

## **VE482 — Introduction to Operating Systems**

### *Contributing to Minix 3*

Manuel — UM-JI (Fall 2020)

This document was composed by a volunteer who desired to contribute to the development of Minix 3. It explains in simple and clear terms how he contacted Minix development team and started to contribute.

The original post can be found [here](#).

## **Why this post?**

In this post I am going to describe my experience with minix community. Recently I made a small code contribution to minix3 (committed on 15 Oct 2012), it is available [here](#), as some of fellow ILUGC (Linux User's group) members were interested to know about my experience I was requested to write a blog post about it. As this is my first code contribution to an open source project I think it will be helpful / interesting to people who want to start code contribution.

## **My plans**

I wanted to contribute to some open source project, I was searching for some place to contribute. My initial thought was to contribute to Linux kernel in form of device drivers as contributing to core kernel needs a good knowledge and time. All the devices that I had drivers for Linux (I am lucky and unlucky :P). I was not able to find device that had documentation but no driver (very rare possibility). So I thought of working for some other project. I wanted to limit myself to OS, drivers, graphics or mathematics related projects.

## **Initial Stages**

I came across minix project. Most of them who had taken a OS course or know Linux history must be aware of minix. At that time minix was for students and contribution to it was not much encouraged (it was intended for teaching purpose). But as of minix 3 and later the license changed to BSD and people are encouraged to contribute to it (I am not aware of the status of minix 2 or other versions). I read the wishlist, roadmap described in the minix3 website. I also downloaded, installed minix3 in a VM in my machine. So far things went well. I chose to work on FAT file system driver (or in minix parlance a file system server) as I did not need any special device and I can test it from within a VM. I started working on FAT 16 and FAT 32 file system driver implementation for minix 3. I wanted to write from scratch so I did not try to port code from other projects (I was able to find some fat16/32 BSD licensed implementation but I did not try to port them intentionally) as I wanted to learn as much as possible.

## **Hard road ahead**

As I have made a choice on the work I started working. I tried to go through many online documentation on FAT 16/32 file systems. As I was new to it I read from many sources and I reached a state where there

was conflict between sources and I was not sure which source is correct. Some authentic and correct source were covered by NDA or some sort of license which does not permit me to write a BSD licensed code using those. I did not go through other fat implementation (like linux's or \*BSD's as I want to do it fully myself). At this stage I had done parsing BIOS parameter block but I did not spend much time on designing or architecturing the code – the code obviously became ugly (not well written).

## **Jump from FAT implementation**

At this point I did not have a stern documentation, not well designed code so I stopped here. I paused I did not actually stop. My mind said “make smaller contributions while working on fat implementation slowly”. I followed my mind. I used to be present in #minix IRC channel as much as possible. I started to speak with people in #minix. I came to know pikpik was working on “removing usage of 64 bit functions” and it seemed I could work on that (pikpik actually had a lot of work done and it was available in his branch and not merged). It seemed doable for me and I plunged into working it. Got the diff's from pikpik and started working from there.

## **Moving forward in “Removing usage of 64bit functions”**

I will describe what the phrase “removing usage of 64bit functions” mean when required. I just checked if the patches are correct – by manually going through the changes (not by testing the changes). I wanted to commit the changes module by module. I chose to work on the changes made by pikpik in top utility (top.c file). Found some bugs and fixed them. As soon as I finished this I went to the minix mailing list and fired a mail requesting to merge the changes along with a link to the change (it is a github link and github allows inline commenting).

Many people in minix group took great effort to review the code that I wrote and I got valuable comments. After some time I got a reply from Saturn (this is his IRC nick) that the changes I have made can be done by some simple scripts and does not do the real required change. Yes I did not go through the code and I did not understand that just changed things mechanically. The code required real change which I missed. I would really thank him as he pinpointed out that I have not done worthy code (a python or perl script could actually do what I had done at that time). Then I decided to plunge into the code to make necessary change as suggested by him.

I would like to explain the phrase “removing usage of 64bit functions” and what the “real change” phrase seen in previous paragraph meant. In Minix to do addition, subtraction, multiplication, division, comparison etc of 64bit data type (`uint64_t` or `unsigned long`) the corresponding functions were called (not operators). This was due to old compiler which did not support these (kind of legacy stuff). Now the compiler has been changed (minix uses gcc and llvm) but the code remains the same in many places. I planned to replace the functions with operators (really a mechanical job). In top utility (top.c file) there was a trick applied. The 64bit values were converted to 32bit values using a shift, the resultant 32bit values and operators are used. It worked but the code does not look cool, Saturn wanted to remove this trick. Removal of the trick(64bit to 32bit conversion using a shift and working on it) is what the phrase “real change” meant.

## Plunge into the “Real change”

I started to work on it. I went through the code flow and understood what I should do. Made the necessary changes in code (implementing was very simple once I got the idea). Requested for merge again through mailing list. This time I got a small amount of review comments (from Saturn again) and it was code readability change (no change in functionality or other major code change). I did them and got back again with new changes.

The code change was acknowledged by Saturn and he said the code is in a good shape and can be merged into minix tree :). After a day I guess there was another mail saying that the code is merged and is available in minix mainline tree. I was very happy to see my code change merged :) :).

## Contributing to open source projects

Tips for newbies who want to contribute to open source projects, especially minix:

- Be available in the projects IRC channel and subscribe to mailing lists.
- Ask the doubts that you have. Even if it seems silly ask it, people will be there to reply or direct you to correct place.
- Reviews and comments are for improving code quality and making you understand how good code is made. Don't take it personal or thing them as negative.
- Ask for TODO tasks, wish list or road map. But before that explain them what you know and what is your area of expertise or interest. In companies people will be assigned tasks but in case of open source work people will take work that suits them (what they like) and work on it. So don't ask someone to assign tasks to you.
- Most of the people in IRC are great. Especially in #minix most were responsive, encouraging – pikpik, cphuntington97 and others (whom I could have missed or forgot in this late hours) were very encouraging and supportive.

If the above basic steps are followed things will go great :). All the best for your contribution to open source projects.

## Notes

Concluding comments:

- I will resume my work on fat file system implementation soon while carrying out other tasks.
- I have written what came through my mind, it may not be well structured. If there is a need to re-structure it or modify some lines please let me know through comments.
- Please let me know about your thoughts. Ask questions if there are any in comments and I will be glad to explain.