Red Hat Enterprise Linux 8 (RHCSA)

Gursimar Singh Follow
Aug 21 · 29 min read

The intelligent OS for hybrid cloud

RHEL 8 official release by Red Hat Inc, the company behind the Development of Red Hat Enterprise Linux (RHEL) 8 was announced on **May 7, 2019**.

Skill in system administration of Red Hat Enterprise Linux systems is a core competency in the modern data center. Linux is an integral technology enabling the hybrid cloud architecture, across physical servers, virtual machines, private and public cloud computing infrastructures, and containers. Red Hat Enterprise Linux is the de facto standard Linux distribution across all these architectures.

With the release of Red Hat Enterprise Linux 8 (RHEL 8) comes new features and improvements as compared to the predecessor — RHEL 7.

Red Hat Enterprise Linux is the world's leading enterprise Linux platform. Red Hat, Inc. is an American IBM subsidiary software company that provides open source software products to enterprises.

Red Hat Enterprise Linux, a trusted platform for business, and RHEL 8 continues to build on this tradition. Features like System-Wide Encryption Policy, Nftables/firewalld, and Red Hat Insights mean less effort and time spent managing and configuring services, without compromising security needs.

Red Hat Enterprise Linux 8.0 is based on Fedora 28 and upstream kernel 4.18. This provides users with a secure, stable and consistent foundation across hybrid cloud and Data Center deployments with tools needed to support all levels of workloads.

Content Distribution

Red Hat Enterprise Linux 8 has two modes of Content distribution and will only need two repositories enabled.

- **BaseOS repository** — The BaseOS repository provides the underlying core OS content in the form of traditional RPM packages. BaseOS components have a life cycle identical to that of content in previous Red Hat Enterprise Linux releases.
- **AppStream repository** — The Application Stream repository provides all the applications you might want to run in a given userspace. Other software that has special licensing are available on a Supplemental repository.

What is AppStream?

The AppStream allows us to install additional versions of software on independent life cycles and keep your operating system up-to-date while having the right version of an application that suits our use case. Note that no two streams can be installed at the same time into the same userspace.

Desktop Environment

RHEL default Desktop Environment is GNOME. The GNOME Project is supported by the GNOME Foundation. A version of Gnome shipped in RHEL 8 is version 3.28 which has automatic downloading of operating systems in Boxes.

Software Management

RHEL 8 YUM package manager is now based on the DNF technology and it provides support for modular content, increased performance, and a well-designed stable API for integration with tooling. The version of RPM is **4.14.2** and it validates the whole package contents before it starts the installation.

YUM version available in RHEL 8 is **v4.0.4. YUM** based on **DNF** has the following advantages over the previous **YUM v3** used on RHEL 7:

- Increased performance

- Support for modular content
- Well-designed stable API for integration with tooling

Linux containers

RHEL 8 has the enterprise support for Linux containers via a lightweight, open standards-based container toolkit.

Podman is the default command line tool that works on OCI images used for running containers without need for daemon. Podman specializes in all of the commands and functions that help us maintain and modify OCI images, such as `pulling` and `tagging`. It also allows you to `create, run, and maintain` containers created from those images.

Red Hat Certified System Administrator (RHCSA) exam

The performance-based Red Hat Certified System Administrator (RHCSA) exam (EX200) tests your knowledge in areas of system administration common across a wide range of environments and deployment scenarios. The skills tested in this exam are the foundation for system administration across all Red Hat® products.

By passing this exam, you become a Red Hat Certified System Administrator. If you choose to continue your learning journey beyond RHCSA, the credential can also serve as a foundational step on your path toward our highest level of certification — Red Hat Certified Architect.

Prerequisites for EX200

- Have either taken Red Hat System Administration I (RH124) and Red Hat System Administration II (RH134) or the RHCSA Rapid Track course (RH199) that combines those courses, or have comparable work experience as a system administrator on Red Hat Enterprise Linux

Content summary

- Package management with new repository structure and appstream modules
- Create storage devices, volumes, and file systems, including Stratis storage management
- Configure network services and security
- Manage processes, scheduling, and tuning
- Manage users, groups, and authentication
- Perform server management with the Cockpit web management utility
- Troubleshoot and obtain support
- Run containers

Understand and use essential tools

- Access a shell prompt and issue commands with correct syntax
- Use input-output redirection (>, >>, |, 2>, etc.)
- Use grep and regular expressions to analyze text

- Access remote systems using SSH
- Log in and switch users in multiuser targets
- Archive, compress, unpack, and uncompress files using tar, star, gzip, and bzip2
- Create and edit text files
- Create, delete, copy, and move files and directories
- Create hard and soft links
- List, set, and change standard ugo/rwx permissions
- Locate, read, and use system documentation including man, info, and files in /usr/share/doc

