



Linux Professional Institute Japan

LPI-JAPAN

LPIC 304

技術解説セミナー

2015年1月18日(日) 13:30～16:30

LPI-Japan認定トレーナー

ゼウス・ラーニングパワー 講師

鯨井 貴博



講師紹介



**Linux・Networkの基礎から、
現場経験を活かしたトラブルシュートまで幅広い講義を行います！**



[プロフィール]

LPIC レベル3・情報処理技術者資格などを保有し、
現場ではネットワーク構築・海外メーカー国内一次代理店でのテクニカルサポート業務を経験。
現在は講師として自身がIT未経験から学んできた経験を交えた講義を行っている。

また、LPI-Japan認定トレーナー、学校法人電波学園 東京電子専門学校 Linux担当非常勤講師、オープンソース技術 開発研究所、日本Nginxユーザー会としても活動中。

[保有資格]

LPIC レベル3 Core(301)

LPIC レベル3 Mixed Environment(302)

LPIC レベル3 Security(303)

LPIC レベル3 Virtualization & High Availability(304)

HTML5プロフェッショナル レベル1

情報処理技術者 応用情報処理技術者/基本情報処理技術者

MCP Microsoft Windows Server 2008 Active Directory

ITIL Foundation

CCNA

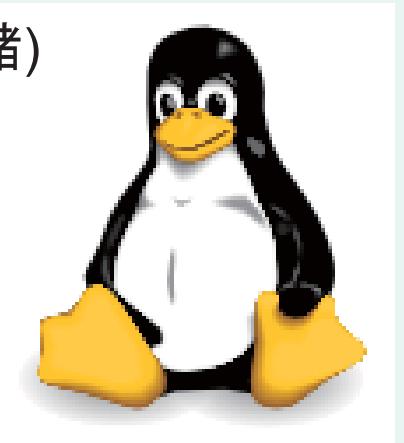




講師紹介



Linux開発者のLinus Torvaldsさんと私
(ゼウス・ラーニングパワーCarl先生も一緒)
in LinuxConJapan



nginx開発者のIgor Sysoevさんと私
in 日本Nginxユーザー会



ゼウス・ラーニングパワー



@ZeusLearning **BLOG**

<http://blog.zeus-learning.jp>



<http://opensourcetech.hatenablog.jp/>
(ちょっとマニアックな技術情報満載)



講師紹介



ITトレーニングで、
LPICレベル2の問題作成などもします。



以下のコマンド結果において、CPUの処理が行われていない割合を示す数値はどれか選択して下さい。(201の範囲からの出題)

```
$ vmstat
proc-----memory-----swap-----io----cystem---
cpu-----
r h wnd free buff cache si on hi hi in nc us
sy id wa st
2 1 0 1110120 183460 1707964 0 0 13009 668 1420 2333
8 12 48 30 0
```

1. 9
 2. 12
 3. 48
 4. 30

解説する

問題集

ゼウス・ラーニングパワー

問題提供: ゼウス・ラーニングパワー株式会社

LPIC-1 資格レベルの万を対象に、カーネル、ファイルシステム等のシステムに関する事項、またWebサーバ(やDNSサーバ)等の構造を学習します。

Copyright (c) 2013-2014 ZEUS LP Co.,Ltd. All Rights Reserved.



本日のアジェンダ

- ①LPIC304の試験概要
- ②LPIC304を学習する上でのポイント
- ③技術解説

主題330:仮想化

- 330.1 仮想化の概念と理論
- 330.2 Xen
- 330.3 KVM

主題332:クラスタ管理

- 332.1 Pacemaker

主題333:クラスタストレージ

- 333.1 DRBD

- ④お知らせ

- ⑤質疑応答



① LPIC304の試験概要



LPICの構成

初級

上級

LPICレベル1 LPIC-1

サーバの
構築、運用・保守

実務で必要なLinuxの基本操作とシステム管理が行える
エンジニアであることを証明できます。

LPICレベル2 LPIC-2

ネットワークを含む、
コンピュータシステムの構築、運用・保守

Linuxのシステムデザイン、ネットワーク構築において、企
画、導入、維持、トラブルシューティング、キャパシティブ
ランニングができるエンジニアであることを証明できます。

LPICレベル3 LPIC-3

各分野の最高レベルの技術力を持つ
専門家レベル

LPIC-3 Specialty Mixed Environment

(300試験):

Linux、Windows、Unixが混在するシステムの設計、構
築、運用・保守ができるエキスパートエンジニアであるこ
とを証明できます。

LPIC-3 Specialty Security (303試験):

セキュリティレベルの高いコンピュータシステムの設計、
構築、運用・保守ができるエキスパートエンジニアである
ことを証明できます。

LPIC-3 Specialty Virtualization & High Availability (304試験):

クラウドコンピューティングシステム(クラウド)の設計、構
築、運用・保守ができるエキスパートエンジニアであるこ
とを証明できます。

該当試験

101試験

102試験

201試験

202試験

300試験

303試験

304試験



LPIC304の試験概要

どのような試験か？

Virtualization & High Availabilityの名の通り、
仮想化・高可用性を理解し、
クラウドの構築/運用の専門家を証明するものです。

試験時間：90分

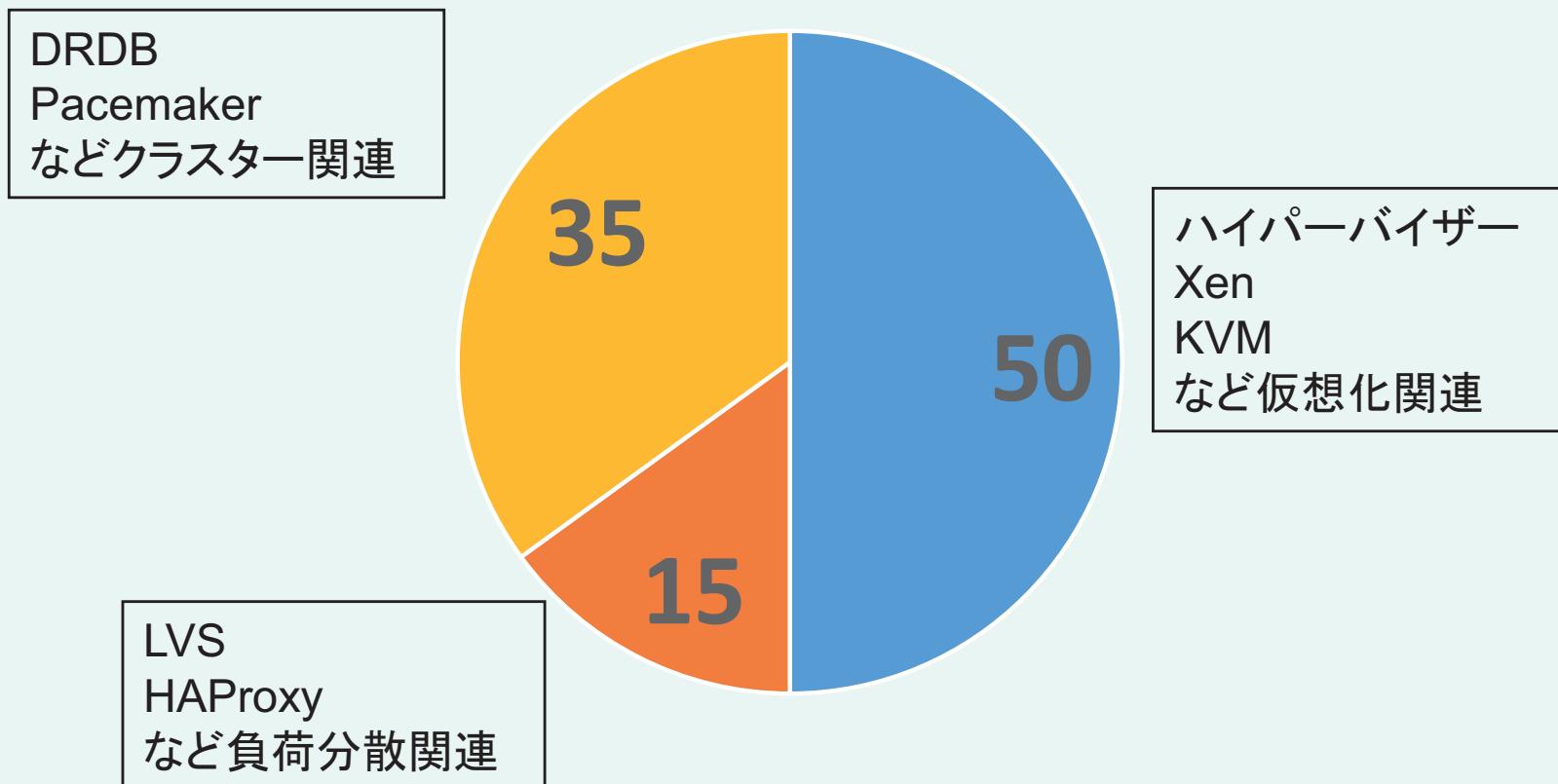
出題数：約60問

出題形式：選択/記述

受験費用：30,000円(消費税別)



出題範囲の割り合い(%)



詳細は、
<http://www.lpi.or.jp/lpic3/range/range304.shtml>を
ご確認下さい。





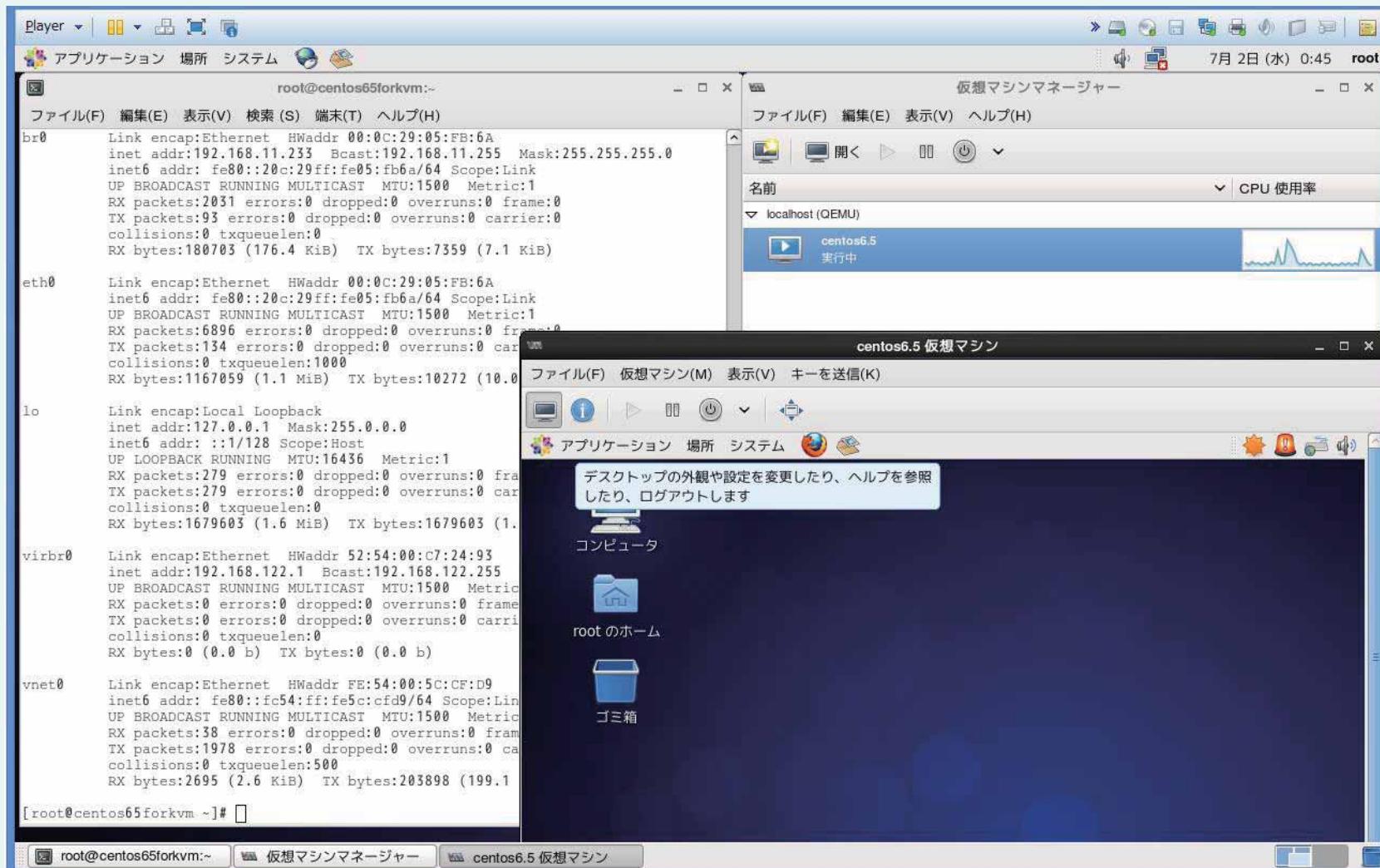
②LPIC304を学習する上のポイント



LPIC 304 技術解説セミナー



- 実機での操作にてどのような機能・操作方法など、
しっかり確認すること！





・注意点！！

XenやKVMなどを仮想マシン上/物理マシン上
どちらで環境構築する場合も、
64ビット・仮想化支援機能(Intel-VT/AMD-V)に
対応したCPU、2~4GB以上の搭載メモリ(目安)、
HD容量にも余裕がある など
ある程度のマシンスペックが必要となります。



LPIC 304 技術解説セミナー



LPI認定教材等

他には専門書を読み解く、各ソフトウェアのサイト(英語情報多い)を読むなどの方法があります。

<http://www.lpi.or.jp/lpic3/book.shtml>



LPIC Level3対応 LPI認定教材	
	<p>NEW!</p> <p><u>Linux仮想化と高可用性</u> <u>- LPI304対応 -</u></p> <ul style="list-style-type: none"> • <u>有限会社ナレッジデザイン</u> • <u>対象:304試験</u>

LPIC-3 304試験 学習向け参考ページ・参考書籍(LPI認定教材ではありません)

<p>LPIC-3 303/304試験 学習向け問題集 LPI認定教材ではありません</p>	<p>徹底攻略LPI問題集 Level3[303/304]対応</p>
<p>仮想化技術 の日本語参考図書</p>	<ul style="list-style-type: none"> • <u>Xen徹底入門 第2版</u> 宮本 久仁男, 平 初, 長谷川 猛, 津村 彰 著 • <u>KVM徹底入門 Linuxカーネル仮想化基盤構築ガイド</u> 平 初, 森若 和雄, 鶴野 龍一郎, まえた こうへい 著
<p>クラスタ技術 の日本語参考図書</p>	<ul style="list-style-type: none"> • <u>Linux高信頼サーバ構築ガイド クラスタリング編</u> 笠野 英松 著 • <u>Linuxで作るアドバンストシステム構築ガイド</u> デージーネット 著 • <u>サーバ負荷分散技術</u> トニー ブルーク (著), Tony Bourke (原著), 鈴島 公章 (翻訳), 横山 晴庫 (翻訳), 上谷 → (翻訳) • <u>[24時間365日] サーバV/インフラを支える技術 ~スケーラビリティ、ハイパフォーマンス、省力運用</u> 伊藤直也, 勝見祐己, 田中慎司, ひろせまさあき, 安井真伸, 横川和哉 著
<p>仮想化</p>	<ul style="list-style-type: none"> • <u>Xen</u> • <u>KVM</u> • <u>QEMU</u>
<p>ロードバランシング (負荷分散)</p>	<ul style="list-style-type: none"> • <u>LVS</u> • <u>HAProxy</u>
<p>クラスタ管理</p>	<ul style="list-style-type: none"> • <u>Pacemaker</u> <ul style="list-style-type: none"> • <u>クラスタグループ、リソースエージェント</u> • <u>Red Hat Cluster Suite</u> <ul style="list-style-type: none"> • <u>OpenAIS</u> • <u>corosync</u>
<p>クラスタストレージ</p>	<ul style="list-style-type: none"> • <u>DRBD</u> • <u>GFS</u> • <u>OCFS2</u>



③技術解説

主題330: 仮想化



330.1 仮想化の概念と理論



330.1 仮想化の概念と理論

仮想化のメリット

- ・複数のサーバを1台の物理マシンに統合することができる。
→業務レベルでは、消費電力やサーバ設置のスペースなどを減らすことが出来、より多くのサーバを扱える。
個人レベルでは、1台の物理マシンで複数のサーバを保持可。
- ・ハードウェアに依存しないので、サーバの追加および廃棄が楽。
- ・サーバリソースの不足にも、サーバのコピー等で容易に対応可。
→サーバのインストールや設定などの管理も楽になる。
- ・過去のハードウェアを仮想化することにより、
最新のハードウェアで動作しないOSを動作させることが出来る。
など



330.1 仮想化の概念と理論

仮想化のデメリット

- ・物理サーバの障害が、格納するサーバ全体に影響を及ぼす。
⇒物理サーバのメンテナンスなどの際にも同様の事が考えられる
- ・特殊過ぎるハードウェアは利用出来ないことがある。
- ・物理マシン・仮想マシンの両方を管理する必要があり、
管理が大変。

など



330.1 仮想化の概念と理論

様々な仮想マシンモニター①(アプリケーションタイプ)

・・・Virtualbox、VMwarePlayerなど

アプリケーション

アプリケーション

ゲストOS

ゲストOS

仮想ハードウェア

仮想ハードウェア

仮想化ソフトウェア

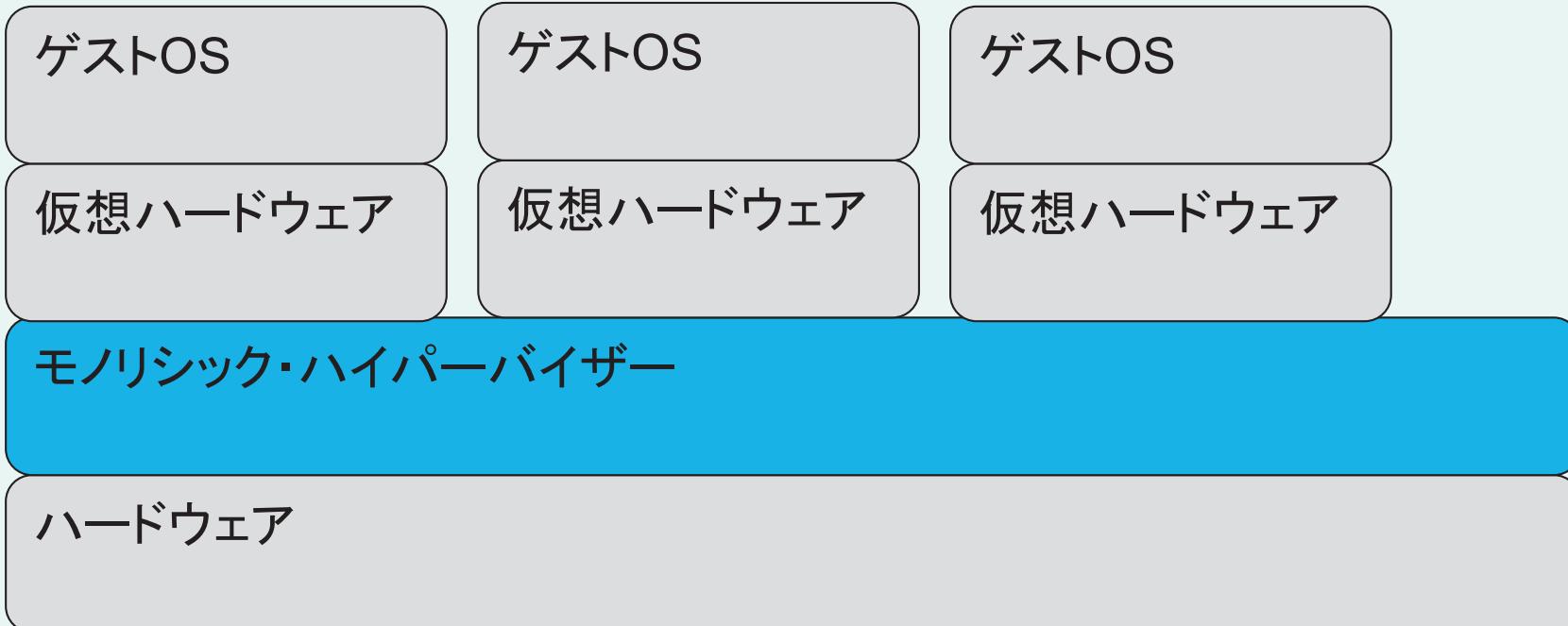
ホストOS

ハードウェア



330.1 仮想化の概念と理論

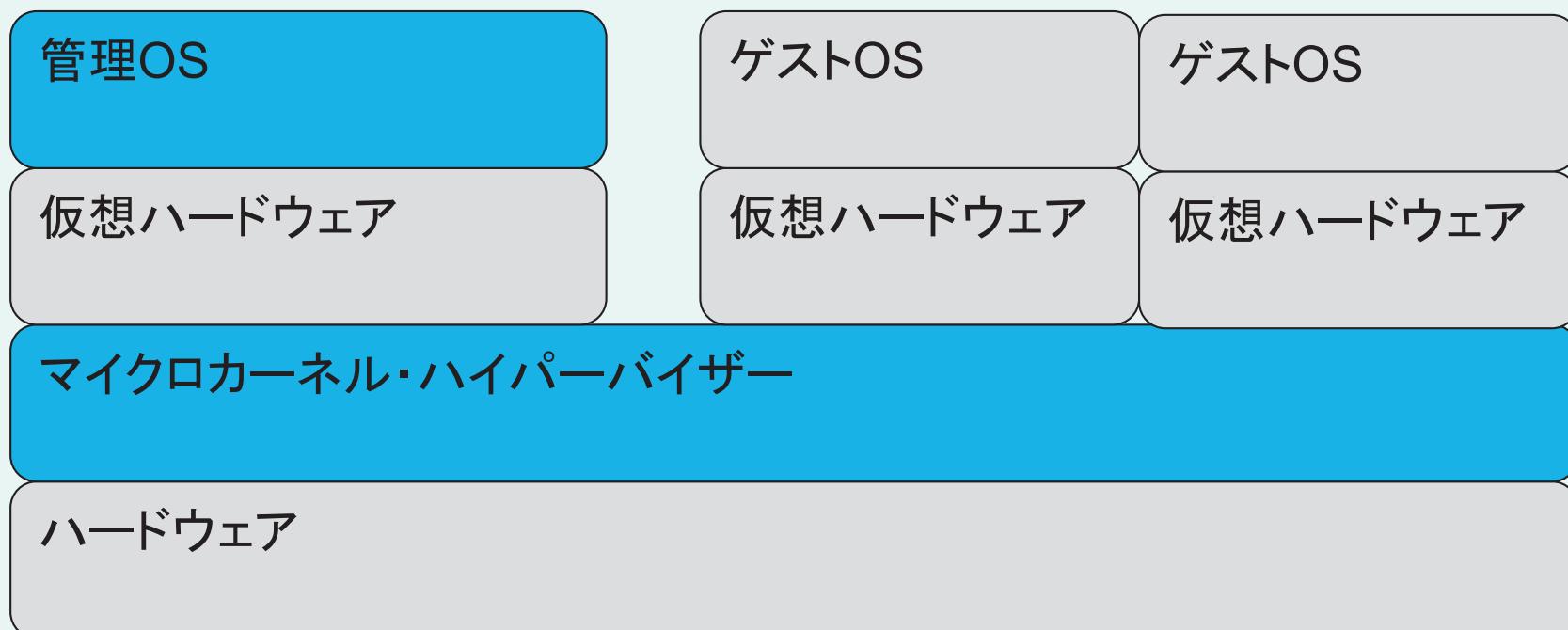
様々な仮想マシンモニター②(ハイパー・バイザータイプ1)
…KVMなど





330.1 仮想化の概念と理論

様々な仮想マシンモニター③(ハイパー・バイザータイプ2)
…Xen、XenServer、Microsoft Hyper-Vなど

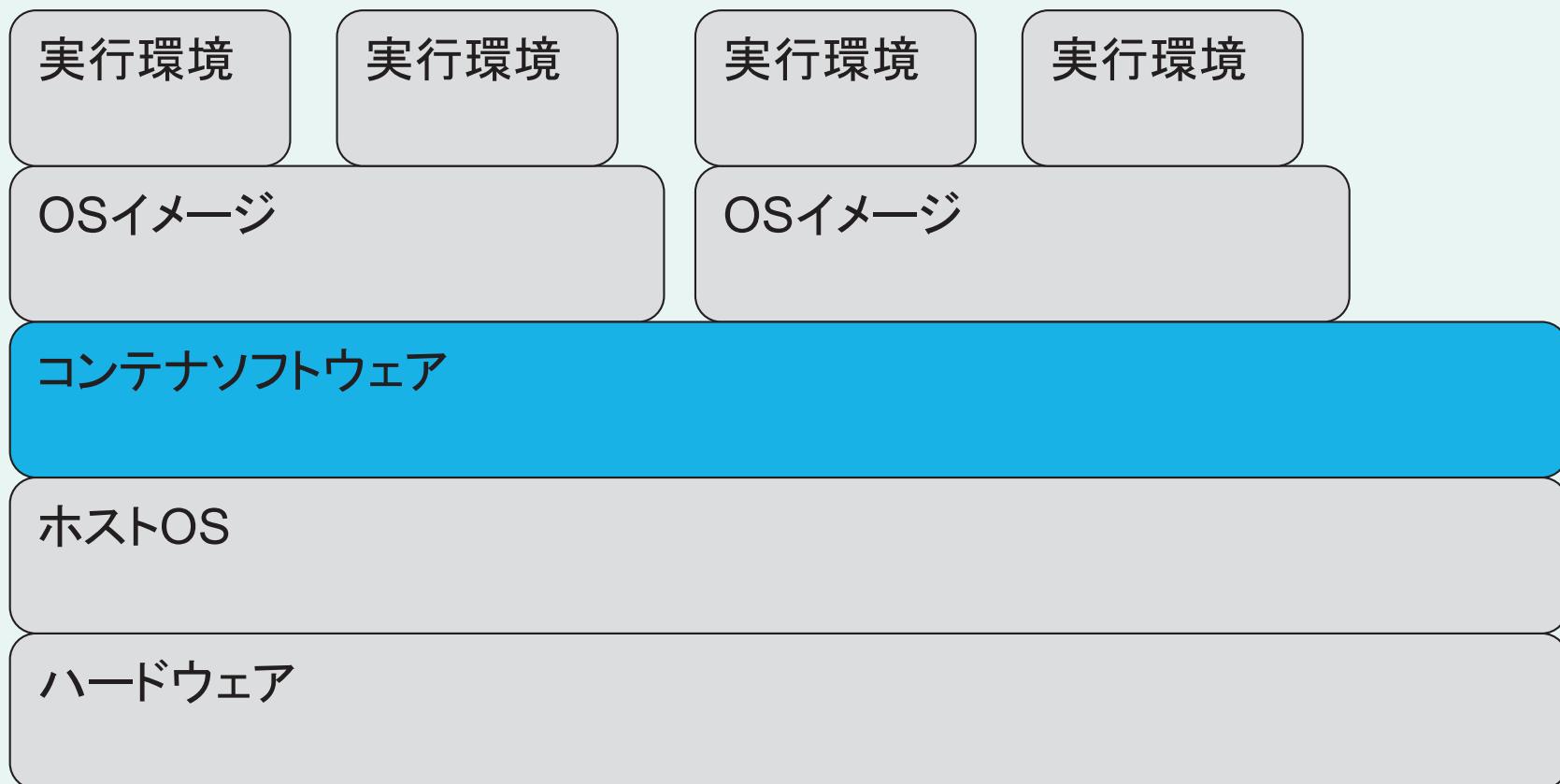




330.1 仮想化の概念と理論

様々な仮想マシンモニター④(コンテナタイプ)

…OpenVZ、LXC(LinuxContainer)、Dockerなど





330.1 仮想化の概念と理論

CPU flags

まず、XenやKVMなどの仮想化を利用するには、
PCのCPUにおいて、
仮想化支援機能が有効になっている必要があります。

/proc/cpuinfoにて行うことが確認出来、
Intel製のCPUの場合にはvmxというflagの有無、
AMD製のCPUの場合にはsvmというflagの有無を確認します。



330.1 仮想化の概念と理論

Intel製CPUの場合

```
[root@centos65forkvm ~]# cat /proc/cpuinfo
processor       : 0
vendor_id      : GenuineIntel
cpu family     : 6
model          : 58
model name     : Intel(R) Core(TM) i5-3470 CPU @ 3.20GHz
stepping        : 9
cpu MHz         : 3192.749
cache size      : 6144 KB
fpu             : yes
fpu_exception   : yes
cpuid level    : 13
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts mmx fxsr sse
                 sse2 ss syscall nx rdtscp lm constant_tsc arch_perfmon pebs bts xtopology tsc_reliable nonstop_tsc aperfmpfperf unf
                 air_spinlock pnpi pclmulqdq vmx ssse3 cx16 pcid sse4_1 sse4_2 x2apic popcnt aes xsave avx f16c rdrand hypervisor la
                 hf_lm ida arat epb xsaveopt pln pts dts tpr_shadow vnmi ept vpid fsgsbase smep
bogomips        : 6385.49
clflush size    : 64
cache_alignment : 64
address sizes   : 40 bits physical, 48 bits virtual
power management:
```



330.1 仮想化の概念と理論

AMD製CPUの場合

```
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)
[root@centos65forKVM ~]# cat /proc/cpuinfo
processor       : 0
vendor_id      : AuthenticAMD
cpu family     : 16
model          : 10
model name     : AMD Phenom(tm) II X6 1090T Processor
stepping        : 0
cpu MHz         : 3206.365
cache size     : 512 KB
fpu             : yes
fpu_exception   : yes
cpuid level    : 6
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
pat pse36 clflush mmx fxsr sse sse2 syscall nx mmxext fxsr_opt pdpe1gb rdtscp lm
3dnowext 3dnow constant_tsc rep_good tsc_reliable nonstop_tsc aperfmpfperf unfair
_spinlock pni cx16 x2apic popcnt hypervisor lahf_lm svm extapic cr8_legacy abm s
se4a misalignsse 3dnowprefetch osvw npt svm_lock nrip_save
bogomips       : 6412.73
TLB size        : 1024 4K pages
clflush size    : 64
cache_alignment : 64
address sizes   : 40 bits physical, 48 bits virtual
power management:
```

