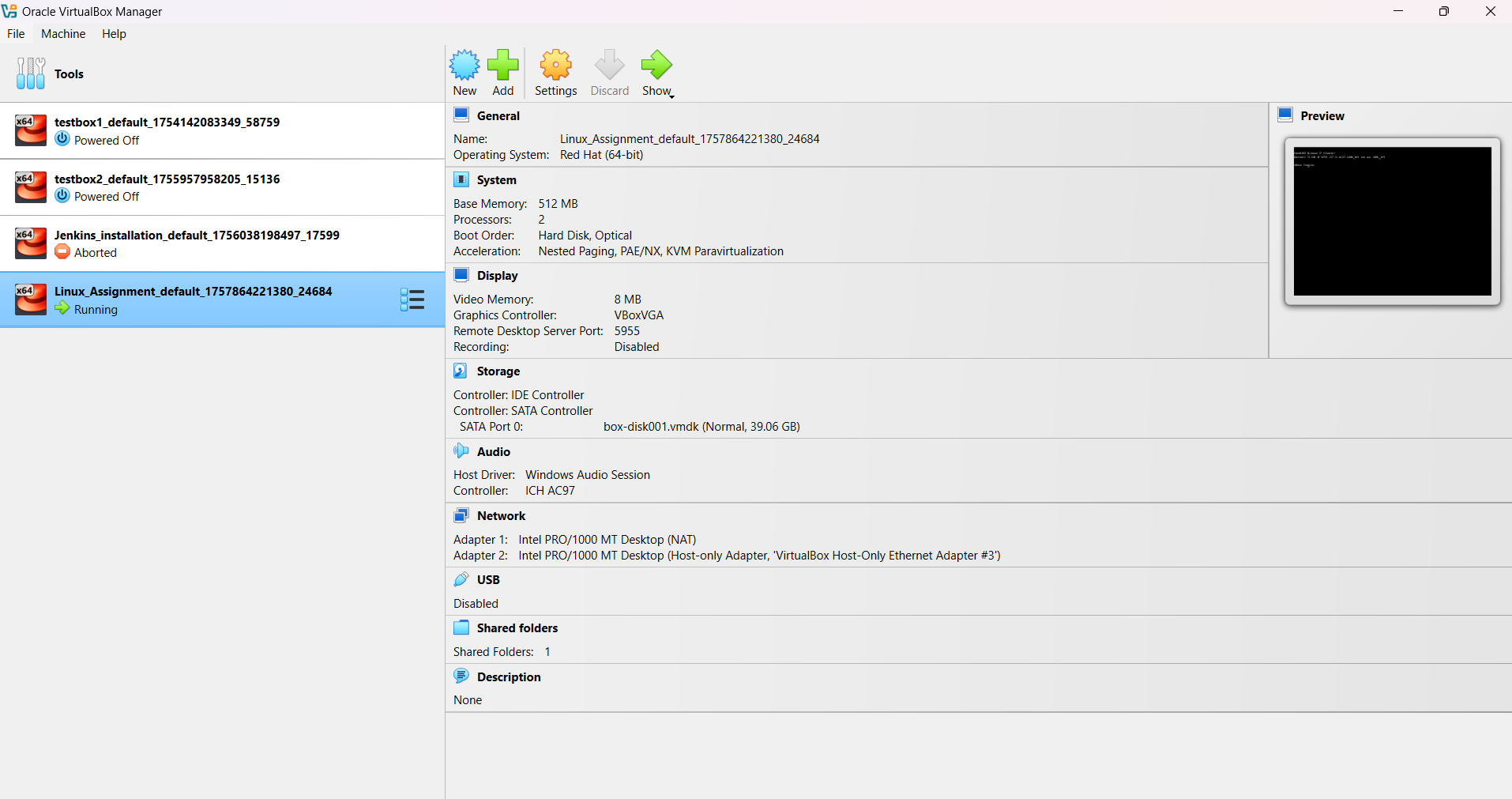
Linux Assignment :

To perform the below assignment first we need to start a virtual machine using Oracle Virtual Box.  
  
I have create a centos vm and started the ssh session to perform the task.  
  
  
  
As shown in the above picture, I have started the vm name Linux\_Assignment.

**Problem Statement**

Rahul, a Senior DevOps engineer at TechCorp, needs to set up and manage development environments for two new developers, Sarah and Mike. The setup must ensure proper system monitoring, user management, troubleshooting capabilities, and automated backup procedures.

