OS Troublesooing Steps During the OS booting Time

==========

Task-1 How we can troubleshoot OS during boot time ?

Task-2 Booting Process OR Booting Sequence of Linux operating system

==========

Machine[RAM+CPU+DISK] <== OS <== Application/Software/Service

==========

Power on =====> step by step ==> boot ==> Sequence ==> Login Screen ==> # ==> activity start

==========

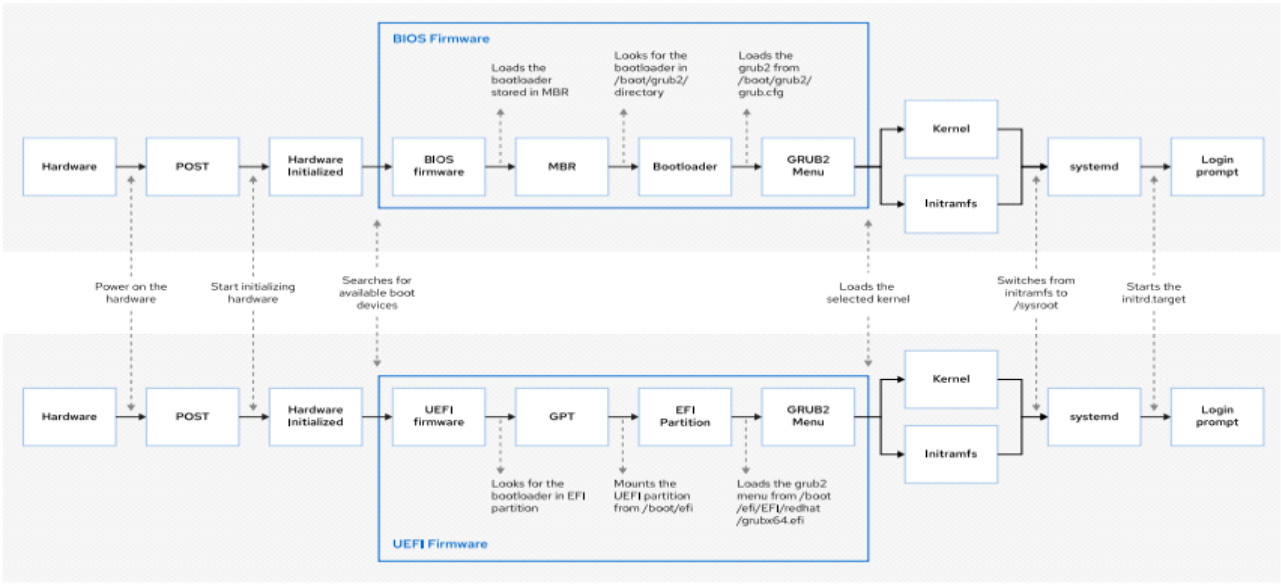


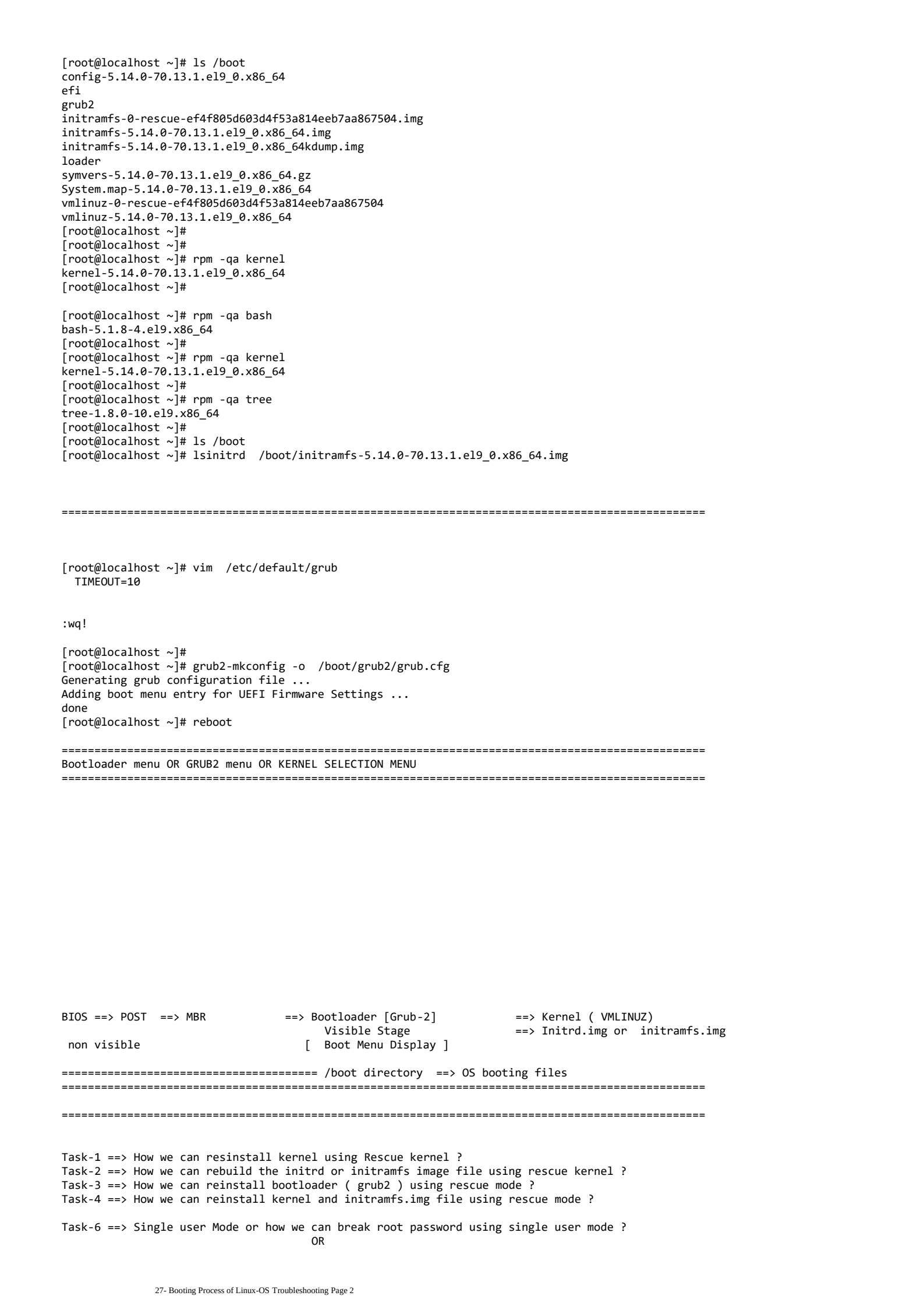
Figure 8.1: Boot process for BIOS-based and UEFI-based systems

