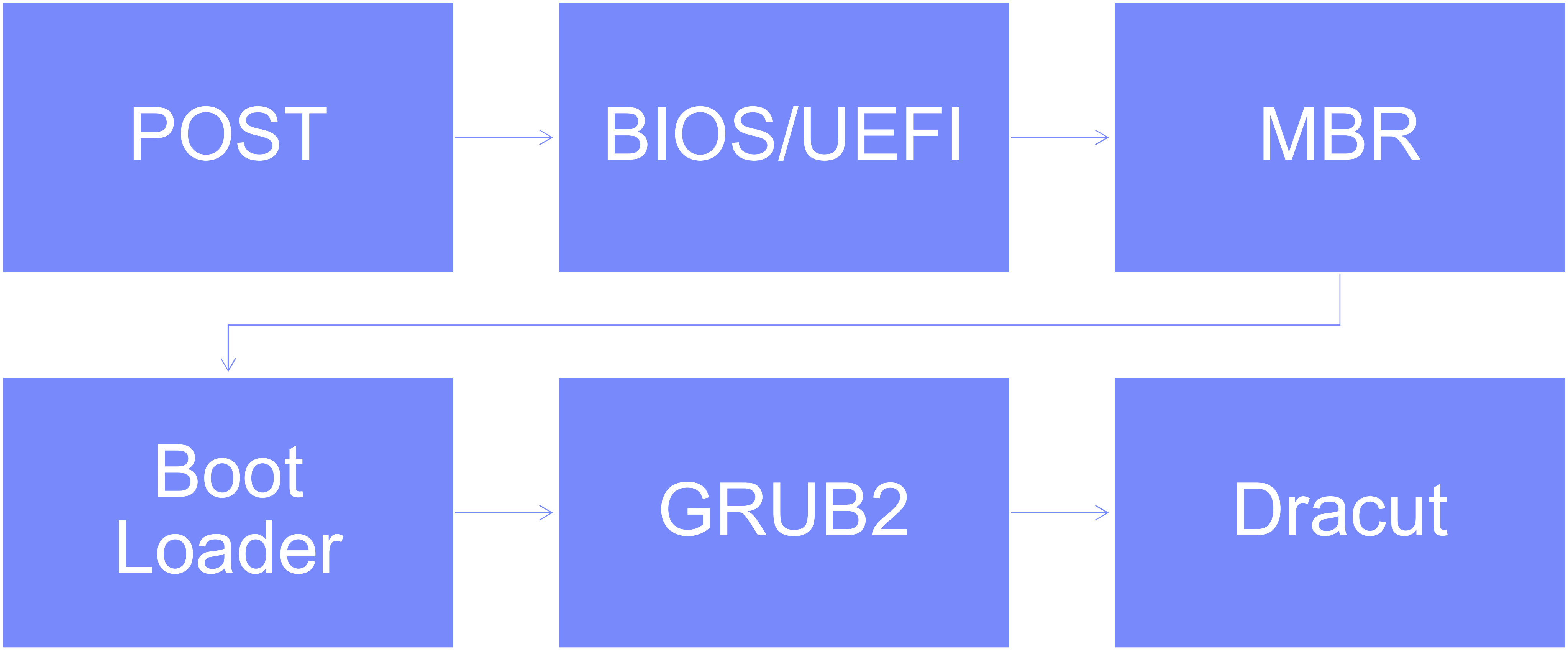
Controlling Boot Process

Unit objectives

After completing this unit, you should be able to:

- Observe old and new method in managing services
- Understanding Systemd Units and Unit Types
- Select Boot Target
- Reset the Root Password
- Troubleshoot failed services