As a Fresher DevOps Engineer, you are assisting Rahul in implementing a secure, monitored, and well-maintained development environment that meets the organization’s operational and security standards. Additionally, Sarah and Mike are tasked with setting up automated backups for their respective web servers.

**Tasks Detail:**

Task 1: System Monitoring Setup

Objective: Configure a monitoring system to ensure the development environment’s health, performance, and capacity planning.

Scenario:

* The development server is reporting intermittent performance issues.
* New developers need visibility into system resource usage for their tasks.
* System metrics must be consistently tracked for effective capacity planning.

Requirements:

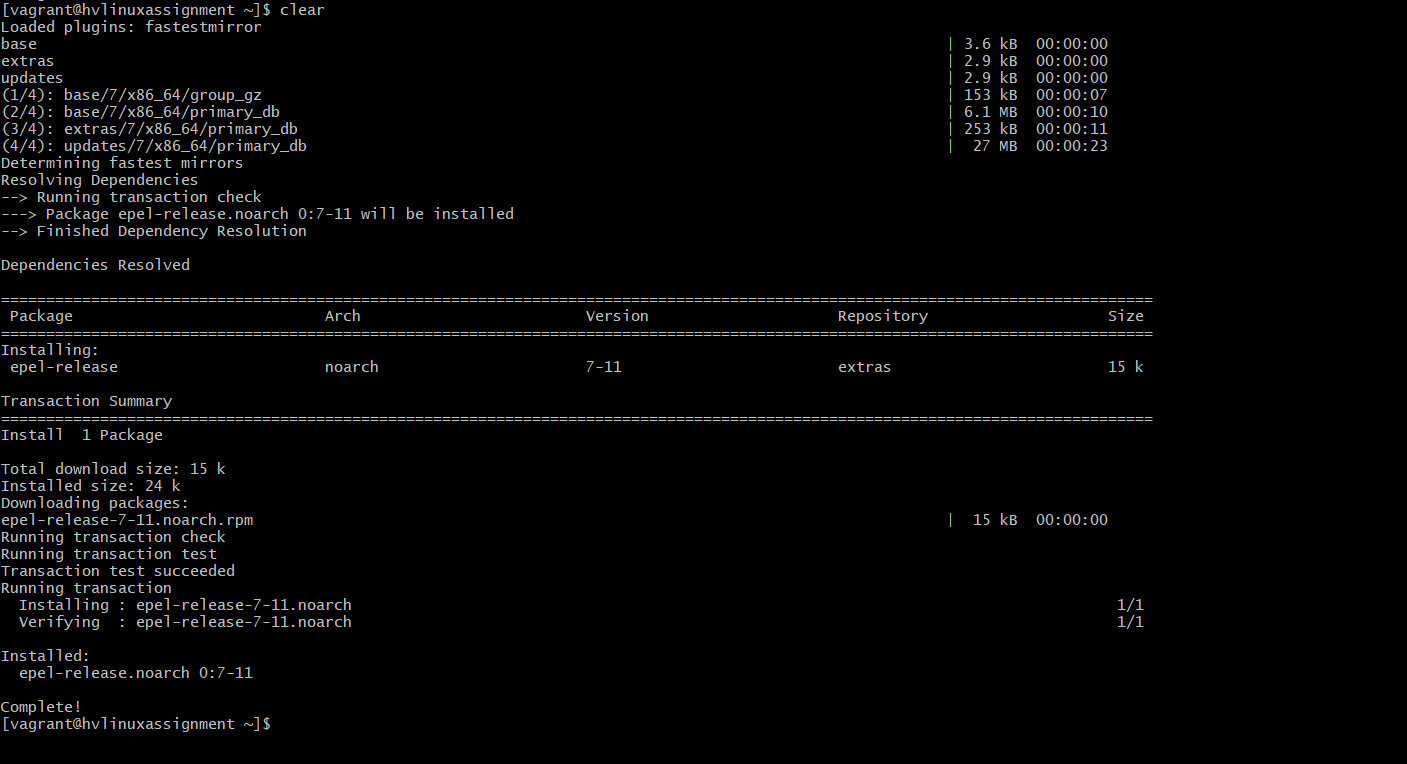
1. Install and configure monitoring tools (htop or nmon) to monitor CPU, memory, and process usage.
2. Set up disk usage monitoring to track storage availability using df and du.
3. Implement process monitoring to identify resource-intensive applications.
4. Create a basic reporting structure (e.g., save outputs to a log file for review).

