Xen for UCS

Topic:	Introduction, installation and configuration of Xen virtualisation with UCS	
Date:	November 9, 2009	
Pages:	10	
Version number:	4222	
Autoren:	Univention GmbH	feedback@univention.de

Contents

1 Introduction

Xen is a virtualisation solution originally developed at Cambridge University, which facilitates the operation of additional guest instances within an operating system instance.

Processes with different privilege levels, so called ***rings***, can be run on x86 CPUs. Direct access to hardware, e.g., for drivers, is only possible in Linux in Ring 0, which is represented by the kernel mode in the operating system. Unprivileged processes, so-called user space, are run in a separate ring.

Xen implements a ***Virtual Machine Monitor***, referred to in Xen terminology as a ***Hypervisor***. This allows a Linux kernel operated as a guest instance to be run in Ring 1 whilst the hypervisor starts in Ring 0. A modified operating system is required for the operation of the hypervisor. Two different approaches are available for starting a guest instance:

- Operating with a modified guest operating system allows high performance, as hardware accesses with low overheads are forwarded to the host instance. This mode can be operated on most 486 CPUs and above and is referred to as a paravirtualization.
- When the CPU of the host system supports virtualisation there is also the possibility of running unmodified operating systems such as Microsoft Windows. Intel calls this technology ***Intel VT***, also known as Vanderpool, AMD calls it ***AMD-V***, also known as Pacifica.

  While CPU accesses are automatically forwarded by these hardware extensions, the remaining hardware components (graphics cards, network cards, sound cards) must be emulated. This is performed by components of the emulator ***qemu***.

Different hardware domains are operated in one Xen system. There are two different types of domain:

- The ***host system***, also referred to as ***Domain 0*** or ***Dom0***, is the privileged domain. There is only one of these on the system and it is responsible for assigning resources for the virtualised systems amongst other duties.
- The non-privileged ***guest systems*** are also referred to as ***DomU***s.

2 Installing the Xen environment on the host system

Xen has been integrated in Univention Corporate Server in Version 3.2. Alongside the necessary management tools and the hypervisor there are also a number of kernel packages specifically adapted to Xen. The kernel packages adapted to Xen feature DomO support and are available in kernel versions 2.6.18 and 2.6.26. UCS 2.3 provides an additional kernel based on UCS 2.6.30, but which provides only support for operation as a DomU (the 64 GB variant of the kernel must be used on the i386 architecture for this purpose).

The Xen components can be selected in the software selection under ***Virtualisation*** in the Univention installer.

Installing the ***univention-xen*** package installs all required packages. As long as there are no other Xen kernel versions already installed on the system, version 2.6.26 of the Xen-compliant kernel is installed automatically when the package is installed. If required, the respective missing kernel version can be installed subsequently using the ***univention-kernel-image-2.6.18-xen*** or ***univention-kernel-image-2.6.26-xen*** package.

During the installation the new entries for direct system start into the hypervisor are set up in the Grub bootloader. Once installation is complete, the system must be rebooted and a kernel image beginning with ***Xen 3.2*** selected.

Unlike when using standard kernels, the system start for kernels with Xen DomO support does not display a graphic bootsplash with a progress bar showing details of the system components starting up.

As of UCS 2.2-0 the graphic desktop login screen is no longer started automatically on Xen DomO systems as Vesa X servers cannot currently function correctly on these systems in combination with the 2.6.26 kernel. If a different X server or kernel is used, the Univention Configuration Registry variable `gdm/autostart` can be set to ***yes*** and the graphic desktop log-in screen started as a test by running `invoke-rc.d gdm start`.

The hypervisor is provided by the ***xend*** service. This must be restarted after any alterations are made to the configuration.

The configuration parameters of a guest instance (DomU) are stored in text-based configuration files. As standard these are saved under `/etc/xen` in a UCS installation. The image files for the partitions of a guest instance (DomU) are saved under `/var/lib/xen`.

The ***ConVirt*** program can be used as a management tool for managing Xen instances.

3 Installing Xen guest systems

3.1 Installing an unmodified guest system (e.g., Windows)

The following describes the necessary steps for operating an unmodified operating systems using Microsoft Windows XP as an example.

Firstly a name must be selected for the guest instance; in this example we have selected ***winxp***.