Boot Sequence 1/2



POST

BIOS/UEFI

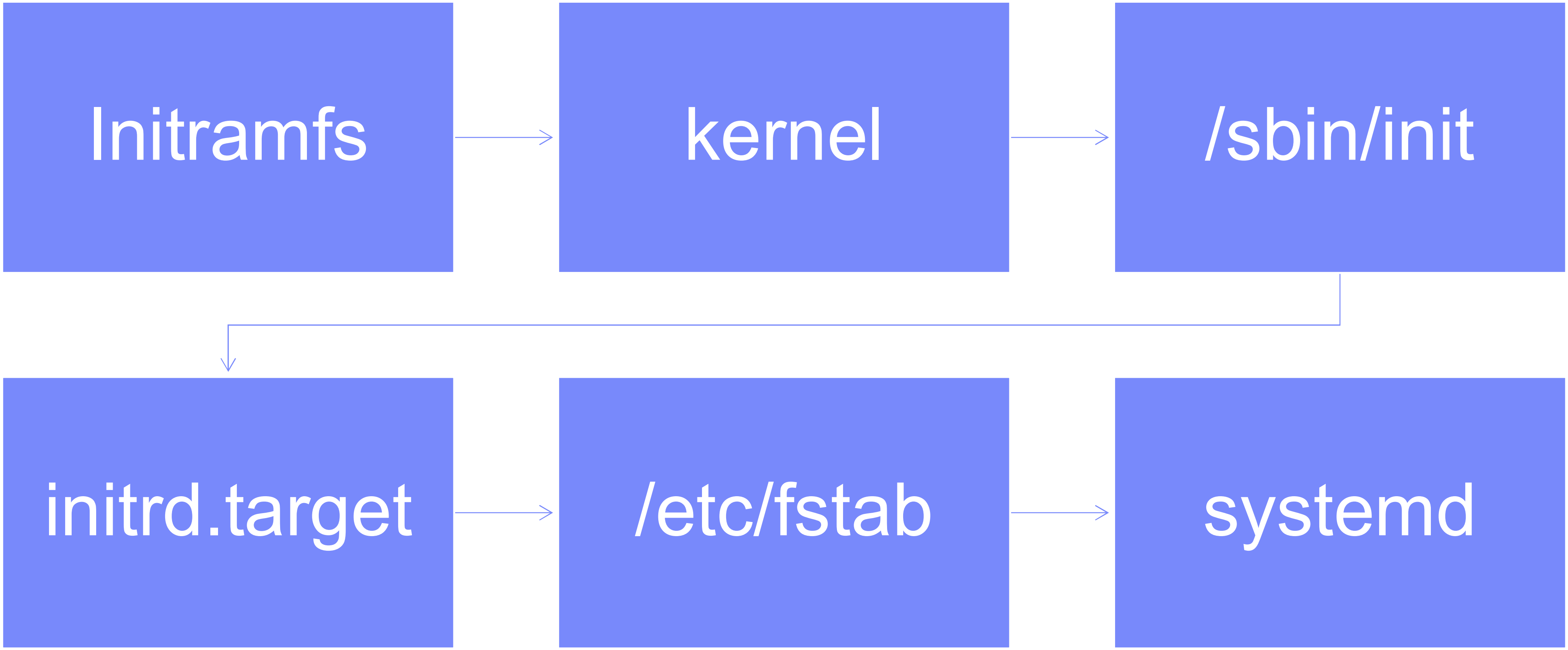
MBR

Boot Loader

GRUB2

Dracut

Boot Sequence 2/2



Initramfs

kernel

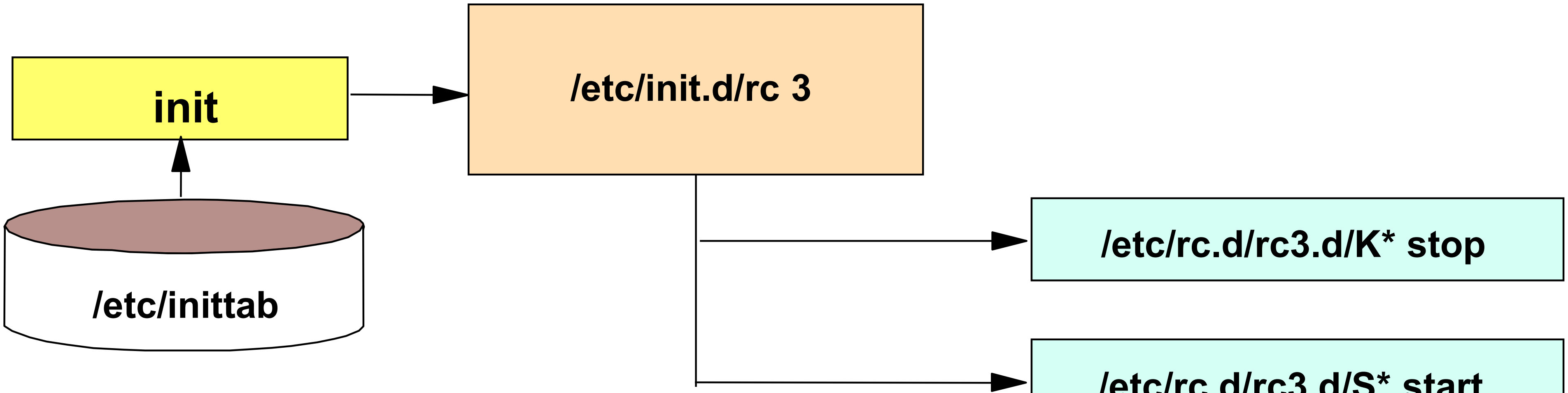
/sbin/init

initrd.target

/etc/fstab

systemd

Managing services (System V init style)

