Installing and Running Windows XP or Vista as a Xen HVM domainU Guest

From Virtuatopia

Purchase and download the full PDF and ePub versions of this Xen eBook for only $8.99

In this chapter we will work through steps involved in installing and running Windows XP and Windows Vista as virtualized Xen HVM guests.

Contents

Pre-Requisites for Installing a Windows Xen Guest

This chapter assumes that the Xen host (domain0) system is running a Xen enabled kernel and that the standard Xen tools (specifically xm) are also installed. In addition, full virtualization requires that the host system hardware include either Intel-VT or AMD-V virtualization support. In order to verify this run the following commands on the host system:

For Intel CPUs:

```
grep vmx /proc/cpuinfo
```

For AMD CPUSs:

```
grep svm /proc/cpuinfo
```

If neither of the above command produce output then it is likely these features are unavailable and that your hardware is unable to run Xen HVM guests. Whilst you can still run modified operating systems (such as Linux) as paravirtualized guests you will not be able to virtualize Windows using Xen. It is also possible that the host CPU supports one of these features, but that the functionality is disabled by default in the system BIOS. To verify this, access the host system's BIOS configuration menu during the boot process and look for an option related to virtualization support. If such an option exists and is disabled, enable it, boot the system and repeat the above commands.

Preparing to Install Windows

In order to install Windows as a Xen HVM guest the first requirement is disk space. The space assigned to a Windows guest may be either a physical disk drive attached to the host system or a disk image. To create a disk image simply use the *dd* command. For example, the following command line created a 6Gb disk image file called xenwin.img:

```
dd if=/dev/zero of=xenwin.img bs=1024k seek=6144 count=0
```

Preparing the Windows XP / Windows Vista for Xen HVM Installation

Installation will require a Windows XP or Vista installation DVD. Installation may be performed directly from the DVD drive, or the DVD image may be extracted from the DVD to an ISO image residing on the filesystem of the host. To extract the ISO image from the DVD insert it into a DVD drive, unmount it if it is auto-mounted and execute the following command (substituting */dev/hdb* for the matching device on your chosen Linux distribution):

```
dd if=/dev/hdb of=Windows.iso
```

Once the copy is completed the DVD can be removed from the drive and stored safely, and the *Windows.iso* file used to perform the installation.

Configuring the Xen Windows Guest Configuration File

Once the disk or disk image is available the next step is to create a suitable Xen HVM configuration file. The following lines are standard and should be used unchanged for all HVM configurations:

```
import os, re
arch = os.uname()[4]
if re.search('64', arch):
    arch_libdir = 'lib64'
else:
    arch_libdir = 'lib'

kernel = "/usr/lib/xen/boot/hvmloader"

builder='hvm'
```

Next the amount of memory to be assigned to the Windows XP or Windows Vista guest needs to be declared, in this case 512Mb:

```
memory = 512
```

Next, the amount of shadow memory may be defined. This should be equal to 2KB per MB of domain memory, plus a few MB per vcpu. In general 8Mb is sufficient:

```
shadow_memory = 8
```

The domain name and networking choices now need to be made. If you wish to bridge to the networking device on the host system use the following *vif* setting:

```
name = "xenhvm"

vif = [ 'type=ioemu, bridge=xenbr0' ]
```

Now, the most important part of the configuration is to specify the disk and CD/DVD settings. If a physical disk and CD/DVD drive are to be used this setting will read something like:

```
disk = [ 'phy:/dev/sdb,hda,w', 'phy:/dev/hdb,hdc:cdrom,r' ]
```

The above line maps physical disk drive */dev/sdb* on the host to */dev/hda* on the guest and */dev/hdc* on the host as the cdrom device on the guest (note that the CD/DVD device name on your host may be different depending on your Linux distribution).

Alternatively, if a disk image file is to be used as the disk drive the line might instead read:

```
disk = [ 'file:/home/xen/xenwin.img,hda,w', 'phy:/dev/hdb,hdc:cdrom,r' ]
```

Alternatively, if you have opted to use an ISO image of the Windows XP or Windows Vista installation DVD this can be referenced in place of the CD/DVD device:

```
disk = [ 'file:/home/xen/xenwin.img,hda,w', 'file:/home/xen/Windows.iso,hdc:cdrom,r' ]
```

Next the boot order of the guest needs to be defined. Clearly the CDROM device needs to be first in the boot order, at least until Windows is installed on the disk drive:

```
# boot on floppy (a), hard disk (c) or CD-ROM (d)
# default: hard disk, cd-rom, floppy
boot="dc"
```

Some configuration settings now need to be made to enable a graphical console for our Windows guest operating system. Xen provides a choice of VNC or SDL for supporting a graphical console when the guest is running. For example the following settings select VNC:

```
vnc = 1
sdl = 0
```

whilst the following selects SDL:

```
vnc = 0
sdl = 1
```

Both SDL and VNC work very well in terms of displaying Windows in a graphical console, although VNC has some distinct advantages over SDL. Firstly, VNC provides greater flexibility than SDL in terms of remote access to the domainU graphical console. With VNC it is possible to connect to the graphical console from other systems, either on the local network or even over the internet. Secondly, when you close a VNC viewer window the guest domain continues to run allowing you to simply reconnect to carry on where you left off. Closing an SDL window, however, immediately terminates the guest domainU system resulting in possible data loss.

