LINUX REHAT PROJECT

TOPIC

**IMPLIMENTING LINUX SERVER ADMINISTRATION IN A COMPANY**

**ABSTRACT**

* A Linux Server Administration proposal has to be developed for Networkz systems.
* Assigning IP address to server and client.
* Assigning DNS to server and client.
* Directory sharing using Network File Sharing.
* Changing primary group of a user to other.
* Assigning permissions to a directory for different users and changing directory group and adding users to it.
* Applying Raid 5 for backup.
* Applying Logical Volume Manager.

**AIM**

* Implementing Linux server administration in Networkz Systems.
* To establish Linux server administration in Kollam branch.

**PROCEDURES**

* Start a new project
* Design the requirements
* Insert the components
* Os Installation
* Applying the settings and functions
* Applying the network setup
* Check whether all the setting, functions and network is established

**THEORY**

**DOMAIN NAME SYSTEM (DNS):**

The domain name system (DNS) is the way that internet [domain names](https://searchwindevelopment.techtarget.com/definition/domain-name) are located and translated into internet protocol ([IP](https://searchunifiedcommunications.techtarget.com/definition/Internet-Protocol)) addresses. The domain name system maps the name people use to locate a website to the [IP address](https://searchwindevelopment.techtarget.com/definition/IP-address) that a computer uses to locate a website.

* It resolves user friendly domain names into computer friendly ip address.
* It also resolves ip addresses into domain names.
* It provides a centralized database for resolution.

**NETWORK FILE SYSTEM (NFS):**

The Network File System (NFS) is a [client](https://searchenterprisedesktop.techtarget.com/definition/client)/[server](https://whatis.techtarget.com/definition/server) [application](https://searchsoftwarequality.techtarget.com/definition/application) that lets a computer user view and optionally store and update [files](https://whatis.techtarget.com/definition/file) on a remote computer as though they were on the user's own computer. The NFS [protocol](https://searchnetworking.techtarget.com/definition/protocol) is one of several [distributed file system](https://searchwindowsserver.techtarget.com/definition/distributed-file-system-DFS) standards for network-attached storage ([NAS](https://searchstorage.techtarget.com/definition/network-attached-storage)).

NFS allows the user or system administrator to [mount](https://whatis.techtarget.com/definition/mount) (designate as accessible) all or a portion of a file system on a server. The portion of the file system that is mounted can be accessed by clients with whatever privileges are assigned to each file (read-only or read-write). NFS uses Remote Procedure Calls ([RPCs](https://searchmicroservices.techtarget.com/definition/Remote-Procedure-Call-RPC)) to route requests between clients and servers. NFS was originally developed by [Sun Microsystems](https://www.theserverside.com/definition/Sun-Microsystems) in the 1980's and is now managed by the Internet Engineering Task Force ([IETF](https://searchmicroservices.techtarget.com/definition/IETF-Internet-Engineering-Task-Force)).

**REDUNDANT ARRAY OF INDEPENDENT DISK (RAID):**

RAID (redundant array of independent disks; originally redundant array of inexpensive disks) is a way of storing the same data in different places on multiple [hard disks](https://searchstorage.techtarget.com/definition/hard-disk) to protect data in the case of a drive failure. However, not all RAID levels provide [redundancy](https://whatis.techtarget.com/definition/redundancy).

RAID works by placing data on multiple disks and allowing input/output ([I/O](https://whatis.techtarget.com/definition/input-output-I-O)) operations to overlap in a balanced way, improving performance. Because the use of multiple disks increases the mean time between failures ([MTBF](https://whatis.techtarget.com/definition/MTBF-mean-time-between-failures)), storing data redundantly also increases [fault tolerance](https://searchdisasterrecovery.techtarget.com/definition/fault-tolerant).

RAID LEVELS:-

**RAID 0 (Disk striping):**

RAID 0 splits data across any number of disks allowing higher data throughput. An individual file is read from multiple disks giving it access to the speed and capacity of all of them. This RAID level is often referred to as striping and has the benefit of increased performance. However, it does not facilitate any kind of redundancy and fault tolerance as it does not duplicate data or store any parity information (more on parity later). Both disks appear as a single partition, so when one of them fails, it breaks the array and results in data loss. RAID 0 is usually implemented for caching live streams and other files where speed is important and reliability/data loss is secondary.

Minimum number of disks: 2  
Pros: Increased performance (Write and read speeds).  
Cons: No redundancy.  
Business use: Live streaming, IPTV, VOD Edge Server

### RAID 1 (Disk Mirroring):

RAID 1 writes and reads identical data to pairs of drives. This process is often called data mirroring and it’s a primary function is to provide redundancy. If any of the disks in the array fails, the system can still access data from the remaining disk(s). Once you replace the faulty disk with a new one, the data is copied to it from the functioning disk(s) to rebuild the array. RAID 1 is the easiest way to create failover storage.

Minimum number of disks: 2  
Pros: Fault tolerance and easy data recovery. Increased read performance.  
Cons: Lower usable capacity. Higher cost per megabyte (double the amount of drives is required to achieve desired capacity).  
Business use: Standard application servers where data redundancy and availability is important.

### RAID 5 (Striping with parity):

RAID 5 stripes data blocks across multiple disks like RAID 0, however, it also stores parity information (Small amount of data that can accurately describe larger amounts of data) which is used to recover the data in case of disk failure. This level offers both speed (data is accessed from multiple disks) and redundancy as parity data is stored across all of the disks. If any of the disks in the array fails, data is recreated from the remaining distributed data and parity blocks. It uses approximately one-third of the available disk capacity for storing parity information.

