

Comboot/menu.c32

From Syslinux Wiki

Additional items that must be documented (in both doc/menu.txt and this page) since 4.0:

- New **MENU IMMEDIATE** --- permit hotkeys to activate immediately without needing Enter.
- ~~New **MENU HELP** statement to display fullscreen help text as a result of a menu selection. *documented*~~
- ~~New **MENU HIDDENKEY** command to provide a one keystroke way to activate a boot option from a hidden menu intro screen. *documented*~~
- ~~New **MENU SAVE** statement. *documented*~~
- ~~New **MENU CLEAR** statement. *documented*~~
- ~~New **MENU SHIFTKEY** statement. *documented*~~
- ~~New **MENU RESOLUTION** statement. *documented*~~

(See also Syslinux 4 Changelog and How do I configure Syslinux)

(Note that since version 3.74, the default directive usage has changed. Use **UI** instead) (See **UI** mode or Changes in 3.74)

Note: Some minor updated documentation can be seen at the Menu wiki page.

The content of doc/menu.txt (release 3.72, with some updates):

There are two menu systems included with Syslinux, the advanced menu system, and the simple menu system.

THE ADVANCED MENU SYSTEM

The advanced menu system, written by Murali Krishnan Ganapathy, is located in the `cmenu/` sub-directory. It allows the user to create hierarchical sub-menus, dynamic options, check-boxes, and just about anything you want. It requires the menu to be compiled from a simple C file, see `menu/simple.c` (<http://git.kernel.org/?p=boot/syslinux/syslinux.git;a=blob;f=menu/simple.c;h=92e8ab125f7c76f45e5f912cb742b9299e9110a0;hb=ed67201fcc004ccb0eb20e5489d71ed69cfb7428>) and `menu/complex.c` (<http://git.kernel.org/?p=boot/syslinux/syslinux.git;a=blob;f=menu/complex.c;h=94627c4f79878ee996a83913c27958a3e24364c0;hb=ed67201fcc004ccb0eb20e5489d71ed69cfb7428>) for examples.

The advanced menu system doesn't support serial console at this time.

See `menu/README` (<http://git.kernel.org/?p=boot/syslinux/syslinux.git;a=blob;f=menu/README;h=d585d2fa792385fa051481deca026c6eb4577a13;hb=ed67201fcc004ccb0eb20e5489d71ed69cfb7428>) for more information.

THE SIMPLE MENU SYSTEM

The simple menu system is a single module located at `com32/menu/vesamenu.c32` (graphical) or `com32/menu/menu.c32` (text mode only). It uses the same configuration file as the regular Syslinux command line, and displays all the `LABEL` statements.

To use the menu system, simply make sure `[vesa]menu.c32` is in the appropriate location for your boot medium (the same directory as the configuration file for SYSLINUX, EXTLINUX and ISOLINUX, and the same directory as pxelinux.0 for PXELINUX), and put the following options in your configuration file:

```
-----
#DEFAULT menu.c32
#PROMPT 0
-----
```

There are a few menu additions to the configuration file, all starting with the keywords `MENU` or `TEXT`. As the rest of the Syslinux config file language, it is case insensitive:

MENU TITLE *title*

Give the menu a title. The title is presented at the top of the menu.

MENU HIDDEN

Do not display the actual menu unless the user presses a key. All that is displayed is a timeout message.

MENU HIDDENKEY *key[,key...] command...*

If the key used to interrupt MENU HIDDEN is "key", then execute the specified command instead of displaying the menu.

Currently, the following key names are recognized:

```
Backspace, Tab, Enter, Esc, Space, F1..F12, Up, Down, Left, Right, PgUp, PgDn, Home, End, Insert,
Delete
```

... in addition to all single characters plus the syntax "^X" for "Ctrl-X". Note that single characters are treated as case sensitive, so "A" and "a" can bind different commands. The same command can be bound to different keys by giving a comma-separated list of keys:

```
menu hiddenkey A,a key_a_command
```

MENU CLEAR

Clear the screen when exiting the menu, instead of leaving the menu displayed. For vesamenu, this means the graphical background is still displayed without the menu itself for as long as the screen remains in graphics mode.

MENU SHIFTKEY

Exit the menu system immediately unless either the Shift or Alt key is pressed, or Caps Lock or Scroll Lock is set.

MENU SEPARATOR

Insert an empty line in the menu.

MENU LABEL *label*

(Only valid after a LABEL statement.)

