

# Introduction to Linux for SQL Server and Windows Professionals

Due to issues with printing large PowerPoint Presentations to PDF format, I have moved the comments from the slide deck to a standalone document. Information will directly to the slide title.

- \*nix Versions (a few anyway)
  - These are a few different versions of \*nix over the years.
    - Coherent was the first version I used on Intel processors
    - I used IBM AIX and HP-UX in the mid-90's to early 2000's for healthcare claims processing
    - Solaris was used in the Public Lands division at TPWD for geographic mapping and CAD for state park enhancements
    - Fedora, Red Hat (enterprise), and CentOS (development, free) are one of three basic distribution groups of Linux

- Debian and Ubuntu are another. Ubuntu is free, published by Canonical Ltd from the UK with an office in Austin. Canonical provides paid support.
  - SUSE and OpenSUSE are another. This was popular in mid '90's because it was relatively low cost. In 2003 SUSE was acquired by Novell. In 2011 Novell and SUSE were acquired by The Attachmate Group who allowed SUSE to operate as an independent company. In October 2014, the Attachmate portfolio was acquired by Micro Focus International (yes, the COBOL folks) with SUSE still operating as an independent. In July 2018 a Swedish-based private equity firm acquired SUSE from Micro Focus.
- Linux File System
    - / – Root
      - Every single file and directory starts from the root directory.
      - Only root user has write privilege under this directory.

- NOTE: /root is the root user's home directory, which is not same as /.
- /bin – User Binaries
  - Contains binary executables.
  - Common Linux commands you need to use in single-user modes are located under this directory.
  - Commands used by all the users of the system are located here.
  - Example: ps, ls, ping, grep, cp.
- /sbin – System Binaries
  - Just like /bin, /sbin also contains binary executables.
  - But the Linux commands located under this directory are used typically by system administrators, for system maintenance purposes.
  - Example: iptables, reboot, fdisk, ifconfig, swapon
- /etc – Configuration Files
  - Contains configuration files required by all programs.

- This also contains startup and shutdown shell scripts used to start/stop individual programs.
- Example: /etc/resolv.conf,  
/etc/logrotate.conf
- /dev – Device Files
  - Contains device files.
  - These include terminal devices, USB, or any device attached to the system.
  - Example: /dev/tty1, /dev/usbmon0
- /proc – Process Information
  - Contains information about system process.
  - This is a pseudo filesystem contains information about running process. For example: /proc/{pid} directory contains information about the process with that pid.
  - This is a virtual filesystem with text information about system resources. For example: /proc/uptime
- /var – Variable Files
  - var stands for variable files.
  - Content of the files that are expected to grow can be found under this directory.

- This includes — system log files (/var/log); packages and database files (/var/lib); emails (/var/mail); print queues (/var/spool); lock files (/var/lock); temp files needed across reboots (/var/tmp);
- /tmp – Temporary Files
  - Directory that contains temporary files created by system and users.
  - Files under this directory are deleted when system is rebooted.
- /usr – User Programs
  - Contains binaries, libraries, documentation, and source-code for second level programs.
  - /usr/bin contains binary files for user programs. If you can't find a user binary under /bin, look under /usr/bin. For example: at, awk, cc, less, scp
  - /usr/sbin contains binary files for system administrators. If you can't find a system binary under /sbin, look under /usr/sbin. For example: atd, cron, sshd, useradd, userdel
  - /usr/lib contains libraries for /usr/bin and /usr/sbin

- /usr/local contains users' programs that you install from source. For example, when you install apache from source, it goes under /usr/local/apache2
- /home – Home Directories
  - Home directories for all users to store their personal files.
  - Example: /home/groot, /home/nebula
- /boot – Boot Loader Files
  - Contains boot loader related files.
  - Kernel initrd, vmlinuz, grub files are located under /boot
  - Example: initrd.img-2.6.32-24-generic, vmlinuz-2.6.32-24-generic
- /lib – System Libraries
  - Contains library files that supports the binaries located under /bin and /sbin
  - Library filenames are either ld\* or lib\*.so.\*
  - Example: ld-2.11.1.so, libncurses.so.5.7
- /opt – Optional add-on Applications
  - opt stands for optional.
  - Contains add-on applications from individual vendors.

- add-on applications should be installed under either /opt/ or /opt/ sub-directory.
- /mnt – Mount Directory
  - Temporary mount directory where sysadmins can mount filesystems.
- /media – Removable Media Devices
  - Temporary mount directory for removable devices.
  - Examples:
    - /media/cdrom for CD-ROM
    - /media/floppy for floppy drives
    - /media/cdrecorder for CD writer
- /srv – Service Data
  - srv stands for service.
  - Contains server specific services related data.
  - Example: /srv/cvs contains CVS related data.
- Introduction to the BASH Shell
  - The BASH shell is the most common default shell distributed with Linux. The original shell in \*nix was called the bourne shell. BASH is the bourne-again shell as it replaced the bourne shell.