Minimum number of disks: 3  
Pros: Fault tolerance and increased performance (lower than RAID 0)   
Cons: Lower performance with servers performing large amounts of write operations because of parity overhead.  
Ideal use: File storage servers and application servers.

### RAID 6 (Striping with double parity):

Raid 6 is similar to RAID 5, however, it provides increased reliability as it stores an extra parity block. That effectively means that it is possible for two drives to fail at once without breaking the array.

Minimum number of disks: 4  
Pros: Even higher redundancy than RAID 5. Increased read performance.  
Cons: Lower performance with servers performing large amounts of write operations because of parity overhead.  
Ideal use: Large file storage servers and application servers.

### RAID 10 (Striping + Mirroring):

RAID 10 combines the mirroring of RAID 1 with the striping of RAID 0. Or in other words, it combines the redundancy of RAID 1 with the increased performance of RAID 0. It is best suitable for environments where both high performance and security is required.

Minimum number of disks: 4  
Pros: Very high performance. Fault tolerance.  
Cons: Lower usable capacity/High cost. Limited scalability  
Ideal use: Highly utilized database servers/ servers performing a lot of write operations.

**LOGICAL VOLUME MANAGER (LVM):-**

Logical volume manager (LVM) introduces an extra layer between the physical disks and the file system allowing file systems to be :  
– resized and moved easily and online without requiring a system-wide outage.

–Using discontinuous space on disk  
– meaningful names to volumes, rather than the usual cryptic device names.  
– span multiple physical disks

With LVM, the hard drive or set of hard drives is allocated to one or more physical volumes. The physical volume is then combined into volume group. Each volume group is divided into logical volume, which is formatted with a file system like ext3 and is then mounted.

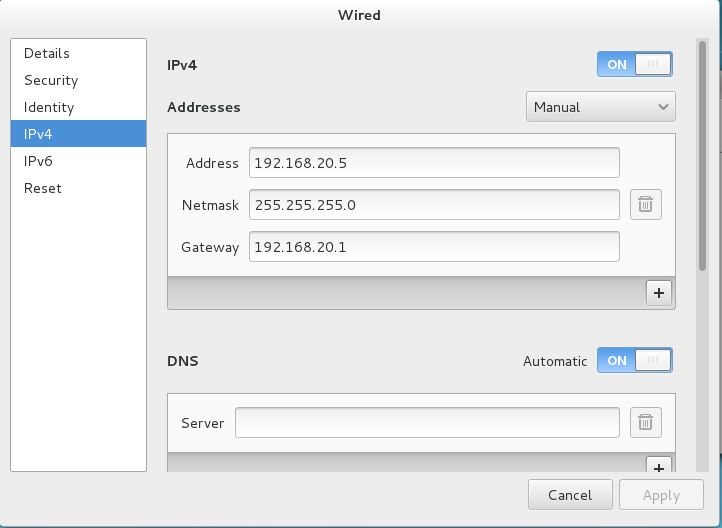
Logical volume manager (LVM) introduces an extra layer between the physical disks and the file system allowing file systems to be :  
– resized and moved easily and online without requiring a system-wide outage.

–Using discontinuous space on disk  
– meaningful names to volumes, rather than the usual cryptic device names.  
– span multiple physical disks

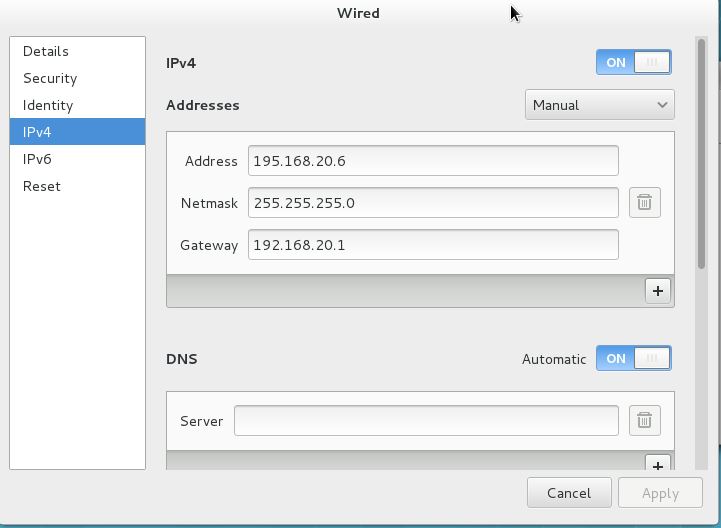
With LVM, the hard drive or set of hard drives is allocated to one or more physical volumes. The physical volume is then combined into volume group. Each volume group is divided into logical volume, which is formatted with a file system like ext3 and is then mounted.

**CONFIGURATION**

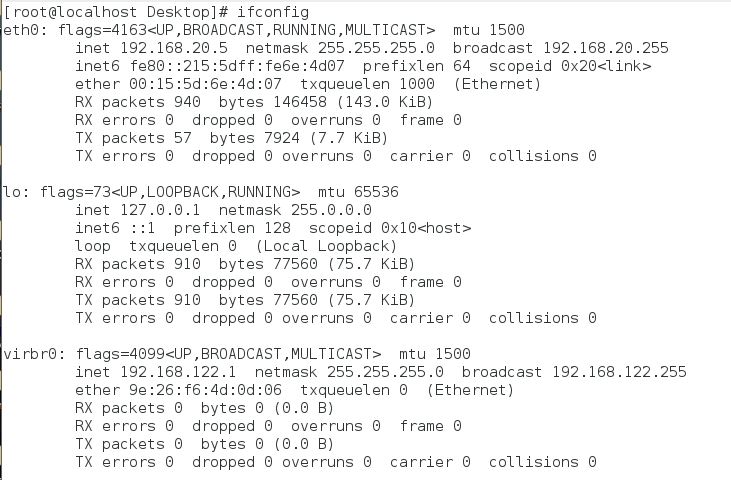
* **APPLYING IP IN SERVER AND CLIENT**
* Applying IP in server: - 192.168.20.5

