



## lk\_filesystem

LK3: Wow file, such system.

Louis Solofrizzo [louis@ne02ptzero.me](mailto:louis@ne02ptzero.me)  
42 Staff [pedago@42.fr](mailto:pedago@42.fr)

*Summary:*

# Contents

<b>I</b>	<b>Forewords</b>	<b>2</b>
<b>II</b>	<b>Introduction</b>	<b>4</b>
	II.0.1 Filesystem . . . . .	4
<b>III</b>	<b>Goals</b>	<b>5</b>
<b>IV</b>	<b>General instructions</b>	<b>6</b>
<b>V</b>	<b>Mandatory part</b>	<b>7</b>
<b>VI</b>	<b>Bonus part</b>	<b>8</b>
<b>VII</b>	<b>Turn-in and peer-evaluation</b>	<b>9</b>

# Chapter I

## Forewords

Genenis of the Bible, Sumerian-first edition:

Gur fabj tybjf juvgr ba gur zbhagnva gbavtug  
Abg n sbbgcevag gb or fra.  
N xvatzbz bs vfbyngvba,  
naq vg ybbxf yvvr V'z gur Dhrra  
Gur jvaq vf ubjyvat yvvr guvf fjveyvat fgbez vafvqr  
Pbhyqa'g xrrc vg va;  
Urnira xabjf V'ir gevrq1

Qba'g yrg gurz va,  
qba'g yrg gurz fir  
Or gur tbbq tvey lbh nyjnlf unir gb or  
Pbaprny, qba'g srly,  
qba'g yrg gurz xabj  
Jryy abj gurl xabj1

Yrg vg tb, yrg vg tb  
Pna'g ubyq vg onpx nalzber1

Yrg vg tb, yrg vg tb  
Ghea njnl naq fynz gur qbbe  
V qba'g pner  
jung gurl'er tbvat gb fnl  
Yrg gur fgbez entr ba.  
Gur pbyq arire obgurerc zr naljnl3

Vg'f shaal ubj fbzr qvfgnapr  
Znxrf rircluvat firz fznyy  
Naq gur srnef gung bapr pbagebyyrc zr  
Pna'g trg gb zr ng nyy

Vg'f gvzr gb fir jung V pna qb  
Gb grfg gur yvzvfg naq oernx guebhtu  
Ab evtug, ab jebat, ab ehvrf sbe zr,  
V'z serr!

Yrg vg tb, yrg vg tb  
V nz bar jvgr gur jvaq naq fxl  
Yrg vg tb, yrg vg tb  
Lbh'yy arire fir zr pel  
Urer V fgnaq

Naq urer V'yy fgnl  
Yrg gur fgbez entr ba1

Zl cbjre syheevrf guebhtu gur nve vagb gur tebhaq  
Zl fbhy vf fcevenyvat va sebmra senpgnyf nyy nebhaq  
Naq bar gubhtug pelfgnyyvmrf yvyr na vpl oynfg  
V'z arire tbvat onpx, gur cnfg vf va gur cnfg

Yrg vg tb, yrg vg tb  
Naq V'yy evfr yvyr gur oernx bs qnja  
Yrg vg tb, yrg vg tb  
Gung cresrpg tvey vf tbar  
Urer V fgnaq  
Va gur yvtug bs qnl  
Yrg gur fgbez entr ba

Gur pbyq arire obgurercq zr naljnl!

# Chapter II

## Introduction

### II.0.1 Filesystem

Filesystems ! Yay !

*In computing, a file system (or filesystem) is used to control how data is stored and retrieved. Without a file system, information placed in a storage area would be one large body of data with no way to tell where one piece of information stops and the next begins.*

*By separating the data into individual pieces, and giving each piece a name, the information is easily separated and identified. Taking its name from the way paper-based information systems are named, each group of data is called a "file". The structure and logic rules used to manage the groups of information and their names is called a "file system".*

Filesystems are structures kernel-side for creating, storing and modifying files / directory. The main purpose of a filesystem is to interact with the hard drive and the system. You know, creating files, chmod, symbolic links, that sort of stuff. Today, filesystems are more than an interface HD / Kernel with 'new' technologies like SSH, Network file system, torrent, etc.

So yeah, filesystem are kinda important to an OS, because, you know, files. Remember the intro of `lk_driver_and_keyboard` ?

*Writing a kernel is only a part of writing an operating system, and to do so, one needs a robust and reliable interface between kernelspace and userspace. This interface is the syscalls.*

*Lies, lies and lies. Kernel and Syscalls cannot be considered as an OS if there are no drivers. `lk_process_and_mem` author is a moron.*

Pff, that's bullshit. How can you call something an Operating System if I can't save my sweet zsh config of mine ? Those subjects authors obviously don't know shit about OS, I tell you.

# Chapter III

## Goals

Welcome to the marvellous world of filesystems !

Aaaah, those 600 lines .h files, structures with 350 members, quadruple linked-list, childs in childs in father of a non existing file, ..

Happiness.

Down to the serious stuff, if you play around with this subject, you should be able to:

- Create a new filesystem (from scratch !) in the Linux Kernel
- Understand and work with superblocks
- Understand and work with inodes
- Understand and work with rights, links and interact with other filesystems.
- Link that filesystem to the fabulous world of user space

Exciting stuff, eh ?

# Chapter IV

## General instructions

- For this subject, you must use your custom linux distribution, made in the ft\_linux subject.
- You must use a kernel version 4.x. Stable or not, as long as it's a 4.x version.
- A Makefile must be turn in
- *\*ALL\** the memory allocated must be properly released. Note the *\*PROPERLY\**
- *\*ALL\** intern kernel function calls must be verified if its needed. Don't want a Kernel Panic, do you ?
- *\*ALL\** of the requests and registers declared to the Kernel must be properly destroyed when the module exit.

# Chapter V

## Mandatory part

You must create a new filesystem module.  
The filesystem must be named "fortytwofs", and a

```
mount -o loop -t fortytwofs dir image
```

should work.

This filesystem should implement the following features (BOTH kernel-side and user-side):

- Directory creation
- File creation
- Owner, group of the entry
- Chmod of the entry
- Links (hard and symbolics)

In this module you MUST use:

- Superblocks. Both Linux and customs.
- Inodes. Both Linux and customs.

With those informations in mind, you will also need to make a binary that format a disk image for your filesystem. The language is free.

That binary must take a disk image as a parameter and prepare superblocks, inodes and all you need in order for your filesystem to mount / run.



# Chapter VI

## Bonus part

Be creative ! You could, for example, create syscalls / misc device for user-friendly information about your filesystem.  
You can also make advanced stats with JSON output for example.

# Chapter VII

## Turn-in and peer-evaluation

Turn your work in using your `Git` repository, as usual. Only work present on your repository will be graded in defense.

Your code will be running on your custom linux distribution, keep it around for the evaluation.