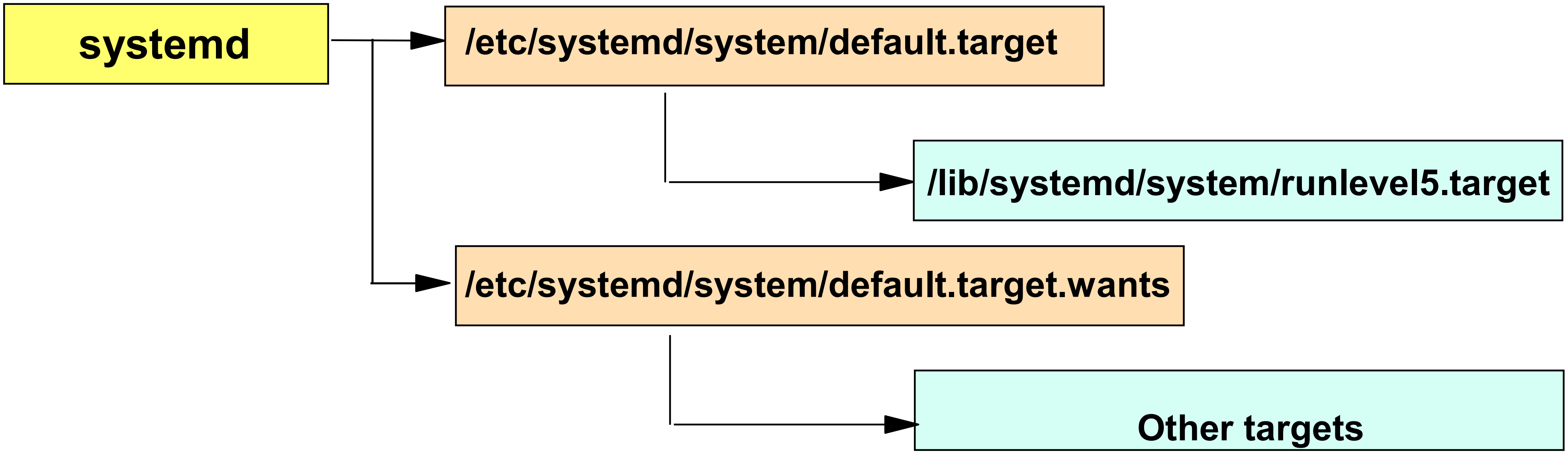


(Symlinks to the actual start/stop script)

```
# ls -l /etc/rc.d/rc3.d
lrwxrwxrwx  1 root root 24 Mar 15 10:47 K02NetworkManager ->
../init.d/NetworkManager
lrwxrwxrwx  1 root root 14 Mar 15 11:45 K05innd -> ../init.d/innd
lrwxrwxrwx  1 root root 19 Mar 15 10:45 K05saslauthd -> ../init.d/saslauthd
. . .
lrwxrwxrwx  1 root root 15 Mar 15 10:48 K15httpd -> ../init.d/httpd
lrwxrwxrwx  1 root root 15 Mar 15 11:45 K16rarpd -> ../init.d/rarpd
. . .
```

Managing services (systemd)

- Upcoming replacement for `init` in RHEL and SLES
  - Already present in Fedora 15 and OpenSUSE 11.4
- Works by defining "targets" which have other targets as dependencies
- Far less reliant on shell scripts
- Can start tasks in parallel and "on demand"
- Downwards compatible with System V init style scripts


systemd
/etc/systemd/system/default.target
/lib/systemd/system/runlevel5.target
/etc/systemd/system/default.target.wants
Other targets

Understanding Systemd Targets

- Is similar runlevel in System V
- Boot into a specific state
- Start group of services at each state
- Find default target in /etc/systemd/system
- Boot with all necessary processes / services
- Managed via
  - systemctl command
  - Cockpit web-based
  - System Management Tool

Understanding Systemd Units and Unit Type

- Query each unit with

```
# systemctl -at <unit_type>
```

Unit Type	Description
Service	Process / Daemon
Target	Group of systemd services / units
Automount	Auto-mount file system
Device	Device file recognied by kernel
Mount	File system mount point
Path	File or directory in a file system
Scope	Externally created process
Slice	Group of hierarchilly organized units that manage system processes
Socket	Inter-process communication
Swap	Swap device or swap file
Timer	Systemd timer

systemd Target Descriptions

Target name	Description
poweroff.target	Shutdown all services. Dont's set as default target ;)
rescue.target	Similar to run-level 1 in System V Similar to single-user mode or run-level 1. Only root can login for troubleshooting, diagnosing or any maintenance work Networking, GUI, multi-user not started
multi-user.target	Similar to run-level 3 in System VAlmost all services except services that Start X Window system (GUI) Text based console login capability
graphical.target	Similar to run-level 5 in System V Start all services with X Window system Default target Common state for desktop or workstation use
reboot.target	Reboot the system. Dont's set as default target
network.target	use as part of dependency. Cannot not be isolated
local-fs.target	mount local file system. use as part of dependency. Cannot not be isolated
Others...	there are about 70 other targets. mostly used as part of milestone or dependency check.

Set default target

- In a **systemd** environment, `service` is a wrapper for `systemctl`

```
List all targets
# systemctl -at target

List all services
# systemctl -at service

Set target
# systemctl set-default multi-user.target

Verify default target
# systemctl get-default
# ls -l /etc/systemd/system/default.target
```

List and check on services

List all services

```
# systemctl -a
# systemctl list-all-units
```

Show all services that depends on target

```
# systemctl list-dependency rescue.target
# systemctl list-dependency multi-user.target
# systemctl list-dependency graphical.target
```

- Colored dots to left
  - green : means ok, running
  - white : stopped, or never enabled to start
  - red : enabled to start but failed. Due to fatal error , diagnose with log

Starting and stopping services

- Default options: `start`, `stop`, `status`, `restart`
- Other options might also be available
- In a **systemd** environment, `service` is a wrapper for `systemctl`

```
Stop service
# systemctl stop httpd

Start service
# systemctl start httpd

Restart service
# systemctl restart httpd

Verify state of service
# systemctl status [-l] httpd
```

Enabling and Disabling services

- Start / Stop : immediate effect
- Enable / Disable : at boot time

```
Enable service
# systemctl enable httpd

Disable service
# systemctl disable httpd

Masking service – ignore start/stop/enable/disable
# systemctl mask httpd

Verify state of service
# systemctl status [-l] httpd
```