****

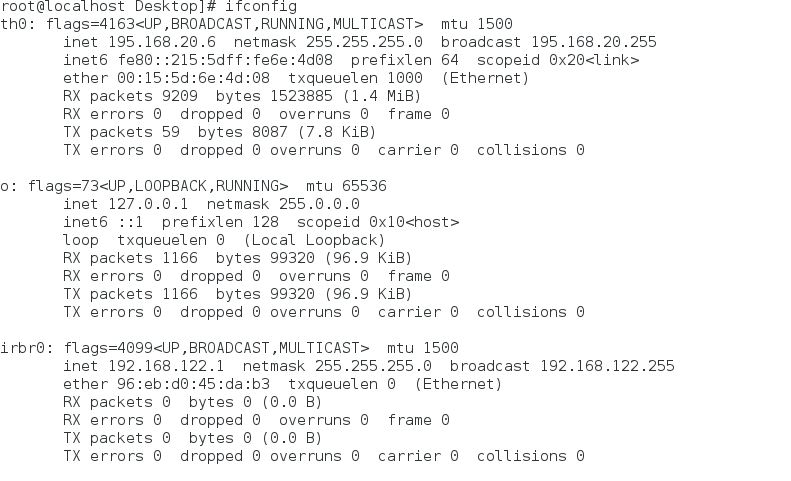
Applying IP in client: - 192.168.20.6



**Output in Server**

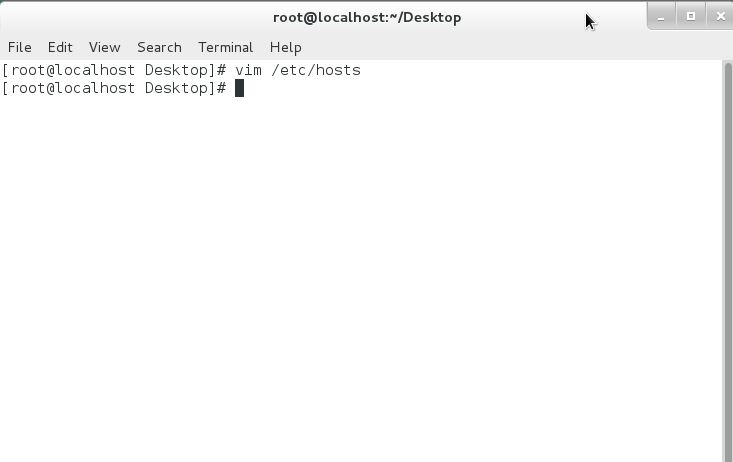


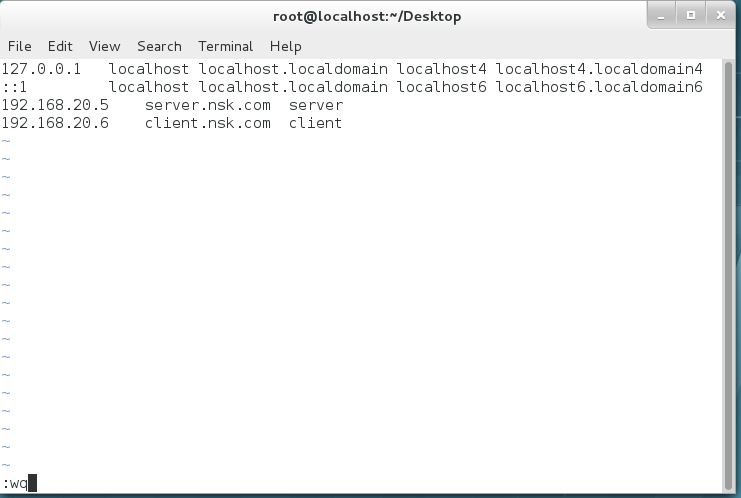
**Output in Client**

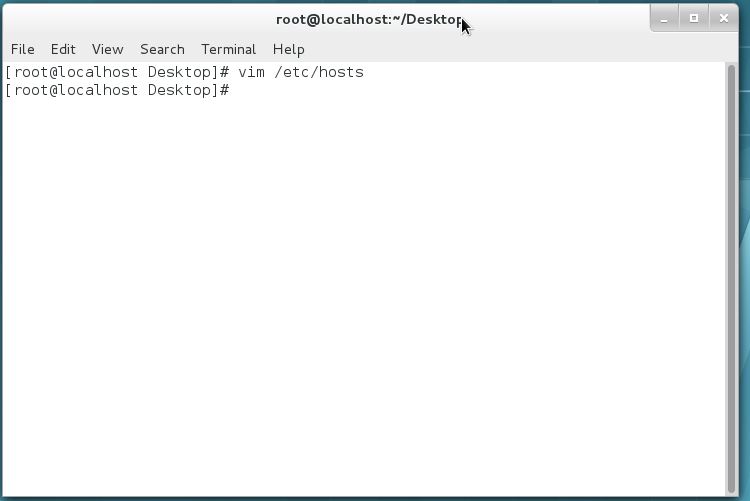


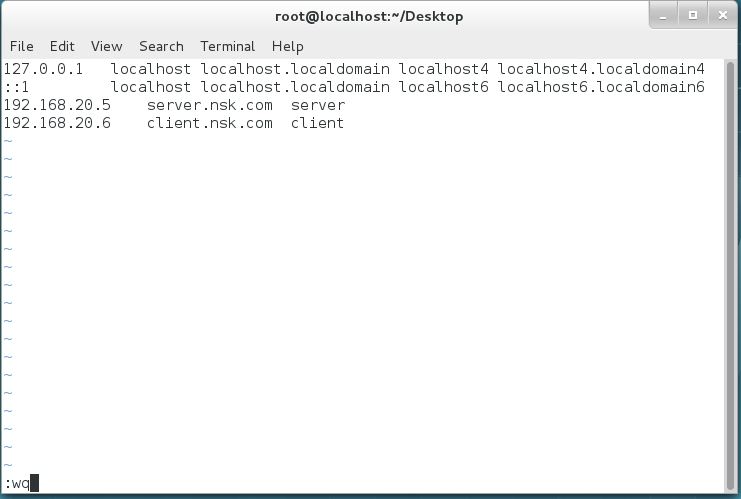
* **APPLYING DOMAIN NAME SYSTEM:-**

Applying in Server:-

