

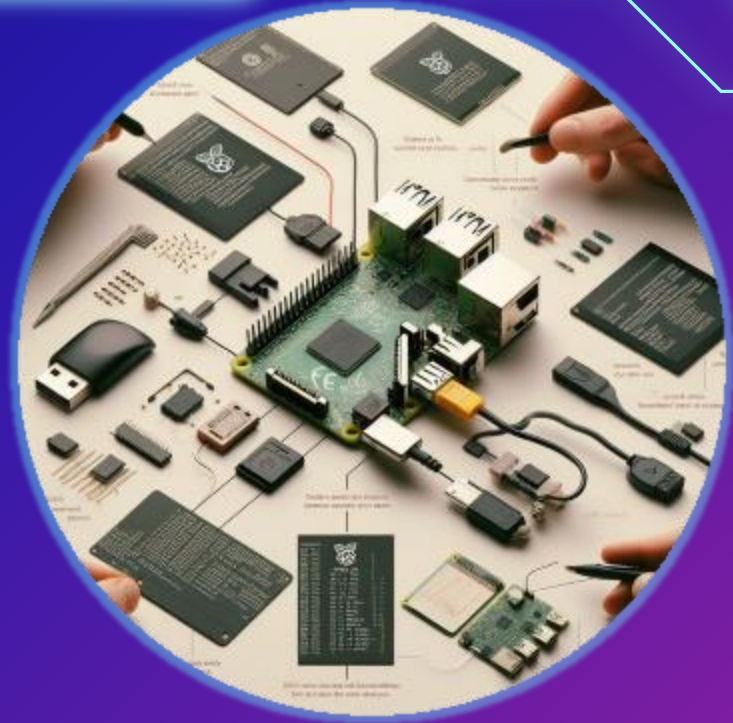
# The 5th Meetup of the Embedded Linux Users Group (E-LUG)

## Topic

Accessing Rootfs without  
CABLE CONNECTION

## Provider

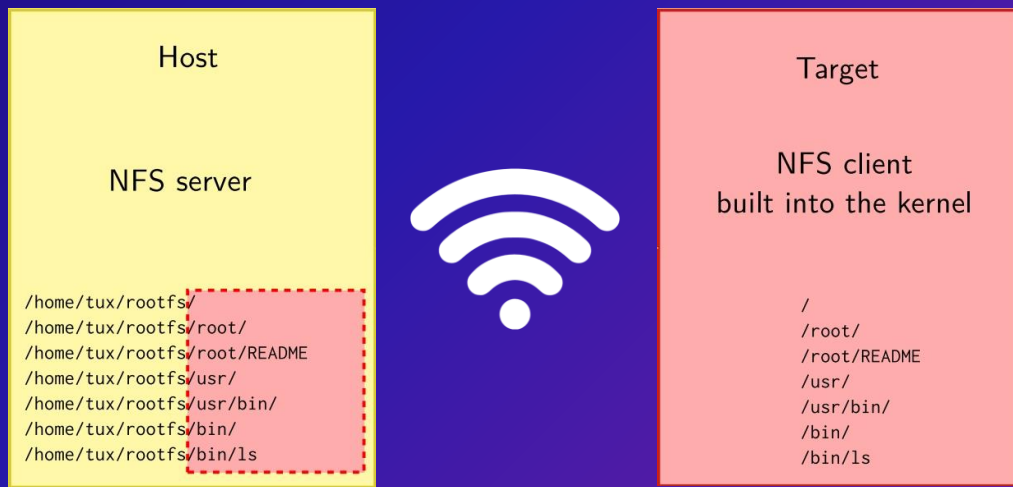
Mersad Karimi



# Accessing NFS Rootfs Over The Air

General description:

1. We are going to mount rootfs with NFS option.
2. With one difference that it is mounted without any cable connection, using wifi.
3. In this case, we prepare connection with initramfs.





# TABLE OF CONTENTS



01

## Rootfs Concept

A simple description about rootfs and init process.

02

## Applications / Drivers

What programs we need for connecting to server.

03

## Initramfs Concept

What is initramfs and some usages of that.