==========

Notes:

```
[root@localhost ~]# cat /etc/redhat-release
Red Hat Enterprise Linux release 9.0 (Plow)
[root@localhost ~]#
[root@localhost ~]# cat /etc/system-release
Red Hat Enterprise Linux release 9.0 (Plow)
[root@localhost ~]#
[root@localhost ~]# ls  /boot
config-5.14.0-70.13.1.el9_0.x86_64
efi
grub2
initramfs-0-rescue-ef4f805d603d4f53a814eeb7aa867504.img
initramfs-5.14.0-70.13.1.el9_0.x86_64.img
initramfs-5.14.0-70.13.1.el9_0.x86_64kdump.img
loader
symvers-5.14.0-70.13.1.el9_0.x86_64.gz
System.map-5.14.0-70.13.1.el9_0.x86_64
vmlinuz-0-rescue-ef4f805d603d4f53a814eeb7aa867504
vmlinuz-5.14.0-70.13.1.el9_0.x86_64
[root@localhost ~]#
[root@localhost ~]# uname
Linux
[root@localhost ~]# uname -r
5.14.0-70.13.1.el9_0.x86_64
[root@localhost ~]#
[root@localhost ~]# uname -m
x86_64
[root@localhost ~]# uname -a
Linux localhost.localdomain 5.14.0-70.13.1.el9_0.x86_64 #1 SMP PREEMPT Thu Apr 14 12:42:38 EDT 2022 x86_64 x86_64 x86_64 GNU/Linux
[root@localhost ~]#
```

```
[root@localhost ~]# ls /boot
config-5.14.0-70.13.1.el9_0.x86_64
efi
grub2
initramfs-0-rescue-ef4f805d603d4f53a814eeb7aa867504.img
initramfs-5.14.0-70.13.1.el9_0.x86_64.img
initramfs-5.14.0-70.13.1.el9_0.x86_64kdump.img
loader
symvers-5.14.0-70.13.1.el9_0.x86_64.gz
System.map-5.14.0-70.13.1.el9_0.x86_64
vmlinuz-0-rescue-ef4f805d603d4f53a814eeb7aa867504
vmlinuz-5.14.0-70.13.1.el9_0.x86_64
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# rpm -qa kernel
kernel-5.14.0-70.13.1.el9_0.x86_64
[root@localhost ~]#

[root@localhost ~]# rpm -qa bash
bash-5.1.8-4.el9.x86_64
[root@localhost ~]#
[root@localhost ~]# rpm -qa kernel
kernel-5.14.0-70.13.1.el9_0.x86_64
[root@localhost ~]#
[root@localhost ~]# rpm -qa tree
tree-1.8.0-10.el9.x86_64
[root@localhost ~]#
[root@localhost ~]# ls /boot
[root@localhost ~]# lsinitrd  /boot/initramfs-5.14.0-70.13.1.el9_0.x86_64.img



==========================================================================================================



[root@localhost ~]# vim  /etc/default/grub
  TIMEOUT=10


:wq!

[root@localhost ~]#
[root@localhost ~]# grub2-mkconfig -o  /boot/grub2/grub.cfg
Generating grub configuration file ...
Adding boot menu entry for UEFI Firmware Settings ...
done
[root@localhost ~]# reboot

==========================================================================================================
Bootloader menu OR GRUB2 menu OR KERNEL SELECTION MENU
==========================================================================================================
```




```
==========================================================================================================

==========================================================================================================


Task-1 ==> How we can resinstall kernel using Rescue kernel ?
Task-2 ==> How we can rebuild the initrd or initramfs image file using rescue kernel ?
Task-3 ==> How we can reinstall bootloader ( grub2 ) using rescue mode ?
Task-4 ==> How we can reinstall kernel and initramfs.img file using rescue mode ?

Task-6 ==> Single user Mode or how we can break root password using single user mode ?
                                    OR
```

How we can secure the grub menu ?
OR
How we can apply username and password on Grub menu or single user mode ?

Task-7 ==> How we can troubleshoot Ctrl + D error OR Filesystem issue OR

Maintenance mode or during the booting time ?

/etc/fstab ==> entry ==> always must be correct + every device must be accesible

==========

Task-1 Solution

How we can resinstall kernel using Rescue kernel ?

```
[root@localhost ~]#
[root@localhost ~]# uname -r
5.14.0-70.13.1.el9_0.x86_64
[root@localhost ~]#
[root@localhost ~]# ls  /boot
config-5.14.0-70.13.1.el9_0.x86_64
efi
grub2
initramfs-0-rescue-23e23d115f09406fa9361b2e83ab81e1.img
initramfs-5.14.0-70.13.1.el9_0.x86_64.img
initramfs-5.14.0-70.13.1.el9_0.x86_64kdump.img
loader
symvers-5.14.0-70.13.1.el9_0.x86_64.gz
System.map-5.14.0-70.13.1.el9_0.x86_64
vmlinuz-0-rescue-23e23d115f09406fa9361b2e83ab81e1
vmlinuz-5.14.0-70.13.1.el9_0.x86_64
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# cd /boot
[root@localhost boot]# rm -rf  vmlinuz-5.14.0-70.13.1.el9_0.x86_64
[root@localhost boot]#
[root@localhost boot]# cd

[root@localhost ~]# reboot
```