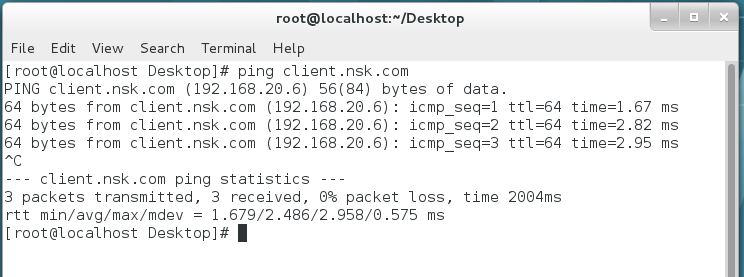




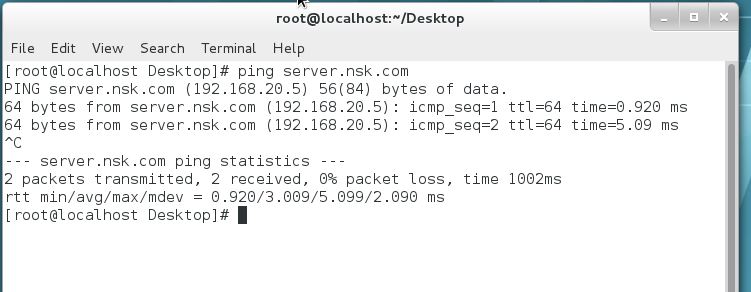
Input in Client:- 



**Output server to client**

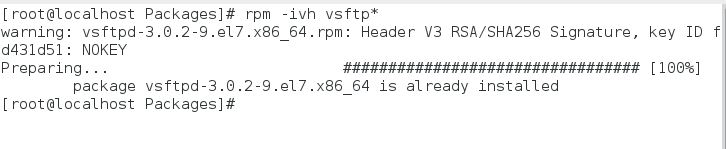
****

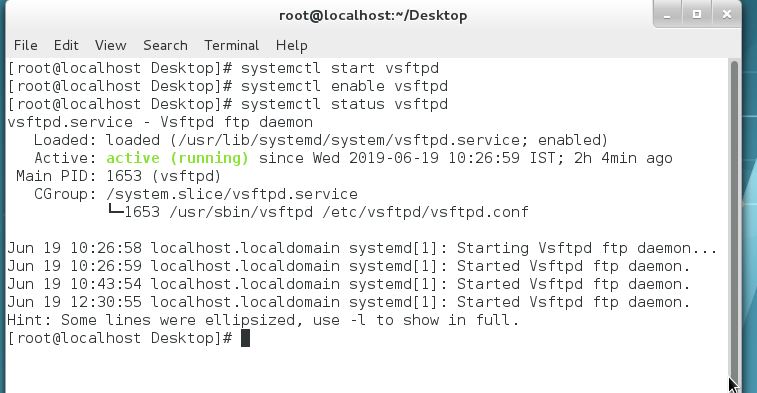
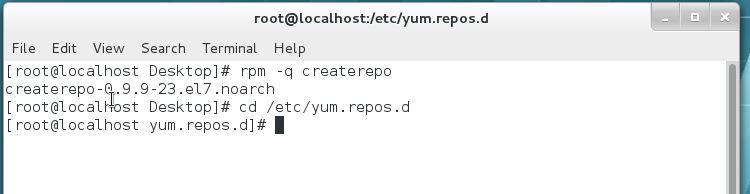
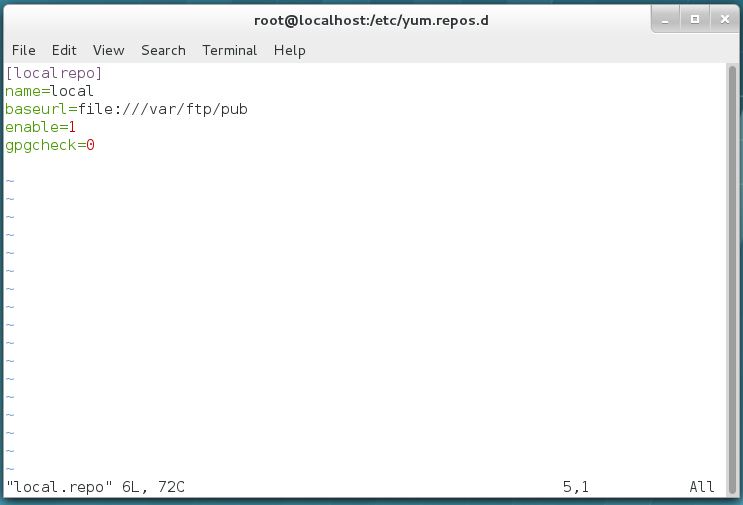
**Output to client to server**