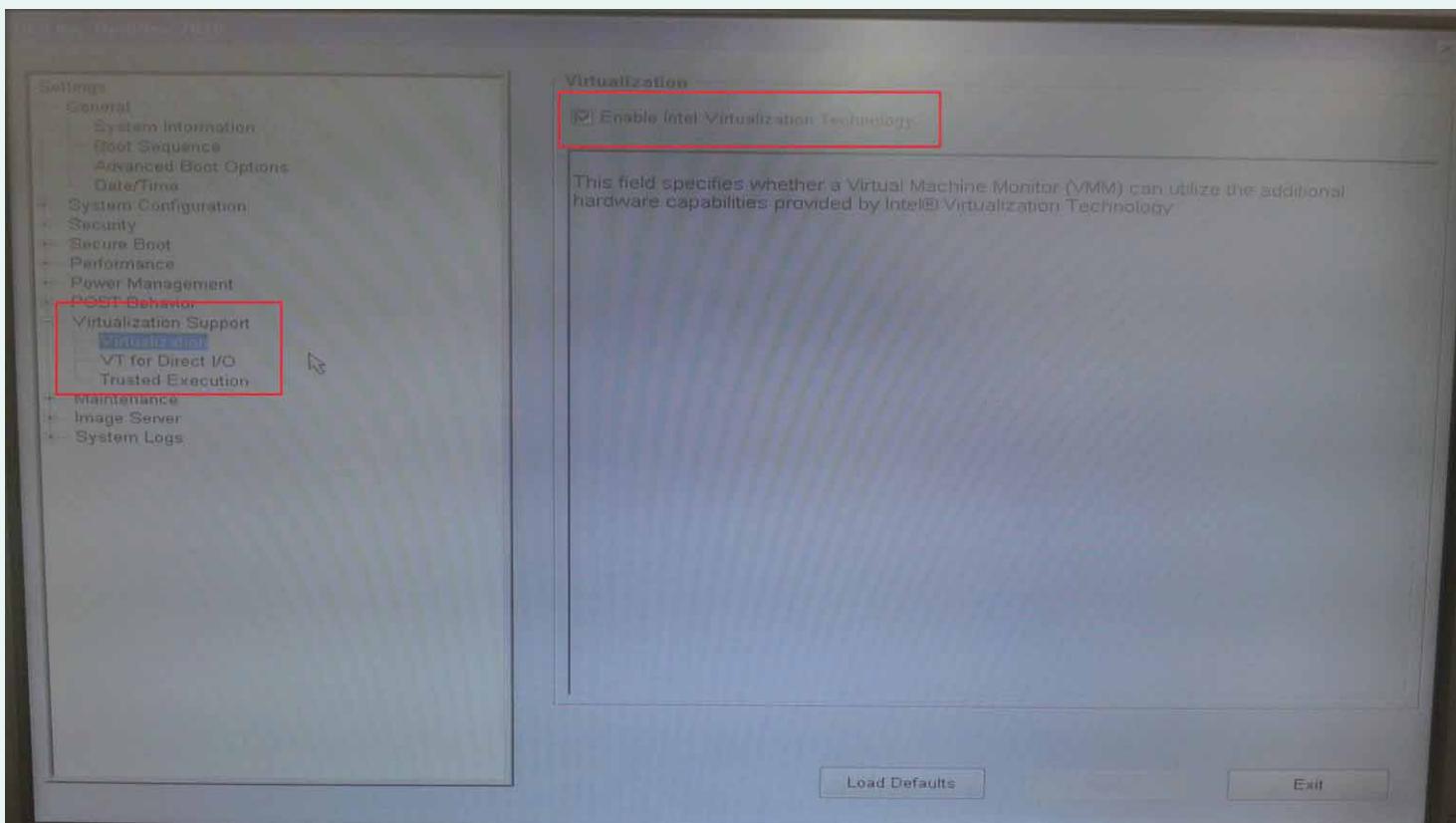


330.1 仮想化の概念と理論

vmxやsvmが確認出来ない場合、

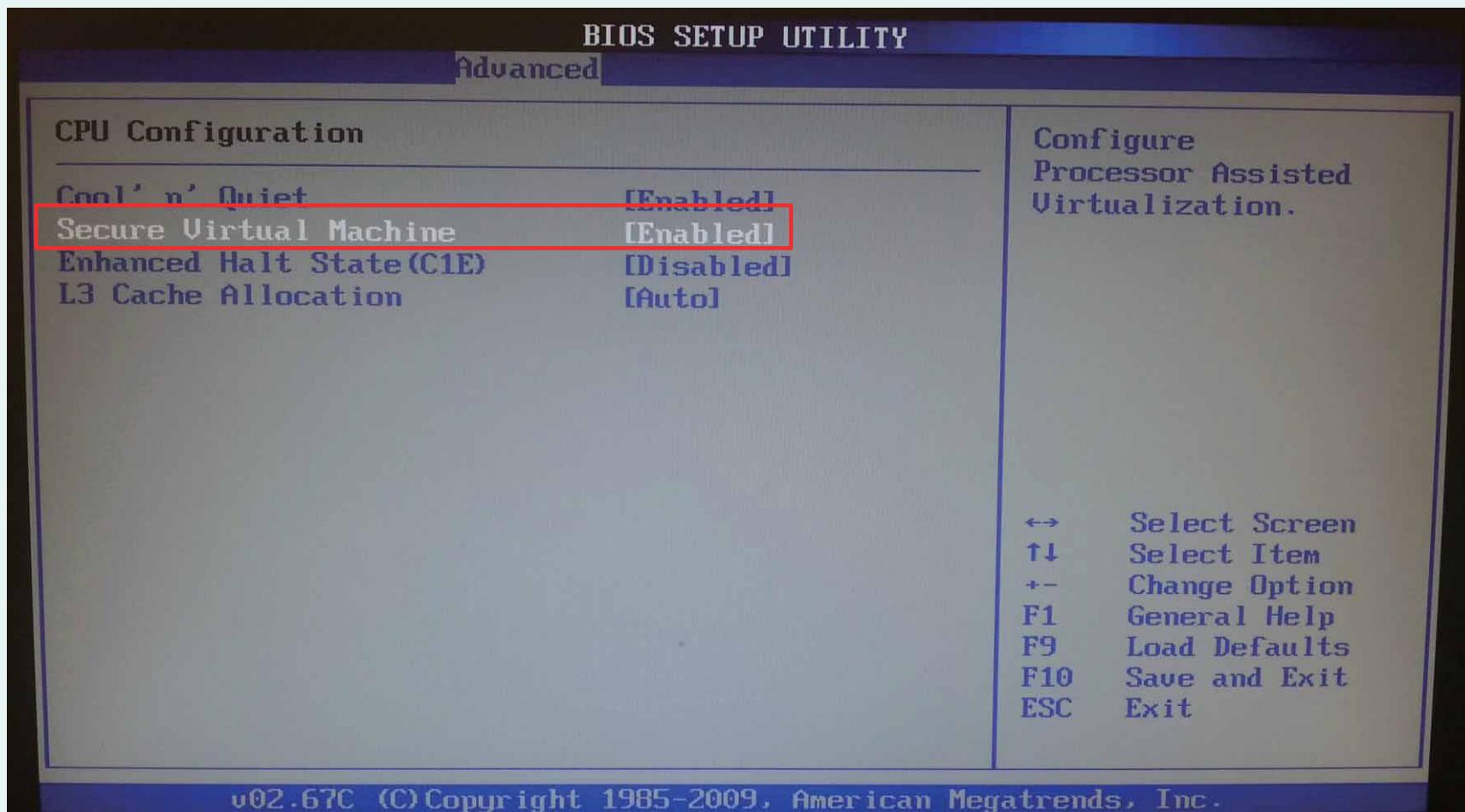
PCのBIOSやEFIなどで仮想化機能が有効化されているか確認し、無効であれば有効にします。

※VirtualizationやVirtualization Support、Secure Virtual Machineなどという項目です。





330.1 仮想化の概念と理論





330.1 仮想化の概念と理論

完全仮想化(Full Virtualization)

ゲストOSのソース修正を伴わず、ハードウェアを完全に仮想化する方式
Intel-VTやAMD-VなどのCPU仮想化機能を利用する
[Xen/KMVで対応]

準仮想化(Para Virtualization)

ゲストOSを一部修正処理性能を向上させる方式
[Xenで対応]

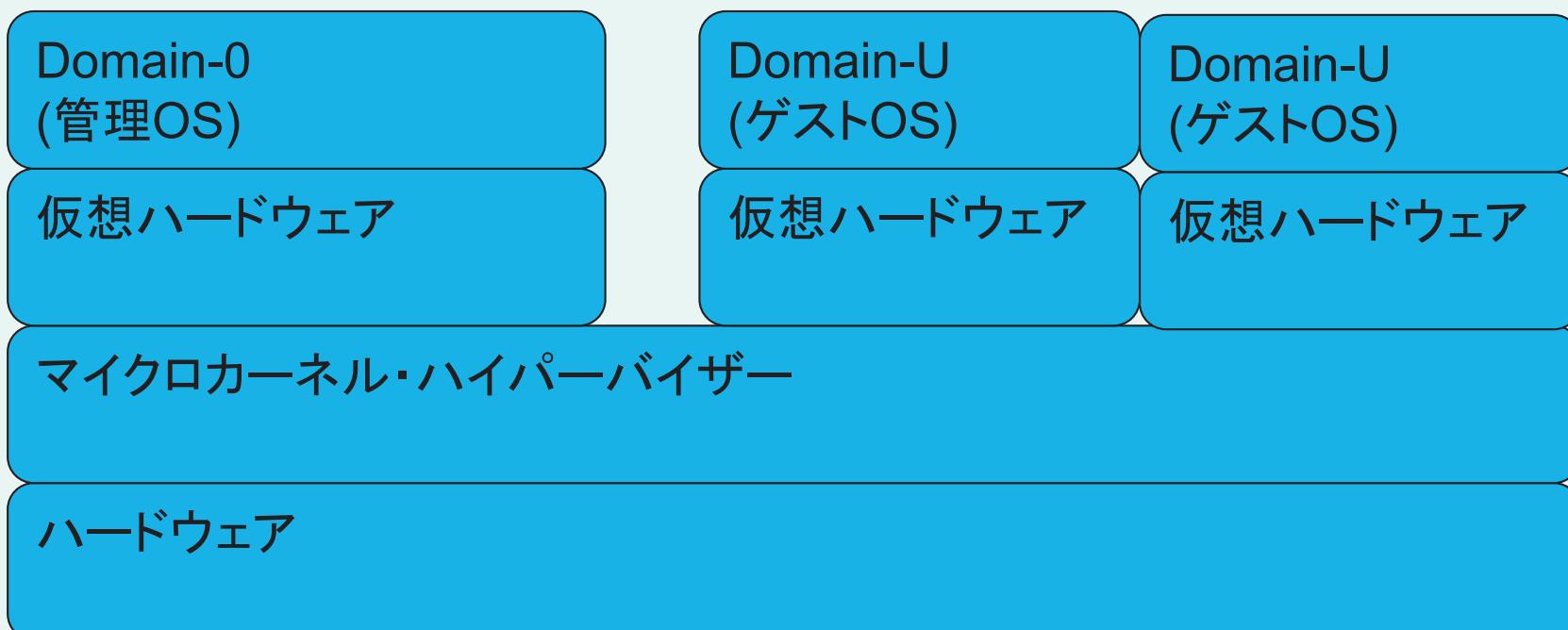


330.2 Xen



330.2 Xen(Xenのインストール)

Xenとは、ハイパーバイザータイプの仮想化を実現するものです。
また、管理OSの事をDomain-0、
ゲストOSの事をDomain-Uと呼びます。





330.2 Xen(Xenのインストール)

レポジトリより、
CentOS-5.10(x86_64)の
ISOイメージを取得します。

※1/7現在では、
5.11が最新です。

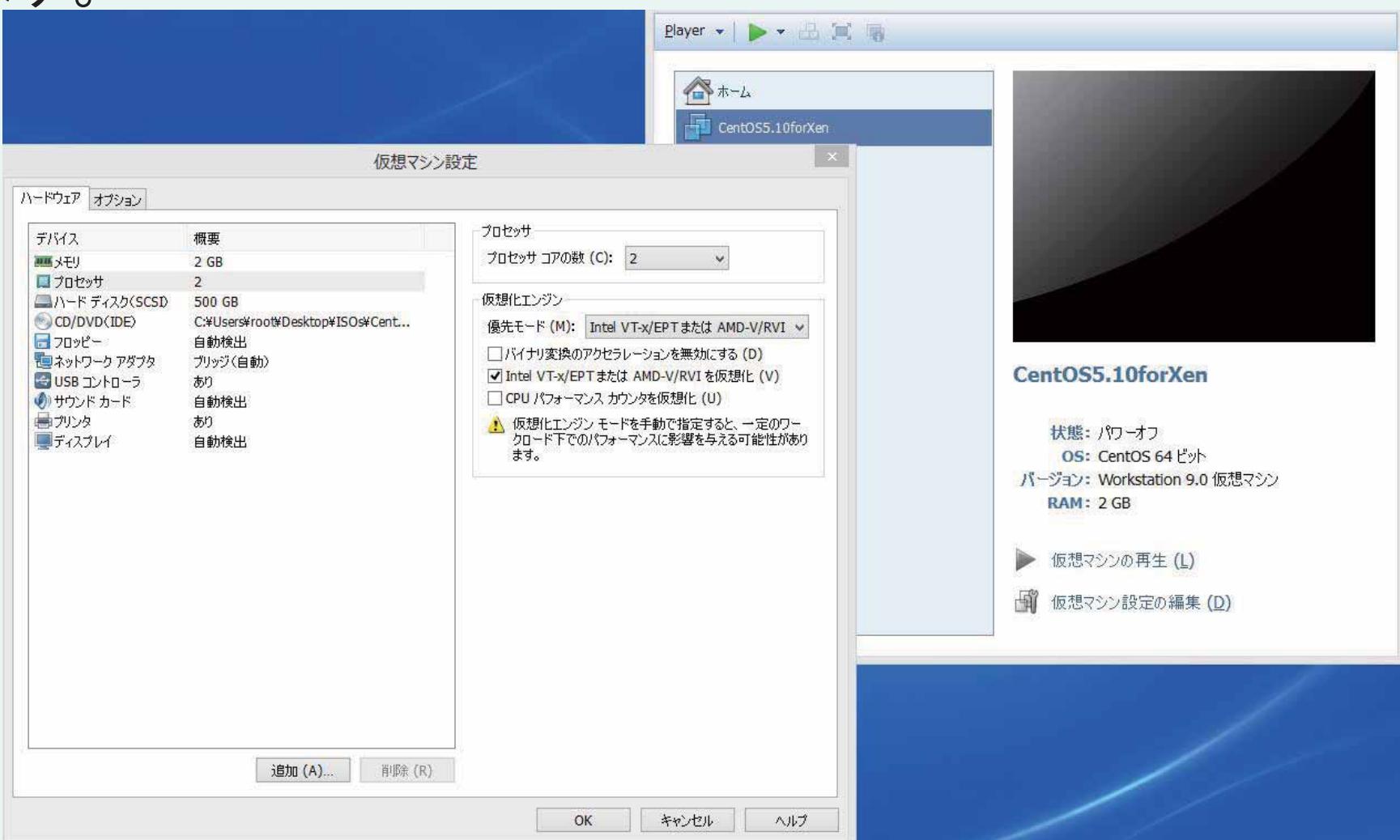
Index of /pub/Linux/centos/5.10/isos/x86_64

Name	Last modified	Size	Description
Parent Directory		-	
CentOS-5.10-x86_64-bin-1of9.iso	15-Oct-2013 04:11	625M	
CentOS-5.10-x86_64-bin-1to9.torrent	24-Oct-2013 00:15	207K	
CentOS-5.10-x86_64-bin-1to9.torrent.md5sum	24-Oct-2013 00:17	70	
CentOS-5.10-x86_64-bin-1to9.torrent.md5sum.asc	24-Oct-2013 00:21	944	
CentOS-5.10-x86_64-bin-1to9.torrent.sha1sum	24-Oct-2013 00:17	78	
CentOS-5.10-x86_64-bin-1to9.torrent.sha1sum.asc	24-Oct-2013 00:21	952	
CentOS-5.10-x86_64-bin-1to9.torrent.sha256sum	24-Oct-2013 00:17	102	
CentOS-5.10-x86_64-bin-1to9.torrent.sha256sum.asc	24-Oct-2013 00:21	1.0K	
CentOS-5.10-x86_64-bin-2of9.iso	15-Oct-2013 04:11	625M	
CentOS-5.10-x86_64-bin-3of9.iso	15-Oct-2013 04:11	635M	
CentOS-5.10-x86_64-bin-4of9.iso	15-Oct-2013 04:11	635M	
CentOS-5.10-x86_64-bin-5of9.iso	15-Oct-2013 04:11	634M	
CentOS-5.10-x86_64-bin-6of9.iso	15-Oct-2013 04:11	619M	
CentOS-5.10-x86_64-bin-7of9.iso	15-Oct-2013 04:11	636M	
CentOS-5.10-x86_64-bin-8of9.iso	15-Oct-2013 04:11	634M	
CentOS-5.10-x86_64-bin-9of9.iso	15-Oct-2013 04:11	222M	
CentOS-5.10-x86_64-bin-DVD-1of2.iso	15-Oct-2013 04:05	4.4G	
CentOS-5.10-x86_64-bin-DVD-2of2.iso	15-Oct-2013 04:05	810M	
CentOS-5.10-x86_64-bin-DVD.torrent	24-Oct-2013 00:15	206K	
CentOS-5.10-x86_64-bin-DVD.torrent.md5sum	24-Oct-2013 00:17	69	
CentOS-5.10-x86_64-bin-DVD.torrent.md5sum.asc	24-Oct-2013 00:21	943	
CentOS-5.10-x86_64-bin-DVD.torrent.sha1sum	24-Oct-2013 00:17	77	
CentOS-5.10-x86_64-bin-DVD.torrent.sha1sum.asc	24-Oct-2013 00:21	951	
CentOS-5.10-x86_64-bin-DVD.torrent.sha256sum	24-Oct-2013 00:17	101	
CentOS-5.10-x86_64-bin-DVD.torrent.sha256sum.asc	24-Oct-2013 00:21	1.0K	
CentOS-5.10-x86_64-netinstall.iso	15-Oct-2013 03:59	15M	



330.2 Xen(Xenのインストール)

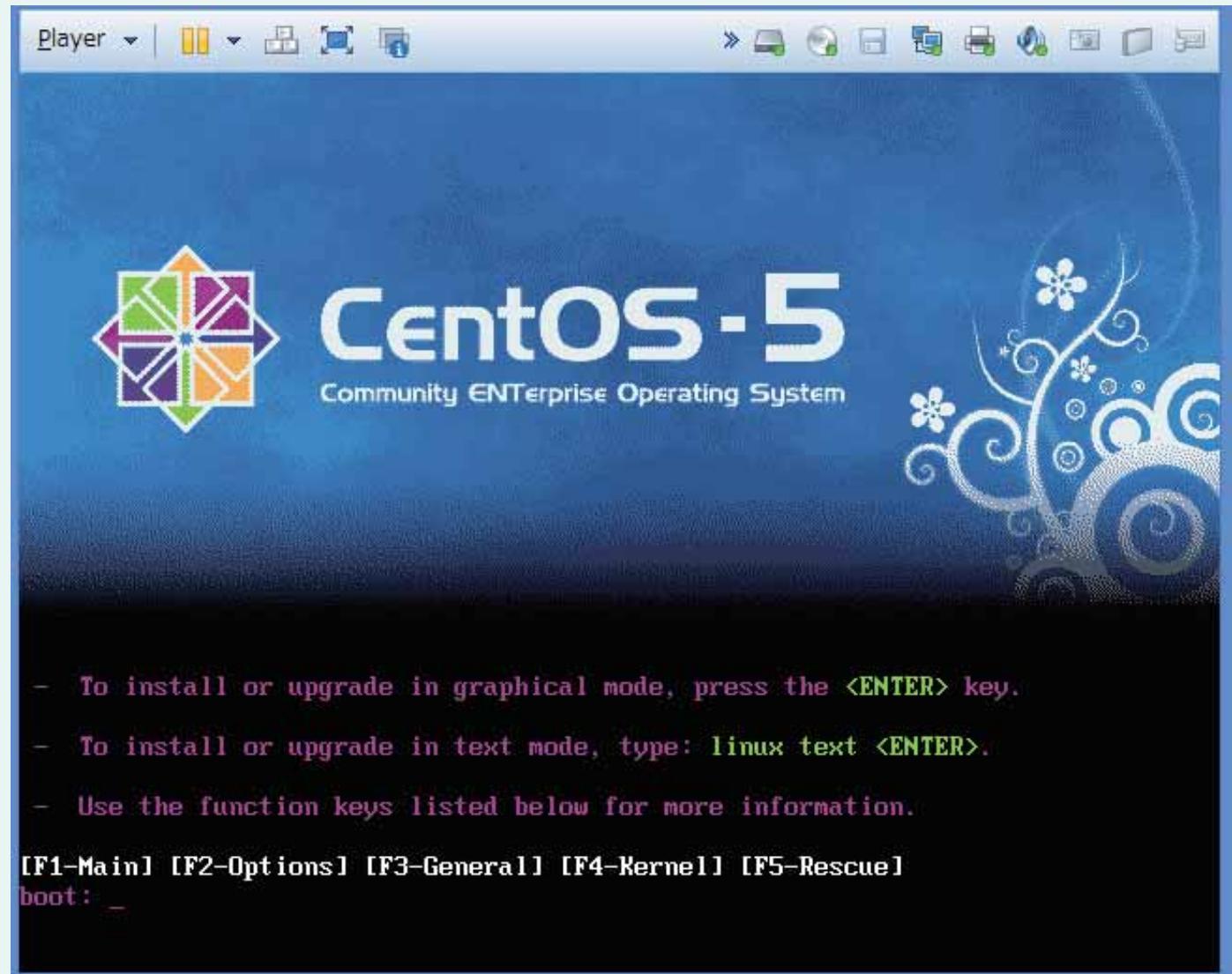
仮想化ソフトウェアにて、CPU仮想化支援機能を有効化した仮想PCを作成します。





330.2 Xen(Xenのインストール)

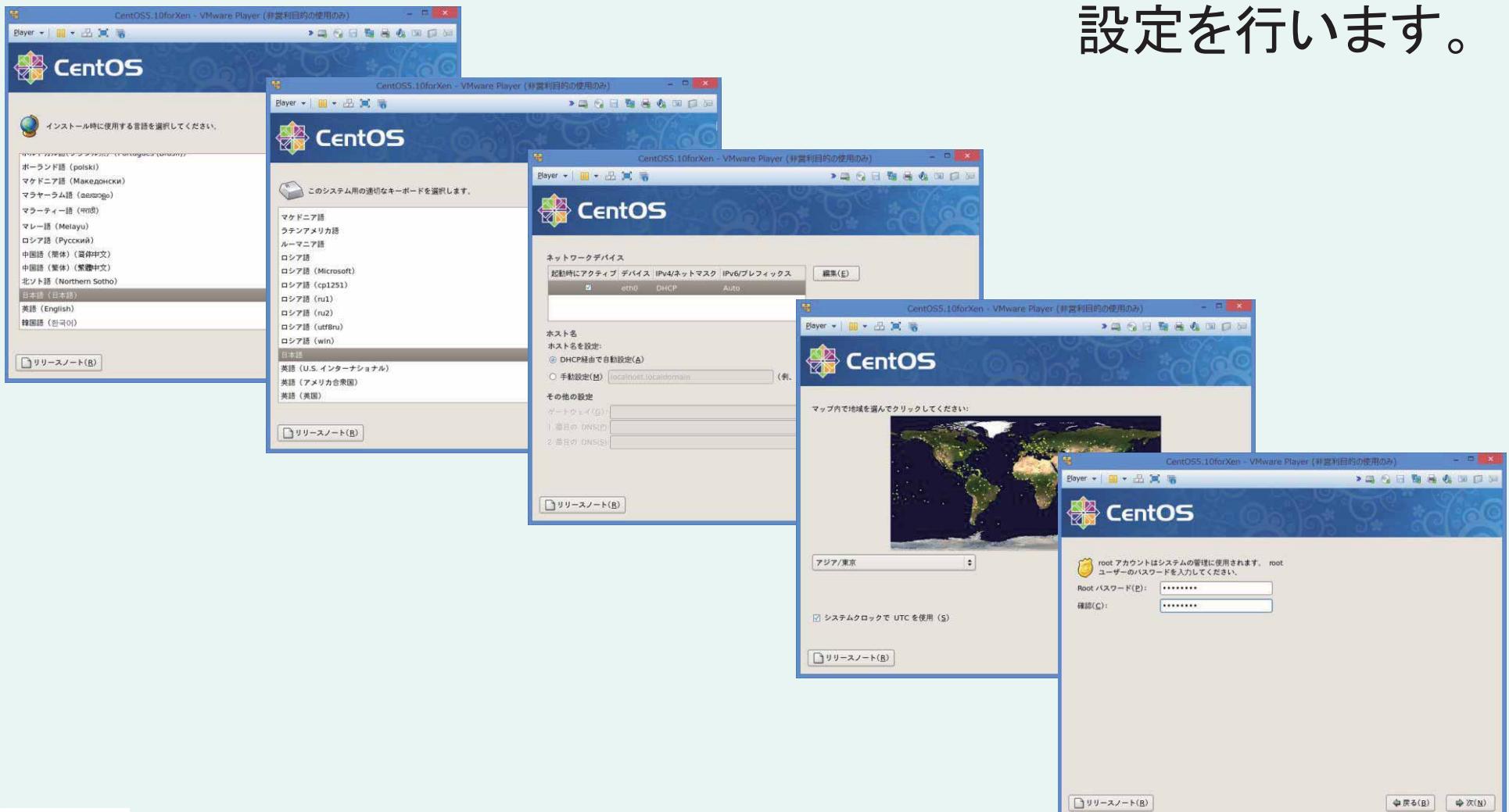
CentOS5.10(64bits)をインストールします。





330.2 Xen(Xenのインストール)

ソフトウェアの選択までは、自身の環境等に合わせて
設定を行います。





330.2 Xen(Xenのインストール)

ソフトウェアの選択にて、
「今すぐカスタマイズする」に
チェックを入れる。





330.2 Xen(Xenのインストール)

仮想化 > Xenに
チェックを入れる。





330.2 Xen(Xenのインストール)

仮想化 > Xenにより、
・libvirt
・virt-managerなどが
インストールされます。





330.2 Xen(Xenのインストール)

libvirtとは？

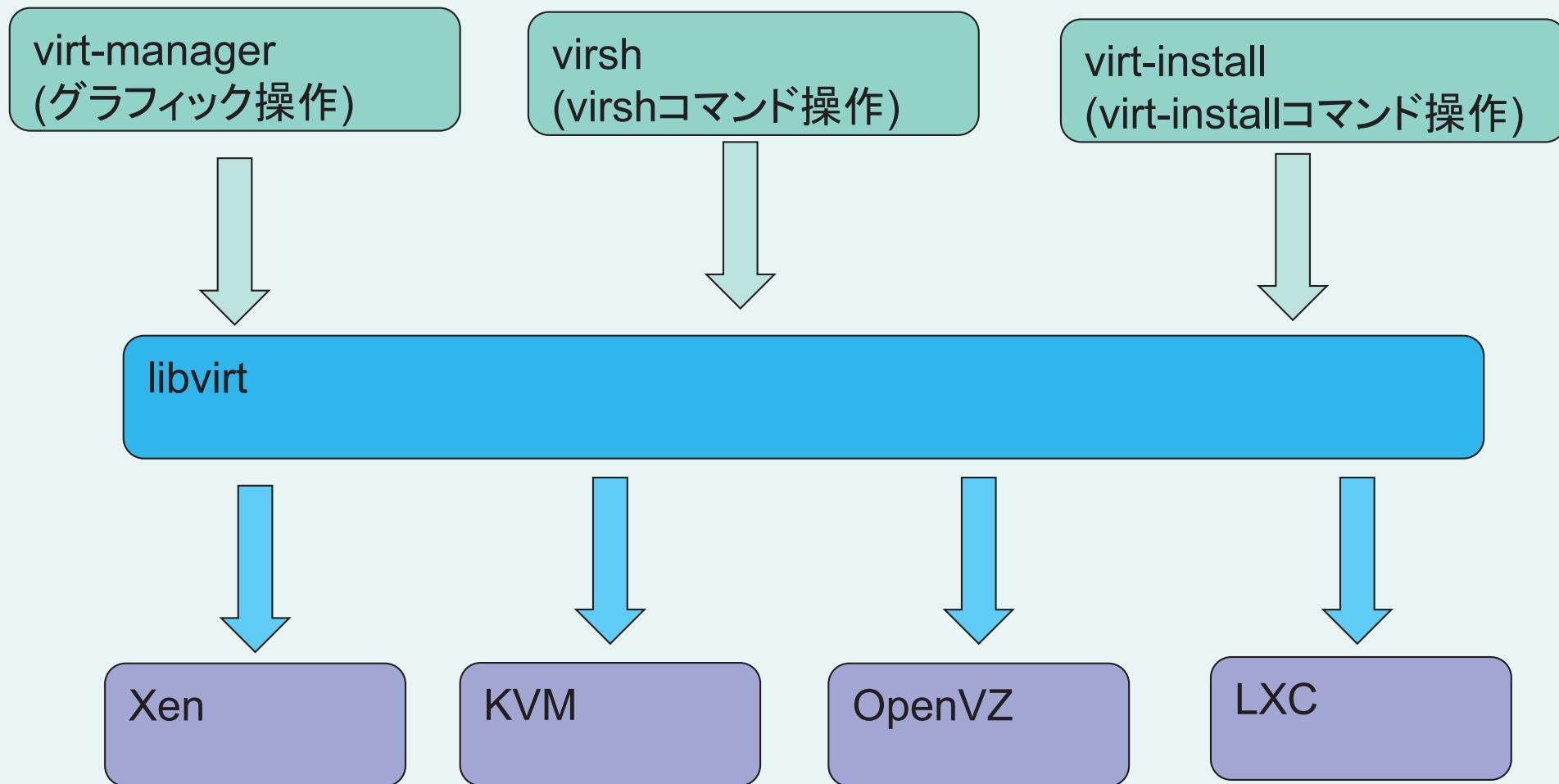
Xen、KVM、OpenVZ、LXCなどを制御するライブラリ

そして、libvirtは、

- ・virt-manager
- ・virshコマンド
- ・virt-installコマンド などからの操作を行う事が出来ます。



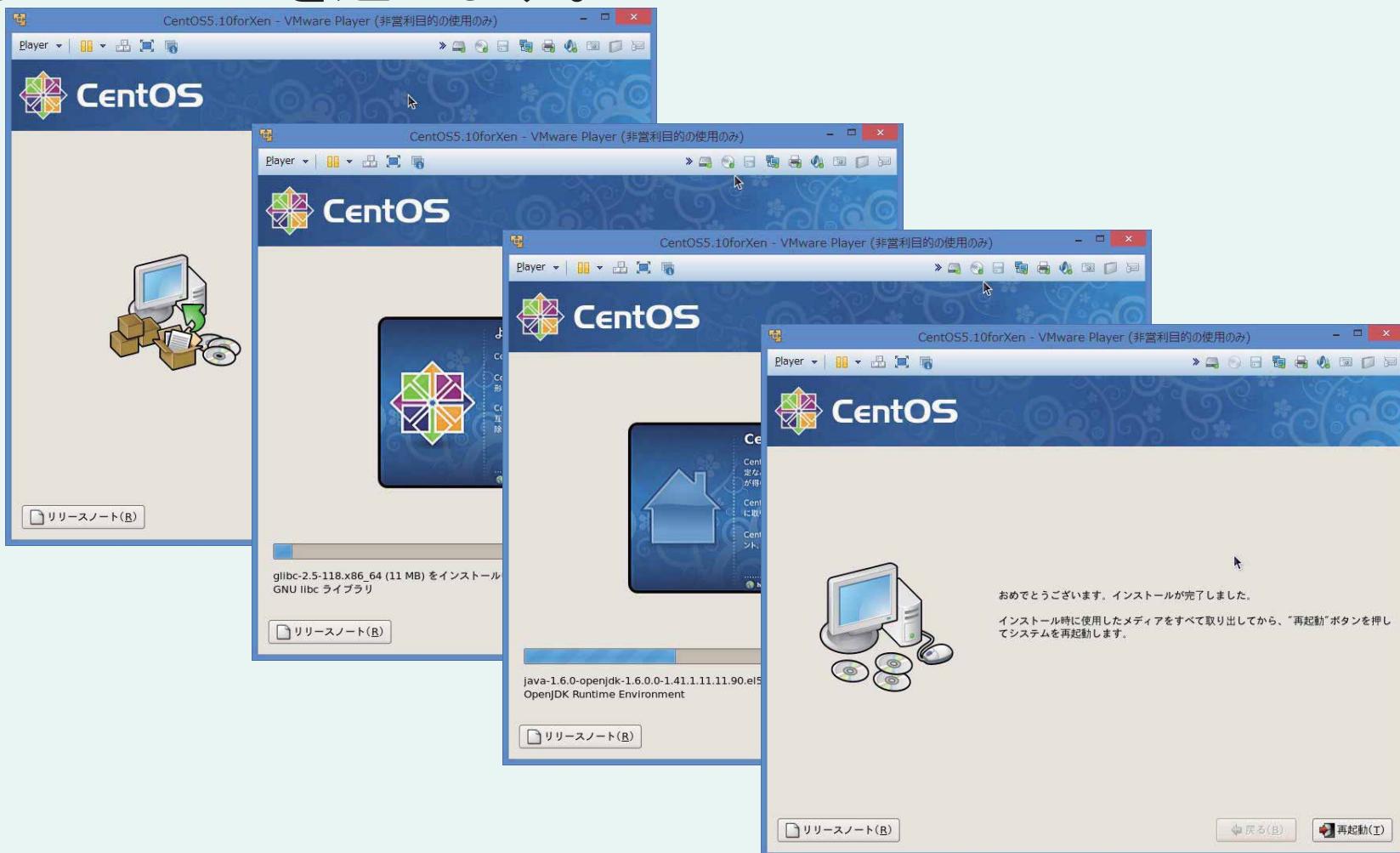
330.2 Xen(Xenのインストール)