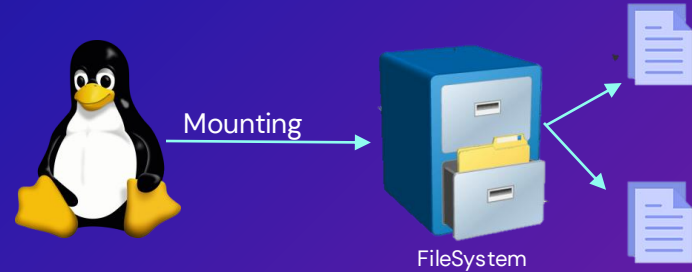
04

## USE THEM TOGETHER

The workshop part of the meeting

# ROOTFS

- A **file system** defines how files are **named**, **stored**, and **retrieved** from a storage device.
- A particular filesystem is mounted at the root of the hierarchy, identified by /
- As mount and umount are programs, they are files inside a filesystem. They are not accessible before mounting at least one filesystem



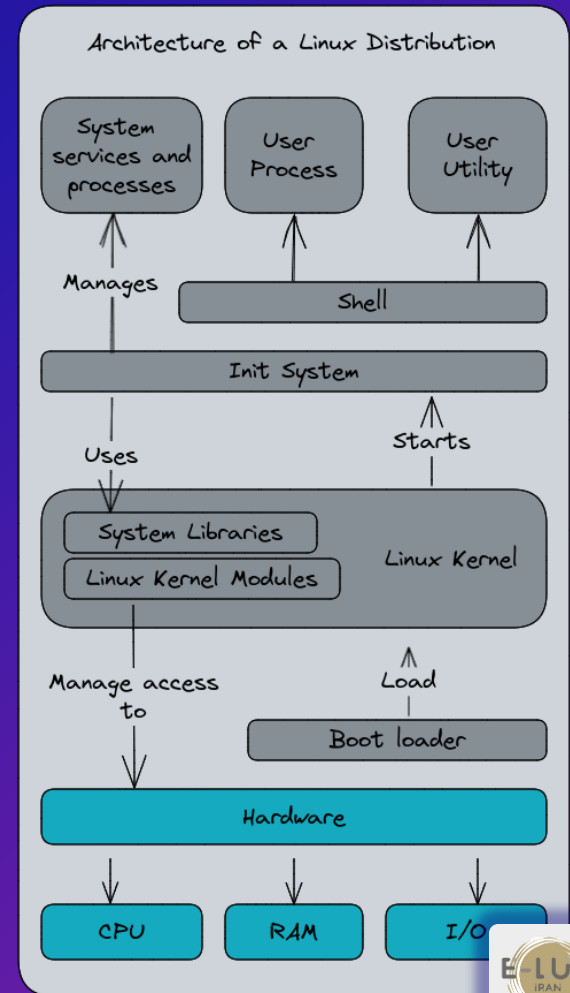
```
sda      8:0    0 119.2G  0 disk
├─sda1   8:1    0   512M  0 part /boot/efi
└─sda2   8:2    0 118.8G  0 part /
sdb      8:16   1    59G   0 disk
└─sdb1   8:17   1    59G   0 part
```

```
→ ~ where mount
   /usr/bin/mount
   /bin/mount
```

# Init Process

In Unix-based computer operating systems, init (short for initialization) is **the first process started during booting of the operating system**. Init is a daemon process that continues running until the system is shut down.

The kernel tries to run the command specified by the `init=` command line parameter if available. • Otherwise, it tries to run `/sbin/init`, `/etc/init`, `/bin/init` and `/bin/sh`. • In the case of an `initramfs`, it will only look for `/init`. Another path can be supplied by the `rdinit=` kernel argument.



# Initramfs

## Bootloader

Loads the initramfs archive to RAM (if separate)  
Loads DTB + kernel to RAM, starts the kernel

## Kernel

Initializes hardware devices and kernel subsystems  
Extracts the initramfs archive to the file cache  
Starts the /init executable if found  
(otherwise falls back to mounting the device specified by root=)

## /init

Starts early user space commands  
(show splashscreen, start time critical application...)  
Loads drivers needed to access the final root filesystem  
Mounts the root filesystem and switches to it (switch\_root)

Intermediate root filesystem (initramfs)

## /sbin/init

Regular system startup

Root filesystem



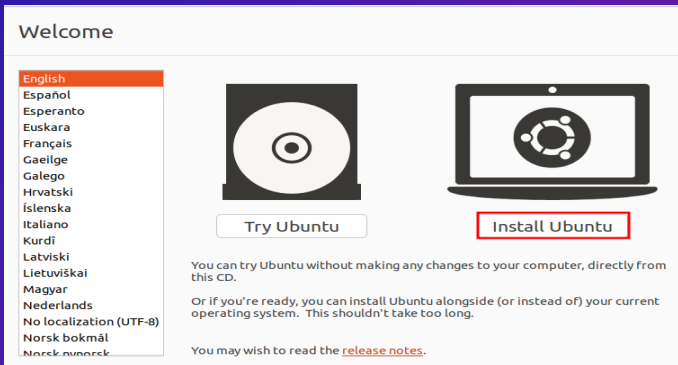
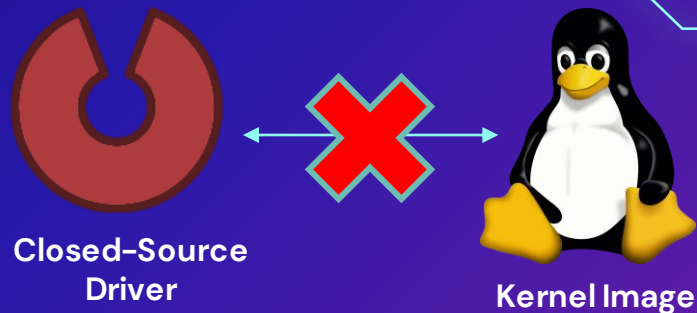
# Initramfs Usages

