

WDSLINUX

From Syslinux Wiki

Creating a PXE Boot menu for deploying Linux with Windows Deployment Services (WDS)

Contents

- 1 Requirements
- 2 How To Setup Windows Deployment Services with PXELinux
- 3 Adding Linux images to boot/install
 - 3.1 Debian and Ubuntu
 - 3.2 Gentoo
 - 3.3 GParted
 - 3.4 F-Secure Antivirus Rescue CD
 - 3.5 Knoppix
 - 3.6 Citrix XenServer (Redhat/Centos)

Requirements

- PXELinux - part of syslinux PXELINUX - Download
- Windows Deployment Services already setup (setting up WDS is outside of the scope of this article)
 - NOTE: \$WDS-ROOT will reference the WDS root folder location in this case D:\RemoteInstall\

How To Setup Windows Deployment Services with PXELinux

- Download syslinux from <http://www.kernel.org/pub/linux/utils/boot/syslinux/>
- Extract core\pxelinux.0 com32\menu\vesamenu.c32 and com32\modules\chain.c32 from the syslinux download and put it on your WDS server in \$WDS-ROOT\Boot\x86\ and \$WDS-ROOT\Boot\x64\ (substitute WDS-ROOT for where your WDS root folder is)
 - Depending on the version, you may also need additional modules such as ldlinux.c32, libcom32.c32 and libmenu.c32. See also ELF missing dependencies
- In the \$WDS-ROOT\Boot\\${ARCH} folders Rename pxelinux.0 to pxelinux.com
- Create a folder named pxelinux.cfg (in the \$WDS-ROOT\Boot\x86\ and \$WDS-ROOT\Boot\x64\ folder)
- In the pxelinux.cfg folder create a text file named default and add the following to it (you can substitute MyMenuBackgroundPicture640x480.jpg for any image you want as your menu background)

```

DEFAULT      vesamenu.c32
PROMPT       0
NOESCAPE     0
ALLOWOPTIONS 0
# Timeout in units of 1/10 s
TIMEOUT 300
MENU MARGIN 10
MENU ROWS 16
MENU TABMSGROW 21
MENU TIMEOUTROW 26
MENU COLOR BORDER 30;44          #20ffffff #00000000 none
MENU COLOR SCROLLBAR 30;44       #20ffffff #00000000 none
MENU COLOR TITLE 0               #ffffffff #00000000 none
MENU COLOR SEL 30;47             #40000000 #20ffffff
MENU BACKGROUND MyMenuBackgroundPicture640x480.jpg
MENU TITLE PXE Boot Menu
# ---
LABEL wds
MENU LABEL Windows Deployment Services
KERNEL pxeboot.0
# ---
LABEL Abort
MENU LABEL AbortPXE
Kernel abortpxe.0
# ---
LABEL local
MENU DEFAULT
MENU LABEL Boot from Harddisk
LOCALBOOT 0
Type 0x80

```

- Make a copy of pxeboot.n12 and name it pxeboot.0

- make a copy from abortpxe.com and rename it to abortpxe.0
- Create a folder named Linux (in the \$WDS-ROOT\Boot\x86\ and \$WDS-ROOT\Boot\x64\ folder)
 - Open the Windows Deployment Services Console,
 - Right Click on your Server and Select Properties,
 - From the Boot Tab change the default boot program for your architecute (x86 and x64 as well) to Boot\x86\pxelinux.com and Boot\x64\pxelinux.com respectively
 - **NOTE:** In the WDS included in Windows Server 2008 R2 the UI has changed and you have to use the command line to set the default boot program.
 - Thus to change the boot program to pxelinux.com, the wdsutil command line tool has to be used: (do this also for x64 if you have x64 clients also)
 - wdsutil /set-server /bootprogram:boot\x86\pxelinux.com /architecture:x86
 - wdsutil /set-server /N12bootprogram:boot\x86\pxelinux.com /architecture:x86
- Now when you pxe boot a machine you will get your custom boot menu

