

ctys-XEN(7)

Installation and Configuration

November 29, 2010

Contents

1	XEN - Basics for Operations	2
2	Supported HOST-OSs	2
3	Supported GuestOSs	2
4	Supported Architectures	3
5	Supported Interfaces	3
6	Supported VM Management Interfaces	3
7	Network Interconnection	3
8	Installation of Components	3
8.1	Xen-3.0.3	3
9	Install Procedures	3
9.1	ctys-createConfVM	3
9.2	Supported/Tested Install-Mechanisms	4
9.2.1	PARA Virtualization	4
9.2.2	HVM Virtualization	5
10	Installation of Guests	5
10.1	CentOS-5	5
10.2	debian-5.0	7
10.3	Fedora-8	7
10.4	OpenSUSE-10.1	7
10.5	OpenSUSE-10.2	8
10.6	OpenSUSE-11.2	8
10.7	Ubuntu-8.04	8
10.8	Ubuntu-9.10	8
11	Installed Systems	9
12	Configuration Files	9
13	SEE ALSO	10
14	AUTHOR	10
15	COPYRIGHT	10

List of Tables

1	PARA Virtualization	4
2	HVM Virtualization	5
3	Overview of Installed-Xen-VMs	9

List of Figures

1	ctys distributed access	2
3	Distributed installation by 'ctys-confCreateVM'	6

1 XEN - Basics for Operations

The ctys-XEN plugin supports paravirtualized and hardware based VMs. Particularly a generic Python module is generated by the installer **ctys-createConfVM(1)**, which provides runtime dynamic re-allocation within filesystem and dynamic assembly of devices. The installer also copies the external runtime kernel into the VM directory, thus providing the usage in multiple distributions. Therefore a meta-layer for an abstract interface is defined, which is implemented by a wrapper script. The wrapper script is written in bash syntax and sourced into the runtime process, but could be used for native command line calls as well. The wrapper-script by default does not overwrite the preconfigured values of the Python module, which is read by the xm-call.

The main advance of using a wrapper script is the ability to perform dynamic scripting within the configuration file, which is standard bash-syntax with a few conventions. When activating the hardcoded CLI variable within the wrapper script the most of the parameters from the configuration file are superposed by command line variables.

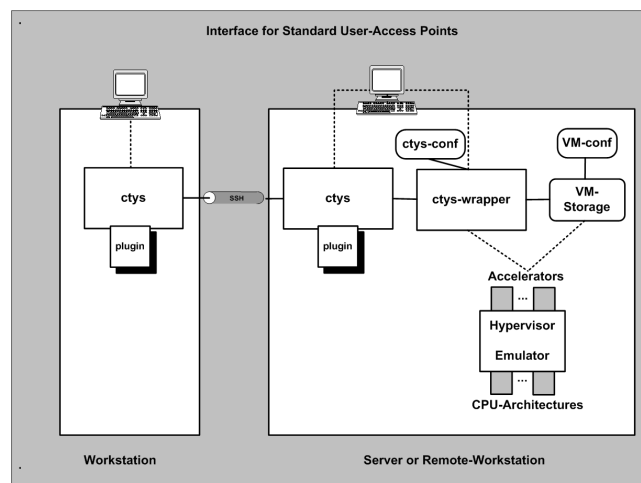


Figure 1: ctys distributed access

An installer for complete setup of a QEMU and/or KVM based VM is contained. The tool **ctys-createConfVM(1)** interactively creates a local or remote configuration by detection of the actual platform and creation of a ready-to-use startup configuration.

The created configuration set comprises the generic wrapper-script, a specific configuration file, and some installation utilities such as a debootstrap-wrapper in case of debian installation. The actual installation of a GuestOS could be performed either by calling the wrapper-script or by calling ctys with the **INSTMODE**. When no additional suboptions are provided the generated defaults are used.

Some differences occur for the automated creation of required virtual HDDs. In case of offline-installers like debootstrap for simplicity reasons multiple virtual partitions in separate image files will be created, whereas the OS based installation by default is performed by the partitioning of one virtual HDD.

Basic Use-Cases for application are contained within the document *ctys-uc-XEN(7)*.

2 Supported HOST-OSs

The current version is tested on: CentOS, debian, Fedora, OpenSUSE, ScientificLinux.

3 Supported GuestOSs

The following GuestOS are tested:

CentOS-5, debian-5.0, Fedora 8, OpenSolaris-2009.6, SL-5.4.1, SuSE-10.1, SuSE-10.2, SuSE-11.2, Ubuntu-8.04, and Ubuntu-9.10

4 Supported Architectures

The x86 platform fo 32bit and 64bit are supported with paravirtualized and hardware based virtualization environments.

