RHCSA.GURU

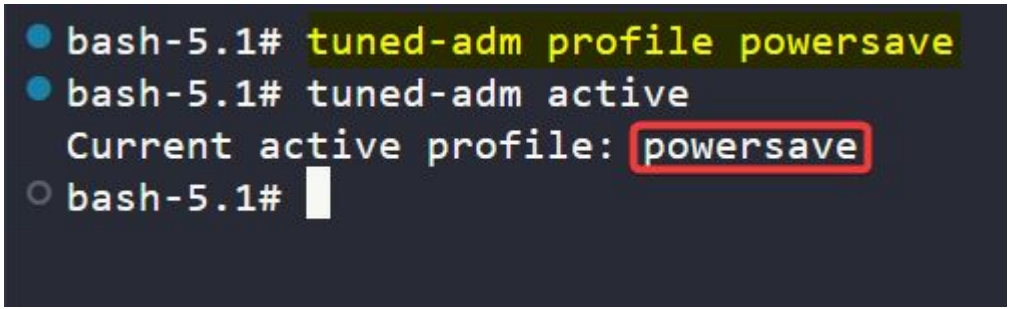
Solutions : Mock Test 5

Task 1 - Set the tuned profile for your system to powersave.

Solution -

1- Run tuned-adm profile powersave to activates the "powersave" profile, optimizing the system for reduced power consumption.

2 - Use tuned-adm active to check the currently active tuned profile.



RHCSA.GURU

Task 2 -Create a shared directory named /home/admins and ensure it has the following characteristics: the directory should belong to the adminuser group, and members of this group should have read and write access to the directory. Additionally, any files created within /home/admins should automatically inherit the adminuser group as the group ownership.

Solution -

1. Create the directory /home/admins.

```
bash-5.1# mkdir -p /home/admins
bash-5.1#
```

2. Create the group adminuser if it doesn't already exist.

```
bash-5.1# groupadd adminuser
bash-5.1#
```

RHCSA.GURU

3. Change the group ownership of /home/admins to adminuser.

```
bash-5.1# chown :adminuser /home/admins
bash-5.1#
```

4. Set the permissions to allow read/write for the group.

```
bash-5.1# chmod 770 /home/admins
bash-5.1#
```

5. Set the setgid (Set Group ID) on the directory so that files created in this directory will automatically inherit the group adminuser.

```
bash-5.1# chmod g+s /home/admins
bash-5.1#
bash-5.1# ls -ld /home/admins
drwxrws---. 2 root adminuser 6 Jan 24 18:37 /home/ad
mins
bash-5.1#
```

Task 3 - Create a shell script named multilines.sh in the /home/coder directory. The script should use a for loop to generate three lines, each containing the text test1, test2, and test3 on separate lines.

Solution - 1- Create a script named multilines in /home/coder directory using vi editor

```
bash-5.1# vi  /home/coder/multilines.sh
bash-5.1#
```

2- Add the following script content that uses a for loop to print test1, test2, and test3 on separate lines and save the file after changes

```
#!/bin/bash
for i in test1 test2 test3
do
  echo $i
done
```

RHCSA.GURU

3- Make the script executable to ensure it can be run

```
bash-5.1# chmod +x /home/coder/multilines.sh
bash-5.1#
```

4- Run the script to verify the output, which should display the three lines with the text test1, test2, and test3:

```
bash-5.1# /home/coder/multilines.sh
test1
test2
test3
```

RHCSA.GURU

Task 4 - Format the two disks, nvme1n1 and nvme2n1, using Linux RAID auto-formatting. Then, create a RAID array named /dev/md0 and mount this RAID array to the directory /raid1.

