1. **What is Operating System? History of OS? Different types of OS? Difference between 32 bit and 64 bit?**

**Answer:**

An operating system (OS) is system software that manages computer hardware and software resources and provides common services for computer programs. All computer programs, excluding firmware, require an operating system to function.

The three most popular types of operating systems for personal and business computing include Linux, Windows and Mac.

Operating systems have evolved through a number of distinct phases or generations which corresponds roughly to the decade**s.**

### The 1940's - First Generations

The earliest electronic digital computers had no operating systems. Machines of the time were so primitive that programs were often entered one bit at time on rows of mechanical switches (plug boards). Programming languages were unknown (not even assembly languages). Operating systems were unheard of .

### The 1950's - Second Generation

By the early 1950's, the routine had improved somewhat with the introduction of punch cards. The General Motors Research Laboratories implemented the first operating systems in early 1950's for their IBM 701. The system of the 50's generally ran one job at a time. These were called single-stream batch processing systems because programs and data were submitted in groups or batches.

### The 1960's - Third Generation

The systems of the 1960's were also batch processing systems, but they were able to take better advantage of the computer's resources by running several jobs at once. So operating systems designers developed the concept of multiprogramming in which several jobs are in main memory at once; a processor is switched from job to job as needed to keep several jobs advancing while keeping the peripheral devices in use.

For example, on the system with no multiprogramming, when the current job paused to wait for other I/O operation to complete, the CPU simply sat idle until the I/O finished. The solution for this problem that evolved was to partition memory into several pieces, with a different job in each partition. While one job was waiting for I/O to complete, another job could be using the CPU.

Another major feature in third-generation operating system was the technique called spooling (simultaneous peripheral operations on line). In spooling, a high-speed device like a disk interposed between a running program and a low-speed device involved with the program in input/output. Instead of writing directly to a printer, for example, outputs are written to the disk. Programs can run to completion faster, and other programs can be initiated sooner when the printer becomes available, the outputs may be printed.

Note that spooling technique is much like thread being spun to a spool so that it may be later be unwound as needed.

Another feature present in this generation was time-sharing technique, a variant of multiprogramming technique, in which each user has an on-line (i.e., directly connected) terminal. Because the user is present and interacting with the computer, the computer system must respond quickly to user requests, otherwise user productivity could suffer. Timesharing systems were developed to multiprogram large number of simultaneous interactive users.

### Fourth Generation

With the development of LSI (Large Scale Integration) circuits, chips, operating system entered in the personal computer and the workstation age. Microprocessor technology evolved to the point that it become possible to build desktop computers as powerful as the mainframes of the 1970s. Two operating systems have dominated the personal computer scene: MS-DOS, written by Microsoft, Inc. for the IBM PC and other machines using the Intel 8088 CPU and its successors, and UNIX, which is dominant on the large personal computers using the Motorola 6899 CPU family.

The terms **32**-**bit and 64**-**bit** refer to the way a computer's processor (also called a CPU), handles information. The **64**-**bit** version of Windows handles large amounts of random access memory (RAM) more effectively than a **32**-**bit** system. A big difference between 32-bit processors and 64-bit processors is the number of calculations per second they can perform, which affects the speed at which they can complete tasks. 64-bit processors can come in dual core, quad core, six core, and eight core versions for home computing. Multiple cores allow for an increased number of calculations per second that can be performed, which can increase the processing power and help make a computer run faster. Software programs that require many calculations to function smoothly can operate faster and more efficiently on the multi-core 64-bit processors, for the most part.

**2. Difference between Linux and windows**

Answer:  
Linux: Available for lot of devices.

Basic version of linux is free.

Linux is an open source software.

Red Hat, Android, Debian are examples of Linux OS.

Linux has less number of games.

Linux is most secure than windows OS.

Windows: Windows is less available than Linux.

Windows costs from $50 to $450

Windows is not an open source software

Windows XP, Vista, 7, 8 and 8.1 are examples of Windows.

More games are available for windows

Windows is more prone to viruses

**3. What are difference between linux and unix?**

**Answer:**

**UNIX**

UNIX is multitasking and multiuser operating system created by Ken Thompson, Dennis Ritchie at AT&T in 1970. Initially intended for use inside the Bell System.

Different flavors of Unix have different cost structures according to vendors.

Unix systems are divided into various other flavors, mostly developed by AT&T as well as various commercial vendors and non-profit organizations.

Three bigest distributions are Solaris (Oracle), AIX (IBM) & HP-UX Hewlett Packard. And Apple Makes OSX, an unix based os.

Unix operating systems were developed mainly for mainframes, servers and workstations except OSX, Which is designed for everyone. The Unix environment and the client-server program model were essential elements in the development of the Internet.

The UNIX operating system is used in internet servers, workstations & PCs. Backbone of the majority of finance infastructure and many 24x365 high availability solutions.

Originally the Bourne Shell. Now it's compatible with many others including BASH, Korn & C..

**Linux**

Linux is unix-like operating system released as free and open source. Linux is created by Linus Torvalds. Linux distributions uses Linux kernel to build the whole operating systems. Not only a Linux, BSD unix also a unix-like operating system. But Linux has more volunteers and evolved rapidly.

BASH (Bourne Again SHell) is the Linux default shell. It can support multiple command interpreters.

Linux is free but support for it is avaialable at a price. Linux has about 60-100 viruses till date. None of them actively spreading nowadays. It supports different kinds of processors.

**4. Explain various distributions of Linux? Explain their Pros and Cons?**

**Answer:**

Linux isn’t a complete operating system — it’s just a kernel. Linux distributions take the Linux kernel and combine it with other free software to create complete packages. There are many different Linux distributions out there.

If you want to “install Linux,” you’ll need to choose a distribution. You could also use Linux From Scratch to compile and assemble your own Linux system from the ground up, but that’s a huge amount of work.

**Ubuntu**

Ubuntu is probably the most well-known Linux distribution. Ubuntu is based on Debian, but it has its own software repositories. Much of the software in these repositories is synced from Debian’s repositories.

## Linux Mint

## Mint is a Linux distribution built on top of Ubuntu. It uses Ubuntu’s software repositories, so the same packages are available on both. Originally, Mint was an alternative distribution loved mainly because it included media codecs and proprietary software that Ubuntu didn’t include by default.

## Debian

## Debian is an operating system composed only of free, open-source software. The Debian project has been operating since 1993 — over 20 years ago! This widely respected project is still releasing new versions of Debian, but it’s known for moving much more slowly than distributions like Ubuntu or Linux Mint. This can make it more stable and conservative, which is ideal for some systems.

## Fedora

Fedora is a project with a strong focus on free software — you won’t find an easy way to install proprietary graphics drivers here, although third-party repositories are available. Fedora is bleeding edge and contains the latest versions of software.

## CentOS / Red Hat Enterprise Linux

Red Hat Enterprise Linux is a commercial Linux distribution intended for servers and workstations. It’s based on the open-source Fedora project, but is designed to be a stable platform with long-term support.

## openSUSE / SUSE Linux Enterprise

openSUSE is a community-created Linux distribution sponsored by Novell. Novell purchased SuSE Linux in 2003, and they still create an enterprise Linux project known as SUSE Linux Enterprise. Where Red Hat has the Fedora project that feeds into Red Hat Enterprise Linux, Novell has the openSUSE project that feeds into SUSE Linux Enterprise.

## Mageia / Mandriva

Mageia is a fork of Mandriva Linux created in 2011. Mandriva — known as Mandrake before that — was once one of the great user-friendly Linux distributions.

## Arch Linux

Arch Linux is more old school than many of the other Linux distributions here. It’s designed to be flexible, lightweight, minimal, and to “Keep it Simple.” Keeping it simple doesn’t mean Arch provides tons of graphical utilities and automatic configuration scripts to help you set up your system. Instead, it means Arch dispenses with that stuff and gets out of your way.

