

ESX on a Stick

Acknowledgement:

I would like to personally thank Paul Lalonde from the VMware Community Forums who first “cracked” this setup. I took Paul’s original post and edited it and added some extra tips and tricks along the way for this book. Paul has also fixed Anaconda bugs in the hdstg2.img file which allows the ESX source ISO to work from a USB stick. Paul very kindly allowed me to use his work in this chapter. If you are interested this is the path to the original thread

<http://www.vmware.com/community/thread.jspa?threadID=75792>

One alternative to using the kickstart appliance shown previously is to transfer the ESX CD to a USB memory stick. This takes surprisingly little time to do and save you having to carry CDs with you. Additionally, having the ESX CD on a memory stick could be useful for systems that do not have a CD-ROM. Instead USB memory sticks are treated by the ESX installer as a hard-drive.

USB sticks come in many standard formats – 64MB, 128MB, 512MB and upwards. However, they are deeply proprietary and many of them represent these different storage capacities by a non-standard sequence of cylinders, heads and sectors that vary from one vendor to another. My first experiences of setting this up were not good until I read help files from various utilities used in the setup routine.

Fortunately, I discovered a utility called “mkdiskimage” that would create a partition on a USB stick which would have correct geometry values. My two experiments were with 1GB an “Imation Mini Flash Drive” and “SanDisk Cruiser Smart U3”. I found both needed mkdiskimage to work properly. The mkdiskimage utility is part of the syslinux project. Syslinux is a open-source boot loader, and we will use it’s tools to make the USB device bootable.

If like me you have problems with making the USB stick bootable – you will need to transfer the syslinux files to Linux or the Service Console and user mkdiskimage. This because the mkdiskimage is perl script which will run natively on any Linux platform whereas Windows does not have the built-in APIs for Perl script. If your USB stick works out of the box you may not need to use mkdiskimage, so you may wish skip called “Basic USB Disk Configuration”. Lastly, using the appropriate tools you could easily carry out this task from Windows, as long as you didn’t need mkdiskimage.

Basic USB Disk Configuration (Optional)

1. **From your Windows PC download the syslinux tools** from:

<http://www.kernel.org/pub/linux/utils/boot/syslinux/syslinux-3.36.zip>

2. **Extract the ZIP file** to you Windows PC
3. **Transfer this file to Linux** or your **Service Console**
4. **Make a directory** and **extract the syslinux files** to that location with

```
mkdir /root/syslinux
unzip syslinux-3.36.zip -d /root/syslinux
```

5. **Plug-in your USB Stick** into Linux or the Service Console
-

Note:

When you do this the system should recognize the device – if you are at the physical console you would see the device being recognized – and would tell you what the `/dev/sdn` name is of the USB stick. If you miss this message then you can use the command

dmesg | grep sd

This will list all the SCSI devices on the system like so and I can see the USB device has been given the identity of `/dev/sdl`

SCSI device sda: 2072576 512-byte hdwr sectors (**1012 MB**)

sda: Write Protect is off

sda: sdl4

sda : READ CAPACITY failed.

sda : status = 1, message = 00, host = 0, driver = 08

Info fld=0xa00 (nonstd), Current sd00:00: sense key Not Ready

sda: block size assumed to be 512 bytes, disk size 1GB.

sda: test WP failed, assume Write Enabled

sda: I/O error: dev 08:b0, sector 0

SCSI device sdl: 2072576 512-byte hdwr sectors (1012 MB)

sda: Write Protect is off

sda: sda4

6. Next we will use Linux command **fdisk** to delete any existing partition table, and **find out the number of cylinders, heads and sectors** are on the disk with

fdisk /dev/sdn

Press [**P**] to print the partition table and make a note of the number of cylinders heads and sectors

Note:

Below is a print out from an "Imation Mini Flash Drive"

Disk /dev/sdl: 1061 MB, 1061158912 bytes

33 heads, 62 sectors/track, 1012 cylinders

Units = cylinders of 2046 * 512 = 1047552 bytes

7. Press [**d**] to delete any existing partition
8. Press [**w**] to write these changes to the USB Stick
9. Next transfer the `mkdiskimage` to the USB device by

cd /root/syslinux

perl mkdiskimage /dev/sdn -4 1012 33 62

Note:

This took about 15-20 minutes on my USB stick. The switch `-4` indicates an image which is suitable for a zipdisks. The following numbers represent cylinders, heads and sectors. After this command completes you should be able to run `fdisk /dev/sdn` again to see a partition FAT16 has been created. In my case this reported:

/dev/sda4

Transfer and Edit Files

1. **Insert the USB stick** into your Windows PC
2. Using Windows Explorer or Disk Management, **Format the USB stick with FAT**
3. **Copy the ESX 3 ISO** to the root of USB stick
4. **Within the ESX CD open the /images/boot.iso file and copy all the files in the \isolinux directory to the root of the USB Stick**
5. **Delete the files** called **updatecd.cfg** and **isolinux.bin**
6. Rename **isolinux.cfg** to **syslinux.cfg**
7. **Save boot.msg and Exit Wordpad**
8. **Open syslinux.cfg with Wordpad** (do not use notepad!) and modify

default esx to be **default text**

9. To each of the append lines add to the end the value **usb**:

Note:

Below is my syslinux.cfg file:

```
default esx
prompt 1
timeout 600
display boot.msg
F1 boot.msg
F7 snake.msg
label debug
    kernel vmlinuz
    append initrd=initrd.img noapic nomediacheck debug usb
label esx
    kernel vmlinuz
    append initrd=initrd.img usb
label text
    kernel vmlinuz append initrd=initrd.img text usb
label expert
    kernel vmlinuz
    append expert initrd=initrd.img usb
label ks
    kernel vmlinuz
    append ks initrd=initrd.img usb
label lowres
    kernel vmlinuz
    append initrd=initrd.img lowres usb
```

Note:

You can edit these files at anytime on the USB stick without having to re-run the next step of the process which transfers the syslinux boot-loader to the USB Stick itself.

