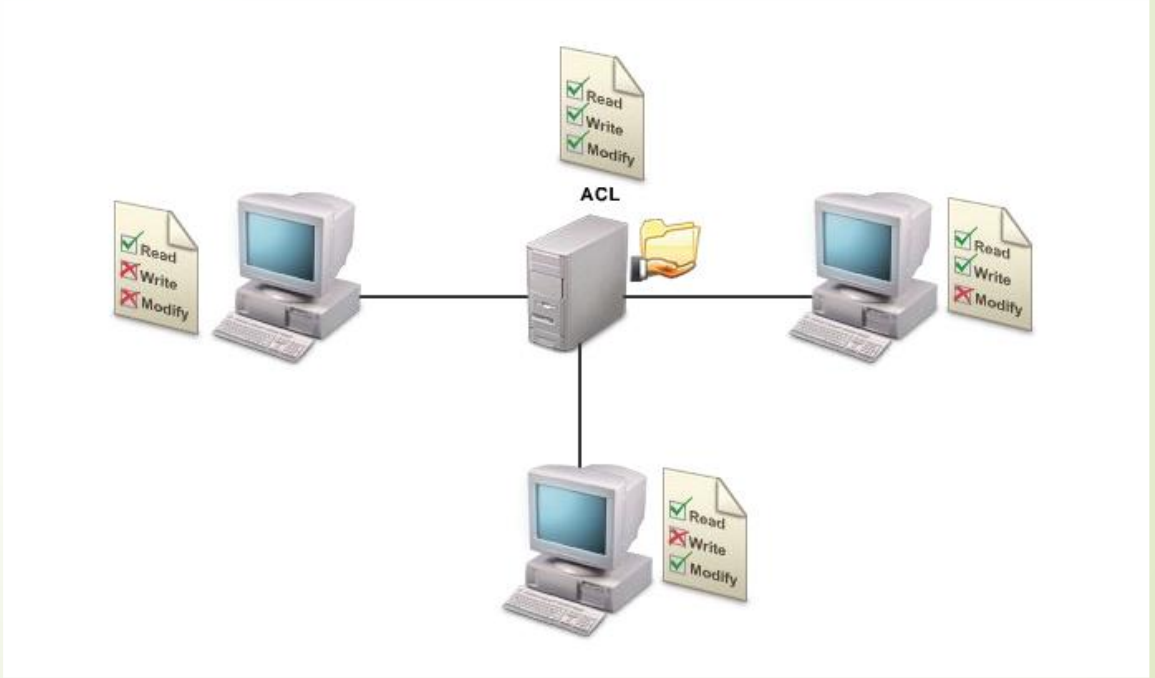
- The NOS manages all communications over the network.
  - *Controls data transfer*
  - *How files are stored*
  - *How print jobs are handled*
- Manages security
  - *Authentication*
  - *File rights*
  - *Resources*

NOS Management Features

- Adding & removing users from the network
- Setting individual & group level permissions
- User policies
  - *Try Poledit on lab VMs (Gpedit.msc in Win 7/8/10)*
- Maintaining login scripts for resource tracking/utilisation

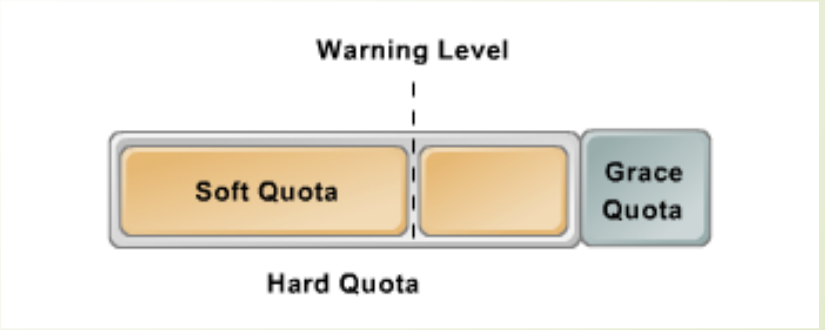
NOS Management Features (cont.)

- Access Control Lists
  - *Very different to ACLs in routers*
  - *Very complex on Windows based servers (user, group, file and directory level often conflict)*



NOS Management Features (cont.)

- Disk quota
  - *Limit the utilization of shared disk space by a user.*
  - *Two basic types of disk quotas*
    - Usage quota
    - File quota
    - Administrators usually define a warning level, or soft quota
    - Also be a small amount of grace quota



NOS Management Features (cont.)