The data from the guest instance's (DomU's) virtual hard drive are saved in an image file, which now needs to be created: The following commands are used to create an 8 GB hard drive image:

```
mkdir -p /var/lib/xen/winxp/
dd if=/dev/zero of=/var/lib/xen/winxp/hdimage bs=1M count=1 seek=8000
```

Xen allows the mounting of ISO9660 images as virtual CD-ROM drives. In this example we assume that a CD image of Microsoft Windows XP is available under `/var/lib/xen/winxp/winxp.iso`.

An example configuration for the operation of unmodified operating systems can be found under `/usr/share/doc/univention-xen/examples/basis-vmx.cfg`. This should be copied as a basis:

```
cp /usr/share/doc/univention-xen/examples/basis-vmx.cfg /etc/xen/winxp.cfg
```

The following settings of this example configuration must be altered at the minimum:

- ***winxp*** should be entered as the ***name***.
- The following lines can be used to mount the image files mentioned above in the guest instance. The image of the hard drive then becomes writeable, the virtual CD-ROM drive is mounted read-only.

```
disk = ['file:/var/lib/xen/winxp/hdimage,hda,w',
        'file:/var/lib/xen/winxp/winxp.iso,ioemu:hdc:cdrom,r']
```

If the guest instance is to be endowed with a network connection, at least one virtual network card must be configured in addition, the MAC address of which is hereby specified. An example for a network card in an unmodified operating system:

```
vif = [ 'type=ioemu', 'mac =00:16:3E:27:1F:A6' ]
```

If a Xen-compliant Linux is used as the operating system kernel for the guest instance, the guest interface can be accessed directly:

```
vif = [ 'mac =00:16:3E:27:1F:A6' ]
```

Further information on setting-up the network operation can be found in Chapter 5.

The configuration of a remote access over VNC or RDP is explained in Chapters 5.3 and 5.4.

The newly created domain can then be started using the command `xm create /etc/xen/winxp.cfg`.

3.2 Installing the drivers for paravirtualisation of Windows operating systems

If an unmodified operating system as described in Chapter 3.1 is installed, the accesses to the network card and block devices such as hard drives are emulated by the hardware emulator QEMU.

Installing drivers in the DomU system which allow direct access to the hardware of the DomO system results in a considerable increase in performance and reliability. Alongside proprietary drivers there is also an Open Source driver called GPLPV. This can be downloaded from http://www.meadowcourt.org/downloads/; the file is called ***Xen PV Drivers.exe***. The Service Pack 2 must be installed for the installation of the GPLPV driver in Windows XP to be successful.

The installer can be run directly; the Windows variants to be used can be selected on the first screen. The shutdown manager should also be installed.

The majority of the drivers for Windows operating systems are cryptographically signed by the manufacturer. This is not the case, however, for GPLPV drivers. On some Windows versions (e.g., Windows 2003) exceptions can be added for the ***logotest*** and ***Authenticode*** during installation. Current 64-bit Windows versions must be signed as described in the instructions in the Microsoft Developer Network: http://msdn.microsoft.com/en-us/library/aa906283.aspx

For GPLPV drivers with a version preceding 0.9.12-pre9 the central boot configuration file must be adapted accordingly after the installation: **Right click on My Computer > Properties > Advanced > Start and Restore > Settings > System start > Edit**. The active line under **Operating systems** must be copied and an additional ***/GPLPV*** added, e.g., (One line respectively and without the backslash)

```
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Windows Server 2003, \
          Enterprise, GPLPV" /noexecute=optout /fastdetect /gplpv
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Windows Server 2003, \
          Enterprise" /noexecute=optout /fastdetect
```

The driver is used as standard for GPLPV drivers with a version above 0.9.12-pre9, unless /NOGPLPV is entered. In this case it can prove practical for safety reasons to adapt the boot configuration file in the same way as in the process described above, so that a line with /NOGPLPV is entered.

The DomU must then be restarted and the GPLPV entry selected during the Windows start-up. The subsequent hardware recognition will set up an array of drivers containing Xen in their name. The network connection must also be reconfigured. It must be noted that the originally configured QEMU network card is deactivated; i.e., the set-up should not be performed over RDP. A further restart will activate the paravirtualised drivers.