## Slackware Linux

Slackware is another institution. Founded in 1993, Slackware is the oldest Linux distribution that’s still maintained and putting out new releases today.

## Puppy Linux

## Puppy Linux is another fairly well-known Linux distribution. Previous versions have been built on Ubuntu, but the latest is built on Slackware. Puppy is designed to be a small, lightweight operating system that can run well on very old computers. The puppy ISO file is 161 MB, and Puppy can boot from that disc in a live environment. Puppy can run on PCs with 256 MB or RAM, although it does recommend 512 MB for the best experience.

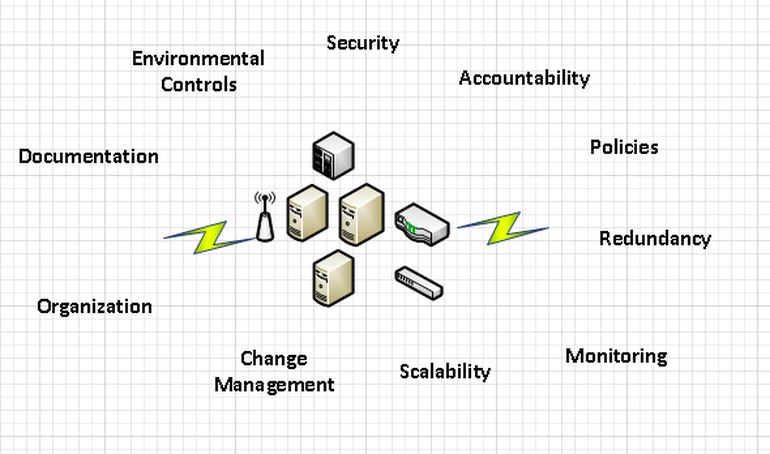
**5.What is Data Center? What are the various factors considered to plan a Data center? Discuss about few important components in Data Center?**

**Answer:**

A data center (or datacenter) is a facility composed of networked computers and storage that businesses or other organizations use to organize, process, store and disseminate large amounts of data. A business typically relies heavily upon the applications, services and data contained within a data center, making it a focal point and critical asset for everyday operations.

For most companies, licensing space from a data center colocation service provider is almost always a better business and technical decision than maintaining private facilities. The exploding demands placed on IT infrastructure as well as the high operating cost make it difficult for in-house data centers to keep pace. As you examine which third-party data center is right for your organization, here are five key characteristics to consider: location, amenities, network services, security and pricing.

Key components of data center are Environmental control, Security, Accountability, Policies, Redundancy, Monitoring, Scalability, Change management, Organization, Documation.



**6. What is RAM, ROM, BIOS, POST, KERNEL?**

**Answer:**

RAM:

**Random-access memory** (**RAM** /ræm/) is a form of computer data storage which stores frequently used program instructions to increase the general speed of a system. A random-access memory device allows data items to be read or written in almost the same amount of time irrespective of the physical location of data inside the memory. In contrast, with other direct-access data storage media such as hard disks, CD-RWs, DVD-RWs and the older drum memory, the time required to read and write data items varies significantly depending on their physical locations on the recording medium, due to mechanical limitations such as media rotation speeds and arm movement.

**ROM**:

ROM is "built-in" computer memory containing data that normally can only be read, not written to. ROM contains the programming that allows your computer to be "booted up" or regenerated each time you turn it on. Unlike a computer's random access memory ([RAM](http://searchmobilecomputing.techtarget.com/definition/RAM)), the data in ROM is not lost when the computer power is turned off. The ROM is sustained by a small long-life battery in your computer.

BIOS:

BIOS (basic input/output system) is the program a personal computer's microprocessor uses to get the computer system started after you turn it on. It also manages data flow between the computer's operating system and attached devices such as the hard disk, video adapter, keyboard, mouse and printer.

POST:

Short for **power-on self-test**, the **POST** is a test the computer must complete to verify all hardware is working properly before starting the remainder of the boot process. If the computer passes the POST the computer may return a single beep and if unsuccessful generate a **beep code** to indicate the error and does not boot. All POST errors are relating to hardware issues with one of the components within the computer.

KERNEL:

A kernel is the core component of an operating system. Using interprocess communication and system calls, it acts as a bridge between applications and the data processing performed at the hardware level. When an operating system is loaded into memory, the kernel loads first and remains in memory until the operating system is shut down again. The kernel is responsible for low-level tasks such as disk management, task management and memory management.

**7. What is Virtualization? Discuss about Vmware/Virtualbox/Xen?**

**Answer:**

In computing, **virtualization** means to create a virtual version of a device or resource, such as a server, storage device, network or even an operating system where the framework divides the resource into one or more execution environments.

VMware is a virtualization and cloud computing software provider based in Palo Alto, California. Founded in 1998, VMware is a subsidiary of Dell Technologies. EMC Corporation originally acquired VMware in 2004; EMC was later acquired by Dell Technologies in 2016. VMware bases its virtualization technologies on its bare-metal hypervisor ESX/ESXi in x86 architecture.

A **VirtualBox** or VB is a software virtualization package that installs on an operating system as an application. **VirtualBox** allows additional operating systems to be installed on it, as a Guest OS, and run in a virtual environment. In 2010, **VirtualBox** was the most popular virtualization software application.

**Xen** is an open source virtual machine monitor for x86-compatible computers. XenSource Inc. and Virtual Iron Software Inc. promoted **Xen** as the primary open source competitor to commercial virtualization products such as VMWare.

**8. What are the Hardware requirements for building a server?**

Answer:

### Minimum hardware requirements

The values below refer to the minimum available hardware required to run Confluence only; for example, the minimum heap size to allocate to Confluence is 1 GB and 1 GB for Synchrony (which is required for collaborative editing). You'll need additional physical hardware, of at least the minimum amount required by your Operating System and any other applications that run on the server.

On small instances, server load is primarily driven by peak visitors, so minimum system requirements are difficult to judge. We provide these figures as a guide to the absolute minimum required to run Confluence, and your configuration will likely require better hardware.

Here is our minimum hardware recommendation:

* CPU: Quad core 2GHz+ CPU
* RAM: 6GB
* Minimum database space: 10GB .

One of the best choices for a small business is a dedicated server built from the ground up as a file server to provide features and expansion options that a desktop computer lacks. Some server hardware decisions you will need to make include the following:

1. Form Factor: For small businesses,  the best choice is a dedicated entry-level server in a tower configuration.   
2. Processor: Choose a server-specific processor to boost performance and data throughput.    
3. Memory: Buy as much memory as you can afford and look for expansion slots for future upgrades.  
4. Storage: Look for SATA or SCSI hard disks, not IDE.

**9. What is a server? How does it differ from Desktop?**

**Answer:**

A server is a computer program that provides services to other computer programs (and their users) in the same or other computers. The computer that a server program runs in is also frequently referred to as a server. That machine may be a dedicated server or used for other purposes as well.

A desktop computer system typically runs a user-friendly operating system and desktop applications to facilitate desktop-oriented tasks.  In contrast, a server manages all network resources. Servers are often dedicated (meaning it performs no other task besides server tasks). Because a server is engineered to manage, store, send and process data 24-hours a day it has to be more reliable than a desktop computer and offers a variety of features and hardware not typically used in the average desktop computer.

**10. Explain various steps involved in the Boot Process?**

**Answer:**

The following are the 6 high level stages of a typical Linux boot process.

### 1. BIOS

* BIOS stands for Basic Input/Output System
* Performs some system integrity checks
* Searches, loads, and executes the boot loader program.
* It looks for boot loader in floppy, cd-rom, or hard drive. You can press a key (typically F12 of F2, but it depends on your system) during the BIOS startup to change the boot sequence.
* Once the boot loader program is detected and loaded into the memory, BIOS gives the control to it.
* So, in simple terms BIOS loads and executes the MBR boot loader.