Changes the label displayed for a specific entry. This allows you to have a label that isn't suitable for the command line, for example:

```
# Soft Cap Linux
LABEL softcap
  MENU LABEL Soft Cap ^Linux 9.6.36
  KERNEL softcap-9.6.36.bzi
  APPEND whatever

# A very dense operating system
LABEL brick
  MENU LABEL ^Windows CE/ME/NT
  KERNEL chain.c32
  APPEND hd0 2
```

The ^ symbol in a MENU LABEL statement defines a hotkey. The hotkey will be highlighted in the menu and will move the menu cursor immediately to that entry.

Reusing hotkeys is disallowed, subsequent entries will not be highlighted, and will not work.

Keep in mind that the LABELS, not MENU LABELS, must be unique, or odd things will happen to the command-line.

MENU INDENT *count*

(Only valid after a LABEL statement.)

Will add "count" spaces in front of the displayed menu entry.

MENU DISABLE

(Only valid after a LABEL statement.)

Makes the entry unselectable. This allows you to make a section in your menu with different options below it. for example:

```
# Entries for network boots
LABEL -
  MENU LABEL Network:
  MENU DISABLE

# Soft Cap Linux
LABEL softcap
  MENU LABEL Soft Cap ^Linux 9.6.36
```

```

        MENU INDENT 1
        KERNEL softcap-9.6.36.bzi
        APPEND whatever

# Dos 6.22
LABEL dos
    MENU LABEL ^Dos 6.22
    MENU INDENT 1
    KERNEL memdisk
    APPEND initrd=dos622.imz

# Separator
MENU SEPARATOR

# Entries for local boots
LABEL -
    MENU LABEL Local:
    MENU DISABLE

# Windows 2000
LABEL w2k
    MENU LABEL ^Windows 2000
    MENU INDENT 1
    KERNEL chain.c32
    APPEND hd0 1

# Windows XP
LABEL xp
    MENU LABEL Windows ^XP
    MENU INDENT 1
    KERNEL chain.c32
    APPEND hd0 2

```

MENU HIDE

(Only valid after a LABEL statement.)

Suppresses a particular LABEL entry from the menu.

MENU DEFAULT

(Only valid after a LABEL statement.)

Indicates that this entry should be the default for the particular (sub)menu. See also the DEFAULT directive. If no default is specified, use the first one.

TEXT HELP

Help text ...

... which can span multiple lines

ENDTEXT

(Only valid after a LABEL statement.)

Specifies a help text that should be displayed when a particular selection is highlighted.

MENU PASSWD *passwd*

(Only valid after a LABEL statement, or a MENU BEGIN statement.)

Sets a password on this menu entry. "*passwd*" can be either a cleartext password or a password encrypted with one of the following algorithms:

```

MD5 (Signature: $1$)
SHA-1 (Signature: $4$)
SHA-2-256 (Signature: $5$)
SHA-2-512 (Signature: $6$)

```

Use the included Perl scripts "shalpass" or "md5pass" to encrypt passwords. MD5 passwords are compatible with most Unix password file utilities; SHA-1 passwords are probably unique to Syslinux; SHA-2 passwords are compatible with very recent Linux distributions. Obviously, if you don't encrypt your passwords they will not be very secure at all.

If using passwords, make sure to use "NOESCAPE 1" and "PROMPT 0", and either set "ALLOWOPTIONS 0" or use a master password (see below).

If *passwd* is an empty string, this menu entry can only be unlocked with the master password.

MENU MASTER PASSWD *passwd*

Sets a master password. This password can be used to boot any menu entry, and is required for the [Tab] and [Esc] keys to work.

MENU RESOLUTION *width height*

Requests a specific screen resolution when in graphics mode. The default is "640 480" corresponding to a resolution of 640x480 pixels, which all VGA-compatible monitors should be able to display.

If the selected resolution is unavailable, the text mode menu is displayed instead.

MENU BACKGROUND *background*

For `vesamenu.c32`, sets the background image. The *background* can either be a color (see `MENU COLOR`) or the name of an image file, which should be the size of the screen (normally 640x480 pixels, but see `MENU RESOLUTION`) and either in PNG, JPEG or LSS16 format.

MENU BEGIN [*tagname*] **MENU END**

Begin/end a submenu. The entries between `MENU BEGIN` and `MENU END` form a submenu, which is marked with a > mark on the right hand of the screen. Submenus inherit the properties of their parent menus, but can override them, and can thus have their own backgrounds, master passwords, titles, timeouts, messages and so forth.

MENU GOTO *tagname*

(Only valid after a LABEL statement.)

This label will transfer to the named submenu instead of booting anything. To transfer to the top-level menu, specify "`menu goto .top`".

MENU EXIT [*tagname*]

(Only valid after a label statement inside MENU BEGIN ... MENU END)

Exit to the next higher menu, or, if *tagname* is specified, to the named menu.

MENU QUIT

(Only valid after a LABEL statement.)