Create simple shell scripts

- Conditionally execute code (use of: if, test, [], etc.)
- Use Looping constructs (for, etc.) to process file, command line input
- Process script inputs ($1, $2, etc.)
- Processing output of shell commands within a script
- Processing shell command exit codes

Operate running systems

- Boot, reboot, and shut down a system normally
- Boot systems into different targets manually
- Interrupt the boot process in order to gain access to a system
- Identify CPU/memory intensive processes and kill processes
- Adjust process scheduling
- Manage tuning profiles
- Locate and interpret system log files and journals
- Preserve system journals
- Start, stop, and check the status of network services

- Securely transfer files between systems

Configure local storage

- List, create, delete partitions on MBR and GPT disks
- Create and remove physical volumes
- Assign physical volumes to volume groups
- Create and delete logical volumes
- Configure systems to mount file systems at boot by universally unique ID (UUID) or label
- Add new partitions and logical volumes, and swap to a system non-destructively

Create and configure file systems

- Create, mount, unmount, and use vfat, ext4, and xfs file systems
- Mount and unmount network file systems using NFS
- Extend existing logical volumes
- Create and configure set-GID directories for collaboration
- Configure disk compression
- Manage layered storage
- Diagnose and correct file permission problems

Deploy, configure, and maintain systems

- Schedule tasks using at and cron
- Start and stop services and configure services to start automatically at boot
- Configure systems to boot into a specific target automatically
- Configure time service clients
- Install and update software packages from Red Hat Network, a remote repository, or from the local file system
- Work with package module streams
- Modify the system bootloader

Manage basic networking

- Configure IPv4 and IPv6 addresses
- Configure hostname resolution
- Configure network services to start automatically at boot
- Restrict network access using firewall-cmd/firewall

Manage users and groups

- Create, delete, and modify local user accounts
- Change passwords and adjust password aging for local user accounts
- Create, delete, and modify local groups and group memberships
- Configure superuser access

Manage security

- Configure firewall settings using firewall-cmd/firewalld
- Create and use file access control lists
- Configure key-based authentication for SSH
- Set enforcing and permissive modes for SELinux
- List and identify SELinux file and process context
- Restore default file contexts
- Use boolean settings to modify system SELinux settings
- Diagnose and address routine SELinux policy violations

Manage containers

- Find and retrieve container images from a remote registry
- Inspect container images
- Perform container management using commands such as podman and skopeo
- Perform basic container management such as running, starting, stopping, and listing running containers

- Run a service inside a container
- Configure a container to start automatically as a systemd service
- Attach persistent storage to a container

Let us learn about some of the important concepts:

Shell Scripting

Steve Bourne wrote the Bourne shell which appeared in the Seventh Edition Bell Labs Research version of Unix.
Many other shells have been written; this particular tutorial concentrates on the Bourne and the Bourne Again shells.
Other shells include the Korn Shell (ksh), the C Shell (csh), and variations such as tcsh.

> Here is a detailed article on shell scripting: Shell Scripting. You might have came across the word… | by Gursimar Singh | Medium

Linux Partitions Format

- We cannot delete data permanently from the hard disk specially from the electromagnetic hard disk.
- Whether we delete a 1mb file or a 200GB file, they get deleted in seconds but when we upload the same files it takes a lot of time depending on the size.
- Shred tools help us remove the data to some extent.
- Disk over disk is platter and combinedly they are called Disk.
- Header is a physical device which keeps on moving round-round rather than up-down and it creates holes in HD to store data.
- Once a hole is created then it is almost impossible to undo it. However we can overwrite it by storing new data in those holes of the hard disk.

- When we format internally an inode table is created which keeps the record of files. And when we upload files it goes to the holes or sectors. But at the time of deletion it only deletes the record from the inode table.
- When we format the drive it just removes all the metadata from the inode table and creates a new inode table.
- If we do not create inode table then searching of files will be very difficult.
- Partition is just like report sheet. When we click on the file the system internally knows where is the file with the help of inode table.
- Metadata of 1 partition hard disk requires 16 bytes and the size of the partition table is 64 bytes. That's why we can only create maximum 4 partitions.
- The types of partition are: Primary partition, Extended partition and Logical partition.
- Physical partitions are of two types namely Primary partition and Logical partition.
- It is always recommended to create 4th partition as extended partition so that we can use this partition to make more partitions in it called as logical partitions.
- Data can be stored only in primary partition or logical partition and not in extended partition.
- Minimum size of a partition is 1 sector. The size of 1 sector is 512 bytes.
- We have many format types available differing from OS to OS which creates inode tables in different manner.
- Few of the format types in linux are: ext3, ext4, xfs, zfs etc. Every format type has its own limitations.
- We can use the command #fdisk -l to see all the hard disk partitions of the system.
- When we format a hard disk partition, internally a table is created called inode table.
- When we reformat, a new table is created but data is not removed and thus we can use recovery tools to recovr the data.
- We can use the command #lsblk to see the hard disk with its partitions.
- We can create partition using #mkfs.<format> <partition_name>.
- After creating any partition, we need to mount it to a folder inorder to store data because we can store data using files and folders only in the storage devices.

