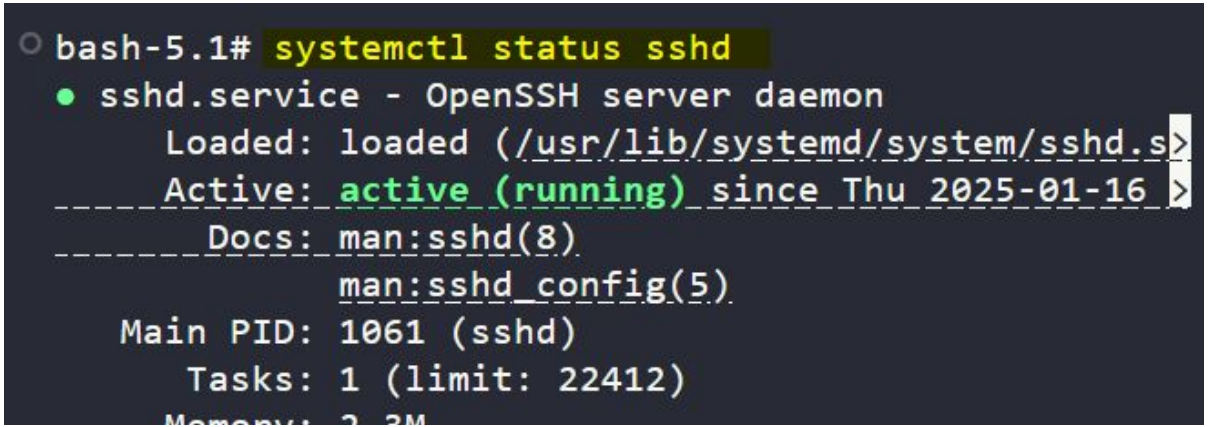
RHCSA.GURU

Solutions : Mock Test 2

Task 1 - Check if the system has running sshd service then configure the sshd to enable password authentication.

Solution - 1- Use systemctl status sshd to ensure the SSH server is active. If not, start it with systemctl start sshd



RHCSA.GURU

2- Edit the sshd configuration file /etc/ssh/sshd_config using vi editor.

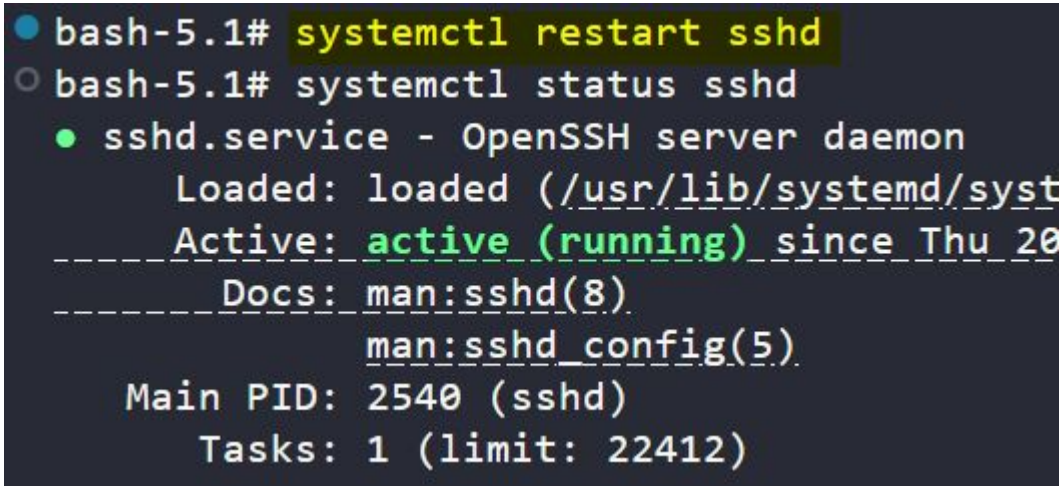
```
bash-5.1# vi /etc/ssh/sshd_config
```

3- Locate the PasswordAuthentication directive and set it to yes to allow password-based logins. Save the file and exit from editor.

```
#       PermitTTY no
#       ForceCommand cvs server
PasswordAuthentication yes
~
:wq!
```

RHCSA.GURU

4- Restart the SSH service with systemctl restart sshd to apply the changes.

