LINUX FOR DEVICES

By: Dr. A K Yadav (9911375598)
Dept of CSE, ASET, AUUP

Objective and Learning Outcome

Objective:

- To be able to understand the basics of Linux
- To explores the basic characteristics of Linux Networking
- To helps in learning about Linux Shell, File Structure and Network Administration Services
- It gives overview about the Linux Security Techniques

Learning Outcomes

Student will be able to

- Perform the basic operations for Linux.
- Compare the various Linux security techniques.
- Implement the Docker in Linux.
- Execute the shell scripts on Linux.
- Devise the network administration services.
- Able to design device drivers

Recommended Reading

- Christopher Negus, "Linux Bible: The comprehensive Tutorial, Resource ", 8nd Edition, John Wiley
- Richard Petersen "Linux: The Complete Reference," 6th Edition, Tata Mc Graw Hill
- Jonathan Corbet, Alessandro Rubini, Greg Kroah-Hartman, "Linux Device Drivers", 3rd ed, O'Reilly

Module Assessment and Self Work

Module Assessment

- Quiz
- Assignment

PSDA (Self Work)

- Minor Experiment
- Group Discussion
- Case study

Module I - Topics to be covered

- Introduction to Linux
- File System of the Linux
- General usage of Linux kernel & basic commands
- Linux users and group
- Permissions for file, directory and users
- Searching a file & directory
- zipping and unzipping concepts
- Linux for the Industry 4.0 Era,
- OPENIL and its advantages, Features of OPENIL

About Linux, Linux System Architecture

Linux Kernel

LINUX FOR DEVICES

By: Dr. A K Yadav (9911375598)
Dept of CSE, ASET, AUUP

File System Structure

- All the files in your Linux system are connected into one overall directory tree
- Parts of the tree may reside on different storage devices such as hard drives or CD-ROMs.
- Files on a particular storage device are organized into that is referred to as a *file system*.
- A file system is a formatted device, with its own tree of directories and files.
- Linux directory tree may encompass several file systems, each on different storage devices.
- The files themselves are organized into one seamless tree of directories, beginning from the root directory

File System Hierarchy Standard (FHS)

- Linux organizes its files and directories into one overall interconnected tree, beginning from the root directory and extending down to system and user directories.
- The organization and layout for the system directories are determined by the file system hierarchy standard (FHS).
- The FHS provides a standardized layout that all Linux distributions should follow in setting up their system directories.
- Linux distributions, developers, and administrators all follow the FHS to provide a consistent organization to the Linux file system.

File System Structure

- Root Directory: /
- System Directories
- Program Directories
- Configuration Directories and Files
- The /proc File System
- The sysfs File System: /sys
- Device Files: /dev, udev, and HAL
- Floppy and Hard Disk Devices
- CD-ROM Devices
- Mounting File Systems
- File System Information
- Journaling
- Mounting File Systems Automatically: /etc/fstab

Root Directory: /

Directory	Function
/	Begins the file system structure—called the root.
/boot	Holds the kernel image files and associated boot information and files.
/home	Contains users' home directories.
/sbin	Holds administration-level commands and any commands used by the root user.
/dev	Holds dynamically generated file interfaces for devices such as the terminal and the printer (see "udev: Device Files" in Chapter 31).
/etc	Holds system configuration files and any other system files.
/etc/opt	Holds system configuration files for applications in /opt.
/etc/X11	Holds system configuration files for the X Window System and its applications.
/bin	Holds the essential user commands and utility programs.
/lib	Holds essential shared libraries and kernel modules.
/lib/modules	Holds the kernel modules.
/media	Holds directories for mounting media-based removable file systems, such as CD-ROMs, floppy disks, USB card readers, and digital cameras.
/mnt	Holds directories for additional file systems such as hard disks.
/opt	Holds added software applications (for example, KDE on some distributions).
/proc	Process directory, a memory-resident directory that contains files used to provide information about the system.
/sys	Holds the sysfs file system for kernel objects, listing supported kernel devices and modules.
/tmp	Holds temporary files.
/usr	Holds those files and commands used by the system; this directory breaks down into several subdirectories.
/var	Holds files that vary, such as mailbox, web, and FTP files.

