RHEL 8 Q & A that were removed in the course latest update

PRACTICE EXAM ONE

Question 1:

Assume that you forget the root password. Reset the root password for ServerA. Change it to "password" to gain access to the system.

Note that

- I have listed practice test requirements for each exam in the exam description, which you will see before viewing the Questions. If this does not happen, please notify Udemy support and let me know, and I will send you a copy of the requirements.

Explanation
(RHEL 8)
1. Restart the system.
2. Catch the grub screen.
3. Stop on your kernel line and press "E" to enter the editing mode.
4. Find the line that starts with Linux and at the end of that line replace "**rhgb quiet**" with "**rd.break**"
5. Press **Ctrl+X** to enter emergency mode.
6. To remount the file system as writable, run:
mount -o remount,rw /sysroot
7. To Enter the chroot environment, run:
chroot /sysroot
8. To reset the root password, run:
passwd
9. To enable the SELinux relabeling process on the following system boot, run:
touch /.autorelabel
10. Press "**Ctrl+D** twice" to exit the chroot environment and the maintenance mode.

Question 5:

On ServerA, add the following secondary IPV6 address statically to the connection profile named "myprofile1". Do this in a way that doesn't compromise your existing settings:
IPV6 – fd01::121/64

Explanation
You can check if IPV6 is enabled on your system using any of the following methods:

First method:
Run the following command:
$ sudo sysctl -a | grep ipv6.*disable
A value of 0 indicates that IPv6 is active on your system.
Note that

- The command “sysctl” is used to manipulate kernel parameters at runtime, and the “-a” option shows all kernel parameters. The “grep” command filters the output and looks for lines containing the words "ipv6" and "disable".

Second method: (RHEL 8 Only)
By showing your network interface in the /etc/network-scripts/ directory:
$ cat /etc/sysconfig/network-scripts/ifcfg-enps03

The output should include the following lines:
IPV6INIT=yes //To enable initialization on the interface.
IPV6_AUTOCONF=yes //To accept Router Advertisements (RA’s).
IPV6_DEFROUTE=yes //Indicates that the default IPv6 route is assigned to the interface.
IPV6_FAILURE_FATAL=no //Indicates that the system will not fail even when IPv6 fails.

Note that

- IPv6 is enabled by default on RHEL 8 & 9.

To add the IPV6 to the "myprofile1" profile, run:
nmcli connection modify myprofile1 +ipv6.addresses fd01::121/64 ipv6.method manual
To reload NetworkManager configuration files without restarting the NetworkManager service, run:
nmcli connection reload
To verify, use:
nmcli con sh myprofile1
OR
$ cat /etc/sysconfig/network-scripts/ifcfg-enps03 (on RHEL 8)
//Here, "enps03" is the interface name (ifname)

Question 9:

On ServerA, use /dev/sdc to do the following:

1. Use the appropriate utility to create a 6T "thin_pool".
2. Using "thin_pool" create a thin volume "thin_vol1" with a size of 1T.
3. Create xfs file system on "thin_vol1", then mount it on “/thin_vol1” directory persistently.

Explanation 1
(RHEL 8)
To create a 6T thin provisioned volume using a 10 GB disk, the appropriate utility for doing this is VDO.
Procedure

1. To install vdo, run:
yum install vdo kmod-kvdo -y
2. To start the vdo service, run:
systemctl start vdo
3. To enable a service to start automatically during the system boot, run:
systemctl enable vdo
4. To check the status of the vdo service, run:
systemctl status vdo
5. To list the available block devices, run:
lsblk
6. To create a VDO (Virtual Data Optimizer) volume, run:
vdo create --name=vdo1 --device=/dev/sdc --vdoLogicalSize=6.2T –force
Here's a breakdown of the parameters used in the example command you provided:

- "--name=vdo1": Specifies the name of the VDO volume to be created as "vdo1".
- "--device=/dev/sdc": Specifies the device or disk to be used for storage as "/dev/sdc".
- "--vdoLogicalSize=6.2T": Specifies the logical size of the VDO volume as 6.2 terabytes.
- "--force": Specifies that any existing data on the device should be overwritten without prompting for confirmation.

7. To reboot the system, run:
systemctl reboot
8. To list the files and directories in the directory “/dev/mapper/vdo1” in a long format, run:
ls -l /dev/mapper/vdo1
9. To list the available block devices, run:
lsblk
10. To create a new volume group "vg_thin" using the physical volume "/dev/mapper/vdo1", run:
vgcreate vg_thin /dev/mapper/vdo1
11. To display information about the volume groups in the LVM (Logical Volume Manager) system, run:
vgs
12. To create a thin logical volume pool (thinpool) in a volume group (VG) named "vg_thin", run:
lvcreate -L 6T --thinpool thin_pool vg_thin
The options used in the command are as follows:

- "-L 6T": This specifies the size of the thinpool to be created, in this case, 6 terabytes.
- "--thinpool thin_pool": This sets the name of the thinpool to be created to "thin_pool". The thinpool is used to allocate thin logical volumes.
- "vg_thin": This specifies the volume group in which the thinpool will be created.

13. To verify, run:
lvs
Note that

- VDO is a thinly provisioned block storage target.
- VDO supports any logical size up to 254 times the physical volume size.
- The minimal disk usage for a VDO volume using default settings is approx 4.7 GB.
- The VDO Questions in the practice exams are for RHEL 8. Red hat has deprecated and removed the Python-based VDO management software from RHEL 9 and replaced it

with the LVM-VDO integration that you can use to compress and deduplicate on LVM volumes.