**Task 1: System Monitoring Setup**

1. Install and configure monitoring tools (htop or nmon) to monitor CPU, memory, and process usage.

Commands to install :

sudo yum install epel-release -y



sudo yum install htop -y



sudo yum install nmon -y

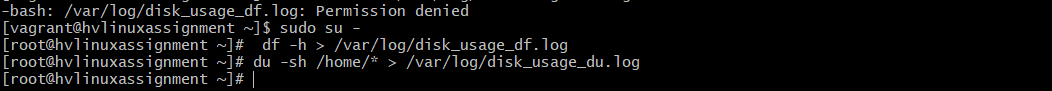


1. Disk usage monitoring

Commands to save the disk usage/directory usage into different files and monitor the usage.

df -h > /var/log/disk\_usage\_df.log

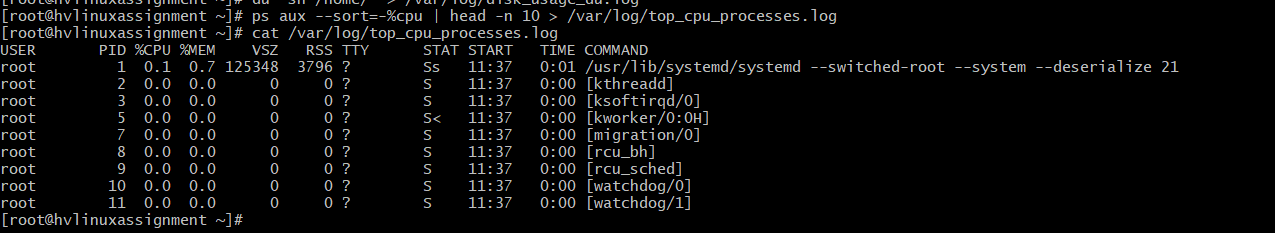
du -sh /home/\* > /var/log/disk\_usage\_du.log



As my log files are stored in the root path, switched to root user.

1. Identifying the CPU usage processor.

ps aux --sort=-%cpu | head -n 10 > /var/log/top\_cpu\_processes.log



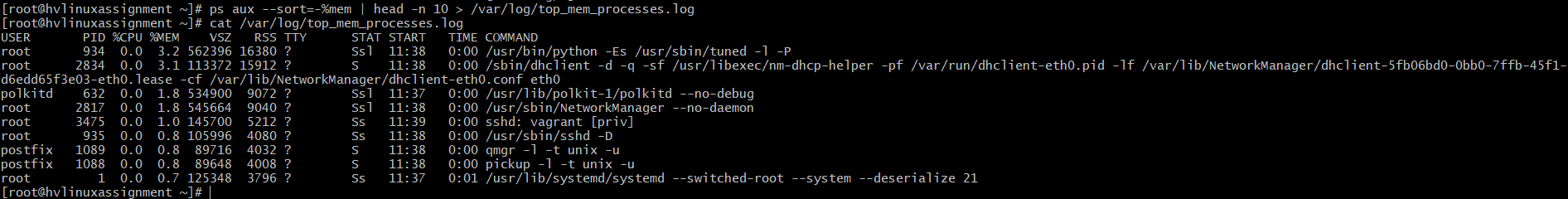
ps aux command used to list the process running

--sort is performing the sort on based on percentage

Head –n 10 will return top 10 process.

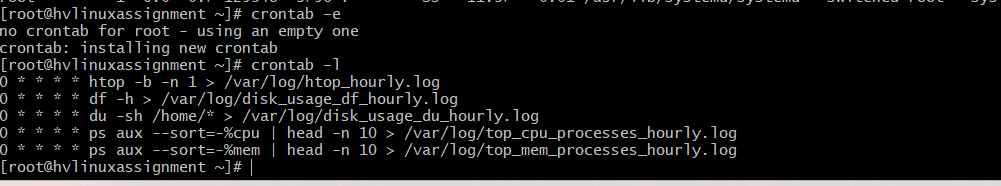
The results stored into the top\_cpu\_processes.log file.

Command : ps aux --sort=-%mem | head -n 10 > /var/log/top\_mem\_processes.log



Similar to the cpu command, now replacing the cpu with mem to get the top 10 memory usage processes.

Step 4: automate the above process, such that any developer working on the machine would notify.