This label quits the menu system.

WARNING: if `MENU MASTER PASSWD` or `ALLOWOPTIONS 0` is set, this will still allow exiting to the CLI; however, a separate `MENU PASSWD` can of course be set for this label.

MENU START

(Only valid inside MENU BEGIN ... MENU END)

Indicates that the menu system should start at the menu being defined instead of at the top-level menu. See also the `DEFAULT` directive.

DEFAULT *label*

Set the global default. If "*label*" points into a submenu, that menu becomes the `start menu`; in other words, this directive has the same effect as both `MENU DEFAULT` and `MENU START`.

For backwards compatibility with earlier versions of Syslinux, this behavior is ignored unless the configuration file also contains a UI directive.

Note: the CLI accepts options after the *label*, or even a non-label. The menu system does not support that.

MENU SAVE **MENU NOSAVE**

Remember the last selected entry and make that one the *default* for the next boot. A password-protected menu entry is **not** saved. This requires the ADV data storage mechanism, which is currently only implemented for EXTLINUX, although the other Syslinux derivatives will accept the command (and ignore it.)

NOTE: MENU SAVE stores the LABEL tag of the selected entry; this mechanism therefore relies on LABEL tags being unique. On the other hand, it handles changes in the configuration file gracefully.

NOTE: In software RAID-1 setups, MENU SAVE only stores the *default* label on the actual boot disk. This may lead to inconsistent reads from the array, or unexpectedly change the default label after array resynchronization or disk failure.

The MENU SAVE information can be fully cleared with "extlinux --reset-adv <bootdir>".

A MENU SAVE or MENU NOSAVE at the top of a (sub)menu affects all entries underneath that (sub)menu except those that in turn have MENU SAVE or MENU NOSAVE declared. This can be used to only save certain entires when selected.

INCLUDE *filename* [*tagname*]

MENU INCLUDE *filename* [*tagname*]

Include the contents of the configuration file *filename* at this point.

In the case of MENU INCLUDE, the included data is only seen by the menu system; the core syslinux code does not parse this command, so any labels defined in it are unavailable.

If a *tagname* is included, the whole file is considered to have been bracketed with a MENU BEGIN *tagname* ... MENU END pair, and will therefore show up as a submenu.

MENU AUTOBOOT *message*

Replaces the message "*Automatic boot in # second[,s]...*". The symbol # is replaced with the number of seconds remaining. The syntax "{singular,[dual,]plural}" can be used to conjugate appropriately.

MENU TABMSG *message*

Replaces the message "*Press [Tab] to edit options*".

MENU NOTABMSG *message*

Takes the place of the TABMSG message if option editing is disabled. Defaults to blank.

MENU PASSPROMPT *message*

Replaces the message "*Password required*".

MENU COLOR *element ansi foreground background shadow*

Sets the color of element "*element*" to the specified color sequence:

screen	Rest of the screen
border	Border area
title	Title bar
unsel	Unselected menu item
hotkey	Unselected hotkey
sel	Selection bar
hotsel	Selected hotkey
disabled	Disabled menu item
scrollbar	Scroll bar
tabmsg	Press [Tab] message
cmdmark	Command line marker
cmdline	Command line
pwdborder	Password box border
pwdheader	Password box header
pwdentry	Password box contents
timeout_msg	Timeout message
timeout	Timeout counter
help	Help text
msgXX	Message (F-key) file attribute XX

... where XX is two hexadecimal digits (the "plain text" is 07).

"ansi" is a sequence of semicolon-separated ECMA-48 Set Graphics Rendition (<ESC>[m) sequences:

```

0      reset all attributes to their defaults
1      set bold
4      set underscore (simulated with color on a color display)
5      set blink
7      set reverse video
22     set normal intensity
24     underline off
25     blink off
27     reverse video off
30     set black foreground
31     set red foreground
32     set green foreground
33     set brown foreground
34     set blue foreground
35     set magenta foreground
36     set cyan foreground
37     set white foreground
38     set underscore on, set default foreground color
39     set underscore off, set default foreground color
40     set black background
41     set red background
42     set green background
43     set brown background
44     set blue background
45     set magenta background
46     set cyan background
47     set white background
49     set default background color

```

These are used (a) in text mode, and (b) on the serial console.

"foreground" and "background" are color codes in #AARRGGBB notation, where AA RR GG BB are hexadecimal digits for alpha (opacity), red, green and blue, respectively. #00000000 represents fully transparent, and #ffffff represents opaque white.

"shadow" controls the handling of the graphical console text shadow. Permitted values are "none" (no shadowing), "std" or "standard" (standard shadowing - foreground pixels are raised), "all" (both background and foreground raised), and "rev" or "reverse" (background pixels are raised.)