### 2. MBR

* MBR stands for Master Boot Record.
* It is located in the 1st sector of the bootable disk. Typically /dev/hda, or /dev/sda
* MBR is less than 512 bytes in size. This has three components 1) primary boot loader info in 1st 446 bytes 2) partition table info in next 64 bytes 3) mbr validation check in last 2 bytes.
* It contains information about GRUB (or LILO in old systems).
* So, in simple terms MBR loads and executes the GRUB boot loader.

### 3. GRUB

* GRUB stands for Grand Unified Bootloader.
* If you have multiple kernel images installed on your system, you can choose which one to be executed.
* GRUB displays a splash screen, waits for few seconds, if you don’t enter anything, it loads the default kernel image as specified in the grub configuration file.
* GRUB has the knowledge of the filesystem (the
* older Linux loader LILO didn’t understand filesystem).
* Grub configuration file is /boot/grub/grub.conf (/etc/grub.conf is a link to this). The following is sample grub.conf of CentOS.
* As you notice from the above info, it contains kernel and initrd image.
* So, in simple terms GRUB just loads and executes Kernel and initrd images.

### 4. Kernel

* Mounts the root file system as specified in the “root=” in grub.conf
* Kernel executes the /sbin/init program
* Since init was the 1st program to be executed by Linux Kernel, it has the process id (PID) of 1. Do a ‘ps -ef | grep init’ and check the pid.
* initrd stands for Initial RAM Disk.
* initrd is used by kernel as temporary root file system until kernel is booted and the real root file system is mounted. It also contains necessary drivers compiled inside, which helps it to access the hard drive partitions, and other hardware.

### 5. Init

* Looks at the /etc/inittab file to decide the Linux run level.
* Following are the available run levels
  + 0 – halt
  + 1 – Single user mode
  + 2 – Multiuser, without NFS
  + 3 – Full multiuser mode
  + 4 – unused
  + 5 – X11
  + 6 – reboot
* Init identifies the default initlevel from /etc/inittab and uses that to load all appropriate program.
* Execute ‘grep initdefault /etc/inittab’ on your system to identify the default run level
* If you want to get into trouble, you can set the default run level to 0 or 6. Since you know what 0 and 6 means, probably you might not do that.
* Typically you would set the default run level to either 3 or 5.

### 6. Runlevel programs

* When the Linux system is booting up, you might see various services getting started. For example, it might say “starting sendmail …. OK”. Those are the runlevel programs, executed from the run level directory as defined by your run level.
* Depending on your default init level setting, the system will execute the programs from one of the following directories.
  + Run level 0 – /etc/rc.d/rc0.d/’
  + Run level 1 – /etc/rc.d/rc1.d/
  + Run level 2 – /etc/rc.d/rc2.d/
  + Run level 3 – /etc/rc.d/rc3.d/
  + Run level 4 – /etc/rc.d/rc4.d/
  + Run level 5 – /etc/rc.d/rc5.d/
  + Run level 6 – /etc/rc.d/rc6.d/
* Please note that there are also symbolic links available for these directory under /etc directly. So, /etc/rc0.d is linked to /etc/rc.d/rc0.d.
* Under the /etc/rc.d/rc\*.d/ directories, you would see programs that start with S and K.
* Programs starts with S are used during startup. S for startup.
* Programs starts with K are used during shutdown. K for kill.
* There are numbers right next to S and K in the program names. Those are the sequence number in which the programs should be started or killed.
* For example, S12syslog is to start the syslog deamon, which has the sequence number of 12. S80sendmail is to start the sendmail daemon, which has the sequence number of 80. So, syslog program will be started before sendmail.

**12. What is agile methodology?**

**Answer:**

Not a methodology! The Agile movement seeks alternatives to traditional project management. Agile approaches help teams respond to unpredictability through incremental, iterative work cadences and empirical feedback. Agilists propose alternatives to waterfall, or traditional sequential development.

“Agile Development” is an umbrella term for several iterative and incremental software development methodologies. The most popular agile methodologies include Extreme Programming (XP), Scrum, Crystal, Dynamic Systems Development Method (DSDM), Lean Development, and Feature-Driven Development (FDD).

**13. What is waterfall and discuss the advantages and disadvantages?**

**Answer:**

The Waterfall Model was first Process Model to be introduced. It is also referred to as a linear-sequential life cycle model.  It is very simple to understand and use.  In a waterfall model, each phase must be completed fully before the next phase can begin. This type of [software development model](http://istqbexamcertification.com/what-are-the-software-development-models/) is basically used for the for the project which is small and there are no uncertain requirements. At the end of each phase, a review takes place to determine if the project is on the right path and whether or not to continue or discard the project. In this model [software testing](http://istqbexamcertification.com/what-is-a-software-testing/) starts only after the development is complete. In waterfall model phases do not overlap.

### Advantages of Waterfall Iterative Model

1)  Waterfall model is simple to implement and also the amount of resources required for it are minimal.  
  
2)   In this model, output is generated after each stage (as seen before), therefore it has high visibility. The client and project manager gets a feel that there is considerable progress. Here it is important to note that in any project psychological factors also play an important role.  
  
3)  Project management, both at internal level and client's level, is easy again because of visible outputs after each phase. Deadlines can be set for the completion of each phase and evaluation can be done from time to time, to check if project is going as per milestones.  
  
4)  This methodology is significantly better than the haphazard approach to develop software. It provides a template into which methods of analysis, design, coding, testing and maintenance can be placed.  
  
5)  This methodology is preferred in projects where quality is more important as compared to schedule or cost.

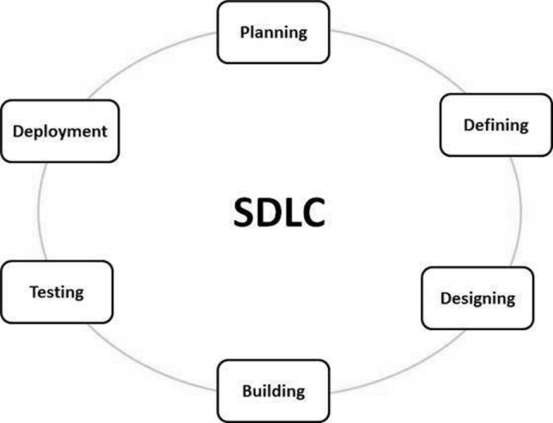
### Disadvantages of Waterfall Iterative Model of SDLC

1)  Real projects rarely follow the sequential flow and iterations in this model are handled indirectly. These changes can cause confusion as the project proceeds.   
  
2)  It is often difficult to get customer requirements explicitly. Thus specifications can't be freezed. If that case arises baseline approach is followed, wherein output of one phase is carried forward to next phase. For example, even if SRS is not well defined and requirements can't be freezed, still design starts. Now if any changes are made in SRS then formal procedure is followed to put those changes in baseline document.  
  
3)  In this model we freeze software and hardware. But as technology changes at a rapid pace,such freezing is not advisable especially in long-term projects.   
  
4)  This method is especially bad in case client is not IT-literate as getting specifications from such a person is tough.  
  
5)  Even a small change in any previous stage can cause big problem for subsequent phases as all phases are dependent on each-other.  
  
6)  Going back a phase or two can be a costly affair.

**14. What is SDLC?**

**Answer:**

Software Development Life Cycle (SDLC) is a process used by the software industry to design, develop and test high quality softwares. The SDLC aims to produce a high-quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates.



Following are the most important and popular SDLC models followed in the industry &miuns;

* Waterfall Model
* Iterative Model
* Spiral Model
* V-Model
* Big Bang Model

Other related methodologies are Agile Model, RAD Model, Rapid Application Development and Prototyping Models.