0 \* \* \* \* htop -b -n 1 > /var/log/htop\_hourly.log

0 \* \* \* \* df -h > /var/log/disk\_usage\_df\_hourly.log

0 \* \* \* \* du -sh /home/\* > /var/log/disk\_usage\_du\_hourly.log

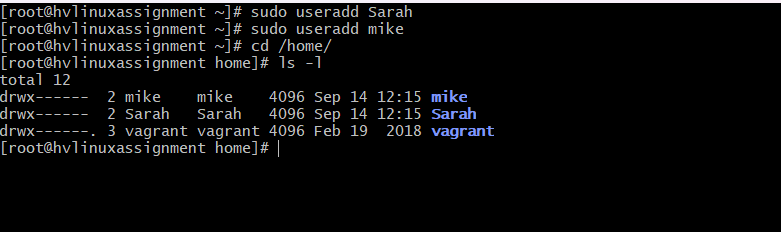
0 \* \* \* \* ps aux --sort=-%cpu | head -n 10 > /var/log/top\_cpu\_processes\_hourly.log

0 \* \* \* \* ps aux --sort=-%mem | head -n 10 > /var/log/top\_mem\_processes\_hourly.log

0 \* \* \* \* => Run the command at minute 0 of every hour. So it triggers at 12:00, 1:00, 2:00, ..., all the way to 23:00.

**Task 2: User Management and Access Control**For the user management, let us create our 2 developers, Sarah and Mike.  
command to create a user : sudo useradd <username>

As shown in the below screenshot we have created “Sarah” and “mike”

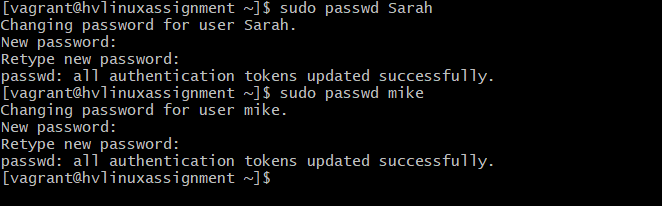


Once the users are created, create a password for the users to do initial login.

Ensure that login password would be alteast 12 digits and has one uppercharacter, one lower character, one special character, and one numerical.

Command to create password: sudo passwd <username>

As shown, in the screenshot the passwords has been created.



Once the users and passwords has been created.  
Now let’s create the workspace directory with restricted permissions.

Command to create the directory : mkdir –p /home/<username>/workspace

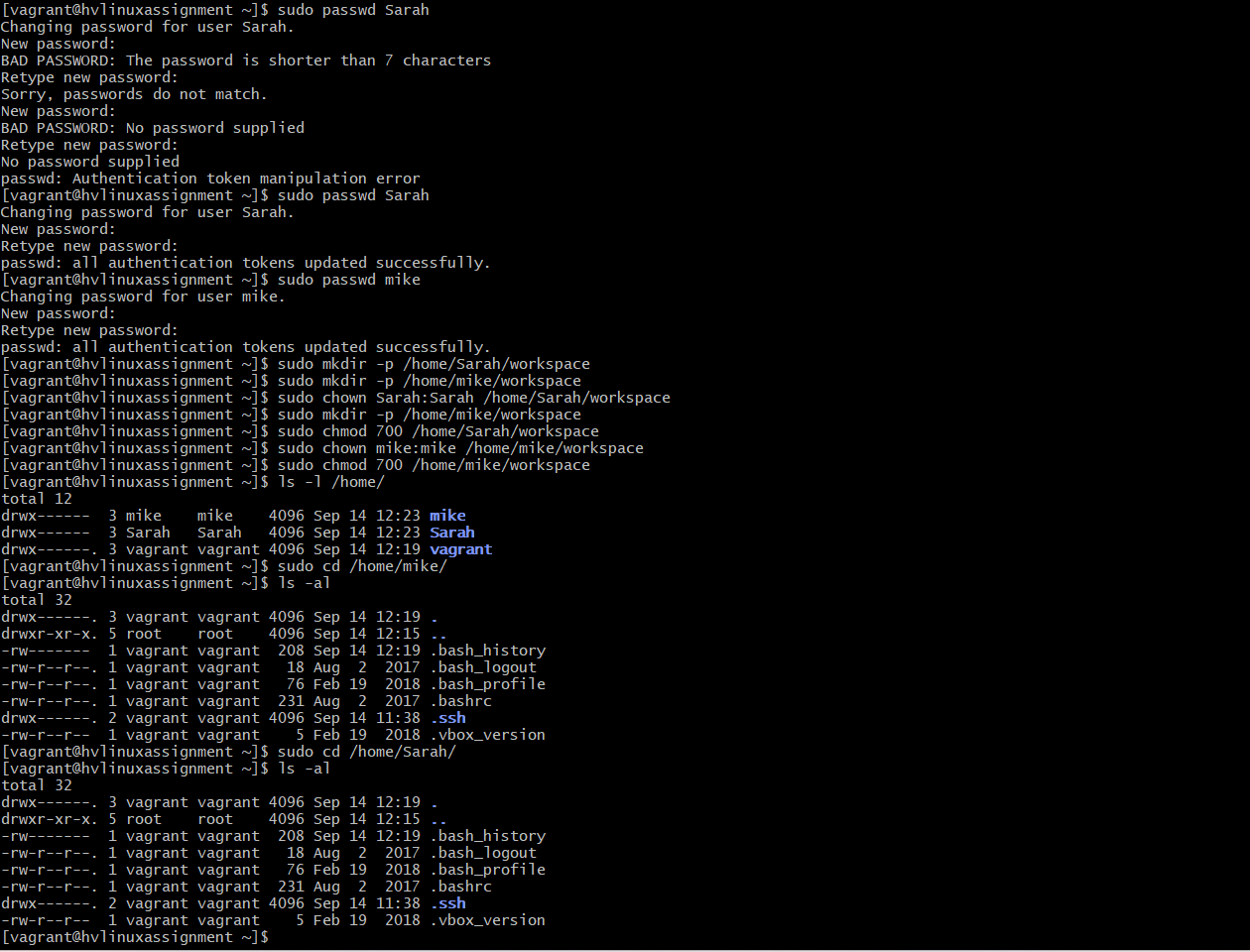
This creates the workspace directory in respective users account.

