



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Experiment No. 3
Explore Linux Commands
Date of Performance:
Date of Submission:
Marks:
Sign:



Aim: Explore user management commands of linux.

Objective:

Explore basic commands of linux

Theory:

A user is an entity, in a Linux operating system, that can manipulate files and perform several other operations. Each user is assigned an ID that is unique for each user in the operating system. In this post, we will learn about users and commands which are used to get information about the users. After installation of the operating system, the ID 0 is assigned to the root user and the IDs 1 to 999 (both inclusive) are assigned to the system users and hence the ids for local user begins from 1000 onwards.

In a single directory, we can create 60,000 users. Now we will discuss the important commands to manage users in Linux.

- **useradd** - create a new user or update default new user information ,useradd is a low

level utility for adding users.

- **userdel** - delete a user account and related files
- **groupadd** - create a new group , The groupadd command creates a new group account

using the values specified on the command line plus the default values from the system. The new group will be entered into the system files as needed.



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

- `groupdel` - delete a group , The `groupdel` command modifies the system account files, deleting all entries that refer to GROUP. The named group must exist
- `who` - show who is logged on , Print information about users who are currently logged in.
- `whoami` - print effective userid
- `passwd` - change user password

The `passwd` command changes passwords for user accounts. A normal user may only change the password for his/her own account, while the superuser may change the password for any account. `passwd` also changes the account or associated password validity period.

1. to enter in root `sudo su` then password

2. to add new user type `useradd csds11 (username)`

3

. to check a newly added user you have to type `cat /etc/passwd` 4 set a password to new user : `sudo passwd csds11`

5. create a new group: `groupadd csds12`

6. Check group `cat /etc/group`

7. add new user in newly created group `useradd -G csds12 piya1 (group name and new user name)`

8. to check : `cat /etc/group`



**9. to enter in new user : su -
csds11 (username)**

1

0. to delete user type : userdel csds (username that you have to delete)

1

**1. Again check whether it is deleted or not cat
/etc/passwd**

1

0. to delete user type : groupdel csds12 (group that you have to delete)

1

**2. Again check whether it is deleted or not cat
/etc/passwd**

13. who -

s

how who is logged on Print information about users who are currently logged in.

14.whoami - print effective userid

Code :

```
#include <stdio.h>
```

```
#include <unistd.h>
```



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

```
#include <sys/types.h>
```

```
#include <dirent.h>
```

```
int main(int argc, char *argv[]) {
```

```
    // Check if directory path is provided as argument
```

```
    const char *dir_path;
```

```
    if (argc > 1) {
```

```
        dir_path = argv[1];
```

```
    } else {
```

```
        dir_path = ".";
```

```
    }
```

```
    // Open the directory
```

```
    DIR *dir = opendir(dir_path);
```

```
    if (dir == NULL) {
```

```
        perror("opendir");
```

```
        return 1;
```

```
    }
```

```
    // Read directory entries
```

```
    struct dirent *entry;
```

```
    while ((entry = readdir(dir)) != NULL) {
```

```
        printf("%s\n", entry->d_name);
```

```
    }
```



// Close the directory

```
closedir(dir);
```

```
return 0;
```

```
}
```

Compile this program using gcc:

