System Administration

# Managing Users

## User Accounts

* User accounts in Ubuntu are stored in the **/etc/passwd** file. Format is the following:
  + **Username:password:uid:gid:comment:<home directory>:<preferred shell>**

Text

Description automatically generated

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Username** | **Password** | **UID** | **GID** | **Comment** | **Home Dir** | **Shell (pref)** |
| cstUser1 | Encrypted | 1001 | 1001 | Null | /home/cstUser1 | /bin/sh |
| cstUser2 | Encrypted | 1002 | 1001 | Null | /home/cstUser2 | /bin/sh |

The preferred shell can be changed. For many distributions, Bash is the default shell, but a system admin may need to set this. On my local VM, SH was the default shell for new users.

We can change context for our use by using the **su** command.

Text

Description automatically generated

Comments:

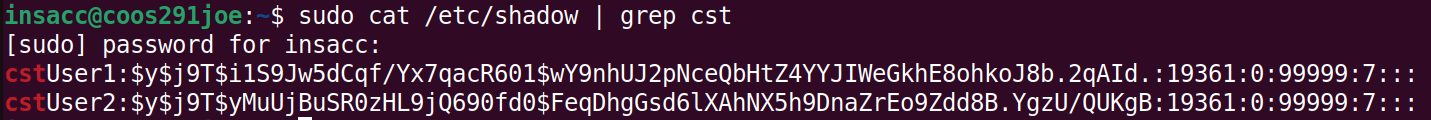
* You’ll be asked for the password of the user you specified in the **su** command
* In the above screen, we that originally I was logged in as **insacc**
* I logged in as cstUser1 via the su command
* If I tried to do a file listing (or even execute some of my scripts that were created by insacc), we got permission denied. (cstUser1 would not be the owner of those scripts we created previously).
* To quit that context of the user, simply type “exit”

Some other notes:

* The directory **/etc/skel** is a template that Linux uses to create a user’s home directory.
* Any files or directories in that /etc/skel will be copied to the users home directory on user creation.

## User Passwords

* Passwords are stored in the **/etc/shadow** file in the following format:
* Username:password:lastchanged:minimum:maximum:warn:inactive:expire



## Groups

* Stored in the **/etc/group** file in the following format:
* **Group\_name:password:gid:<userlist of that group>**

**Text

Description automatically generated**

Comments:

* In our VM, we have a custom group called “CST2”
* We can see our cst users (cstUser1, cstUser2, as well as insacc) are members of the adm (Administrator), sudo, and potentially “CST2” groups.

## Creating, Deleting, Changing Users

Some information regarding accounts for Ubuntu:

* list of documentation: <https://help.ubuntu.com/>
* desktop: <https://help.ubuntu.com/lts/ubuntu-help/user-accounts.html>
* server: <https://help.ubuntu.com/lts/serverguide/index.html>

Let’s take a look at our /etc/skel directory:

Graphical user interface

Description automatically generated with medium confidence

It’s currently empty. Let’s create a file called readmeFirst.txt and put it in that /etc/skel directory.

Text

Description automatically generated

### Adding Users:

To add a user, use the command **adduser**

Text

Description automatically generated

Comments:

* We need to elevate our permissions via sudo
* When creating a new user, we probably need to specify a password, unless otherwise overridden via a switch in our **adduser** command
* You can specify the Full Name, Room Number, Work Phone, Home Phone, Other Information when you add new users (Pressing ENTER will use default values)

Text

Description automatically generated

Comments:

* Notice for our new user that we created, the files in **/etc/skel** that we created earlier were copied to the new user’s home directory.
* The files themselves in /etc/skel, when copied, will have ownership and groups changed to reflect the new user and the new group.

### Adding Groups

We can use the **addgroup** command to add a group (if you’re logged in as cstUser3, type “exit” to go back to your user with sudo permissions):

Text

Description automatically generated

And then view the /etc/group file to confirm:

Text

Description automatically generated

Again, there is a special file: /etc/adduser.conf, which contains information what the system will do when adding a user.

### Deleting a User

Use the **deluser** command (We’ll create a temp user and then delete it):

Text

Description automatically generated

When we delete a user, their home directory should still exist.

### Changing a user’s shell:

We can use the **chsh** command, or we can alter the passwd file (avoid this). The list of available shells are in the file /etc/shells:

Text

Description automatically generated

Let’s change cstUser1 and cstUser2 to use the /bin/bash shell instead of /bin/sh shell.

Text

Description automatically generated

Comments:

* User shells are listed in /etc/passwd
* All shells can be listed from /etc/shells
* Use the command **chsh** to change a user’s shell

### Changing Passwords

Use the command **passwd** to change the current user’s password, or pass it a username to change that user’s password

Text

Description automatically generated

Comments:

* Notice the password hashes for cstUser3 have been changed after invoking the passwd command.

# Storage Administration

One of the main commands we use for Storage Administration is **fdisk**. We can use this to view existing disks and partitions.

Text

Description automatically generated

Let’s take a look at our hypervisor to see what kind of virtual drive we have attached:

Graphical user interface, text, application

Description automatically generated

Here, I have a 20GB drive. Let’s see if we can find the corresponding entry in our fdisk list:

Text

Description automatically generated

Comments:

* /dev/sda
  + This is what Linux calls to reference the physical disk (since we’re using VMs, there are VHD files)
* 20 GiB
  + 20 GBs in size
* Sector Size:
  + 512 bytes. This is the smallest allocation unit that the drive can use to store files.
  + General rule:
    - The large the sector size, the more wasted space, but the less sectors that need to be read to access a file
    - A smaller sector size will waste less space, but may take longer to read/write.
* /dev/sda1, sda2, sda3
  + These are individual partitions that were created for us when we installed Linux

We can use the command **lshw -C disk**  to view physical disks:

Text

Description automatically generated

Comments:

* \*-disk
  + Denotes a hard drive
  + Logical name: /dev/sda
  + Size: 20GB
* \*-cdrom
  + Our “virtual” cdrom drive
  + Logical name: /dev/cdrom
* \*-medium
  + Our “virtual” CD inserted into our “virtual” cdrom
  + Same logical name

Practice: Let’s add some users and groups via the command line

* Add a group cststaff with GID 2000
* Add three new accounts: Alex, George, Kelci, all members of the cststaff group.
  + UIDs: Alex 2000, George 2001, Kelci 2002
* Delete Alex’s account.

Text

Description automatically generated

We’ve added group called cststaff with group id of 2000

Text

Description automatically generated

Add a user alex

Text

Description automatically generated

Added a user for George

Text

Description automatically generated

Added a user for Kelci

Next, let’s delete Alex.

Text

Description automatically generated

Let’s try creating a new user **heath** with Alex’s UID:

Text

Description automatically generated

As we can see, heath (a new account, not related at all to alex) has ownership of alex’s old files.

The inode stores the owner ID, not the owner; ll / ls commands converts from the UID to the account name, if the account exists. In this case, Heath now owns all of Alex’s files.