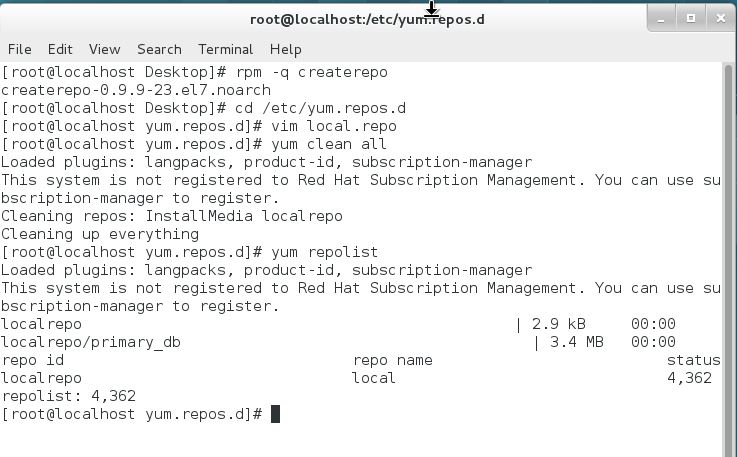
****

* **APPLYING NETWORK FILE SHARING (NFS):-**

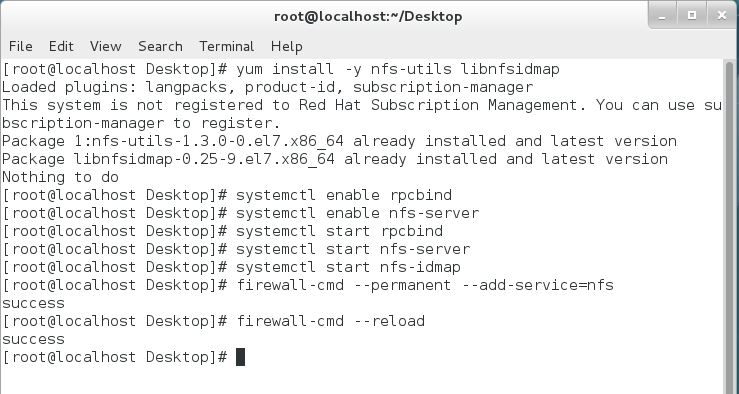
Applying in server:-

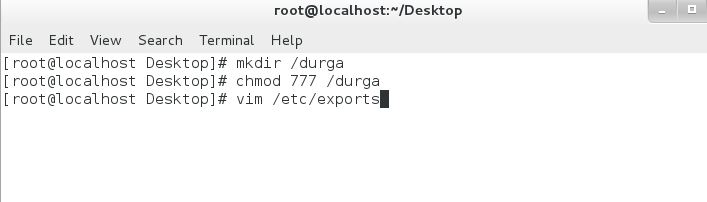


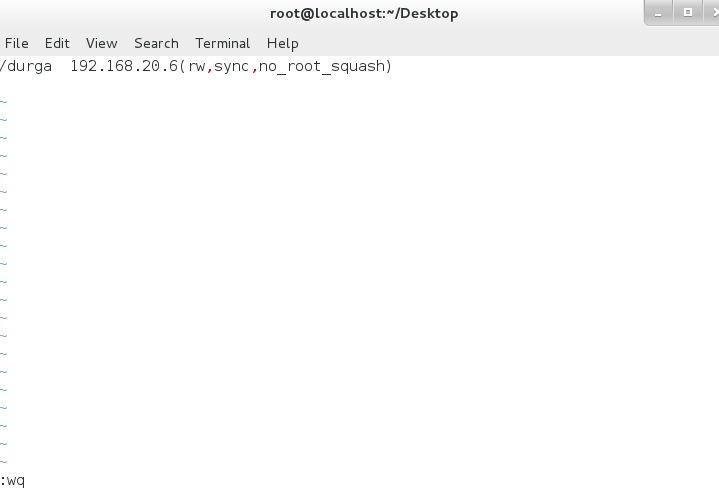
  





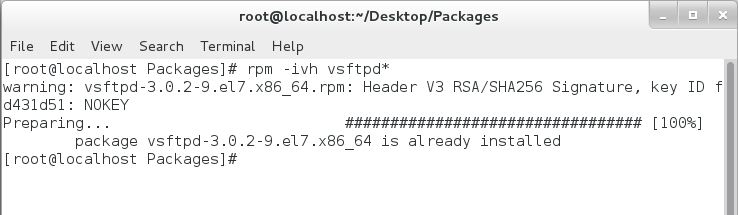


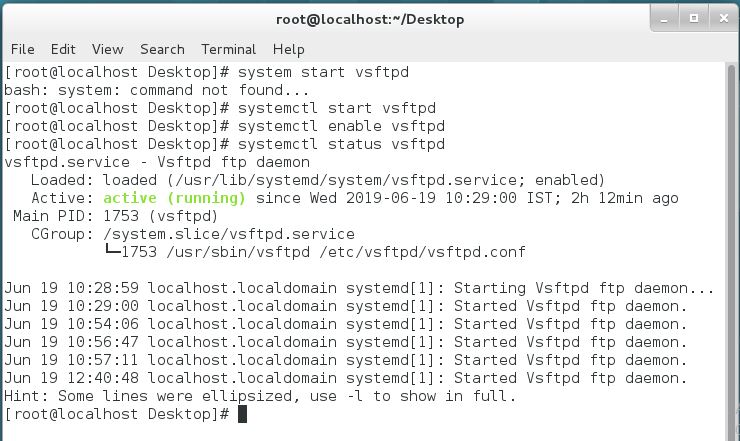


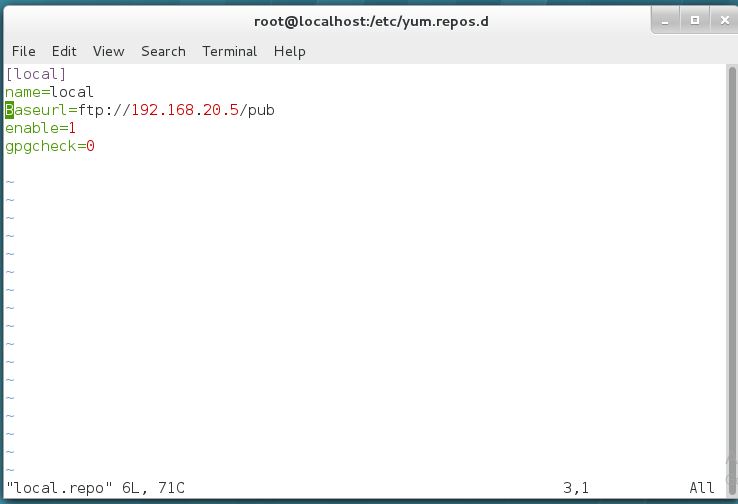




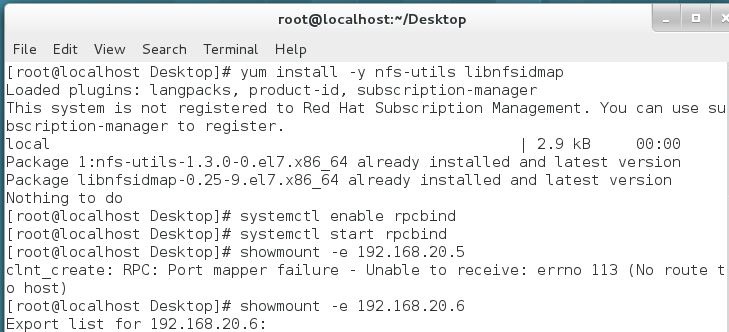
