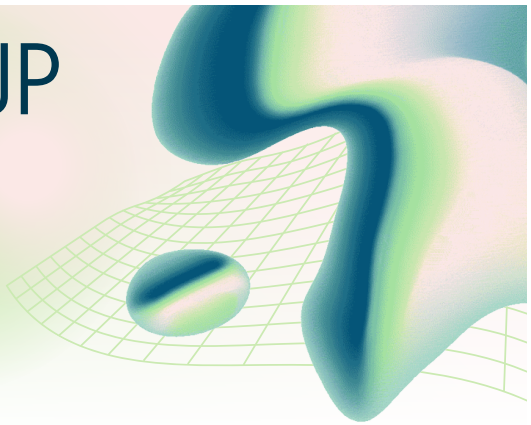
LINUX USER AND GROUP MANAGEMENT



By: Jayesh Gurav.

Overview

This document explains the purpose and functionality of a Bash script used to automate user and group management in a Linux environment. The script is designed to create a new group, add users, assign permissions, and remove users as needed.

Step 1: Connect to Your EC2 Instance

1. Open a terminal.
2. Use SSH to connect to your EC2 instance:
3. chmod 400 “your-key.pem”
4. ssh -i your-key.pem ubuntu@your-ec2-instance-ip

Step 2: Creating a Bash Script

Creating a New Group

```
GROUP_NAME="Devops_Group"
```

```
sudo groupadd $GROUP_NAME
```

```
echo "Group $GROUP_NAME created."
```

- Creates a new group named Devops_Group.
- The groupadd command is used, with sudo ensuring administrative rights.

Creating Three Users

```
USER1="jayesh"
```

```
USER2="gaurav"
```

`USER3="ferin"`

 # Creating users with the group as their primary group

`sudo useradd -m -g $GROUP_NAME $USER1`

`sudo useradd -m -g $GROUP_NAME $USER2`

`sudo useradd -m -g $GROUP_NAME $USER3`

`echo "Users $USER1, $USER2, and $USER3 created."`

- Defines users: jayesh, gaurav, and ferin.
- The useradd -m -g command creates home directories and assigns the group as their primary group.

Assigning Permissions

`DIR="/home/Devops_User_Project"`

`sudo mkdir -p $DIR`

`echo "Directory $DIR created."`

- Creates the directory /home/Devops_User_Project for shared use.

Permissions Assignment

Jayesh: Full Access

`sudo chmod 700 $DIR`

`sudo chown $USER1:$GROUP_NAME $DIR`

`echo "Permissions set for $USER1 (read/write/execute)."`

- Only jayesh can read, write, and execute files.

Gaurav: Read and Execute

`sudo chmod 755 $DIR`

`sudo chown $USER2:$GROUP_NAME $DIR`

`echo "Permissions set for $USER2 (read/execute)."`

- gaurav can read and execute but cannot modify files.

Ferin: Read-Only

```
sudo chmod 744 $DIR
```

```
sudo chown $USER3:$GROUP_NAME $DIR
```

```
echo "Permissions set for $USER3 (read-only)."
```

- ferin can only read the files but cannot modify them.

Removing a User from the Group

```
sudo gpasswd -d $USER2 $GROUP_NAME
```

```
echo "User $USER2 removed from group $GROUP_NAME."
```

- Removes gaurav from Devops_Group.

Optional: Deleting the User

```
# sudo userdel -r $USER2
```

```
# echo "User $USER2 deleted."
```

- If needed, deletes gaurav and removes his home directory.

Final Confirmation Message

```
echo "Group $GROUP_NAME and users have been set up and configured."
```

- Displays a message confirming successful execution.

Step 3: Make the Script Executable

Run the following command to grant execution permission:

```
chmod +x setup_users.sh
```

Step 4: Execute the Script

Run the script with:

```
sudo ./setup_users.sh
```

Step 5: Verify the Setup

Check the created group:

```
getent group Devops_Group
```

List the users:

```
cat /etc/passwd | grep -E "jayesh|gaurav|ferin"
```

Check directory permissions:

```
ls -ld /home/Devops_User_Project
```

Conclusion

This script automates user and group management, simplifying administrative tasks. It ensures controlled access permissions and allows easy modification or removal of users as required.