Solution - 1- Ensure that mdadm is installed, which is required for managing RAID arrays:

```
bash-5.1# dnf install mdadm -y
Updating Subscription Management repositories.
Unable to read consumer identity
```

2- Use mdadm to create a RAID 1 array (/dev/md0) with the two disks (/dev/nvme1n1 and /dev/nvme2n1). The --assume-clean option is used to assume that the disks are already formatted

Command: mdadm --create /dev/md0 --assume-clean --level=1 --raid-devices=2 /dev/nvme1n1 /dev/nvme2n1

RHCSA.GURU

```
bash-5.1# mdadm --create /dev/md0 --assume-clean --l
evel=1 --raid-devices=2 /dev/nvme1n1 /dev/nvme2n1

mdadm: Note: this array has metadata at the start an
d
```

3- Create the directory /raid1 where the RAID array will be mounted and create a filesystem on the RAID array.

```
bash-5.1# mkdir /raid1
bash-5.1#
bash-5.1# mkfs.ext4 /dev/md0
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 1309440 4k b
```

4- Mount the RAID array (/dev/md0) to the /raid1 directory:

```
bash-5.1# mount /dev/md0 /raid1
bash-5.1#
```

RHCSA.GURU

3- Verify the RAID array's status by checking /proc/mdstat:

```
bash-5.1# cat /proc/mdstat
Personalities : [raid1]
md0 : active raid1 nvme2n1[1] nvme1n1[0]
      5237760 blocks super 1.2 [2/2] [UU]
```

4- To ensure the RAID array is automatically mounted on boot, add it to the /etc/fstab file:

```
bash-5.1# echo '/dev/md0 /raid1 ext4 defaults 0 0' >
 > /etc/fstab
```

RHCSA.GURU

Task 5 - Add a new environment variable named EXAM with the value redhat. Ensure that this variable is available for all users during remote login sessions.

Solution - 1- Add the environment variable EXAM to the /etc/environment file.

```
bash-5.1# echo 'EXAM=redhat' >> /etc/environment
bash-5.1#
```

2- To make sure the variable is available for all users during remote login sessions, ensure the /etc/profile file sources the /etc/environment file.

```
bash-5.1# vi /etc/profile
bash-5.1#
```

RHCSA.GURU

Add the following line to /etc/profile and save the changes

```
# /etc/profile
if [ -f /etc/environment ]; then
    . /etc/environment
fi
```

3- After modifying /etc/profile, reload the file to apply the changes to the current session and verify the environment variable EXAM

```
bash-5.1# source /etc/profile
bash-5.1# echo $EXAM
redhat
bash-5.1#
```

RHCSA.GURU

Task 6 - Create a new user named Tom and configure the system to restrict SSH access so that only the user Tom is permitted to connect.

Solution - 1- Create the new user Tom and set a password for the user Tom to ensure they can log in

```
bash-5.1# useradd Tom
```

```
bash-5.1# passwd Tom
Changing password for user Tom.
New password:
```

2- Edit the SSH configuration file `/etc/ssh/sshd_config` to restrict SSH access:

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

RHCSA.GURU

3- In the SSH configuration file, add the following line and save the changes

- AllowUsers: This directive restricts SSH login to specific users. By setting it to Tom, only the user Tom will be able to log in via SSH.

```
# sshd_config(5) for more inf
.
AllowUsers Tom
# This sshd was compiled with
```

4- Restart the SSH service to apply the change

```
bash-5.1# systemctl restart sshd
bash-5.1#
```

RHCSA.GURU

Task 7 - Using Stratis, create a pool named redhat on the disk /dev/nvme1n1, create a filesystem named rhcsa, mount it to a directory named /guru, and then create a snapshot of the filesystem named rhcsa-snap.

Solution - 1- Install Stratis if not already installed:

```
bash-5.1# dnf install stratis-cli
Updating Subscription Management repo
sitories.
```

2- Start and enable the stratis service:

```
bash-5.1# systemctl start stratisd
bash-5.1# systemctl enable stratisd
```

RHCSA.GURU

Note: Remove RAID1 and free up the disks (follow steps 3 and 4 if task 4 is done; otherwise, skip step 3 and 4).

3- Unmount the RAID Array and stop the RAID device /dev/md0

```
bash-5.1# umount /raid1
bash-5.1# mdadm --stop /dev/md0
```

4- Erase the RAID metadata from both disks to free them for other uses:

```
bash-5.1# mdadm --zero-superblock /dev/nvme1n1
bash-5.1# mdadm --zero-superblock /dev/nvme2n1
bash-5.1# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
nvme0n1     259:0    0   30G  0 disk
├─nvme0n1p1 259:1    0    1M  0 part
├─nvme0n1p2 259:2    0  200M  0 part /boot/efi
├─nvme0n1p3 259:3    0  600M  0 part /boot
└─nvme0n1p4 259:4    0 29.2G  0 part /
nvme2n1     259:5    0    5G  0 disk
nvme1n1     259:6    0    5G  0 disk
```