Explanation 2

1. To display information about the logical volumes that have been created in the LVM system, run:

lvs

2. To create a thin logical volume named "thin_vol1" in the "vg_thin/thin_pool" volume group that has a virtual size of 1 terabyte, run:

lvcreate -V 1T --thin -n thin_vol1 vg_thin/thin_pool

The options used in the command are as follows:

- "-V 1T": This specifies the virtual size of the thin logical volume to be created. In this case, the virtual size is 1 terabyte.
- "--thin": This specifies that the logical volume to be created is a thin logical volume.
- "-n thin_vol1": This specifies the name of the logical volume to be created, which in this case is "thin_vol1".
- "vg_thin/thin_pool": This specifies the volume group and thinpool in which the logical volume will be created.

3. To display information about the logical volumes that have been created in the LVM system, run:

lvs

Explanation 3

1. To create a new directory “/thin_vol1”, run:

mkdir /thin_vol1

2. To create an XFS filesystem on the logical volume named "thin_vol1" which belongs to the "vg_thin" volume group, run:

mkfs.xfs -K /dev/mapper/vg_thin-thin_vol1

The options used in the command are as follows:

- "mkfs.xfs": This is the command used to create an XFS filesystem.
- "-K": This option is used to not attempt to discard blocks at mkfs time.
- "/dev/mapper/vg_thin-thin_vol1": This specifies the block device that will be formatted with the XFS filesystem.

3. To watch the udev event queue, and exits if all current events are handled, run:

udevadm settle

The "udevadm settle" command waits for all pending udev events to be processed, ensuring that all devices are properly initialized before continuing. This can be useful in various scenarios where device configuration is important, such as when automating device deployment or when using virtualization technologies.

4. To display information about block devices, and to filter the output to show only information related to "vol1", run:

blkid | grep vol1

5. To open the file “/etc/fstab”, run:

vim /etc/fstab

6. Enter the following line:

/dev/mapper/vg_thin-thin_vol1 /thin_vol1 xfs defaults 0 0

7. Press Esc to switch to command mode, then type :wq followed by Enter to save and exit.
8. To mount all filesystems specified in the /etc/fstab configuration file, run:
mount -a
9. To list the available block devices, run:
lsblk

Question 15:

On ServerA, all user passwords should expire after 90 days and be at least 8 characters in length.

Explanation
(RHEL 8)
1. To edit the system-wide configuration file for login-related settings, run:
vim /etc/login.defs
2. Change PASS_MAX_DAYS to 90
3. Change PASS_MIN_LEN to 8
4. Press Esc to switch to command mode, then type :wq followed by Enter to save and exit.

Question 22:

On ServerA, do the following:
1. Install container-tools.

Explanation 1
(RHEL 8)
To Install container-tools, run:
dnf install @container-tools -y
Or
yum install @container-tools -y
Or
yum module install container-tools -y

Detailed (RHEL 8)
1. Register RHEL: Enter your username and password. The username and password are the same as your login credentials for the Red Hat Customer Portal:
subscription-manager register
2. Subscribe to RHEL:
subscription-manager attach --auto
3. Install the container-tools module:
yum module install -y container-tools

Reference

· https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/building_running_and_managing_containers/index

PRACTICE EXAM TWO

Question 1:

Assume that you forget the root password. Reset the root password for ServerB. Change it to "secret" to gain access to the system.

Explanation
(RHEL 8)
1. Restart the system.
2. Catch the grub screen.
3. Stop on your kernel line and press "E" to enter the editing mode.
4. Find the line that starts with Linux and at the end of that line replace "**rhgb quiet**" with "**rd.break**"
5. Press **Ctrl+X** to enter the emergency mode.
6. To remount the file system as writable, run:
mount -o remount,rw /sysroot
7. To Enter the chroot environment, run:
chroot /sysroot
8. To reset the root password, run:
passwd
9. To enable the SELinux relabeling process on the following system boot, run:
touch /.autorelabel
10. Press "**Ctrl+D** twice".

Question 5:

On ServerB, add the following IPV6 address statically to your current running interface. Do this in a way that doesn't compromise your existing settings:

- IPV6 – fd01::103/64

Explanation
You can check if IPV6 is enabled on your system using any of the following methods:
First method:
Run the following command:
$ sudo sysctl -a | grep ipv6.*disable
A value of 0 indicates that IPv6 is active on your system.
Note that

- The command “sysctl” is used to manipulate kernel parameters at runtime, and the “-a” option shows all kernel parameters. The “grep” command filters the output and looks for lines containing the words "ipv6" and "disable".

Second method: (RHEL 8 Only)
By showing your network interface in the /etc/network-scripts/ directory:
$ cat /etc/sysconfig/network-scripts/ifcfg-enps03

The output should include the following lines:
IPV6INIT=yes //To enable initialization on the interface.
IPV6_AUTOCONF=yes //To accept Router Advertisements (RA’s).
IPV6_DEFROUTE=yes //Indicates that the default IPv6 route is assigned to the interface.
IPV6_FAILURE_FATAL=no //Indicates that the system will not fail even when IPv6 fails.

Note that
- IPv6 is enabled by default on RHEL 8 & 9.