Adding Linux images to boot/install

Debian and Ubuntu

- Download the netboot files (initrd.gz and linux) from a debian/ubuntu mirror
 - Debian x86 files
 - <http://ftp.nl.debian.org/debian/dists/stable/main/installer-i386/current/images/netboot/debian-installer/i386/>
 - Debian amd64 files
 - <http://ftp.nl.debian.org/debian/dists/stable/main/installer-amd64/current/images/netboot/debian-installer/amd64/>
 - Ubuntu x86 files
 - <http://ubuntu.virginmedia.com/archive/dists/karmic/main/installer-i386/current/images/netboot/ubuntu-installer/i386/>
 - Ubuntu amd64 files
 - <http://ubuntu.virginmedia.com/archive/dists/karmic/main/installer-amd64/current/images/netboot/ubuntu-installer/amd64/>
- Create a folder on your WDS server under \$WDS-ROOT\Boot\arch\Linux called Debian (or Ubuntu)
- Copy the downloaded netboot files initrd.gz and linux to the new folder
- Create a menu entry at the end of the \$WDS-ROOT\Boot\arch\pxelinux.cfg\default file (boot menu items for debian and ubuntu can be found in the download site under boot-screens in the text.cfg file)

```

label debian5-0expert
menu label ^Debian 5-0 Expert install:
kernel /Linux/Debian5.0/linux
append priority=low vga=normal initrd=/Linux/Debian5.0/initrd.gz --
#--
label debian5-0rescue
menu label ^Debian 5-0 Rescue mode
kernel /Linux/Debian5.0/linux
append vga=normal initrd=/Linux/Debian5.0/initrd.gz rescue/enable=true -- quiet
#--
label debian5-0auto
menu label ^Debian 5-0 Automated install
kernel /Linux/Debian5.0/linux
append auto=true priority=critical vga=normal initrd=/Linux/Debian5.0/initrd.gz -- quiet
#--
label debian5-0amd64expert
menu label ^Debian 5-0 amd64 Expert install
kernel /Linux/Debian5.0/linux-amd64
append priority=low vga=normal initrd=/Linux/Debian5.0/initrd-amd64.gz --
#--
label debian5-0amd64rescue
menu label ^Debian 5-0 amd64 Rescue mode
kernel /Linux/Debian5.0/linux-amd64
append vga=normal initrd=/Linux/Debian5.0/initrd-amd64.gz rescue/enable=true -- quiet
#--
label debian5-0amd64auto
menu label ^Debian 5-0 amd64 Automated install
kernel /Linux/Debian5.0/linux-amd64
append auto=true priority=critical vga=normal initrd=/Linux/Debian5.0/initrd-amd64.gz -- quiet
#--
label ubuntu-expert
menu label ^Ubuntu Expert install
kernel /linux/ubuntu/linux
append priority=low vga=normal initrd=/linux/ubuntu/initrd.gz --
#--
label ubuntu-cli-expert
menu label Ubuntu Command-line expert install
kernel /linux/ubuntu/linux
append tasks=standard pkgset/language-pack-patterns= pkgset/install-language-support=false priority=low vga=normal initrd=/linux/ubuntu/initrd.gz --
#--
label ubuntu-rescue
menu label Ubuntu Rescue mode
kernel /linux/ubuntu/linux
append vga=normal initrd=/linux/ubuntu/initrd.gz rescue/enable=true -- quiet
#--
label ubuntu-expert-amd64
menu label ^Ubuntu-amd64 Expert install

```

```
kernel /linux/ubuntu/linux-amd64
append priority=low vga=normal initrd=/linux/ubuntu/initrd-amd64.gz --
#--
label ubuntu-amd64-cli-expert
menu label Ubuntu-amd64 Command-line expert install
kernel /linux/ubuntu/linux-amd64
append tasks=standard pkgset/language-pack-patterns= pkgset/install-language-support=false priority=low vga=normal initrd=/linux/ubuntu/initrd-amd64.gz rescue
#--
label ubuntu-amd64-rescue
menu label Ubuntu-amd64 Rescue mode
kernel /linux/ubuntu/linux-amd64
append vga=normal initrd=/linux/ubuntu/initrd-amd64.gz rescue/enable=true -- quiet
```