330.2 Xen(Xenのインストール)

更にインストールを進めます。





LPIC 304 技術解説セミナー



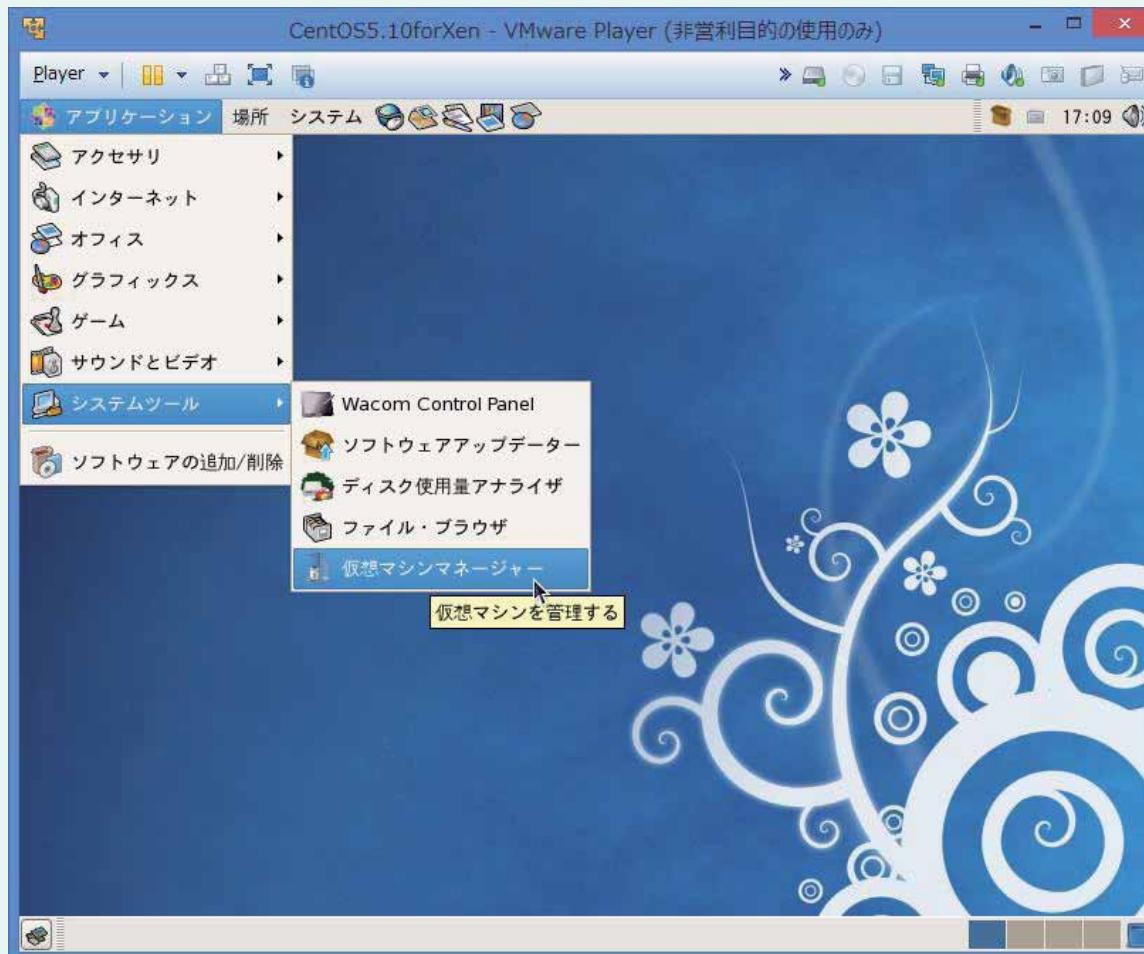
330.2 Xen(Xenのインストール)





330.2 Xen(Xenを使ってみる by virt-manager)

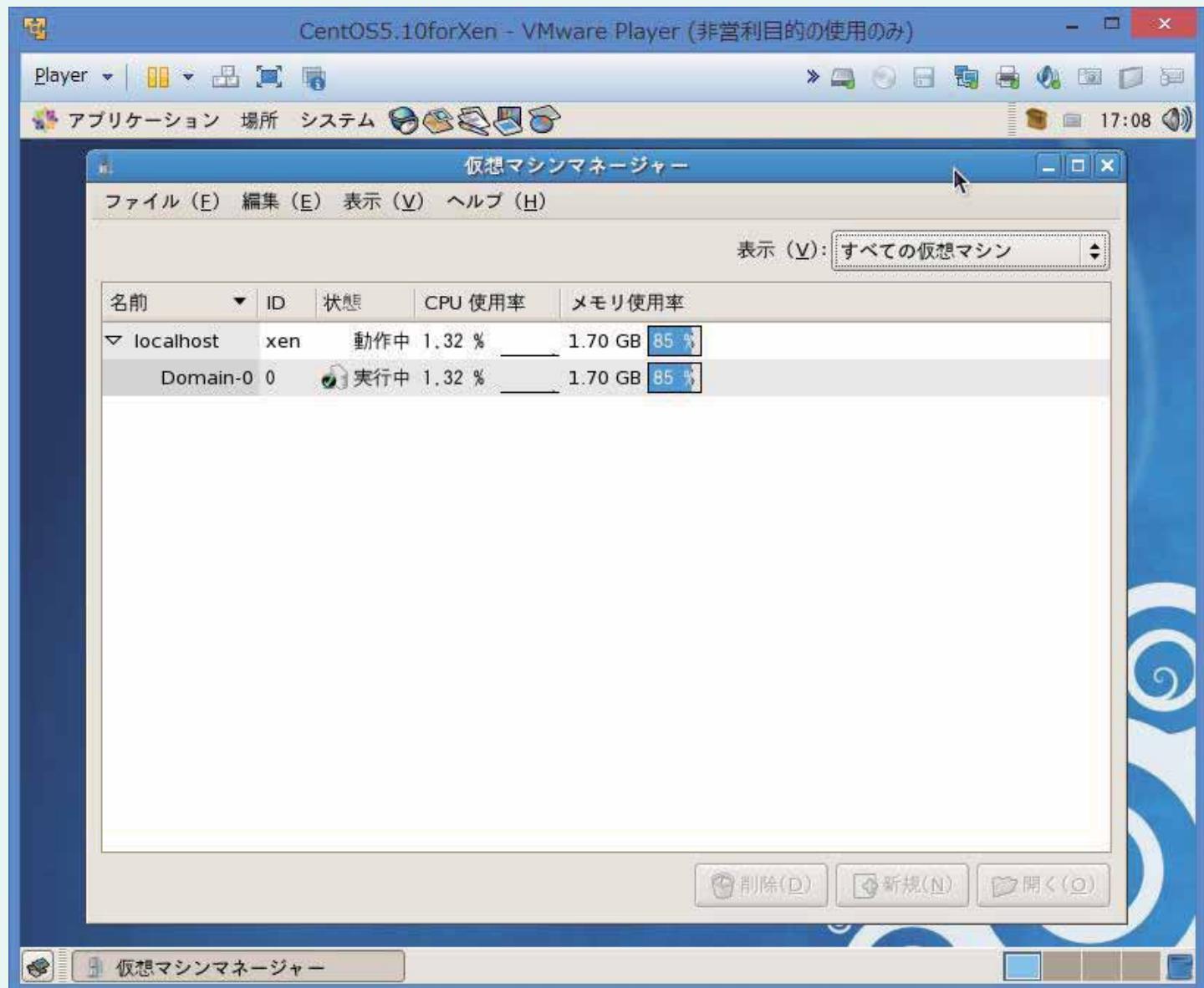
アプリケーション > システムツール > 仮想マシンマネージャーから
virt-managerを起動します。