- We can create and mount a folder by:

Mkdir /<name>

Mount /dev/sdb1 /<name> (mount <partition_name> <folder_to_be_mount>)

- We can unmount using #Unmount /<name> command.
- We can use the #df command to check if the folder is mounted. To convert it to human readable form we can use #df -h command.
- We can load driver for newly created partition in RHEL 8 using the command #udevadm settle.

Webserver and Linux basic commands:

- Web Server is the one that provides the services and clients are those who utilize these services.
- The steps for configuring web server are:
  1. install software
  2. Setup
  3. Execute
- #yum install httpd command to install Apache webserver.
- #systemctl restart httpd command to start webserver.
- #systemctl stop firewalld command to stop firewall.
- #systemctl disable firewalld command to disable firewall.
- rpm command is used for installing, uninstalling, upgrading, querying, listing, and checking RPM packages on your Linux system.

- We can use #dnf list command to see the software status.

- To start or stop the services we use systemctl start/stop <service-name> command.

- We #service command in RHEL 5/6 and we use systemctl in RHEL 7/8

- URL(Uniform Resource Locator) is used to specify addresses of a webpage or file on the internet. We use URL to get the server output.

- Firewall is a network security system that controls incoming and outgoing network traffic.

- We use SSH protocol to connect to linux system and we use RDP to connect to windows system.

Linux and Python Integration

- 1MB = 1000 Kilobytes and 1MiB = 1024 Kilobytes. Computers manage the units in MiB, GiB and not in MB, GB.

- Hard disk doesn't understand MB or GB. It only understands sectors.

- #fdisk -l command shows all the hard disks connected.

- #fdisk -l <hard_disk_name> command shows particular disk information.
  *To get description of partition we use 'p'.
  *To create the new partition we use 'n'.
  *To delete the partition we use 'd'.
  *To quit we use 'q'.
  *To save we use 'w'.

- Demanding storage from hard disk is called partitions.

- In the hard disk, first few sectors(0–2047) are reserved for partition table. 1 MiB space reserved in every hard disk.

- We can create maximum of 4 partitions in a hard disk.

- After creation of partition, its details are stored in partition table. Storing 1 partition takes 16 bytes in partition table.

- The size of partition table is 64 bytes.

- The general partition created on hard disk are known as Primary Partition.

- Always create the 4th partition as Extended partition.

- We can treat the unallocated space in hard disk as a new hard disk and we can create partitions in it. This is known as Extended Partition.
- In Extended Partition, we can’t store data but we create partitions in it known as Logical partitions.
- The minimum size of a partition can be 1 sector i.e, 512 bytes.
- There are 64 partitions(3 primary partitions + 1 extended partition + 60 logical) in which 63 are usable partitions.
- While creating partitions using GUI in windows, we often see that minimum size of partition is 4 MB. This is because the developer of GUI set the constraint to 4 MB for the GUI program.
- When a new device is connected to OS, OS loads its drivers for communicating with it.
- /dev is the folder where we manage all the devices.
- Till RHEL 7, whenever we create a new partition it acts as a new device so we need to load the driver using the command #partprobe /dev/<name_of_HD>.
- For loading drivers we use #udevadm settle command in RHEL8.
- To check the block size and list of partitions we use #lsblk command.

RHEL8 Linux LVM

- LVM (Logical Volume Management) is a concept used to pool storage from different physical storage devices, give a choice to create multiple partitions over the Logical volume and other features to manage the logical volumes.
- Volume group is used to pool the storages from the physical drives and create a new virtual hard disk. It is created using the command #vgcreate <vg_name>

<drive_name_1> <drive_name_2> and so on.

- We create a common node and all the storage(Physical Volumes) are connected to the node(Volume Group).
- Logical Volume is created from the volume group which can also be regarded as storage partition.
- First we need to create a partition then format it and finally mount the storage.
- We create a physical volume using the command #pvcreate <hd_name>. For example, pvcreate /dev/sdc.
- #pvdisplay /dev/sdc command displays the physical volume.
- To create Volume Group (VG) from Physical Volumes, use # vgcreate command and assign name to LG and give names of PV from which you want to create VG. For example, #vgcreate myvg1 /dev/sdb /dev/sdc /dev/sdd command will create VG from 3 PVs.
- #vgdisplay <vg_name> displays the volume group details.
- To create Logical Volume (LV) from VG we use #lvcreate command. For example, #lvcreate — size 10G — name mylv1 myvg1 creates mylv1 (LV) from myvg1 (VG).
- #lvdisplay <vg_name>/lv_name> displays the details of given logical volume.
- After creating LV, we need to format it using #mkfs command. Syntax: mkfs.ext4 <path_of_partition> //can be obtained from lvdisplay and ext4 is the format type.
- Now we need to mount the formatted LV on the folder using #mount command.
- We can extend the partition by:
  a)To extend the LV use #lvextend command. For example, #lvextend — size +5G /dev/myvg1/mylv1.
  b)After extending the size of LV, resize the formatted LV using #resize2fs command. For example #resize2fs /dev/myvg1/mylv1.
- For decreasing the size of the LV, use the command #lvreduce — size -5G <lv_name>.