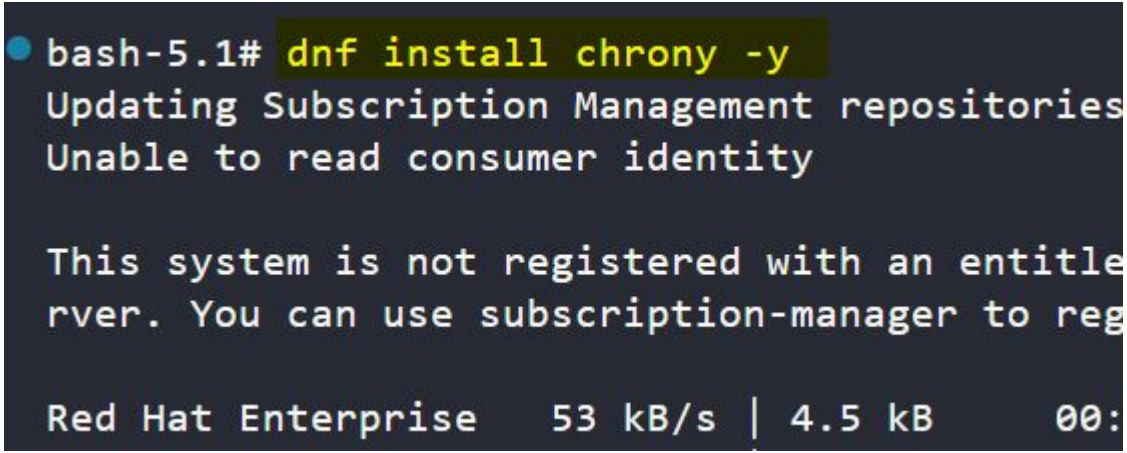


RHCSA.GURU

Task 2 - Set up an NTP server to synchronize its clock with the server at 'time.google.com'.

Solution -

1- Install Chrony:



RHCSA.GURU

2- Using vi editor, edit /etc/chrony.conf to specify time.google.com for synchronization.

```
# server 169.254.169.123 prefer iburst minpoll 4 max
poll 4
server time.google.com iburst
# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (https://www.pool
.ntp.org/join.html).

# Use NTP servers from DHCP.
sourcedir /run/chrony-dhcp
```

Note - make sure to comment the default server

RHCSA.GURU

3- Restart the service to apply changes.

```
bash-5.1# systemctl restart chronyd
bash-5.1# systemctl status chronyd
● chronyd.service - NTP client/server
     Loaded: loaded (/usr/lib/systemd/system/chrony>
     Active: active (running) since Thu 2025-01-16 >
       Docs: man:chronyd(8)
```

4- Verify that your system syncs with the specified servers using chrony sources -v command.

```
bash-5.1# chronyc sources
MS Name/IP address         Stratum Poll Reach LastRx
 Last sample
===============================================================
^* time1.google.com              1   6    17    37
+8270ns[  +90us] +/-  989us
bash-5.1#
```

RHCSA.GURU

Task 3 - With root access on a RHEL 9 system, create the user Micky and assign password 'rhel@123'. Then, construct a script named rhcsa that shows Welcome to user Micky when anyone log in as Micky.

Solution - 1- Create the user and set a password:

```
bash-5.1# useradd Micky
bash-5.1# echo 'rhel@123' | sudo passwd --stdin Micky
Changing password for user Micky.
passwd: all authentication tokens updated successfully
.
bash-5.1#
```

RHCSA.GURU

2- Create a script named rhcsa:

- /usr/local/bin is used for custom scripts to make them globally accessible and executable from anywhere
- $LOGNAME holds the username of the logged-in user, allowing personalized messages.
- chmod 755 grants full permissions to the owner and read/execute permissions to others for the file /usr/local/bin/rhcsa.

```
bash-5.1# echo -e '#!/bin/bash\n echo "Welcome to user
  $LOGNAME"' > /usr/local/bin/rhcsa
bash-5.1# chmod 755 /usr/local/bin/rhcsa
bash-5.1#
```

RHCSA.GURU

3- Add the script path to .bashrc to ensure it runs on login

```
bash-5.1# echo '/usr/local/bin/rhcsa'
  >> /home/Micky/.bashrc
bash-5.1#
```

4- Verify it by logging into Micky user.