5 Supported Interfaces

Any provided interface by Xen.

6 Supported VM Management Interfaces

The vm management is performed by xm and virsh.

7 Network Interconnection

The default is to provide one interface only, which could be any of the provided interfaces.

8 Installation of Components

The installation of the Dom0 is recommended to be based on a available distribution as provided if not required else. A quite straight forward installation could be set up by CentOS/RHEL-5, with additional updates The used from-the-box kernel is **2.6.18-8.1.15.el5.centos.plusxen**. For additional information refer to [?, CENTOS].

8.1 Xen-3.0.3

The installation of Xen depends on the used kernel. The installation of the standard kernel requires just the installation of the Xen kernel and modules.

Configure the kernel with

```
kernel: dom0_mem=1536M com1=115200,8n1
module: console=ttyS0,9600n8 console=tty1
```

9 Install Procedures

9.1 ctys-createConfVM

The current version provides a generic installer for all supported hypervisors. The functionality related to the plugin Xen includes the complete configuration of a VM, the creation of a generic wrapper-script, and some specific install utilities. The process could be performed either interactively by about 20 to 30 steps, or automatically, when appropriate default values within configuration files are in place. Refer to **ctys-createConfVM(1)**.

The provided install procedures are based on ISO images of CDROMs/DVDs and on network based installation, this comprises kickstart, debootstrap, and PXE. The network based installation requires one or more of the following services to be present.

DHCP + DNS HTTP, FTP, NFS TFTP

For additional services the following will be helpful:

- PXE/PXELinux
- Kerberos+LDAP+(Samba/SMB)+automount

9.2 Supported/Tested Install-Mechanisms

The actual applicable install mechanism partly depends on the host system, e.g. debootstrap is currently available on debian hosts only.

9.2.1 PARA Virtualization

OS	debootstrap	KS	PXE	CD	FD	HD	USB
CentOS-5.0	-	X	X	X			
CentOS-5.4	-	X	X	X			
Debian-4.0r3-ARM	-	-		X	-		
Debian-4.0r3		-	X	X			
Debian-5.0.0	X	-		X			
Fedora 8	-	X	X	X			
Fedora 10	-	X	X	X			
Fedora 12	-	X	X	X			
FreeBSD-7.1	-	-		X			
FreeBSD-8.0	-	-		X			
FreeDOS/balder	-	-		X			
Mandriva-2010	-			X			
ScientificLinux-5.4.1	-		X	X			
OpenBSD-4.0	-	-	X	X			
OpenBSD-4.3	-	-	X				
OpenSuSE-10.2	-	-	X	X			
OpenSuSE-11.1	-	-	X	X			
OpenSuSE-11.2	-	-		X			
Solaris 10	-	-		X			
OpenSolaris 2009.6	-	-		X			
Ubuntu-8.04-D		-	X	X			
Ubuntu-9.10-D		-		X			
uClinux-arm9	-	-	-		-	-	
uClinux-coldfire	-	-	-		-	-	-

Table 1: PARA Virtualization

9.2.2 HVM Virtualization

OS	debootstrap	KS	PXE	CD	FD	HD	USB
CentOS-5.0	-	X	X	X			
CentOS-5.4	-	X	X	X			
Debian-4.0r3-ARM	-	-		X			
Debian-4.0r3		-	X	X			
Debian-5.0.0		-	X	X			
Fedora 8	-	X	X	X			
Fedora 10	-	X	X	X			
Fedora 12	-	X	X	X			
FreeBSD-7.1	-	-		X			
FreeBSD-8.0	-	-		X			
FreeDOS/balder	-						
Mandriva-2010	-			X			
ScientificLinux-5.4.1	-	X	X	X			
OpenBSD-4.0	-	-	X	X			
OpenBSD-4.3	-	-	X				
OpenSuSE-10.2	-	-	X	X			
OpenSuSE-11.1	-	-		X			
OpenSuSE-11.2	-	-		X			
Solaris 10	-			X			
OpenSolaris 2009.6	-			X			
Ubuntu-8.04-D		-		X			
Ubuntu-9.10-D		-		X			
uClinux-arm9	-	-	-	-	-	-	-
uClinux-coldfire	-	-	-	-	-	-	-

Table 2: HVM Virtualization

10 Installation of Guests

10.1 CentOS-5

The given example is based on the following configuration:

- Host executing ctys-createConfVM on debian-5.0.0, no qemu installation.
- Target host for execution of Xen with Scientific Linux 5.4.1, present Xen installation.
- GuestOS within the Xen PARA virtualized VM is CentOS-5.4.