**15. What are the daily duties of an Administrator?**

### The role of administrator involves a great deal of multitasking. You will work with teams, oversee the operations within your company, manage groups, coordinate with management and engage in planning according to the needs of your company. If there are office resource or administrative issues, you will be the person expected to deal with them. The system administrator is responsible for following things:

1. User administration (setup and maintaining account)
2. Maintaining system
3. Verify that peripherals are working properly
4. Quickly arrange repair for hardware in occasion of hardware failure
5. Monitor system performance
6. Create file systems
7. Install software
8. Create a backup and recover policy
9. Monitor network communication
10. Update system as soon as new version of OS and application software comes out
11. Implement the policies for the use of the computer system and network
12. Setup security policies for users. A sysadmin must have a strong grasp of computer security (e.g. firewalls and intrusion detection systems)
13. Documentation in form of internal wiki
14. Password and identity management

**16.What is a repository in GIT/TFS?**

**Answer:**

Git allows a team of people to work together, all using the same files. And it helps the team cope with the confusion that tends to happen when multiple people are editing the same files. A Git repository, or repo, is a folder that you’ve told Git to help you track file changes in. You can have any number of repos on your computer, each stored in their own folder. Each Git repo on your system is independent, so changes saved in one Git repo don’t affect the contents of another.  
A Git repository, or repo, is a folder that you’ve told Git to help you track file changes in. You can have any number of repos on your computer, each stored in their own folder. Each Git repo on your system is independent, so changes saved in one Git repo don’t affect the contents of another.

**17.What is an active directory in Windows?**

**Answer:**

Active Directory (AD) is a directory service that [Microsoft](https://en.wikipedia.org/wiki/Microsoft) developed for [Windows domain](https://en.wikipedia.org/wiki/Windows_domain) networks. It is included in most [Windows Server](https://en.wikipedia.org/wiki/Windows_Server) [operating systems](https://en.wikipedia.org/wiki/Operating_system) as a set of [processes](https://en.wikipedia.org/wiki/Process_(computing)) and [services](https://en.wikipedia.org/wiki/Windows_service). A server running Active Directory Domain Services (AD DS) is called a [domain controller](https://en.wikipedia.org/wiki/Domain_controller). It [authenticates](https://en.wikipedia.org/wiki/Authentication) and [authorizes](https://en.wikipedia.org/wiki/Authorization) all users and computers in a Windows domain type network—assigning and enforcing security policies for all computers and installing or updating software. For example, when a user [logs into](https://en.wikipedia.org/wiki/Login) a computer that is part of a Windows domain, Active Directory checks the submitted password and determines whether the user is a [system administrator](https://en.wikipedia.org/wiki/System_administrator) or normal user.[[4]](https://en.wikipedia.org/wiki/Active_Directory#cite_note-adw2k3-4) Also, it allows management and storage of information, provides authentication and authorization mechanisms, and establishes a framework to deploy other related services

**18.What is LDAP in Linux?**

**Answer:**

LDAP stands for Lightweight Directory Access Protocol. As the name suggests, it is a lightweight client-server protocol for accessing directory services,A directory is similar to a database, but tends to contain more descriptive, attribute-based information. The information in a directory is generally read much more often than it is written. Directories are tuned to give quick-response to high-volume lookup or search operations. They may have the ability to replicate information widely in order to increase availability and reliability, while reducing response time. When directory information is replicated, temporary inconsistencies between the replicas may be OK, as long as they get in sync eventually.

**19. What is Devops?**

**Answer:**

DevOps is a term for a group of concepts that, while not all new, have catalyzed into a movement and are rapidly spreading throughout the technical community.  Like any new and popular term, people have somewhat confused and sometimes contradictory impressions of what it is.  Here’s my take on how DevOps can be usefully defined; I propose this definition as a standard framework to more clearly discuss the various issues DevOps covers. Like “Quality” or “Agile,” DevOps is a large enough concept that it requires some nuance to fully understand.

**20.Explain about different phases DEVELOPMENT/TESTING/PRODUCTION?**

**Answer:**

**There are following g six phases in every Software development life cycle model:**

* Requirement gathering and analysis.
* Design.
* Implementation or coding.
* Testing.
* Deployment.
* Maintenance.

**Below are the phases of STLC:**

* Requirements phase.
* Planning Phase.
* Analysis phase.
* Design Phase.
* Implementation Phase.
* Execution Phase.
* Conclusion Phase.
* Closure Phase.

**The three main stages of production are:**

* Pre-production: Planning, scripting & storyboarding, etc.
* Production: The actual shooting/recording.
* Post-production: Everything between production and creating the final master copy.

**21. What is Root and Sudo?**

**Answer:**

**root** is the user name or account that by default has access to all commands and files on a **Linux** or other Unix-like operating system. It is also referred to as the **root** account, **root** user and the superuser.

The sudo command is to execute one command with root permissions. It is very configurable. Some distribution let the first user of the system to execute with sudo everything. The entering own password is optional.

**22.What is Automation in regards to Linux?**

**Answer:**

Automation is essential to running Linux in the enterprise effectively. Automation lets you minimize costs by reducing manual operations, helps ensure compliance across the data center, standardizes your software infrastructure and accelerates deployments for your bare-metal and cloud infrastructures.

**23. Discuss about GREP Command in Linux? Provide 15 Use Case Scenarios.**

**Answer:**

GREP is a powerful tool that searches for matching a regular expression against text in a file, multiple files or a stream of input. It searches for the pattern of text that you specify on the command line and prints output for you.

Searching a string in a files

$ grep “samplestring” samplefile.txt

Insensitive case search

$ grep -i “samplestring” samplefile.php

Searching for a string in multiple files

$ grep “sample string” file\*.\*

Specifying a string as a regular expression

$ grep “sample.\*host” samplefile.html

To display line numbers of a word

$ grep -n “sample string” samplefile.html

Highlighting a string

$ grep --color “sample string” samplefile.html

Printing a line excluding a pattern

$ grep -v “sample string” /etc/password

To display all the lines that starts with a pattern

$ grep ^samplestring /sampleFolder/Sample.txt

Display all the lines ends with a pattern

$ grep sample$ samplefolder/sample.txt

Recursive searching

$ grep -r “sample string” /etc/

Counting the lines when words match

$ grep -c “sample text” /home/sample.txt

**24. Discuss about FIND Command in Linux? Provide 25 Use Case Scenarios?**

**Answer:**

#### The Linux ****Find Command**** is one of the most important and much used command in Linux sytems. Find command used to search and locate list of files and directories based on conditions you specify for files that match the arguments. Find can be used in variety of conditions like you can find files by ****permissions****, ****users****, ****groups****, ****file type****, ****date****, ****size**** and other possible criteria. List all files in current and sub directories

List all files in a Directories and Sub-directories

$ find

.

./abc.txt

./cool.php

Search file by name

$ find . -name samplefile.txt

Searching file by their extensions

$ find -name “\*.php”

Searching the filename under a Directory

$ find /home -name samplefile.txt

Insensitive search for a file

$ find -iname samplefile.txt

Searching names that doesn’t match a pattern

$ find . -not -name “\*.txt”

To display the empty files

$ find . -empty

Searh files belonging to a particular group

$ find . -group Direcotryname -name “\*.html”

To find files owned by particular user

$ find . -user username -name “\*.txt”

To display directories in output

$ find -type d

Searching files based on their last access time

Find -amin 1 -name “\*.txt”

To searh recently modified files

$ find . -mmin 1 -name “\*.txt”

Finding files modified Ndays back

$ find . -mtine 50

Files accessed in last N days

$ find . -atime 50

Files modified in a range of days

$ find . -mtime +25 -mtime -100

Files modified inlast hour

$ find . -mmin 60

Finding files of 50MB

$ find . -size 50M

Finding files of size range

$ find . -size +50M -size -100M

Find all hidden files

$ find . -type f -name “.\*”

**25. Discuss about SED Command in Linux? Provide 10 Use Case Scenarios?**

**Answer:**

**sed** is a *stream editor*. A stream editor is used to perform basic text transformations on an input stream. While in some ways similar to an editor which permits scripted edits, **sed** works by making only one pass over the input(s), and is consequently more efficient. But it is **sed**'s ability to filter text in a pipeline which particularly distinguishes it from other types of editors.