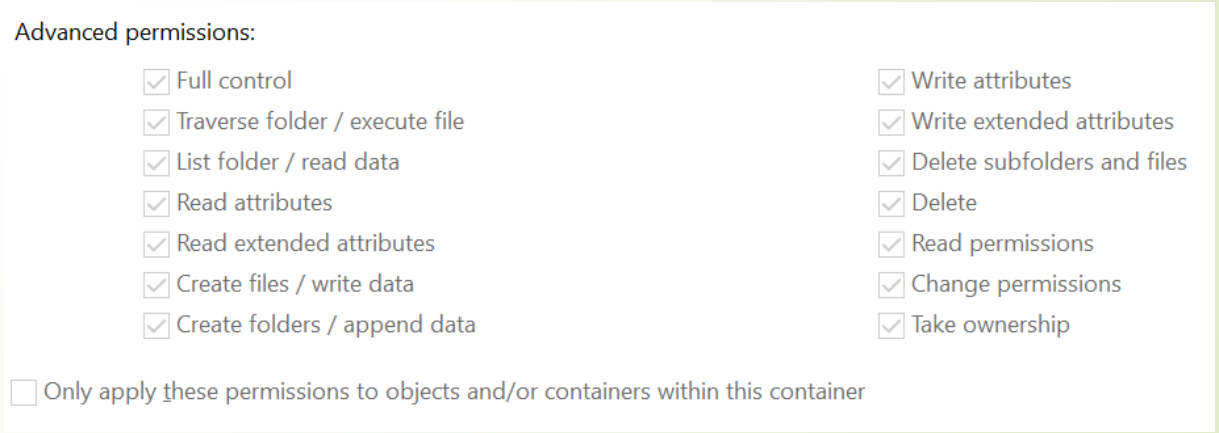
- Virtualisation
  - *Space consolidation.*
  - *It is comparatively easy and reduces the cost of installation and maintenance.*
  - *It reduces the power required for processing data and cooling devices.*
  - *Newer technologies in server virtualisation allow virtual machines to automatically restart on different host servers in case of any failure in the main server.*

File Systems

- Windows
  - *FAT**
    - File Allocation Table
    - Number indicates size of cluster table
      - *FAT12 => 32MB volume max (DOS3)-1.5 bytes per cluster*
      - *FAT16 => 2GB (DOS6 the last version)- 2 bytes per cluster*
      - *FAT32 => 2TB (Win95B)-4 bytes per cluster*
      - *Large disk=large clusters=waste*
    - No permissions or ACLs

File Systems (cont.)

- Windows
  - NTFS
    - Permissions for folders are different to file
    - Share permissions are different again



File Systems (cont.)

- Linux
  - UFS
  - ext*, and many, many others
  - mkfs
  - fsck (journaling on ext3, reiser, etc.)
- *Check http://en.wikipedia.org/wiki/Comparison_of_file_systems*
  - *Good article on file systems*
  - (with standard Wikipedia caveat)

File Systems (cont.)

- Each hard disk can contain up to four different "true“ partitions, which are called primary partitions.
- One of the four partitions may be designated as an extended partition.
- The extended partition may then be subdivided into multiple logical partitions.
- There is no limit on the number of logical volumes other than the fact that there are only 26 letters in the alphabet and A, B, and C have been taken.

Devices

- Windows
  - Detects disk devices
  - Automatically attaches as drive letter
- Linux
  - Can auto-detect some, but PNP support not complete
  - Usually need to mount (and umount – note the missing 'n')

File System Permissions

- Windows
  - GUI
  - cacls – change acls
    - Now deprecated and replaced with icacls
- Linux
  - Chmod – change mode (-rwxrw-r--=764)
  - Chown – change owner (owner:group)
  - Chgrp – change group
  - Umask - sets the mask applied to newly created file and directory permissions (0666-0002=0664 / 0777-0002=0775)
  - GUI (for some settings)

Linux File Permissions

- Each file and directory has 3 user-based permission groups:
  - Owner - apply only the owner of the file or directory
  - Group - apply only to the group that has been assigned to the file or directory, they will not effect the actions of other users.
  - All users - applies to all other users on the system, this is the permission group that you want to watch the most.

Permission Types

- Each file or directory has 3 basic permission types:
  - Read - The Read permission refers to a user's capability to read the contents of the file.
  - Write - The Write permissions refer to a user's capability to write or modify (inc. delete) a file or directory.
  - Execute - The Execute permission affects a user's capability to execute a file or view the contents of a directory.

Explicitly Defining Permissions

- To explicitly define permissions, reference the Permission Group and Permission Types.
- The Permission Groups used are:
  - *u - Owner*
  - *g - Group*
  - *o or a - All Users*
- The potential Assignment Operators are + and - The Permission
  - *r - Read = 4*
  - *w - Write= 2*
  - *x - Execute= 1*
- To represent rwx triplet use 4+2+1=7
- To represent rw- triplet use 4+2+0=6
- To represent r– triplet use 4+0+0=4
- To represent r-x triplet use 4+0+1=5

File System Permissions (con.t)

- Special Linux Modes - sticky bit, SUID and SGID.
  - *1000: If the sticky bit is set on a directory, then only the file owner, the directory owner, or superuser can delete a file in that directory.*
    - If a directory has permissions 0770, then the directory owner or anyone in the directory's group can add files or delete any files (regardless of who the file's owner is).
    - If the sticky bit is set, so the permissions are 1770, then anyone in the group can add files to the directory, but each user can only delete his or her own files. (chmod +t - drwxrwxrwt)

File System Permissions (cont.)

- *2000: set group ID: Executables with this will run with effective gid set as the gid of the file owner. When a directory has this permission, files created in the directory have the group ID of the directory, rather than the default group setting for the user who created the file. (chmod g=s -rwx--Sr-x)*
- *4000: set user ID: Executables with this will run with effective uid set as the uid of the file owner. Directories with set-user-id bit force all files/folders created in them to be owned by the directory owner and not by the uid of the creating process. (chmod u=s d--Sr-xr-x)*
  - NOTE – Not every *NIX implementation supports all functions of these permissions

File System Permission (cont.)

