RHCSA Exam Practice - Questions

This document contains a series of lab questions for the RHCSA exam. Use these to simulate an exam environment. Both VMs are named respectively servera and serverb, but servera can also sometimes be referred to as client and serverb as server. Use the Vagrantfile provided in this repository to create the VMs with VirtualBox.

Lab 1: Networking (nmtui/nmcli)

Objective: Configure networking and hostnames.

1. Configure the following networking settings:
2. **IP:** 172.16.1.12/24
3. **Gateway:** 172.16.1.1
4. **DNS server:** 172.16.1.1
5. Give it the following hostname: choubiclient.lab

Ensure all changes are persistent across reboots.

Lab 2: Root Password Recovery and Networking

Objective: Reset lost root administrative access on serverb and configure its networking as well.

1. Ensure SELinux labels are corrected before returning to normal boot.
2. Configure the following networking settings:
3. **IP:** 172.16.1.13/24
4. **Gateway:** 172.16.1.1
5. **DNS server:** 172.16.1.1
6. Give it the following hostname: choubiserver.lab

Lab 3: YUM/DNF Repositories

Objective: Configure local software repositories ON BOTH SERVERA and SERVERB.

Configure your VMs to use the following repositories:

1. BaseOS repository: http://choubidoubidouuu.lab/rhel9.0/x86_64/dvd/BaseOS
2. AppStream repository: http://choubidoubidouuu.lab/rhel9.0/x86_64/dvd/AppStream

Requirements: 1. Repositories must be enabled and available, gpg must be disabled. 2. Configuration must persist across reboots. 3. Verify you can list packages from both repositories.

Lab 4: SELinux - Custom Port

Objective: Configure SELinux to allow a service on a non-standard port.

Configure your system so that the Apache HTTP server is able to listen on TCP port 82.

Requirements: 1. Do not disable SELinux or set it to permissive. 2. Do not move or relabel files under /var/www/html. 3. The change must be persistent across reboots.

Lab 5: Users, Groups, and Sudo

Objective: Manage user accounts and administrative privileges.

Perform the following configurations:

1. Create a group named sysadm.
2. Create three users:
   - harry: add to secondary group sysadm.
   - natasha: add to secondary group sysadm.
   - sarah: login shell must be disabled (no interactive shell) and not a member of sysadm.
3. Set the password for all three users to: lalala
4. Members of the sysadm group must be able to create new users via sudo.
5. User harry must be able to change other users' passwords without being prompted for a sudo password.
6. Configuration must be persistent across reboots.

Lab 6: Shared Directory (Setgid)

Objective: Create a collaborative directory with group inheritance.

1. Create /shared/sysadm.
2. Group ownership: sysadm.
3. Only accessible to group members.
4. New files inside must inherit the sysadm group automatically (setgid).
5. Configuration must persist across reboots.

Lab 7: Cron Scheduling

Objective: Schedule a recurring task for a specific user.

- **User:** natasha
- **Task:** Run every 1 minute.
- **Action:** Log the message "Ex200 Testing" using the logger command.
- **Requirement:** Configuration must persist across reboots.

Lab 8: Autofs with NFS

Objective: Configure automatic mounting of remote directories.

On the server (172.16.1.13): * Export the directory /home/guests/netuserX via NFS.

On the client (172.16.1.12): * Configure autofs so that accessing /netdir/netuserX automatically mounts the export from the server. * Create the user netuserX with password choubidou. * The configuration must persist across reboots.

Lab 9: Archive Creation (Tar)

Objective: Create and compress file archives.

- Create /root/myetcbackup.tar.bz2 containing /etc.
- Compression: bzip2 (.tar.bz2).

Lab 10: ACL and Permissions

Objective: Set fine-grained file permissions using Access Control Lists.

1. Copy /etc/fstab to /var/tmp/fstab.
2. Set owner: root, group: root.
3. **Permissions:**
    - No execution bit.
    - natasha: read/write.
    - harry: no access.
    - Others: read only.

Lab 11: Time Synchronization (Chrony)

Objective: Configure NTP client synchronization.

Configure the client (172.16.1.12) to synchronize its time against the server (172.16.1.13). Ensure the configuration persists across reboots.

Lab 12: File Search (Find)

Objective: Locate and process files based on attributes.

1. Find all files created by user natasha and modified during the last 24 hours.
2. Copy them to /root/natashafiles.
3. Keep the directory structure.

Lab 13: Text Filtering (Grep)

Objective: Extract specific content from a file.

1. From /usr/share/dict/words, extract words containing "ich".
2. Save the results to /root/lines.

Lab 14: User Creation (Fixed UID)

Objective: Create a user with specific identification.

Create a user: * **Username:** choubidou * **UID:** 1111 * **Password:** rhcsa-exam

Lab 15: LVM Management

Objective: Create and mount logical volumes.

1. Create a volume group wgroup and a logical volume wshare.
    - **Physical extent size:** 8 MB.
    - **Logical volume size:** 50 extents.
2. Format with ext4 and mount to /mnt/wshare.
3. Ensure it mounts automatically on boot.

Lab 16: Swap Space

Objective: Add and enable additional swap storage.

1. Create a 400 MB swap partition.
2. Enable it and make it persistent across reboots.

Lab 17: Resizing Logical Volumes

Objective: Modify the size of existing LVM volumes.

Resize Logical Volume vo (in vgroup) to around 50 MB (acceptable range: 30–70 MB).

Lab 18: Performance Tuning (Tuned)

Objective: Apply system performance profiles.

Apply the recommended tuned profile on your system.

Lab 19: User Login Message

Objective: Configure custom shell greetings.

Task A (Specific User): When user nico logs in, display the message: Welcome to you, user Nico, you are amazing!

Task B (All Users): Configure the system so that **any** user who logs in (e.g., harry, natasha) receives a dynamic greeting: Welcome [username], you are logged in! *(Replace [username] with the actual name of the user logging in)*

Lab 20: Container image (Podman)

Objective: Build and tag a container image

1. Clone the Text-To-PDF application from GitHub: https://github.com/sachinyadav3496/Text-To-PDF.git
2. Build a container image named text2pdf using the provided Dockerfile.
3. Tag the image as localhost/text2pdf:latest.
4. Verify that the image exists locally.

Lab 21: Pulling and Running Containers

Objective: Run a container from an image.

1. Run a container named mycontainer from image localhost/text2pdf:latest.
2. Mount /opt/file to /data/input and /opt/processed to /data/output.

Lab 22: Container Autostart (Systemd)

Objective: Configure containers to start as services.

1. Run a container as user student named mycontainer, mounting:
   - Host /opt/file to /data/input
   - Host /opt/processed to /data/output
2. Ensure the container autostarts on boot using a systemd user service.
3. Use loginctl enable-linger student to ensure the service runs without an active session.