A common use of [sed](https://www.lifewire.com/example-uses-of-sed-2201058) is to modify each line of a file or stream by replacing specified parts of the line.

To show a specific range of files

$ *sed -n ‘5,10p’ file.txt*

To view all lines except a given range

$ *sed ’20,35d’ file.txt*

View non consecutive lines and ranges

$ *sed -n -e ‘5,10p’ -e ’20,30p’ file.txt*

Replacing characters or words

$ *sed ‘s/first1/first2/g’ file.txt* ( global replacement)

$ *sed ‘s/first1/first2/gi’ file.txt* ( global replacement with ignoring character case)

Replacing characters inside a range

$ *sed ‘5,10 s/first1/first2/g’ myfile.txt*

To remove empty lines

$ *sed ‘/^$/d’* *file.txt*

To remove lines which has a pattern

$ *sed ‘/word/d’ file.txt*

To replace a word which starts with small or capital letter

$ *sed ‘/[Zz]ip/not/g’ file.txt*

List all the lines which starts with a pattern

$ sed -n ‘/^jul/p’ file.txt

Inserting spaces in files

$ *sed G file.txt*

*$ sed ‘G;G’ myfile.txt*

Performing two or more substitutions at once

$ *sed -i ‘s/this/that/gi;s/linw/wine/gi’ file.txt*

Replace a word if a separate match is found

$ *sed ‘/service/ s/first1/first2/g’ file.txt*

**26. Discuss about various commands used in “VI EDITOR”? Explain each case with individual scenarios?**

**Answer:**Vi is a powerful text editor included with most Linux systems, even embedded ones. Sometimes you’ll have to edit a text file on a system that doesn’t include a friendlier text editor, so knowing Vi is essential.

Creates a new file

$ vi filename

Open an existing file in read-only mode

$ vi -R filename or $ view filename

~ represents an unused line

:q is used to quit the command mode

:w is used to save the contents of the editor

:wq is used to save and quit command and return

ZZ is same as the :wq

To move in the file you need to be in command mode by pressing Esc twice

K move cursor up oneline

J move cursor downone line

H move cursor to the left

L moves the cursor to the right

Adding a numerical before h,j,k,l, will move that many times.

Inserts text before the cursor – i

Inserts text at the beginning of current line – I

Inserts text after the current cursor location – a

Inserts text at the end of the current line – A

Creates a new line fortext entry below the cursor – o

Creates a new line for text entry above the cursor – O

**Deleting Characters**

Delete a character under the cursor location – x

Deletes the character before the cursor location – X

Deletes the current cursor location to the next word – dw

Deletes the current cursor position to the beginning of the line – d^

Deletes the current cursor position to the end of the line – d$

Deletes from cursor position to the endof current line – D

Deletes the line the cursor is on -dd

**Change commands:**

Remove the contents of the line, leaving you in insert mode – cc

Change the word the cursor is on from the cursor to the lowercase w endof the word – cw

Replaces the haracter under the cursor. Vi returns to the command mode after the replacement is entered – r

Overwrites multiple characters beginning with the character currently under the cursor. You must Esc to stop the overwriting – R

Replaces the current haracter with the character you type. Afterward, you are left in in the insert mode – s

Deletes the line the cursor is on and replaces it with the new tex. After the new text is entered, vi remain in the insert mode – S

**Copy and Paste Commands**

yy- copies the current line

yw- copies the current word from the character the lowercase w cursor is on , until the end of the word.

p- Puts the copied text after the cursor

P-Puts the yanked text before the cursor

**27. Discuss about AWK Command in Linux? Provide 10 Use Case Scenarios?**

**Answer:**

Awk is a programming language which allows easy manipulation of structured data and the generation of formatted reports. Awk stands for the names of its authors “**A**ho, **W**einberger, and **K**ernighan”. The Awk is mostly used for pattern scanning and processing. It searches one or more files to see if they contain lines that matches with the specified patterns and then perform associated actions.

Basic format of awk command looks like

*awk ‘pattern {action}’ input-file > output-file*

To print all the lines of a document

*$ awk ‘{print;}’ file.txt*

To print specific columns of a file

$ *awk ‘{print $1,$2;}’ file.txt*

To print a matching pattern

$ *awk ‘/pattern/’ file.txt*

$ *awk ‘/pattern/ {print}’ file.txt*

$ *awk ‘/302/{print}’ file.txt*

To print a specific line of a matching pattern

$ *awk ‘/pattern/{print $4}’ file.txt*

To print something >,< a number

$ *awk ‘$4>100’ file.txt*

$ *awk ‘$2 <20’ file.txt*

To display words starts with a pattern

$ *awk ‘/^ra/{print}’ file.txt*

To match the pattern by range

$ *awk ‘/[a-f]/{print}’ file.txt* (prints all if it matches letters from a to f)

To display words which matches even a single word

$ *awk ‘/[ab1]/{print}’ file.txt*  (It will every line if it has a or b or 1)

To display words followed by a pattern

$ *awk ‘/[ab]cc/{print}’ file.txt* (Prints line which has patterns acc or bcc)

To display words starts with a pattern end ends with a pattern

$ *awk ‘/la\*al/{print}’ file.txt* (Prints which starts with la and ends with al)

To awk(.) wild card in a pattern

$ *awk ‘/l.c/{print}’ file.txt* ( . is replaced by single character)

To Display the number of columns in the file

$ awk ‘{print NF}’ samplefile

Display the Total number of lines in the file

$ awk ‘END {print NR}’ samplefile

This will display the line numbers from 1

$ awk ‘{print NR}’ samplefile

**28. Discuss about various commands to give appropriate permissions in Linux?**

**Answer:**

Chmod is used to change the file access permissions. rwxrwxrwx is the common way of represnting permissions of a filename. r stands for read w stands for write and x stands for executable rights. First rwx is for the sudo user, second rwx are for the group permissions and third rwx is for the regular user permissions.

Rwx is represented in octal binary so it ranges from 7to0.

Example: chmod 740 sample.txt

7 = 4+2+1 (read/write/execute)

6 = 4+2 (read/write)

5 = 4+1 (read/execute)

4 = 4 (read)

3 = 2+1 (

write/execute)

2 = 2 (write)

1 = 1 (execute)

Adding single permission to a file or directory

$ chmod u+x filename

Adding multiple permissions

$ chmod u+r,g+x,o+w samplefile

Remove permission

$ chmod u-r samplefile

Remove multiple permission

$ chmod u-x,g-w samplefile

Change permissions to all the files

$ chmod a+x samplefile

Make permission of a file with a reference

$ chmod –reference=file1 file2 (file 1 permission will be mirrored to file2)

Apply permissions recursively

$ chmod -R 724 /File1

**29. Discuss various scenarios of how to create users and groups?**

**Answer:**

Adding a new user in Linux

1. useradd samplename or adduser sampleuser

passwd sample

2. Creating a new user under different directory other than home  
useradd -d /data/projects samplename

3. New user with a UID

useradd -u 347 samplename

4. Creating a new user with Group ID

useradd -u 999 -g 401 samplename

5. Adding a user to multiple groups

useradd -G group1,group2,group3,group4 sampleuser

6.Adding user without Home directory

useradd -M samplename

7. User with expiry date

useradd -e yyyy-mm-dd sampleuser

change -l sampleuser (to verify age of account and password)

8. Create a user with password expiry date

useradd -e yyyy-mm-dd -f 50 sampleuser

50 is the number of days for the password expiration

**30. Explain the command “ls” and discuss about all the columns displayed during “ls –a”?**

**Answer:**

ls stands for list in Linux. List information about the FILEs (the current directory by default).Sort entries alphabetically if nothing is specified

ls -a shows the hidden files.