- Linux Access Control Lists
  - *Give stu1 and the group SAstu rwx on SAdir. Using the setfacl utility (note – file system must be mounted with ACL support)*
    - setfacl -m user:stu1:rwx,group:SAstu:rwx SAdir
    - getfacl should return the following:
      - *# file: SAdir*
      - *# owner: foo*
      - *# group: bar*
      - *user::rwx*
      - *user:stu1:rwx*
      - *group::r-x*
      - *group:SAstu:rwx*
      - *mask::rwx*
      - *other::---*

Thinking About Permissions

- Explain the simplest way to change permissions on myfile.txt so that Student1 has access to the file, but Student2 does not.

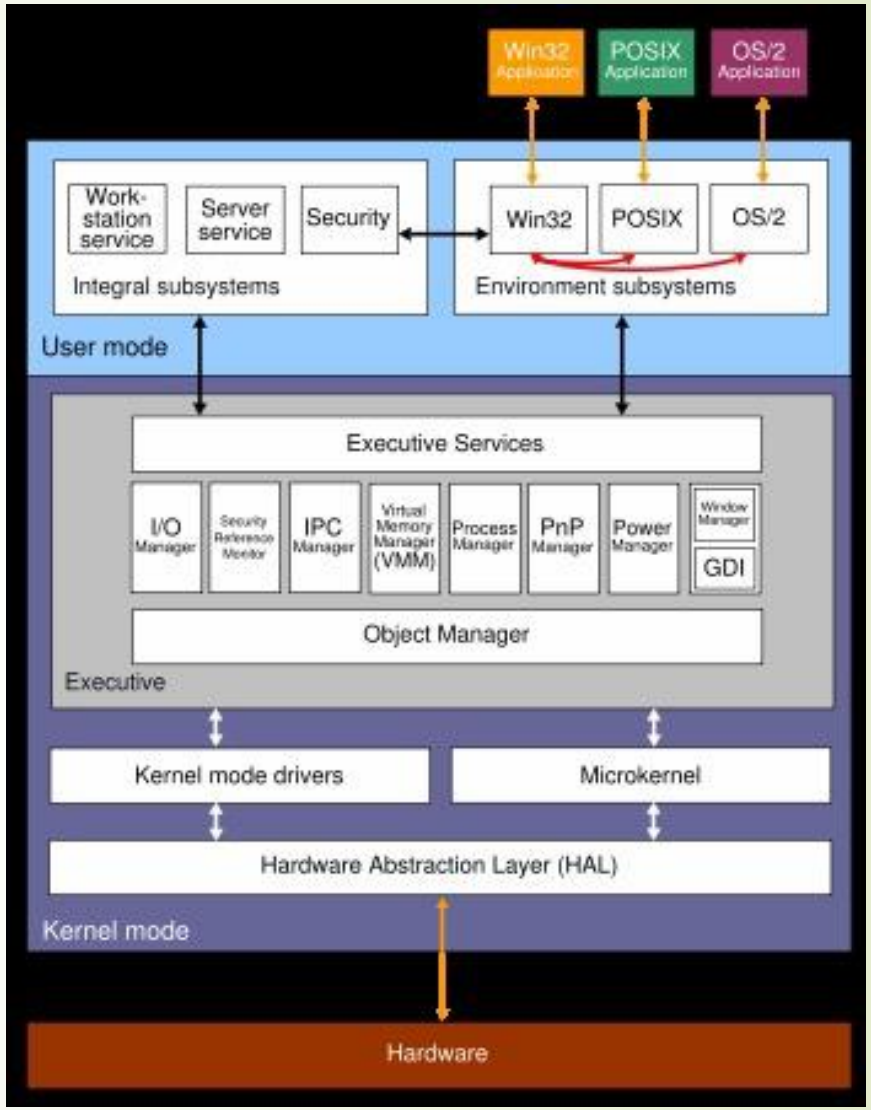
- Do you need to "give" Student1 access?
- Do you need to "deny" Student2 access?