Here is a blog on linux partitions: Increase or Decrease the Size of Static Partition in Linux | by

Gursimar Singh | Medium

RHEL8 LVM , Fstab

- If one LV is almost full and we need some space then we can either take up the space from another LV or from another hard disk. To take the space from another LV we need to reduce the LV so that space is allocated back to the VG pool. Both logical volumes should be of same volume group.

- Reducing size of one Logical volume will increase size of volume group and from volume group we can increase size of another logical volume.

- To reduce size of logical volume we have to follow five steps:
  a. Unmount the partition using the unmount command.
  b. Use the e2fsck -f command on partition (Ex: e2fsck -f /dev/sdb1) to scan and clean the inode tables for any garbage/bad blocks.
  c. Format the partition using resize2fs command. Basically we are just updating the inode table and saying that your max sector size has been reduced without losing the data.
  Here at this point we have storage/space in our logical volume but according to inode table we have less space.
  d. Reduce the partition using lvreduce command
  e. Mount the partition so that it is live again.

- After these five steps we can increase size of vg using vgextend command and then from volume group we can increase size of logical volume.

- resize2fs is strictly supported for ext4 type of storage space. resize2fs works for both lvextend and lvreduce.

- xfs_growfs is strictly used for xfs type of storage space and it only supports lvextend. It does not support lvreduce. In RHEL8 we use mount point in place of partition name with xfs-growfs command.

- When VG goes to the PV then the blocks are called as physical extends or extents whereas when VG goes to LV then the blocks are called as Logical extends or extents.

- By default the block size is set to 4 MiB. We can also change it using the -s option field. Whenever we ask for storage space say suppose 10 MiB then it will always give us 12 MiB storage as 12 is the multiple of 4 which is the block size.

> Here is a python project to automate LVM: Automating LVM Partition using Python-Script | by Gursimar Singh | Medium

YUM or DNF Configuration on RHEL8

- We can install any software in RHEL 8 using the commands rpm, yum or dnf (DaNdiFied).

- Using rpm we can query about any software using the command #rpm -q <name>, install a software using the command #rpm -i <name>, uninstall a software using the command #rpm -e <name>, etc. — the rpm command doesn't install the software dependencies.

- ISO file works as a DVD in a virtual machine.

- We use yum over rpm because in rpm, we have to manually go to each repo and install whereas yum automatically goes in each repo(folder) and dependency issues are solved by yum.

- We need to configure yum before using.

- We can configure it by:

1. Attach DVD, in Rhel8 there are 2 repos: AppStream & BaseOS, which provide softwares.

2. Go into yum.repos.d folder & create a repo with extension(.repo): #cd /etc/yum.repos.d/

3. Copy the paths to the software folders and give each path a repo id in "[]" brackets. Ex: [dvd1]. Then add "baseurl=file://" before each path to indicate the software is in the local hard disk. and finally add "gpgcheck=0" after each path & save the repo file.

- To check if yum is configured we use the command #yum repolist which shows all the configured repositories in yum.

- With yum we can query of any software with #yum list <name>, install a software using #yum install <name>, uninstall a software using #yum remove <name>, etc.

- dnf is a mature version of yum. dnf is faster than yum. We use it to overcome the drawbacks of yum. If we have configured yum, we can directly use dnf instead of yum, and its commands are the same as yum, instead of using the word 'yum' in commands we replace it with 'dnf'.

- We can use #wget <URL> to download a software from the terminal.

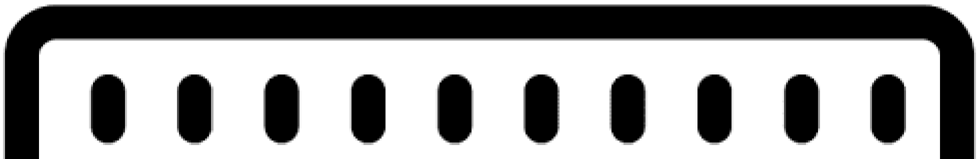
- We can Epel-Release for auto-configuration of the yum repositories. It provides most of the yum repositories available on the internet.