- Default naming convention for BASH shell scripts is name.sh. This is just for human readability as the extension in \*nix, unlike DOS, doesn't cause the shell to do anything special.
- If you want the script to be executable, you give the file execute attribute using the chmod command.
- You can also place a header in the script that will tell the shell which command processor should be used to execute the script. This is sometimes referred to as "sh-bang" and the format is `#!/<path to processor>`.
  - For example, to execute using the default shell use `#!/bin/sh` or `#!/bin/bash`. If you've created a python script it would be `#!/bin/python`. Once this is added, you can invoke by simply typing `./scriptname`. As long as you've set the correct permissions of course.
- A nice alternative to BASH is zsh, or z-shell. Among the nice features:
  - Direct URL access by simply control-click
  - Case insensitive
  - Auto command correction

- Auto complete feature
- Quick directory change using z instead of cd
  - You can now change directories by simply typing the directory name. No need for z or cd. You can even short cut if there are no conflicts.
  - Example: to get to `/var/opt/mssql` (assuming you have rights) you could just type `/v/o/mss` and tab.
- Command option help without having to switch to man pages
- To install zsh on Ubuntu 18.04
  - `sudo apt update`
  - `sudo apt install zsh`
  - `zsh --version`
- If you would like to change your default shell to zsh, use the `chsh` (change shell) command. This updates your entry in the `/etc/passwd` file to your desired shell. A list of valid shells is in `/etc/shells`.
  - To set zsh as your default the command would be: `chsh -s /usr/bin/zsh`

- You will be prompted for your password. For the change to take effect you will need to log out and back in.
- Common Command Equivalents between Windows and Linux
  - There are three big differences between Windows and \*nix that will take some getting used to:
    - \*nix is case sensitive (unless you are using a shell that ignores case)
    - Where Windows (usually) requires a backslash, \*nix always uses forward slash. In \*nix the backslash is an escape character
    - Line terminators are different. Linux uses newline (1 ascii character), Windows/DOS use CR/LF (2 ascii characters)
  - bat is a cat replacement available on GitHub. Link is in Online Resources section of the deck.
- User and group setup and associated security
  - Groups are just that, groups. They have no inherent meaning (other than root)
- Location of database, transaction log, and server log files

- You must be root or a member of mssql group to get to this folder as it is owned by mssql user.
  - log – location of sql log files
  - data – default location for system and user databases. With CU1, this can be customized by mssql-conf
  - secrets –
  - .system
- mssql.conf – Since there is no registry in Linux, settings you would expect to find in the registry are stored here.
- Task scheduling using cron
  - The crond daemon runs jobs on the system according to information stored in /etc/crontab. You can also have jobs run hourly, daily, weekly, and monthly by simply placing your desired script in a special folder named /etc/cron.<period>.
  - Use the crontab command to modify the crontab. Arguments are –e (edit) –l (list) –r (remove the file) –v (when was crontab last modified)

- You can set a default editor for crontab by issuing command `export EDITOR=vi` (or editor of your choice)
- If you want a task to run every 2 minutes use `*/2` for the mm value
- Accessing files on a Windows System from Linux
  - As previously mentioned, most \*nix commands and parameters are either acronyms or the name of the original developers pet. In this case, cifs is an acronym which stands for Common Internet File System.
  - Adding the entry to `/etc/fstab` will cause the share to be mounted on each restart.
  - `/etc/fstab` entry
  - Note: be careful editing this file as it could corrupt the system. Need to run editor in elevated mode.
  - `//servernameoraddress/sharename`  
`/mnt/mountname cifs`  
`credentials=/home/username/.smbcredentials,i`  
`charset=utf8,sec=ntlm,vers=3.0,uid=username,`  
`gid=groupid 0 0`
  - Mount the share
  - `sudo mount -a`

- Doing a non-persistent mount:
- `sudo mount -t cifs //servernameoraddress /sharename /mnt/mountname cifs -o credentials=/home/username/.smbcredentials,i ocharset=utf8,sec=ntlm,vers=3.0,uid=username, gid=groupid`
- Get cifs:
  - 1) `sudo apt-get update`
  - 2) `sudo apt-get install cifs-utils -y`
- Create credentials:
  - 1) `cd $HOME`
  - `vi .smbcredentials`
  - Insert `username=somewindowsuser`
  - Insert `password=somewindowspassword`
  - Insert (optional)
    - `domain=domainnameofauthuser`
  - `chmod 600 ~/.smbcredentials`
- Accessing Linux files from a Windows System (Ubuntu)
  - To install samba on Ubuntu:
    - `sudo apt-get update`
    - `sudo apt-get install samba`
  - To create a user in samba. Note: The user, if it doesn't exist in the normal passwd, will be

added. Recommend using an existing user.  
Password can be different.

- `sudo smbpasswd -a <username>`
- To add entry to samba configuration
- Make a backup copy of the existing file: `sudo cp /etc/samba/smb.conf ~`
- Edit the file using editor of choice. Ex: `sudo vi /etc/samba/smb.conf`
- Add the following information to the end of the file:
  - `[<sharename>]`
  - `path = <path to folder to share>`
  - `valid users = <sambausername>`
  - `read only = no`
- Note: Spaces before and after equal sign is required. Also indent each line under the share name tag.
- When creating the samba user, you will be prompted for a password and verification. If you use a username other than your own, you will need to be sure to grant access to any shares you create.
- To restart the Samba service: `systemctl restart smbd`

- PowerShell and Linux
  - Summarized here
  - # Import the public repository GPG keys
  - curl  
`https://packages.microsoft.com/keys/microsoft.asc | sudo apt-key add -`
  - # Register the Microsoft Ubuntu repository
  - curl  
`https://packages.microsoft.com/config/ubuntu/16.04/prod.list | sudo tee /etc/apt/sources.list.d/microsoft.list`
  - # Update the list of products
  - `sudo apt-get update`
  - # Install PowerShell
  - `sudo apt-get install -y powershell`
- Commands for Tuning
  - Some commands are not installed by default. To install: `sudo apt install sysstat -y`
- Advanced and/or not ready for Prime Time Topics
  - SQL Server Agent is installed by default starting with CU4. It just needs to be enabled.
- After thoughts
  - Cal options: -h turns off highlighting of current date, -j shows Julian dates, -3 shows previous

and next month, -y specific year, -m specific month

- Df -h shows a more friendly display
- Wc -l produces line count only, -c counts bytes, -m counts characters, -w counts words, -L length of longest line