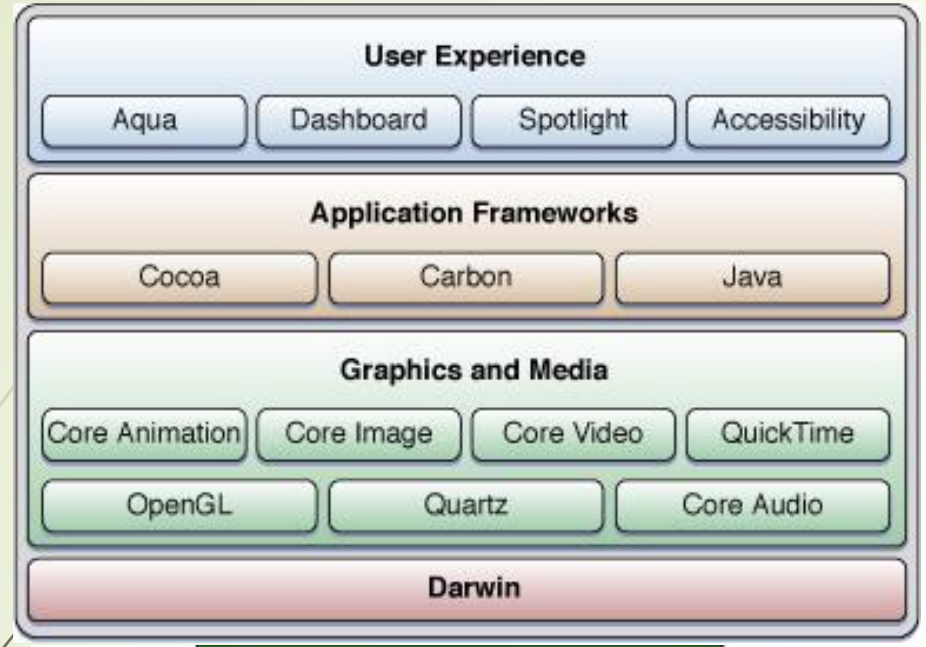
Review of Linux Runlevels

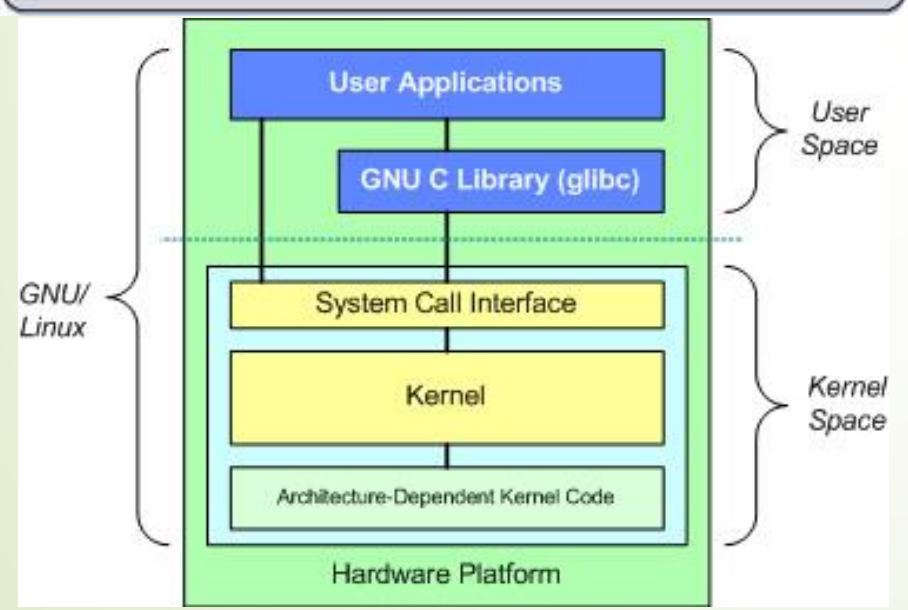
- Runlevels
  - 0 – Shutdown
  - 1 – Single user
  - 2 – Multi-user w/o networking
  - 3 – Multi-user w/networking
  - 4 – Unused
  - 5 – Multi-user w/networking and GUI
  - 6 – Reboot