If any field is set to "*" or omitted (at the end of the line) then that field is left unchanged.

The current defaults are:

```

menu color screen      37;40      #80ffffff #00000000 std
menu color border      30;44      #40000000 #00000000 std
menu color title        1;36;44    #c00090f0 #00000000 std
menu color unsel        37;44      #90ffffff #00000000 std
menu color hotkey       1;37;44    #ffffff #00000000 std
menu color sel          7;37;40    #e0000000 #20ff8000 all
menu color hotssel      1;7;37;40  #e0400000 #20ff8000 all
menu color disabled     1;30;44    #60cccccc #00000000 std
menu color scrollbar    30;44      #40000000 #00000000 std
menu color tabmsg       31;40      #90ffff00 #00000000 std
menu color cmdmark      1;36;40    #c000ffff #00000000 std
menu color cmdline      37;40      #c0ffffff #00000000 std
menu color pwdborder    30;47      #80ffffff #20ffffff std
menu color pwdheader    31;47      #80ff8080 #20ffffff std
menu color pwdentry     30;47      #80ffffff #20ffffff std
menu color timeout_msg  37;40      #80ffffff #00000000 std
menu color timeout     1;37;40    #c0ffffff #00000000 std
menu color help         37;40      #c0ffffff #00000000 std
menu color msg07        37;40      #90ffffff #00000000 std

```

MENU MSGCOLOR *fg_filter bg_filter shadow*

Sets all the msgXX colors to a color scheme derived from the fg_filter and bg_filter values. Background color zero is always treated as transparent. The default corresponds to:

```
menu msgcolor #90ffffff #80ffffff std
```

This directive should come before any directive that customizes individual msgXX colors.

MENU WIDTH 80

MENU MARGIN 10

MENU PASSWORDMARGIN 3

MENU ROWS 12

MENU TABMSGROW 18

MENU CMDLINEROW 18
MENU ENDROW -1
MENU PASSWORDROW 11
MENU TIMEOUTROW 20
MENU HELPMMSGROW 22
MENU HELPMMSGENDROW -1
MENU HIDDENROW -2
MENU HSHIFT 0
MENU VSHIFT 0

These options control the layout of the menu on the screen.
 The values above are the defaults.

A negative value is relative to the calculated length of the screen (25 for text mode, 28 for VESA graphics mode.)

F1 *textfile* [*background*]

...

F12 *textfile* [*background*]

Displays full-screen help (also available at the command line.) The same control code sequences as in the command line interface are supported, although some are ignored.

Additionally, an optional second argument allows a different background image (see `MENU BACKGROUND` for supported formats) to be displayed.

MENU HELP *textfile* [*background*]

Creates a menu entry which, when selected, displays full-screen help in the same way as the F-key help.

The menu system honours the `TIMEOUT` command; if `TIMEOUT` is specified it will execute the `ONTIMEOUT` command if one exists, otherwise it will pick the default menu option. **WARNING:** the `TIMEOUT` action will bypass password protection even if one is set for the specified or default entry!

Normally, the user can press [Tab] to edit the menu entry, and [Esc] to return to the Syslinux command line. However, if the configuration file specifies `ALLOWOPTIONS 0`, these keys will be disabled, and if `MENU MASTER PASSWD` is set, they require the master password.

The simple menu system supports serial console, using the normal `SERIAL` directive. However, it can be quite slow over a slow serial link; you probably want to set your baudrate to 38400 or higher if possible. It requires a Linux/VT220/ANSI-compatible terminal on the other end.

USING AN ALTERNATE CONFIGURATION FILE

It is also possible to load a secondary configuration file, to get to another menu. To do that, invoke `menu.c32` with the name of the secondary configuration file.

```

LABEL othermenu
    MENU LABEL Another Menu
    KERNEL menu.c32
    APPEND othermenu.conf
  
```

If you specify more than one file, they will all be read, in the order specified. The dummy filename `~` (tilde) is replaced with the filename of the main configuration file.

```

# The file graphics.conf contains common color and layout commands for
# all menus.
LABEL othermenu
    MENU LABEL Another Menu
    KERNEL vesamenu.c32
    APPEND graphics.conf othermenu.conf

# Return to the main menu
LABEL mainmenu
    MENU LABEL Return to Main Menu
    KERNEL vesamenu.c32
    APPEND graphics.conf ~
  
```

See also the `MENU INCLUDE` directive above.

Retrieved from "<http://www.syslinux.org/wiki/index.php?title=Comboot/menu.c32&oldid=4402>"

Categories: [Release Documentation](#) | [Comboot](#) | [Menu](#) | [Modules](#)

- This page was last modified on 23 June 2015, at 21:39.