330.2 Xen(Xenを使ってみる)

初期状態では、
Domain-0(管理OS)
のみが起動

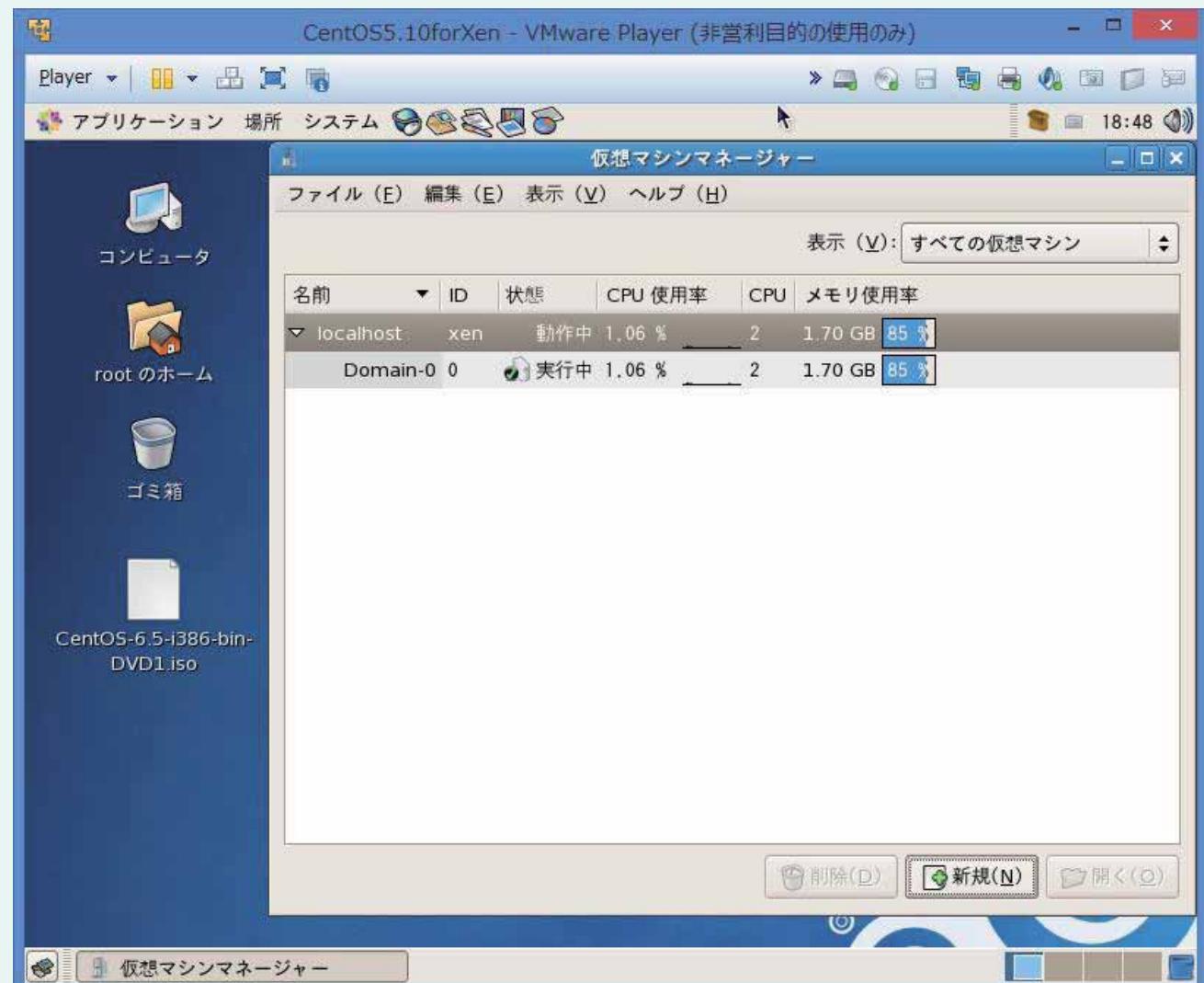




330.2 Xen(Xenを使ってみる)

Domain-U(ゲストOS)の作成

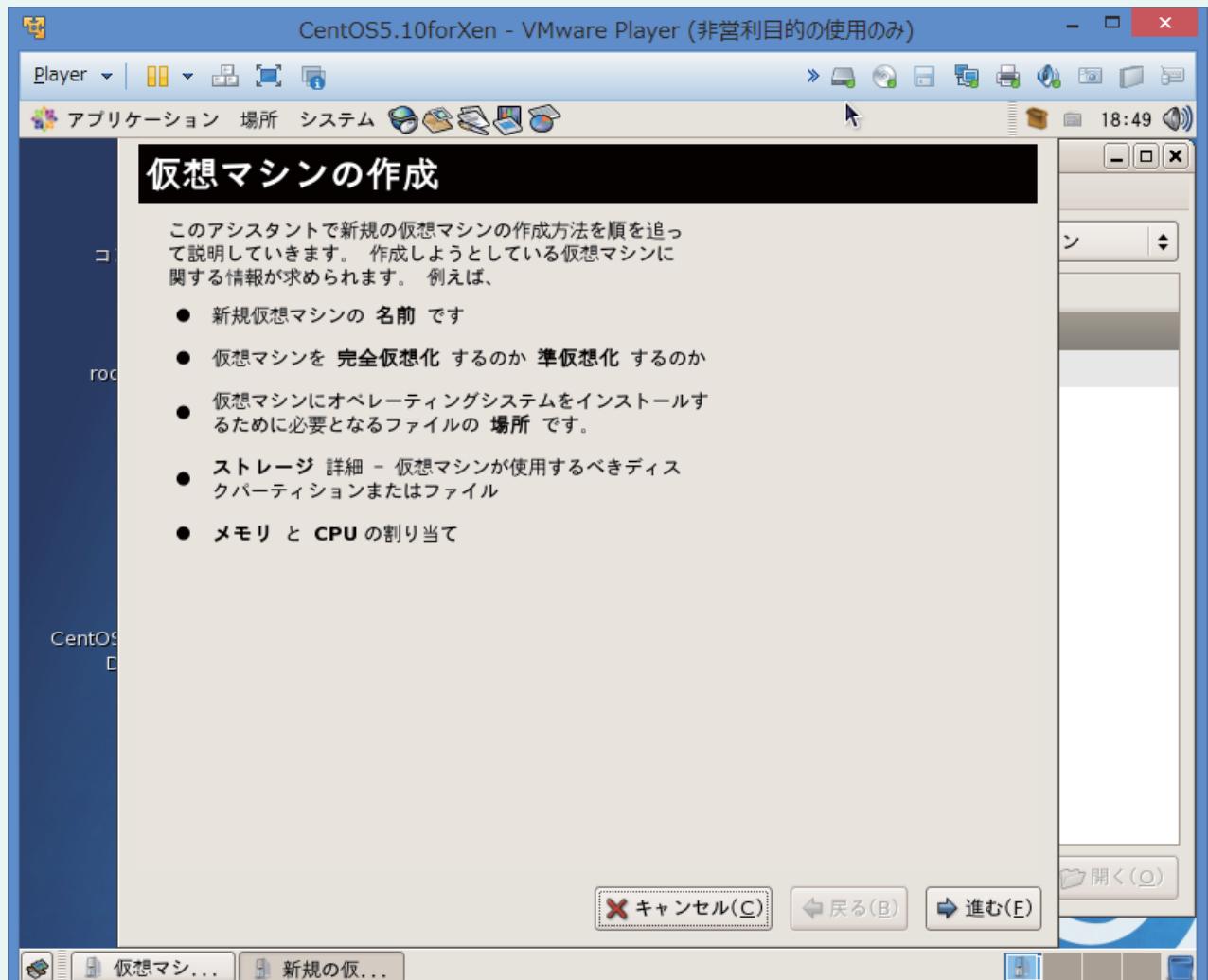
仮想マシンマネージャより
「新規」をクリック





330.2 Xen(Xenを使ってみる)

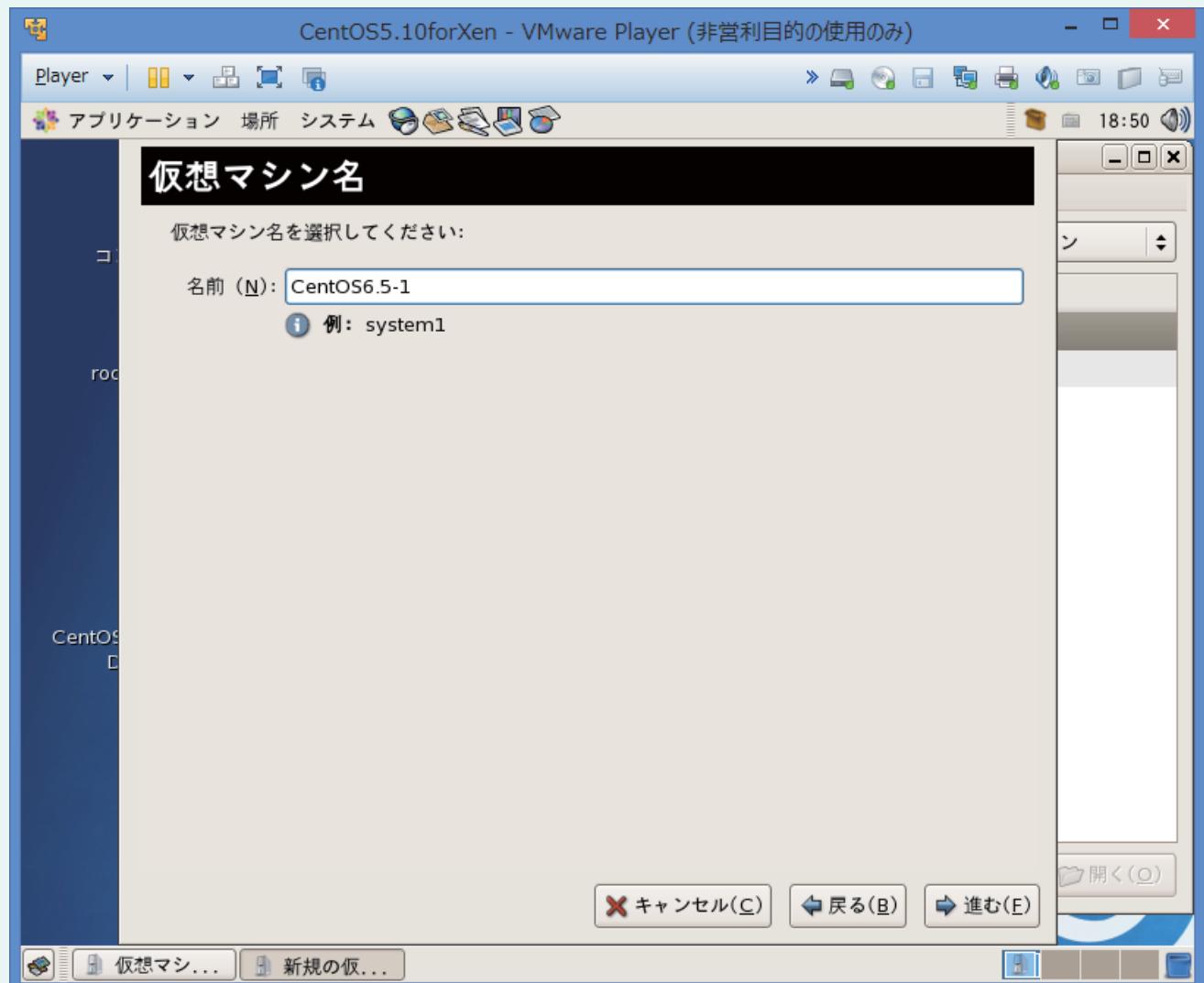
Domain-U(ゲストOS)の作成





330.2 Xen(Xenを使ってみる)

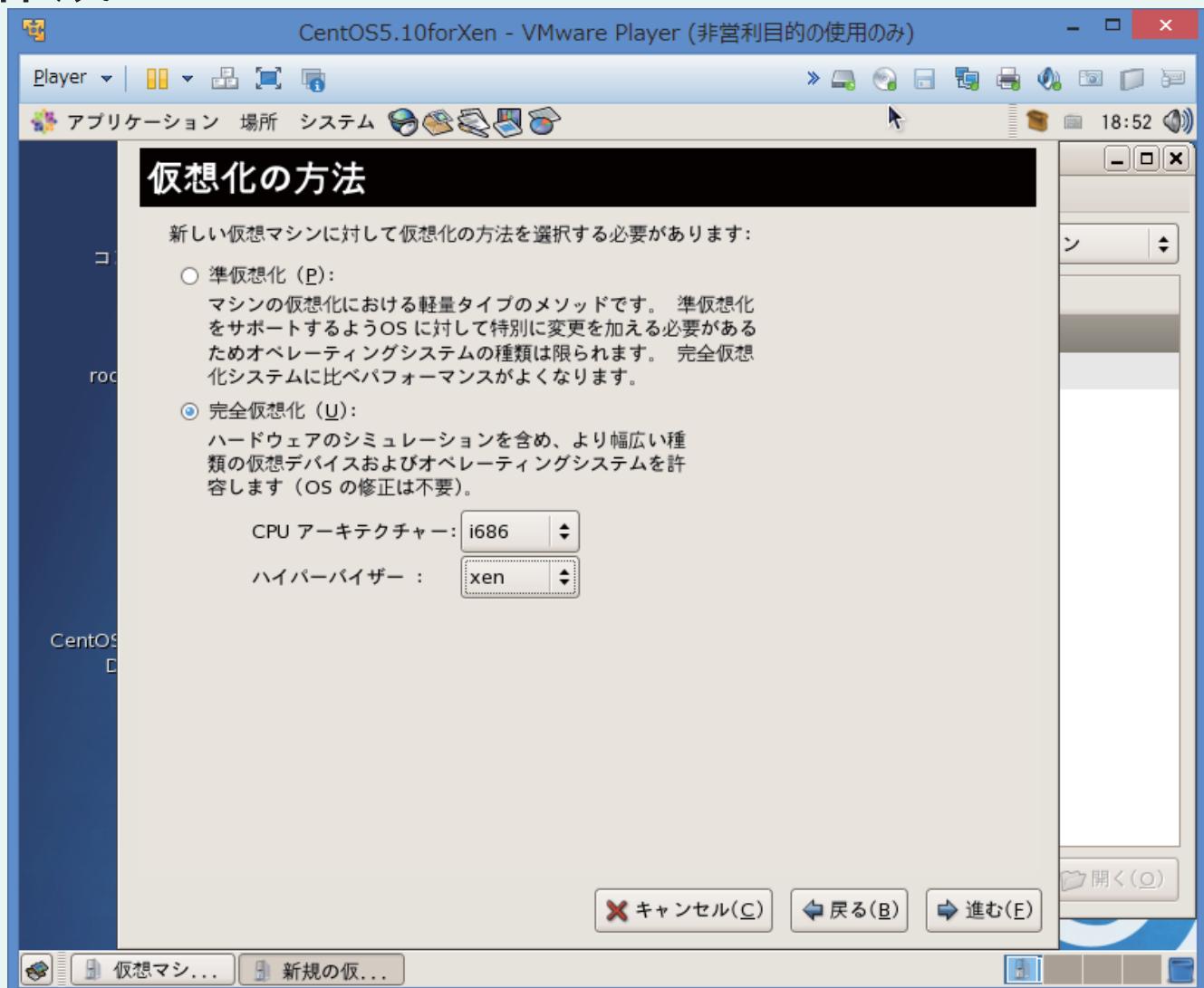
Domain-U(ゲストOS)の作成





330.2 Xen(Xenを使ってみる)

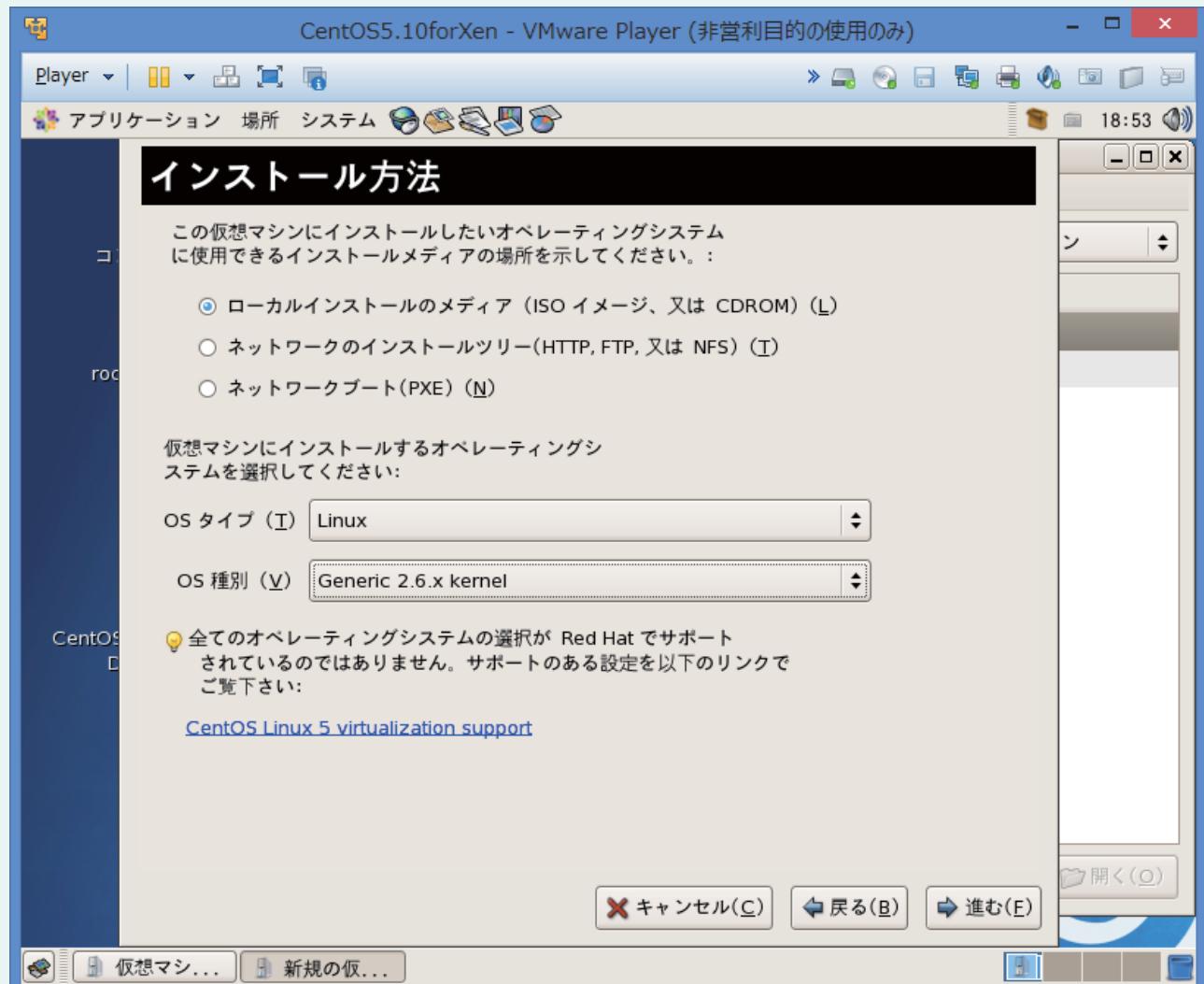
Domain-U(ゲストOS)の作成





330.2 Xen(Xenを使ってみる)

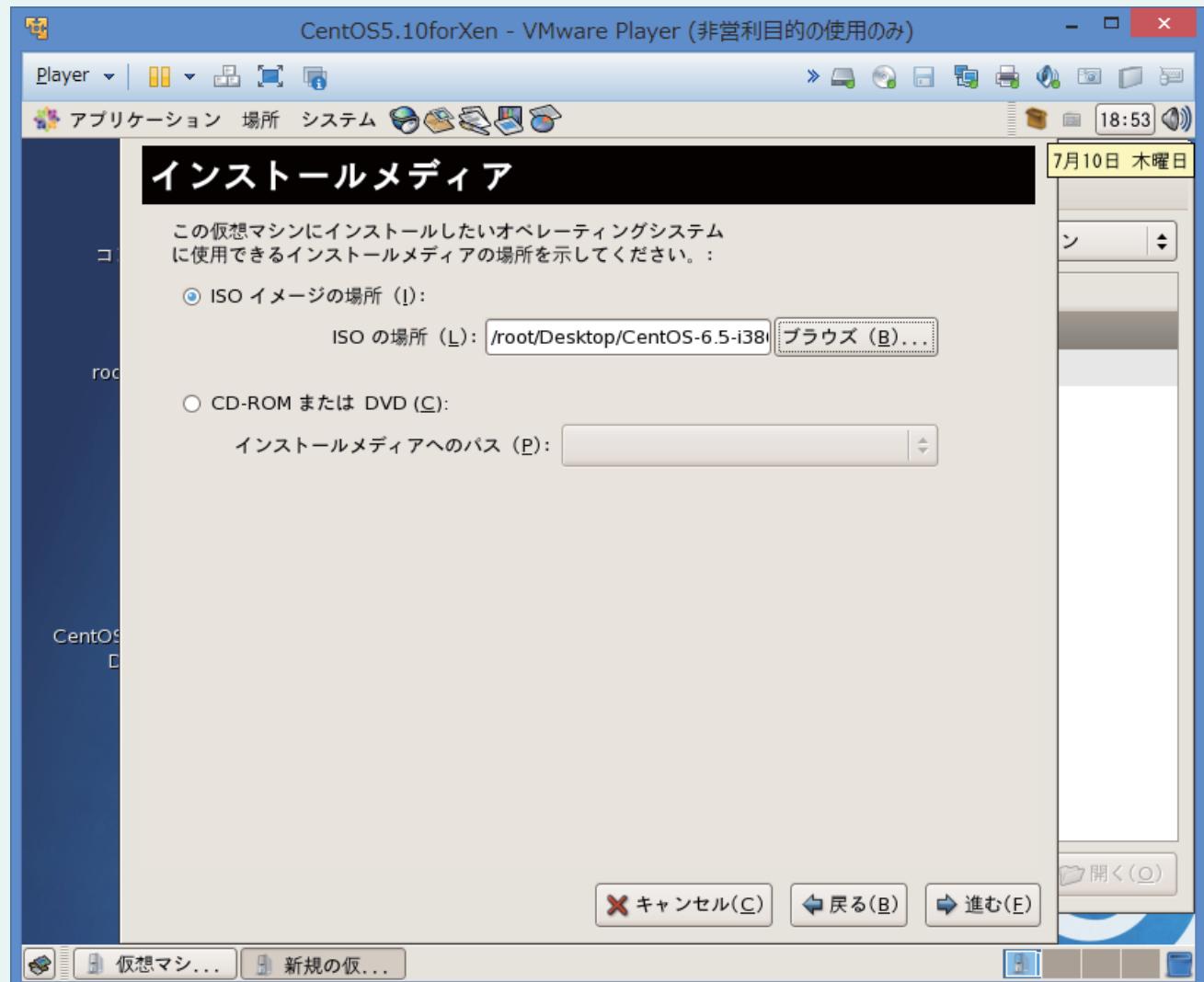
Domain-U(ゲストOS)の作成





330.2 Xen(Xenを使ってみる)

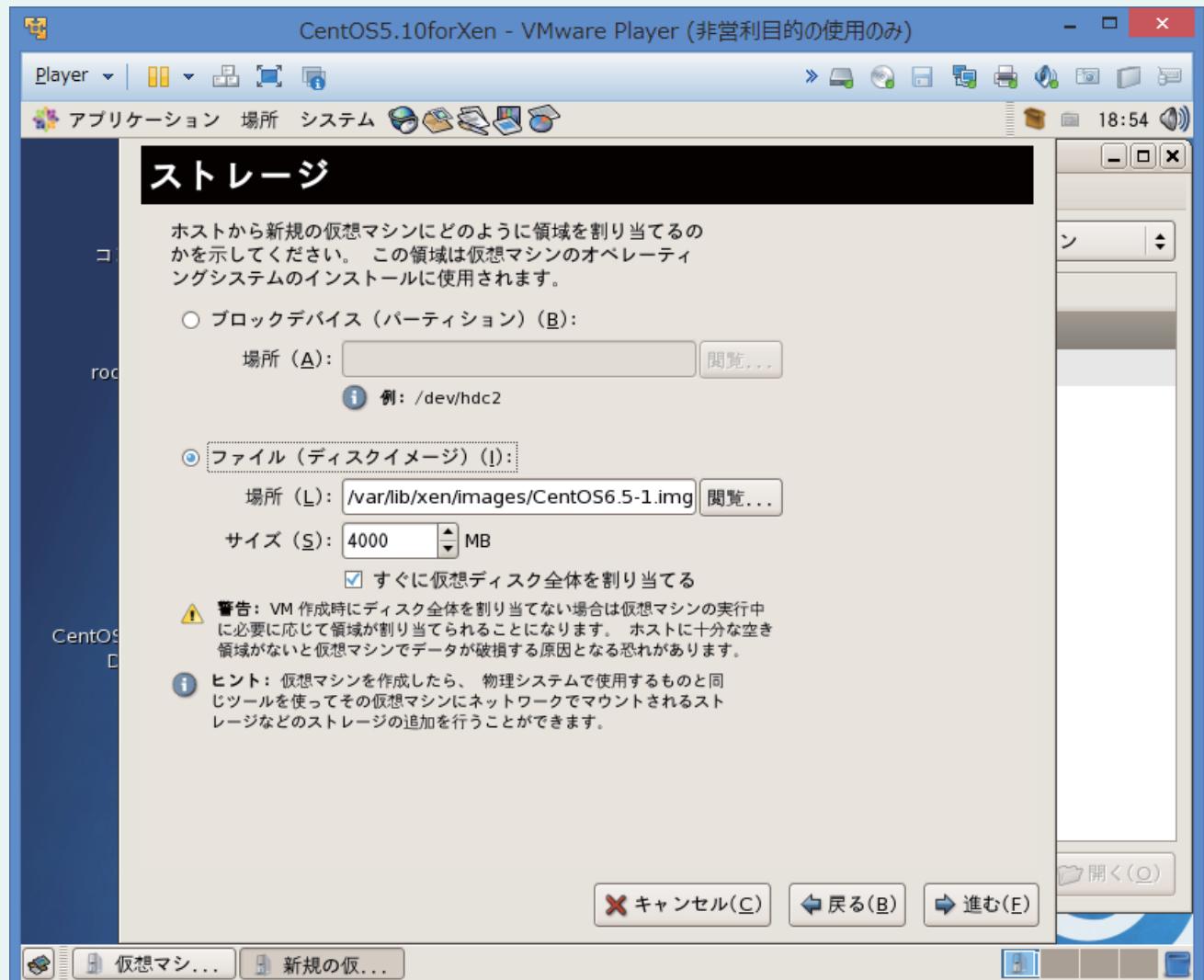
Domain-U(ゲストOS)の作成





330.2 Xen(Xenを使ってみる)

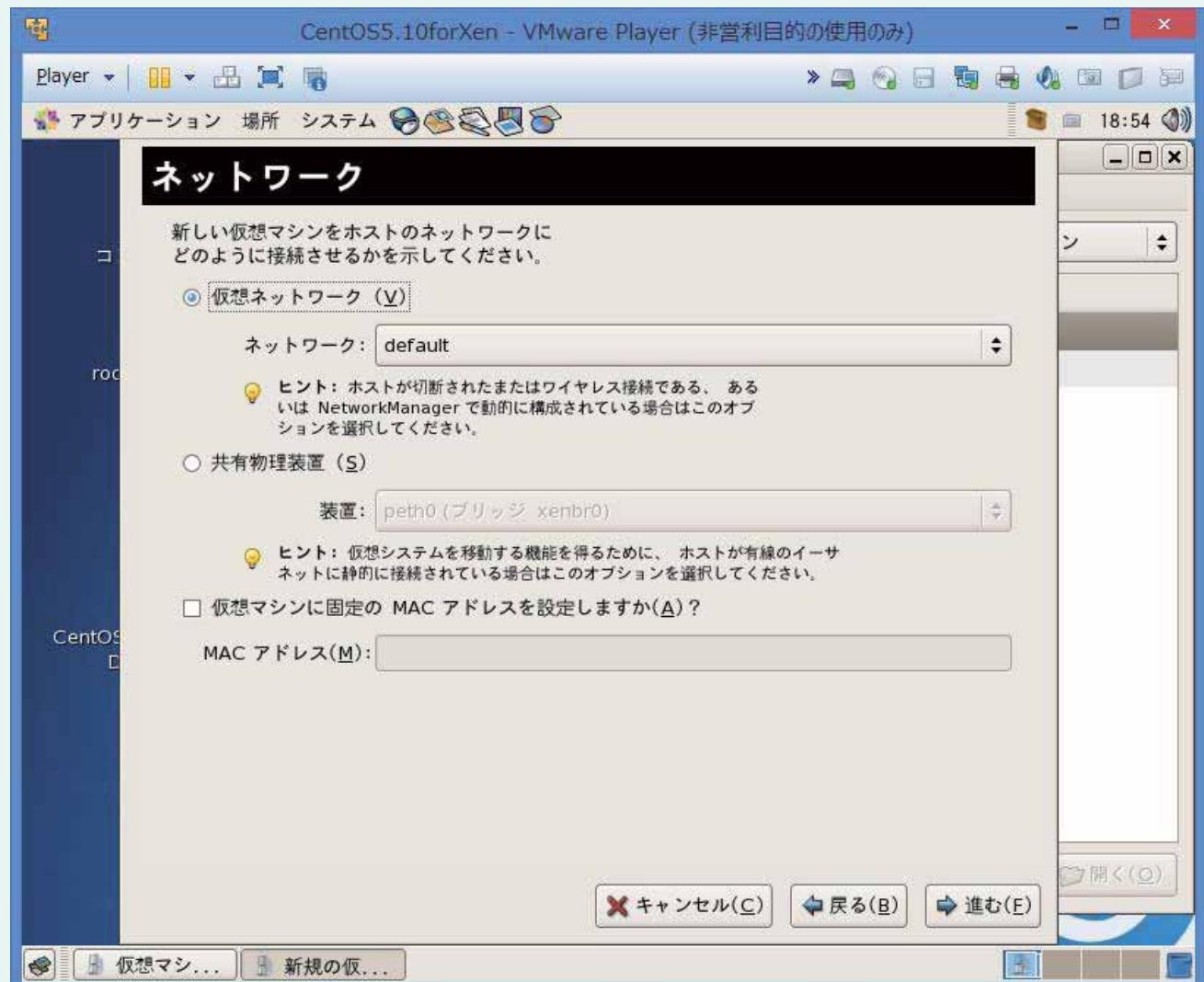
Domain-U(ゲストOS)の作成





330.2 Xen(Xenを使ってみる)

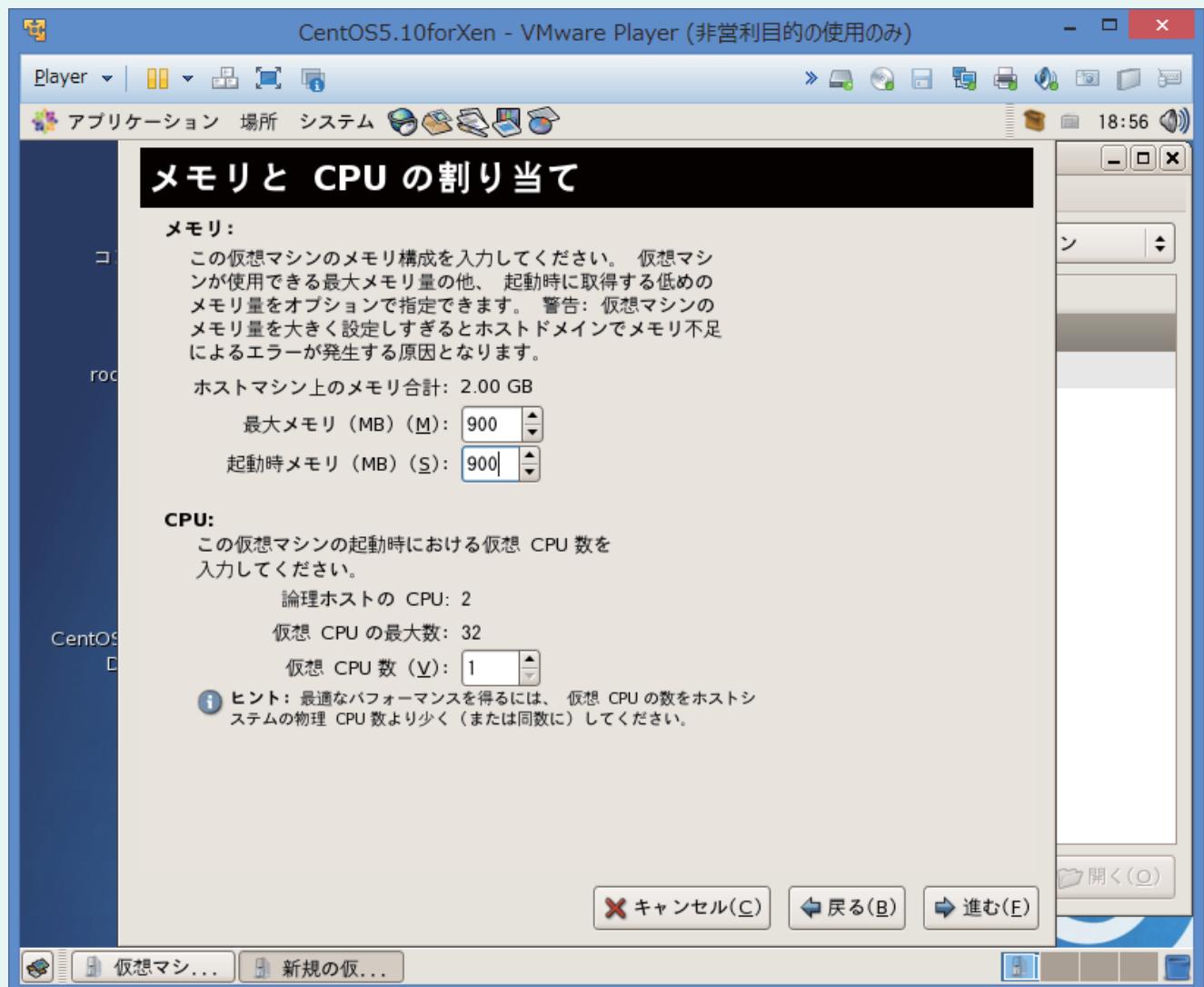
Domain-U(ゲストOS)の作成





330.2 Xen(Xenを使ってみる)

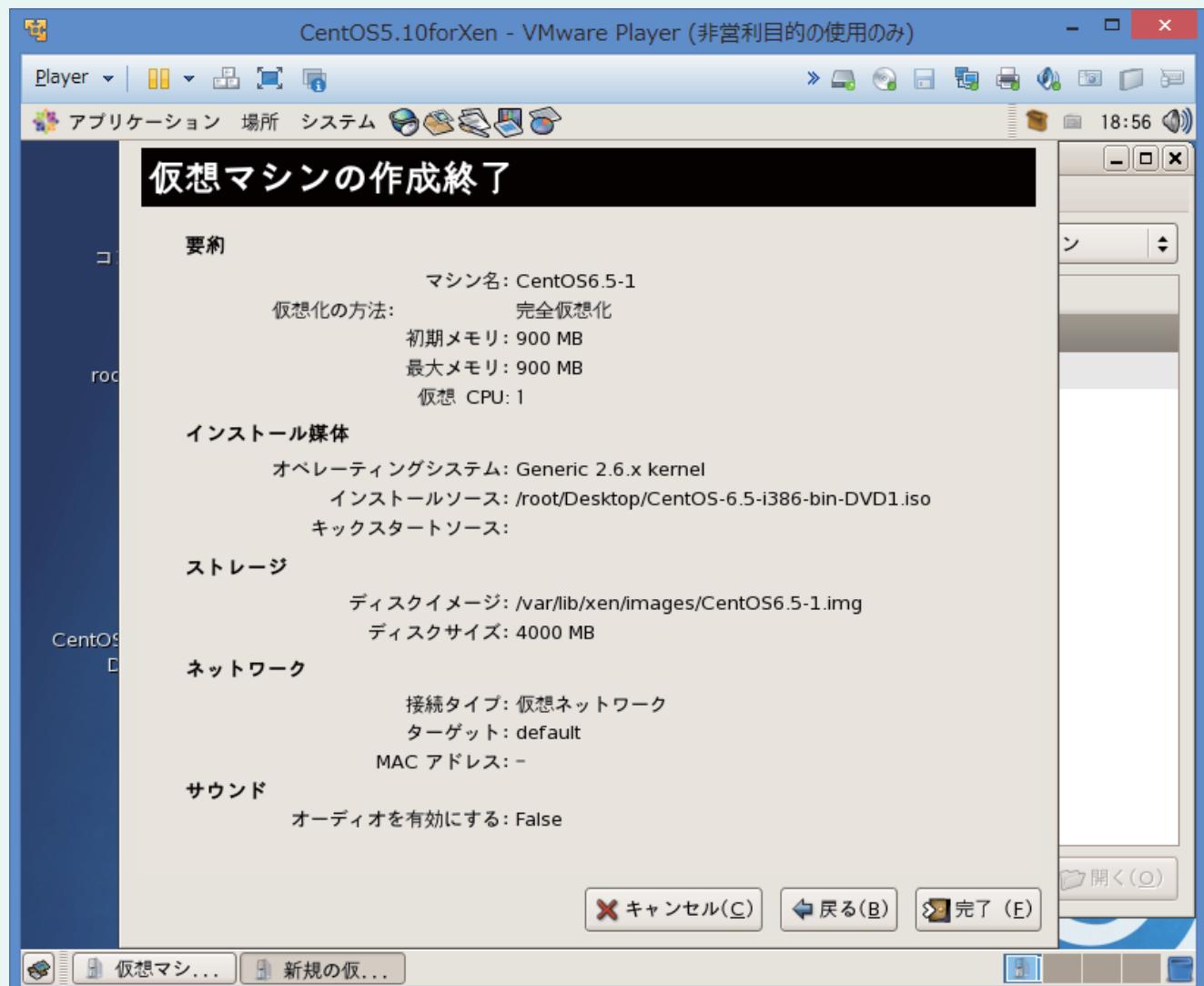
Domain-U(ゲストOS)の作成





330.2 Xen(Xenを使ってみる)

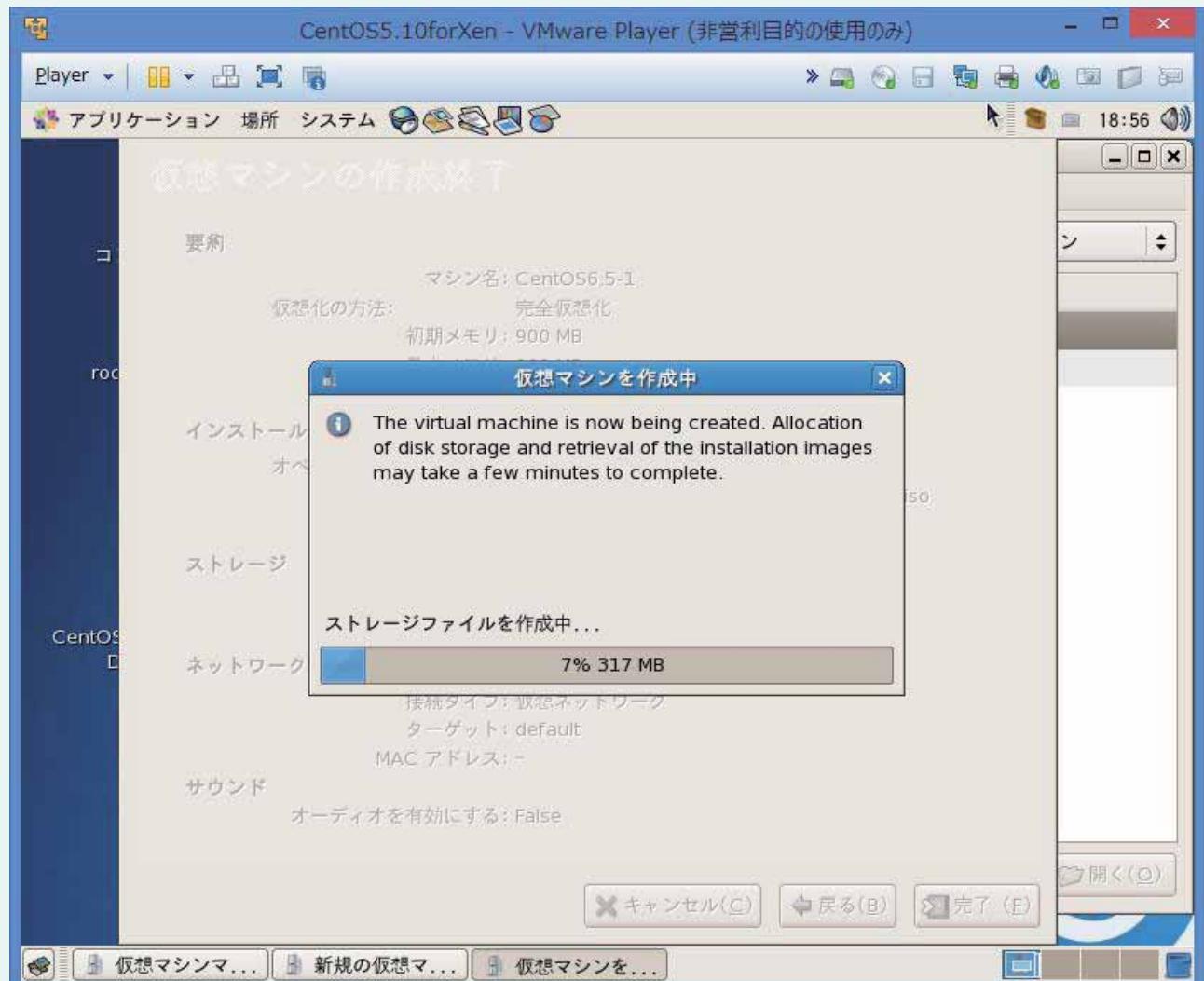
Domain-U(ゲストOS)の作成





330.2 Xen(Xenを使ってみる)

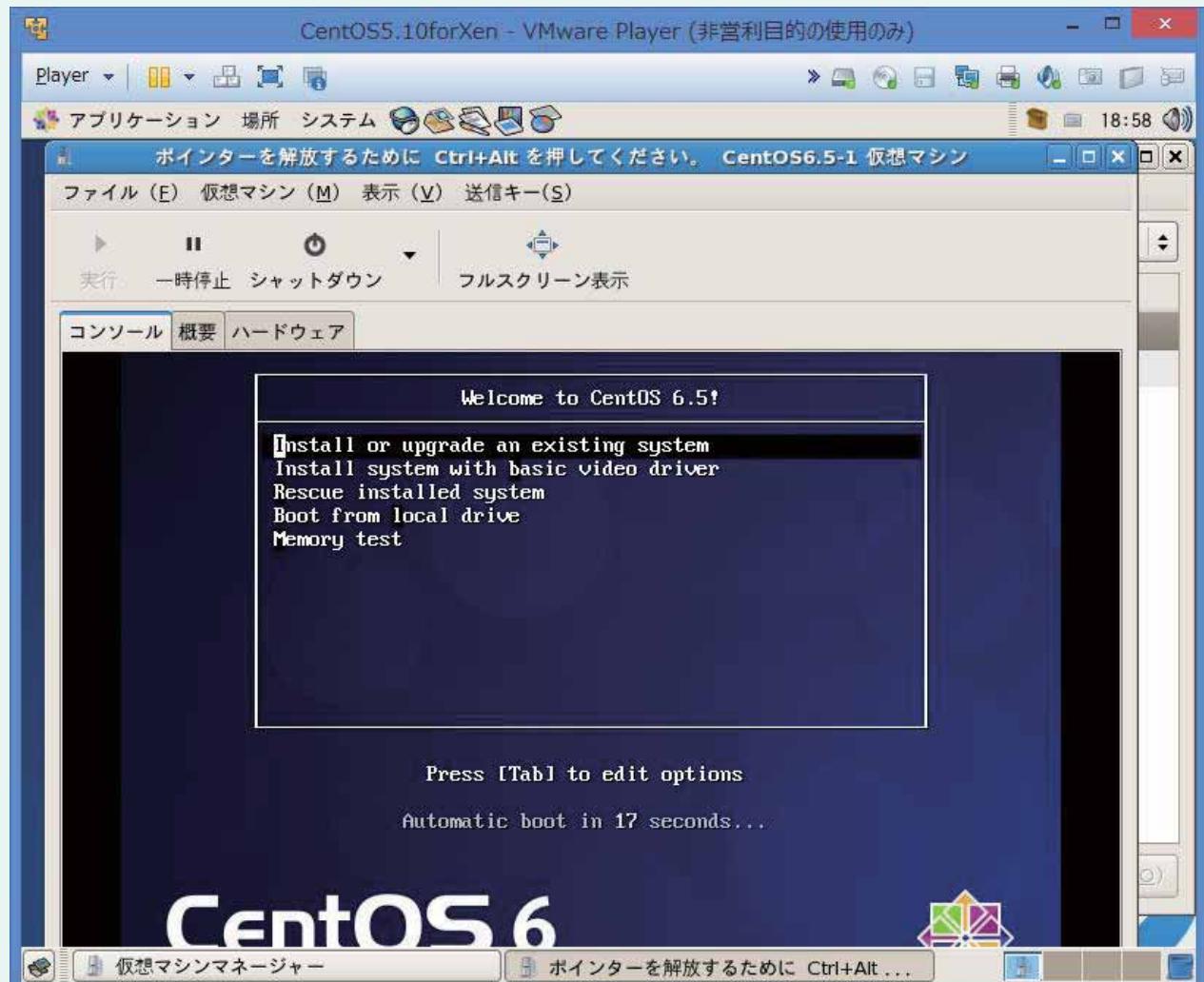
Domain-U(ゲストOS)の作成





330.2 Xen(Xenを使ってみる)

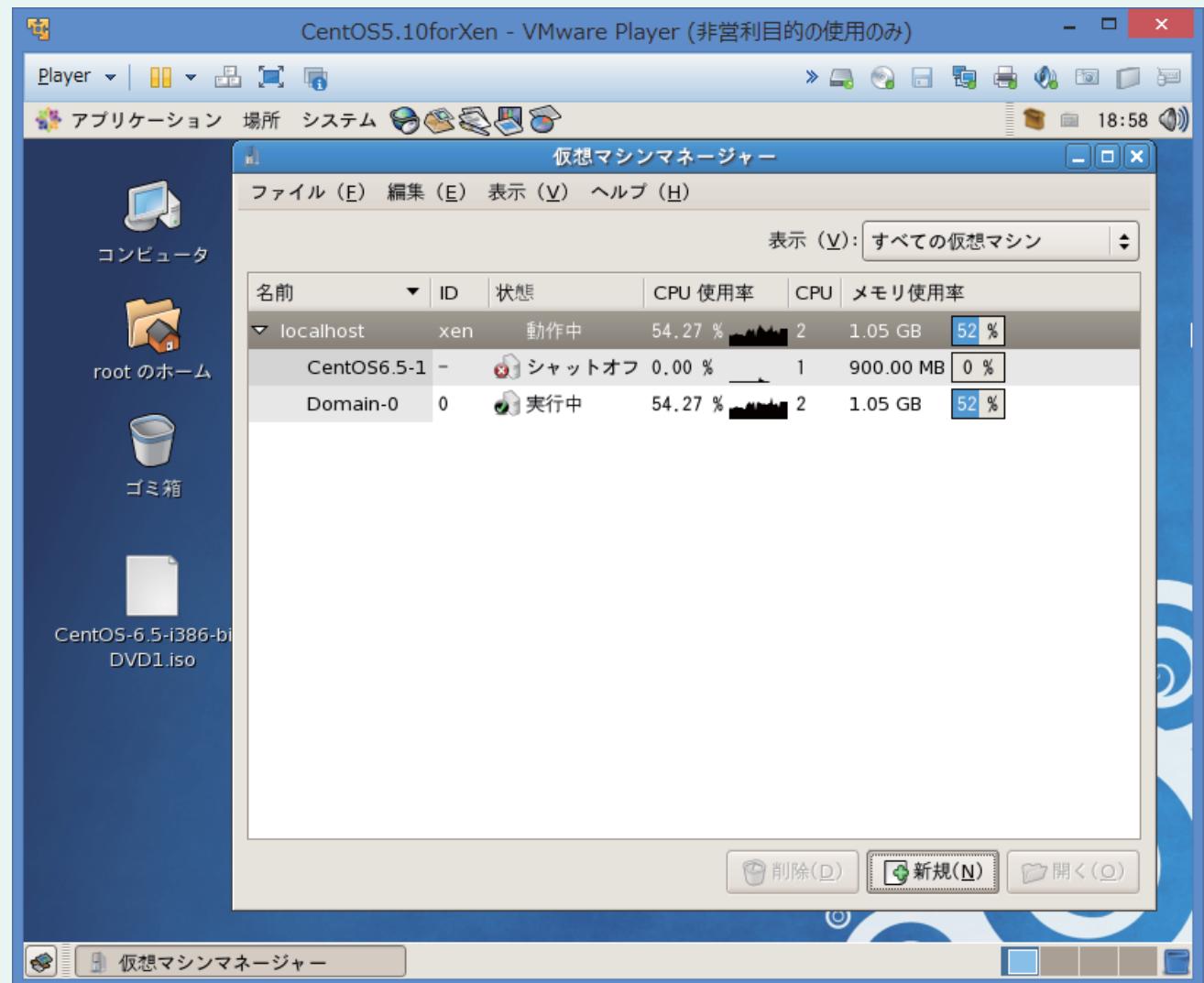
Domain-U(ゲストOS)の起動





330.2 Xen(Xenを使ってみる)

Domain-U(ゲストOS)の起動





330.2 Xen

Xenの設定

Xenの設定は、/etc/xenディレクトリ配下にあります

```
[root@localhost ~]# cd /etc/xen
[root@localhost xen]# ls -l
合計 108
-rw----- 1 root root 539 7月 10 18:57 CentOS6.5-1
drwxr-xr-x 2 root root 4096 5月 21 2013 auto
-rw xr-xr-x 1 root root 482 5月 21 2013 qemu-ifup
drwxr-xr-x 2 root root 4096 7月 11 01:02 scripts
-rw-r--r-- 1 root root 7206 5月 21 2013 xend-config.sxp
-rw-r--r-- 1 root root 1256 5月 21 2013 xend-pci-permissive.sxp
-rw-r--r-- 1 root root 4129 5月 21 2013 xend-pci-quirks.sxp
-rw-r--r-- 1 root root 7264 5月 21 2013 xmexample.hvm
-rw-r--r-- 1 root root 5315 5月 21 2013 xmexample.vti
-rw-r--r-- 1 root root 5463 5月 21 2013 xmexample1
-rw-r--r-- 1 root root 6746 5月 21 2013 xmexample2
```