Troubleshooting

- Services failed to start could due few reasons:
  - Configuration not valid, not complete
  - Dependent service not started
  - Check service's state → Restart service → Read logs → Reinstall software package → Enter rescue mode

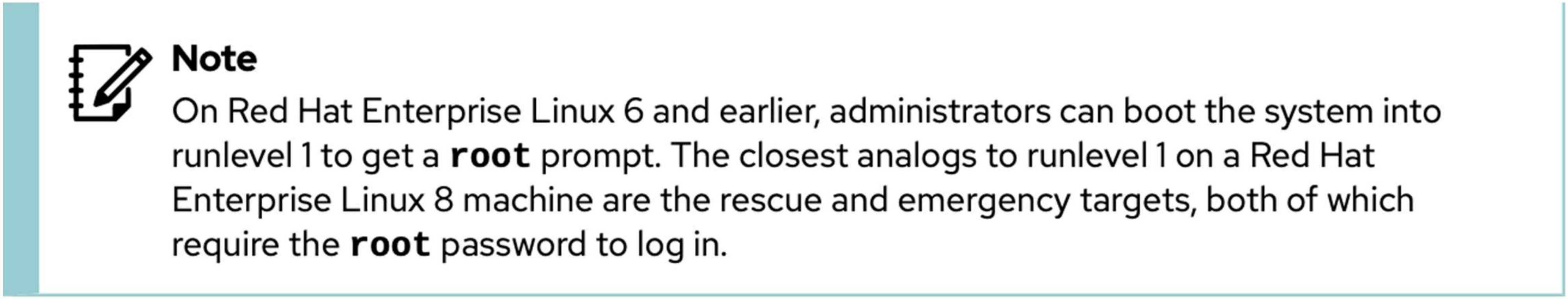
```
Verify state of service
# systemctl status -l httpd

Restart service
# systemctl restart httpd

Read logs -look for errorneous message
# grep httpd /var/logs/messages

Reinstall software package
# dnf -y remove httpd; dnf -y install httpd
```

Rebooting and Shutting Down

Method	Descriptiong
systemctl poweroff	stops all running services. unmount all file systems (remount ro if cant unmount) Powers down system
systemctl reboot	poweroff > reboot
systemctl halt	stops all running services. unmount all file systems (remount ro if cant unmount) (never power down system)
poweroff	symbolic to system poweroff
reboot	symbolic to system reboot

Resetting Root Password

1. Reboot system
2. During grub / boot loadin, press any key or enter to interrupt
3. Move cursor to non-rescue mode kernel entry
4. Press **e** to edit selected entry
5. Move to line that start with **linux** (kernel command)
6. Append rd.break
7. Press **ctrl+x** to boot



Note

On Red Hat Enterprise Linux 6 and earlier, administrators can boot the system into runlevel 1 to get a **root** prompt. The closest analogs to runlevel 1 on a Red Hat Enterprise Linux 8 machine are the rescue and emergency targets, both of which require the **root** password to log in.

Resetting Root Password

- Prompted with root shell with /**sysroot** mounted as **RO**

8. Remount /sysroot with read-write, then switch to the FS

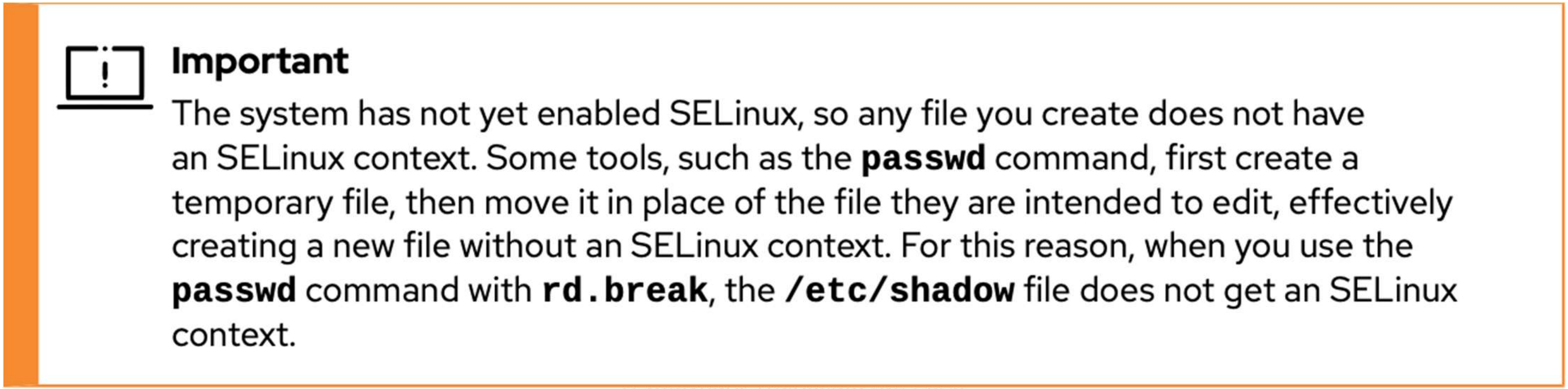
```
# mount -o rw,remount /sysroot
# chroot /sysroot
```

9. Now assign any intended password

```
# passwd root
```

10. Relabel all files including /etc/shadow with SELinux policy

```
$ touch /.autorelabel
```



Important

The system has not yet enabled SELinux, so any file you create does not have an SELinux context. Some tools, such as the **`passwd`** command, first create a temporary file, then move it in place of the file they are intended to edit, effectively creating a new file without an SELinux context. For this reason, when you use the **`passwd`** command with **`rd.break`**, the **`/etc/shadow`** file does not get an SELinux context.