Gentoo

- Download the latest gentoo minimal install CD from a mirror
- Extract the files isolinux/gentoo isolinux/gentoo.igz image.squashfs from the iso to a temporary folder
- Make a folder on the WDS server under \$WDS-ROOT\Boot\arch\Linux called gentoo
- Copy the extracted gentoo file to the \$WDS-ROOT\Boot\arch\Linux\gentoo folder on the WDS server
- Copy the extracted gentoo.igz and image.squashfs files to a temporary location on a linux machine (i.e. /root/tmp)
- In the /root/tmp location create a folder called gentoo and change into this directory
 - mkdir /root/tmp/gentoo
 - cd /root/tmp/gentoo
- Type the following command and press enter (this extracts all the files from the gentoo.igz file)
 - gunzip -c ../gentoo.igz | cpio -idv
- Make the following folders (in the extracted folder)
 - mkdir mnt
 - mkdir mnt/cdrom
- Copy the image.squashfs to the new created mnt/cdrom folder
 - cp ../image.squashfs mnt/cdrom
- Make a backup copy of the init file
 - cp init init.orig
- Edit the init file with the following modifications

```
Change the line:
-- if [ "${REAL_ROOT}" != "/dev/nfs" ] && [ "${LOOPTYPE}" != "sgimips" ]
To:
++ if [ "${REAL_ROOT}" != "/dev/nfs" ] && [ "${LOOPTYPE}" != "sgimips" ] && [ "${REAL_ROOT}" != "/" ]
Change the line (under # Check for a block device or /dev/nfs):
-- elif [ -b "${REAL_ROOT}" ] || [ "${REAL_ROOT}" = "/dev/nfs" ]
To:
++ elif [ -b "${REAL_ROOT}" ] || [ "${REAL_ROOT}" = "/dev/nfs" ] || [ "${REAL_ROOT}" = "/" ]
Remove the line:
-- cache_cd_contents
Change the line:
-- mount -t squashfs -o loop,ro ${NEW_ROOT}/mnt/cdrom/${LOOPEXT}${LOOP} ${NEW_ROOT}/mnt/lived
To:
++ mount -t squashfs -o loop,ro /mnt/cdrom/${LOOPEXT}${LOOP} ${NEW_ROOT}/mnt/lived
```

- Compress everything back into a file again by this typing the following and press enter
 - find . -print | cpio -o -H newc | gzip -9 -c -> ../gentoo.igz.new
- Copy the new gentoo.igz.new to the \$WDS-ROOT\Boot\arch\Linux\gentoo folder on the WDS server and name it gentoo.igz
- Create a menu entry at the end of the \$WDS-ROOT\Boot\arch\pxelinux.cfg/default file

```
LABEL gentoo2008-x86
MENU LABEL Deploy Gentoo 32 bit
KERNEL /linux/gentoo/gentoo
append root=/dev/ram0 init=/linuxrc dokeymap looptype=squashfs loop=/image.squashfs cdroot initrd=/linux/gentoo/gentoo.igz real_root=/ vga=
#--
LABEL gentoo2008-x64
MENU LABEL Deploy Gentoo amd64
KERNEL /linux/gentoo/gentoo-amd64
append root=/dev/ram0 init=/linuxrc dokeymap looptype=squashfs loop=/image.squashfs cdroot initrd=/linux/gentoo/gentoo-amd64.igz real_root=
```

GParted

- Download GParted live zip file (You have to use 0.3.7-2 or later, network drivers are only included after that), and unzip all the files in a temp dir
- Copy necessary boot files (vmlinuz1 and initrd1.img) to a new folder on your WDS server (\$WDS-ROOT\Boot\arch\Linux\gparted)
- Copy /tmp/gparted/live/filesystem.squashfs to your http web server. (If on IIS create a new virtual directory and set the mime type for .* extension to text/plain)
- Edit your PXE config file /Boot/arch/pxelinux.cfg/default, and append the following:

```
label GParted Live
```

```

MENU LABEL GParted Live
kernel \Linux\gparted\mlinuz1
append initrd=\Linux\gparted\initrd1.img boot=live union=aufs noswap noprompt vga=788 fetch=http://$webserverIP/gparted/filesystem.squashfs
###//NOTE// Replace $webserverIP with your IP address of http server.

```

As of GParted-live version 0.3.8-3, fetch=tftp is supported. Just copy the filesystem.squashfs file into the same folder as mlinuz1 and initrd1.img (in the above example it would be \Linux\gparted. The above append can then be replaced with:

```

append initrd=\Linux\gparted\initrd1.img boot=live union=aufs noswap noprompt vga=788 fetch=tftp://$tftpserverIP/linux/gparted/filesystem.sq
# //NOTE// Replace $tftpserverIP with the IP address of your tftp server.