To add the IPV6 to the "enp0s3" interface, run:
1. To display the currently active network connections, run:
nmcli con sh --active
2.
nmcli con mod myprofile2 +ipv6.addresses fd01::103/64 ipv6.method manual
3. To reload NetworkManager configuration files without restarting the NetworkManager service, run:
nmcli connection reload
OR
To bring up “myprofile2” network connection, run:
nmcli con up myprofile2
4. To verify, use:
nmcli dev sh enp0s3
OR
nmcli con sh myprofile2
OR
$ cat /etc/sysconfig/network-scripts/ifcfg-enps03 (on RHEL 8)
//Here, "enps03" is the interface name (ifname)

Question 18:

On ServerB, all user passwords should expire after 100 days and be at least 9 characters in length.

Explanation
(RHEL 8)
1. To edit the system-wide configuration file for login-related settings, run:
vim /etc/login.defs

The "/etc/login.defs" file contains various configuration settings that determine the behavior of the login process, such as password aging policies, password length and complexity requirements, and other security-related settings.
2. Change PASS_MAX_DAYS to 100
3. Change PASS_MIN_LEN to 9
4. Press Esc to switch to command mode, then type :wq followed by Enter to save and exit.

Question 30:

On rhel.server.com, do the following:

1. Install container-tools.

Explanation 1
(RHEL 8)
To Install container-tools, run:
dnf install @container-tools -y
Or
yum install @container-tools -y

PRACTICE EXAM THREE

Question 1:

Assume that you forget the root password. Reset the root password for ServerB. Change it to "passmypass" to gain access to the system.

Explanation
(RHEL 8)
1. Restart the system.
2. Catch the grub screen.
3. Stop on your kernel line and press "E" to enter the editing mode.
4. Find the line that starts with **linux** and at the end of that line replace "**rhgb quiet**" with "**rd.break**"
5. Press **Ctrl+X** to enter the emergency mode.
6. To remount the file system as writable, run:
mount -o remount,rw /sysroot
7. To Enter the chroot environment, run:
chroot /sysroot
8. To set up a new password for the root user, run:
passwd
9. To enable the SELinux relabeling process on the following system boot, run:

touch /.autorelabel
10. Press “**Ctrl+D** twice”.

Question 17:

On ServerB, all user passwords should expire after 60 days and be at least 9 characters in length.

Explanation
(RHEL 8)
1. To edit the system-wide configuration file for login-related settings, run:
vim /etc/login.defs
2. Change PASS_MAX_DAYS to 60
3. Change PASS_MIN_LEN to 9
4. Press Esc to switch to command mode, then type :wq followed by Enter to save and exit.

Question 19:

(RHEL 8)
On ServerB, using the virtual hard disk "sdc", do the following:

1. Create a VDO volume with a size of 100G and the name "vdo_vol".
2. Create an LVM volume group "vdo_vg" on top of the "vdo_vol" VDO volume.
3. Create 2 equally sized logical volumes (vdo_lv1, vdo_lv2) each with a capacity of 49G on top of the “vdo_vg” volume group.
4. Creates an XFS filesystem on top of the vdo_lv1 logical volume.
5. Mount the “vdo_lv1” logical volume persistently on “/vdo_lv1”.

Explanation 1
(RHEL 8)
1. To update the installed packages, run:
dnf update -y
2. To install the vdo and kmod-kvdo packages, run:
dnf install vdo kmod-kvdo -y
3. To enable the vdo service to start automatically during the system boot, run:
systemctl enable vdo --now
4. To list information about all available block devices on the system, run:
lsblk
5. To create a VDO (Virtual Data Optimizer) volume “vdo_vol”, run:
vdo create --name vdo_vol --device=/dev/sdc --vdoLogicalSize=100G –force

- This command creates a VDO volume named "vdo_vol" using the device /dev/sdc with a logical size of 100 GB.
- The "--force" option is used to overwrite any existing metadata on the device.

Note that

- VDO is a block storage optimization technology that can increase storage capacity and improve I/O performance by compressing and deduplicating data on the fly.
-

6. To list information about all available block devices on the system, run:

lsblk
7. To list the details of a device mapper device file named “vdo_vol”, run:
ll /dev/mapper/vdo_vol
8. To display various statistics and information about the VDO (Virtual Data Optimizer) device, run:
vdostats --human-readable
The “--human-readable” option displays the output in a human-readable format, making it easier to read and understand.

Note that

- VDO is a thinly provisioned block storage target.
- VDO supports any logical size up to 254 times the physical volume size.
- The minimal disk usage for a VDO volume using default settings is approx 4.7 GB.
- The VDO Questions in the practice exams are for RHEL 8. Red hat has deprecated and removed the Python-based VDO management software from RHEL 9 and replaced it with the LVM-VDO integration that you can use to compress and deduplicate on LVM volumes.
- Although VDO is not in the EX200v9 exam topics, Red Hat may add it as part of LVM, so I mentioned the steps you would follow to use it.