## Loading Essential Drivers:

initramfs includes kernel modules (drivers) necessary for accessing storage devices. If your root filesystem is on an external device and Driver of that is closed source (e.g., USB, RAID, or network), initramfs loads the relevant drivers so the kernel can access it.

## Live Environments and Rescue Systems:

- In live Linux systems (e.g., for troubleshooting or recovery), initramfs provides a minimal operating environment with essential tools and drivers, allowing access to different partitions and devices.



# Using Initramfs



Either from a compressed CPIO archive integrated into the kernel image Or from such an archive loaded by the bootloader into memory

## 1 – Integrated into the kernel image

```
[*] Initial RAM filesystem and RAM disk (initramfs/initrd) support
()  Initramfs source file(s)
[*]  Support initial ramdisk/ramfs compressed using gzip
[*]  Support initial ramdisk/ramfs compressed using bzip2
[*]  Support initial ramdisk/ramfs compressed using LZMA
[*]  Support initial ramdisk/ramfs compressed using XZ
[*]  Support initial ramdisk/ramfs compressed using LZ0
[*]  Support initial ramdisk/ramfs compressed using LZ4
[*]  Support initial ramdisk/ramfs compressed using ZSTD
```

## 2 – Loading by Bootloader

```
## Loading init Ramdisk from Legacy Image at 40000000 ...
Image Name:   Ramdisk Image
Image Type:   ARM Linux RAMDisk Image (gzip compressed)
Data Size:    64539648 Bytes = 61.5 MiB
Load Address: 00000000
Entry Point:  00000000
```



# Applications and Drivers

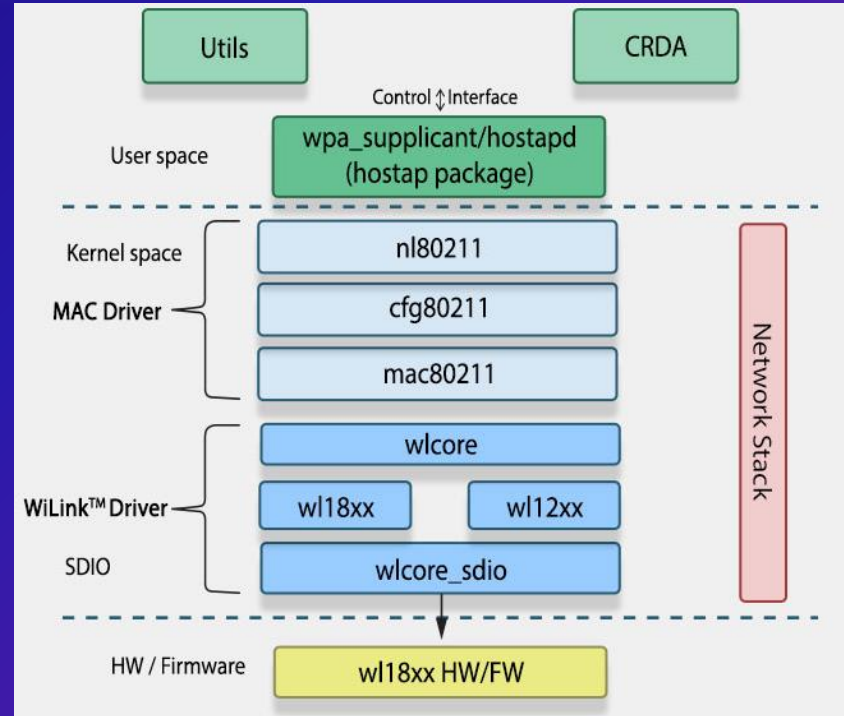


## Essential Drivers

- Device Driver
- Mac Driver
- Device Firmware

## Essential Programs

- iw
- ifconfig
- mount
- switch\_root
- bash / sh



# Work Shop

# USE THEM TOGETHER



## HOST x86-64

- Creating initramfs
- Setting NFS Server Rootfs
- Connecting to Access Point + IP

## Target ARM64

- Setting initramfs -> Boot Time
- Connecting to Access Point
- Mounting Rootfs
- Switching Root

# Thank YOU



Follow us in social media :