after reboot we will get kernel related error

now we can boot this machine using rescue kernel to install actual regular kernel [main kernel]

connect ISO image with your machine OR else we can download or copy the kernel rpm

```
    7  lsblk
    8  mount /dev/sr0  /mnt
    9  cd /mnt
   10  ls
   11  cd BaseOS
   12  ls
   13  cd Packages
   14  ls
   15  rpm -ivh kernel-5.14.0-70.13.1.el9_0.x86_64.rpm
   16  rpm -ivh kernel-5.14.0-70.13.1.el9_0.x86_64.rpm  --force
       rpm -ivh kernel-core-5.14.0-70.13.1.el9_0.x86_64.rpm  --force
   17  cd
   18  grub2-mkconfig -o  /boot/grub2/grub.cfg
   19  history
   20  ls  /boot
[root@localhost ~]# reboot
```

Now this machine will boot properly with regular kernel OR main kernel

==========

Task-2 Solution

```
    7  ls /boot
    8  lsinitrd /boot/initramfs-5.14.0-70.13.1.el9_0.x86_64.img
    9  rm -rf   /boot/initramfs-5.14.0-70.13.1.el9_0.x86_64.img
    10 reboot
```

After reboot we will get error related to kernel panic OR initrmfs image file not found

again select resuce kernel to load the machine

```
[root@localhost ~]# ls /boot
config-5.14.0-70.13.1.el9_0.x86_64                        loader
efi                                                       symvers-5.14.0-70.13.1.el9_0.x86_64.gz
grub2                                                     System.map-5.14.0-70.13.1.el9_0.x86_64
initramfs-0-rescue-23e23d115f09406fa9361b2e83ab81e1.img  vmlinuz-0-rescue-23e23d115f09406fa9361b2e83ab81e1
initramfs-5.14.0-70.13.1.el9_0.x86_64kdump.img           vmlinuz-5.14.0-70.13.1.el9_0.x86_64
[root@localhost ~]#
[root@localhost ~]#
```

Method-1 Support till rhel-8   ==> using mkinitrd

Method-2 by rhel-9 ==> using dracut

==========

on rhel-8

```
[root@localhost ~]# which mkinitrd
/usr/bin/mkinitrd
[root@localhost ~]#
[root@localhost ~]# which dracut
/usr/bin/dracut
[root@localhost ~]#
[root@localhost ~]# rpm -qf  /usr/bin/mkinitrd
dracut-049-10.git20190115.el8.x86_64
[root@localhost ~]#
[root@localhost ~]# rpm -qf  /usr/bin/dracut
dracut-049-10.git20190115.el8.x86_64
[root@localhost ~]#
[root@localhost ~]#

# cd /boot
# mkinitrd initramfs-$(uname -r).img   $(uname -r)
# ls
[root@localhost ~]# grub2-mkconfig  -o  /boot/grub2/grub.cfg
Generating grub configuration file ...
done
[root@localhost ~]# reboot
```