RHEL8 Sudo

- OS can give different privileges to the users created. In servers we disable the root account to make it secure.
- Normal/General user is one who has limited powers and we can create a general user using the command #useradd <user_name>.
- #sudo -l is used to check the privileges given to the user by root account. In servers we never give the all the powers of the available account to anyone account.
- Syntax to add sudo access to user: Username Access_PC/ALL=(Users to allow access/ALL) NOPASSWD: ALL/ usr/bin/cat, usr/bin/id. ID: Program to give the ID of current user.
- NO PASSWD: ALL is used in the configuration file of sudo to disable the feature of asking password as we can't provide the password while doing automation using ansible or python scripts.
- A root user is the one which can perform all the admin tasks and system tasks like changing password, applying changes to the configuration files. The user ID of the root account is 0.
- etc/shadow is a secure file of the system and only root user can read it using the cat command a rule over there for the normal user by writing the name of the user ALL=(ALL). Now the user will be able to read the shadow file.
- For a normal user to read the secure file etc/shadow you can go into the file vim etc/sudoers using the root account and add.
- To run any command with the power of root you can use the sudo program. sudo cat /etc/shadow will help you to read the file with the power of root.
- We can give general user the system/admin powers by configuring the /etc/sudoers file of the sudo program. This concept is called privilege escalation. Sudo stands for superuser do.
- When we try to give extra permission to a general user then it is known as PEA(privilege escalation attach).
- #sudo id will give the id of root and not of the user which is using it.
- [username] [escalation power] [path of the commands] is written in the configuration file of sudo to give certain privileges to the user.

- visudo is the program that opens the sudoers file in debug mode. visudo is more powerful than gedit, vim editor by writing visudo we can go directly into the etc/sudoers.
- If there is any mistake in the file we can correct it easily because visudo tells us the kind of error, if any before saving, which we can edit by the help of visudo by writing e.
- We can create groups in which there can be a number of users and we can give them the power of cat command by going into the etc/sudoers file.
- We can also provide combined permission to the group of users. We use % symbol to specify that it is a group. Syntax: %<group_name> ALL=(ALL) ALL.
- We can create groups as a way to manage members. We can give all members the privileges at the same time.
- #cat /etc/group is used to check the members of a group. [wheel] is already a group inbuilt in sudo.
- Using SSH if we login by #ssh <ip__address>. By default, it logs into the root account in the remote system.
- #ssh -l name_of_user ip, will run the command with the power of the general user.
- #ssh -l name_of_the_user ip sudo id, will show error so you need to create a virtual terminal for this which is called the pseudo terminal.
- So we can use #ssh -t -l name_of_the_user ip sudo id. By running this command we get 2 prompts for running this command, one for login password and the other for using sudo id.
- Pseudo terminal is the virtual terminal to get the sudo password from the user, to get the pseudo terminal we use '-t ' flag along with ssh command.
- SSH is preferred to use with NOPASSWD command in sudoers files, or give -t option, else sudo will fail.

Linux SSH server



- Whenever we copy any data, the copied data is always stored in the clipboard. This data is extracted using the xclip software.
- The use of xclip command to prevent the copy of our data into memory element and if we are having any remote login to other laptop then we can easily get the things that he has copied using the mouse.
- We use the command #xclip -o to read the clipboard. -o is used to get output on screen. We can copy the data with xclip using #xclip -i.
- To install xclip, we use #yum install xclip -y.
- To store the output of some command in xclip we use, command | xclip -i
  Ex: date | xclip -i
- Local execution means the program is present in the same system in which user is present.
- We can run programs in two ways: either by local execution or remote execution.
- Local execution means executing a program in our local system. Remote execution means to running a program in another system.
- Server client relations: server is the machine which provides services and client is the machine which receive these services.
- To access any command remotely, server program provides the service to that system.

- Remote Execution Server has to be configured in the system that has the command.
- Remote executions can be done with protocols like ssh, telnet, rsh etc. But SSH is most appropriate way. So it is also known as SSH server.
- We need toconfigure the server before using. There are three steps to configure a server:
  a)#rpm -q openssh-server
  b)configure the file #vim /etc/ssh/sshd_config
  c)start service #systemctl start sshd
- When we perform operations locally, then it is known as localhost, whereas when we perform remote operations then it is known as remotehost.
- Client has 2 steps for connecting with ssh server. First download the client-side software by #yum openssh-clients then connect to the SSH server.
- To configure SSH, we need to install SSH server application in server machine. While in client machine, install client application. #yum install openssh-server and #yum install openssh-client.
- To remotely login using ssh, #ssh -l <user> <ip_address>.
- SSH is also called secure shell. By default it works on port 22. SSH is known as remote program execution server.
- For opening GUI of server we use option -X.
- Windows has no ssh client. Putty is one of the most commonly used in windows.
- To run multiple commands in ssh in a single line the command used is #ssh root@ip_address "date; cal; mkdir zz".
- We use ';' for running multiple programs at one and we can achieve this is by putting it in " " so as to make it work.
- But the problem is that if we just use ; the execution of the command after the semi colon so for those we use -X to meet the requirements.