Inspect Logs

- Look into previously failed boot process
- Need to set up systemd-journald service

```
# vi /etc/systemd/journald.conf
…
Storage=persistent
…
# systemctl restart systemd-journald.service
```

Inspect Logs

- Look into first boot logs

```
# journalctl –b 1
```

- Look into second, third boot logs

```
# journalctl –b 2
# journalctl –b 3
```

- Look into current boot session logs

```
# journalctl –b
```

- Look for error logs only in first boot

```
# journaltctl –b 1 –p err
```

Repair Systemd Boot Issues

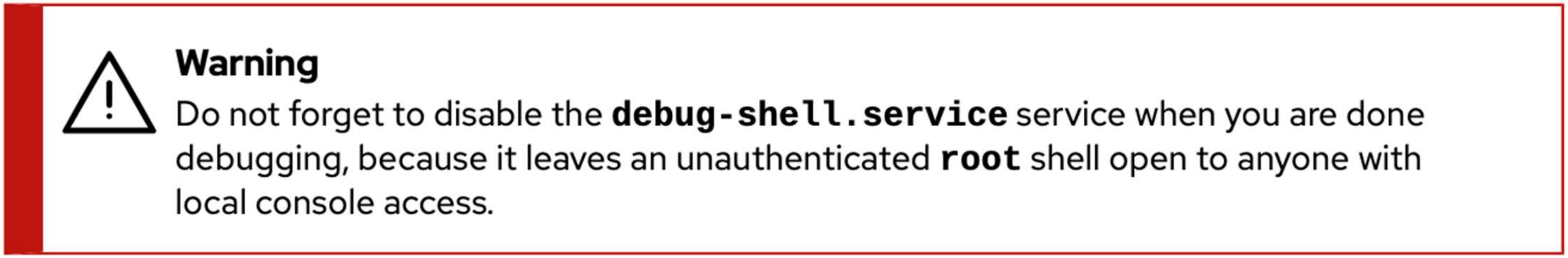
- Enable Early Debug Shell
- Use Emergency and Rescue Targets
- Identify Stuck Jobs

Enabling EarlyDebug Shell

- in v8
- Troubleshoot service startup issues at boot time
- Run following command after boot

```
# systemctl enable debug-shell.service
# reboot
```

- This shell automatically logged in as root, so that admin can debug system while OS is still booting



Warning

Do not forget to disable the **`debug-shell.service`** service when you are done debugging, because it leaves an unauthenticated **`root`** shell open to anyone with local console access.

Using Emergency and Rescue Targets

- Emergency : Only root file system mounted with read-only; immediately stops all others processes
- Rescue : All important file systems is mounted. Services in sysinit.target are started

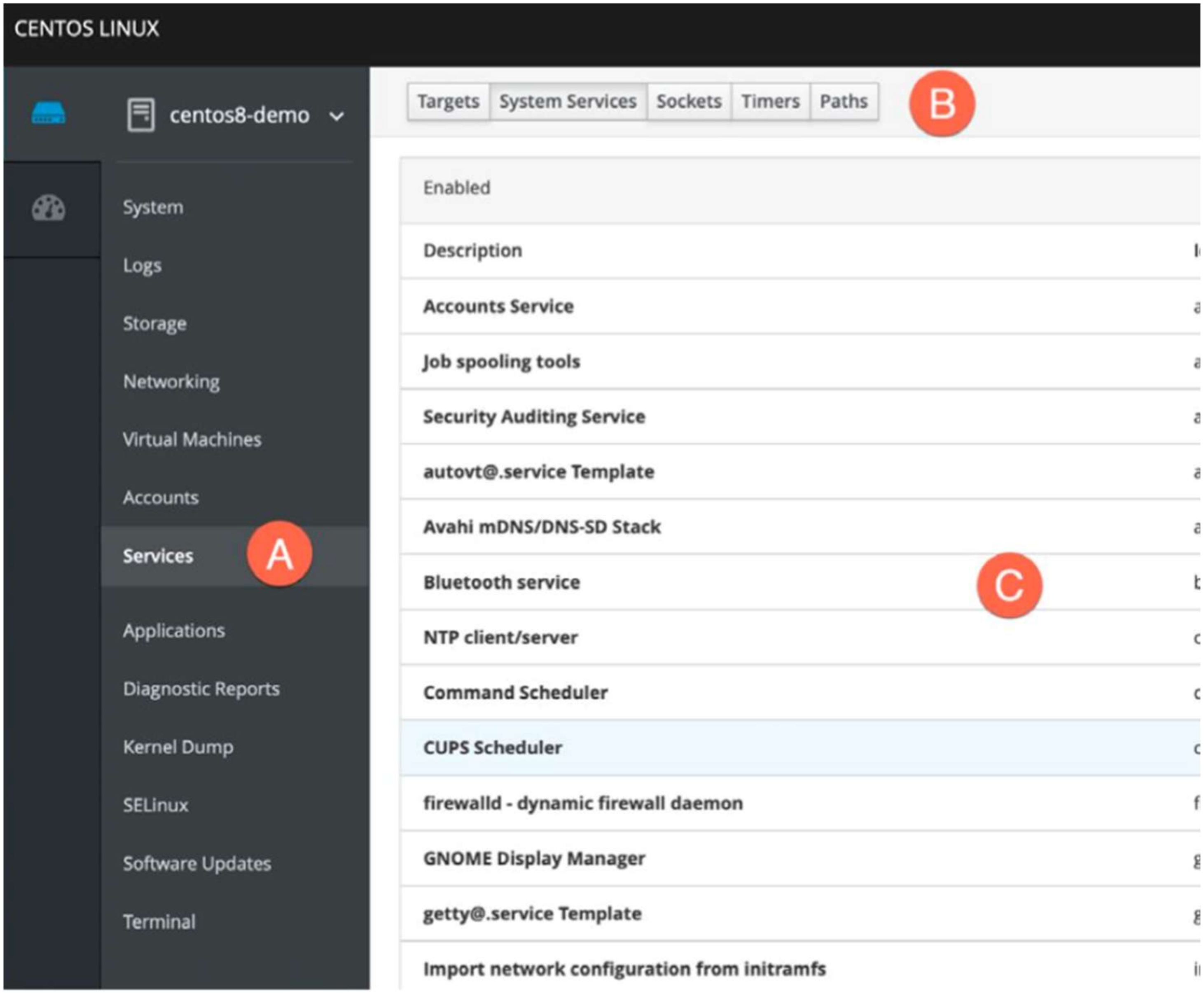
```
Enter into rescue mode immediately
# systemctl isolate rescue.target
# systemctl rescue
# systemctl emergency

Quickly back to default target
# systemctl default
or
# <press control+d>

Enter into rescue mode on next and every reboot
# systemctl set-default rescue
# reboot
# who -r
```