System Directories

Directory	Description
/bin	Holds system-related programs.
/sbin	Holds system programs for specialized tasks.
/lib	Holds system libraries.
/etc	Holds configuration files for system and network services and applications.
/home	Holds user home directories and server data directories, such as website and FTP site files.
/media	Where removable media file systems like CD-ROMs, USB drives, and floppy disks are mounted.
/var	Holds system directories whose files continually change, such as logs, printer spool files, and lock files.
/usr	Holds user-related programs and files. Includes several key subdirectories, such as **/usr/bin**, **/usr/X11**, and **/usr/share/doc**.
/usr/bin	Holds programs for users.
/dev	Holds device files.
/sys	Holds the sysfs file system with device information for kernel-supported devices on your system.
/usr/X11	Holds X Window System configuration files.
/usr/share	Holds shared files.
/usr/share/doc	Holds documentation for applications.
/usr/share/hal	Holds configuration for HAL removable devices.
/etc/udev	Holds configuration for device files.
/tmp	Holds system temporary files.

Program Directories

- Directories with bin in the name are used to hold programs.
- The /bin directory holds basic user programs, such as login, shells and file commands
- The /sbin directory holds specialized system programs for such tasks as file system management and system operations.
- The /usr/bin directory holds program files designed for user tasks.
- The /usr/sbin directory holds user-related system operation.
- The /lib directory holds all the libraries your system makes use of and subdirectories such as modules, which holds all the current kernel modules.

Configuration Directories and Files

- When you configure different elements of your system, such as user accounts, applications, servers, or network connections, you make use of configuration files kept in certain system directories.
- Configuration files are placed in the /etc directory.

The /usr Directory

Directory	Description
/usr/bin	Holds most user commands and utility programs.
/usr/sbin	Holds administrative applications.
/usr/lib	Holds libraries for applications, programming languages, desktops, and so on.
/usr/games	Holds games and educational programs.
/usr/include	Holds C programming language header files (.h).
/usr/doc	Holds Linux documentation.
/usr/local	Holds locally installed software.
/usr/share	Holds architecture-independent data such as documentation.
/usr/src	Holds source code, including the kernel source code.
/usr/X11R6	Holds X Window System–based applications and libraries.

The /var Directory

Directory	Description
/var/account	Processes accounting logs.
/var/cache	Holds application cache data for Man pages, web proxy data, fonts, or application-specific data.
/var/crash	Holds system crash dumps.
/var/games	Holds varying games data.
/var/lib	Holds state information for particular applications.
/var/local	Holds data that changes for programs installed in **/usr/local**.
/var/lock	Holds lock files that indicate when a particular program or file is in use.
/var/log	Holds log files such as **/var/log/messages** that contain all kernel and system program messages.
/var/mail	Holds user mailbox files.
/var/opt	Holds variable data for applications installed in **/opt**.
/var/run	Holds information about the system's running processes.
/var/spool	Holds applications' spool data such as that for mail, news, and printer queues, as well as **`cron`** and **`at`** jobs.
/var/tmp	Holds temporary files that should be preserved between system reboots.
/var/yp	Holds Network Information Service (NIS) data files.
/var/www	Holds web server website files.
/var/ftp	Holds FTP server FTP files.
/var/named	Holds DNS server domain configuration files.