Update the permission for restricted access.  
  
sudo chown <username>:<username> /home/<username>/workspace

sudo chmod 700 /home/<username>/workspace

The first command change the owner for the workspace folder, set  
the **user owner** and **group owner** to the specified username.

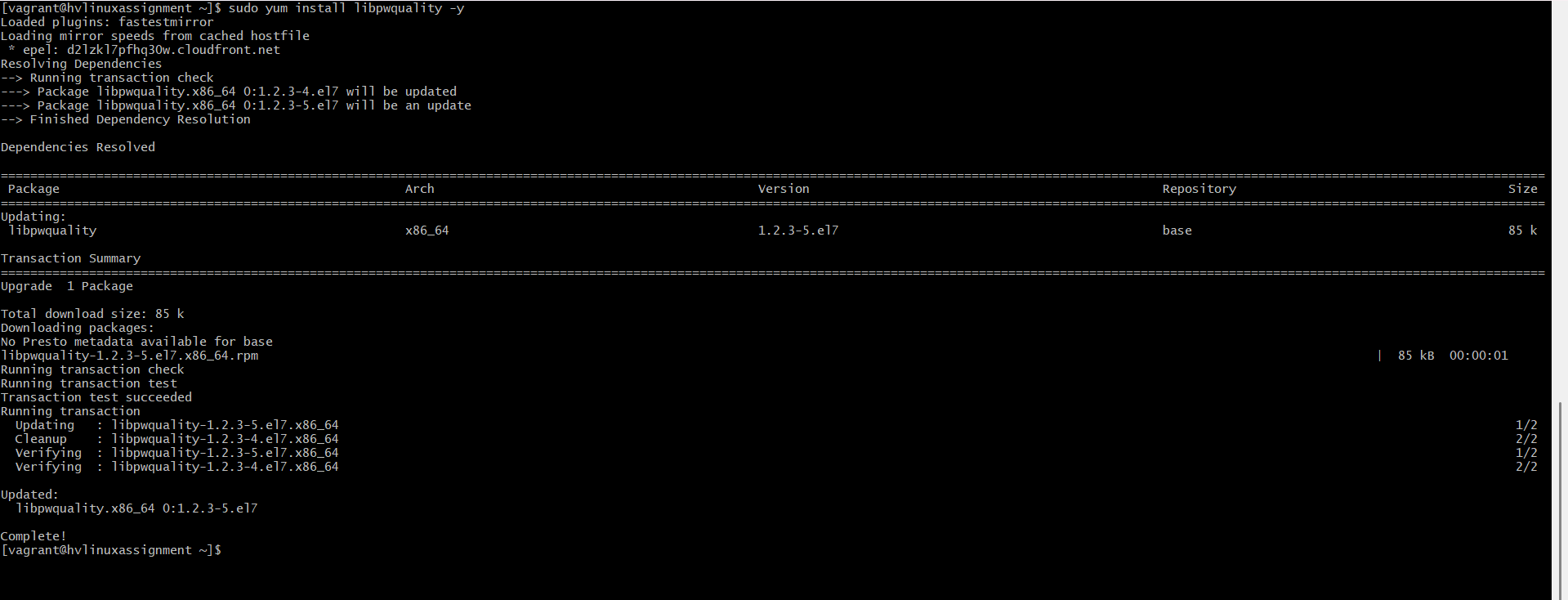
and the second command allows the only the specific user to allow have access. **Owner** can read, write, and execute (rwx), **Group** and **Others** have **no access** (---)