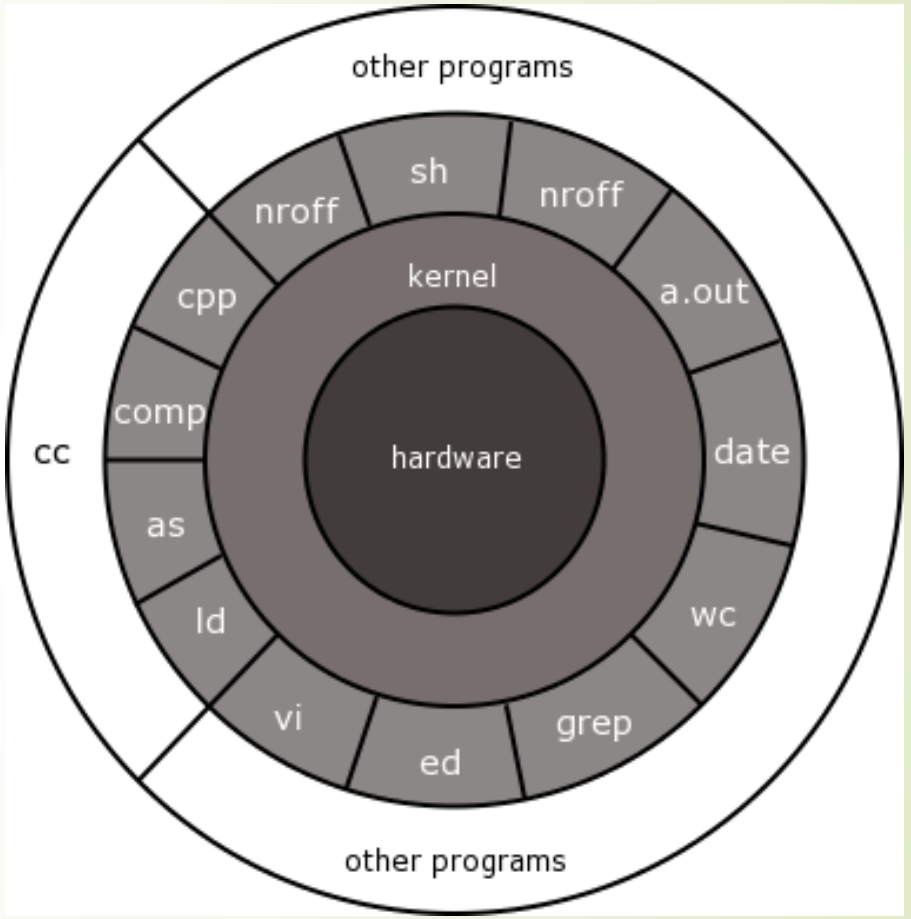
System Components

- (Win32)




User Experience
Aqua
Dashboard
Spotlight
Accessibility
Application Frameworks
Cocoa
Carbon
Java
Graphics and Media
Core Animation
Core Image
Core Video
QuickTime
OpenGL
Quartz
Core Audio
Darwin


User Applications
GNU C Library (glibc)
System Call Interface
Kernel
Architecture-Dependent Kernel Code
Hardware Platform
User Space
Kernel Space
GNU/ Linux


other programs
sh
nroff
nroff
kernel
cpp
a.out
comp
hardware
cc
date
as
wc
ld
vi
ed
grep
other programs

System Components (cont.)

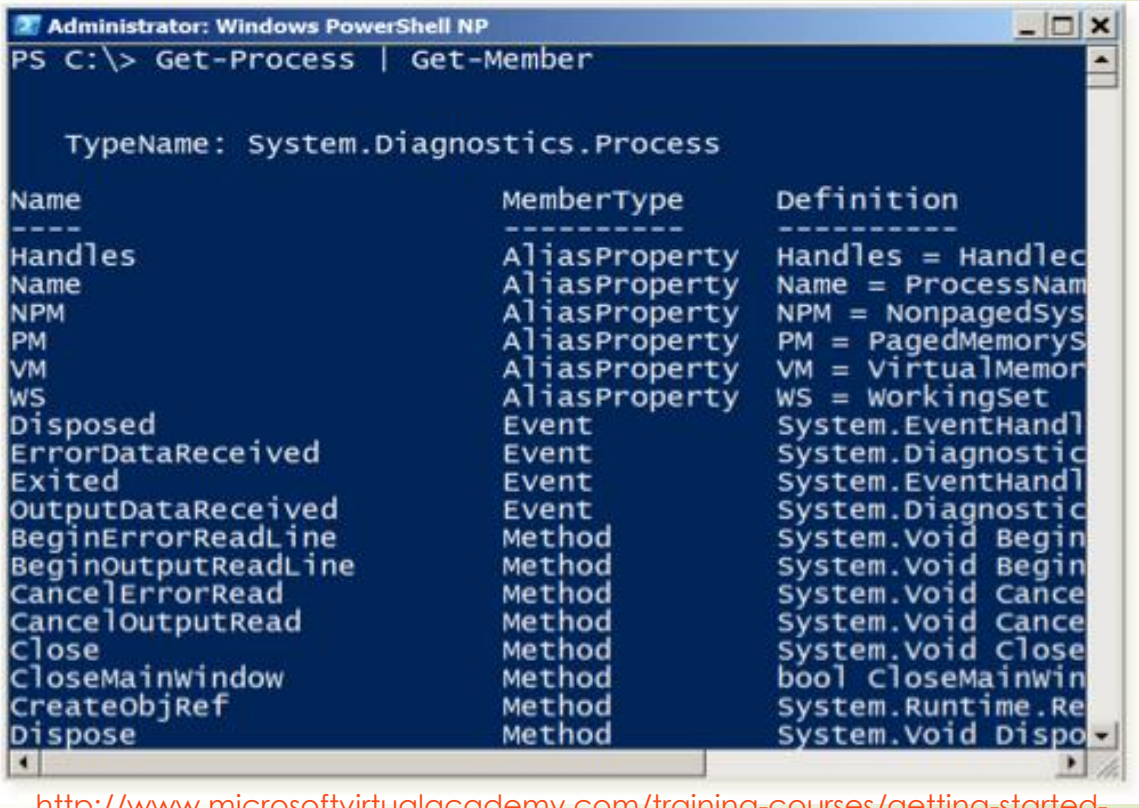
- *Layered OS*
- *Protection*
- *Kernel*
- *Privileged accounts*
  - Administrator
  - Root (UID=0)
- *Devices -> partitions -> filesystems (disk -> partition -> file system - > files; fdisk)*
- *Processes (more later)*
- *Startup/shutdown (more later)*

Processes

- Windows User Mode processes
  - *Use individual memory spaces*
  - *Runs "as" a particular user*
  - *Three types*
    - System processes – Manage User Mode environment (winlogon)
    - Windows Services (services)
    - User applications

Processes (cont.)

- Windows Kernel Mode processes
  - *Share memory space*
  - *Have direct access to hardware*
  - *Includes Executive Services, Microkernel, HAL*
- Viewing Processes
  - *Task Manager*
- Stopping Processes
  - *Services Utility*
  - *Task Manager*
  - *Net commands*



http://www.microsoftvirtualacademy.com/training-courses/getting-started-with-powershell-3-0-jump-start#fbid=6P5y_spy8gp

Processes (cont.)