Steps to create a deduplicated and compressed logical volume on RHEL 9
1. To install the vdo and kmod-kvdo packages, run:
dnf install vdo kmod-kvdo -y
2. Create the VDO LV:
lvcreate --type vdo --name vdo-name --size physical-size --virtualsize logical-size vg-name
3. Create a file system on the VDO LV:
- For the XFS file system:
mkfs.xfs -K /dev/vg-name/vdo-name
- For the ext4 file system:
mkfs.ext4 -E nodiscard /dev/vg-name/vdo-name
4. Mounting an LVM-VDO volume permanently at boot:
- For the XFS file system, add the following line to the /etc/fstab file:
/dev/vg-name/vdo-name mount-point xfs defaults 0 0
- For the ext4 file system, add the following line to the /etc/fstab file:
/dev/vg-name/vdo-name mount-point ext4 defaults 0 0

Reference

- https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html/deduplicating_and_compressing_logical_volumes_on_rhel/creating-a-deduplicated-and-compressed-logical-volume_deduplicating-and-compressing-logical-volumes-on-rhel
- lvmvdo(7) man page

Explanation 2
1. To initialize the physical volume “/dev/mapper/vdo_vol” for use by LVM to be allowed for use in a volume group (VG), run:

pvcreate /dev/mapper/vdo_vol
2. To verify, run:
pvs
3. To create a new volume group named "vdo_vg" that uses the VDO logical volume "/dev/mapper/vdo_vol" as a physical volume, run:
vgcreate vdo_vg /dev/mapper/vdo_vol
4. To verify, run:
vgs

Explanation 3
1. To create a new logical volume (LV) named vdo_lv1 with a size of 49GB and add it to the volume group (VG) named vdo_vg, run:
lvcreate -n vdo_lv1 -L 49G vdo_vg
2. To create a new logical volume named "vdo_lv2" with a size of 49 gigabytes in the volume group "vdo_vg", run:
lvcreate -n vdo_lv2 -L 49G vdo_vg
3. To verify, run:
lvs

Explanation 4
1. To create an XFS file system on “/dev/mapper/vdo_vg-vdo_lv1”, run:
mkfs.xfs -K /dev/mapper/vdo_vg-vdo_lv1
- "-K" no discard

2. To display information about block devices, and to filter the output to show only information related to "lv1", run:
blkid | grep lv1

Note that
- Discard is the process of sending commands (e.g. TRIM) to block devices to indicate blocks that are no longer in use by a mounted file system.
- Sometimes the “mkfs” command issues a block discard command on the disk, but the underlying storage device cannot handle it. So we use the “-K” (Keep) option to disable the block discard at creating time. Also, it lets the command return faster.
- Using the “-K” (keep) option adds some time to “mkfs”, but for most devices, it should not be significant and will lead to better performance and space utilization for solid-state & Thinly-provisioned storage.

Explanation 5
1. To create a new directory “/vdo_lv1”, run:
mkdir /vdo_lv1
2. To edit the file system table file using vim, run:
vim /etc/fstab
3. Add the following line:
/dev/mapper/vdo_vg-vdo_lv1 /vdo_lv1 xfs defaults 0 0
4. Press Esc to switch to command mode, then type :wq followed by Enter to save and exit.
5. To mount all filesystems mentioned in “fstab”, run:

mount -a
6. To list information about all available block devices on the system, run:
lsblk

Note that

- If the VDO volume is on a block device that requires networking, such as iSCSI, add the "_netdev" mount option.

Question 23:

On ServerB, do the following:

1. Install container-tools.

Explanation 1
(RHEL 8)
dnf install @container-tools -y
Or
yum install @container-tools -y

PRACTICE EXAM FOUR

Question 1:

Assume that you forget the root password. Reset the root password for ServerB. Change it to "countersign" to gain access to the system.

Explanation
(RHEL 8)
1. Restart the system.
2. Catch the grub screen.
3. Stop on your kernel line and press "E" to enter the editing mode.
4. Find the line that starts with **linux** and at the end of that line replace "**rhgb quiet**" with "**rd.break**"
5. Press **Ctrl+X** to enter the emergency mode.
6. To remount the file system as writable, run:
mount -o remount,rw /sysroot
7. To Enter the chroot environment, run:
chroot /sysroot
8. To set up a new password for the root user, run:
passwd
9. To enable the SELinux relabeling process on the following system boot, run:
touch /.autorelabel

10. Press “**Ctrl+D** twice”.

Question 5:

On ServerB, add the following IPV6 address statically to your current running interface. Do this in a way that doesn’t compromise your existing settings:

- IPV6 – fd01::105/64

Explanation

You can check if IPV6 is enabled on your system using any of the following methods:

First method:

Run the following command:

$ sudo sysctl -a | grep ipv6.*disable

A value of 0 indicates that IPv6 is active on your system.

Note that

- The command “sysctl” is used to manipulate kernel parameters at runtime, and the “-a” option shows all kernel parameters. The “grep” command filters the output and looks for lines containing the words "ipv6" and "disable".

Second method: (RHEL 8 Only)

By showing your network interface in the /etc/network-scripts/ directory:

$ cat /etc/sysconfig/network-scripts/ifcfg-enps03

The output should include the following lines:

IPV6INIT=yes //To enable initialization on the interface.

IPV6_AUTOCONF=yes //To accept Router Advertisements (RA’s).

IPV6_DEFROUTE=yes //Indicates that the default IPv6 route is assigned to the interface.

IPV6_FAILURE_FATAL=no //Indicates that the system will not fail even when IPv6 fails.

Note that

- IPv6 is enabled by default on RHEL 8 & 9.