Once the restricted permission has been provided.  
Now enforce the password policy.

Step 3: Enforce Password Policy

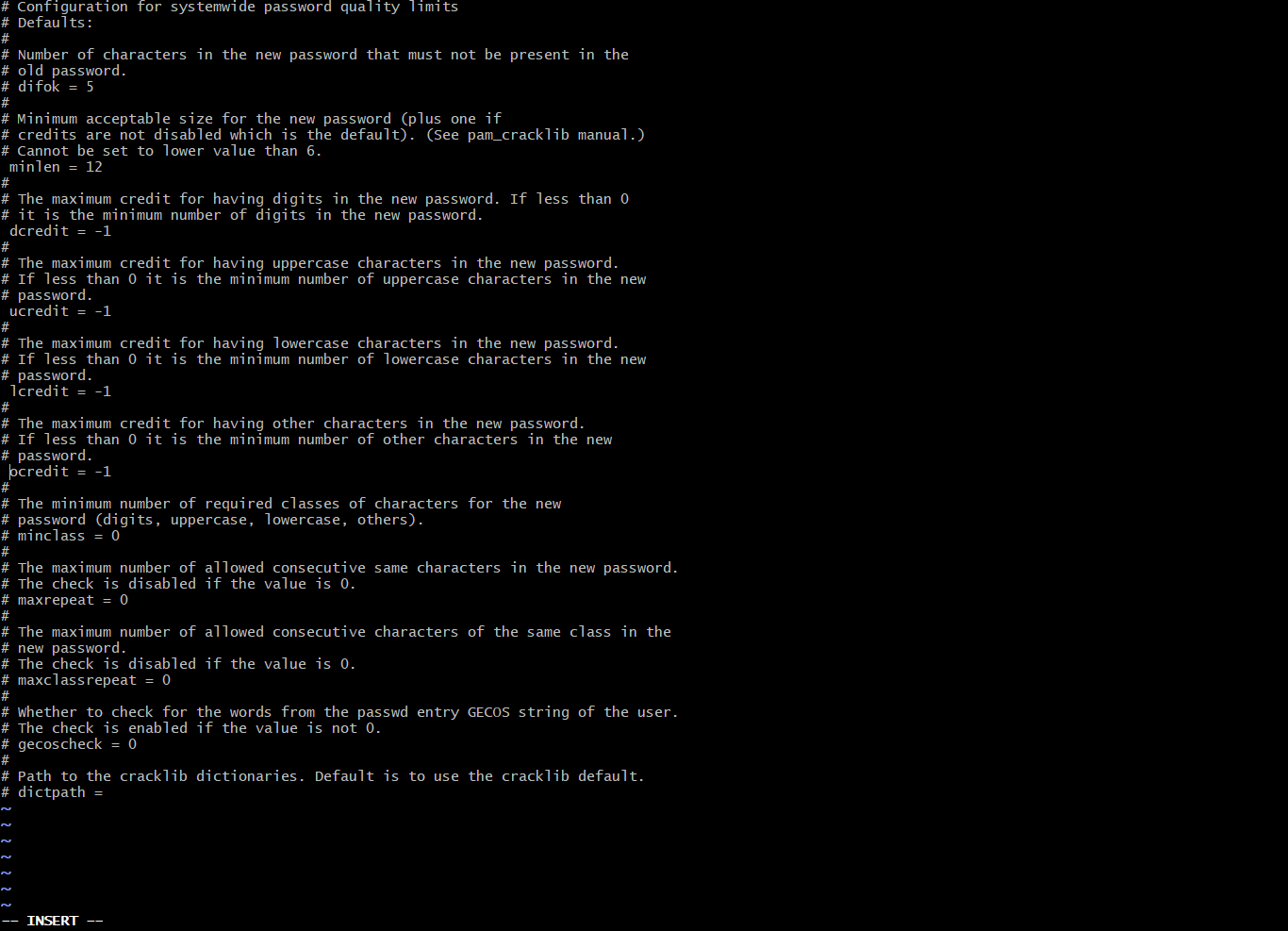
Install libpwquality for complexity enforcement



now, configure the password complexity,

Enforces at least one digit, uppercase, lowercase, and special character.