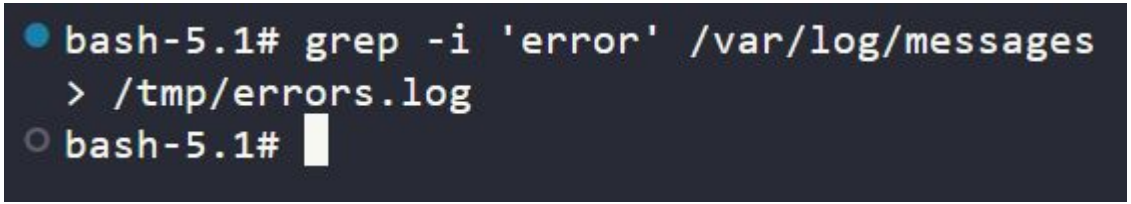
```
bash-5.1# su - Micky
Last login: Thu Jan 16 13:15:30 UTC 2
025 on pts/0
Welcome to user Micky
[Micky@ip-172-31-36-195 ~]$
```

RHCSA.GURU

Task 4 - Extract all rows containing the word 'error' from the /var/log/messages file and save them to a file named 'errors.log' in the /tmp directory in same order as they appear in the original file.

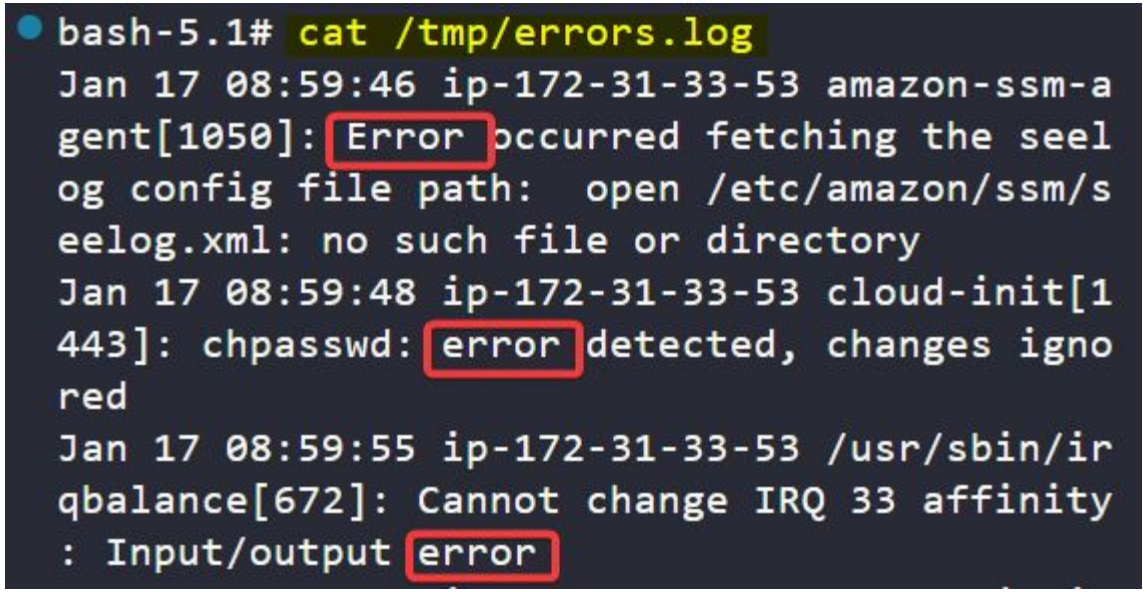
Solution - 1- Extract and save rows containing 'error':

```
grep -i 'error' /var/log/messages > /tmp/errors.log
```



RHCSA.GURU

2-Verify by displaying the content of errors.log



RHCSA.GURU

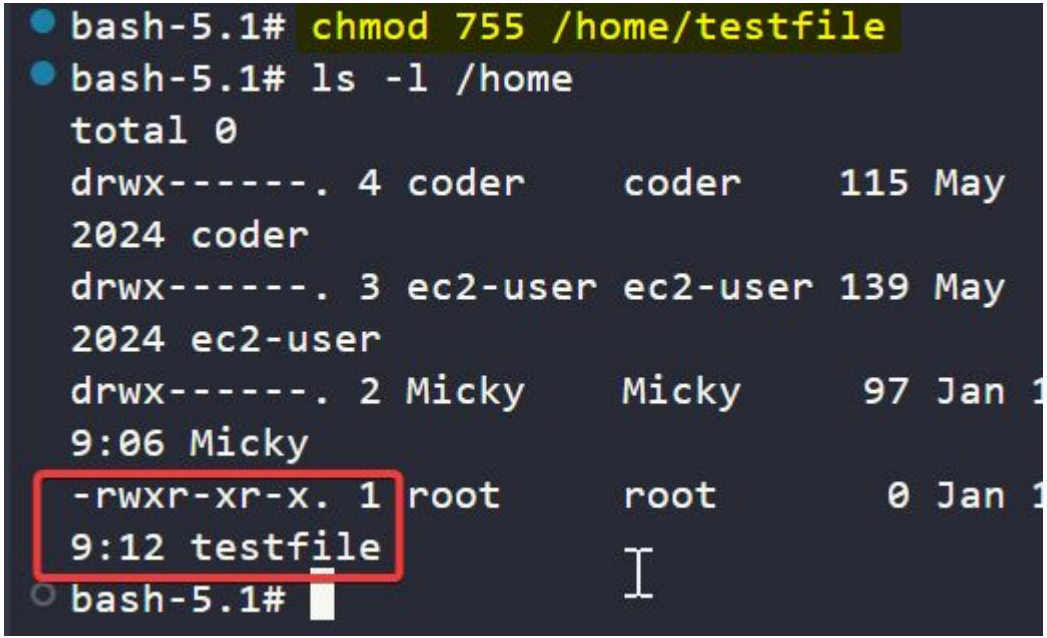
Task 5 - Create a new file in /home directory named 'testfile' and give read-write-execute permissions to the owner and read-execute permissions to the group and others.