RHEL8 SElinux Security

- SELinux stands for security enhanced linux.
- If a hacker runs a code of php on the process which is running in our webserver by this the hacker will get the shell and now he can change the password because now he has the power of root account and by this he can go into our configuration files and can do whatever he wants. This is where SELinux comes into play.
- So to avoid these types of attacks we started using the apache user for running the webserver instead of root account. Apache user has only limited powers.
- But still there are chances that the hacker can read or write something unusual in the file. So to avoid any kind of loopholes in our server, we use selinux.
- Every process in the RAM is associated with a user.
- When a process starts it is associated with the user. #ps -aux | less is the command used to check all the processes running on the RAM.
- To know the context/tag of process use -Z option with ps -aux command.
- To know the context/tag of file use -Z option with ls -l command.
- SElinx was created by NSA, used to enhance security, secure the webservers.
- #getenforce tells us whether the selinux has been enforced or not
- #setenforce 1 command can be use to enforce the security of linux and #setenforce 0 to disble selinux.
- There are basically two type of security model DAC (discretionary access control) and MAC (mandatory access control).

- In Linux, every file has its own set of permissions called Discretionary access control (DAC). We can change these permissions using the #chmod command.
- The concept of permission decision of files by files only is known as DAC system(Discretionary Access control). With DAC system there are some disadvantages that anyone can look all the system of the data.
- MAC(Mandatory Access Control) restricts a process to access by only some specific folders. It is supported by all the OS. It is used to restrict/confine the files that the process can access by exploiting bugs.
- SELinux is a product to implement DAC to MAC system. SELinux is used to solve the issues of DAC. It uses the concept of Mandatory Access Control (MAC).
- In DAC each file decides who/which process can access it that is read or edit it. In MAC context/tag is provided to each and every process and file. So if tag of process equal to tag of file then and then only process can access that file.
- The processes are restricted to access files based on some tags called SELinux Type.
- Tags are the things by which a file or folder checks whether the given user can use our services. On the basis of this file or folders deny or allow.
- To provide or remove permissions of a file we use chmod commands.
- To make a user owner of some file, we use a command #chown apache my.html. chown=CHange OWNer.
- #chmod -o -r <name_of_html_file> command helps us remove the other users to read the file. Now only the root user can read the file.
- To change context/tag use chcon keyword, chcon=CHange CONtext. Syntax: #chcon -t <new_context> <file_name>.
- To restore context use restorecon keyword. Syntax: restorecon <file_name>.
- To manage the process and their restrictions, we have some policies and rules designed. With SElinux every process and folder's tags will be matched or compared and if they match to the policy of SElinux then only they have permissions to access the folder of files else they can't move out of specified virtual spaces that is also known as virtual jail.
- #setenforce 1 to set the chroot jail for different process.
- In selinux each process and required files are associated with type. If both the type matches then the user can access the specified file. #ps -aux -Z | grep

<program_name> to get the SELinux context from which we can get the type that the SELinux has given to it.

Preparation

Red Hat encourages to consider taking Red Hat System Administration I (RH124) and Red Hat System Administration II (RH134) to help prepare.

While attending Red Hat classes can be an important part of your preparation, attending class does not guarantee success on the exam. Previous experience, practice, and native aptitude are also important determinants of success.

Many books and other resources on system administration for Red Hat products are also available. Nevertheless, additional reading might be helpful to deepen the understanding.

Exam format

The Red Hat Certified System Administrator (RHCSA) exam is a hands-on, practical exam that requires you to undertake real-world tasks. Internet access is not provided during the in-person exam, and bringing any hard copy or electronic documentation into the exam will not be permitted. This prohibition includes notes, books, or any other materials. For most exams, the documentation that ships with the product is available during the exam.

This exam can also be taken virtually as part of our remote testing format.

Scores and reporting

Official scores for exams come exclusively from Red Hat Certification Central. Red Hat does not authorize examiners or training partners to report results to candidates directly. Scores on the exam are usually reported within 3 U.S. business days.

Exam results are reported as section scores. Red Hat does not report performance on individual items, nor will it provide additional information upon request.

- The exam duration is 150 minutes
- The passing score is 210/300

Question Dumps

Below are few of the question dumps for EX200 for better understanding of the exam.

Question 1

Create one partitions having size 100MB and mount it on data.

Answer:
1. Use fdisk /dev/hda to create new partition.
2. Type n For New partitions.
3. It will ask for Logical or Primary Partitions. Press l for logical.
4. It will ask for the Starting Cylinder: Use the Default by pressing Enter Key.
5. Type the Size: +100M you can specify either Last cylinder of size here.
6. Press P to verify the partitions lists and remember the partitions name.
7. Press w to write on partitions table.
8. Either Reboot or use partprobe command.
9. Use mkfs -t ext3 /dev/hda?