In linux hidden files starts with “.” .

.. is the parent directory

.bash\_history conatains all the commands that have been tried in the terminal.

.bash\_logout- The file *.bash\_logout* is read and executed every time a login shell exits.,

.bashrc- is shell script that Bash runs whenever it is started.

.Xauthority- The .Xauthority (not .xAuthority ) file can be found in each user home directory and is used to store credentials in cookies used by xauth for authentication of X sessions.

.xsession-errors-It's the error log produced by your X windows system (which the Ubuntu GUI is built on top of). Other installed packages like package installers, packages like mozilla,gnome and the normal directories Music, Pitures, Videos, Templates, Desktop, Documents etc.,

**31. Network basics and Linux Network commands( ip,ipv4,ipv6,subnet masking, ifconfig, netstat, etc)**

**Answer:**

Ifconfig -Interface configurator commandis used to initialize an interface, assign IP address to interface and enable or disable interface or demand.

$ ifconfig eth0 192.168.0.5 netmask 255.255.255.0 enables the IP address and gateway to interface. These settingswill be removed on reboot

To eth0 enable $ ifup eth0

To disable eth0 $ ifdown eth0

Setting the MTU size

$ ifonfig eth0 mtu 1400

To enter promiscuous mode

$ ifconfig etho – promisc

PING stands for Packet Internet Groper

$ ping [www.google.com](http://www.google.com)

Traceroute command

$ traceroute 192.168.0.1

NETSTAT command

$ netstat -r (-r is used for displaying routing table)

DIG Command (Domain Information Groper) gives Record, CNAME, MX Record. This is used for DNS troubleshooting

$ dig ww.google.com

NSLOOKUP command- nslookup is a DNS related query

$ nslookup [www.google.com](http://www.google.com)

HOST command

Host command is used to find name to IP name in IPv4 or IPv6 and also DNSrecords

$ host [www.google.com](http://www.google.com)

ARP ommand – Address Resolution Protocol is used view/add ontents to ARP tables

$ arp -e

ETHTOOL command- This is used to view, setspeed and duplexof your NIC.

$ ethtool eth0

IWCONFIG is used to configure a wireless network interface

$ iwconfig [interface]

GUI tool system-config-network

$ system-config-network

**32. Linux performance**

**Answer:**

Linux performance depends mainly on the CPU, Memory, I/O,Network  
CPU:

#### Context Switch

* When CPU switches from one process (or thread) to another, it is called as context switch.
* When a process switch happens, kernel stores the current state of the CPU (of a process or thread) in the memory.
* Kernel also retrieves the previously stored state (of a process or thread) from the memory and puts it in the CPU.
* Context switching is very essential for multitasking of the CPU.
* However, a higher level of context switching can cause performance issues.

#### Run Queue

* Run queue indicates the total number of active processes in the current queue for CPU.
* When CPU is ready to execute a process, it picks it up from the run queue based on the priority of the process.
* Please note that processes that are in sleep state, or i/o wait state are not in the run queue.
* So, a higher number of processes in the run queue can cause performance issues.

#### Cpu Utilization

* This indicates how much of the CPU is currently getting used.
* This is fairly straight forward, and you can view the CPU utilization from the top command.
* 100% CPU utilization means the system is fully loaded.
* So, a higher %age of CPU utilization will cause performance issues.

#### Load Average

* This indicates the average CPU load over a specific time period.
* On Linux, load average is displayed for the last 1 minute, 5 minutes, and 15 minutes. This is helpful to see whether the overall load on the system is going up or down.
* For example, a load average of “0.75 1.70 2.10” indicates that the load on the system is coming down. 0.75 is the load average in the last 1 minute. 1.70 is the load average in the last 5 minutes. 2.10 is the load average in the last 15 minutes.
* Please note that this load average is calculated by combining both the total number of process in the queue, and the total number of processes in the uninterruptable task status.

Memory:

* As you know, RAM is your physical memory. If you have 4GB RAM installed on your system, you have 4GB of physical memory.
* Virtual memory = Swap space available on the disk + Physical memory. The virtual memory contains both user space and kernel space.
* Using either 32-bit or 64-bit system makes a big difference in determining how much memory a process can utilize.
* On a 32-bit system a process can only access a maximum of 4GB virtual memory. On a 64-bit system there is no such limitation.
* The unused RAM will be used as file system cache by the kernel.
* The Linux system will swap when it needs more memory. i.e when it needs more memory than the physical memory. When it swaps, it writes the least used memory pages from the physical memory to the swap space on the disk.
* Lot of swapping can cause performance issues, as the disk is much slower than the physical memory, and it takes time to swap the memory pages from RAM to disk.

I/O:

* I/O wait is the amount of time CPU is waiting for I/O. If you see consistent high i/o wait on you system, it indicates a problem in the disk subsystem.
* You should also monitor reads/second, and writes/second. This is measured in blocks. i.e number of blocks read/write per second. These are also referred as bi and bo (block in and block out).
* tps indicates total transactions per seconds, which is sum of rtps (read transactions per second) and wtps (write transactions per seconds).

Network:

* A good understanding of TCP/IP concepts is helpful while analyzing any network issues. We’ll discuss more about this in future articles.
* For network interfaces, you should monitor total number of packets (and bytes) received/sent through the interface, number of packets dropped, etc.,

Linux Performance commands:

$ top – It is performance monitoring program to monitor linux performance. It displays all the running real-time process in ordered list and updates regularly. CPU usage, Memory usage, swap memory, Cache size, Buffer size, Process PID, User, Commands etc.,

$ vmstat – Vmstat command is used to display statitics of virtual memory, kernel threads, disks, system processes, I/O blocks, interrupts, CPU activity etc.,

$ Lsof – This open diskfiles, network, sockets, pipes, devices and processes. Whenever a disk cannot be unmounted because of reason that some file isopened running. IIn that moment we can open th list of open files and the specific app.

$ tcpdump -i eth0 – It is a network packet analyzer or packet sniffer which is used to capture TCP/IP packets. It also provides an option to save the packets in a file.

$ htop – htop is advanced interactive and real time linux process monitoring tool . It has GUI to manage process, shortcut keys, vertical and horizontall view of the processes etc., Htop is noy builtin, you should install it separately.

$ iotop – This has the function to monitor and display realtime Disk I/o andprocesses. This command is used to find the exact process and high used disk read/write of the processes.

$ iostat – iostat is simple tool that will collect and show system input and output storage device performance issues including devices,local disks, remote disks such as NFS.

$ monit – Monit is a free open source and web based process utility that automatically monitors and managers system processes, programs, files, directories,permissions,checksums and file systems. It also monitors services like Apache, MySQL, Mail, FTP, ProFTP, Nginx, SSH and so on

$ nethogs – NetHogs is an open source nice small program that keeps a tab on each process network activity on your system. It also keeps a track of real time network traffic bandwidth used by each program or application.

$ iftop – iftop is another open source system monitoring utlity that displays a frequently updated list of network bandwidth utilization that’s passing through network interface on your system. Iftop is for network usage and top is for CPU usage

$ monitorix – Monitorixisutility that is designed to run and monitor system and network resourcesas many as possible servers. It has builtin HTTP web server that collects system and network information and display in graphics. It monitors system load average, memory allocation, disk driverhealth, system services, network ports, mail statitics etc., It helps to monitor overall system performance and helps in detetcting failures, bottlenecks, abnormal activities etc.,

**33. RPM and YUM**

**Answer:**

RPM - **RPM** (**Red Hat Package Manager**) is an default open source and most popular package management utility for **Red Hat** based systems like (**RHEL**, **CentOS** and **Fedora**). The tool allows system administrators and users to **install**, **update**, **uninstall**, **query**, **verify** and manage system software packages in **Unix/Linux** operating systems. The RPM formerly known as **.rpm** file, that includes compiled software programs and libraries needed by the packages. This utility only works with packages that built on **.rpm** format.