Applying in Client:-

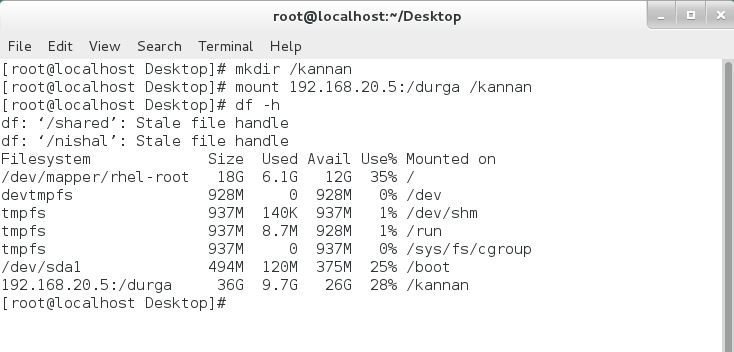




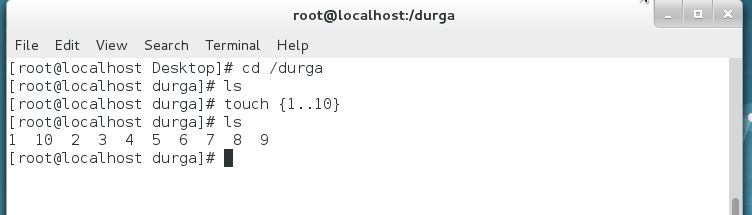




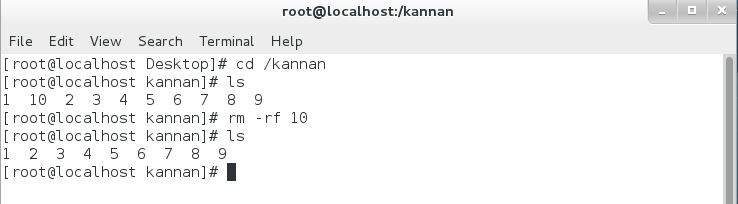




OUTPUT IN SERVER

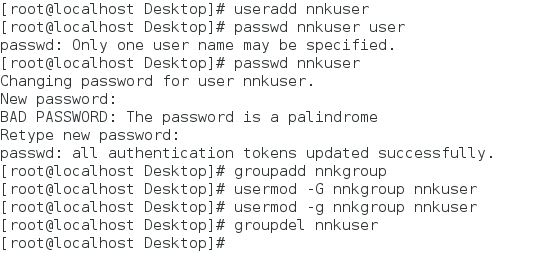


OUTPUT IN CLIENT



ADDING USER AND CHANGING PRIMARY GROUP

INPUT:-

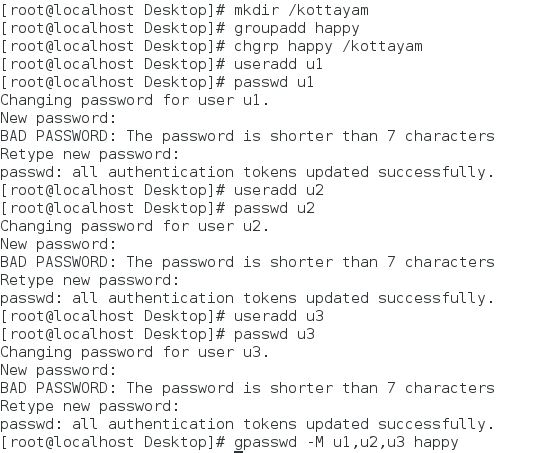


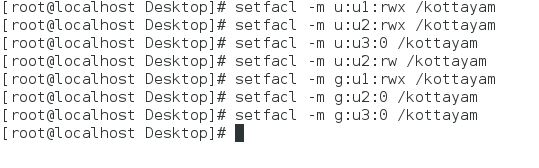
OUTPUT:-



* **PERMISSION SETTING FOR USERS AND ADDING USERS TO GROUP IN DRIECTORY:-**