==========

on rhel-9

```
[root@localhost ~]# rpm -qa dracut
dracut-055-45.git20220404.el9_0.x86_64
[root@localhost ~]#
[root@localhost ~]# which dracut
/usr/bin/dracut
[root@localhost ~]#
cd /boot
[root@localhost boot]#
[root@localhost boot]# dracut --help
[root@localhost boot]# dracut -f  initramfs-5.14.0-70.13.1.el9_0.x86_64.img  5.14.0-70.13.1.el9_0.x86_64
                                  OR
[root@localhost boot]# dracut -f   initramfs-$(uname -r).img   $(uname -r)
[root@localhost boot]#
[root@localhost boot]#
[root@localhost boot]# ls
config-5.14.0-70.13.1.el9_0.x86_64                        loader
efi                                                       symvers-5.14.0-70.13.1.el9_0.x86_64.gz
grub2                                                     System.map-5.14.0-70.13.1.el9_0.x86_64
initramfs-0-rescue-23e23d115f09406fa9361b2e83ab81e1.img  vmlinuz-0-rescue-23e23d115f09406fa9361b2e83ab81e1
initramfs-5.14.0-70.13.1.el9_0.x86_64.img                 vmlinuz-5.14.0-70.13.1.el9_0.x86_64
initramfs-5.14.0-70.13.1.el9_0.x86_64kdump.img
[root@localhost boot]#
[root@localhost boot]#
[root@localhost boot]# cd
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# grub2-mkconfig  -o  /boot/grub2/grub.cfg
Generating grub configuration file ...
Adding boot menu entry for UEFI Firmware Settings ...
done
[root@localhost ~]# reboot
```

==========

Task-3 Solution

```
   20  cd /boot
   21  ls
   22  rm -rf grub2
   23  cd
   24  history
   25  reboot
```

We need now bootable media [DVD/pendrive] if want to go in rescue mode

but in case of virtual machine we can boot this machine directly using ISO image

power on firmware ==> then set the booting priority to cdrom and f10 save it

Method-1 Troubleshooting ==> rescue ==> then press enter   ===> in RHEL-7-8-9
Method-2 press ESC key ==> boot: linux rescue [Press Enter] ===> support in old version

==========

```
# chroot   /mnt/sysroot
# lsblk
# cd /boot
# grub2-install  /dev/sda
# grub2-mkconfig  -o  /boot/grub2/grub.cfg
# exit
# exit
```

==========

Task-4

```
[root@localhost ~]# cd /boot
[root@localhost boot]#
[root@localhost boot]# ls
config-5.14.0-70.13.1.el9_0.x86_64
efi
grub2
initramfs-0-rescue-23e23d115f09406fa9361b2e83ab81e1.img
initramfs-5.14.0-70.13.1.el9_0.x86_64.img
initramfs-5.14.0-70.13.1.el9_0.x86_64kdump.img
loader
symvers-5.14.0-70.13.1.el9_0.x86_64.gz
System.map-5.14.0-70.13.1.el9_0.x86_64
vmlinuz-0-rescue-23e23d115f09406fa9361b2e83ab81e1
vmlinuz-5.14.0-70.13.1.el9_0.x86_64
[root@localhost boot]#
[root@localhost boot]# rm -rf vmlinuz-5.14.0-70.13.1.el9_0.x86_64
[root@localhost boot]#
[root@localhost boot]# rm -rf  initramfs-5.14.0-70.13.1.el9_0.x86_64.img
[root@localhost boot]#
[root@localhost boot]# rm -rf  grub2
[root@localhost boot]# reboot
```

==========

==========

Now we can troubleshoot these errors using only rescue mode

1- first we will install grub
2- now we will install kernel
3- we can rebuild the initramfs image file

```
# chroot /mnt/sysroot
```

Note: We will execute the same commands to troubleshoot these erros.

==========

already we have covered in RUNLEVEL and Targets chapter ?

How we can go in single user mode ?

How we can apply the password on single user mode OR protect the grub menu OR grub2 Menu OR booting screen ?