By default Xen does not automatically start the VNC console when the domainU guest starts up. In order to have the graphical console spawned automatically, change the *vncconsole =* value to 1:

```
vncconsole=1
```

After selecting the graphical display options a number of other settings are required and/or recommended:

```
acpi = 1
apic = 1
device_model = '/usr/' + arch_libdir + '/xen/bin/qemu-dm'
stdvga=0
serial='pty'
```

One final setting is needed to avoid problems with the mouse pointer:

```
usbdevice='tablet'
```

Bringing all these settings together gives us a minimal Xen HVM configuration file for a Windows guest which looks similar to the following:

```
import os, re
arch = os.uname()[4]
if re.search('64', arch):
    arch_libdir = 'lib64'
else:
    arch_libdir = 'lib'

kernel = "/usr/lib/xen/boot/hvmloader"

builder='hvm'

memory = 512
shadow_memory = 8
name = "xenhvm"
vif = [ 'type=ioemu, bridge=xenbr0' ]
disk = [ 'file:/home/nas/xen/xenguest.img,hda,w', 'phy:/dev/hdb,hdc:cdrom,r' ]
device_model = '/usr/' + arch_libdir + '/xen/bin/qemu-dm'
# boot on floppy (a), hard disk (c) or CD-ROM (d)
# default: hard disk, cd-rom, floppy
boot="dc"

sdl=0
vnc=1
vncconsole=1
vncpasswd=''
stdvga=0
serial='pty'
usbdevice='tablet'
```

A number of other configuration options are available to modify the guest environment but the above settings are the minimal settings recommended to run a Windows guest with Xen. Once the configuration file is complete save it as xenwin.cfg.

Starting the Xen Windows HVM Guest

With the configuration complete the next step is to start up the Xen guest environment:

```
xm create xenwin.cfg
Using config file "./xenwin.cfg".
Started domain xenhvm
```

After a few seconds the VNC or SDL console (depending on the configuration settings) will appear and the Windows installation process will begin. The following figure shows the installation running in a VNC window :
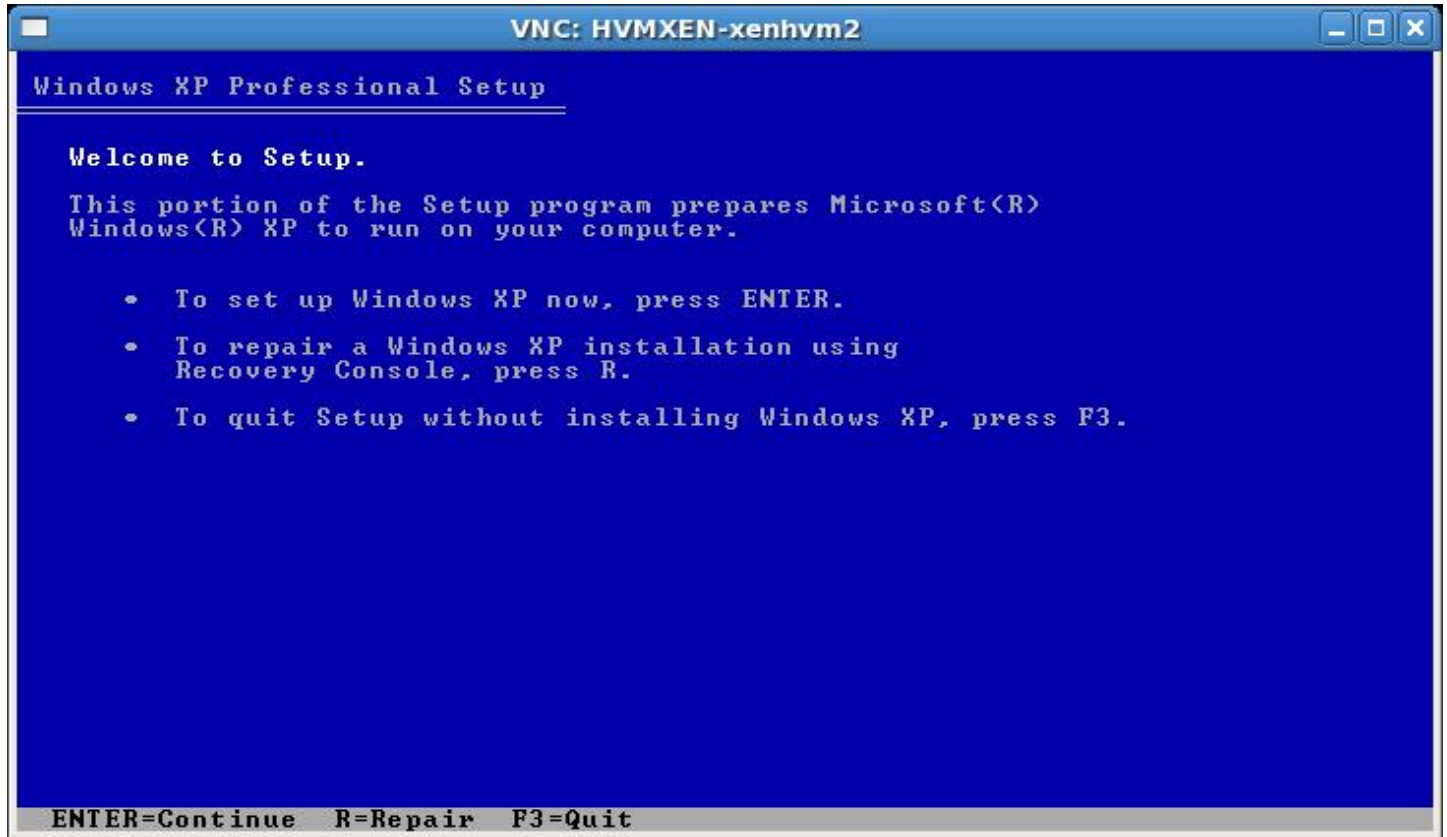
Purchase and download the full PDF and ePub versions of this Xen eBook for only $8.99

Retrieved from "http://www.virtuatopia.com/index.php?
title=Installing_and_Running_Windows_XP_or_Vista_as_a_Xen_HVM_domainU_Guest&oldid=2086"