330.2 Xen

Xenの設定

xendの設定ファイルは、xend-config.sxpです

```
[root@localhost xen]# cat xend-config.sxp | grep "^("
(xend-unix-server yes)
(xend-unix-path /var/lib/xend/xend-socket)
(xend-relocation-hosts-allow '^localhost$' '^localhost$|.localdomain$')
(network-script network-bridge)
(vif-script vif-bridge)
(dom0-min-mem 256)
(dom0-cpus 0)
(vncpasswd '')
(keymap 'en-us')
```



330.2 Xen

Xenの設定

仮想マシン設定はそれぞれの名前の付いたファイルが作成されます

```
[root@localhost xen]# cat CentOS6.5-1
name = "CentOS6.5-1"
uuid = "28bc59a0-5212-6230-0520-3d74e9fa0ee0"
maxmem = 900
memory = 900
vcpus = 1
builder = "hvm"
kernel = "/usr/lib/xen/boot/hvmloader"
boot = "c"
pae = 1
acpi = 1
apic = 1
localtime = 0
on_poweroff = "destroy"
on_reboot = "restart"
on_crash = "restart"
device_model = "/usr/lib64/xen/bin/qemu-dm"
sdl = 0
vnc = 1
vncunused = 1
keymap = "ja"
disk = [ "file:/var/lib/xen/images/CentOS6.5-1.img,hda,w", ",hdc:cdrom,r" ]
vif = [ "mac=00:16:3e:0a:82:21,bridge=virbr0,script=vif-bridge" ]
parallel = "none"
serial = "pty"
```



330.2 Xen

Xenのユーティリティ

xm list(稼働中のドメインの一覧表示)

※ドメインとは、仮想マシン名の事

xm create ドメイン(ドメインの起動)

xm uptime(ドメインの稼働時間表示)

```
[root@localhost xen]# xm list
Name                           ID Mem(MiB) VCPUs State   Time(s)
Domain-0                        0   1074      2 r----- 145.0
[root@localhost xen]# xm create CentOS6.5-1
Using config file "./CentOS6.5-1".
Started domain CentOS6.5-1
[root@localhost xen]# xm list
Name                           ID Mem(MiB) VCPUs State   Time(s)
CentOS6.5-1                      5    908      1 r-----  1.6
Domain-0                        0   1074      2 r----- 147.1
[root@localhost xen]# xm uptime
Name                           ID Uptime
CentOS6.5-1                      5  0:00:30
Domain-0                        0  0:41:12
```



330.2 Xen

Xenのユーティリティ

xm info

(Xenの環境情報表示)

```
[root@localhost xen]# xm info
host                  : localhost.localdomain
release               : 2.6.18-371.el5xen
version               : #1 SMP Tue Oct 1 09:15:30 EDT 2013
machine               : x86_64
nr_cpus               : 2
nr_nodes              : 1
sockets_per_node     : 2
cores_per_socket      : 1
threads_per_core      : 1
cpu_mhz               : 3191
hw_caps               : 0fabfbff:28100800:00000000:00000940:f6ba2223:00000000:0
00000001
total_memory          : 2047
free_memory           : 1
node_to_cpu            : node0:0-1
xen_major              : 3
xen_minor              : 1
xen_extra              : .2-371.el5
xen_caps               : xen-3.0-x86_64 xen-3.0-x86_32p hvm-3.0-x86_32 hvm-3.0-x
86_32p hvm-3.0-x86_64
xen_pagesize          : 4096
platform_params        : virt_start=0xfffff8000000000000
xen_changeset         : unavailable
cc_compiler            : gcc version 4.1.2 20080704 (Red Hat 4.1.2-54)
cc_compile_by          : mockbuild
cc_compile_domain      : (none)
cc_compile_date        : Tue Oct 1 08:26:28 EDT 2013
xend_config_format    : 2
```



330.2 Xen

Xenのユーティリティ

xm top(稼働中のドメイン情報をtopコマンド風に表示)

※xentopコマンドでも同様の事は可能です

xentop - 21:48:56 Xen 3.1.2-371.el5																
2 domains: 2 running, 0 blocked, 0 paused, 0 crashed, 0 dying, 0 shutdown																
Mem: 2096636k total, 2094692k used, 1944k free CPUs: 2 @ 3191MHz																
NAME	STATE	CPU(sec)	CPU(%)	MEM(k)	MEM(%)	MAXMEM(k)	MAXMEM(%)	VCPUS	NETS	NETTX(k)	NETRX(k)	VBDS	VBD_00	VBD_RD	VBD_WR	SSID
CentOS6.5-1	----r	278	98.1	929664	44.3	937984	44.7	1	1	0	0	1	0	0	0	0
Domain-0	----r	159	4.7	1099776	52.5	no limit	n/a	2	4	300	107	0	0	0	0	0



330.2 Xen

Xenのユーティリティ

xm shutdown

(ドメインの終了)

xm destroy

(ドメインの強制終了)

xm reboot

(ドメインの再起動)

```
[root@localhost xen]# xm list
Name                                     ID Mem(MiB) VCPUs State   Time(s)
CentOS6.5-1                               5    908     1 r----  347.8
Domain-0                                  0   1074     2 r----  163.0
[root@localhost xen]# xm shutdown CentOS6.5-1
[root@localhost xen]# xm list
Name                                     ID Mem(MiB) VCPUs State   Time(s)
Domain-0                                 0   1074     2 r----  165.1
[root@localhost xen]#
[root@localhost xen]# xm create CentOS6.5-1
Using config file "./CentOS6.5-1".
Started domain CentOS6.5-1
[root@localhost xen]# xm list
Name                                     ID Mem(MiB) VCPUs State   Time(s)
CentOS6.5-1                               6    908     1 r----  3.1
Domain-0                                  0   1074     2 r----  167.5
[root@localhost xen]#
[root@localhost xen]# xm destroy CentOS6.5-1
[root@localhost xen]# xm list
Name                                     ID Mem(MiB) VCPUs State   Time(s)
Domain-0                                 0   1074     2 r----  168.9
[root@localhost xen]#
[root@localhost xen]#
[root@localhost xen]# xm create CentOS6.5-1
Using config file "./CentOS6.5-1".
Started domain CentOS6.5-1
[root@localhost xen]# xm list
Name                                     ID Mem(MiB) VCPUs State   Time(s)
CentOS6.5-1                               7    908     1 r----  1.2
Domain-0                                  0   1074     2 r----  171.0
[root@localhost xen]# xm reboot CentOS6.5-1
```



330.2 Xen

Xenのユーティリティ

xm pause(ドメインの一時停止)

xm unpause(一時停止中のドメインの再開)

```
[root@localhost xen]# xm list
```

Name	ID	Mem(MiB)	VCPUs	State	Time(s)
CentOS6.5-1	8	908	1	r-----	5.4
Domain-0	0	1074	2	r-----	175.9

```
[root@localhost xen]# xm pause CentOS6.5-1
```

```
[root@localhost xen]# xm list
```

Name	ID	Mem(MiB)	VCPUs	State	Time(s)
CentOS6.5-1	8	908	1	--p---	15.0
Domain-0	0	1074	2	r-----	176.8

```
[root@localhost xen]#
```

```
[root@localhost xen]# xm unpause CentOS6.5-1
```

```
[root@localhost xen]# xm list
```

Name	ID	Mem(MiB)	VCPUs	State	Time(s)
CentOS6.5-1	8	908	1	r-----	18.3
Domain-0	0	1074	2	r-----	177.7



330.2 Xen

Xenのユーティリティ

他のコマンドについても、xm help(利用可能コマンド表示)で確認出来ます

```
[root@localhost ~]# xm help
Usage: xm <subcommand> [args]

Control, list, and manipulate Xen guest instances.

xm full list of subcommands:

console          Attach to <Domain>'s console.
create           Create a domain based on <ConfigFile>.
destroy          Terminate a domain immediately.
domid            Convert a domain name to domain id.
domname          Convert a domain id to domain name.
dump-core        Dump core for a specific domain.
list              List information about all/some domains.
mem-max          Set the maximum amount reservation for a domain.
mem-set          Set the current memory usage for a domain.
migrate          Migrate a domain to another machine.
pause            Pause execution of a domain.
reboot           Reboot a domain.
rename           Rename a domain.
restore          Restore a domain from a saved state.
save              Save a domain state to restore later.
shutdown         Shutdown a domain.
sysrq            Send a sysrq to a domain.
trigger          Send a trigger to a domain.
top               Monitor a host and the domains in real time.
```



330.3 KVM



330.3 KVM

KVMのインストール

仮想化ソフトウェアにて、CPU仮想化支援機能を有効化した仮想PCを作成します。

The screenshot shows the VMware Player interface with a virtual machine named "CentOS6.5forKVM2". The configuration dialog is open, specifically the "Hardware" tab under "Virtual Machine Settings". In the hardware list, the "Processor" item is selected, showing a value of "2". The "Processor Cores" dropdown is set to "2". Under the "Processor" section, the "Priority Mode" is set to "Intel VT-x/EPT または AMD-V/RVI". There are three checked checkboxes: "Disable page translation lookaside buffer (D)", "Enable Intel VT-x/EPT または AMD-V/RVI (V)", and "Enable CPU Performance Counter Virtualization (U)". A warning message states: "Virtualization engine mode can affect performance if specified manually in certain workloads". At the bottom of the dialog, there are "Add (A)...", "Delete (R)...", "OK", "Cancel", and "Help" buttons.

VMware Player (非営利目的の使用のみ)

ホーム
CentOS6.5forKVM2

仮想マシン設定

ハードウェア オプション

デバイス	概要
メモリ	4 GB
プロセッサ	2
ハードディスク(SCSI)	50 GB
CD/DVD(IDE)	C:\Users\root\Desktop\ISOs\Cent...
フロッピー	自動検出
ネットワークアダプタ	ブリッジ(自動)
USBコントローラ	あり
サウンドカード	自動検出
プリンタ	あり
ディスプレイ	自動検出

プロセッサ
プロセッサコアの数 (C): 2

仮想化エンジン
優先モード (M): Intel VT-x/EPT または AMD-V/RVI

バイナリ変換のアクセラレーションを無効にする (D)
 Intel VT-x/EPT または AMD-V/RVI を仮想化 (V)
 CPU パフォーマンスカウンタを仮想化 (U)

⚠️ 仮想化エンジンモードを手動で指定すると、一定のワークロード下でのパフォーマンスに影響を与える可能性があります。

追加 (A)... 削除 (R) OK キャンセル ヘルプ

CentOS6.5forKVM2

状態: パワーオフ
OS: CentOS 64ビット
バージョン: Workstation 9.0 仮想マシン
RAM: 4 GB

▶ 仮想マシンの再生 (L)
🔧 仮想マシン設定の編集 (D)



330.3 KVM

KVMのインストール(CentOS6.5 64bitsを使用)

※1/7現在、6.6が最新。





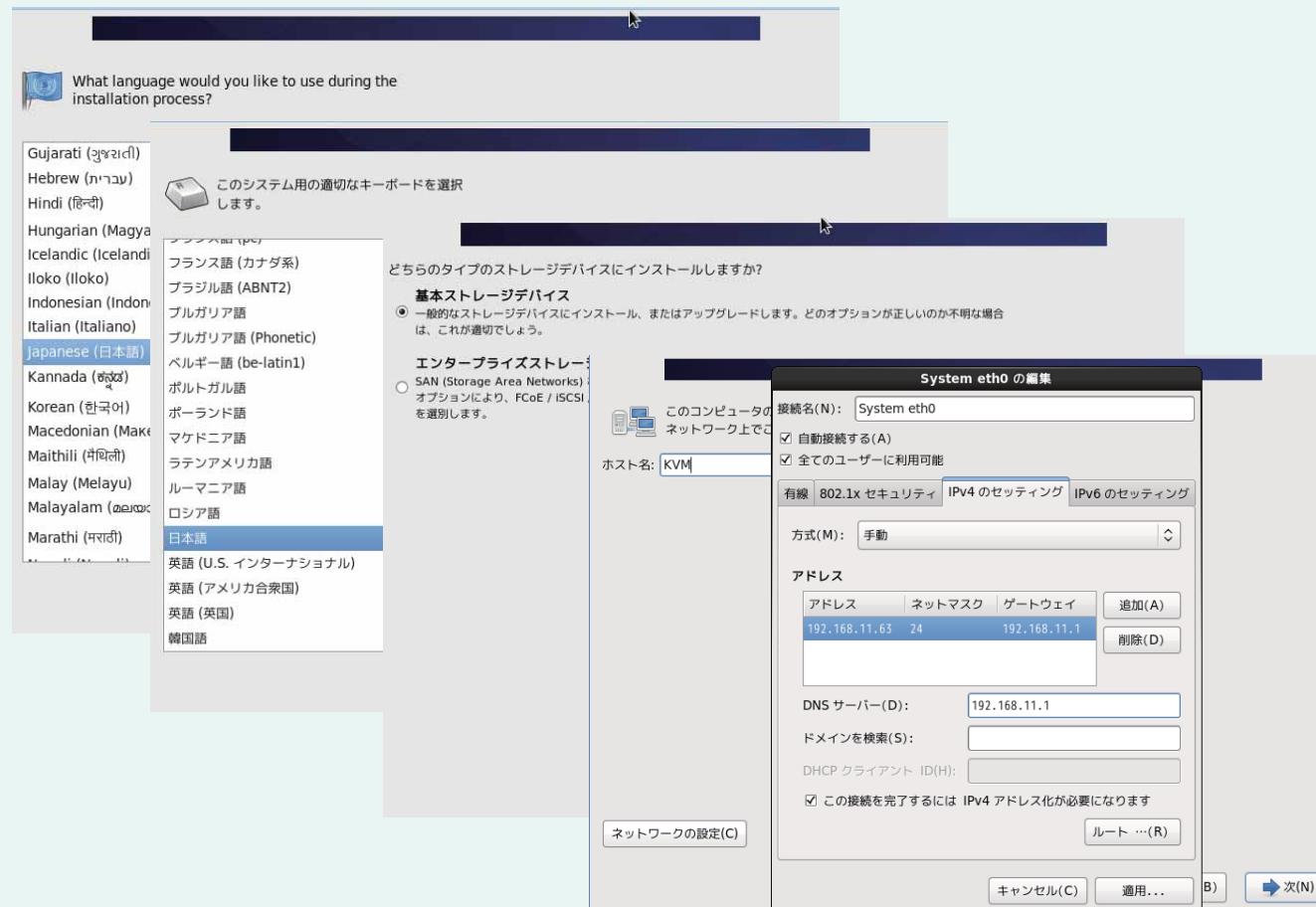
LPIC 304 技術解説セミナー



330.3 KVM

KVMのインストール

ソフトウェアの選択までは、自身の環境等に合わせて設定を行います。





330.3 KVM KVMのインストール

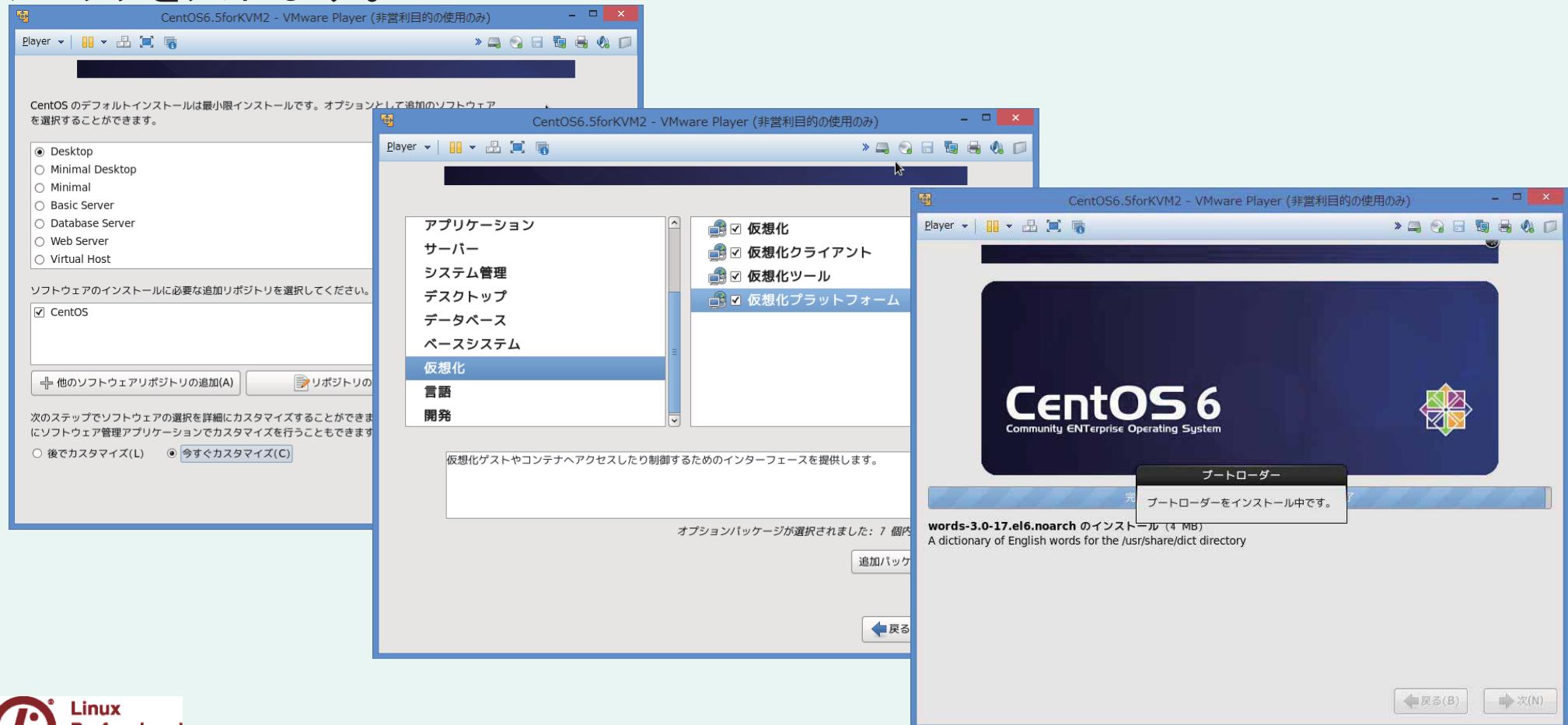




330.3 KVM

KVMのインストール

ソフトウェアの選択では、「今すぐカスタマイズ」から、
仮想化 > 仮想化、仮想化クライアント、仮想化ツール、仮想化プラットフォームに
チェックを入れます。

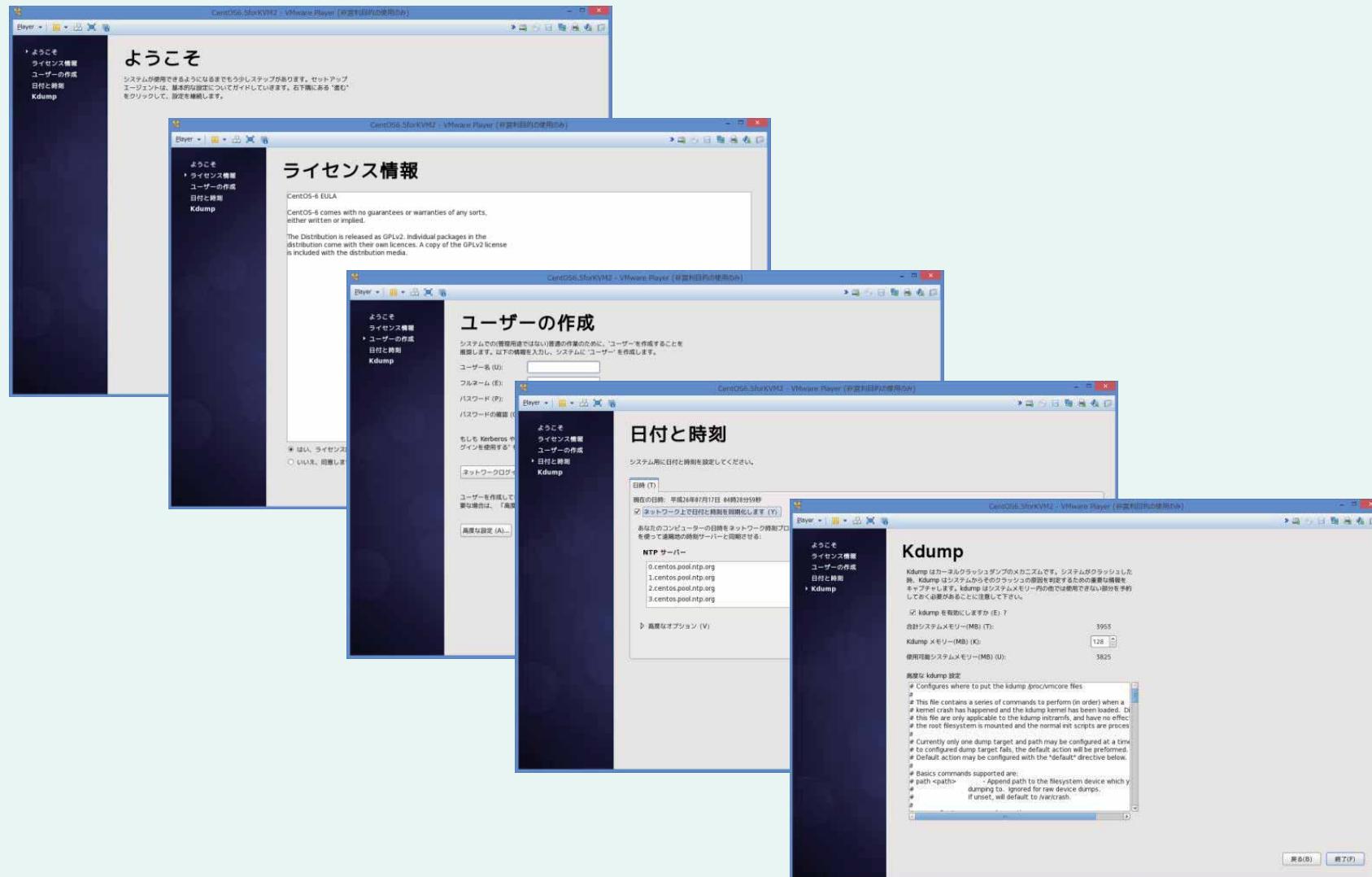




LPIC 304 技術解説セミナー



330.3 KVM KVMのインストール





330.3 KVM

KVMのインストール

起動後、コンソールからCPUの仮想化機能対応を確認します。(vmx or svm)

cat /proc/cpuinfo

```
[root@KVM ~]# cat /proc/cpuinfo
processor      : 0
vendor_id     : GenuineIntel
cpu family    : 6
model         : 58
model name    : Intel(R) Core(TM) i5-3470 CPU @ 3.20GHz
stepping       : 9
cpu MHz       : 3192.743
cache size    : 6144 KB
fpu           : yes
fpu_exception : yes
cpuid level   : 13
wp            : yes
flags          : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clfl
ush dts mmx fxsr sse sse2 ss syscall nx rdtscp lm constant_tsc arch_perfmon pebs bts xtopology
tsc_reliable nonstop_tsc aperfmpfperf unfair_spinlock pni pclmulqdq vmx sse3 cx16 pcid sse4_1
sse4_2 x2apic popcnt aes xsave avx f16c rdrand hypervisor lahf_lm ida arat epb xsaveopt pln pt
s dts tpr_shadow vnmi ept vpid fsgsbase smep
bogomips      : 6385.48
clflush size  : 64
cache_alignment: 64
address sizes  : 40 bits physical, 48 bits virtual
power management:
```



330.3 KVM

KVMのインストール

また、kvm用のモジュールがカーネルでロードされているか確認します。

```
lsmod | grep kvm
```

なお、Intel製CPUの場合、kvmとkvm_intel

AMD製CPUの場合、kvmとkvm_amdとなります。

```
[root@KVM ~]# lsmod | grep kvm
kvm_intel               54285   3
kvm                   333172   1 kvm_intel
[root@KVM ~]#
```



330.3 KVM

KVMのインストール

qemu-kvm(KVMコマンドライン)、libvirt、bridge-utils(ブリッジ管理)、python-virtinst(virt-installなどのツール)のインストールを確認。

```
root@KVM:~  
ファイル(F) 編集(E) 表示(V) 検索(S) 端末(T) ヘルプ(H)  
[ root@KVM ~ ]# rpm -q qemu-kvm libvirt bridge-utils python-virtinst  
qemu-kvm-0.12.1.2-2.415.el6.x86_64  
libvirt-0.10.2-29.el6.x86_64  
bridge-utils-1.2-10.el6.x86_64  
python-virtinst-0.600.0-18.el6.noarch  
[ root@KVM ~ ]#
```



330.3 KVM

KVMの設定

仮想マシンをブリッジ接続する為に、ifcfg-br0を作成します。

```
cd /etc/sysconfig/network-scripts  
cp -p ifcfg-eth0 ifcfg-br0
```

```
[root@KVM ~]# cd /etc/sysconfig/network-scripts/  
[root@KVM network-scripts]# cp -p ifcfg-eth0 ifcfg-br0
```

ifcfg-eth0の編集

NM_CONTROLLED=no
BRIDGE=br0

```
[root@KVM network-scripts]# vi ifcfg-eth0  
[root@KVM network-scripts]# cat ifcfg-eth0  
DEVICE=eth0  
TYPE=Ethernet  
UUID=b8dc0ffd-9bc4-490f-ab95-b6285d8c653b  
ONBOOT=yes  
NM_CONTROLLED=no  
BOOTPROTO=none  
HWADDR=00:0C:29:5C:61:8A  
IPADDR=192.168.11.63  
PREFIX=24  
GATEWAY=192.168.11.1  
DNS1=192.168.11.1  
DEFROUTE=yes  
IPV4_FAILURE_FATAL=yes  
IPV6INIT=no  
NAME="System eth0"  
BRIDGE=br0
```



330.3 KVM

KVMの設定

ifcfg-br0の編集

DEVICE=br0

TYPE=Bridge

NM_CONTROLLED=no

#NAME="System eth0"

```
[root@KVM network-scripts]# vi ifcfg-br0
[root@KVM network-scripts]# cat ifcfg-br0
DEVICE=br0
TYPE=Bridge
UUID=b8dc0ffd-9bc4-490f-ab95-b6285d8c653b
ONBOOT=yes
NM_CONTROLLED=no
BOOTPROTO=none
HWADDR=00:0C:29:5C:61:8A
IPADDR=192.168.11.63
PREFIX=24
GATEWAY=192.168.11.1
DNS1=192.168.11.1
DEFROUTE=yes
IPV4_FAILURE_FATAL=yes
IPV6INIT=no
#NAME="System eth0"
```



330.3 KVM

KVMの設定

ネットワークサービスを再起動します。

/etc/init.d/network restart

```
[root@KVM network-scripts]# /etc/init.d/network restart
インターフェース br0 を終了中:                                [  OK  ]
ループバックインターフェースを終了中:                            [  OK  ]
ループバックインターフェイスを呼び込み中:                          [  OK  ]
インターフェース eth0 を活性化中: device eth0 is already a member of a bridge; can't enslave it to bridge br0.          [  OK  ]
インターフェース br0 を活性化中: Determining if ip address 192.168.11.63 is already in use for device br0...
                                                               [  OK  ]
```



330.3 KVM

KVMの設定

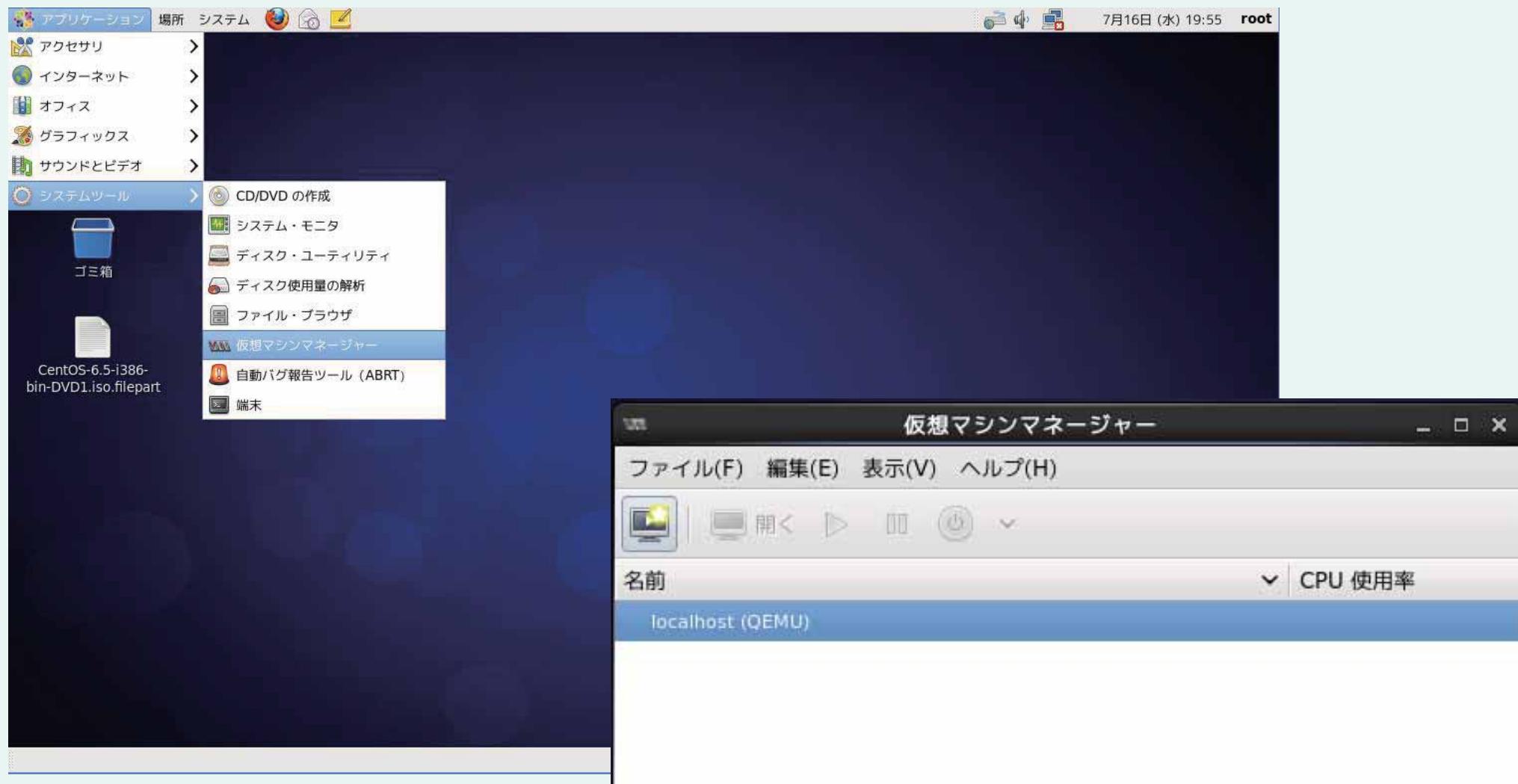
ifconfigで確認

```
[root@KVM network-scripts]# ifconfig  
br0      Link encap:Ethernet HWaddr 00:0C:29:5C:61:8A  
          inet addr:192.168.11.63 Bcast:192.168.11.255 Mask:255.255.255.0  
          inet6 addr: fe80::20c:29ff:fe5c:618a/64 Scope:Link  
             UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
             RX packets:4219 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:1616 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:0  
             RX bytes:3182451 (3.0 MiB) TX bytes:112430 (109.7 KiB)  
  
eth0      Link encap:Ethernet HWaddr 00:0C:29:5C:61:8A  
          inet6 addr: fe80::20c:29ff:fe5c:618a/64 Scope:Link  
             UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
             RX packets:5002622 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:846083 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:1000  
             RX bytes:4136527211 (3.8 GiB) TX bytes:54901858 (52.3 MiB)  
  
lo        Link encap:Local Loopback  
          inet addr:127.0.0.1 Mask:255.0.0.0  
          inet6 addr: ::1/128 Scope:Host  
             UP LOOPBACK RUNNING MTU:16436 Metric:1  
             RX packets:16 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:16 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:0  
             RX bytes:960 (960.0 b) TX bytes:960 (960.0 b)  
  
virbr0    Link encap:Ethernet HWaddr 52:54:00:E6:3B:05  
          inet addr:192.168.122.1 Bcast:192.168.122.255 Mask:255.255.255.0  
             UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1  
             RX packets:0 errors:0 dropped:0 overruns:0 frame:0  
             TX packets:0 errors:0 dropped:0 overruns:0 carrier:0  
             collisions:0 txqueuelen:0  
             RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
```