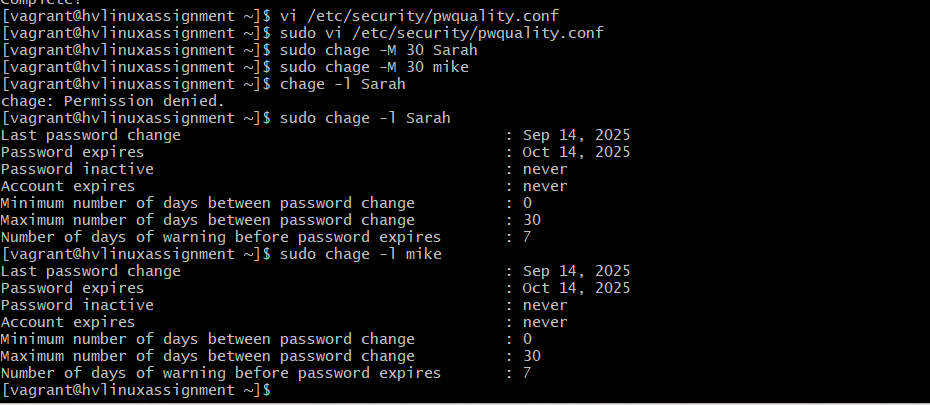
As show in below picture, complexity can be enabled by uncommenting and providing the below values as shown.



Once the complexity is set, now we can set the password expiration time.  
command to set : sudo chage –M 30 <username>

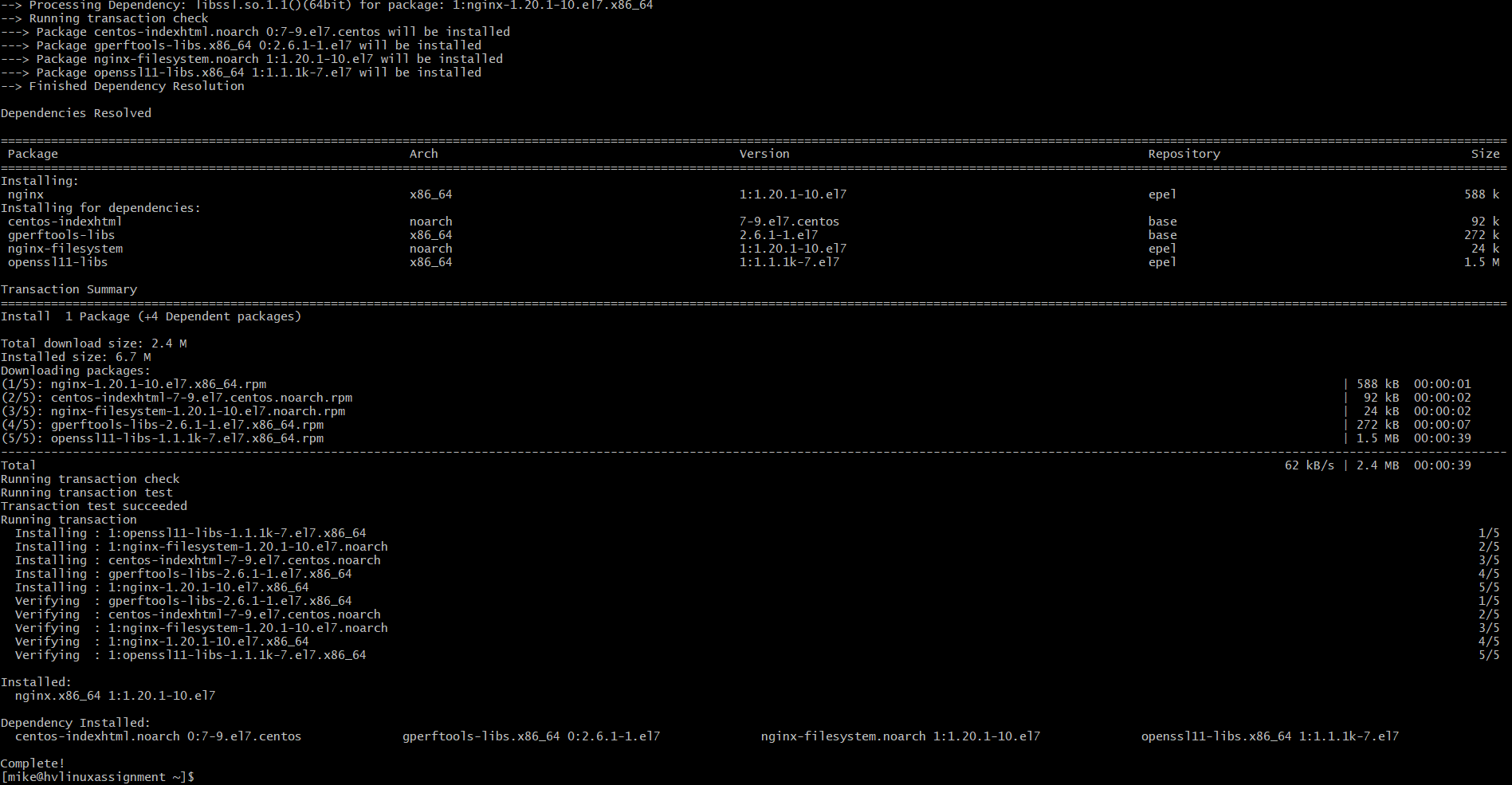
This command tells that password will expire in 30days.  
  