5- Create the Stratis pool named 'redhat' on the disk /dev/nvme1n1

```
bash-5.1# stratis pool create redhat
/dev/nvme1n1
bash-5.1#
```

6- Create a filesystem named 'rhcsa' on the 'redhat' pool:

```
bash-5.1# stratis filesystem create r
edhat rhcsa
bash-5.1#
```

RHCSA.GURU

7-Create /guru directory and mount the 'rhcsa' filesystem to the /guru directory:

```
bash-5.1# mkdir -p /guru
bash-5.1# mount -t xfs /dev/stratis/r
edhat/rhcsa /guru
bash-5.1#
```

8- Add the mount entry to /etc/fstab for persistent mounting:

```
bash-5.1# echo "/dev/stratis/redhat/r
hcsa /guru xfs defaults 0 0" >> /etc/
fstab
bash-5.1#
```

RHCSA.GURU

9- Create a snapshot of the 'rhcsa' filesystem named 'rhcsa-snap'

```
bash-5.1# stratis filesystem snapshot redhat rhcsa rhcsa-snap
```

10- Verify the snapshot:

```
bash-5.1# stratis filesystem list
 Pool     Filesystem   Total / Used / Free / Limit            Created
                       UUID
 redhat   rhcsa        1 TiB / 545 MiB / 1023.47 GiB / None   Jan 24 2025 12:31
 s/redhat/rhcsa        731f37a3-3c6b-434e-a853-d3c6a37cfe88
 redhat   rhcsa-snap   1 TiB / 545 MiB / 1023.47 GiB / None   Jan 24 2025 12:36
 s/redhat/rhcsa-snap   a5f0218f-fa77-4749-9225-1178303e20a4
bash-5.1#
```

RHCSA.GURU

Task 8 - Set up an FTP server on a RHEL system. Ensure that it allows anonymous users to download files from the `/var/ftp/pub` directory. Add a file named `testfile.txt` inside `/var/ftp/pub` and verify that an anonymous user can download it using FTP.

Solution - 1- Install the FTP server (vsftpd):

```
bash-5.1# dnf install vsftpd -y
Updating Subscription Management repositories.
Unable to read consumer identity
```

2- Start and enable the vsftpd service:

```
bash-5.1# systemctl start vsftpd
bash-5.1# systemctl enable vsftpd
```

RHCSA.GURU

3- Configure vsftpd to allow anonymous access:

```
bash-5.1# vi /etc/vsftpd/vsftpd.conf
bash-5.1#
```

```
# Allow anonymous FTP? (Beware - allowed by default if y
ou comment this out).
anonymous_enable=YES
#
```

4- Create the /var/ftp/pub directory if it doesn't already exist. Add a testfile.txt inside /var/ftp/pub

```
bash-5.1# mkdir -p /var/ftp/pub
bash-5.1# echo "This is a test file for FTP download
" > /var/ftp/pub/testfile.txt
```

5- give necessary permission to testfile

```
bash-5.1# chmod 644 /var/ftp/pub/testfile.txt
bash-5.1#
```