To add the IPV6 to the "enp0s3" interface, run:

1. To display the currently active network connections, run:

nmcli con sh --active

2. To configure the IPv6 address and method of the network connection "myprofile4" using the Network Manager Command-Line Interface (nmcli), run:

nmcli con mod myprofile4 +ipv6.addresses fd01::105/64 ipv6.method manual

- "+ipv6.addresses fd01::105/64" adds an IPv6 address "fd01::105/64" to the network connection "myprofile4".
- "ipv6.method manual" sets the IPv6 method for "myprofile4" to "manual" which means that the IPv6 address is manually configured rather than obtained through DHCPv6 or other automatic means.

3. To reload NetworkManager configuration files without restarting the NetworkManager service, run:

nmcli connection reload

OR

To bring up “myprofile4” network connection, run:

nmcli con up myprofile4

4. To verify, use:
nmcli dev sh enp0s3
OR
nmcli con sh myprofile4
OR
$ cat /etc/sysconfig/network-scripts/ifcfg-enps03 (on RHEL 8)
Here, "enps03" is the interface name (ifname)

Question 14:

(RHEL 8)
On ServerB, use the virtual hard disk "sdc" to do the following:

1. Create a VDO volume with a size of 50G and the name "vdo_vol4".
2. Create an LVM volume group "vdo_vg4" below the "vdo_vol4" VDO volume.
3. Create 25G logical volume "vdo_lv4" below the vdo_vg4 volume group.
4. Creates an ext4 filesystem on top of the vdo_lv4 logical volume.
5. Mount the “vdo_lv4” logical volume persistently on “/vdo_lv4”.

Explanation 1
(RHEL 8)
1. To update the installed packages, run:
dnf update -y
2. To install the vdo and kmod-kvdo packages, run:
dnf install vdo kmod-kvdo -y

- vdo – This is a set of Management tools for Virtual Data Optimizer.
- kmod-kvdo – This is a group of Kernel Modules for Virtual Data Optimizer.

3. To enable the vdo service to start automatically during the system boot, run:
systemctl enable vdo --now
4. To list information about all available block devices on the system, run:
lsblk
5. To create a VDO (Virtual Data Optimizer) volume on the device “/dev/sdc” with a logical size of 50GB and the name “vdo_vol4”, run:
vdo create --name vdo_vol4 --device=/dev/sdc --vdoLogicalSize=50G --force
Here, the recommended virtual hard disk size is 5gb (1:10). The “--force” option is used to overwrite any existing data on the device.
6. To list information about all available block devices on the system, run:
lsblk
7. To list the details of a device mapper device file named “vdo_vol4”, run:
ll /dev/mapper/vdo_vol4
8. To display various statistics and information about the VDO (Virtual Data Optimizer) device, run:
vdostats --human-readable
The “--human-readable” option displays the output in a human-readable format, making it easier to read and understand.
Note that

- VDO is a thinly provisioned block storage target.
- VDO supports any logical size up to 254 times the physical volume size.

- The minimal disk usage for a VDO volume using default settings is approx 4.7 GB.
- The VDO Questions in the practice exams are for RHEL 8. Red hat has deprecated and removed the python-based VDO management software from RHEL 9 and replaced it with the LVM-VDO integration that you can use to compress and deduplicate on LVM volumes.
- Although VDO is not in the EX200v9 exam topics, Red Hat may add it as part of LVM, so I mentioned the steps you would follow to use it.

Steps to create a deduplicated and compressed logical volume on RHEL 9
1. To install the vdo and kmod-kvdo packages, run:
dnf install vdo kmod-kvdo -y
2. Create the VDO LV:
lvcreate --type vdo --name vdo-name --size physical-size --virtualsize logical-size vg-name
3. Create a file system on the VDO LV:
- For the XFS file system:
mkfs.xfs -K /dev/vg-name/vdo-name
- For the ext4 file system:
mkfs.ext4 -E nodiscard /dev/vg-name/vdo-name
4. Mounting an LVM-VDO volume permanently at boot:
- For the XFS file system, add the following line to the /etc/fstab file:
/dev/vg-name/vdo-name mount-point xfs defaults 0 0
- For the ext4 file system, add the following line to the /etc/fstab file:
/dev/vg-name/vdo-name mount-point ext4 defaults 0 0

Reference
- https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html/deduplicating_and_compressing_logical_volumes_on_rhel/creating-a-deduplicated-and-compressed-logical-volume_deduplicating-and-compressing-logical-volumes-on-rhel
- lvmvdo(7) man page

Explanation 2
1. To initialize the physical volume “/dev/mapper/vdo_vol4” for use by LVM to be allowed for use in a volume group (VG), run:
pvcreate /dev/mapper/vdo_vol4
2. To verify, run:
pvs
3. To create a new volume group "vdo_vg4" using the physical volume "/dev/mapper/vdo_vol4", run:
vgcreate vdo_vg4 /dev/mapper/vdo_vol4
4. To verify, run:
vgs

Explanation 3
1. To create a new logical volume named "vdo_lv4" with a size of 25 GB in the volume group "vdo_vg4", run:

lvcreate -n vdo_lv4 -L 25G vdo_vg4
2. To verify, run:
lvs

Explanation 4
1. To create a new ext4 file system on the logical volume "vdo_lv4", run:
mkfs.ext4 /dev/mapper/vdo_vg4-vdo_lv4
2. To display information about block devices, and to filter the output to show only information related to "lv4", run:
blkid | grep lv4