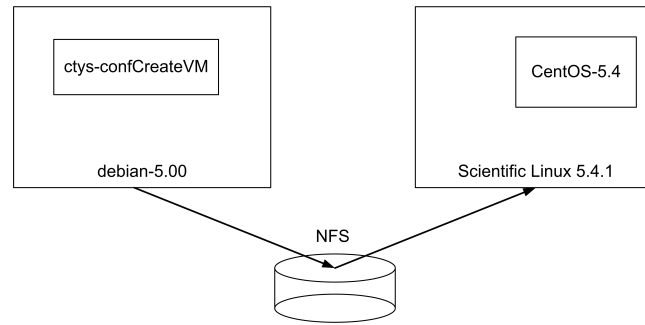


Figure 3: Distributed installation by 'ctys-confCreateVM'

The 'cross-installation' for a different machine requires various options to be set for the target execution environment of the hypervisor which cannot be detected automatically on the local machine. One example is the actual support for the architecture.

Several of the parameters has to be set by environment VARIABLE based options. The available options could be visualized by the call option '-list-environment-variable-options' or for short '-levo'. The initial default values wre displayed too, which is not available for dependent values. E.g. by convention wihtin ctys the LABEL of a VM is the hostname of the conatined GuestOS, therefore the value of TCP/IP parameters could be determined only after the LABEL value is defined. The options by environment variable are applicable for local execution only, else the interactive dialogue is available only. The most value of the pre-defined values is gained in combination with oneof the options '-auto' or '-auto-all'. These pre-define confirmation answers for the interactive dialogues.

The following call creates in place of appropriate default values fully automated the complete set of configuration files and wrapper-scripts, including a basic kickstart file.

```
ACCELERATOR=PARA \
DIST=CentOS  RELEASE=5.4 \
OS=Linux      OSVERSION=xen-3.1.2-164.15.1.el5 \
ctys-createConfVM \
  -t XEN \
  --label=tst488 \
  --auto-all
```

When dropping the '-auto-all' option an interactive dialogue is performed, where no default values are required neccessarily. The environment variable based automation is available for local installation only, whereas the dialogue could be performed on a remote maschine.

The first step prepares the required infrastructure for starting the VM. The installation is foreseen to be performed by the facilities provided by the GuestOS. In case of CentOS the alternatieves PXE, CD/DVD, and KS(kickstart) are provided. The start of the intsallations could be executed by the following call.

```
ctys -t XEN -a create=1:tst488,b:<base-path-VM>,reuse,console:gterm,\
  instmode:KS%default%HDD%default%init \
  -c off \
  tstUser@testHost'(-d pf)'
```

The **instmode** suboption initiates an installation, where the source and target paths are provided by default behaviour. The 'console:gterm' starts for the console a 'gnome-terminal', any supported type could be provided. The '-c off' deactivates the cache evaluation. The 'b:<base-path-VM>' suboption is a specific enhancement when working with NFS for reduction of filescans. The generic remote option '-d pf' is a debugger option, where 'pf' is a shortcut for 'printfinal', which traces all final assemblies of system calls just before execution. These calls are generally executable by cut-and-paste manually.

Alternatively either the direct call of the wrapper script or the Xen configuration file could be executed. E.g. by direct call of the generic Python script.

```
/usr/sbin/xm create /mntn/vmpool/vmpool05/Xen/test/tst-ctys/tst488/tst488.conf\
con=nongraphic
```

10.2 debian-5.0

1. Add packages:

- (a) bridge-utils
- (b) openssh-server
- (c) krb5
- (d) openldap
- (e) vnc4server
- (f) vnc4viewer

2. Configure udev and network

The following example shows the call for the update of configuration files for a previous installation of debian-5.0.0 PARA virtualized VM. Here the complete set of required variables is pre-set by environment variables, particularly a kernel is copied into the VM directory, where the origin is set to a non-standard PATH. The execution is performed in batch mode by usage of '-auto-all'.

```
DOMU_KERNEL=old/boot/vmlinuz-2.6.26-2-xen-amd64 \
DOMU_INITRD=old/boot/initrd.img-2.6.26-2-xen-amd64 \
INST_KERNEL=old/boot/vmlinuz-2.6.26-2-xen-amd64 \
INST_INITRD=old/boot/initrd.img-2.6.26-2-xen-amd64 \
ACCELERATOR=PARA \
DIST=debian \
DISTREL=5.0 \
OS=Linux \
OSREL=2.6.27.7-9-xen \
BRIDGE=eth1 \
MAC=00:0e:0c:36:a8:aa \
IP=172.20.1.78 \
ctys-createConfVM \
-t XEN \
--label=tst256 \
--no-create-image \
--auto-all
```

The following call starts the HVM based VM.

```
./tst256.sh --console=vnc
```

10.3 Fedora-8

Straight forward.