3.3 Installing a UCS guest instance with paravirtualisation support

The following describes the necessary steps for installing a modified guest instance (DomU) using Univention Corporate Server 2.3 as an example. The instructions can, however, be used in the same way for UCS 2.2. You also have the possibility of installing Univention Corporate Server unmodified, although this will result in a loss of speed do to the emulation of ***qemu***.

The process described here uses the Univention installer with an adapted initial ramdisk (also referred to as Initrd).

3.3.1 Generating the initial ramdisk

The initial ramdisk only needs to be created once and can then be used for further installations.

First, the ISO image of the installation DVD must be copied and mounted via the loopback device.

```
mkdir -p /mnt/iso
mount -o loop ucs_2.3-0-*-dvd-amd65.iso /mnt/iso
```

The Initrd must then be extracted. In this case using the Initrd for the 2.6.26 kernel as an example. The Initrds for other kernel versions can also be used. The file name for UCS 2.2 is `linux-2.6.bin`.

```
gzip < /mnt/iso/boot/linux-2.6.26.bin > /tmp/ucs_2.3-initrd.ext3
```

The file system should now be enlarged as the installation kernel contains more modules:

```
dd if=/dev/zero bs=1M count=50 >> /tmp/ucs_2.3-initrd.ext3
resize2fs -f /tmp/ucs_2.3-initrd.ext3
```

The following mounts the initrd and replaces it with the modules of the current kernel:

```
umount /mnt/iso
mount -o loop /tmp/ucs_2.3-initrd.ext3 /tmp/iso/
rm -rf /mnt/iso/lib/modules/2.6.*
cp -Rf /lib/modules/$(uname -r)/ /mnt/iso/lib/modules/$(uname -r)
```

As the terminal emulation of the Xen console only offers limited visualization possibilities, the graphic mode of the Univention installer is now set to a simpler version:

```
sed -i ’s|params="|params="--simple |’ /mnt/iso/sbin/univention-installer
```

The initrd is then unmounted and compressed once more:

```
umount /mnt/iso
gzip < /tmp/ucs_2.3-initrd.ext3 > /tmp/ucs_2.3_$(uname -r)_initrd
```

3.3.2 Installation with the Univention installer

Firstly a hard drive image must be created for the guest instance. In this example a 12 GB file is created for the hard drive.

```
 mkdir -p /var/lib/xen/ucs-master/
 dd if=/dev/zero of=/var/lib/xen/ucs-master/disk bs=1M count=1 seek=12000
 fdisk /var/lib/xen/ucs-master/disk
```

Two Xen configuration files must now be created; one for the installation and one for subsequent system operation. The details in this example must be adapted to reflect the respective environment. The path to the initial ramdisk created above should be entered for RAMDISK:

```
# /etc/xen/ucs-master-installation.cfg
name = 'ucs-master'
kernel = "/boot/vmlinuz-2.6.18-ucs105-xen-686"
memory = 512
vcpus = 1
disk = [ 'file:/var/lib/xen/ucs-master/disk,hda,w',
         'file:/var/lib/xen/ucs-master/ucs_2.3-0-*-dvd-amd65.iso,scd0,r' ]
vif = [ 'type=ioemu', 'mac=00:16:3E:27:1F:FF' ]
boot = 'c'
ramdisk='RAMDISK'
root='/dev/ram ro'
extra='xencons=tty vt.default_utf8=0 ramdisk_size=204800 flavor=linux
       text forceascii'
```

```
# /etc/xen/ucs-master.cfg
name = 'ucs-master'
kernel = "/boot/vmlinuz-2.6.18-ucs105-xen-686"
memory = 512
vcpus = 1
disk = [ 'file:/var/lib/xen/ucs-master/disk,hda,w' ]
vif = [ 'type=ioemu', 'mac=00:16:3E:27:1F:FF' ]
boot = 'd'
root = "/dev/hda2 ro"
extra='xencons=tty'
```

The Xen installation can then be started with the command `xm create -c /etc/xen/ucs-master-installation.cfg`.

Following successful installation, the configuration file planned for the running operations should be used to start the DomU.