Explanation 5
1. To create a new directory "/vdo_lv4", run:
mkdir /vdo_lv4
2. To edit the file system table file using vim, run:
vim /etc/fstab
3. Add the following line:
/dev/mapper/vdo_vg4-vdo_lv4 /vdo_lv4 ext4 defaults 0 0
4. Press Esc to switch to command mode, then type :wq followed by Enter to save and exit.
5. To mount all filesystems mentioned in "fstab", run:
mount -a
6. To list information about all available block devices on the system, run:
lsblk

Question 24:

On test.server.com, do the following:

1. Install container-tools.

Explanation 1
(RHEL 8)
To Install container-tools, run:
dnf install @container-tools -y
Or
yum install @container-tools -y

PRACTICE EXAM FIVE

Question 1:

Assume that you forget the root password. Reset the root password for ServerB. Change it to "mypass" to gain access to the system.

Explanation
(RHEL 8)
1. Restart the system.
2. Catch the grub screen.
3. Stop on your kernel line and press “E” to enter the editing mode.
4. Find the line that starts with **linux** and at the end of that line replace “**rhgb quiet**” with “**rd.break**”
5. Press **Ctrl+X** to enter the emergency mode.
6. To remount the file system as writable, run:
mount -o remount,rw /sysroot
7. To Enter the chroot environment, run:
chroot /sysroot
8. To set up a new password for the root user, run:
passwd
9. To enable the SELinux relabeling process on the following system boot, run:
touch /.autorelabel
10. Press “**Ctrl+D** twice”.

Question 5:

On ServerB, add the following secondary IPV6 address statically to your current running interface. Do this in a way that avoids compromising your existing settings:

- IPV6 – fd01::125/64

Explanation
You can check if IPV6 is enabled on your system using any of the following methods:
First method:
Run the following command:
$ sudo sysctl -a | grep ipv6.*disable
A value of 0 indicates that IPv6 is active on your system.
Note that

- The command “sysctl” is used to manipulate kernel parameters at runtime, and the “-a” option shows all kernel parameters. The “grep” command filters the output and looks for lines containing the words "ipv6" and "disable".

Second method: (RHEL 8 Only)
By showing your network interface in the “/etc/network-scripts/” directory:
$ cat /etc/sysconfig/network-scripts/ifcfg-enps03
The output should include the following lines:
IPV6INIT=yes //To enable initialization on the interface.
IPV6_AUTOCONF=yes //To accept Router Advertisements (RA’s).
IPV6_DEFROUTE=yes //Indicates that the default IPv6 route is assigned to the interface.
IPV6_FAILURE_FATAL=no //Indicates that the system will not fail even when IPv6 fails.
Note that

- IPv6 is enabled by default on RHEL 8 & 9.

To add the IPV6 to the "myprofile5" profile, run:

nmcli connection modify myprofile5 +ipv6.addresses fd01::125/64 ipv6.method manual
To reload NetworkManager configuration files without restarting the NetworkManager service, run:
nmcli connection reload
To verify, use:
nmcli con sh myprofile5
OR
$ cat /etc/sysconfig/network-scripts/ifcfg-enps03 (on RHEL 8)
Here, "enps03" is the interface name (ifname)

Question 13:

On ServerB, use the appropriate utility to do the following on /dev/sdc:

1. Create a 5T thin pool "thin_pool5".
2. Using "thin_pool5" create a thin volume "thin_vol5" with a size of 2T.
3. Create an ext4 file system on "thin_vol5", then mount it on the /thin_vol5 directory persistently.

Explanation 1
(RHEL 8)
1. To install the vdo and kmod-kvdo packages, run:
dnf install vdo kmod-kvdo -y
2. To enable the vdo service to start automatically during the system boot, run:
systemctl enable vdo --now
3. To check the status of the vdo service, run:
systemctl status vdo
4. To list information about all available block devices on the system, run:
lsblk
5. To create a VDO (Virtual Data Optimizer) device named "vdo5" on the device (/dev/sdc), run:
vdo create --name=vdo5 --device=/dev/sdc --vdoLogicalSize=5.2T --force
The “--vdoLogicalSize” option specifies the logical size of the VDO device, which is the maximum amount of data that can be stored on it. The “--force” option is used to force the creation of the VDO device even if it may overwrite existing data on the specified device.
6. To reboot the system, run:
reboot
7. To display a long listing of files and directories in the “/dev/mapper/vdo5” directory, run:
ls -l /dev/mapper/vdo5
8. To list information about all available block devices on the system, run:
lsblk
9. To create a new volume group "vg_thin5" using the physical volume "/dev/mapper/vdo5", run:
vgcreate vg_thin5 /dev/mapper/vdo5
10. To verify, run:
vgs
11. To create a thin provisioned logical volume named “thin_pool5” in the volume group “vg_thin5” with a size of 5 terabytes, run:
lvcreate -L 5T --thinpool thin_pool5 vg_thin5
12. To verify, run:

lvs

Note that

- VDO is a thinly provisioned block storage target.
- VDO supports any logical size up to 254 times the physical volume size.
- The minimal disk usage for a VDO volume using default settings is approx 4.7 GB.
- The VDO Questions in the practice exams are for RHEL 8. Red hat has deprecated and removed the python-based VDO management software from RHEL 9 and replaced it with the LVM-VDO integration that you can use to compress and deduplicate on LVM volumes.
- In RHEL 8, VDO is registered as a physical volume so that it can be managed by LVM. So there is no need to use “pvcreate”, and you can use “vgcreate” directly.