330.3 KVM

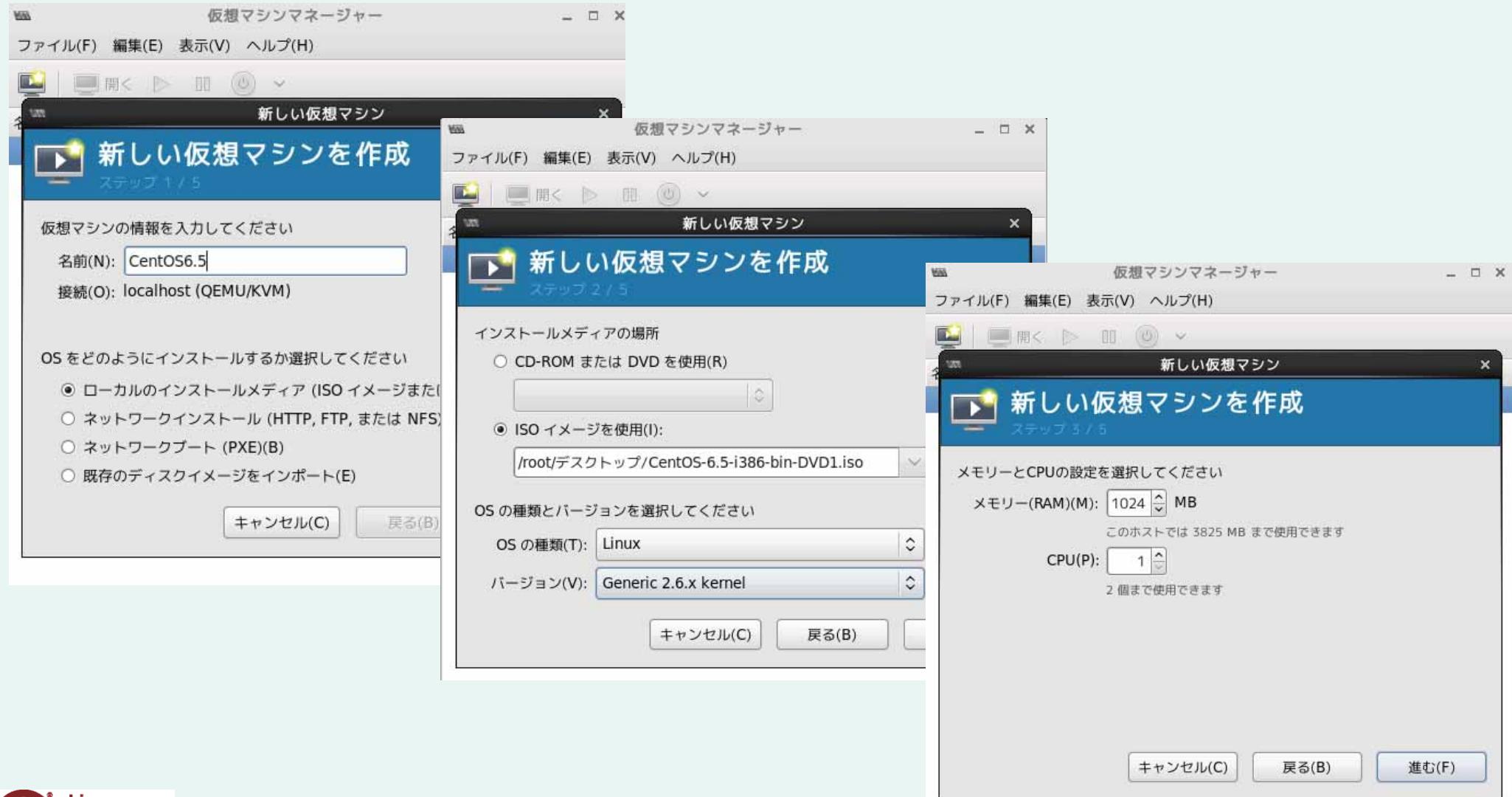
KVMを使用(仮想マシンマネージャによるゲストOS作成)





330.3 KVM

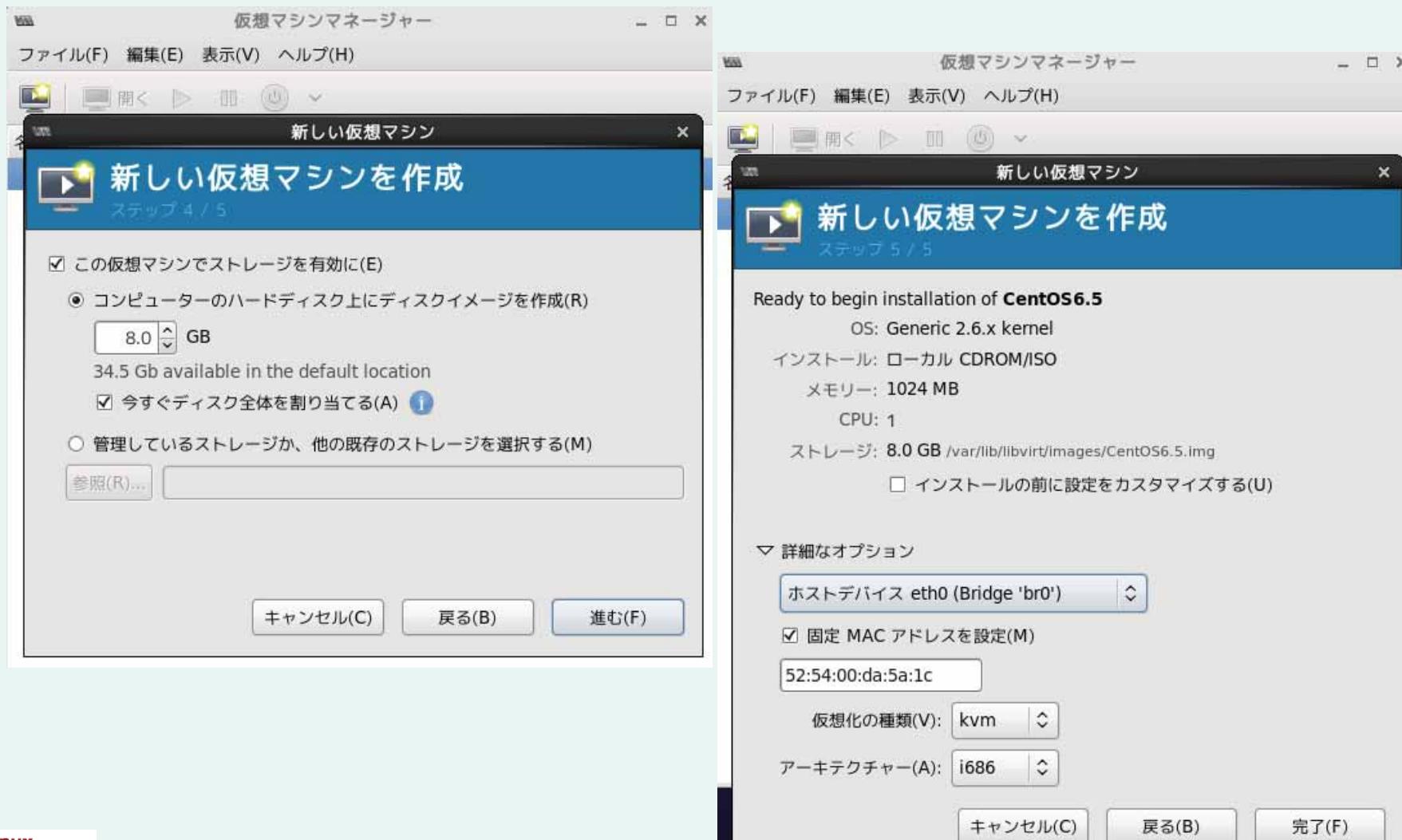
KVMの使用(仮想マシンマネージャによるゲストOS作成)





330.3 KVM

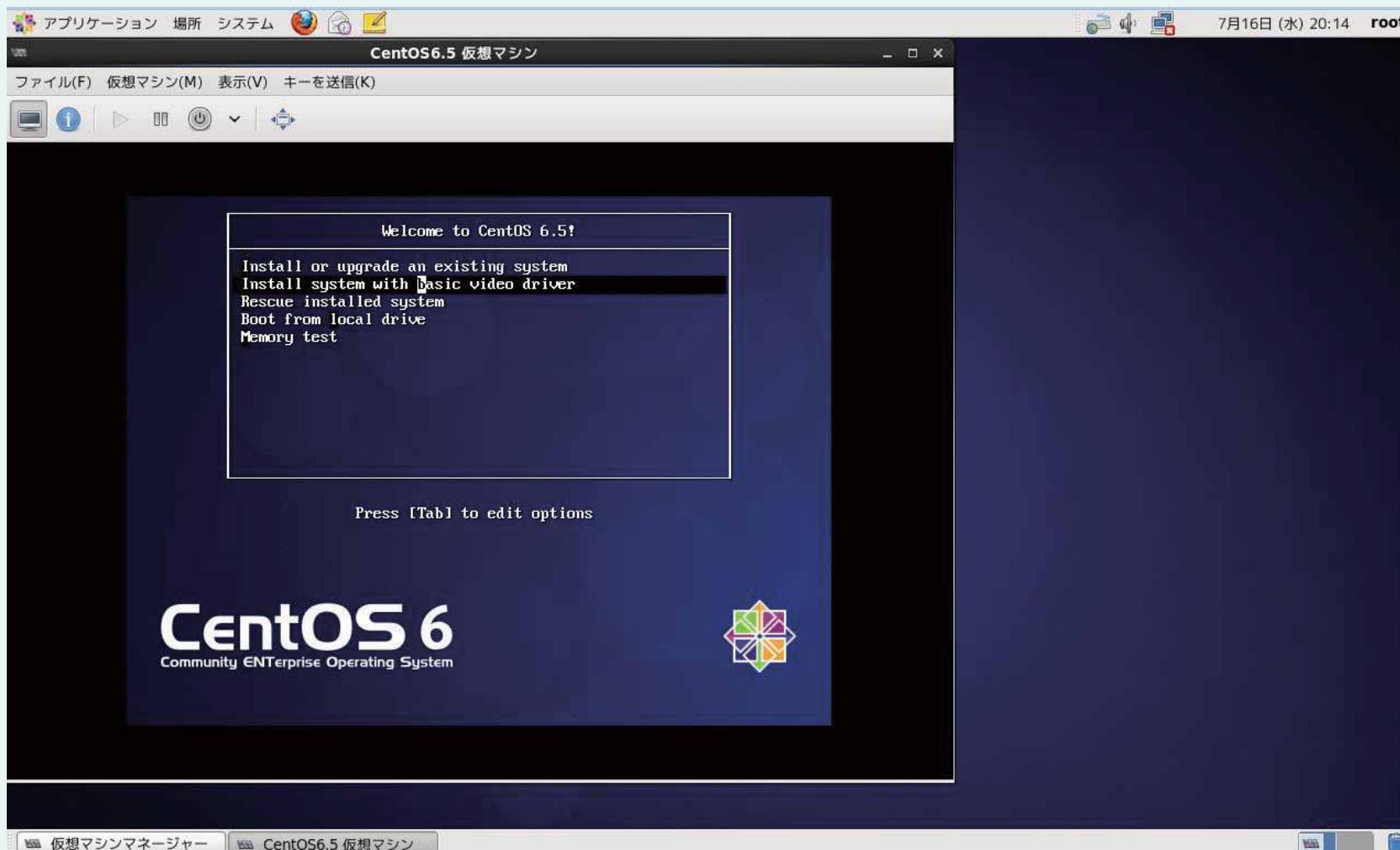
KVMの使用(仮想マシンマネージャによるゲストOS作成)





330.3 KVM

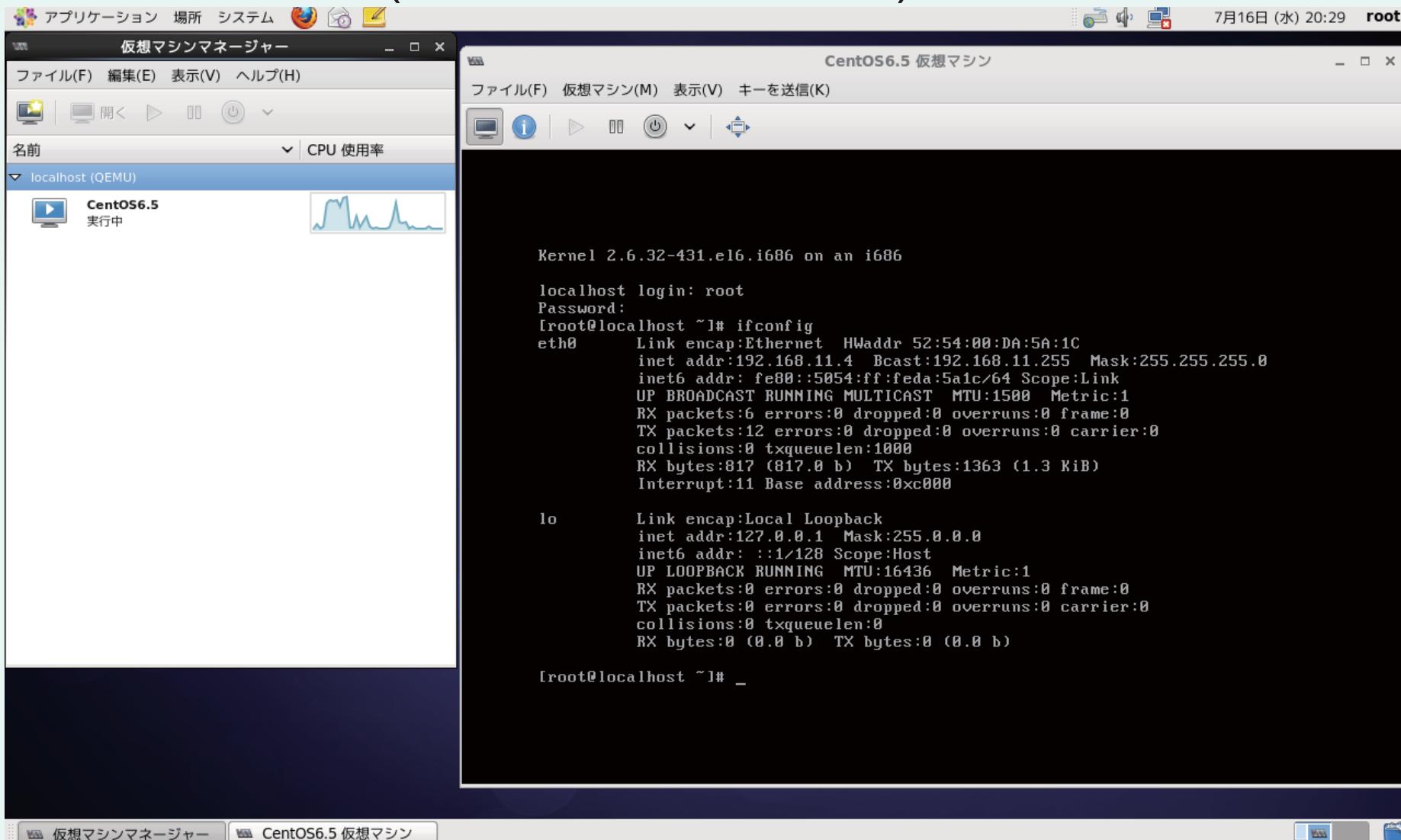
KVMの使用(仮想マシンの起動)





330.3 KVM

KVMのユーティリティ(仮想マシンマネージャ)





330.3 KVM

KVMのユーティリティ (virshコマンド)

virsh version

(virshのバージョン確認)

virsh list

(稼働中マシンリスト表示)

virsh dominfo

(仮想マシン情報表示)

virsh domstate

(仮想マシン状態表示)

```
[root@KVM ~]# virsh version
コンパイル時に使用したライブラリ: libvirt 0.10.2
使用中のライブラリ: libvirt 0.10.2
使用中の API: QEMU 0.10.2
実行中のハイパーバイザ: QEMU 0.12.1

[root@KVM ~]# virsh list
 Id   名前           状態
 -----
 2    CentOS6.5      実行中

[root@KVM ~]# virsh dominfo CentOS6.5
Id:          2
名前:        CentOS6.5
UUID:        57aae801-ff60-6bb3-03b6-9d10b2fb77ce
OS タイプ:   hvm
状態:        実行中
CPU:          1
CPU 時間:   124.3s
最大メモリー: 1048576 KiB
使用メモリー: 1048576 KiB
永続:        はい (yes)
自動起動:   無効にする
管理済み保存: いいえ (no)
セキュリティモデル: selinux
セキュリティ DOI: 0
セキュリティラベル: system_u:system_r:svirt_t:s0:c211,c928 (enforcing)

[root@KVM ~]# virsh domstate CentOS6.5
実行中
```



330.3 KVM

virsh console
(仮想マシンへの接続)
virsh shutdown
(仮想マシンの終了)
virsh start
(仮想マシンの開始)
virsh reboot
(仮想マシンの再起動)
virsh suspend
(仮想マシンの一時停止)
virsh resume
(仮想マシンの再開)
virsh destroy
(仮想マシンの強制終了)

```
[root@KVM ~]# virsh console 2  
ドメイン CentOS6.5 に接続しました  
エスケープ文字は ^] です
```

```
[root@KVM ~]# virsh shutdown CentOS6.5  
ドメイン CentOS6.5 をシャットダウンしています
```

```
[root@KVM ~]# virsh start CentOS6.5  
ドメイン CentOS6.5 が起動されました
```

```
[root@KVM ~]# virsh reboot CentOS6.5  
ドメイン CentOS6.5 を再起動しています
```

```
[root@KVM ~]# virsh suspend CentOS6.5  
ドメイン CentOS6.5 は一時停止されました
```

```
[root@KVM ~]# virsh resume CentOS6.5  
ドメイン CentOS6.5 が再開されました
```

```
[root@KVM ~]# virsh destroy CentOS6.5  
ドメイン CentOS6.5 は強制停止されました
```



330.3 KVM

他のコマンドについても、
virsh helpで確認出来ます。

```
[root@KVM ~]# virsh help
グループ別コマンド:

Domain Management (ヘルプのキーワード 'domain'):
attach-device          XML ファイルによるデバイスの接続
attach-disk            ディスクデバイスの接続
attach-interface       ネットワークインターフェースの接続
autostart              ドメインの自動起動
blkdeviotune           ブロックデバイスの I/O チューニングパラメーターの設定または取得
blkiotune              ブロック I/O パラメーターの取得または設定
blockcommit             ブロックのコミット操作の開始
blockcopy               ブロックコピー操作の開始
blockjob                アクティブなブロック操作の管理
blockpull               既存のイメージからのディスクの配置
blockresize             ドメインのブロックデバイスの容量変更
change-media            CD またはフロッピードライブのメディアの変更
console                 ゲストのコンソールへの接続
cpu-baseline            ベースライン CPU の計算
cpu-compare             XML ファイルに記載された CPU とホスト CPU の比較
cpu-stats               ドメインの CPU 統計情報の表示
create                  XML ファイルによるドメインの作成
define                 XML ファイルによるドメインの定義（起動なし）
desc                   ドメインの説明またはタイトルの表示または設定
destroy                ドメインの強制停止
detach-device           XML ファイルによるデバイスの切断
detach-disk             ディスクデバイスの切断
detach-interface        ネットワークインターフェースの切断
domdisplay              ドメインのディスプレイ接続 URI
domhostname            ドメインのホスト名の表示
domid                  ドメイン名または UUID からドメイン ID への変換
```



ところで、XenやKVMなどのハイパーバイザーは
どんな場面で利用されているんだろう？

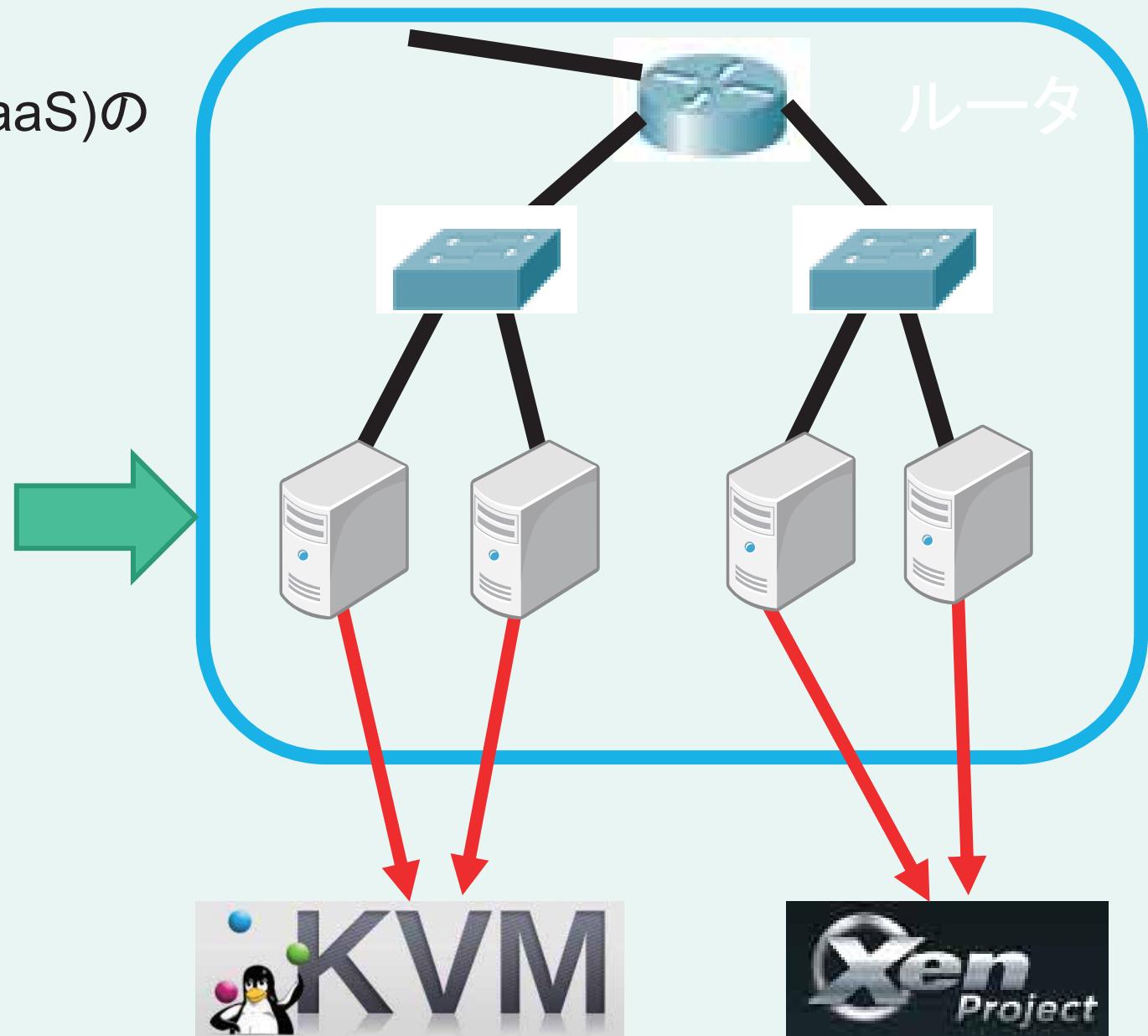
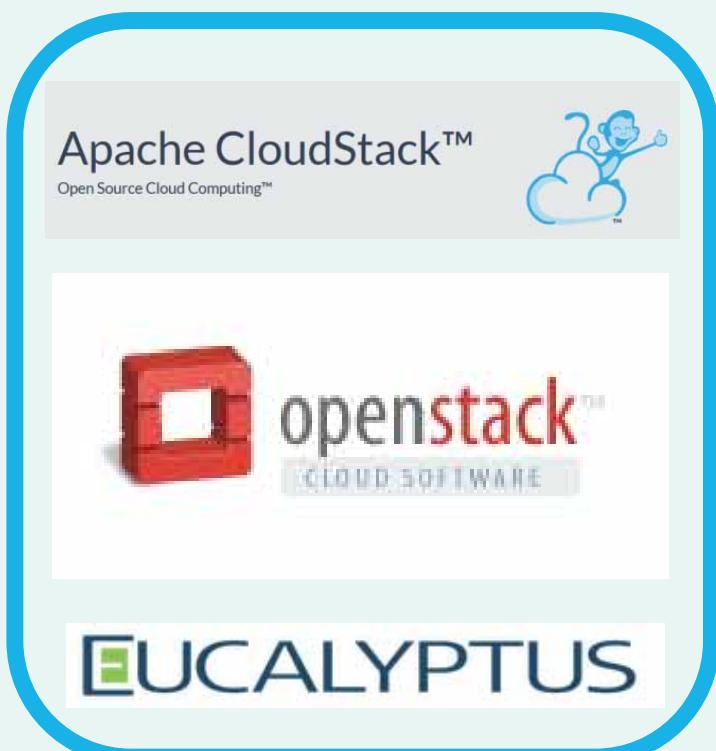
検証環境や物理サーバからの移行先？？