Now check the info on password change with below command.  
sudo chage –l <username>

As shown below, password expiry has been set to Oct 14, which is 30 days from today.



**Task 3: Backup Configuration for Web Servers**

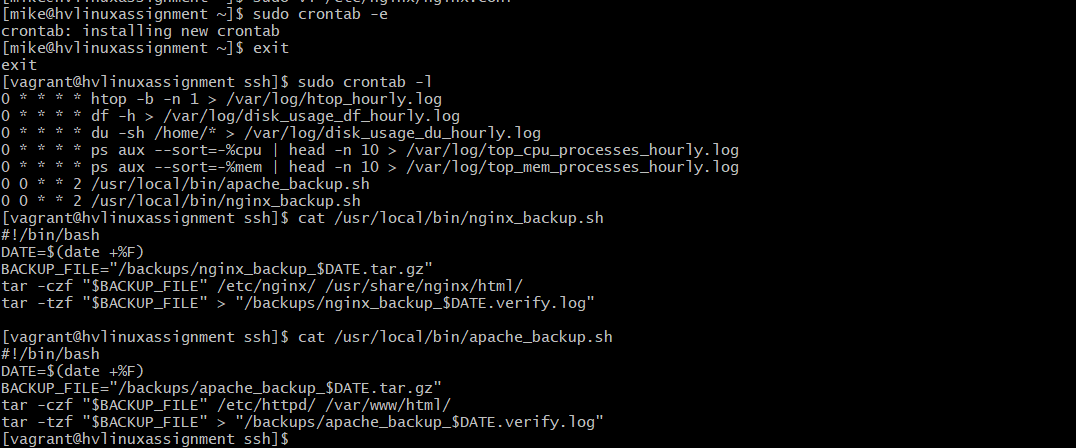
Login with user mike and install nginx, similarly do the same for Sarah and install apache

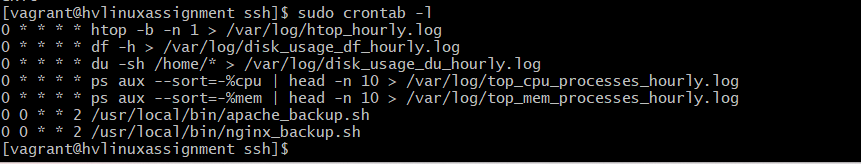
Commands for install nginx :   
sudo yum install -y epel-release  
sudo yum install -y nginx  
  


Enable nginx  
  
commands : sudo systemctl start nginx

  
  
ensure that configuration are /usr/share/nginx/html

similarly install the apache.

Backup script to be added, once the issue of the server is resolved.  


As shown in the picture, the backup scripts are enabled and added.  
  
  
the cron jobs for the Runs every **Tuesday at 12:00 AM**

 \* \* 2 → Day of week = 2 (Tuesday)  
  
Verify the logs :  
  
I have manually run the commands for the back scripts.  
  
here is the screenshot for the logs.  
