

Cmd.c32

From Syslinux Wiki

Contents

- 1 Introduction
- 2 Example (I)
- 3 Example (II)
- 4 Example (III)
- 5 Example (IV)
- 6 Example (V)
- 7 Example (VI)

Introduction

cmd.c32 executes its arguments as if they were typed in directly at the Syslinux prompt. It is commonly used to reference other labels within a cfg file. It can also partially replace the need for a global append directive and/or variables in a cfg file. Let's see some examples.

The following code shows a typical boot menu.

```
UI menu.c32

LABEL normal
LINUX bzImage
INITRD initrd.img
APPEND option_1 option_2 option_3

LABEL live
LINUX bzImage
INITRD initrd.img
APPEND option_1 option_2 option_3 livemedi
# The "live" label entry adds one additional option
# in comparison to the "normal" label entry.
```

The following boot methods are equivalent alternatives:

- A_ Start the second entry in the above menu. Or,
- B_ Select the first entry in the above menu; press [TAB]; typing in " livemedi" results in something like ".linux bzImage initrd=initrd.img option_1 option_2 option_3 livemedi"; press [ENTER]. Or,
- C_ In a clean "boot:" prompt, type in "live"; [ENTER]. Or,
- D_ In a clean "boot:" prompt, type in "normal livemedi"; [ENTER].

Example (I)

The following label entry is also equivalent:

```
LABEL live
COM32 cmd.c32
APPEND normal livemedi
# Equivalent to typing "normal livemedi" in CLI.
```

By using `cmd.c32` in the `cfg` file, editing the 'APPEND' line in the 'normal' boot entry will directly affect any other boot entry that includes 'cmd.c32 normal'.

Example (II)

The following `cfg` file:

```
UI menu.c32
```

```

LABEL normal
LINUX bzImage
INITRD initrd.img
APPEND option_1 option_2 option_3

LABEL us
LINUX bzImage
INITRD initrd.img
APPEND option_1 option_2 option_3 kbrd=us

LABEL de
LINUX bzImage
INITRD initrd.img
APPEND option_1 option_2 option_3 kbrd=de

LABEL fr
LINUX bzImage
INITRD initrd.img
APPEND option_1 option_2 option_3 kbrd=fr

LABEL es
LINUX bzImage
INITRD initrd.img
APPEND option_1 option_2 option_3 kbrd=es

```

is equivalent to:

```

UI menu.c32

LABEL normal
LINUX bzImage
INITRD initrd.img
APPEND option_1 option_2 option_3

LABEL us
COM32 cmd.c32
APPEND normal kbrd=us

LABEL de
COM32 cmd.c32
APPEND normal kbrd=de

LABEL fr
COM32 cmd.c32
APPEND normal kbrd=fr

LABEL es
COM32 cmd.c32
APPEND normal kbrd=es

```

Editing the second cfg file is simpler: just change the 'normal' boot entry.

Example (III)

Let's take the last boot entry from the above #Example (II):

```

LABEL es
COM32 cmd.c32
APPEND normal kbrd=es

```

In the menu, select this entry and press [TAB] so as to edit the command line before acting on it. The CLI will show:

```
>.com32 cmd.c32 normal kbrd=es
```

which is shorter than the original:

```
>.linux bzImage initrd=initrd.img option_1 option_2 option_3 kbrd=es
```

This example shows one **disadvantage** of using cmd.c32. To get a command line such as:

```
>.linux bzImage initrd=initrd.img option_3 kbrd=es
```

the user needs to select the first entry, 'normal', press [TAB] and edit the whole command by deleting the unwanted options ('option_1 option_2') and typing in the additional wanted ones ('kbrd=es').

Example (IV)

More than one additional option can be added to the original label.

```

LABEL live_fr
COM32 cmd.c32
APPEND normal livemedia kbrd=fr

```

Example (V)

Changing the `INITRD` is possible. Let's follow the explanation.

The 'normal' entry:

```

LABEL normal
LINUX bzImage
INITRD initrd.img
APPEND option_1 option_2 option_3

```

is seen in CLI as:

```
>.linux bzImage initrd=initrd.img option_1 option_2 option_3
```

Since the last `INITRD` in a boot command overrides previous ones, then the following command (with an additional '`initrd=new.img`')

```
>.linux bzImage initrd=initrd.img option_1 option_2 option_3 initrd=new.img
```

is in fact equivalent to:

```
>.linux bzImage option_1 option_2 option_3 initrd=new.img
```

So the following entry in the `cfg` file:

```

LABEL new_initrd
COM32 cmd.c32
APPEND normal initrd=new.img

```

uses '`initrd=new.img`' instead of the original '`initrd=initrd.img`'.

Example (VI)

Merging more than one `INITRD` is possible.

The command line:

```
>.linux bzImage option_1 option_2 option_3 initrd=initrd.img,new.img
```

uses both, `initrd.img` and `new.img`. The following entry is equivalent:

```

LABEL new_initrd_merged
COM32 cmd.c32
APPEND normal initrd=initrd.img,new.img

```

since the original '`initrd=initrd.img`' in 'normal' is "replaced" by the last '`initrd=initrd.img,new.img`'.

Retrieved from "<http://www.syslinux.org/wiki/index.php?title=Cmd.c32&oldid=4529>"

Category: Modules

- This page was last modified on 24 February 2017, at 18:58.
- This page has been accessed 3,683 times.