Identify Stuck Jobs

- During boot, systemd starts multiple jobs.
- Jobs cannot complete (blocked by other)
- To inspect unfinished job initiated during boot time

```
# systemctl list-jobs
```

Working with systemd Units in Cockpit



Enabled

Description
Accounts Service
Job spooling tools
Security Auditing Service
autovt@.service Template
Avahi mDNS/DNS-SD Stack
Bluetooth service
NTP client/server
Command Scheduler
CUPS Scheduler
firewalld - dynamic firewall daemon
GNOME Display Manager
getty@.service Template
Import network configuration from initramfs

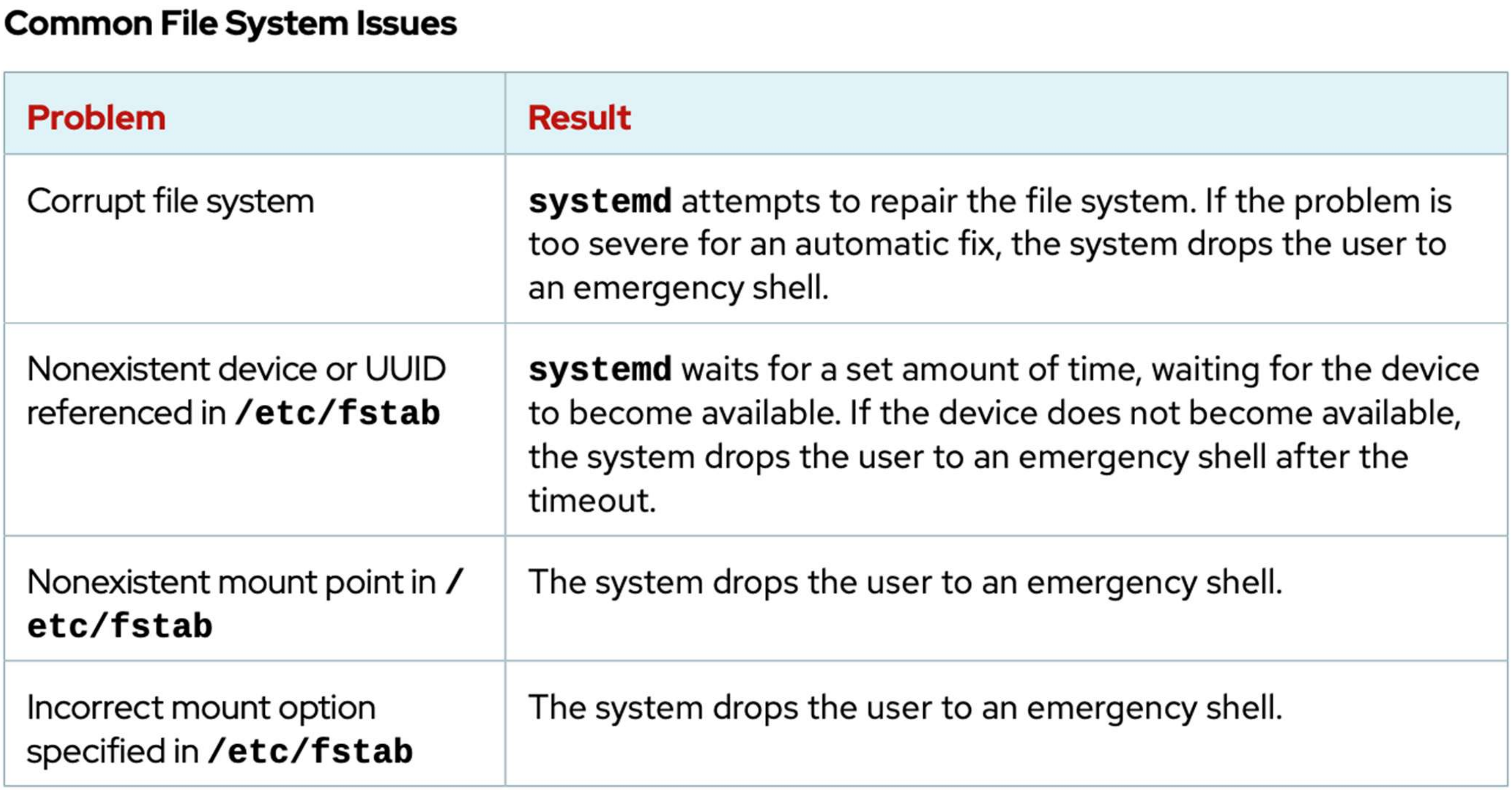
A – Navigation panel

B – Unit types

C – Services / units with status and other information

Repairing File System Issues at Boot

- Diagnosing and Fixing File System Issues
  - errors in /etc/fstab will prevent system from booting
  - Systemd drops to emergency repair shell (require root password)



Common File System Issues

Problem	Result
Corrupt file system	**`systemd`** attempts to repair the file system. If the problem is too severe for an automatic fix, the system drops the user to an emergency shell.
