

OS2CSM Idea

Veit Kannegieser wrote a program, called `os2csm`. It was inspired by some DOS program, which modifies `config.sys` in memory. `Os2csm` installs itself in place of `os2ldr`, and renames `os2ldr` to `os2ldr.bin`. `Os2csm` hooks onto `int 13h` interrupt handler and analyzes the information read through `int 13` routine. The `config.sys` file contains directives, analogous to C preprocessor directives, and each 512 bytes of `config.sys` file (each disk sector of `config.sys`) has a special comment with a special signature in it. The procedure, which hooks onto `int 13h` interrupt handler, checks each sector for these signatures, and if it finds them, it assumes that the `config.sys` file is being read. So, it substitutes preprocessor defines and patches `config.sys` on the fly.

The `config.sys` preprocessor is useful to substitute some parts of `config.sys` by some variables (aka preprocessor symbols). These variables can be obtained from menus, which are presented to user by `os2csm`. The `os2csm` then loads `os2ldr` from the file `os2ldr.bin`; the loader starts the kernel. When the kernel reads `config.sys`, `os2csm` changes it on the fly, and substitutes variables. So, settings set by user substitute variables in `config.sys`. This way, the set of parameters is passed to the kernel through `config.sys` preprocessor.

OS2CSM is now used in eCS (in eCS demo disk and in eCS installation disk). You can download the demo disk from <http://ecomstation.com/>.

We can use this idea in our bootloader, so, we may implement a simple preprocessor with syntax similar to the C preprocessor (or, even, PPWizard syntax). I propose to embed this preprocessor into loader and instead of hooking onto `int 13` routines, we can hook onto blackbox routines. The configs are read through blackbox (not `minifsd`, like in present OS/2!). I suggest to explicitly mark a number of files as configs in the loader config. So, the signatures in each `config.sys` file sector are not needed. And also we can use more attractive and beautiful looking syntax instead of ugly `os2csm` syntax.

The preprocessor directives may include analogues to `"#define"`, `"#include"`, `"#ifdef"` etc., so, it will be possible to define symbols, include one config file to another, and to conditionally include files or define symbols. So, the resulting config file can be flexibly constructed from parts and variables can be substituted in it.

The preprocessor idea complements the GRUB idea of passing command lines to a kernel and modules. That is, we suggest to use passing parameters to the kernel as multiboot command lines, as well as through substituting variables in config files. So, we combine these two approaches.

From:
<http://osfree.org/doku/> - **osFree wiki**

Permanent link:
<http://osfree.org/doku/en:docs:boot:freldr:os2csm>

Last update: **2014/05/21 23:13**