Transfer the Boot Loader to the USB Stick

1. **Open a command prompt** to where you **extracted the syslinux files** on the Windows PC and **move into the Win32 directory**
2. Type command:

syslinux -s <driveletter>:

Note:

Where <driveletter> is the drive letter used by the USB stick in Windows Explorer

Using USB on a Stick

TIP:

I would recommend disconnecting the SAN cables before you begin the installation (unless your configuring a boot from SAN setup). This makes identifying the USB stick much easier

1. **Insert the USB stick into your ESX server**
2. **Power on the server**
3. Press [**Enter**] at the splash screen

Note:

Notice how the splash screen reflects our changes to the boot.msg –and that pressing [ENTER] runs a text mode installation. Notice how various drivers load at this point, including USB device drivers

4. **Select your language** from the list
5. **Select your keyboard** type from the list
6. Choose **Hard Drive** as the media type
7. **Scroll down the list to find the USB drive**

Note:

You might find the /dev/sdnn values have changed here from the values you used to build the USB stick. Once you have select the drive and pressed OK, the anaconda install searches for the ISO image

8. **Choose to skip the CD-ROM test**

Note:

At this point you should find that you are in the main install utility for ESX server

Customizing ESX on a Stick

As you can see there are number of prompts if you do a manual setup which can be irritating, and of course as with the UDA you might wish to have kickstart scripts held on the USB key designed for scripted installations. This is very easy to do – all you need is a text editor. Samples of these files are available to download from the vi3book.com website.

First thing we can do is modify boot.msg file which prints friendly information underneath the VMware “splash” log.

1. **With Wordpad (not notepad!) open boot.msg from the USB Stick**
2. **Edit the boot.msg to give friendly information to the operator.**

Below is a sample of my boot.msg file:

□ splash.lss

- To install or upgrade □ VMWare ESX Server□ in text mode, press □ f<ENTER>□

- To install esx1.vi3book.com, type: esx01 and press the

□0f<ENTER>□07

- To install esx1.vi3book.com, type: esx01 and press the

□0f<ENTER>□07

- To install esx1.vi3book.com, type: esx01 and press the

□0f<ENTER>□07

- Note: Graphical Mode is not supported with ESX-on-a-Stick

3. **Next with Wordpad** (not notepad!) **open syslinux.cfg**

4. Within this file we will change:

The default boot option when the operator presses [ENTER]

Create boot options for esx01, esx02 and esx03

Add KS commands to point to kickstart files held in /kickstart directory on the USB key

Below is a sample syslinux.cfg file, and I have highlighted in bold the main changes I made

```
default text
prompt 1
timeout 600
display boot.msg
F1 boot.msg
F7 snake.msg
label debug
    kernel vmlinuz
    append initrd=initrd.img noapic nomediacheck debug usb
label esx
    kernel vmlinuz
    append initrd=initrd.img usb
label text
    kernel vmlinuz
    append ks initrd=initrd.img text usb ks=hd:sda4:/kickstart/default.cfg
label expert
    kernel vmlinuz
    append expert initrd=initrd.img usb
label ks
    kernel vmlinuz
    append ks initrd=initrd.img usb
label lowres
    kernel vmlinuz
    append initrd=initrd.img lowres usb
label esx01
    kernel vmlinuz
    append ks initrd=initrd.img usb ks=hd:sda4:/kickstart/esx01.cfg
label esx02
    kernel vmlinuz
    append ks initrd=initrd.img usb ks=hd:sda4:/kickstart/esx02.cfg
label esx03
    kernel vmlinuz
    append ks initrd=initrd.img usb ks=hd:sda4:/kickstart/esx03.cfg
```

Note:

The change to default text, makes a text mode installation the default – rather than boot load option of “esx” which is graphically based. Append ks allows us to append kickstart commands the main one being an instruction of where to find the kickstart files using ks=hd:sda4:/kickstart/kickstartfile.cfg

GOTCHA:

The references to sda4 are very specific to my hardware. This is how my Proliant DL 385 with a Smart Raid Array 6i controller sees my USB stick – as device /dev/sda4. This changes when I have the SAN connected to /dev/sdj. It is also different on my older Dell hardware which see the USB stick as /dev/sdc. One of those Dells has external box of disks connected with a SCSI adapter – it sees the USB stick as /dev/sdl

1. **On the USB stick create a kickstart directory**
2. **Copy the KS files you created in the UDA** part of this chapter
3. Open each on and modify the following lines

Installation Method

harddrive --partition=sda4 --dir= /

Note:

Next we will create the default.cfg kickstart script used if anyone tries a manual text installation

4. **Copy one of your existing kickstart files, and remove the vast majority of the information leaving behind** the following information

Auto-Generated Scripted Install Configuration file.

This file is used for VMware ESX Server Scripted Install Deployment

Installation Method

harddrive --partition=sda4 --dir= /

%packages

@base

%post