Explanation 2

1. To create a thinly provisioned logical volume named "thin_vol5" with a size of 2 terabytes in the volume group "vg_thin5", run:
lvcreate -V 2T --thin -n thin_vol5 vg_thin5/thin_pool5
The “--thin” option indicates that the logical volume should be thinly provisioned, meaning that it will only consume physical disk space as needed. This allows for more efficient use of available storage space.
The “vg_thin5/thin_pool5” specifies that the thin volume should be created on the thin pool named "thin_pool5" which has already been created.
2. To verify, run:
lvs

Explanation 3

1. To create a new directory “/thin_vol5”, run:
mkdir /thin_vol5
2. To create a new ext4 file system on the logical volume "/dev/mapper/vg_thin5-thin_vol5", run:
mkfs.ext4 /dev/mapper/vg_thin5-thin_vol5
The logical volume is a thinly provisioned volume, which means that it is initially created with minimal space and will expand as needed when data is written to it. Once the file system is created, it can be mounted and used like any other file system in Linux.
3. To wait for all current events to be processed, run:
udevadm settle
The “udevadm settle” command waits for all current events to be processed by udev before exiting. This is useful when working with device management, as it ensures that all relevant devices have been discovered and initialized before proceeding with other operations.
4. To edit the file system table file using vim, run:
vim /etc/fstab
5. To automatically mount the thin volume “thin_vol5” at “/thin_vol5” during system boot, add the following line:
/dev/mapper/vg_thin5-thin_vol5 /thin_vol5 ext4 x-systemd.requires=vdo.service 0 0
The fields in the line have the following meaning:

- **/dev/mapper/vg_thin5-thin_vol5**: specifies the device to mount.
- **/thin_vol5**: specifies the mount point.

- **ext4**: specifies the file system type.
- **x-systemd.requires=vdo.service**: specifies that the VDO service must be started before the file system is mounted.
- **0**: specifies the dump frequency (not used on modern systems).
- **0**: specifies the file system check order (not used on modern systems).

6. Press Esc to switch to command mode, then type :wq followed by Enter to save and exit.
7. To mount all filesystems mentioned in "fstab", run:
mount -a
8. To list information about all available block devices on the system, run:
lsblk

Question 29:

On rhel.server.com, do the following:

1. Install container-tools.

Explanation 1
(RHEL 8)
To Install container-tools, run:
dnf install @container-tools -y
Or
yum install @container-tools -y

PRACTICE EXAM SIX

Question 1:

Assume that you forget the root password. Reset the root password for ServerC. Change it to "watchword" to gain access to the system.

Explanation
(RHEL 8)
1. Restart the system.
2. Catch the grub screen.
3. Stop on your kernel line and press "E" to enter the editing mode.
4. Find the line that starts with **linux** and at the end of that line replace "**rhgb quiet**" with "**rd.break**"
5. Press **Ctrl+X** to enter the emergency mode.
6. To remount the file system as writable, run:
mount -o remount,rw /sysroot
7. To Enter the chroot environment, run:
chroot /sysroot

8. To set up a new password for the root user, run:
passwd
9. To enable the SELinux relabeling process on the following system boot, run:
touch /.autorelabel
10. Press "**Ctrl+D** twice".

Question 6:

On ServerB, add the following IPV6 address statically to your current running interface. Do this in a way that doesn't compromise your existing settings:

- IPV6 – fd01::107/64

Explanation
You can check if IPV6 is enabled on your system using any of the following methods:
First method:
Run the following command:
$ sudo sysctl -a | grep ipv6.*disable
A value of 0 indicates that IPv6 is active on your system.
Note that

- The command "sysctl" is used to manipulate kernel parameters at runtime, and the "-a" option shows all kernel parameters. The "grep" command filters the output and looks for lines containing the words "ipv6" and "disable".

Second method: (RHEL 8 Only)
By showing your network interface in the "/etc/network-scripts/" directory:
$ cat /etc/sysconfig/network-scripts/ifcfg-enps03
The output should include the following lines:
IPV6INIT=yes //To enable initialization on the interface.
IPV6_AUTOCONF=yes //To accept Router Advertisements (RA's).
IPV6_DEFROUTE=yes //Indicates that the default IPv6 route is assigned to the interface.
IPV6_FAILURE_FATAL=no //Indicates that the system will not fail even when IPv6 fails.
Note that

- IPv6 is enabled by default on RHEL 8 & 9.

To add the IPV6 to the "enp0s3" interface, run:
1. To display the currently active network connections, run:
nmcli con sh --active
2. To modify the network connection "myprofile6" to add a new static IPv6 address of "fd01::107/64" to it, run:
nmcli con mod myprofile6 +ipv6.addresses fd01::107/64 ipv6.method manual

- The "+ipv6.addresses" option is used to add the new IPv6 address to the connection.
- The "ipv6.method manual" option specifies that the IPv6 address will be assigned manually rather than using DHCPv6.