10.4 OpenSUSE-10.1

Somewhat tricky, dropped due to canceled support.

10.5 OpenSUSE-10.2

Somewhat tricky, dropped due to canceled support.

10.6 OpenSUSE-11.2

Standard-Installation on the PM, following the steps:

1. Reconfigure the partition sizes for additional kernels
2. Additional packages: Xen-Kernel, krb5, pam, cyrus-sasl.
3. QEMU

The configuration comprises:

4. Eventually disable the firewall for a lab-machine.

10.7 Ubuntu-8.04

The following example shows the call for the update of configuration files for a previous installation of Ubuntu-8.04 as HVM on a debian-5.0.0 host. Here the complete set of required variables is pre-set by environment variables, and the execution is performed in batch mode by usage of '-auto-all'.

```
ACCELERATOR=HVM \  
DIST=Ubuntu \  
DISTREL=8.04 \  
OS=Linux \  
OSREL=2.6.27.7-9-xen \  
BRIDGE=eth1 \  
MAC=00:50:56:13:12:35 \  
IP=172.20.2.53 \  
ctys-createConfVM \  
  -t XEN \  
  --label=tst253 \  
  --no-create-image \  
  --auto-all
```

The local call of the wrapper script for the first test displays the resulting call.

```
./tst253.sh --console=vnc --print --check
```

The following call starts the HVM based VM.

```
./tst253.sh --console=vnc
```

10.8 Ubuntu-9.10

Standard-Installation on the PM, following the steps:

1. Add vnc4, bridge-utils, openssh-server

11 Installed Systems

OS	name	Inst-VM	Dom0	DomU
CentOS-5.0	tst100	Xen-3.0.2-PARA	X	X
CentOS-5.1	tst101	Xen-3.0.2-PARA	X	X
CentOS-5.4	tst228	Xen-3.0.2-PARA	X	X
CentOS-5.4	tst247	Xen-3.0.2-HVM	X	X
debian-5.0.0	tst256	Xen-3.0.2-PARA	X	X
debian-5.0.0	tst248	Xen-3.0.2-HVM	X	X
Fedora 8	tst114	Xen-3.0.2-PARA	X	X
OpenSolaris-2009.6	tst244	Xen-3.0.2-HVM		X
SL-5.4.1	tst258	Xen-3.0.2-HVM	X	X
SL-5.4.1	tst229	Xen-3.0.2-PARA	X	X
SuSE-10.1	tst153	Xen-3.0.2-PARA		(X)
SuSE-10.2	tst153	Xen-3.0.2-PARA		X
SuSE-11.2	tst250	Xen-3.0.2-PARA	X	X
Ubuntu-8.04	tst253	Xen-3.0.2-HVM	X	X
Ubuntu-9.10	tst252	Xen-3.0.2-PARA		X
Ubuntu-9.10	tst235	Xen-3.0.2-HVM		X

Table 3: Overview of Installed-Xen-VMs

12 Configuration Files

For additional information refer to configuration directory.

`$HOME/.ctys`

13 SEE ALSO

ctys(1) , *ctys-CLI(1)* , *ctys-configuration-XEN(7)* , *ctys-createConfVM(1)* , *ctys-plugins(1)* , *ctys-PM(1)* , *ctys-uc-XEN(7)* , *ctys-vhost(1)* , *virsch(18)* , *ctys-VNC(1)* , *ctys-X11(1)* , *ctys-XEN(7)* , *xm(1)*

14 AUTHOR

Maintenance: <acue_sf1@sourceforge.net>
Homepage: <<http://www.UnifiedSessionsManager.org>>
Sourceforge.net: <<http://sourceforge.net/projects/ctys>>
Berlios.de: <<http://ctys.berlios.de>>
Commercial: <<http://www.i4p.com>>



15 COPYRIGHT

Copyright (C) 2008, 2009, 2010 Ingenieurbuero Arno-Can Uestuensoez
For BASE package following licenses apply,

- for software see GPL3 for license conditions,
- for documents see GFDL-1.3 with invariant sections for license conditions,

This document is part of the **DOC package**,

- for documents and contents from DOC package see
'Creative-Common-Licence-3.0 - Attrib: Non-Commercial, Non-Deriv'
with optional extensions for license conditions.

For additional information refer to enclosed Releasenotes and License files.