```

F-Secure Antivirus Rescue CD

- Download the latest f-secure Rescue CD from http://www.f-secure.com/en_EMEA/security/tools/rescue-cd/
- Extract the kernel from the ISO image (boot/isolinux/fsecure/linux) to a new folder on your WDS server (\$WDS-ROOT\Boot\arch\Linux\fsecure)
- Extract the boot/isolinux/fsecure/minirt.gz file and whole KNOPPIX folder from the ISO
- Copy the extracted minirt.gz file and KNOPPIX folder to a temporary location on a linux machine (i.e. /root/tmp)
- In the /root/tmp location create a folder called fsecure and change into this directory
 - mkdir /root/tmp/fsecure
 - cd /root/tmp/fsecure
- Type the following command and press enter (this extracts all the files from the minirt.gz file)
 - gunzip -c ../minirt.gz | cpio -idv
- Copy the KNOPPIX folder extracted from the ISO to the mnt-system folder
 - cp -r ../KNOPPIX mnt-system
- Make a backup copy of the init file
 - cp init init.orig
- Edit the init file with the following modifications

```

find the findknoppix devices section and delete everything from it

findknoppix(){
-- local dev
-- for dev in "$@"; do
--   [ -b "$dev" ] || continue
--   echo -n -e "\r${CRE}${BLUE}${SEARCHINGFOR} $DISTRO in: ${MAGENTA}$dev${NORMAL}  "
--   trymount "$dev" /mnt-system >/dev/null 2>&1 || continue
--   if [ -r /mnt-system/"$knoppix_dir"/KNOPPIX ] ; then
--     # We must have also fsecure structure present else we might try to mount something wrong.
--     if [ -d /mnt-system/fsecure/rescuecd ] || [ -d /mnt-system/boot/isolinux/fsecure ] ; then
--       echo -e "\r${CRE}${GREEN}$DISTRO ${FOUNDAT}: ${MAGENTA}$dev${NORMAL}  "
--       return 0
--     fi
--   fi
--   umount /mnt-system
-- done
-- return 1
}

Add return 0 to the findknoppix devices section

++ return 0
}

```

- Compress everything back into a file again by this typing the following and press enter
 - find . -print | cpio -o -H newc | gzip -9 -c - > ../minirt.gz.new
- Copy the new minirt.gz.new to the \$WDS-ROOT\Boot\arch\Linux\fsecure folder on the WDS server and name it minirt.gz
- Create a menu entry at the end of the \$WDS-ROOT\Boot\arch\pxelinux.cfg\default file

```

LABEL fsecurerescuecd
MENU LABEL F-Secure AV Rescue CD
KERNEL linux/fsecure/linux
APPEND ramdisk_size=100000 lang=en vt.default_utf8=0 apm=power-off vga=0x311 initrd=Linux/fsecure/minirt.gz nomce quiet loglevel=1 nolapio

```

Knoppix

See http://pxe.dev.aboveaverageurl.com/index.php/PXE_Booting/Knoppix for details

Citrix XenServer (Redhat/Centos)

- Download the latest XenServer ISO's
- Make a folder on the WDS server under \$WDS-ROOT\Boot\arch\Linux called xenserver
- Extract the following files from the first XenServer ISO

```
/install.img  
/boot/xen.gz  
/boot/vmlinuz  
/boot/isolinux/mboot.c32  
the whole /packages.main folder
```

- Extract the following files from the second !XenServer ISO
 - the whole /packages.linux folder
- Copy the extracted files to the \$WDS-ROOT\Boot\arch\Linux\xenserver folder on the WDS server
- Create a virtual directory in IIS on the WDS server called Xen\$version (i.e. xen5-5) with read access pointing to the \$WDS-ROOT\Boot\arch\Linux\xenserver folder
- On the new IIS virtual directory add the mime type for extension .* text/plain
- Edit your PXE config file /Boot/arch/pxelinux.cfg/default, and append the following:

```
label xenserver5-5beta  
menu label Citrix ^Xen Server 5.5 install  
kernel /Linux/xenserver/mboot.c32  
APPEND /Linux/xenserver/xen.gz dom0_mem=752M com1=115200,8n1 console=com1,tty --- /Linux/xenserver/vmlinuz console=ttyS0,115200n8 console=tty0
```

- when installing xen select install from http/ftp and type in the location of your wds/web server (IIS webserver as configured above, i.e. <http://172.16.0.1/xen5-5>)

Retrieved from "<http://www.syslinux.org/wiki/index.php?title=WDSLINUX&oldid=3993>"

Categories: PXELINUX | WDS

- This page was last modified on 29 July 2013, at 23:45.
- This page has been accessed 2,561 times.