LPIC 304 技術解説セミナー



それ以外にも、
クラウド基盤ソフトウェア(IaaS)
の
サーバ格納先などとして
利用する事が出来ます。



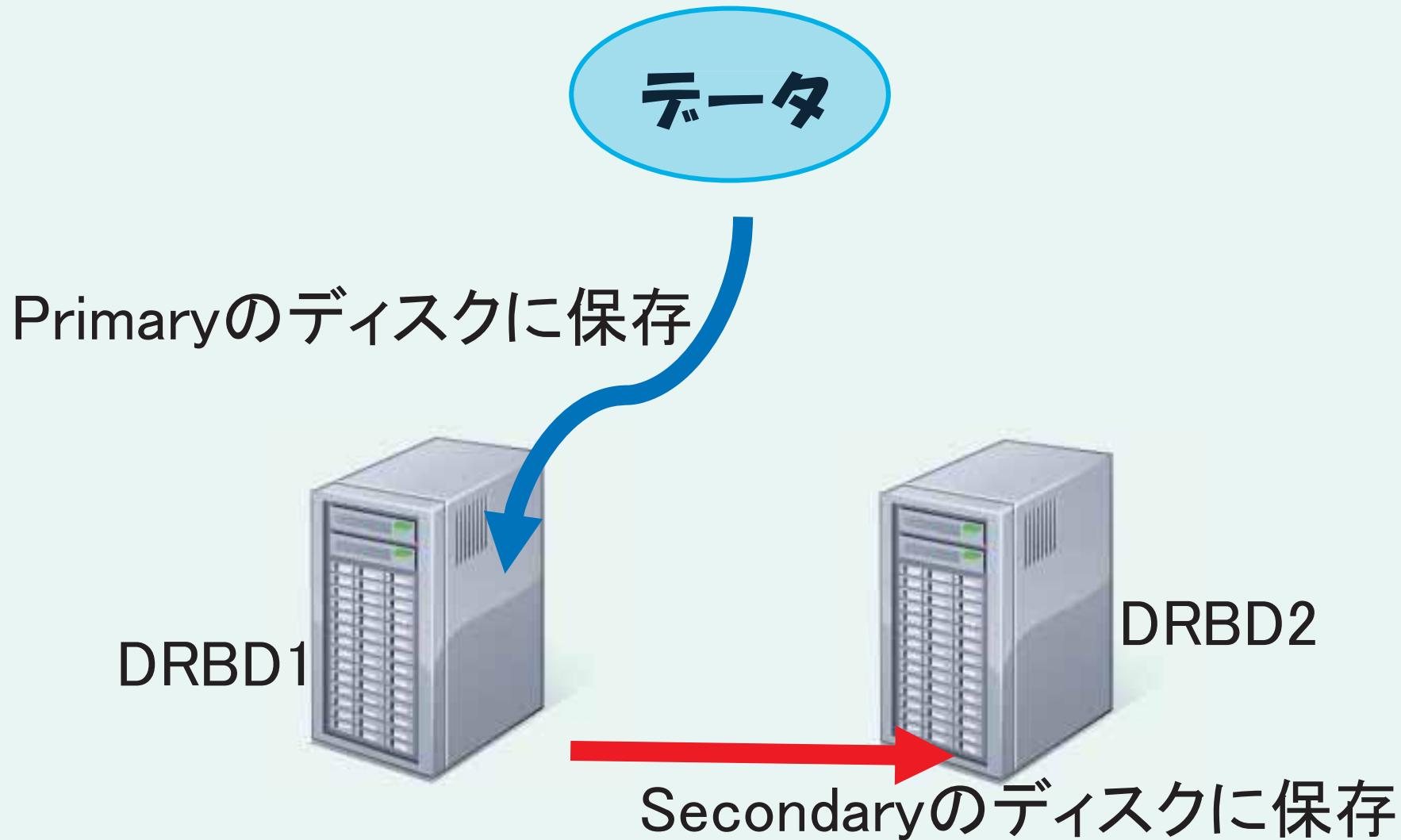


③技術解説

主題333: クラスタストレージ



クラスタストレージとは

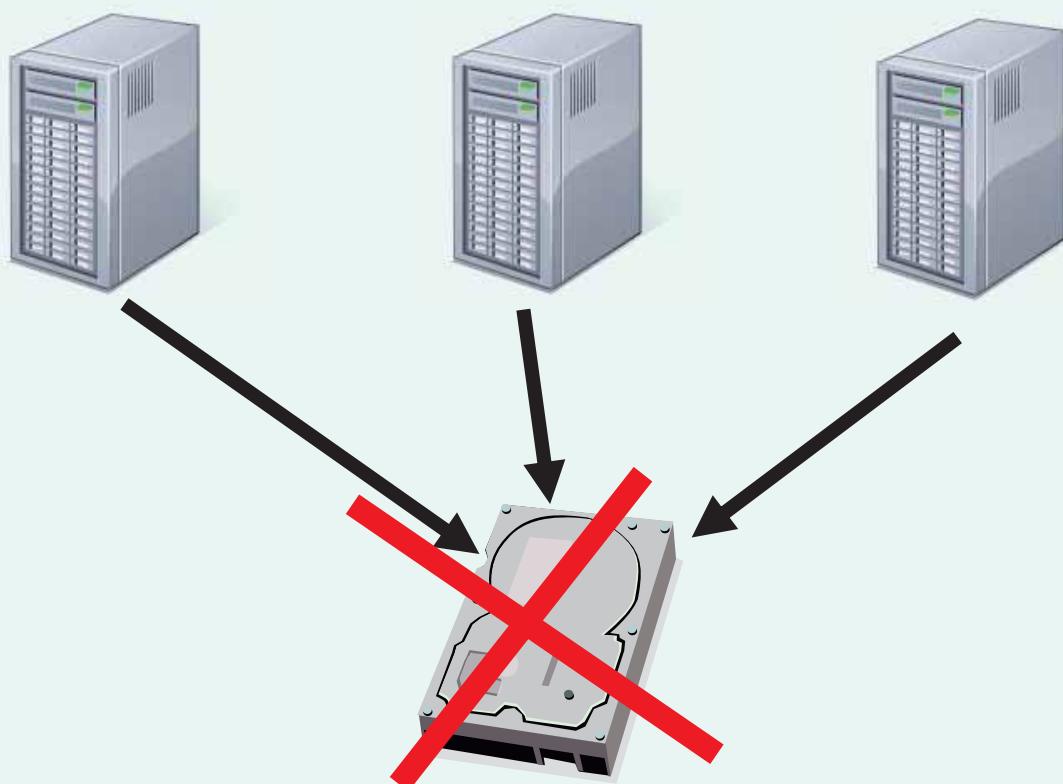




クラスタストレージとは

ディスクを冗長化するRAIDなどと同様に、データのコピーを行うものである。

また、共有ストレージのように SPF(Single Point Failure)とならない。





クラスタストレージの構築

ホスト名の確認、iptablesの無効化、SELinuxの無効化

The image displays two terminal windows side-by-side, both titled "192.168.11.61:22 - root@DRDB1:~ VT" and "192.168.11.62:22 - root@DRDB2:~ VT". Each window has a menu bar with Japanese labels: ファイル(E)、編集(E)、設定(S)、コントロール(O)、ウインドウ(W)、ヘルプ(H). The left terminal (DRDB1) shows the following commands:

```
[root@DRDB1 ~]# uname -n  
DRBD1  
[root@DRDB1 ~]# iptables -F  
[root@DRDB1 ~]# setenforce 0  
[root@DRDB1 ~]#  
[root@DRDB1 ~]# iptables-save > /etc/  
[root@DRDB1 ~]# vi /etc/sysconfig/iptables  
[root@DRDB1 ~]#
```

The right terminal (DRDB2) shows the following commands:

```
[root@DRDB2 ~]# uname -n  
DRBD2  
[root@DRDB2 ~]# iptables -F  
[root@DRDB2 ~]# setenforce 0  
[root@DRDB2 ~]#  
[root@DRDB2 ~]# iptables-save > /etc/sysconfig/iptables  
[root@DRDB2 ~]# vi /etc/sysconfig/selinux  
[root@DRDB2 ~]#
```



クラスタストレージの構築

2台目のハードディスクにパーティション(/dev/sdb1)を追加

※ファイルシステムは作成しません。

The screenshot shows two terminal windows side-by-side. Both windows are titled '192.168.11.61:22 - root@DRDB1:~ VT' and '192.168.11.62:22 - root@DRDB2:~ VT'. The left window (IP 192.168.11.61) displays the output of the 'ls -l /dev/sdb' command, which shows a single device entry: 'brw-rw----. 1 root disk 8, 16 7月 1'. Below this, the 'fdisk /dev/sdb' command is run, leading to a series of prompts for partition creation. The right window (IP 192.168.11.62) shows a similar sequence of commands and prompts, indicating the configuration of the second hard disk on both hosts.

```
[root@DRDB1 ~]# ls -l /dev/sdb
brw-rw----. 1 root disk 8, 16 7月 1
[root@DRDB1 ~]# fdisk /dev/sdb
デバイスは正常な DOS 領域テーブルも、
含んでいません
新たに DOS ディスクラベルをディスク識別子
あなたが書き込みを決定するまで、変更
その後はもちろん以前の内容は修復不可
警告：領域テーブル 4 の不正なフラグ
正常になります

警告：DOS互換モードは廃止予定です。こ
    強く推奨します。 and change dis
        sectors (command 'u')。

コマンド (m でヘルプ): n
コマンドアクション
    e 拡張
    p 基本パーティション (1-4)
p
パーティション番号 (1-4): 1
最初 シリンダ (1-261, 初期値 1):
初期値 1 を使います
Last シリンダ, +シリンダ数 or +size{K,M,G} (1-261, 初期値 261):
```



```
[root@DRDB2 ~]# ls -l /dev/sdb
brw-rw----. 1 root disk 8, 16 7月 11 15:31 2014 /dev/sdb
[root@DRDB2 ~]# fdisk /dev/sdb
デバイスは正常な DOS 領域テーブルも、Sun, SGI や OSF ディスクラベルも
含んでいません
新たに DOS ディスクラベルをディスク識別子 0x65ba1ec5 で作成します。
あなたが書き込みを決定するまで、変更はメモリ内だけに残します。
その後はもちろん以前の内容は修復不可能になります。
警告：領域テーブル 4 の不正なフラグ 0x0000 (は w(書き込み))によって
正常になります

警告：DOS互換モードは廃止予定です。このモード (コマンド 'c') を止めることを
    強く推奨します。 and change display units to
        sectors (command 'u')。

コマンド (m でヘルプ): n
コマンドアクション
    e 拡張
    p 基本パーティション (1-4)
p
パーティション番号 (1-4): 1
最初 シリンダ (1-261, 初期値 1):
初期値 1 を使います
Last シリンダ, +シリンダ数 or +size{K,M,G} (1-261, 初期値 261):
```



クラスタストレージの構築

e1repoレポジトリの追加

※Primary/Secondaryで実施。

```
192.168.11.61:22 - root@DRDB1:~ VT
[ファイル(E) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)]
[root@DRDB1 ~]# rpm -ivh http://elrepo.org/linux/elrepo/el6/i386/RPMS/elrepo-release-6-6.el6.elrepo.noarch.rpm
http://elrepo.org/linux/elrepo/el6/i386/RPMS/elrepo-release-6-6.el6.elrepo.noarch.rpm を取得中
警告: /var/tmp/rpm-tmp.Gj9phr: ヘッダ V4 DSA/SHA1 Signature, key ID baadae52: NO
KEY
準備中...          #####[100%]
1:elrepo-release      #####[100%]
[root@DRDB1 ~]# ls -l /etc/yum.repos.d/
合計 20
-rw-r--r--. 1 root root 1926 11月 27 20:52 2013 CentOS-Base.repo
-rw-r--r--. 1 root root  638 11月 27 20:52 2013 CentOS-Debuginfo.repo
-rw-r--r--. 1 root root  630 11月 27 20:52 2013 CentOS-Media.repo
-rw-r--r--. 1 root root 3664 11月 27 20:52 2013 CentOS-Vault.repo
-rw-r--r--. 1 root root 2150  2月 10 08:27 2014 elrepo.repo
[root@DRDB1 ~]#
```

主:



クラスタストレージの構築

drbd84-utilsとkmod-drbd84のインストール

```
192.168.11.62:22 - root@DRDB2:~ VT
[ファイル(E) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)]
[root@DRDB2 ~]# yum install drbd84-utils kmod-drbd84
Loaded plugins: fastestmirror, security
base                                         | 3.7 kB   00:00
base/primary_db                             | 3.5 MB   00:10
elrepo                                       | 2.9 kB   00:00
elrepo/primary_db                           | 656 kB   00:08
extras                                        | 3.4 kB   00:00
extras/primary_db                           | 18 kB    00:00
updates                                       | 3.4 kB   00:00
updates/primary_db                           | 3.4 MB   00:17
Setting up Install Process
Resolving Dependencies
--> Running transaction check
--> Package drbd84-utils.i686 0:8.4.4-2.el6.elrepo will be installed
--> Package kmod-drbd84.i686 0:8.4.4-1.el6.elrepo will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package           Arch      Version       Repository  Size
=====
Installing:
drbd84-utils     i686     8.4.4-2.el6.elrepo 289 k
```



クラスタストレージの構築

設定ファイル(/etc/drbd.conf)の確認

⇒/etc/drbd.d内のglobal_common.confと*.resがインクルードされていることを確認。

```
192.168.11.61:22 - root@DRDB1:~ VT
ファイル(E) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)
[root@DRDB1 ~]# cat /etc/drbd.conf
# You can find an example in /usr/share/doc/drbd.../drbd.conf.example

include "drbd.d/global_common.conf";
include "drbd.d/*.res";
[root@DRDB1 ~]#
```



クラスタストレージの構築

/etc/drbd.d/global_common.confを編集

usage-count

(<http://usage.drbd.org>での
統計情報への報告)

protocol

(ローカル/リモート両ディスクに
データ保存で書込完了)

local-io-error

(物理デバイスのIOエラー発生時
shutdown)

Primaryで作成後、
scpなどでSecondaryにコピーする

```
192.168.11.61:22 - root@DRDB1:/etc/drbd.d VT
[ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)]
[root@DRDB1 drbd.d]# cat /etc/drbd.d/global_common.conf
global {
    usage-count no;
}

common {
    protocol C;

    handlers {
        local-io-error "/usr/lib/drbd/notify-io-error.sh; /usr/lib/drbd/notify-emergency-shutdown.sh; echo o > /proc/sysrq-trigger ; halt -f";
    }
}
[root@DRDB1 drbd.d]#
```



http://usage.drbd.orgでの統計情報(参考情報)



usage.drbd.org/cgi-bin/show_usage.pl

Usage DRBD.org

On this site we try to estimate the number of DRBD installations. In case you deploy DRBD, please consider to take part in this process.

When you start drbd for the first time, its tools will ask you if you like to participate in this process. In case you agree a random number to <http://usage.drbd.org> and ist stored on your machine. In case you update your DRBD installation later, the same random number is used.

When you create DRBD's meta data on a block device, you also have the chance to take part in this process. As there are no other ways to estimate the number of installations.

The benefits for you are:

- As a response to your data, the server will tell you how many users before you have installed this version.
- With a high counter the DRBD developers have a high motivation to continue development of the software.

The numbers in the grey boxes are generated from the database every time you load the page, they are always up-to-date.

I assume that any serious user of DRBD will upgrade his cluster nodes at least once. So to estimate the number of DRBD installations (see the comparison with tgz download count below).

DRBD Statistics	
Number of node count events	589179
Number of resources	617853
Number of nodes that were upgraded at least once	110969
node with most resources, number	388134467724613651, 45071



クラスタストレージの構築

リソース設定ファイル(r0.res)の作成

Primaryで作成後、scpなどでSecondaryにコピーする

192.168.11.61:22 - root@DRDB1:/etc/drbd.d VT

```
ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)
[root@DRDB1 drbd.d]# cat r0.res
resource r0 {
    protocol C;
    device /dev/drbd0;
    disk /dev/sdb1;
    meta-disk internal;
    net {
        cram-hmac-alg sha1;
        shared-secret "password";
    }
    on DRBD1 {
        address 192.168.11.61:7777;
    }

    on DRBD2 {
        address 192.168.11.62:7777;
    }
}
[root@DRDB1 drbd.d]#
```



クラスタストレージの構築

r0.resの設定内容

cram-hmac-alg

(Master/Slave間の認証アルゴリズム)

shared-secret

(Master/Slave間の共通秘密鍵)

device

(DRBDリソースでのブロックデバイス名)

disk

(DRBD用に追加したブロックデバイス名)

meta-disk

(メタデータの格納に関する設定)

※メタデータとは、レプリケートするデータに関する情報の事

on DRBD1

(uname -nで表示される名前を設定)

address

(対向ノードからの接続を受付けるIPアドレスとポート)



クラスタストレージの構築

メタデータ領域の初期作成

```
drbdadm create-md r0
```

Primary/Secondaryで実施します。

```
[root@DRDB1 drbd.d]# drbdadm create-md r0
Writing meta data...
initializing activity log
NOT initializing bitmap
New drbd meta data block successfully created.
```

```
[root@DRDB2 drbd.d]# drbdadm create-md r0
Writing meta data...
initializing activity log
NOT initializing bitmap
New drbd meta data block successfully created.
```



クラスタストレージの構築

drbdに起動(Primary/Secondaryで実施)

```
[root@DRDB1 drbd.d]# /etc/init.d/drbd start
Starting DRBD resources: [
    create res: r0
    prepare disk: r0
    adjust disk: r0
    adjust net: r0
]
.
```

また、起動状況の確認も行う(Primary/Secondaryで実施)

```
[root@DRDB1 drbd.d]# netstat -tan | grep 7777
tcp        0      0 192.168.11.61:50480          192.168.11.62:7777          ESTABLISHED
tcp        0      0 192.168.11.61:7777          192.168.11.62:52571          ESTABLISHED
[root@DRDB1 drbd.d]# ps aux | grep drbd
root     31619  0.0  0.0      0      0 ?        S   17:22   0:00 [drbd-reissue]
root     31624  0.0  0.0      0      0 ?        S   17:22   0:00 [drbd_submit]
root     31632  0.0  0.0      0      0 ?        S   17:22   0:00 [drbd_w_r0]
root     31643  0.0  0.0      0      0 ?        S   17:22   0:00 [drbd_r_r0]
root     31648  0.0  0.0      0      0 ?        S   17:22   0:00 [drbd_a_r0]
root     31661  0.0  0.0  5436   812 pts/0       S+  17:24   0:00 grep drbd
```



クラスタストレージの構築

DRBDの状態を確認

/etc/init.d/drbd status

```
[root@DRDB1 drbd.d]# /etc/init.d/drbd status
drbd driver loaded OK; device status:
version: 8.4.4 (api:1/proto:86-101)
GIT-hash: 599f286440bd633d15d5ff985204aff4bccffadd build by phil@Build32R6, 2013-10-14 15:36:54
m:res cs ro ds p mounted fstype
0:r0 Connected Secondary/Secondary Inconsistent/Inconsistent C
```

cs(Connected)より、2ノード間での接続を確認
ro(Secondary/Seconday)より、Primaryが存在しないことを確認
Ds(Inconsistent/Inconsistent)より、データの未同期を確認

ちなみに、cat /proc/drbdでも同じような情報が確認出来ます。

```
[root@DRDB1 drbd.d]# cat /proc/drbd
version: 8.4.4 (api:1/proto:86-101)
GIT-hash: 599f286440bd633d15d5ff985204aff4bccffadd build by phil@Build32R6, 2013-10-14 15:36:54
0: cs:Connected ro:Secondary/Secondary ds:Inconsistent/Inconsistent C r-----
      ns:0 nr:0 dw:0 dr:0 al:0 bm:0 lo:0 pe:0 ua:0 ap:0 ep:1 wo:f oos:2096348
```



クラスタストレージの構築

初期状態では2台ともSecondary状態になっているので、Primaryとする方を強制的にPrimaryへと変更し、データの同期を行います。

```
[root@DRDB1 drbd.d]# drbdadm -- --overwrite-data-of-peer primary r0
[root@DRDB1 drbd.d]# /etc/init.d/drbd status
drbd driver loaded OK; device status:
version: 8.4.4 (api:1/proto:86-101)
GIT-hash: 599f286440bd633d15d5ff985204aff4bccffadd build by phil@Build32R6, 2013-10-14 15:36:54
m:res cs ro ds p mounted fstype
0:r0 SyncSource Primary/Secondary UpToDate/Inconsistent C
.. sync'ed: 1.4% (2069444/2096348)K
```

Primary/Secondaryとなり、データ同期中(PrimaryはSyncSourceと表示)

```
[root@DRDB1 drbd.d]# /etc/init.d/drbd status
drbd driver loaded OK; device status:
version: 8.4.4 (api:1/proto:86-101)
GIT-hash: 599f286440bd633d15d5ff985204aff4bccffadd build by phil@Build32R6, 2013-10-14 15:36:54
m:res cs ro ds p mounted fstype
0:r0 Connected Primary/Secondary UpToDate/UpToDate C
```



クラスタストレージの構築

Secondaryでの状態確認

```
[root@DRDB2 drbd.d]# /etc/init.d/drbd status
drbd driver loaded OK; device status:
version: 8.4.4 (api:1/proto:86-101)
GIT-hash: 599f286440bd633d15d5ff985204aff4bccffadd build by phil@Build32R6, 2013-10-14 15:36:54
m:res cs ro ds p mounted fstype
0:r0 SyncTarget Secondary/Primary Inconsistent/UpToDate C
.. sync'ed: 10.4% (1883492/2096348)K
```

Primary/Secondaryとなり、データ同期中(SecondaryはSyncTgargetと表示)

```
[root@DRDB2 drbd.d]# /etc/init.d/drbd status
drbd driver loaded OK; device status:
version: 8.4.4 (api:1/proto:86-101)
GIT-hash: 599f286440bd633d15d5ff985204aff4bccffadd build by phil@Build32R6, 2013-10-14 15:36:54
m:res cs ro ds p mounted fstype
0:r0 Connected Secondary/Primary UpToDate/UpToDate C
```



クラスタストレージの構築

Primaryにて、/dev/drbd0にファイルシステム(今回はext4を使用)を作成、
およびマウント

※/mnt/r0がマウントポイント

主題333

```
[root@DRDB1 drbd.d]# mkfs -t ext4 /dev/drbd0
mke2fs 1.41.12 (17-May-2010)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
131072 inodes, 524087 blocks
26204 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=536870912
16 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
      32768, 98304, 163840, 229376, 294912

Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

This filesystem will be automatically checked every 29 mounts or
180 days, whichever comes first.  Use tune2fs -c or -i to override.
```

```
[root@DRDB1 drbd.d]# mkdir /mnt/r0
[root@DRDB1 drbd.d]# mount -t ext4 /dev/drbd0 /mnt/r0
[root@DRDB1 drbd.d]# mount | grep /dev/drbd0
/dev/drbd0 on /mnt/r0 type ext4 (rw)
```



クラスタストレージの構築

Primaryにて、DRBDデバイスにファイル(drbd_testfile)作成
そして、Secondary状態に切り替える

drbdadm secondary r0

主題333: クラスタストレージより

```
[root@DRDB1 r0]# echo "Add by primary(192.168.11.61)" > /mnt/r0/drbd_testfile
[root@DRDB1 r0]# cd
[root@DRDB1 ~]# umount /mnt/r0
[root@DRDB1 ~]# drbdadm secondary r0
[root@DRDB1 ~]# /etc/init.d/drbd status
drbd driver loaded OK; device status:
version: 8.4.4 (api:1/proto:86-101)
GIT-hash: 599f286440bd633d15d5ff985204aff4bccffadd build by phil@Build32R6, 2013-10-14 15:36:54
m:res cs ro ds p mounted fstype
0:r0 Connected Secondary/Secondary UpToDate/UpToDate C
```



クラスタストレージの構築

DRBD2をPrimaryに昇格させる

```
[root@DRDB2 drbd.d]# mkdir /mnt/r0
[root@DRDB2 drbd.d]# drbdadm primary r0
[root@DRDB2 drbd.d]# /etc/init.d/drbd status
drbd driver loaded OK; device status:
version: 8.4.4 (api:1/proto:86-101)
GIT-hash: 599f286440bd633d15d5ff985204aff4bccffadd build by phil@Build32R6, 2013-10-14 15:36:54
m:res cs          ro          ds          p  mounted  fstype
0:r0  Connected Primary/Secondary UpToDate/UpToDate  C
[root@DRDB2 drbd.d]# mount -t ext4 /dev/drbd0 /mnt/r0
```

すると、drbd_testfileが同期されていることが確認出来ます

```
[root@DRDB2 drbd.d]# ls -l /mnt/r0/
合計 20
-rw-r--r--. 1 root root    30  7月 11 22:01 2014 drbd_testfile
drwx-----. 2 root root 16384  7月 11 21:49 2014 lost+found
[root@DRDB2 drbd.d]# cat /mnt/r0/drbd_testfile
Add by primary(192.168.11.61)
```



クラスタストレージの構築

DRBD2でdrbd_testfileへ追記を行った後、secondaryに降格させる

```
[root@DRDB2 drbd.d]# echo "Add by Slave(192.168.11.62)" >> /mnt/r0/drbd_testfile  
[root@DRDB2 drbd.d]# umount /mnt/r0  
[root@DRDB2 drbd.d]# drbdadm secondary r0
```

DRBD1を再度、primaryに昇格させデータ同期を確認

主題333: クラスタストレージにより

```
[root@DRDB1 ~]# drbdadm primary r0  
[root@DRDB1 ~]# mount -t ext4 /dev/drbd0 /mnt/r0  
[root@DRDB1 ~]# ls -l /mnt/r0/drbd_testfile  
-rw-r--r--. 1 root root 58 7月 11 22:08 2014 /mnt/r0/drbd_testfile  
[root@DRDB1 ~]# cat /mnt/r0/drbd_testfile  
Add by primary(192.168.11.61)  
Add by Slave(192.168.11.62)
```

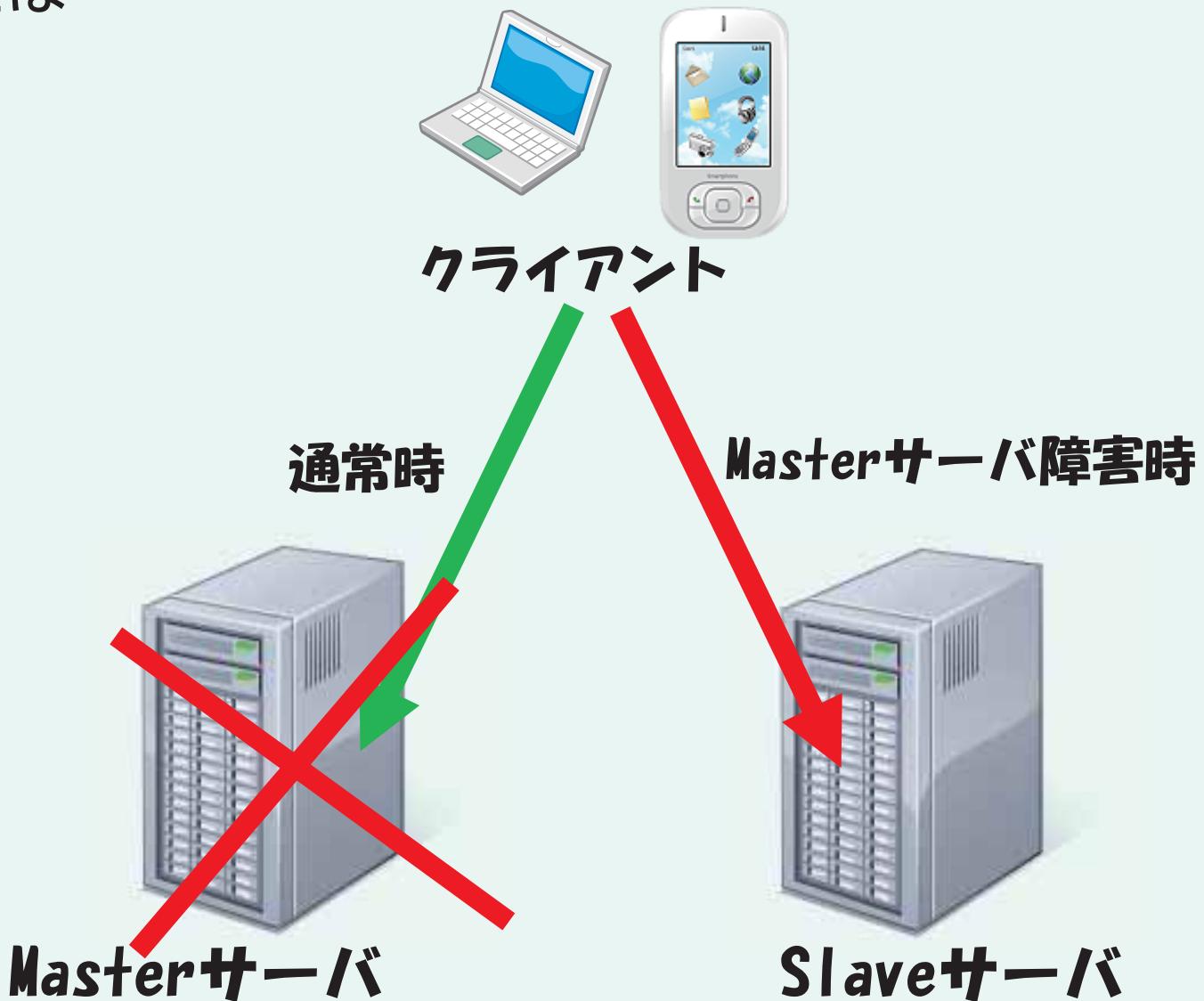


③技術解説

主題332: クラスタ管理



クラスタとは





heartbeat,corosync,pacemakerとは



Masterサーバ
(DRBD, apache)

192.168.11.61

仮想IP:192.168.11.60

M/S切替 (pacemaker)

Masterサーバ
(DRBD, apache)
仮想

IP: 192.168.11.60



相互監視
(heartbeatやcorosync)

Slaveサーバ
(DRBD, apache)

192.168.11.62



クラスタの構築

apacheのインストール

```
yum -y install httpd
```

```
[root@DRBD1 ~]# yum -y install httpd
Loaded plugins: fastestmirror, security
Loading mirror speeds from cached hostfile
 * base: ftp.tsukuba.wide.ad.jp
 * elrepo: ftp.ne.jp
 * extras: ftp.tsukuba.wide.ad.jp
 * updates: ftp.tsukuba.wide.ad.jp
base                                         | 3.7 kB     00:00
elrepo                                       | 2.9 kB     00:00
extras                                        | 3.4 kB     00:00
updates                                       | 3.4 kB     00:00
Setting up Install Process
Resolving Dependencies
--> Running transaction check
--> Package httpd.i686 0:2.2.15-30.el6.centos will be installed
--> Processing Dependency: httpd-tools = 2.2.15-30.el6.centos for package: httpd-2.2.15-30.el6.centos.i686
```



クラスタの構築

/var/www/htmlと/mnt/r0/htmlをリンク

Primary操作

```
[root@DRBD1 ~]# mv /var/www/html /mnt/r0
[root@DRBD1 ~]# ls -l /mnt/r0/
合計 24
-rw-r--r--. 1 root root 58 7月 11 22:08 2014 drbd_testfile
drwxr-xr-x 2 root root 4096 4月  4 08:54 2014 html
drwx----- 2 root root 16384 7月 11 21:49 2014 lost+found
[root@DRBD1 ~]# ln -s /mnt/r0/html /var/www/html
[root@DRBD1 ~]# ls -l /var/www/html
lrwxrwxrwx 1 root root 12 7月 15 20:32 2014 /var/www/html -> /mnt/r0/html
```

Secondary操作

```
[root@DRBD2 ~]# rm -rf /var/www/html
[root@DRBD2 ~]# ln -s /mnt/r0/html /var/www/html
```



クラスタの構築

Primaryにて、httpdのコンテンツファイル(index.html)を作成

```
[root@DRBD1 ~]# touch /var/www/html/index.html
[root@DRBD1 ~]# vi /var/www/html/index.html
[root@DRBD1 ~]# cat /var/www/html/index.html
<!doctype html>
<html>
<head>
<title>Test cluster html</title>
</head>
<body>
<p>Test cluster html's document!</p>
</body>
</html>
[root@DRBD1 ~]# ls -l /mnt/r0/html/
合計 4
-rw-r--r-- 1 root root 131 7月 15 20:37 2014 index.html
```



クラスタの構築

heartbeatのインストール

```
rpm -ivh http://ftp-srv2.kddilabs.jp/Linux/packages/CentOS/6.6/extras/i386/Packages/epel-release-6-8.noarch.rpm
```

```
yum install heartbeat
```

```
[root@DRBD1 ~]# rpm -ivh http://ftp-srv2.kddilabs.jp/Linux/packages/fedora/epel/6/i386/epel-release-6-8.noarch.rpm  
http://ftp-srv2.kddilabs.jp/Linux/packages/fedora/epel/6/i386/epel-release-6-8.noarch.rpm を取得中  
警告: /var/tmp/rpm-tmp.rzQ0xW: ヘッダ V3 RSA/SHA256 Signature, key ID 0608b895:  
NOKEY  
準備中...          ##### [100%]  
1:epel-release     ##### [100%]
```

```
[root@DRBD1 ~]# yum install heartbeat  
Loaded plugins: fastestmirror, security  
Loading mirror speeds from cached hostfile  
epel/metalink  
* base: www.ftp.ne.jp  
* elrepo: ftp.ne.jp  
* epel: ftp.iij.ad.jp  
* extras: www.ftp.ne.jp  
* updates: www.ftp.ne.jp  
epel  
epel/primary_db  
Setting up Install Process  
Resolving Dependencies  
--> Running transaction check  
--> Package heartbeat.i686 0:3.0.4-2.el6 will be installed  
--> Processing Dependency: resource-agents for package: heartbeat-3.0.4-2.el6.i686
```