4 Administration, migration and snapshots of DomUs

The central administration tool within the host system (Dom0) is `xm`. An overview of the most important commands:

Command	**Description**
xm list	An overview of the domains in the current system. The parameter -l can be used to display an verbose overview of the domain resources.
xm console DOMAIN	Connects to the terminal console of a Linux-based domain.
xm create CONFIGFILE	Starts a domain using the configuration specified in the CONFIGFILE.
xm save DOMAIN SNAPSHOTFILE	Saves a snapshot of the working memory of a domain and then ends the domain. The parameter -c must be passed if the domain should not be ended.
xm restore SNAPSHOTFILE	Loads a snapshot of a domain previously saved with the command `xm save`.

xm migrate DOMAIN HOST	The command is used to migrate a domain to another Xen system. If the -live option is passed, the migration is performed in running operation and is transparent for the users of the domains.
xm top	Gives an overview of the currently running domains and their resource consumptions.
xm shutdown DOMAIN	Shuts the domain down. (Comparable with the `halt` command in Linux.)
xm destroy DOMAIN	Closes the domain directly. (Comparable with pressing a system's power switch.)
xm reboot DOMAIN	Restarts a domain.
xm pause DOMAIN	Stops the running of a domain.
xm unpause DOMAIN	Resumes a paused domain.

5 Remote access to guest instances (DomUs)

5.1 Configuring the network connection of a guest domain (DomU)

The network connection managed by ***univention-xen*** uses a ***bridge*** to connect guest instances to the host system.

The bridge is created under the name ***xenbr0***. The host system's network card appears inside the bridge as ***vif0.0***, the guest instances' (DomUs') as ***vifX.Y***, where ***X*** identifies the guest instance (DomU) and ***Y*** from 0 upwards represents the virtual network devices. E.g., ***vif1.0*** and ***vif1.1*** for a system with a guest instance (DomU) with two virtual network cards.

The ***peth0*** device is used to connect the bridge to the system's physical network card. As standard there is no interface defined for the bridge. It can be configured using the Univention Configuration Registry variable `xen/bridge/interface`. Once the `xen/bridge/interface` has been set, the ***xend*** service must be restarted by running the `/etc/init.d/xend restart` command.

The network cards appear unchanged in the guest instances. Linux network cards can be actuated as e.g., ***eth0***, and Windows network cards can be configured in the network overview. The default gateway and the DNS resolution must also be configured as usual.

Alongside the bridge there are two further approaches for creating a network connection: via a direct configuration of the routing and resolution of the MACs via Proxy ARP or via Network Address Translation. They are, however, as of yet not supported by ***univention-xen*** and must be configured manually in `/etc/xen/xend-config.sxp`.

5.2 Terminal access in Linux-based guest instances

`xm console DOMAIN` can be used to access the terminal console of Linux-based guest instance. The key combination ***Strg +]*** can be used to end the console again. The active DomU instances can be ascertained using `xm list`.

5.3 Graphical VNC access on guest instances

To allow an access via VNC, the following entries must be made in the guest instance's configuration file.

```
vnc = 1
vncpasswd="secret"
```

As standard the access is only possible from a local host, but the Univention Configuration Registry variable `xen/vnc/listen` can be used to approve a different network area, e.g., ***0.0.0.0***, to allow access across the network.

A VNC viewer (in Univention Corporate Server with `vncviewer` or `krdc`) can then be used to access the graphic version under the IP address of the host system. As standard the first free VNC display can be found under ***:0***.

5.4 Graphical RDP access on guest instances

The use of the ***Remote Desktop Protocol*** (RDP) is recommended for the productive operations of a Windows guest instance. Access via RDP - and particularly the display of the mouse pointer - is considerably smoother in comparison with VNC.

RDP access can be activated in Microsoft Windows Server 2003, Windows XP Professional and Windows Vista Business/Enterprise/Ultimate under **System settings > Computer configuration > Remote assistence** by activating the [**Remote assistance**] and [**Remote desktop**] options.

RDP clients are delivered with Microsoft Windows as standard as of Windows XP. The `rdesktop` and `krdc` packages are delivered with Univention Corporate Server for the access of Linux-based systems.

5.5 Local access on graphically run guest instances

The Xen packages are compiled with support for the ***Simple DirectMedia Layer*** (SDL) graphics library in Univention Corporate Server. A direct, local access from the host system is then possible. For this, the access via VNC must be deactivated:

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

A graphic display of the guest instance is created following a `xm create`.