**PXESERVA configuration file language**

Serva 2.0 multi-OS PXE Install Menu system (pxeserva.0/vesamenu.c32) is based on SYSLINUX-PXELINUX ([www.syslinux.org](http://www.syslinux.org)). Its configuration file (\pxeserva.cfg\default) is automatically created and maintained by Serva BINL services, using the following configuration language commands.

NOTE 1: The configuration file language is case insensitive.

NOTE 2: Blank lines are ignored.

NOTE 3: PXESERVA will automatically load vesamenu.c32

**Commands:**

**# comment**

A comment line. The whitespace after the hash mark is mandatory.

**INCLUDE** *filename*

 Inserts the contents of another file at this point in the configuration file. Files can currently be nested up to 16 levels deep, but it is not guaranteed that more than 8 levels will be supported in the future.

**APPEND** *options...*

 Add one or more options to the kernel command line. These are added both for automatic and manual boots. The options are added at the very beginning of the kernel command line, usually permitting explicitly entered kernel options to override them.

**IPAPPEND** *flag\_val*

 The flag\_val is an OR of the following options:

1: indicates that an option of the following format should be generated and added to the kernel command line:

 ip=<client-ip>:<boot-server-ip>:<gw-ip>:<netmask>

... based on the input from the DHCP/BOOTP or PXE boot server.

THE USE OF THIS OPTION IS NOT RECOMMENDED. If you have to use it, it is probably an indication that your network configuration is broken. Using just "ip=dhcp" on the kernel command line is a preferable option, or, better yet, run dhcpcd/dhclient, from an initrd if necessary.

2: indicates that an option of the following format should be generated and added to the kernel command line:

 BOOTIF=<hardware-address-of-boot-interface>

... in dash-separated hexadecimal with leading hardware type

This allows an initrd program to determine from which interface the system booted.

**LABEL** *label*

 KERNEL image

 APPEND options...

 IPAPPEND flag\_val

 Indicates that if "label" is entered as the kernel to boot, PXESERVA should instead boot "image", and the specified APPEND and IPAPPEND options should be used instead of the ones specified in the global section of the file (before the first LABEL command.) The default for "image" is the same as "label", and if no APPEND is given the default is to use the global entry (if any).

Note: The "kernel" doesn't have to be a Linux kernel; it can be a boot sector or a COMBOOT file (see below.)

 The following commands are available after a LABEL statement:

 LINUX image - Linux kernel image (default)

 BOOT image - Bootstrap program (.bs, .bin)

 BSS image - BSS image (.bss)

 PXE image - PXE Network Bootstrap Program (.0)

 FDIMAGE image - Floppy disk image (.img)

 COMBOOT image - COMBOOT program (.com, .cbt)

 COM32 image - COM32 program (.c32)

 CONFIG image - New configuration file

 Using one of these keywords instead of KERNEL forces the file type, regardless of the filename.

 CONFIG means restart the boot loader using a different configuration file.

 **APPEND -**

 Append nothing. APPEND with a single hyphen as argument in a LABEL section can be used to override a global APPEND.

**IMPLICIT** *flag\_val*

 If flag\_val is 0, do not load a kernel image unless it has been explicitly named in a LABEL statement. The default is 1.

**ALLOWOPTIONS** *flag\_val*

 If flag\_val is 0, the user is not allowed to specify any on the kernel command line. The only options recognized are those specified in an APPEND statement. The default is 1.

**TIMEOUT** *timeout*

 Indicates how long to wait on init until booting automatically, in units of 1/10 s. The timeout is cancelled as soon as the user types anything on the keyboard. A timeout of zero will disable the timeout completely, this is also the default.

**TOTALTIMEOUT** *timeout*

 Indicates how long to wait until booting automatically, in of 1/10 s. This timeout is \*not\* cancelled by user input, and can thus be used to deal with serial port glitches or "the user walked away" type situations. A timeout of zero will disable the timeout completely, this is also the default.

 Both TIMEOUT and TOTALTIMEOUT can be used together, for

 example:

 # Wait 5 seconds unless the user types something, but

 # always boot after 15 minutes.

 TIMEOUT 50

 TOTALTIMEOUT 9000

**ONTIMEOUT** *kernel options...*

 Sets the command line invoked on a timeout. Normally this is the same thing as invoked by "DEFAULT". If this is specified, then "DEFAULT" is used only if the user presses <Enter> to boot.

**ONERROR** *kernel options...*

 If a kernel image is not found (either if it’s not existing, or because IMPLICIT is set), run the specified command. The faulty command line is appended to the specified options, so if the ONERROR directive reads as:

 ONERROR xyzzy plugh

 ... and the command line as entered by the user is:

 foo bar baz

 ... PXESERVA will execute the following as if entered by the

 user:

 xyzzy plugh foo bar baz

 should be the first directive in the configuration file.

**DISPLAY** *filename*

 Displays the indicated file on the screen at boot time. Please see the section below on DISPLAY files.

 NOTE: If the file is missing, this option is simply ignored.

**SAY** *message*

Prints the message on the screen.

**PROMPT** *flag\_val*

 If flag\_val is 0, display the boot: prompt only if the Shift or Alt key is pressed, or Caps Lock or Scroll lock is set (this is the default). If flag\_val is 1, always display the boot: prompt.

**NOESCAPE** *flag\_val*

 If flag\_val is set to 1, ignore the Shift/Alt/Caps Lock/Scroll Lock escapes. Use this (together with PROMPT 0) to force the default boot alternative.

**NOCOMPLETE** *flag\_val*

If flag\_val is set to 1, the Tab key does not display labels at the boot: prompt.

**F1** *filename1*

**F2** *filename2*

**...**

**F11** *filename11*

**F12** *filename12*

 Displays the indicated file on the screen when a function key is pressed. This can be used to implement pre-boot online help. Please see the section below on DISPLAY files.

There are a few menu additions to the configuration file, all starting with the keywords MENU or TEXT;

**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 CLEAR**

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

**MENU SHIFTKEY**

 Exit the menu system immediately unless either the Shift or Alt 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 an 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 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 this

 particular submenu. See also the DEFAULT directive below.

**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** *passwdHash*

 (Only valid after a LABEL statement.) Sets a password on this menu entry. "passwdHash" must be a password hash (encrypted password) made with one of the following algorithms:

 MD5 (Signature: $1$)

 SHA-1 (Signature: $4$)

 SHA-2-256 (Signature: $5$)

 SHA-2-512 (Signature: $6$)

 On Linux systems use the included ISOLINUX Perl scripts "sha1pass" or "md5pass" to encrypt passwords. MD5 passwords are compatible with most Unix password file utilities. Alternatively you can use Serva’s on-line MD5 and SHA1 hash calculator.

 If you are using passwords, you want to make sure you also use the settings "PROMPT 0", and either set "ALLOWOPTIONS 0" or use a master password (see below.)

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

**MENU MASTER PASSWD** *passwdHash*

 Sets a master password. This password can be used to password protect all the entries having a MENU PASSWD statement with an empty *passwdHash*.

 Entries not having a MENU PASSWD statement remain not password protected even if there’s a master password defined.

**MENU RESOLUTION** *height width*

 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 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 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 below.

**DEFAULT** *label*

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

 This directive is ignored unless the configuration file also contains a UI directive.

**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 PXESERVA 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 in text mode

 "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 #ffffffff 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 #ffffffff #00000000 std

 menu color sel 7;37;40 #e0000000 #20ff8000 all

 menu color hotsel 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 HELPMSGROW** *22*

**MENU HELPMSGENDROW** *-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.)