YUM - **YUM** (**Yellowdog Updater Modified**) is an open source command-line as well as graphical based package management tool for RPM (RedHat Package Manager) based **Linux** systems. It allows users and system administrator to easily install, update, remove or search software packages on a systems.

**34. RPM and YUM cheat sheet**

**answer:**

RPM cheatsheet

$ rpm -ivh {rpm-file} ------------------🡪 Install the package

$ rpm -Uvh {rpm-file} ------------------🡪 Upgrade package

$ rpm -ev {package} ------------------🡪 Erase/ Remove/ an installed package

$ rpm -ev - - nodeps {package} ------------------🡪 Erase/Remove/ an installed package without checking or dependecies

$ rpm -qa ------------------🡪 Display list all installed packages

$ rpm -qi {package} ------------------🡪 Display installed information along with the package version and short description

$ rpm -qf {/path/to/file} ------------------🡪 Find out what package a file belongs to

$ rpm -qc {package-name} ------------------🡪 Display list of configuration files for a package

$ rpm -qcf {/path/to/file} ------------------🡪 Display list of config files for a command

$ rpm -qa --last ------------------🡪 Display list of all recently installed RPMs

$ rpm -qpR {.rpm-file} or rpm -qR {package} ------------------🡪 Find out what dependencies a rpm file has

YUM cheatsheet

$ --y ------------------------------🡪Assume yes for all prompts

$ --assumeno ------------------------------🡪Assume no for all prompts

$ --nogpgcheck------------------------------🡪Disable GPG verification

$ -- skip-broken ------------------------------🡪Skip broken packages

$ enable-repo=[path]------------------------------🡪Enable repository

$ disable -repo=[path] ------------------------------🡪Disable repository

$sudo yum install [package-name ] ------------------------------🡪Installing rpm package

$ sudo yum install /path/to/vinn-1.2.rpm ------------------------------🡪Installing a local RPM package

$ sudo yum install packagename-3.6 ------------------------------🡪Installing a specific version of a package

$ sudo yum remove [package-name] ------------------------------🡪Removing an RPMpackage and its dependencies

$ sudo yum downgrade [package name ]------------------------------🡪Downgrading apackage

$ sudo yum deplist [package-name ]------------------------------🡪View packages dependencies

$ sudo yum list avaialble------------------------------🡪List all available packages

$ sudo yum list installed ------------------------------🡪List installedpackages

$ sudo yum list all ------------------------------🡪List installed andavaialble packages

$ sudo yum search ------------------------------🡪Searh for a package

$ sudo yum upgrade------------------------------🡪Upgrade all system packages

$sudo yum reinstall [package name] ------------------------------🡪Reinstall a single package

$ sudo yum info [package-name] ------------------------------🡪View info for a package

$ sudo yum repolist ------------------------------🡪 List all repositories

$ sudo yum repo-pkgs [repository] install ------------------------------🡪Install packages from a repo

$ sudo yum makecache ------------------------------🡪 Update local meta cahce

**35. Apt-get**

**Answer:**

APT stands for Advanced Packaging Tool. apt-get is the command line tool for working with APT software packages.It is a rapid, practical, and efficient way to install packages on your system. Dependencies are managed automatically, configuration files are maintained, and upgrades and downgrades are handled carefully to ensure system stability.   
Few commands related to apt-get are:

To install a new package

apt-get install <package-name>

This command searches the repositoriesandinstalls the nuild dependenies for the respectivepackafe

apt-get build-dep <package-name>

To install multiple packages with a single line ommand we use

apt-get install <package1> <package2> <package3>

apt-get update checks for the sources whether they are uptodate or not.

Apt-get upgrade whixh upgrades all installed packages.

Apt-get remove <package name> is used to remove the package.

The ‘clean‘ command is used to free up the disk space by cleaning retrieved (downloaded) .deb files (packages) from the local repository. Apt-get clean

**36. PS commands**

**Answer:**  
The ps command is used to provide information about the currently running processes including their process identification numbers(PIDs). Every process is assigned a unique PID by the system itself.

The four items are labeled PID, TTY, TIME and CMD.   
TTY is a placce where the running command runs.  
PID is a Process ID of the running command (CMD)  
TIME tells about how much time is used by the CPU while running the command  
CMD is a command that run as current process.  
ps is most often used to obtain the PID of a malfunctioning process in order to terminate it with the *kill* command. For example, if the PID of a *frozen* or *crashed* program is found to be 1125, the following can usually terminate the process:  
 kill 1125  
Its a commonly used example with a ps command to list down all the process which are currently running in a machine.  
  
$ ps -ef  
-e to display all the processes.  
-f to display full format listing.

To see all the processes in hierarchical form we use the commands $ps axjf or pstree  
If we need to know the thread of a particular process, we can use -L option followed by its Process ID (PID). Here’s an example of -L option in action  
ps -L PID

 we can use -C option followed by the keyword. Let say, we want to show processes named getty. We can type :

ps -C name

Filter processes by CPU or memory usage

Another thing that you might want to see is filter the result by CPU or memory usage. With this, you can grab information about which processes that consume your resource. To do this, we can use aux options. Here’s an example of it :

ps -aux | less  
Sort by the highest CPU utilization in ascending order  
ps -aux --sort -pcpu | less  
Sort by the highest Memory utilization in ascending order  
ps -aux --sort -pmem | less

**37. SSH concept**

**Answer:** SSH stands for Seure Shell which allows a person to connect to a different computer to gain terminal access to it. SSH doesn’t give you the access graphical desktop environment but will give you the terminal access. To check whether your system has SSH or not by typing “ssh -V”. It will display the SSH client you are running and its corresponding version number. If it doesn’t give any info then we should install the SSH. We use “*yum -y install openssh-server openssh-clients*”

most of these methods (Telnet, rsh and X, to name a few) send everything over the network in clear text, including passwords. In order to connect to a computer, you need to have the username and the corresponding IP address and you need to enter the password in order to get the access. openSSH uses the port22by default. In order to change the port we can change in the sshd\_confg file and search for port directive and change it.

The commands we used to login are: *ssh username@ipaddress* (In place of ipaddress we can use a domain also)

If the username is same on both the machines we can use *ssh domain.com*

To restart ssh we use *sudo etc/init.d/ssh restart*

To list the files on a server we use *ssh username:ipaddress “ls”*

To copy files from remote host to local host

Localhost$ scp [username@remotehost.eaxple.com:/home/jsmith/remotefile.txt remotefile.txt](mailto:username@remotehost.eaxple.com:/home/jsmith/remotefile.txt%20remotefile.txt)

Localhost$ scp localfile.txt [jsmith@remotehost.example.com:/home/jsmith/file.txt](mailto:jsmith@remotehost.example.com:/home/jsmith/file.txt)

To troubleshoot a SSH client we need to debug it. For this we use

localhost$ ssh -l jsmith remotehost.example.com (without debug message)

localhost$ ssh -v -l jsmith remotehost.example.com (with debug message)

Sometimes you need change frequently in between remote host and local hosts. In that moment we use ~ and Ctrl-Z and press enter.

**38. LVM:**

**Answer:**

LVM is a tool for logical volume management which includes allocating disks, striping, mirroring and resizing logical volumes. LVM comprises of few conceptual layers such as physical volume, logical volume and file systems.