Nonexistent device or UUID referenced in **`/etc/fstab`**	**`systemd`** waits for a set amount of time, waiting for the device to become available. If the device does not become available, the system drops the user to an emergency shell after the timeout.
Nonexistent mount point in **`/etc/fstab`**	The system drops the user to an emergency shell.
Incorrect mount option specified in **`/etc/fstab`**	The system drops the user to an emergency shell.

- After edit /etc/fstab, run systemctl daemon-reload

Use *mount* to

- Determine mounted file system

```
# mount
```

- Remount read-only root file system

```
# mount –o rw,remount /
```

- Mount all file system listed in /etc/fstab

```
# mount --all
```

Fix mount error

- If mount point does not exists:

```
# mount --all
Error: /mnt/mountfolder: mount point does not exist.
```

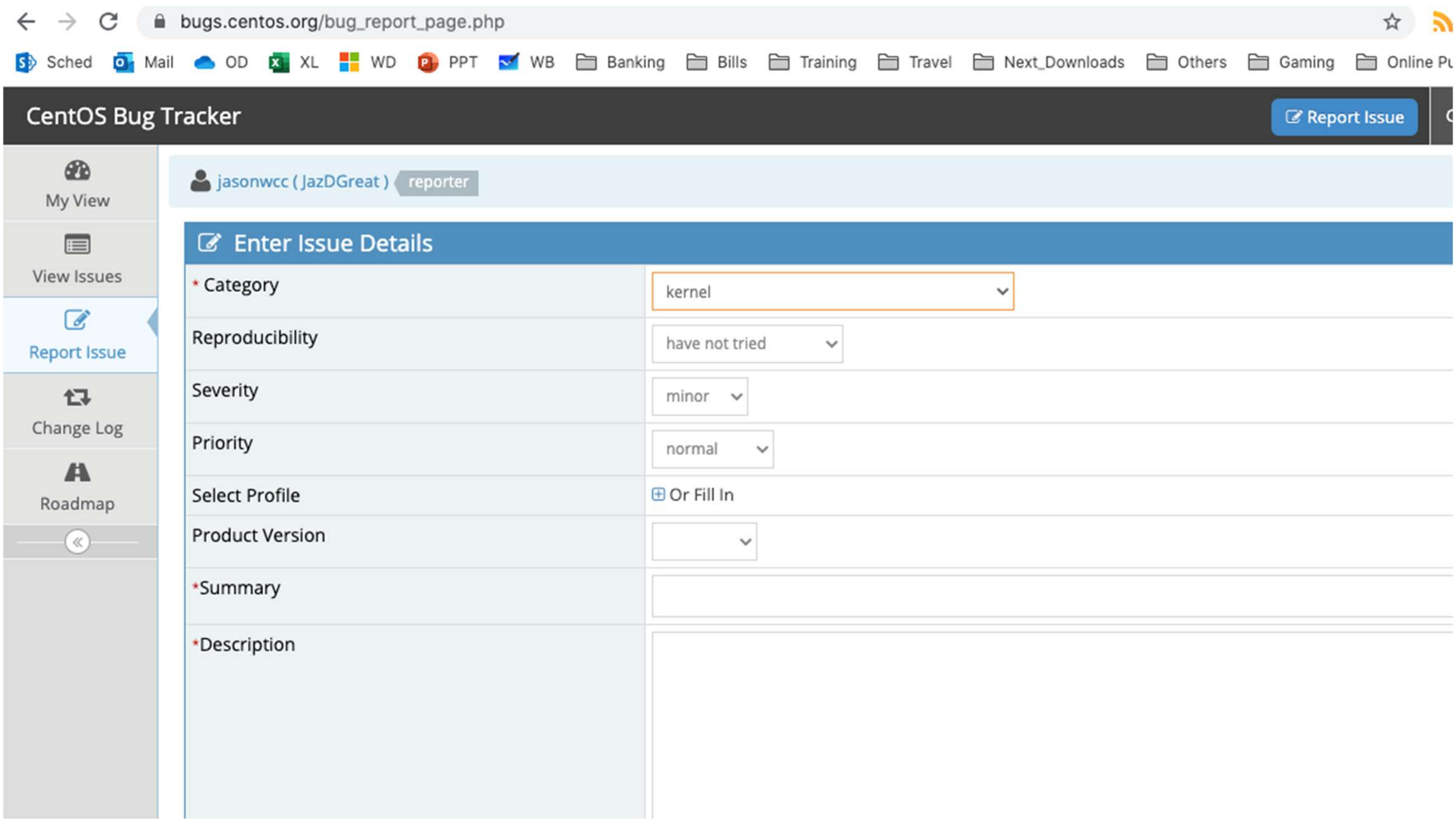
- Then create the mount point, reload systemd and then mount all file system again

```
# mkdir /mnt/mountfolder
# systemctl daemon-reload
# mount --all
```

