1. **Analyze the structure of the /etc/passwd and /etc/group file, what fields are present in it, what users exist on the system? Specify several pseudo-users, how to define them?**

**/etc/passwd**

From the above image:

1. **Username**: It is used when user logs in. It should be between 1 and 32 characters in length.
2. **Password**: An x character indicates that encrypted password is stored in /etc/shadow file. Please note that you need to use the passwd command to computes the hash of a password typed at the CLI or to store/update the hash of the password in /etc/shadow file.
3. **User ID (UID)**: Each user must be assigned a user ID (UID). UID 0 (zero) is reserved for root and UIDs 1-99 are reserved for other predefined accounts. Further UID 100-999 are reserved by system for administrative and system accounts/groups.
4. **Group ID (GID)**: The primary group ID (stored in /etc/group file)
5. **User ID Info (GECOS)**: The comment field. It allow you to add extra information about the users such as user’s full name, phone number etc. This field use by finger command.
6. **Home directory**: The absolute path to the directory the user will be in when they log in. If this directory does not exists then users directory becomes /
7. **Command/shell**: The absolute path of a command or shell (/bin/bash). Typically, this is a shell. Please note that it does not have to be a shell. For example, sysadmin can use the nologin shell, which acts as a replacement shell for the user accounts. If shell set to **/sbin/nologin** and the user tries to log in to the Linux system directly, the /sbin/nologin shell closes the connection.

**/etc/group:**

group\_name: It is the name of group. If you run ls -l command, you will see this name printed in the group field.

Password: Generally password is not used, hence it is empty/blank. It can store encrypted password. This is useful to implement privileged groups.

Group ID (GID): Each user must be assigned a group ID. You can see this number in your /etc/passwd file.

Group List: It is a list of user names of users who are members of the group. The user names, must be separated by commas.

1. What are the uid ranges? What is UID? How to define it?

Множество допустимых значений UID зависит от системы; в общем случае UID допускает использование значений от 0 до 65535 с некоторыми оговорками:

* Суперпользователь всегда должен иметь UID, равный нулю (0).
* Пользователю [nobody](https://ru.wikipedia.org/wiki/Nobody_(%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8C)" \o "Операционная система) обычно присваивается или наибольший из возможных UID (в противоположность суперпользователю), или один из системных UID (см. ниже).
* UID с 1 по 100 по соглашению резервируются под системные нужды; некоторые руководства рекомендуют резервировать UID со 101 по 499 (в [Red Hat](https://ru.wikipedia.org/wiki/Red_Hat)) или даже 999 (в [Debian](https://ru.wikipedia.org/wiki/Debian" \o "Debian)).

A unique identifier (UID) is a numeric or alphanumeric string that is associated with a single entity within a given system. UIDs make it possible to address that entity, so that it can be accessed and interacted with.

Unique identifiers can be assigned to anything that needs to be distinguished from other entities, such as individual users, companies, machines or websites. These distinctive values are usually assigned depending on the needs of the specific application, but can either be randomly auto-generated with an algorithm, allocated incrementally or chosen by the user.

Uses of UIDs

The most widely known use of unique identifiers occurs when users register for a website or service. Customers are often provided with a username or user ID that allows the company they are registering with to differentiate them within their user logs. These identifiers are then also used for security and log on purposes.

In a database or spreadsheet, unique identifiers may be designated as a specific column or field to help make sorting and filtering through information easier. This also helps trace information back to a specific user or entity within the system.

Another popular application of UIDs is in a physical supply chain. Manufacturers often mark individual pieces of a larger component, such as computer parts, or an entire product with a serial number. This allows users to trace back the origin of the product in case of a malfunction, defect or recall.

3) What is GID? How to define it?

unique identifier of the group within the system to which the user belongs

GID defines the id or name of the group to which the user belongs

4) How to determine belonging of user to the specific group?

$ groups [username]

5) What are the commands for adding a user to the system? What are the basic parameters required to create a user?

1) $ sudo adduser

1. $ sudo useradd -s /path/to/shell -d /home/{dirname} -m -G {secondary-group} {username}  
   $ sudo passwd {username}
2. Add user through the gui

6) How do I change the name (account name) of an existing user?

**usermod -l login-name old-name**

**usermod -u UID username**

7) What is skell\_dir? What is its structure?

Directory /etc/skel/ (skel is derived from the “skeleton”) is used to initiate home directory when a user is first created.

skel is derived from the skeleton because it contains basic structure of home directory

The /etc/skel directory contains files and directories that are automatically copied over to a new user’s when it is created from useradd command.