- Linux doesn't really make a distinction between types of processes
  - *su command allows for user switching*
- Viewing Processes
  - *ps command*
- Stopping Processes
  - *Services Utility*
  - *Kill command*

System Startup

- Normal startup sequence: POST to MBR to program
  - *Windows startup sequence:*
    - Boot phase - NTLDR (boot.ini –See ITE1)
    - Kernel phase (ntoskernel.exe), kernel initialisation (drivers)
    - Services phase (smss.exe),
    - Logon phase (winlogon.exe and lsass.exe)

System Startup (cont.)

- *Linux startup sequence:*
  - lilo/grub
  - kernel
  - init(/etc/inittab) – determines runlevel
  - /etc/rc*
    - *Scripts used to control how the system will startup/shutdown are /etc/inittab and /etc/rc.d (or /etc/rcX.d, where X corresponds to runlevel. "S" scripts designate items to run at startup.)*
- Multi-booting: NTLDR vs Lilo/Grub vs VM
  - *Think how you would decide on your own machine if you required several operating systems*

System Startup (cont.)

- Advanced startup
  - *Windows advanced startup options (Startup Settings Win8 are different)*
    - Last known good
    - System Restore
    - Safe mode
    - Recovery console
    - Emergency repair disk (ERD)
  - *Linux advanced startup options*
    - boot from floppy
    - single-user mode

http://windows.microsoft.com/en-AU/windows-8/windows-startup-settings-safe-mode

System Shutdown

- Why shutdown cleanly?
- Windows (get used to doing this)
  - *Ctrl-Alt-Del – Shutdown*
  - *Start – Shutdown*
  - *From command line - shutdown –s (or –r)*
  - *Applications are closed/Services stopped*
- Linux
  - *Init 0 (init 6 will reboot)*
  - *shutdown -h now (-r will reboot)*
  - */etc/rc* handles some process shutdown, using "K" scripts*

Configuration

- Server is of little use until configured
- Each NOS will have a unique installation process
  - *But there are commonalities*
- Network Planning
  - *WINS or DNS*
  - *IP Scheme/IPv6*
  - *Services to be offered*
  - *Locate device drivers and service packs/patches*

Peripheral Access

- Each NOS will have unique methods e.g. print
  - *Novell – NDPS*
  - *Window – Print services*
  - **nix/OSX – CUPS*
  - *Many other methods such as SMB (Samba)*
- Set Access rights
- Shares
- Permissions

Server Utilities

- Most NOSs ship with several utilities for managing the server and the network.
  - *Backup software*
  - *System monitors*
  - *Event or error logs.*
  - *Hardware management layer that enables users to monitor and troubleshoot hardware problems.*
- Management protocols
  - *SNMP (Simple Management Protocol)*
  - *DMI (Desktop Management Interface)*
  - *RMON (Remote Monitoring)*
  - *Separate purchase of a management console application*

Server Logs

- Event Types
  - *Informational*
  - *Warning*
    - Not necessarily significant but may indicate future problems e.g. low disk space
- *Error*
  - Events of the highest significance e.g. service didn't load at boot
- *Success*
  - e.g. user successfully logged in
- *Failure*
  - Audited security access attempts that fail. eg user types incorrect password

Documentation

- This is **IMPORTANT!**
  - *High priority server documentation includes*
    - Server name and location
    - Brand, model, hardware configuration and serial number
    - NOS type, version and serial number/license codes
    - Network config information
    - *IP, DNS, subnet masks, protocols etc*
  - *Low Priority server documentation includes*
    - Warranty information and agreement expirations
    - Asset management documentation Inc. service tags
    - Technical support contact details

Server Log Books

- Logbooks should contain
  - *Configuration information*
  - *A complete record of problems encountered and their solutions*
  - *Any preventative maintenance performed on the server.*
  - *Pertinent network information, such as cabling diagrams and topology maps*
    - Helps determine if a reported problem is due to the server itself or due to the network or segment it resides on.

Server Management Plans

- A server management plan is a plan that is used as a road map for monitoring and servicing both OS-dependent and OS-independent server components efficiently.
  - *Procedures for managing software installation, availability of service providers, change requirements, security requirements, and remote management. Server management plans also identify the person or persons responsible for carrying out each management task.*

You will need to include this in your assignment

Server Management Plans

- This plan is implemented by
  - *Developing quantifiable performance expectations*
  - *Collecting and documenting baseline statistics.*
  - *It is a good practice to regularly document the performance statistics and compare the server event logs with the baseline expectations.*

Services

- This section covers:
  - *Services in general (i.e., how are the Unix daemons or Windows server processes managed)*
  - *Directory services*
- Directory services is a large topic that tries to get at the heart of a hard problem: how is data managed across a system?
  - *More concretely*
    - Why does every application have a different configuration file in different locations, each of which uses a different syntax?
    - Why is Operating System configuration information scattered around a system?
  - *Would it then be better to create some standards for managing this data?*

Common Ground

- In this section, we'll be discussing:
  - *Windows Services*
  - *Windows Registry*
  - *Active Directory*
  - *Unix daemons started from xinetd*
  - *Unix's /etc/nsswitch.conf*
  - *NIS*

Common Ground (cont.)