The /proc File System

File	Description
/proc/*num*	There is a directory for each process labeled by its number. **/proc/1** is the directory for process 1.
/proc/cpuinfo	Contains information about the CPU, such as its type, make, model, and performance.
/proc/devices	Lists the device drivers configured for the currently running kernel.
/proc/dma	Displays the DMA channels currently used.
/proc/filesystems	Lists file systems configured into the kernel.
/proc/interrupts	Displays the interrupts in use.
/proc/ioports	Shows the I/O ports in use.
/proc/kcore	Holds an image of the physical memory of the system.
/proc/kmsg	Contains messages generated by the kernel.
/proc/loadavg	Lists the system load average.
/proc/meminfo	Displays memory usage.
/proc/modules	Lists the kernel modules currently loaded.
/proc/net	Lists status information about network protocols.
/proc/stat	Contains system operating statistics, such as page fault occurrences.
/proc/uptime	Displays the time the system has been up.
/proc/version	Displays the kernel version.

Linux for Devices

Module-1, Lecture-4

By: Dr. A K Yadav (9911375598)
Dept of CSE, ASET, AUUP

Linux Shell

- The shell is program used to interpret and manage commands.
- It provides a way to create executable script files, run programs, work with file systems, compile computer code, and manage the computer.
- It is a command interpreter that provides a line-oriented interactive and noninteractive interface between the user and the operating system.
- The commands entered on a command line are interpreted by the shell and then sent as instructions to the operating system
- Commands can be placed in a script file to be executed consecutively a normal program.
- Different types of shells have been developed for Linux

Linux Shell

- Bourne shell-sh (first shell on Unix systems)
- Bourne Again shell (BASH)-bash (Linux standard shell)
- Korn shell-ksh (UNIX System V users)
- C shell-csh (BSD UNIX users)
- TCSH shell-tcsh (an improved C shell)
- Z shell-zsh (extended Bourne shell with many improvements)
- Dash shell-dash (Ubuntu default)

Linux Shell

- Linux commands
- Shell scripts
- chmod command
- Command line arguments
- Taking input
- Printing output
- man, help, etc
- Internal and External commands
- cat for input

- expr
- Backslash (\) for escapes
- Command Line Editing

Movement Commands	Operation
CTRL-F, RIGHT-ARROW	Move forward a character.
CTRL-B, LEFT-ARROW	Move backward a character.
CTRL-A or HOME	Move to beginning of line.
CTRL-E or END	Move to end of line.
ALT-F	Move forward a word.
ALT-B	Move backward a word.
CTRL-L	Clear screen and place line at top.

Editing Commands	Operation
CTRL-D or DEL	Delete character cursor is on.
CTRL-H or BACKSPACE	Delete character before the cursor.
CTRL-K	Cut remainder of line from cursor position.
CTRL-U	Cut from cursor position to beginning of line.
CTRL-W	Cut previous word.
CTRL-C	Cut entire line.
ALT-D	Cut the remainder of a word.
ALT-DEL	Cut from the cursor to the beginning of a word.

ALT-Y	Paste from set of previously cut text.
CTRL-Y	Paste previous cut text.
CTRL-V	Insert quoted text, used for inserting control or meta (ALT) keys as text, such as CTRL-B for backspace or CTRL-T for tabs.
ALT-T	Transpose current and previous word.
ALT-L	Lowercase current word.
ALT-U	Uppercase current word.
ALT-C	Capitalize current word.
CTRL-SHIFT-_	Undo previous change.

Linux for Devices

Module-1, Lecture-5

By: Dr. A K Yadav (9911375598)
Dept of CSE, ASET, AUUP

Linux users and group

- Understanding /etc/group
- Adding a Group (groupadd) (**groupadd -g GID GNAME**)
- Understanding /etc/passwd
- Understanding /etc/shadow
- Adding a User (useradd)
- **useradd -u 210 -g cse -c “Dr A K Yadav” -d /home/akyadav -s /bin/ksh -m akyadav**
- Modifying and Removing Users (usermod and userdel)

Permissions for file & directory

- Listing File Attributes (ls –l)
- Type and Permissions
- Links
- Ownership and Group Ownership
- Size
- Last Modification Time
- Filename