6- Restart the vsftpd service to apply changes and install ftp client to test

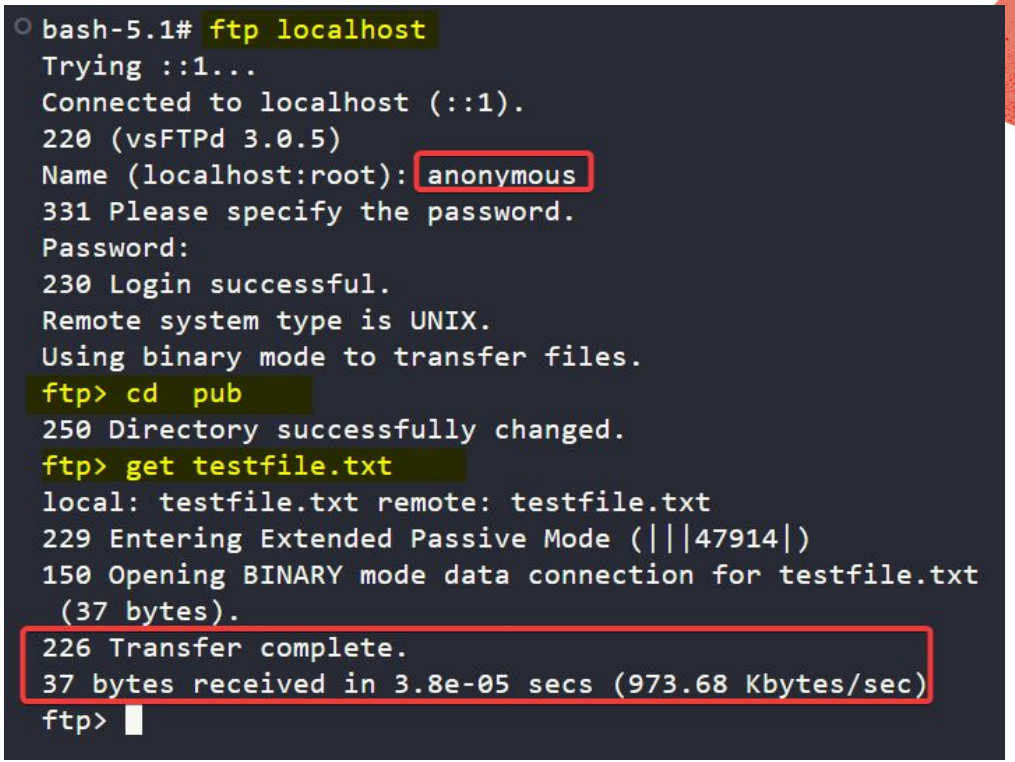
```
bash-5.1# systemctl restart vsftpd
bash-5.1#
```

```
bash-5.1# dnf install ftp -y
Updating Subscription Management repositories.
Unable to read consumer identity
```

RHCSA.GURU

7- Verify that anonymous users can connect and download the test file

- When prompted for a username, enter anonymous.
- Press Enter for the password.
- Once connected, navigate to the pub directory
- Then download testfile.txt using ‘get testfile.txt’



RHCSA.GURU

Task 9 - Configure the system to log all daemon-related messages to a file named daemonlog.log located in the /var/log directory.

Solution - 1- Edit the main rsyslog configuration file:

```
bash-5.1# vi /etc/rsyslog.conf
bash-5.1#
```

2- Add the following line at the end of the file to log all daemon-related messages to /var/log/daemonlog.log:

```
# rsyslog configuration file
daemon.*    /var/log/daemonlog.log

# For more information see /usr/share/doc/
```

RHCSA.GURU

3- Ensure that the /var/log/daemonlog.log file has the appropriate permissions:

```
bash-5.1# touch /var/log/daemonlog.log
bash-5.1# chmod 644 /var/log/daemonlog.log
bash-5.1#
```

4- Restart the rsyslog service to apply the changes and Verify the configuration by checking the contents of the daemonlog.log file

```
bash-5.1# systemctl restart rsyslog
bash-5.1# tail -f /var/log/daemonlog.log
2025-01-24T13:04:28.389124+00:00 ip-172-31-43-28 sys
temd[1]: Stopping System Logging Service...
2025-01-24T13:04:28.884435+00:00 ip-172-31-43-28 sys
```

RHCSA.GURU

Task 10 - Search for all files in the /usr/share directory that are greater than 30KB and less than 50KB. Save the search results into a file called search.txt in the /home/coder/workspace directory.

Solution - 1- **Search the files in /usr/share** using find command with size conditions +30k (greater than 30KB) and -50k (less than 50KB). **Redirect the output** to the file /home/coder/workspace/search.txt.

Command: find /usr/share -type f -size +30k -size -50k > /home/coder/workspace/serach.txt

```
bash-5.1# find /usr/share -type f -size +30k -size -
50k > /home/coder/workspace/serach.txt
bash-5.1#
```

RHCSA.GURU

Explanation of the command:

- **find /usr/share**: Start searching in the /usr/share directory.
- **-type f**: Only consider regular files (not directories or other types).
- **-size +30k**: Search for files that are greater than 30KB.
- **-size -50k**: Search for files that are less than 50KB.
- **> /home/coder/workspace/serach.txt**: Redirect the output (file paths) into the file /home/coder/workspace/serach.txt