==========
Note: we can create new LVM and we can currupt this LVM

```
      to understand about to manintance mode OR Ctrl +D error ?
================================================================================================

[root@localhost ~]# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda      8:0    0   30G  0 disk
├─sda1   8:1    0    1G  0 part /boot
├─sda2   8:2    0 14.6G  0 part /
└─sda3   8:3    0    2G  0 part [SWAP]
sdb      8:16   0    5G  0 disk
sr0     11:0    1 1024M  0 rom
[root@localhost ~]#

[root@localhost ~]# fdisk  /dev/sdb

Welcome to fdisk (util-linux 2.37.4).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0xe85a89de.

Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-10485759, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-10485759, default 10485759): +2G

Created a new partition 1 of type 'Linux' and of size 2 GiB.

Command (m for help): t
Selected partition 1
Hex code or alias (type L to list all): 8e
Changed type of partition 'Linux' to 'Linux LVM'.

Command (m for help): p
Disk /dev/sdb: 5 GiB, 5368709120 bytes, 10485760 sectors
Disk model: VMware Virtual S
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xe85a89de

Device     Boot Start     End Sectors Size Id Type
/dev/sdb1        2048 4196351 4194304   2G 8e Linux LVM

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.

[root@localhost ~]#
[root@localhost ~]# partprobe  /dev/sdb
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# pvcreate  /dev/sdb1
  Physical volume "/dev/sdb1" successfully created.
  Creating devices file /etc/lvm/devices/system.devices
[root@localhost ~]#
[root@localhost ~]# vgcreate  myvg1  /dev/sdb1
  Volume group "myvg1" successfully created
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# lvcreate   -L  +1G  -n  mylv1  /dev/myvg1
  Logical volume "mylv1" created.
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# mkfs.xfs  /dev/myvg1/mylv1
meta-data=/dev/myvg1/mylv1       isize=512    agcount=4, agsize=65536 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=1, rmapbt=0
         =                       reflink=1    bigtime=1 inobtcount=1
data     =                       bsize=4096   blocks=262144, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0, ftype=1
log      =internal log           bsize=4096   blocks=2560, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
[root@localhost ~]#
[root@localhost ~]# mkdir /storage
```

```
[root@localhost ~]#
[root@localhost ~]# mount /dev/myvg1/mylv1  /storage
[root@localhost ~]#


[root@localhost ~]# vim  /etc/fstab



/dev/myvg1/mylv1                          /storage                xfs     defaults        0 0


:wq


[root@localhost ~]# systemctl daemon-reload

[root@localhost ~]#
[root@localhost ~]# lsblk
NAME            MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda               8:0    0   30G  0 disk
├─sda1            8:1    0    1G  0 part /boot
├─sda2            8:2    0 14.6G  0 part /
└─sda3            8:3    0    2G  0 part [SWAP]
sdb               8:16   0    5G  0 disk
└─sdb1            8:17   0    2G  0 part
  └─myvg1-mylv1 253:0    0    1G  0 lvm  /storage
sr0              11:0    1 1024M  0 rom
[root@localhost ~]# reboot

There is no ERROR

================================================================================================


[root@localhost ~]# umount  /storage
[root@localhost ~]#
[root@localhost ~]# lvremove   /dev/myvg1/mylv1
Do you really want to remove active logical volume myvg1/mylv1? [y/n]: y
  Logical volume "mylv1" successfully removed.
[root@localhost ~]#
[root@localhost ~]# reboot
================================================================================================

==========================================You will get This Error now
================================================================================================

 Control + D error OR maintenance Mode

give root password : redhat

# journalctl -xb | grep failed     ==> to read the error


# mount -o remount rw  /
# vim /etc/fstab

comment LVM OR respective storage line and save it

# reboot -f


================================================================================================




How we can recover GRUB if we are working with UEFI mode  ?


[root@localhost ~]# rm -rf    /boot/grub2/
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# mount /dev/sr0 /mnt
mount: /mnt: WARNING: source write-protected, mounted read-only.
[root@localhost ~]#
[root@localhost ~]# vim /etc/yum.repos.d/abc.repo

[path-1]
name=abc
baseurl=file:///mnt/AppStream
enabled=1
gpgcheck=0
```

```
[path-2]
name=xyz
baseurl=file:///mnt/BaseOS
enabled=1
gpgcheck=0

:wq!
[root@localhost ~]# yum reinstall grub2-efi-x64 shim-x64   -y

[root@localhost ~]# grub2-mkconfig  -o   /boot/efi/EFI/redhat/grub.cfg

[root@localhost ~]# reboot
```

==========