Changing File Permissions

- chmod
- relative manner
- absolute manner

Linux for Devices

Module-1, Lecture-6

By: Dr. A K Yadav (9911375598)
Dept of CSE, ASET, AUUP

Searching a file or directory

- Find command
- Locate command
- Which command
- whereis command

Find command

- find options path_list selection_criteria action
- find [-H] [-L] [-P] [-D debugopts] [-Olevel] [starting-point...] [expression]
- The options controls the treatment of the symbolic links, debugging options, and optimization method.
- The path defines the starting directory or directories from where find will start searching the files.
- The expression attribute is made up of options, search patterns, and actions separated by operators.

- [-H] option: Do not follow symbolic links, except while processing the command line arguments. When find examines or prints information about files, the information used shall be taken from the properties of the symbolic link itself. The only exception to this behaviour is when a file specified on the command line is a symbolic link, and the link can be resolved.
- [-L] option: Follow symbolic links. When find examines or prints information about files, the information used shall be taken from the properties of the file to which the link points, not from the link itself (unless it is a broken symbolic link or find is unable to examine the file to which the link points)

- [-P] option: Never follow symbolic links. This is the default behaviour. When find examines or prints information about a file, and the file is a symbolic link, the information used shall be taken from the properties of the symbolic link itself.
- [-D debugopts] option: Print diagnostic information; this can be helpful to diagnose problems with why find is not doing what you want. The list of debug options should be comma separated.
- [-Olevel] option: Enables query optimization. The find program reorders tests to speed up execution while preserving the overall effect

Locate command

- Locate command: find files by name
- locate [OPTION]... PATTERN...
- Example locate -b '\NAME': To search for a file named exactly NAME (not *NAME*). \ is a globing character, this disables the implicit replacement of NAME by *NAME*.

which, whereis and type

- The which command tells you the directory that contains the command
- It searching the directories of PATH in sequence, and abandons its search just after it locates a file
- The locate command is faster than the find command because it uses a previously built database, whereas the find command searches in the real system, through all the actual directories and files. The locate command returns a list of all path names containing the specified group of characters

- The “which” command returns the absolute path of the executable that is called when a command is issued.
- This is useful in finding the location of an executable for creating a shortcut to the program on the desktop, on a panel, or other place in the desktop manager.
- The whereis command is used to find out where the binary, source, and man page files for a command are located

Linux for Devices

Module-1, Lecture-8

By: Dr. A K Yadav (9911375598)
Dept of CSE, ASET, AUUP

- Industry 4.0 as the Fourth Industrial Revolution
- Origin and Characterization of Industry 4.0
- Industry 4.0—A Challenge

Linux for Devices

Module-1, Lecture-7

By: Dr. A K Yadav (9911375598)
Dept of CSE, ASET, AUUP

Zipping and Unzipping concepts

- gzip & gunzip
- zip & unzip
- bzip2
- tar
- zipcloak
- gpg-zip
- gpgtar

Linux for Devices

Module-1, Lecture-8

By: Dr. A K Yadav (9911375598)
Dept of CSE, ASET, AUUP

- Industry 4.0 as the Fourth Industrial Revolution
- Origin and Characterization of Industry 4.0
- Industry 4.0 - A Challenge

OpenIL

- NXP Semiconductors, a leader in advanced secure connectivity solutions, has announced an industrial Linux distribution with real-time OS extensions and Time-Sensitive Networking (TSN) support for factory-automation Original Equipment Manufacturer (OEMs).
- By breaking down the barriers of real-time computing and networking in a standard, community-based distribution, Open Industrial Linux (OpenIL) helps OEMs usher in the Industry 4.0 era.

- OpenIL provides industrial grade security, trusted computing, hardened software, cryptographic operations and end to end security

OpenIL features

- Xenomai — a real-time framework for Linux
- XML and NETCONF network configuration utilities
- Precision time synchronization using gPTP
- Ethernet drivers for time-sensitive networking
- Support for edge computing services