- For directory services, these are some helpful dimensions to consider:
  - *What data?*
  - *Where is it kept?*
  - *How does the application/system know how to find the data?*
  - *How does the application/system access the data?*
- In addition
  - *How are each of those changed?*
  - *Which can or can't be changed?*
  - *What are the implications of changing (or not changing) them?*

Common Ground (cont.)

- Some kinds of data that can go into directory services:
  - *Hosts/IPs*
  - *Users*
  - *Passwords*
  - *Groups*
  - *Netgroups*
  - *Services*
  - *Networks*
  - *Software configuration*
  - *Software installation/availability*

Windows Services

- Windows Services:
  - *Control Panel -> Administrative Tools -> Services or net start*
  - *Tabs*
    - General
      - *Startup Type – Auto, Manual, Disabled*
      - *Status - Start, Pause, Resume, Stop*
    - Log On – Local system acct or other acct
    - Recovery – What should be done if service fails?
    - Dependencies – What depends on this service and what does this service depend on?

Windows Services (cont.)

- How is data managed across a system?
  - *Registry*
    - Hives
      - *System.dat & User.dat*
    - Setting
      - *Keys*
        - HKEY_CURRENT_USER\Control Panel\Appearance\New Schemes\11
  - *Active Directory*
    - Centrally stores information and settings
    - Some items are distributed back to the clients

Linux Services

- Xinetd – a daemon that manages other daemons
  - *Listens on behalf of the daemons it manages*
  - *Starts the managed daemons as needed*
  - *Shuts down the managed daemons as needed*
- Advantages of Xinetd over inetd
  - *Flexible configuration language*
  - *More secure*
  - *Better log management*

Linux Services (cont.)

- Xinetd
  - *Can do more than just start, pause, or stop services*
  - *However, it doesn't handle dependencies*
  - *Daemon is xinetd*
  - *Default configuration is /etc/xinetd.conf*
  - *Configuration can include the directive includedir /etc/xinetd.d*
  - *Supports options such as: only_from, no_access, access_times, redirect, cps, max_load, and instances*

Linux Services (cont.)

```
Defaults
{
log_type = SYSLOG local4 info
log_on_success = HOST EXIT
log_on_failure = HOST ATTEMPT RECORD
instances = 2
}
Service ftp
{
socket_type = stream
Protocol = tcp
wait = no
user = root
server = /usr/sbin/wu.ftpd
server_args = -a
only_from = 128.138.0.0/16
log_on_success += Duration
}
Includedir /etc/xinetd.d
```

Linux Services (cont.)

Sample of file included from xinetd.d:

```
# default: off
# description: rsync server is a good addition to an
# ftp server, as it allows crc checksumming etc.
service rsync
{
disable = yes
socket_type = stream
wait = no
user = root
server = /usr/bin/rsync
server_args = --daemon
log_on_failure += USERID
}
```

Linus Services (cont.)

- /etc/nsswitch.conf enables various Directory-related libraries
- *Information sources for the following items can be configured using nsswitch.conf in Linux*
  - passwd
  - shadow
  - group
  - ethernet addresses
  - hosts (name/IP address mappings)
  - network names and numbers
  - network protocols
  - services
  - mail aliases
  - automount

Linux Services (cont.)

- *Locations to include can be: flat files in standard locations, DNS, NIS, NIS+, etc.*
- The entry '[NOTFOUND=return]' means that the search should stop if the search in the previous location turned up nothing. If the search failed due to some other reason (like no NIS server responding) then the search continues with the next entry
- Example:
  - passwd: files db nisplus nis
  - hosts: files dns
  - group: files
  - shadow: db files
  - services:nisplus [NOTFOUND=return] files

Linux Service (cont.)

- NIS: Network Information Services, a protocol developed by Sun to help manage directory services, who made the standard public, and it became a de facto standard. However, the lack of security is a serious problem today.
  - *Simple (built on RPCs, master-slave server relationship with replication, a broadcast-based protocol for client-master communication)*
  - *No security*
  - *Client can only be in one domain (which is separate from DNS domain!)*
  - *Client: ypbind*
  - *Server: ypserv*
  - *Other commands: ypcat, ypwhich, ypdomainname and ypset*

Linux Services (cont.)

- *Files: /etc/yp.conf and*
- */var/ypbinding/[domainname]*
- *NIS server configuration:*
  - starts with /etc/ypserv.conf
  - /var/yp/Makefile is the basic configuration file.
  - Configuration is basically:
    - *Pre-setup Makefile (determining maps and locations of data input).*
    - *Edit contents*
    - *ypmake (which just does a make -f Makefile usually).*

HOST AND DOMAIN NAME RESOLUTION

Domain Name System (DNS)

NetBIOS

DNS Names

- DNS is a name resolution method.
  - *Invented to overcome the problems with "host" files.*
- Hierarchical distributed database.
  - *A domain name is a two (three) tier name.*
  - *Administered centrally (e.g. InterNIC)*
  - *First part identifies the organisation e.g. Curtin*
  - *Second part is a classification e.g. edu*
  - *Last comes the country e.g. au*
  - *Also referred to as top level domains (TLD)*

DNS Names (cont.)