Verify mount unit created by systemd

```
# systemctl –at mount
```

File bug report



Enter Issue Details

* Category	kernel
Reproducibility	have not tried
Severity	minor
Priority	normal
Select Profile	⊞ Or Fill In
Product Version	
*Summary	
*Description	

The journalctl command

- May used to query contents of systemd
  - All interesting things happening in systemd
  - Structured logs with well-known fields
  - man systemd.journal-fields for documentation

```
Show full content with oldest entries
# journalctl
Show all system related and jump to end of journal
# journalctl --system -e
Additional warning / errors with possible solutions
# journalctl -xe
Show message after specific date & time
# journalctl -S "2020-09-29 13:00:00"
Show additional msg and perform tail like
# journalctl -xef
Show all messages rated as error or worse from last boot
# journalctl -b -1 -p err
Identifying Stuck jobs
# systemctl list-jobs
```

Checkpoint

1. You execute # systemctl enable sshd. But you still can't connect to the service. Why?

   a) Its not a known service
   b) The service is not started yet
   c) Firewall blocks sshd
   d) Its probably you try execute the command with non-root account

Checkpoint

1. You execute # systemctl enable sshd. But you still can't connect to the service. Why?

   a) Its not a known service
   b) The service is not started yet
   c) Firewall blocks sshd
   d) Its probably you try execute the command with non-root account

Checkpoint

2. How do you stop the firewalld service?

a) # service stop firewalld

b) # service firewalld stop

c) # systemctl stop firewalld

d) # systemctl firewalld stop

Checkpoint

2. How do you stop the firewalld service?

   a) # service stop firewalld
   b) # service firewalld stop
   c) # systemctl stop firewalld
   d) # systemctl firewalld stop

Checkpoint

3. How do you verify the firewalld service is stopped?

   a) # check status firewalld
   b) # status check firewalld
   c) # systemctl status firewalld
   d) # systemctl verify firewalld

Checkpoint

3. How do you verify the firewalld service is stopped?

   a) # check status firewalld

   b) # status check firewalld

   c) # systemctl status firewalld

   d) # systemctl verify firewalld

Guided Exercise

Topic	Page number on student-guide.pdf	Time (min)
Select the Boot Target	234	10
Reset the Root Password	241	10
Repair File System Issues at Boot	246	10
Lab: Control the Boot Process	249	25

Checkpoint

1. Which are the methods to get logs about systemd? [Choose all that applies]
   a) more /var/log/messages
   b) more /var/adm/messages
   c) journalctl
   d) systemctl

2. You execute # systemctl enable sshd. But you still can't connect to the service. Why?
   a) Its not a known service
   b) The service is not started yet
   c) Firewall blocks sshd
   d) Its probably you try execute the command with non-root account

3. What are the possible solution to fix any failed state service? [Choose all that applies]
   a) Restart the service
   b) Restart the operating system
   c) Use journalctl cmd
   d) Call RedHat support team
   e) Enter rescue target to perform further diagnosis and troubleshooting

4. True or False: multi-user.target is run-level 3 in older System V

Checkpoint

1. Which are the methods to get logs about systemd? [Choose all that applies]
   a) more /var/log/messages
   b) more /var/adm/messages
   c) journalctl
   d) systemctl

2. You execute # systemctl enable sshd. But you still can't connect to the service. Why?
   a) Its not a known service
   b) The service is not started yet
   c) Firewall blocks sshd
   d) Its probably you try execute the command with non-root account

3. What are the possible solution to fix any failed state service? [Choose all that applies]
   a) Restart the service
   b) Restart the operating system
   c) Use journalctl cmd
   d) Call RedHat support team
   e) Enter rescue target to perform further diagnosis and troubleshooting

4. True or False: multi-user.target is run-level 3 in older System V

Unit summary

Having completed this unit, you should be able to:

- Understanding Systemd Units and Unit Types
- Select Boot Target
- Reset the Root Password
- Troubleshoot failed services