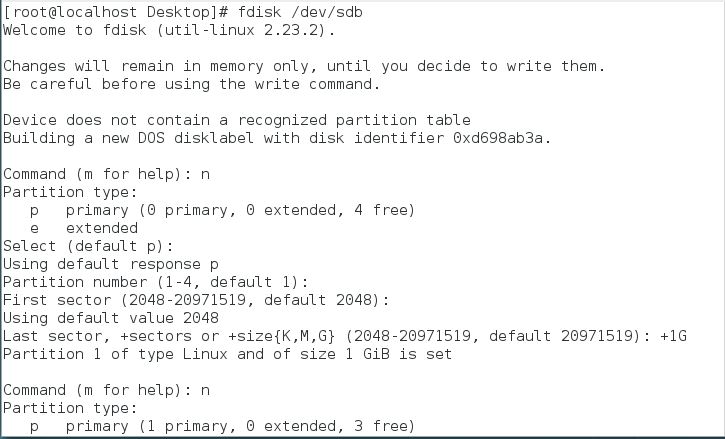
Input:-

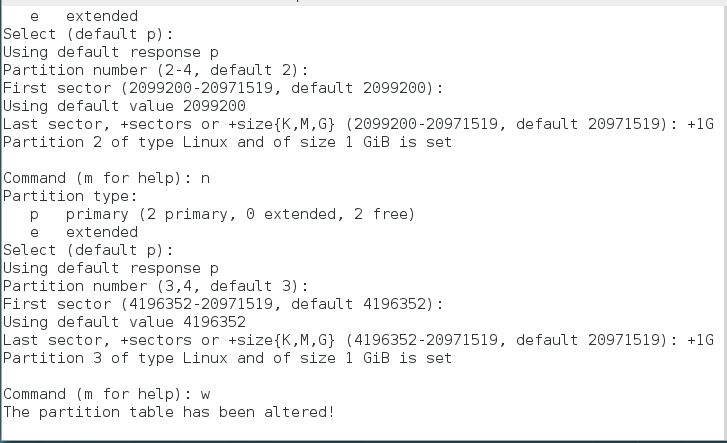


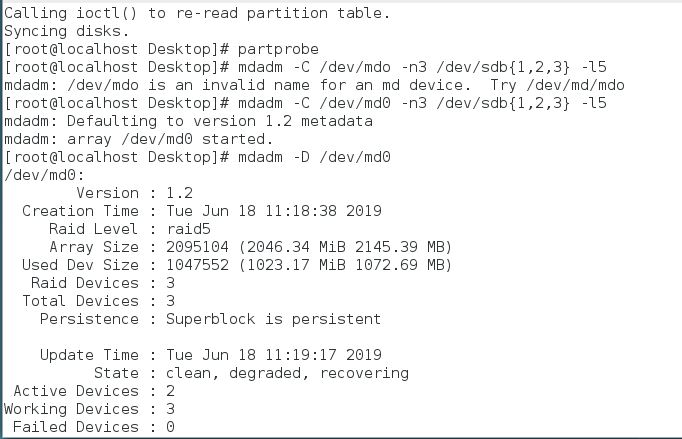


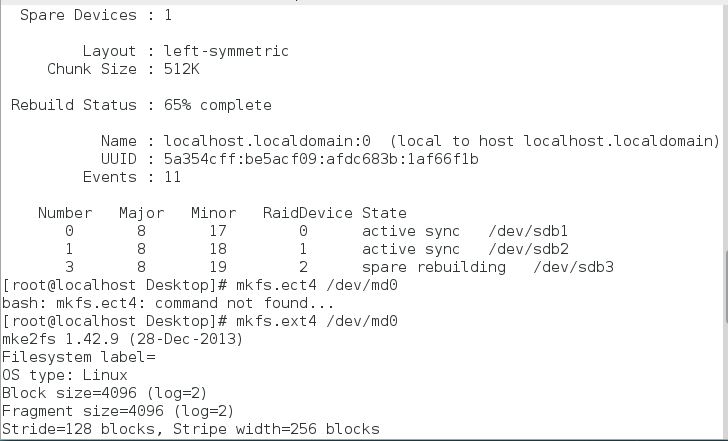
**Output**

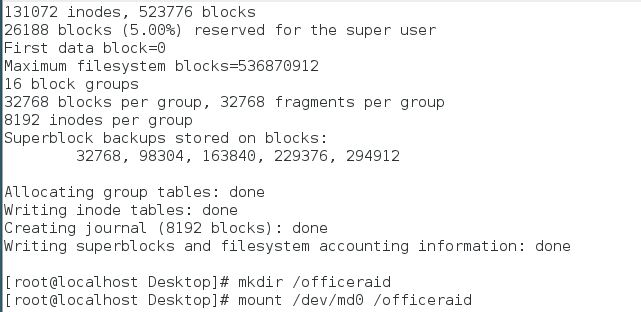


* **APPLYING RAID AND MOUNTING DIRECTORY:-**
* Input:-
* 

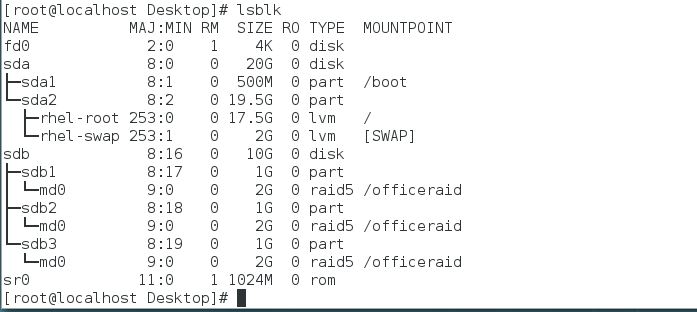






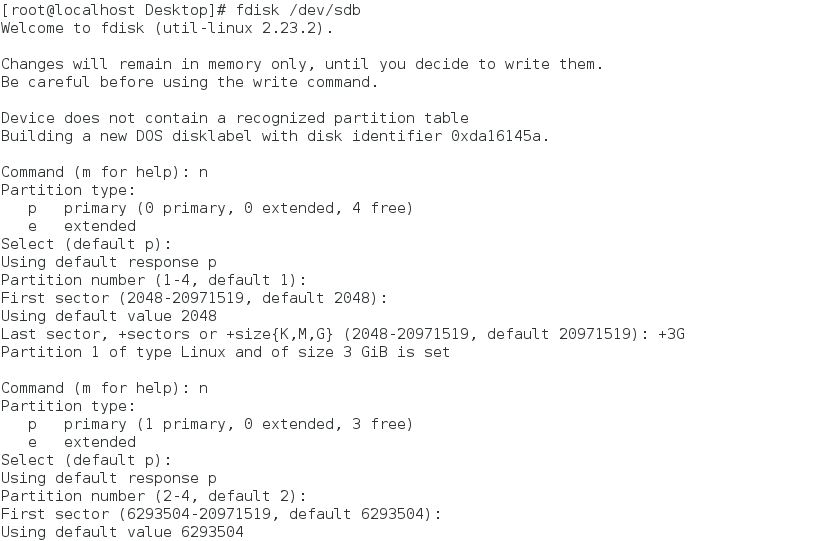


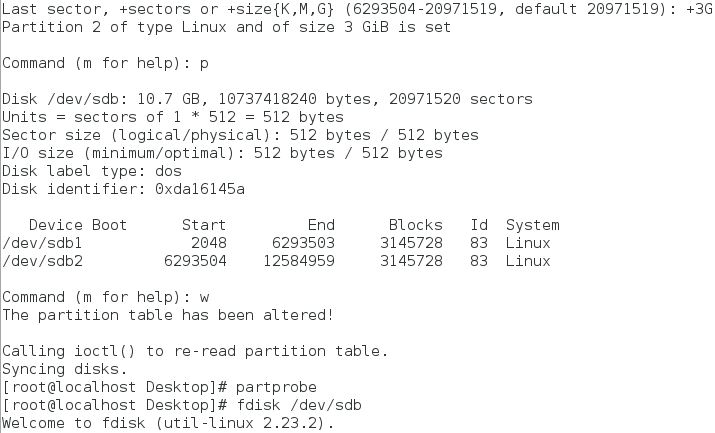
