Boot Procedure:
- Press the power button on your system,
- System goes through POST (Reads BIOS settings and check basic hardware).
- System determines the primary boot device
- System reads MBR of device, it read information about the boot loader
- The BIOS search and load the Boot Loader program into the memory
- Depending on installed with either start Grub or LILO: Boot Loader determines default kernel image
- Kernel decompresses and extract itself
- Kernel mounts the root file system as specified in the “root=” in grub.conf
- Kernel starts the init process (pid =1)
- Init reads its configuration file (/etc/inittab) to determine the default runlevel.
- mounting all filesystems from /etc/fstab
- Init starts processes located in the default runlevel's rc directory.
- Init starts processes in rc.local.
- Init start TTY sessions and the user is presented with a login prompt.

(GRUB is accountable for loading OS kernel from disk.)
Modifying the of GRUB parameters in a persistent way:

```
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# vi /etc/default/grub _
```

```
GRUB_TIMEOUT=5
GRUB_DISTRIBUTOR="$(sed 's, release .*$,,g' /etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_TERMINAL_OUTPUT="console"
GRUB_CMDLINE_LINUX="crashkernel=auto resume=/dev/mapper/ol-swap rd.lvm.lv=ol/root rd.lvm.lv=ol/swap
rhgb quiet"
GRUB_DISABLE_RECOVERY="true"
GRUB_ENABLE_BLSCFG=true
~
~
~
```

I have edited the ‘GRUB_CMDLINE_LINUX’ line as per my requirement. Save and exit the (grub) file.
Run ‘grub2-mkconfig -o /boot/grub2/grub.cfg’ for updating configuration.

```
grub.cfg  grubenv
[root@localhost ~]# grub2-mkconfig -o /boot/grub2/grub.cfg
Generating grub configuration file ...
done
```

Then I rebooted my system.

To Troubleshoot a server, there are 2 modes they are ‘Emergency.target(which has countable services) and Rescue.target (which has minimal services).

Setting up the default target:
Get the default target.

```
[root@localhost ~]# systemctl get-default
graphical.target
[root@localhost ~]# _
```

Set up a default target.

```
[root@localhost ~]# systemctl set-default graphical.target
[root@localhost ~]#
```

Booting into a specific Target:
Power on the system, after a few seconds type ‘Esc’ (which will show the Grub boot menu).

```
load_video
set gfx_payload=keep
insmod gzio
linux ($root)/vmlinuz-5.4.17-2136.300.7.el8uek.x86_64 root=/dev/mapper/ol-root\
 ro crashkernel=auto resume=/dev/mapper/ol-swap rd.lvm.lv=ol/root rd.lvm.lv=ol\
/swap rhgb quiet
initrd  ($root)/initramfs-5.4.17-2136.300.7.el8uek.x86_64.img $tuned_initrd




     Press Ctrl-x to start, Ctrl-c for a command prompt or Escape to
     discard edits and return to the menu. Pressing Tab lists
     possible completions.
```

Then, Look for the line where it’s loading the kernel and add ‘systemd.unit=rescue.target’

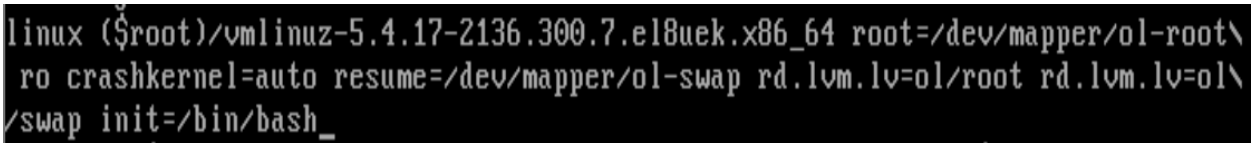
```
linux ($root)/vmlinuz-5.4.17-2136.300.7.el8uek.x86_64 root=/dev/mapper/ol-root\
 ro crashkernel=auto resume=/dev/mapper/ol-swap rd.lvm.lv=ol/root rd.lvm.lv=ol\
/swap systemd.unit=rescue.target
```

```
[  OK  ] Reached target Rescue Mode.
         Starting Update UTMP about System Runlevel Changes...
[  OK  ] Started Update UTMP about System Runlevel Changes.
You are in rescue mode. After logging in, type "journalctl -xb" to view
system logs, "systemctl reboot" to reboot, "systemctl default" or "exit"
to boot into default mode.
Give root password for maintenance
(or press Control-D to continue):
```

Result, I'm in the rescue mode and start troubleshooting.

Forget the Root Password: (Resetting the root password)
Power on the system, after a few seconds type ‘Esc’ (which will show the Grub boot menu).

Then, Look for the line where it’s loading the kernel and at the end of the line add ‘ init=/bin/bash’.



And press Ctrl-x to start.
It will prompt a bash shell, then type ‘mount’ and then look for /dev/mapper/ol-root check for permissions like ‘ro’.

```
none on /sys/kernel/tracing type tracefs (rw,relatime)
configfs on /sys/kernel/config type configfs (rw,relatime)
/dev/mapper/ol-root on / type xfs (ro,relatime,attr2,inode64,logbufs=8,logbsize=32k,noquota)
bash-4.4#
```

It was having read only permissions.

```
/dev/mapper/ol-root on / type xfs (ro,relatime,attr2,inode64,logbufs=8,logbsize=32k,noquota)
bash-4.4# mount -o remount,rw /
[  180.892990] xfs filesystem being remounted at / supports timestamps until 2038 (0x7fffffff)
bash-4.4# _
```

Then change the password for ‘root’ and then I have typed the password.

```
[  180.892990] xfs filesystem being remounted at / supports timestamps until 2038 (0x7fffffff)
bash-4.4# passwd root
Changing password for user root.
New password:
BAD PASSWORD: The password fails the dictionary check - it is too simplistic/systematic
Retype new password:
passwd: all authentication tokens updated successfully.
```

Then let the SELinux know about the changes.

Final step would be to re-initialize the systemd process.

```
passwd: all authentication tokens updated successfully.
bash-4.4# touch /.autorelabel
bash-4.4# exec /usr/lib/systemd/systemd
```