Solution - 1- Create the file and check its permission using -l option with ls command

```
bash-5.1# touch /home/testfile
bash-5.1# ls -l /home
total 0
drwx------. 4 coder    coder    115 May  2
2024 coder
drwx------. 3 ec2-user ec2-user 139 May  2
2024 ec2-user
drwx------. 2 Micky    Micky     97 Jan 17 0
9:06 Micky
-rw-r--r--. 1 root     root       0 Jan 17 0
9:12 testfile
```

RHCSA.GURU

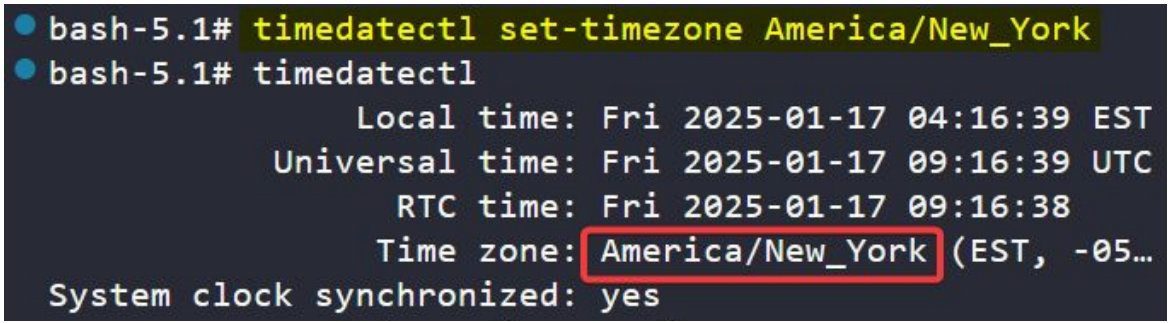
2- Gives read-write-execute (7) permissions to the owner and read-execute (5) permissions to the group and others using chmod command



RHCSA.GURU

Task 6 - Set the system timezone to America/New_York via timedatectl; afterwards, archive and compress the /var/tmp directory into /root/test.tar.gz

Solution - 1- timedatectl set-timezone America/New_York sets the system's timezone to America/New_York. timedatectl confirms the active system timezone.



RHCSA.GURU

2- Use tar -czf /root/test.tar.gz /var/tmp creates a compressed tarball of the /var/tmp directory.

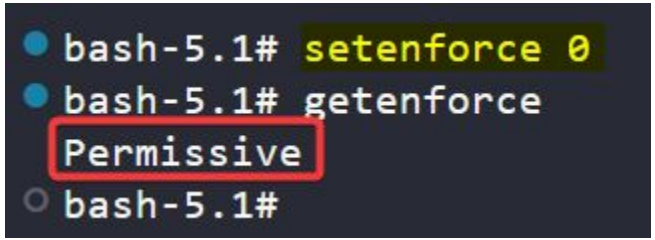
- **-c**: Creates a new archive.
- **-f**: Specifies the filename for the archive.
- **-z**: Compresses the archive using gzip.