This will ensure that all the users gets same intial settings and environment

8) How to remove a user from the system (including his mailbox)?

userdel -r username

9) What commands and keys should be used to lock and unlock a user account?

To lock:

passwd -l username

usermod -l username

To unlock:

passwd -u username

usermod -U username

10) How to remove a user's password and provide him with a password-free

login for subsequent password change?

passwd -e [username]

11) Display the extended format of information about the directory, tell about

the information columns displayed on the terminal.

Ls -al

* file permissions (-rwxrw-r--),
* number of (hard) links (1),
* owner name (root),
* owner group (root),
* file size in bytes (2048),
* time of last modification (Jan 13 07:11), and
* file/directory name (afile.exe)

12) What access rights exist and for whom (i. e., describe the main roles)?

Briefly describe the acronym for access rights.

There are 3 access right: read, write and execute. We give them separately to owner, group users and to all users.

13) What is the sequence of defining the relationship between the file and the

user?

First of all, cat /etc/group only shows the groups created in the system.

14) What commands are used to change the owner of a file (directory), as well

as the mode of access to the file? Give examples, demonstrate on the terminal.

chown newuser filename(s)

example:

chown whales chownSample.txt

15) What is an example of octal representation of access rights? Describe the

umask command.

Octal notation is a numerical system for modifying the permissions on Linux, Mac and other Unix like file systems. Each octal permission can be represented by 3 or 4 numbers; where each of these numbers is an "octal", meaning they range from 0-7. Each one of the numbers represents permissions that can be set to either a file or directory.  
**example octal: "724"**  
  
Owner - 7  
Group - 2  
Other - 4

16) Give definitions of sticky bits and mechanism of identifier substitution. Give

an example of files and directories with these attributes.

For Example:  
The files user\_file\_0 and user\_file\_1 are created by different users but have read-write-execute access on for all the users. This means that the user ‘guest’ can delete or rename the file created by user ‘guest-2’.

In order to avoid this, sticky bit can be set on the directory allAccess

# ls -ld allAccess/

drwxrwxrw**t** 2 himanshu himanshu 4096 Oct 24 16:19 allAccess/

Turn ON the sticky bit on the directory by using +t flag of chmod command.

s can be observed, a permission bit ‘t’ is introduced in the permission bits of the directory.

Now, if the user ‘guest’ tries to rename the file ‘user\_file\_1’, here is what happens :

$ mv /home/himanshu/allAccess/user\_file\_1 /home/himanshu/allAccess/user\_file\_0

mv: cannot move `/home/himanshu/allAccess/user\_file\_1' to `/home/himanshu/allAccess/user\_file\_0': Operation not permitted

So we see that the operation was not permitted.

17) What file attributes should be present in the command script?

Some filesystems support additional attributes (other than those described in the preceding sections). In particular, some Linux-native filesystems support several attributes that you can adjust with the [chattr command](https://linoxide.com/change-attributes-of-file/" \t "_blank). The files and directories can have following attributes:

The detailed meaning of these attributes according to the manual page is:

* **a - append only:**this attribute allows a file to be added to, but not to be removed. It prevents accidental or malicious changes to files that record data, such as log files.
* **c - compressed:**it causes the kernel to compress data written to the file automatically and uncompress it when it’s read back.
* **d - no dump:**it makes sure the file is not backed up in backups where the dump utility is used
* **e - extent format:**it indicates that the file is using extents for mapping the blocks on disk.
* **i - immutable:**it makes a file immutable, which goes a step beyond simply disabling write access to the file. The file can’t be deleted, links to it can’t be created, and the file can’t be renamed.
* **j - data journaling:**it ensures that on an Ext3 file system the file is first written to the journal and only after that to the data blocks on the hard disk.
* **s - secure deletion:**it makes sure that recovery of a file is not possible after it has been deleted.
* **t - no tail-merging:**Tail-merging is a process in which small data pieces at a file’s end that don’t fill a complete block are merged with similar pieces of data from other files.
* **u - undeletable:**When a file is deleted, its contents are saved which allows a utility to be developed that works with that information to salvage deleted files.
* **A - no atime updates:**Linux won’t update the access time stamp when you access a file.
* **D - synchronous directory updates:**it makes sure that changes to files are written to disk immediately, and not to cache first.
* **S - synchronous updates:**the changes on a file are written synchronously on the disk.
* **T - and top of directory hierarchy:**A directory will be deemed to be the top of directory hierarchies for the purposes of the Orlov block allocator.