- DNS consists of thousands of servers.
  - *Each contains its own part of the database.*
- Example
  - *Hostname = mycomputer*
  - *domain = curtin.edu.au*
  - *FQDN = mycomputer.curtin.edu.au*

DNS - Overview

- Three components:
  - *resolver (client)*
  - *name server (named)*
  - *zone files (or some form of database, like AD). A zone corresponds roughly to a domain.*
- Resolver is set in platform-specific ways. The two key pieces of information are:
  - *What domain am I in by default?*
  - *What servers should I get DNS information from?*
  - *Other information can be configured as well*

DNS Overview (cont.)

- Servers can be:
  - *Masters that are the authoritative source for a domain*
  - *Slaves that download information from a master via zone transfers*
  - *Caching-only servers that only cache queries and don't pre-fetch*
- Zone files are databases -- we'll discuss this in a bit more detail when we see actual named.conf and zone files

DNS – Platform-specific Issues

- Windows client configuration is handled via DHCP or through the GUI
- Linux client is managed in /etc/resolv.conf
- Windows server DNS is usually managed through the Microsoft Management Console (MMC) and can be configured as part of Active Directory (AD)
- Linux server is named and base configuration is in /etc/named.conf, but the actual zone files themselves are typically in /var/named

Host Files

- InterNIC's host.txt file is no longer maintained.
- Host files are still relevant.
  - *Linux, UNIX, NT and static networks.*
- Points to note when editing/creating host files.
  - *Names must be separated by at least one space.*
  - *Additional names on a line become aliases.*
  - *The file is parsed from top to bottom. Ie when the first match is found parsing stops.*
    - Therefore place server entries at the top.
  - *# is the comment symbol.*
  - *FQDNs are allowed but not encouraged.*
    - They can cause difficult to trace problems if incorrect.

DNS

- Tools to test DNS include
  - *ping, nslookup, dig, telnet, ftp etc*
  - *When checking DNS, make sure you disable the hosts file - its checked first.*
  - *Ping returns*
    - IP Number
    - FQDN
    - Packet size
    - Round trip delay
    - TTL setting

DNS (cont.)

- Local administrators maintain their part of the DNS.
  - *This must be accessible to the rest of the DNS system.*
  - *Individual servers handle queries.*
    - Authoritative and non authoritative answers.
- Allows rapid, up to date answers to name queries.

Address Resolution Protocol (RFC826)

- Resolves hardware address if only IP is known
  - *Retrieves address from cache if present.*
  - *If not present ARP issues a broadcast (layer 2).*
    - Includes the IP of the intended node.
    - All nodes examines the IP contained.
    - Matching node (if present) issues an ARP response.
  - *ARP then adds the IP to its Cache.*
- To aid efficiency all that receives an ARP request adds the physical/IP pair to its cache.

Address Resolution Protocol (cont.)

- RARP stands for reverse ARP.
  - *Looks up the IP from the physical address*
  - *Used in conjunction with bootp*
    - Obsolete
- Bootp (boot prom)
  - *Many NICs contain a socket for an Eprom.*
  - *This contains the bootp protocol.*
  - *Loads the OS from a network server.*
    - Prebuilt for a node for a specific IP
    - MAC address selects which copy of the OS to load.

Dynamic Host Configuration Protocol

- DHCP
- Allows a set of hosts to share a pool of IP addresses.
- Newly booted computer broadcasts to discover subnet.
  - *Datagram destined for UDP port 68.*
    - This port is reserved for bootp and DHCP.
    - This contains the MAC address of the DHCP Client.
    - And other configuration information.

DHCP (cont.)

- DHCP servers reply with offers of IP addresses.
  - *Only if the server has unleased IPs available.*
  - *Sent via broadcast to the node that issued the DHCP discover.*
  - *Sent to UDP port 67.*
  - *Contains the MAC address of the DHCP Client.*
  - *Also contains the IP and physical address of the DHCP server.*
  - *Contains the IP address being offered and the subnet mask for this particular network.*
    - *Note : The client may receive many offers at this point.*

DHCP (cont.)

- *Host picks one and broadcasts a request to a particular server.*
  - This contains the IP address of the server and the MAC address of the client.
  - This performs two functions.
    - *Notifies the selected DHCP server that the IP is requested.*
    - *Notifies all other DHCP servers that they may retract their offers.*

DHCP (cont.)

- All other servers withdraw offers, and selected server sends an ack.
  - *This is the final datagram of the DHCP transaction.*
  - *Includes the IP and subnet mask for the client.*
  - *May also include default gateway and WINS server addresses.*
  - *Other fields include:*
    - Lease period.
    - T1 and T2 - used when the client attempts to renew the lease.

DHCP (cont.)

- When done, host sends a release.
- Server reuses IP addresses when their lease is over.
- Time Fields
  - *T1 indicates when the client should begin the process of renewing the lease.(usually 50%)*
  - *T2 is the time to start querying other DHCP servers if the original server does not respond.*
- Renewals are not broadcast.