**Output**

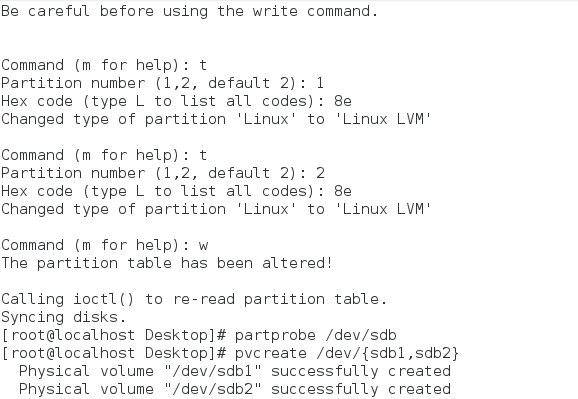


* **APPLYING LOGICAL VOLUME MANAGER AND RESIZING:-**

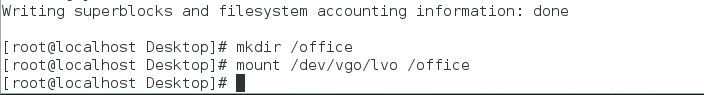
Input:-

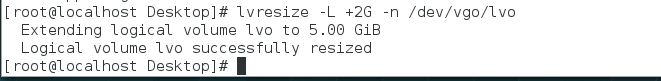




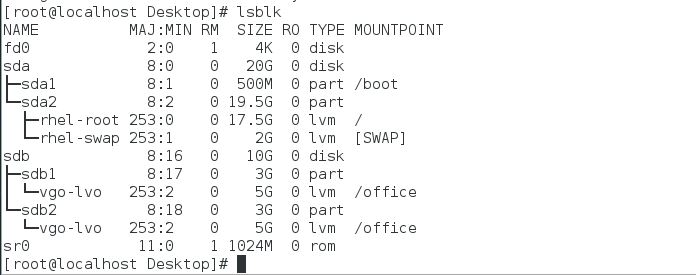








**Output**



**RESULT**

The Implementing of Linux Server Administration proposal for a company is successfully completed. Aim of this project is fulfilled.