3. To reload NetworkManager configuration files without restarting the NetworkManager service, run:

nmcli connection reload
OR
To bring up “myprofile6” network connection, run:
nmcli con up myprofile6
4. To verify, use:
nmcli dev sh enp0s3
OR
nmcli con sh myprofile6
OR
$ cat /etc/sysconfig/network-scripts/ifcfg-enps03 (on RHEL 8)
Here, "enps03" is the interface name (ifname)

Question 15:

On ServerB, use the virtual hard disk "sdc" to do the following:

1. Create a VDO volume with a size of 70G and the name "vdo_vol6".
2. Create an LVM volume group "vdo_vg6" above the "vdo_vol6" VDO volume.
3. Create a 40G logical volume above the vdo_vg6 volume group.
4. Creates an xfs filesystem on top of the “vdo_lv6” logical volume.
5. Mount the “vdo_lv6” logical volume persistently on “/vdo_lv6”.

Explanation 1
(RHEL 8)
1. To update the installed packages, run:
dnf update -y
2. To install the vdo and kmod-kvdo packages, run:
dnf install vdo kmod-kvdo
3. To enable the vdo service to start automatically during the system boot, run:
systemctl enable vdo --now
4. To list information about all available block devices on the system, run:
lsblk
5. To create a virtual data optimizer (VDO) volume on /dev/sdc, with the name “vdo_vol6” and the logical size of 70 gigabytes, run:
vdo create --name vdo_vol6 --device=/dev/sdc --vdoLogicalSize=70G --force
The **--force** option is used to force the creation of the VDO volume even if the device already has data.
6. To list information about all available block devices on the system, run:
lsblk
7. To list the details of a device mapper device file named “vdo_vol6”, run:
ll /dev/mapper/vdo_vol6
8. To display various statistics and information about the VDO (Virtual Data Optimizer) device, run:
vdostats --human-readable
The “--human-readable” option displays the output in a human-readable format, making it easier to read and understand.
Note that

- VDO is a thinly provisioned block storage target.

- VDO supports any logical size up to 254 times the physical volume size.
- The minimal disk usage for a VDO volume using default settings is approx 4.7 GB.
- The VDO Questions in the practice exams are for RHEL 8. Red hat has deprecated and removed the python-based VDO management software from RHEL 9 and replaced it with the LVM-VDO integration that you can use to compress and deduplicate on LVM volumes.
- Although VDO is not in the EX200v9 exam topics, Red Hat may add it as part of LVM, so I mentioned the steps you would follow to use it.

Steps to create a deduplicated and compressed logical volume on RHEL 9
1. To install the vdo and kmod-kvdo packages, run:
dnf install vdo kmod-kvdo -y
2. Create the VDO LV:
lvcreate --type vdo --name vdo-name --size physical-size --virtualsize logical-size vg-name
3. Create a file system on the VDO LV:
- For the XFS file system:
mkfs.xfs -K /dev/vg-name/vdo-name
- For the ext4 file system:
mkfs.ext4 -E nodiscard /dev/vg-name/vdo-name
4. Mounting an LVM-VDO volume permanently at boot:
- For the XFS file system, add the following line to the /etc/fstab file:
/dev/vg-name/vdo-name mount-point xfs defaults 0 0
- For the ext4 file system, add the following line to the /etc/fstab file:
/dev/vg-name/vdo-name mount-point ext4 defaults 0 0

Reference
- https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html/deduplicating_and_compressing_logical_volumes_on_rhel/creating-a-deduplicated-and-compressed-logical-volume_deduplicating-and-compressing-logical-volumes-on-rhel
- lvmvdo(7) man page

Explanation 2
1. To initialize the physical volume “/dev/mapper/vdo_vol6” for use by LVM to be allowed for use in a volume group (VG), run:
pvcreate /dev/mapper/vdo_vol6
2. To verify, run:
pvs
3. To create a new volume group "vdo_vg6" using the physical volume "/dev/mapper/vdo_vol6", run:
vgcreate vdo_vg6 /dev/mapper/vdo_vol6
4. To verify, run:
vgs

Explanation 3

1. To create a new logical volume "vdo_lv6" with a size of 40 gigabytes in the volume group "vdo_vg6", run:
lvcreate -n vdo_lv6 -L 40G vdo_vg6
2. To verify, run:
lvs

Explanation 4
1. To create an XFS file system on "/dev/mapper/vdo_vg6-vdo_lv6", run:
mkfs.xfs /dev/mapper/vdo_vg6-vdo_lv6
2. To display information about block devices, and to filter the output to show only information related to "lv6", run:
blkid | grep lv6

Explanation 5
1. To create a new directory "/vdo_lv6", run:
mkdir /vdo_lv6
2. To edit the file system table file using Vim, run:
vim /etc/fstab
3. Add the following line:
/dev/mapper/vdo_vg6-vdo_lv6 /vdo_lv6 xfs defaults 0 0
4. Press Esc to switch to command mode, then type :wq followed by Enter to save and exit.
5. To mount all filesystems mentioned in "fstab", run:
mount -a
6. To list information about all available block devices on the system, run:
lsblk

Question 29:

On ServerB, do the following:
1. Install container-tools.

Explanation 1
(RHEL 8)
To Install container-tools, run:
dnf install @container-tools -y
Or
yum install @container-tools -y

Detailed (RHEL 8)
1. Register RHEL: Enter your username and password. The username and password are the same as your login credentials for the Red Hat Customer Portal:
subscription-manager register
2. Subscribe to RHEL:
subscription-manager attach --auto
3. Install the container-tools module:
yum module install -y container-tools

Reference

- https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/building_running_and_managing_containers/index