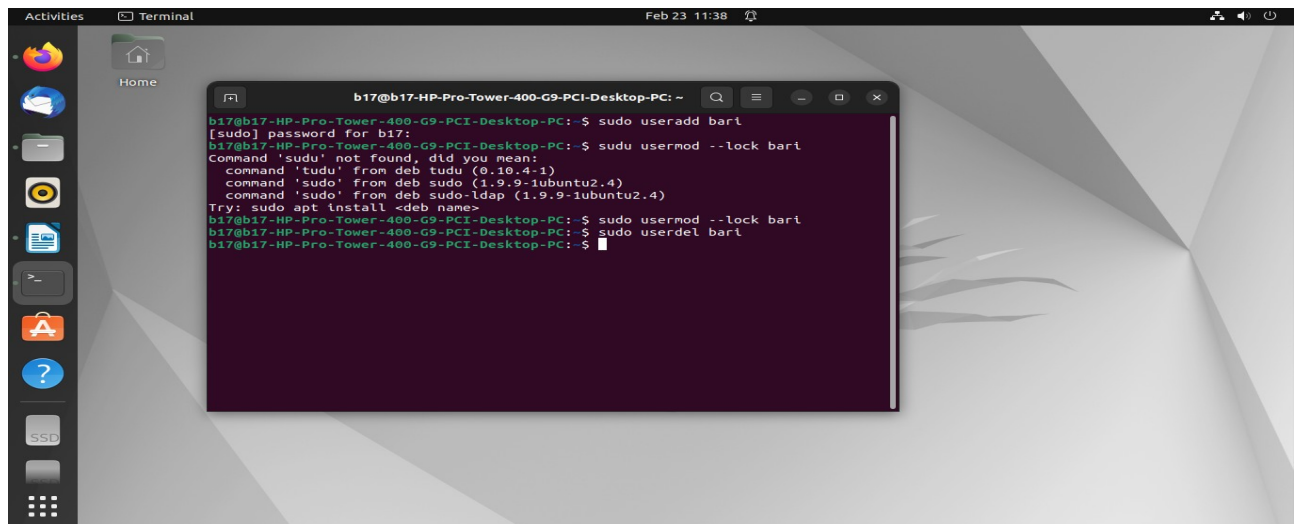
```
gcc -o myls myls.c
```

Now you can run it like a regular ls command, providing an optional directory path as an argument:

```
./mysls      # Lists current directory contents
```

```
./mysls /path/to/directory # Lists contents of specified directory
```

This program uses the opendir(), readdir(), and closedir() functions from the <dirent.h> header to open, read, and close directories, respectively. It prints the names of all directory entries to the standard output.





Output :

```
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo useradd bart  
[sudo] password for b17:  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo usermod --lock bart  
Command 'sudo' not found, did you mean:  
  command 'tudu' from deb tudu (0.10.4-1)  
  command 'sudo' from deb sudo (1.9.9-1ubuntu2.4)  
  command 'sudo' from deb sudo-ldap (1.9.9-1ubuntu2.4)  
Try: sudo apt install <deb name>  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo usermod --lock bart  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo userdel bart  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ ^C  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ id bart  
id: 'bart': no such user  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo useradd bart  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ id bart  
uid=1001(bart) gid=1001(bart) groups=1001(bart)  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$
```

```
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo usermod --lock bart  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo userdel bart  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ ^C  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ id bart  
id: 'bart': no such user  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo useradd bart  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ id bart  
uid=1001(bart) gid=1001(bart) groups=1001(bart)  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ paaswd bart  
command 'paaswd' not found, did you mean:  
  command 'passwd' from deb passwd (1:4.8.1-2ubuntu2.1)  
Try: sudo apt install <deb name>  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo paaswd bart  
sudo: paaswd: command not found  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo passwd bart  
New password:  
BAD PASSWORD: The password is shorter than 8 characters  
Retype new password:  
passwd: password updated successfully  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo groupadd bart  
groupadd: group 'bart' already exists  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo groupadd bart  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo groupdel bart  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$
```

```
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC: ~  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo usermod --lock bart  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo userdel bart  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ ^C  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ id bart  
id: 'bart': no such user  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo useradd bart  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ id bart  
uid=1001(bart) gid=1001(bart) groups=1001(bart)  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ paaswd bart  
command 'paaswd' not found, did you mean:  
  command 'passwd' from deb passwd (1:4.8.1-2ubuntu2.1)  
Try: sudo apt install <deb name>  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo paaswd bart  
sudo: paaswd: command not found  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo passwd bart  
New password:  
BAD PASSWORD: The password is shorter than 8 characters  
Retype new password:  
passwd: password updated successfully  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo groupadd bart  
groupadd: group 'bart' already exists  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$ sudo groupadd bart  
b17@b17-HP-Pro-Tower-400-G9-PCI-Desktop-PC:~$
```



Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

Conclusion:

In conclusion, delving into the user management commands of Linux unveils a versatile toolkit essential for system administrators and users alike. Through commands like `useradd`, `userdel`, `passwd`, and others, Linux empowers users to efficiently manage accounts, access permissions, and security settings. This exploration underscores the robustness and flexibility of Linux in tailoring user environments to specific needs, whether for individual users or across organizational networks. By mastering these commands, users gain greater control over their Linux systems, enhancing both security and productivity.