OR -
mke2fs -j /dev/hda? To create ext3 filesystem.
vi /etc/fstab
Write:
/dev/hda? /data ext3 defaults 1 2
Verify by mounting on current Sessions also: mount /dev/hda? /data

Question 2

You are a new System Administrator and from now you are going to handle the system and your main task is Network monitoring, Backup and Restore. But you don't know the root password. Change the root password to redhat and login in default Runlevel.

Answer: When you Boot the System, it starts on default Runlevel specified in /etc/inittab:
Id:?:initdefault:
When System Successfully boot, it will ask for username and password. But you don't know the root's password. To change the root password you need to boot the system into single user mode. You can pass the kernel arguments from the boot loader.
1. Restart the System.
2. You will get the boot loader GRUB screen.
3. Press a and type 1 or s for single mode ro root=LABEL=/ rhgb queit s
4. System will boot on Single User mode.
5. Use passwd command to change.
6. Press ctrl+d

Question 3

You are a System administrator. Using Log files very easy to monitor the system. Now there are 50 servers running as Mail, Web, Proxy, DNS services etc. You want to centralize the logs from all servers into on LOG Server. How will you configure the LOG Server to accept logs from remote host?

Answer: By default, system accept the logs only generated from local host. To accept the Log from other host configure: vi /etc/sysconfig/syslog SYSLOGD_OPTIONS="-m 0 -r"

Where -
-m 0 disables 'MARK' messages.
-r enables logging from remote machines
-x disables DNS lookups on messages received with -r
service syslog restart

Question 4

Your System is configured in 192.168.0.0/24 Network and your nameserver is 192.168.0.254. Make successfully resolve to server1.example.com.

Answer : nameserver is specified in question,
1. Vi /etc/resolv.conf
nameserver 192.168.0.254
2. host server1.example.com

Question 5

One Package named zsh is dump on ftp://server1.example.com under /pub/updates directory and your FTP server is 192.168.0.254. Install the package zsh.

Answer:
rpm -ivh ftp://server1/example.com/pub/updates/zsh-*
or
Login to ftp server : ftp ftp://server1.example.com using anonymous user.
Change the directory: cd pub and cd updates
Download the package: mget zsh-*

Quit from the ftp prompt : bye -

Install the package -
rpm -ivh zsh-*
Verify either package is installed or not : rpm -q zsh

Question 6

Some users home directory is shared from your system. Using showmount -e localhost command, the shared directory is not shown. Make access the shared users home directory.

Answer:

Verify the File whether Shared or not ? : cat /etc/exports
Start the NFS service: service nfs start
Start the portmap service: service portmap start
Make automatically start the nfs service on next reboot: chkconfig nfs on
Make automatically start the portmap service on next reboot: chkconfig portmap on
Verify either sharing or not: showmount -e localhost
Check that default firewall is running on system?
If running flush the iptables using iptables -F and stop the iptables service.

Question 7

Add a new logical partition having size 100MB and create the data which will be the mount point for the new partition.

Answer:

1. Use fdisk /dev/hda-> To create new partition.
2. Type n ->For New partitions
3. It will ask for Logical or Primary Partitions. Press l for logical.
4. It will ask for the Starting Cylinder: Use the Default by pressing Enter

Keys -

5. Type the size: +100M you can specify either Last cylinder of size here.
6. Press P to verify the partitions lists and remember the partitions name.
7. Press w to write on partitions table.
8. Either Reboot or use partprobe command.
9. Use mkfs -t ext3 /dev/hda?

OR -

1. mke2fs -j /dev/hda? ->To create ext3 filesystem.
2. vi /etc/fstab
3. Write:
/dev/hda? /data ext3 defaults 0 0
4. Verify by mounting on current sessions also:
mount /dev/hda? /data

Question 8

You have a domain named www.rhce.com associated IP address is 192.100.0.2. Configure the Apache web server by implementing the SSL for encryption communication.

Answer:

vi /etc/httpd/conf.d/ssl.conf <VirtualHost 192.100.0.2> ServerName www.rhce.com
DocumentRoot /var/www/rhce DirectoryIndex index.html index.htm
ServerAdmin webmaster@rhce.com SSLEngine on SSLCertificateFile
/etc/httpd/conf/ssl.crt/server.crt SSLCertificateKeyFile
/etc/httpd/conf/ssl.key/server.key </
VirtualHost>
cd /etc/httpd/conf
3 make testcert
Create the directory and index page on specified path. (Index page can download from ftp://server1.example.com at exam time) service httpd start|restart chkconfig httpd on
Apache can provide encrypted communications using SSL (Secure Socket Layer). To make use of encrypted communication, a client must request to https protocol, which is uses port 443. For HTTPS protocol required the certificate file and key file.

Question 9

There is a server having 172.24.254.254 and 172.25.254.254. Your System lies on 172.24.0.0/16. Make successfully ping to 172.25.254.254 by Assigning following IP: 172.24.0.x where x is your station number.