Use ls to checks the existence of file

RHCSA.GURU

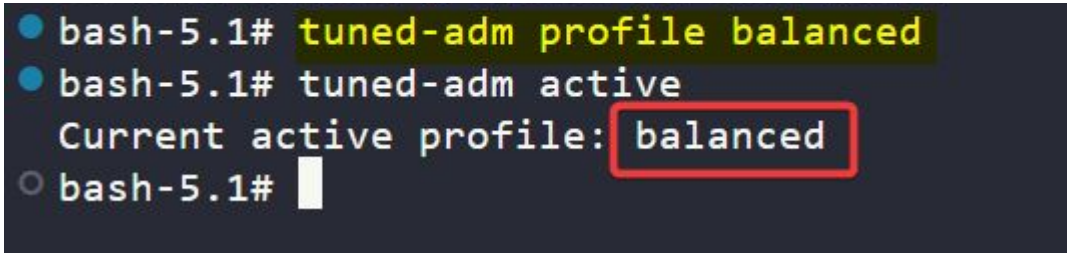
Task 7 - Configure SELinux (Security-Enhanced Linux) to operate in 'Permissive' mode and change the active tuned profile to balanced.

Solution - 1- setenforce 0 changes SELinux to Permissive mode, where it logs violations without enforcing policies. Use getenforce to verify.



RHCSA.GURU

2- Run sudo tuned-adm profile balanced to apply the 'balanced' profile, optimizing both performance and power usage.
Use tuned-adm active to check the currently active tuned profile.



RHCSA.GURU

Task 8 - Create a user 'coder' if it doesn't exist and then configure the system for 'coder' to have a password which expires in 90 days and must be at least 10 characters long.

Solution - 1- chage -M 90 coder sets the password for 'coder' to expire after 90 days, with the -M option defining the maximum expiration period.

```
bash-5.1# chage -M 90 coder
bash-5.1#
```

RHCSA.GURU

2- Modify the /etc/security/pwquality.conf file by changing minlen to 10 for setting the minimum password length to 10 characters. Use vi editor to edit and save the file .

```
bash-5.1# vi /etc/security/pwquality.conf
```

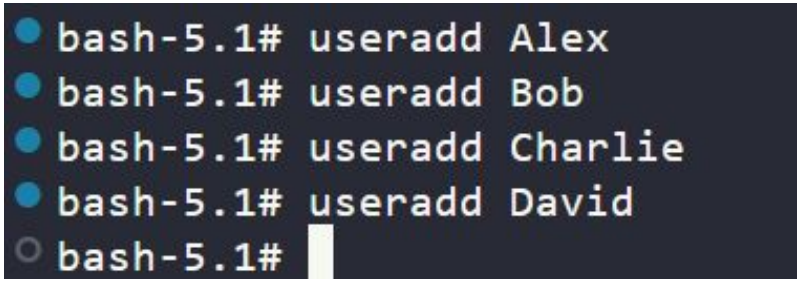
```
# Cannot be set to lower value than 6.
 minlen = 10
#
# The maximum credit for having digits i
word. If less than 0
```

RHCSA.GURU

Task 9 - Using the provided RHEL system, complete the following tasks:

- Create Users named 'Alex', 'Bob', 'Charlie' and 'David'.
- Create a group named 'developers' and add 'Alex' & 'Bob' as secondary members.
- Create a group named 'managers' and add 'Charlie' & 'David' as secondary members.
- Add members to group in same order as mentioned

Solution - 1- Create users 'Alex', 'Bob', 'Charlie', and 'David' using useradd command.



RHCSA.GURU

2- Create 'developers' group using groupadd command and adds 'Alex' and 'Bob' as members.

```
bash-5.1# groupadd developers
bash-5.1# usermod -aG developers Alex
bash-5.1# usermod -aG developers Bob
bash-5.1#
```

3- Create ‘managers’ group using groupadd command and adds 'Charlie' and 'Davis' as members.

```
bash-5.1# groupadd managers
bash-5.1# usermod -aG managers Charlie
bash-5.1# usermod -aG managers David
bash-5.1#
```

RHCSA.GURU

Task 10 - Create a group named 'friends' and add 'Alex', 'Bob'(create users if not present) as secondary members. Create a file named 'friendsCircle' in the /home directory and assign the access of the file to 'friends' group only, make Alex the owner of the file and friends group as the group owner.

Solution -

1. Create the 'friends' group and adds 'Alex' and 'Bob' as secondary members .

```
bash-5.1# groupadd friends
bash-5.1# usermod -aG friends Alex
bash-5.1# usermod -aG friends Bob
bash-5.1#
```

RHCSA.GURU

2- Creates the 'friendsCircle' file using touch, assigns ownership with chown to 'Alex' as the owner and 'friends' as the group, and sets permissions with chmod 770, meaning full access for the owner and group, and no access for others.

```
bash-5.1# touch /home/friendsCircle
bash-5.1# chown Alex:friends /home/friendsCircle
bash-5.1# chmod 770 /home/friendsCircle
bash-5.1#
```