

# Redundant GRUB2 env file




GRUB2 and 3mdeb minisummit 2019

Maciej Pijanowski

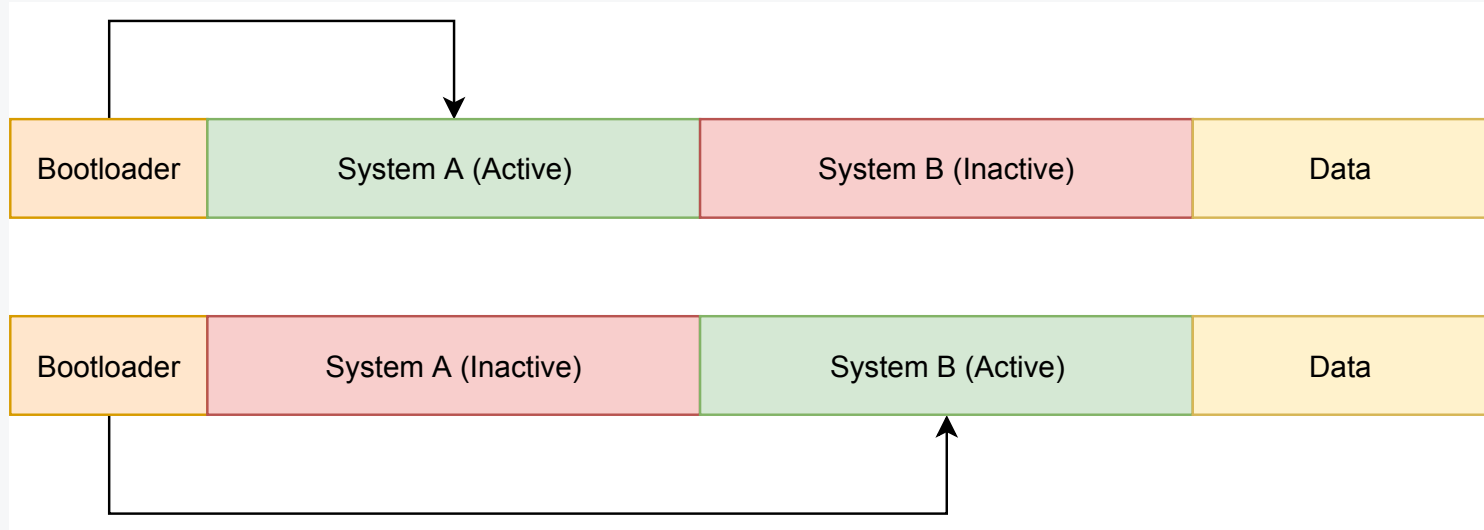




Maciej Pijanowski  
*Embedded Systems Engineer*

-  @macpijan
-  maciej.pijanowski@3mdeb.com
-  linkedin.com/in/maciej-pijanowski-9868ab120
- interested in:
  - Embedded Linux and bootloaders
  - build systems (OpenEmbedded/Yocto)
  - software updates

- What?
  - basically a key-value data store
  - stored somewhere on a boot device
- Why?
  - drive the system update process
  - the kernel / rootfs to load
  - update in progress flag
  - bootcount statistics



- decide which system partition to boot based on the bootloader environment

- a preallocated 1024-byte file
- GRUB `load_env` command can be used to load variables
- GRUB `save_env` command can be used to save variables
- `grub-editenv` - a userspace tool to edit the environment block
- likely not extensively used in typical desktop scenarios
- the main typical usecase I found: set the default menu entry to boot from

- No integrity check of the GRUB environment
  - No basic verification (checksum) of the environment content
- The environment block can be corrupted
  - system crash / power loss in the middle of write procedure
  - usually resides on the vfat partition (no FS block journaling)
- Maybe the environment block was not designed with such usecases in mind, but in embedded systems we want to be able to rely on the bootloader environment during the update process

- a framework for updating embedded systems
- efficient and safe way to update an emebdded system
- supports local and remote updates
- supports multiple update strategies and policies
- easily integrates with common build systems (Yocto, Buildroot)

- grub-editenv code could not be reused (GPLv2 vs GPLv3)

As Stefano Babic wrote:

First at all, there is a quite big licensing issue [...] is licensed under GPLv3, and this conflicts with SWUpdate (GPLv2). [...] This means that manufacturers must provide with whatever information or data is necessary to install modified software on the device. This is a big issue on embedded device, and one of the reasons why a lot of projects (including kernel) are GPLv2 (or LGPL).

<https://groups.google.com/d/msg/swupdate/DsAVmjBwhMg/260yLdCiCAAJ>



- SWUpdate plugin for GRUB env block handling:

<https://groups.google.com/d/msg/swupdate/fQ0gWQ6tHFo/9zdRrqZUCwAJ>

- Concern about the environment block corruption:

[...] there is a power-cut when the fwrite() is called. The grubenv is in most cases truncated to zero after reboot or corrupted. This can lead that the device cannot boot.

- Implemented solution to *reduce* the risk:
  - read grubenv, but write it to a grubenv.new
  - when fwrite completes, use the `mv grubenv.new grubenv` system call which is at least atomic
  - in most cases we should have a working environment copy (either the old one or the new one)

Stefano Babic feedback:

We do not reach the security of U-Boot with the two copies, but we reduce the risks. To have the same safety as U-Boot, GRUB should manage itself two copies of the environment, but this feature is as far as I know not (yet ?) implemented.

- Set of shell script wrappers over grub-editenv tool:

<https://github.com/mendersoftware/grub-mender-grubenv>

- Redundancy (double copy of environment block)
- lock file for each block
  - a grubenv file with just one variable: editing (can be either 0 or 0)
- hash file with the expected sha256sum of the lock file

- To increase the reliability, handling the bootloader integrity and redundancy should ideally be handled by the GRUB itself
- To make it easier for other projects to integrate with GRUB, a library (and a standalone `grub-editenv` tool) for interacting with GRUB environment could be implemented as a separate project with a less strict license
  - such as [libubootenv](#) was rewritten based on the U-Boot `fw_setenv` tool with a LGPLv2.1 license
- Retaining of backwards compatibility is important and can be difficult at the same time (e.g. the structure of environment block cannot be changed)

- GRUB environment block:  
[https://www.gnu.org/software/grub/manual/grub/html\\_node/Environment-block.html](https://www.gnu.org/software/grub/manual/grub/html_node/Environment-block.html)
- GRUB integration with SWUpdate:  
<https://groups.google.com/d/msg/swupdate/fQ0gWQ6tHFo/9zdRrqZUCwAJ>
- Mender's grub-editenv wrapper:  
[https://github.com/mendersoftware/grub-mender-grubenv/blob/master/fw\\_printenv](https://github.com/mendersoftware/grub-mender-grubenv/blob/master/fw_printenv)
- libubootenv announcement:  
<https://groups.google.com/d/msg/swupdate/fQ0gWQ6tHFo/9zdRrqZUCwAJ>
- libubootenv discussion: <http://u-boot.10912.n7.nabble.com/SWUpdate-U-Boot-environment-library-dependency-td340530.html>

# Q&A