Answer:

Use netconfig command -
Enter the IP Address as given station number by your examiner: example: 172.24.0.1

Enter Subnet Mask -
Enter Default Gateway and primary name server
press on ok
ifdown eth0
ifup eth0
verify using ifconfig
In the lab server is playing the role of router, IP forwarding is enabled. Just set the Correct IP and gateway, you can ping to 172.25.254.254.

Question 10

Successfully resolve to server1.example.com where your DNS server is 172.24.254.254.

Answer:

vi /etc/resolv.conf
nameserver 172.24.254.254
host server1.example.com

On every clients, DNS server is specified in /etc/resolv.conf. When you request by name it tries to resolve from DNS server.

Question 11

Your System is going use as a router for 172.24.0.0/16 and 172.25.0.0/16. Enable the IP Forwarding.
1. echo “1” >/proc/sys/net/ipv4/ip_forward
2. vi /etc/sysctl.conf net.ipv4.ip_forward=1

Answer : /proc is the virtual filesystem, containing the information about the running kernel.
To change the parameter of running kernel you should modify on /proc. From Next reboot the system, kernel will take the value from /etc/sysctl.conf.

Question 12

Who ever creates the files/directories on archive group owner should be automatically should be the same group owner of archive.

Answer:
chmod g+s /archive
Verify using: ls -ld /archive Permission should be like:
drwxrws — — 2 root sysuser 4096 Mar 16 18:08 /archive
If SGID bit is set on directory then who every user creates the files on directory group owner automatically the owner of parent directory.
To set the SGID bit: chmod g+s directory
To Remove the SGID bit: chmod g-s directory

Question 13

Make on /archive directory that only the user owner and group owner member can fully access.

Answer:
chmod 770 /archive
Verify using : ls -ld /archive Preview should be like:
drwxrwx — — 2 root sysuser 4096 Mar 16 18:08 /archive
To change the permission on directory we use the chmod command. According to the question that only the owner user (root) and group member (sysuser) can fully access the directory so: chmod 770 /archive

Question 14

SELinux must run in force mode.

Answer: /etc/sysconfig/selinux

```
SELINUX=enforcing
```

Question 15

Find the files owned by harry, and copy it to catalog: /opt/dir

Answer:

```
# cd /opt/
# mkdir dir
# find / -user harry -exec cp -rfp {} /opt/dir/ \;
```

Question 16

Configure a task: plan to run echo hello command at 14:23 every day.

Answer:

```
# which echo
# crontab -e
23 14 * * * /bin/echo hello
# crontab -l (Verify)
```

Question 17

Create a catalog under /home named admins. Its respective group is requested to be the admin group. The group users could read and write, while other users are not allowed to access it. The files created by users from the same group should also be the admin group.

Answer:

```
# cd /home/
# mkdir admins /
# chown .admin admins/
# chmod 770 admins/
# chmod g+s admins/
```

Question 18

Find the rows that contain abcde from file /etc/testfile, and write it to the file/tmp/testfile, and the sequence is requested as the same as /etc/testfile.

Answer:

```
# cat /etc/testfile | while read line;
do
```

```
echo $line | grep abcde | tee -a /tmp/testfile
done
```

OR -

```
grep `abcde’ /etc/testfile > /tmp/testfile
```

Question 19

Install a FTP server, and request to anonymous download from /var/ftp/pub catalog. (it needs you to configure yum direct to the already existing file server.)

Answer:

```
# cd /etc/yum.repos.d
# vim local.repo
[local]
name=local.repo
baseurl=file:///mnt
enabled=1
gpgcheck=0
# yum makecache
# yum install -y vsftpd
# service vsftpd restart
# chkconfig vsftpd on
# chkconfig — list vsftpd
# vim /etc/vsftpd/vsftpd.conf
anonymous_enable=YES
```

Question 20

Configure autofs to make sure after login successfully, it has the home directory autofs, which is shared as /rhome/ldapuser40 at the ip: 172.24.40.10. and it also requires that, other ldap users can use the home directory normally.

Answer:

```
# chkconfig autofs on
# cd /etc/
# vim /etc/auto.master
/rhome /etc/auto.ldap
# cp auto.misc auto.ldap
# vim auto.ladp
ldapuser40 -rw,soft,intr 172.24.40.10:/rhome/ldapuser40
* -rw,soft,intr 172.16.40.10:/rhome/&
# service autofs stop
# server autofs start
# showmount -e 172.24.40.10
# su — ladpuser40
```

Question 21:

Upgrading the kernel as 2.6.36.7.1, and configure the system to Start the default kernel, keep the old kernel available.

Answer:

```
# cat /etc/grub.conf
# cd /boot
# lftp it
# get dr/dom/kernel-xxxx.rpm
# rpm -ivh kernel-xxxx.rpm
# vim /etc/grub.conf
default=0
```

I hope this article helps in getting started with RHCSA.

> Linkedin: https://www.linkedin.com/gursimar-/

Redhat Linux Linux Rhcsa Exam Shell Script