Physical Volume(PV): Each Physical Volume can be a disk partition, whole disk, meta-device, or a loopback file. Use the command pvcreate to initialize storage for use by LVM. Initializing a block device as physical volume places a label at the start of the device.

Creating Physical Volumes

The pvcreate command is used to initialize the physical volume for use by LVM. Before creating the PV, make sure the disk is visible in the OS. To scan the block devices to be used as PVs, use the lvmdiskscan command. We use the commands pvdisplay, pvs and pvscan to display the PVs we just created.  
  
Logical Volume (LV)

A Logical Volume is the conceptual equivalent of a disk partition in a non-LVM system. Logical volumes are block devices which are created from the physical extents present in the same volume group. We use the command lvcreate to create a logical volume in an existing volume group.  
Create Logical Volume

The Logical volume can now be created in the VG using the lvcreate command. If you do not specify the LV name in the command, by default the LV is given the name lvol#. Normally if you do not specify which PV to span the LV, Logical volume will be created on the PV on a next-free basis.

– To create a logical volume lvol01 of size 5 GB :

# lvcreate -L 5G -n lvol01 vg01

Logical volume “lvol01” created

File system

File systems are built on top of logical volumes. The command mkfs can be used to create file system on top of a logical volume. Once the file system is created we can mount the logical volume as per our need.

Creating File system

The final step is to create a file system on the new LV and mount it on a directory to be able to access it and store data in it. The command mkfs can be used to create file system on top of the LV.  
  
Create a Volume Group

Use the vgcreate command to create the new Volume Group vg01 using a few PVs. We can specify the extents with -s option and maximum number of PVs and LVs in the VG by using the options -p and -l respectively. All these option are optional and need not be necessarily used. The commands vgs and vgdisplay can be used to display the information about the VG we just created  
  
The logical volume can be mounted, once the file system is created. Make sure to add an entry to /etc/fstab, so that it is mounted automatically when the system boots.There is also a cool graphical tool available for the LVM which is system-config-lvm.

**39. Kickstart:**

**Ans:**Kickstart is a method in which the administrator creates a single file which contains a few instructions for automated installation of a typical OS on multiple machines making it good for network and system administrators. Kickstart configuration files can be built three ways:  
By hand, GUI system-config-kickstart tool and using the standard Red Hat installation program Anaconda which will produce an anaconda-ks.cfg configuration file at the end of any manual installation. This file can be used to automatically reproduce the same installation or edited (manually or with system-config-kickstart).

**40. Bonding**

**Ans:**Bonding is one of the Linux kernel feature that allows to aggregate multiple network interfaces (such as eth0, eth1) into a single virtual link such as bond0. The idea is pretty simple, get higher data rates and as well as link failover. Linux allows binding of multiple network interfaces into a single channel/NIC using special kernel module called bonding.

**Usermod:**

To add user info

$ usermod –c "User info" username

To change a user home directory

$ usermod –d 'file1/folder/' username

Set user acount expiry date

$ usermod –e yyyy-mm-dd username

Change user primary group

$ usermod –g groupname username

Adding group to an exisiting user

$ usermod –G group username

Change user login name

$ userdmod –l username-admin username

Lock a user account

$ usermod –L username

Unlock a user account

$ usermod –U username

Create Un-encrypted password for a user

$ usermod –p password username

Change user shell

$ usermod –s /bin/sh username

Change user id

$ usermod –u 543 username

Changing UID and GIO of a user

$ usermod –u 546 –g 567 username

**Userdel:**

**Immutable bit:** A file with an immutable attribute cannot be modified, deleted, renamed and no soft or hard link can created by anyone inluding root user. We use chattr command to set the immutable bit.

# chattr +i file (+iadds immutable bit, - i removes immutable bit)

# chattr +i /path/to/filename

# lsattr is used to list the attributes

Appending data to a immutablebit file

# chattr +a file (Everyone can only append the data without any modifications)

To secure the entire diretory

# chattr –R -i folder (-R recursive)

**Sticky bit:** A Sticky bit is a permission bit that is set on a file or a directory that lets only the owner of the file/directory or the root user to delete or rename the file. No other user is given privileges to delete the file created by some other user.

To add stickybit on a directory we use –t flag of chmod

# chmod +t folder

To remove the sticky bit

# chmod -t folder

**Groupadd:**

**Chage:** Chage stands for change age. This command is used to change users passwords expiry information.

To display password expiry info of a user

$ chage –l username

To disable password aging for a user

$ chage –l –1 –m 0 –M 99999 –E –1 username

-l –1--> This will set password inactive to never

-m 0 --> This will set minimum numberof days between password change to 0

-M 99999 --> This will set maximum number of days between password change to 99999

-E –1 --> This will set "account expires" to never

Enable password expiry date of a user

$ chage –M 20 username

Set the account expiry date

$ chage –E "2015-05-28"

Set the password expiry warning message

$ chage -W 10 username

Forcing users to change the password on next logon

$ chage –d 0 testuser

**Setfacl:**

**Getfacl:**

**Setuid:** SUID (Set owner User ID up on execution) is a special type of permission given to a file. SUID is defined as giving temporary permissions to a user to run a program file with permissions of the file owner rather that user who runs it. In simple words users will get the file owners permissions as well as owner UID and GID when executing a file/program/command.  
$ chmod u+s username (replace u with g for group and o for others)

1) Where root login is required to execute some commands/programs/scripts.

2) Where you don’t want to give credentials of a particular user, but want to run some programs as the owner.

3) Where you don’t want to use SUDO command, but want to give execute permission for a file/script etc.

**Getuid:** This is used to get the user identity.

**Softlink:** Softlink is also called as symlink or a symbolic link is a file that links to another file or directory using a path. In this links are created with the ln command.

$ ln –s /path/to/original/file /path/to/soft/link/file

**Hardlink:** Hard link nothing but the mirror copy of the original file at the other location. Hard links can also be created for other hardlinks. All hardlinks point to the same inode.

$ ln /path/to/original/file /full/path/to/hard/file

To create a hardlink for a File

$ ls –l file.txt

**Wget:**

Wget is utility which retrieves files from World Wide Web (WWW) using the protocols like HTTP, HTTPS and FTP.

To install wget

$ yum –y install wget

To download a single file

$ wget <http://ftp.gnu/wget/wget-1.5.tar.gz>

Download file with a different name

$ wget –O wget.zip <http://ftp.gnu.com/wget-hisfd.tar.gz>

Download multiple files with http and ftp protocols

$ wget <http://ftp.gnu.com/file.zip> ftp://ftp.gnu.org/file2.zip

To donwload few files whose urls stored in a textfile

$ wget -i/home/username/Desktop/file.txt

To resume an un ompleted download

$ wgt –c [**http://mirrors.hns.net.in/centos/6.3/isos/x86\_64/CentOS-6.3-x86\_64-LiveDVD.iso**](http://mirrors.hns.net.in/centos/6.3/isos/x86_64/CentOS-6.3-x86_64-LiveDVD.iso) (-c to download bigfiles)

Download files in bakground

$ wget –b /wget/log.txt <ftp://ftp.iinet.au/debian.zip>

To restrict download speeds

$ wget –c - - limit-rate=100k /wget/log.txt <ftp://ftp.linet.net/debian.zip>

To know the version of wget

$ wget - -version

$ wget --help

**Curl:** cURL is a software package which consists of command line tool and a library for transferring data using URL syntax.

To get the contents of a url

$ curl <http://www.centos.org/>

To store the ouput to a different file

$ curl <http://ww.google.com> > centos.txt

Using –o or –O will save the content to a file

$ curl –O <http://www.gnu.org/gettext.html> (saved with the same name)

$ curl -o mytext.html http//www.gnu.ord/mytext.html

To fetch multiple files at a time

$ curl –O url1 –O url2