クラスタの構築

rpm -qi heartbeatで情報確認

```
[root@DRBD1 ~]# rpm -qi heartbeat
Name        : heartbeat
Version     : 3.0.4
Release     : 2.el6
7秒
Relocations: (not relocatable)
Vendor: Fedora Project
Build Date: 2013年12月03日 01時37分2
Install Date: 2014年07月15日 20時50分08秒      Build Host: buildvms-15.phx2.fedoraproject.org
Group       : System Environment/Daemons      Source RPM: heartbeat-3.0.4-2.el6.src
c.rpm
Size        : 263978                         License: GPLv2 and LGPLv2+
Signature   : RSA/8, 2013年12月03日 07時59分16秒, Key ID 3b49df2a0608b895
Packager    : Fedora Project
URL         : http://linux-ha.org/
Summary     : Messaging and membership subsystem for High-Availability Linux
Description :
heartbeat is a basic high-availability subsystem for Linux-HA.
It will run scripts at initialization, and when machines go up or down.
This version will also perform IP address takeover using gratuitous ARPs.

Heartbeat contains a cluster membership layer, fencing, and local and
clusterwide resource management functionality.
```



LPIC 304 技術解説セミナー



クラスタの構築

<http://sourceforge.jp/projects/linux-ha/releases/>より、
pacemaker-1.0.13-2.1.el6.i686.repo.tar.gzを/tmpにダウンロード

The screenshot shows the SourceForge project page for Linux-HA Japan. The URL in the address bar is sourceforge.jp/projects/linux-ha/releases/60151. The page title is "Linux-HA Japan". The main content area displays the "ダウンロードリスト" (Download List) for the "[1-01] Pacemakerリポジトリパッケージ (RHEL6) 1.0.13-1.2" package. The list includes several files:

名前	サイズ	MD5	日付	ダウンロード数
pacemaker-debuginfo-1.0.13-1.2.el6.x86_64.repo.tar.gz	4.8 MB	76df5376306a6...	2013-12-25 17:05	84
pacemaker-debuginfo-1.0.13-1.2.el6.i686.repo.tar.gz	4.8 MB	77cb2cfb78e23...	2013-12-25 17:05	25
pacemaker-1.0.13-1.2.el6.x86_64.repo.tar.gz	8.9 MB	edde5138db6f2...	2013-12-25 17:05	1816
pacemaker-1.0.13-1.2.el6.srpm.tar.gz	21.2 MB	b77abfc6bb1d8...	2013-12-25 17:05	132
pacemaker-1.0.13-1.2.el6.i686.repo.tar.gz	3.9 MB	edae559294c2...	2013-12-25 17:05	308

```
[root@DRBD1 ~]# ls /tmp  
pacemaker-1.0.13-1.2.el6.i686.repo.tar.gz yum.log
```



クラスタの構築

ダウンロードファイルを解凍すると、
作成されたディレクトリがローカルレポジトリとなっていることが確認出来る。

```
[root@DRBD1 tmp]# ls -lR pacemaker-1.0.13-1.2.el6.i686.repo
pacemaker-1.0.13-1.2.el6.i686.repo:
合計 12
-rw-r--r-- 1 root root 105 12月 24 13:14 2013 pacemaker.repo
drwxr-xr-x 2 root root 4096 12月 25 09:53 2013 repodata
drwxr-xr-x 2 root root 4096 12月 24 13:25 2013 rpm

pacemaker-1.0.13-1.2.el6.i686.repo/repodata:
合計 92
-rw-r--r-- 1 root root 9815 12月 25 09:53 2013 1277b605cd6b6ce1bde62fe3645886205d48f7c0d6cf79836aeebc0f192f72b9-primary.xml.gz
-rw-r--r-- 1 root root 18598 12月 25 09:53 2013 5ed69cdaed012f531fe7aff924174d46a1552f62547bd4d5b8c0ecb5f2d0eb77-filelists.sqlite.bz2
-rw-r--r-- 1 root root 4016 12月 25 09:53 2013 8886049b57fcc5e3f5527893d30891826164c3fbcb8af212938b4c656b37a3a7-other.xml.gz
-rw-r--r-- 1 root root 25241 12月 25 09:53 2013 8a5c6fac8ffab28a706e049eb301bdd35e0811a92c68712f6e780ffb11b87d7c-primary.sqlite.bz2
-rw-r--r-- 1 root root 7226 12月 25 09:53 2013 8c17c384796129a52638431b7f806bd91f7dbb5d0c3e44d7a4b44052940a2118-other.sqlite.bz2
-rw-r--r-- 1 root root 13892 12月 25 09:53 2013 c2733e261c05ae7e27059d5b8f07f2fedb9359f4ab2d3b25c8e5cf3aa45b7e3-filelists.xml.gz
-rw-r--r-- 1 root root 2988 12月 25 09:53 2013 repomd.xml

pacemaker-1.0.13-1.2.el6.i686.repo/rpm:
合計 4212
-rw-r--r-- 1 root root 254772 12月 24 13:14 2013 cluster-glue-1.0.11-1.el6.i686.rpm
-rw-r--r-- 1 root root 109040 12月 24 13:14 2013 cluster-glue-libs-1.0.11-1.el6.i686.rpm
-rw-r--r-- 1 root root 109784 12月 24 13:14 2013 cluster-glue-libs-devel-1.0.11-1.el6.i686.rpm
-rw-r--r-- 1 root root 167008 12月 24 13:14 2013 corosync-1.4.6-1.el6.i686.rpm
```



クラスタの構築

pacemakerのインストール

```
yum -c /tmp/pacemaker-1.0.13-2.1.el6.i686.repo/pacemaker.repo install pacemaker-1.0.13  
pm_extras
```

```
[root@DRBD1 tmp]# yum -c /tmp/pacemaker-1.0.13-1.2.el6.i686.repo/pacemaker.repo install pacemaker-1.0.13  
pm_extras  
base | 3.7 kB 00:00  
base/primary_db | 3.5 MB 00:05  
el repo | 2.9 kB 00:00  
el repo/primary_db | 656 kB 00:05  
epel/metalink | 5.2 kB 00:00  
epel | 4.4 kB 00:00  
epel/primary_db | 5.2 MB 00:10  
extras | 3.4 kB 00:00  
extras/primary_db | 18 kB 00:00  
pacemaker | 2.9 kB 00:00 ...  
pacemaker/primary_db | 25 kB 00:00 ...  
updates | 3.4 kB 00:00  
updates/primary_db | 3.4 MB 00:06  
  
Setting up Install Process  
Resolving Dependencies  
--> Running transaction check  
--> Package pacemaker.i686 0:1.0.13-1.el6 will be installed  
--> Processing Dependency: pacemaker-libs = 1.0.13-1.el6 for package: pacemaker-1.0.13-1.el6.i686  
--> Processing Dependency: libxslt.so.1 for package: pacemaker-1.0.13-1.el6.i686
```



クラスタの構築

pacemakerのインストール情報を確認

```
[root@DRBD1 tmp]# rpm -qi pacemaker
Name        : pacemaker
Version     : 1.0.13
Release     : 1.el6
Install Date: 2014年07月15日 21時02分08秒
Group       : System Environment/Daemons
Size        : 2785208
Signature   : (none)
URL         : http://www.clusterlabs.org
Summary     : Scalable High-Availability cluster resource manager
Description :
Pacemaker is an advanced, scalable High-Availability cluster resource
manager for Linux-HA (Heartbeat) and/or OpenAIS.

It supports "n-node" clusters with significant capabilities for
managing resources and dependencies.

It will run scripts at initialization, when machines go up or down,
when related resources fail and can be configured to periodically check
resource health.

Available rpmbuild rebuild options:
  --without : heartbeat ais esmtp snmp
```



クラスタの構築

heartbeatの設定ファイル及び認証用ファイルの雛形ファイルを
コピーする

```
cp -p /usr/share/doc/heartbeat-3.0.4/ha.cf /etc/ha.d/ha.cf
cp -p /usr/share/doc/heartbeat-3.0.4/authkeys /etc/ha.d/authkeys
```

```
[root@DRBD1 tmp]# ls -l /usr/share/doc/heartbeat-3.0.4/ha.cf
-rw-r--r-- 1 root root 10502 12月  3 01:37 2013 /usr/share/doc/heartbeat-3.0.4/ha.cf
[root@DRBD1 tmp]# cp -p /usr/share/doc/heartbeat-3.0.4/ha.cf /etc/ha.d/ha.cf
[root@DRBD1 tmp]# ls -l /usr/share/doc/heartbeat-3.0.4/authkeys
-rw-r--r-- 1 root root 645 12月  3 01:37 2013 /usr/share/doc/heartbeat-3.0.4/authkeys
[root@DRBD1 tmp]# cp -p /usr/share/doc/heartbeat-3.0.4/authkeys /etc/ha.d/authkeys
```



クラスタの構築

heartbeatの設定ファイルを編集

※scpなどでDRBD2にコピーする

```
[root@DRBD1 tmp]# vi /etc/ha.d/ha.cf
[root@DRBD1 tmp]# grep -v "^#" /etc/ha.d/ha.cf
pacemaker yes
logfacility    local1
keepalive 2
deadtime 30
warntime 10
initdead 120
udpport 694
bcast   eth1          # Linux
auto_failback on
node    DRBD1
node    DRBD2
```

なお、nodeで記載する名前については、uname -nと一致させる

```
[root@DRBD1 tmp]# uname -n
DRBD1
```



クラスタの構築

ha.cfの設定内容

pacemaker ····· pacemakerの使用

logfacility ····· 出力ログのファシリティ

keepalive ····· 相互確認の間隔

deadtime ····· 相手ノードがダウンと判断する時間

warn time ····· 相手ノードがダウンした警告を出力するまでの時間

initdead ····· 起動時にheartbeatによる監視を行うまでの時間
(OS起動正常起動までの十分な時間を指定)

udpport ····· 使用ポート

bcast ····· heartbeatをブロードキャストで行う場合のインターフェイス指定

auto_fallback ···· 自動フェイルバックの設定

node ····· クラスタを組むノード名(uname -nで表示されるもの)



クラスタの構築

rsyslogの設定追加 及びrsyslogの再起動

```
[root@DRBD1 tmp]# vi /etc/rsyslog.conf
[root@DRBD1 tmp]# grep "local1" /etc/rsyslog.conf
local1.*                                     /var/log/ha.log
```

```
[root@DRBD1 tmp]# /etc/init.d/rsyslog restart
システムロガーを停止中:                      [ OK ]
システムロガーを起動中:                      [ OK ]
```

これで、HAのログがtail -f /var/log/ha.logで監視可能となる。



クラスタの構築

authkeysの編集及び権限変更(600)

※scpなどでSecondaryにコピーする

```
[root@DRBD1 tmp]# vi /etc/ha.d/authkeys
[root@DRBD1 tmp]# grep -v "^\#" /etc/ha.d/authkeys
auth 2
2 sha1 password
```

```
[root@DRBD1 tmp]# ls -l /etc/ha.d/authkeys
-rw-r--r-- 1 root root 660 7月 15 21:19 2014 /etc/ha.d/authkeys
[root@DRBD1 tmp]# chmod 600 /etc/ha.d/authkeys
[root@DRBD1 tmp]# ls -l /etc/ha.d/authkeys
-rw----- 1 root root 660 7月 15 21:19 2014 /etc/ha.d/authkeys
```



クラスタの構築

Primary、Secondaryの順でheartbeatを起動する

```
[root@DRBD1 tmp]# /etc/init.d/heartbeat start
Starting High-Availability services: Done.
```

```
[root@DRBD2 tmp]# /etc/init.d/heartbeat start
Starting High-Availability services: Done.
```

すると、/var/lib/heartbeat/crm配下にcib.xmlなどが生成される

```
[root@DRBD1 tmp]# ls -l /var/lib/heartbeat/crm/
合計 36
-rw----- 1 hacluster haclient 232 7月 15 21:22 2014 cib-0.raw
-rw-r--r-- 1 hacluster haclient 32 7月 15 21:22 2014 cib-0.raw.sig
-rw----- 1 hacluster haclient 572 7月 15 21:23 2014 cib-1.raw
-rw-r--r-- 1 hacluster haclient 32 7月 15 21:23 2014 cib-1.raw.sig
-rw----- 1 hacluster haclient 752 7月 15 21:23 2014 cib-2.raw
-rw-r--r-- 1 hacluster haclient 32 7月 15 21:23 2014 cib-2.raw.sig
-rw-r--r-- 1 hacluster haclient 1 7月 15 21:23 2014 cib.last
-rw----- 1 hacluster haclient 815 7月 15 21:23 2014 cib.xml
-rw-r--r-- 1 hacluster haclient 32 7月 15 21:23 2014 cib.xml.sig
```



クラスタの構築

Primaryにて、crm_mon -rfAにて、クラスタ状態を確認

```
[root@DRBD1 tmp]# crm_mon -rfA
```

```
=====
Last updated: Tue Jul 15 21:24:22 2014
Stack: Heartbeat
Current DC: drbd2 (6b8910e9-00ca-43b4-8836-a10956005982) - partition with quorum
Version: 1.0.13-30bb726
2 Nodes configured, unknown expected votes
0 Resources configured.
=====

Online: [ drbd1 drbd2 ]

Full list of resources:

Node Attributes:
* Node drbd1:
* Node drbd2:

Migration summary:
* Node drbd1:
* Node drbd2:
```



クラスタの構築

Secondaryでheartbeatを停止すると、状態が**OFFLINE**に変更される。

```
[root@DRBD2 tmp]# /etc/init.d/heartbeat stop  
Stopping High-Availability services: Done.
```

```
[root@DRBD1 tmp]# crm_mon -rfA
```

```
=====  
Last updated: Tue Jul 15 21:26:17 2014  
Stack: Heartbeat  
Current DC: drbd1 (9dc26ec6-3a08-4bdf-b52e-5ae673e99f03) - partition with quorum  
Version: 1.0.13-30bb726  
2 Nodes configured, unknown expected votes  
0 Resources configured.  
=====
```

```
Online: [ drbd1 ]  
OFFLINE: [ drbd2 ]
```

```
Full list of resources:
```

```
Node Attributes:  
* Node drbd1:
```

```
Migration summary:  
* Node drbd1:
```



クラスタの構築

また、cluster2のheartbeatのみが起動している場合は、
以下のようになる

```
=====
Last updated: Tue Jul 15 21:27:55 2014
Stack: Heartbeat
Current DC: drbd2 (6b8910e9-00ca-43b4-8836-a10956005982) - partition with quorum

Version: 1.0.13-30bb726
2 Nodes configured, unknown expected votes
0 Resources configured.
=====

Online: [ drbd2 ]
OFFLINE: [ drbd1 ]

Full list of resources:

Node Attributes:
* Node drbd2:

Migration summary:
* Node drbd2:
```



クラスタの構築

また、ha.cfに

respawn root /usr/lib/heartbeat/ifcheckd

(実行する外部プログラムと実行ユーザーを指定)を追加すると、
※heartbeatを再起動します。

```
[root@DRBD1 tmp]# vi /etc/ha.d/ha.cf
[root@DRBD1 tmp]# grep -v "^#" /etc/ha.d/ha.cf
pacemaker yes
logfacility    local1
keepalive 2
deadtime 30
warnetime 10
initdead 120
udpport 694
bcast   eth0          # Linux
auto_failback on
node    DRBD1
node    DRBD2
[redacted]
respawn root /usr/lib/heartbeat/ifcheckd
```



クラスタの構築

eth1の監視状態が追加される。

```
=====
Last updated: Tue Jul 15 21:47:03 2014
Stack: Heartbeat
Current DC: drbd2 (6b8910e9-00ca-43b4-8836-a10956005982) - partition with quorum
Version: 1.0.13-30bb726
2 Nodes configured, unknown expected votes
0 Resources configured.
=====

Online: [ drbd1 drbd2 ]

Full list of resources:

Node Attributes:
* Node drbd1:
  + drbd2-eth1 : up
* Node drbd2:
  + drbd1-eth1 : up

Migration summary:
* Node drbd1:
* Node drbd2:
```



クラスタの構築

続いて、apacheに関する監視設定を追加します。

pacemakerのhttpd監視では、server-statusを使用するので、
httpd.confの以下の部分を有効化します。

```
<Location /server-status>
    SetHandler server-status
    Order deny,allow
    Deny from all
    Allow from 127.0.0.1
</Location>
```



クラスタの構築

pacemaker全体の設定

```
[root@DRBD1 tmp]#crm
crm(live)#configure
INFO: building help index
crm(live)configure#property stonith-enabled="false"
crm(live)configure#property no-quorum-policy="ignore"
crm(live)configure#property default-resource-stickiness="200"
```

リソースの設定(DRBD)

```
crm(live)configure#primitive res_drbd0 oct:linbit:drbd params drbd_resource="r0" drbdconf="/etc/drbd.conf" op monitor interval="20s"
WARNING: res_drbd0: default timeout 20s for start is smaller than the advised 240
WARNING: res_drbd0: default timeout 20s for stop is smaller than the advised 100
WARNING: res_drbd0: action monitor not advertised in meta-data, it may not be supported by the RA
crm(live)configure#ms ms_drbd0 res_drbd0 meta master-max="1" master-node-max="1" clone-max="2" clone-node-max="1" notify="true"
```



クラスタの構築

リソースの設定(ファイルシステム)

```
crm(live)configure# primitive res_filesystem ocf:heartbeat:Filesystem params device="/dev/drbd0" fstype="ext4" directory="/mnt/r0" op monitor interval="20s"
WARNING: res_filesystem: default timeout 20s for start is smaller than the advised 60
WARNING: res_filesystem: default timeout 20s for stop is smaller than the advised 60
WARNING: res_filesystem: default timeout 20s for monitor is smaller than the advised 40
```

リソースの設定(apache)

```
crm(live)configure# primitive res_httpd ocf:heartbeat:apache params configfile="/etc/httpd/conf/httpd.conf" port="80" op monitor interval="20s"
WARNING: res_httpd: default timeout 20s for start is smaller than the advised 40s
WARNING: res_httpd: default timeout 20s for stop is smaller than the advised 60s
```

リソースの設定(仮想IP)

```
crm(live)configure# primitive res_ip ocf:heartbeat:IPaddr2 params ip="192.168.11.60" cidr_netmask="24" nic="eth0" op monitor interval="10s"
```



クラスタの構築

リソースグループの設定

```
crm(live)configure# group rg_http res_ip res_filesystem res_httpd
```

プログラムの起動順番と、リソースの関連性に関する設定

```
crm(live)configure# colocation c_rg_http_on_drbd0 inf: rg_http ms_drbd0:Master  
crm(live)configure# order o_drbd_before_rg_http inf: ms_drbd0:promote rg_http:start
```



クラスタの構築

設定内容のチェック、コミット(確定)

```
crm(live)configure# verify
WARNING: res_drbd0: default timeout 20s for start is smaller than the advised 240
WARNING: res_drbd0: default timeout 20s for stop is smaller than the advised 100
WARNING: res_drbd0: action monitor not advertised in meta-data, it may not be supported by the RA
WARNING: res_filesystem: default timeout 20s for start is smaller than the advised 60
WARNING: res_filesystem: default timeout 20s for stop is smaller than the advised 60
WARNING: res_filesystem: default timeout 20s for monitor is smaller than the advised 40
WARNING: res_httpd: default timeout 20s for start is smaller than the advised 40s
WARNING: res_httpd: default timeout 20s for stop is smaller than the advised 60s
crm(live)configure# commit
WARNING: res_drbd0: default timeout 20s for start is smaller than the advised 240
WARNING: res_drbd0: default timeout 20s for stop is smaller than the advised 100
WARNING: res_drbd0: action monitor not advertised in meta-data, it may not be supported by the RA
WARNING: res_filesystem: default timeout 20s for start is smaller than the advised 60
WARNING: res_filesystem: default timeout 20s for stop is smaller than the advised 60
WARNING: res_filesystem: default timeout 20s for monitor is smaller than the advised 40
WARNING: res_httpd: default timeout 20s for start is smaller than the advised 40s
WARNING: res_httpd: default timeout 20s for stop is smaller than the advised 60s
exicrm(live)configure# exit
bye
```



クラスタの構築

crm > configure > showとすることで、以下のように設定を表示することも出来ます。

```
node $id="6b8910e9-00ca-43b4-8836-a10956005982" drbd2
node $id="9dc26ec6-3a08-4bdf-b52e-5ae673e99f03" drbd1
primitive res_drbd0 ocf:linbit:drbd ¥
    params drbd_resource="r0" drbdconf="/etc/drbd.conf" ¥
    op monitor interval="20s"
primitive res_filesystem ocf:heartbeat:Filesystem ¥
    params device="/dev/drbd0" fstype="ext4" directory="/mnt/r0" ¥
    op monitor interval="20s"
primitive res_httpd ocf:heartbeat:apache ¥
    params configfile="/etc/httpd/conf/httpd.conf" port="80" ¥
    op monitor interval="20s"
primitive res_ip ocf:heartbeat:IPaddr2 ¥
    params ip="192.168.11.60" cidr_netmask="24" nic="eth0" ¥
    op monitor interval="10s"
group rg_http res_ip res_filesystem res_httpd
ms ms_drbd0 res_drbd0 ¥
    meta master-max="1" master-node-max="1" clone-max="2" clone-node-max="1" notify="true"
colocation c_rg_http_on_drbd0 inf: rg_http ms_drbd0:Master
order o_drbd_before_rg_http inf: ms_drbd0:promote rg_http:start
property $id="cib-bootstrap-options" ¥
    dc-version="1.0.13-30bb726" ¥
    cluster-infrastructure="Heartbeat" ¥
    stonith-enabled="false" ¥
    no-quorum-policy="ignore" ¥
    default-resource-stickiness="200"
~
~
```

(END)



クラスタの構築

設定内容変更の確認

```
[root@DRBD1 tmp]# crm_mon -rfA
```

```
=====
Last updated: Tue Jul 15 22:20:38 2014
Stack: Heartbeat
Current DC: drbd2 (6b8910e9-00ca-43b4-8836-a10956005982) - partition with quorum
Version: 1.0.13-30bb726
2 Nodes configured, unknown expected votes
2 Resources configured.
=====

Online: [ drbd1 drbd2 ]

Full list of resources:

Master/Slave Set: ms_drbd0
  Masters: [ drbd1 ]
  Slaves: [ drbd2 ]
Resource Group: rg_http
  res_ip    (ocf::heartbeat:IPaddr2):     Started drbd1
  res_filesystem    (ocf::heartbeat:Filesystem):   Started drbd1
  res_httpd   (ocf::heartbeat:apache):      Started drbd1

Node Attributes:
* Node drbd1:
  + drbd2-eth1           : up
  + master-res_drbd0:0   : 10000
* Node drbd2:
  + drbd1-eth1           : up
  + master-res_drbd0:1   : 10000

Migration summary:
* Node drbd1:
* Node drbd2:
```



クラスタの構築

フェイルオーバー動作の確認

crm resource migrateを使用

```
[root@DRBD1 tmp]# crm
crm(live)# resource
crm(live)resource# migrate rg_http
WARNING: Creating rsc_location constraint 'cli-standby-rg_http' with a score of -INFINITY for resource rg_http on drbd1.
        This will prevent rg_http from running on drbd1 until the constraint is removed using the 'crm_resource -U' command or manually with cibadmin
        This will be the case even if drbd1 is the last node in the cluster
        This message can be disabled with -Q
```



クラスタの構築

DRBD2がPrimaryとなる

```
=====
Last updated: Tue Jul 15 22:24:15 2014
Stack: Heartbeat
Current DC: drbd2 (6b8910e9-00ca-43b4-8836-a10956005982) - partition with quorum
Version: 1.0.13-30bb726
2 Nodes configured, unknown expected votes
2 Resources configured.
=====

Online: [ drbd1 drbd2 ]

Full list of resources:

Master/Slave Set: ms_drbd0
  Masters: [ drbd2 ]
  Slaves: [ drbd1 ]
Resource Group: rg_http
  res_ip    (ocf::heartbeat:IPAddr2):      Started drbd2
  res_filesystem    (ocf::heartbeat:Filesystem):   Started drbd2
  res_apache (ocf::heartbeat:apache):       Started drbd2

Node Attributes:
* Node drbd1:
  + drbd2-eth1          : up
  + master-res_drbd0:0  : 10000
* Node drbd2:
  + drbd1-eth1          : up
  + master-res_drbd0:1  : 10000

Migration summary:
* Node drbd1:
* Node drbd2:
```



クラスタの構築

フェイルオーバー動作の確認が出来たら、
DRBD1をクラスタに戻しておきます。

crm resource unmigrateを使用

```
crm(live)resource# unmigrate rg_http
crm(live)resource# exit
bye
```



クラスタの構築

なお、chkconfigでdrbd/httpdは自動起動しないようにしておく。
※pacemaker/heartbeatによる切替が実施される為

```
[root@DRBD1 ~]# chkconfig httpd off
[root@DRBD1 ~]# chkconfig drbd off
[root@DRBD1 ~]# chkconfig --list | grep httpd
httpd           0:off   1:off   2:off   3:off   4:off   5:off   6:off
[root@DRBD1 ~]# chkconfig --list | grep drbd
drbd           0:off   1:off   2:off   3:off   4:off   5:off   6:off
```



スプリットブレインとは
「相互監視が出来なくなった場合に、
両方ともMasterサーバになってしまう現象」

Masterサーバ
(DRBD, apache)
仮想IP: 192. 168. 11. 60



Masterサーバ
(DRBD, apache)
仮想IP: 192. 168. 11. 60



Masterサーバ
(DRBD, apache)
192. 168. 11. 61



Slaveサーバ
(DRBD, apache)
192. 168. 11. 62



STONITHとは

「スプリットブレインにならないように、

相互監視が出来なくなったら相手を強制停止等を行う機能」

※専用デバイス等が必要となります。

Masterサーバ

(DRBD, apache)

仮想IP: 192. 168. 11. 60



相互監視

強制停止等



Masterサーバ

(DRBD, apache)

192. 168. 11. 61

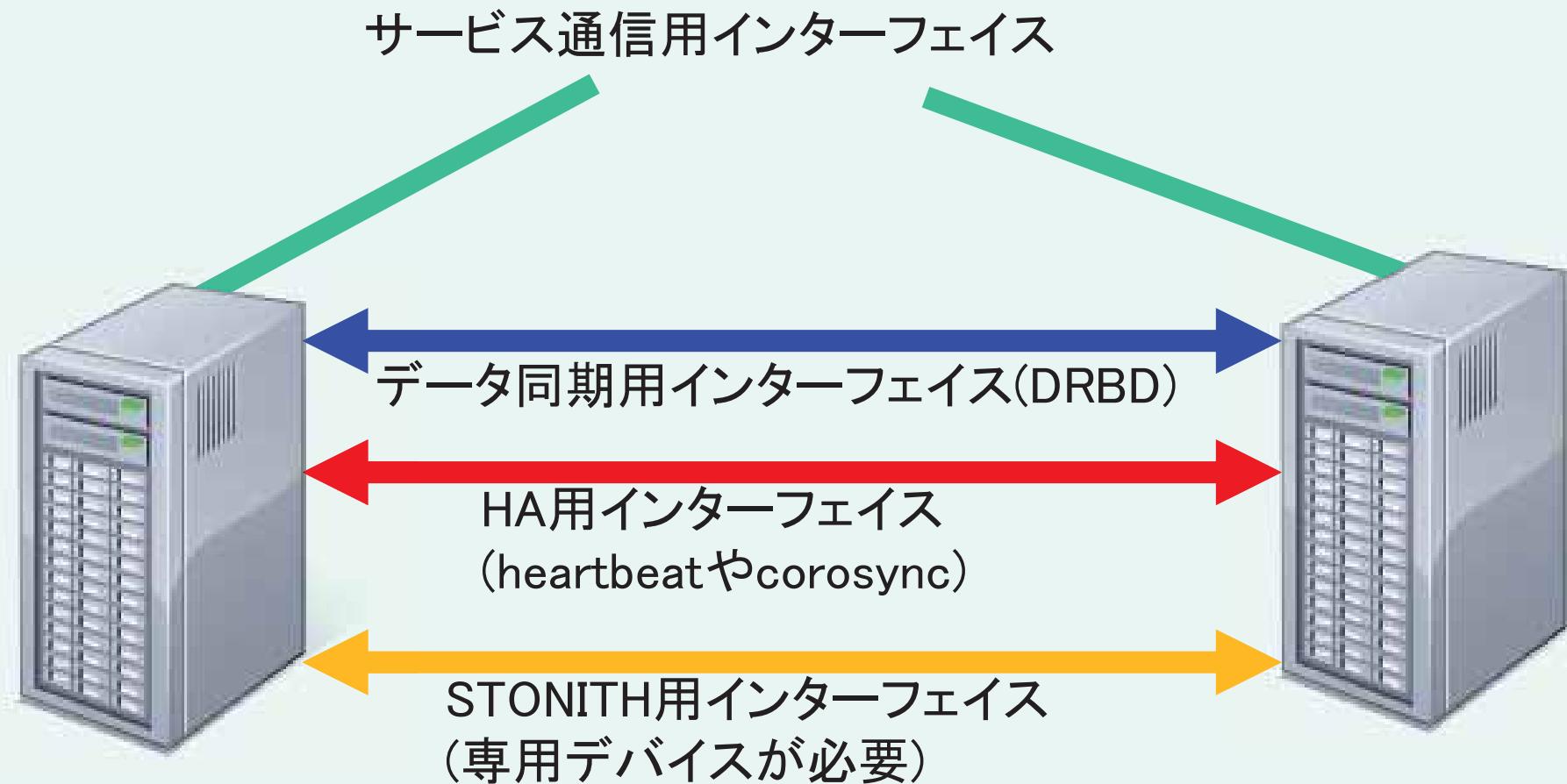
Slaveサーバ

(DRBD, apache)

192. 168. 11. 62



前述までの要素を全て考慮すると、
以下のような構成になる。





LPIC 304 技術解説セミナー



heartbeat/pacemakerについてもっと詳しく知るには、
Linux-HAのページが非常に役に立ちます。
<http://linux-ha.sourceforge.jp/wp/>



LINUX-HA JAPAN
High-Availability Clustering on Linux

Posts Comments

HOME メーリングリスト ダウンロード&インストール マニュアル デスクトップテーマ・壁紙等 コミュニティ概要 関連リンク
その他 ニュース イベント情報 読み物 WEBラジオ

Linux-HA Japan プロジェクト

127 Check

Linux上で高可用(HA)クラスタシステムを構築するための部品として、オープンソースの、クラスタリソースマネージャ、クラスタ通信レイヤ、ブロックデバイス複製、その他、さまざまなアプリケーションに対応するための数多くのリソースエージェント等を、日本国内向けに維持管理、支援等を行っているプロジェクトです。

今は主に Pacemaker , Heartbeat , Corosync , DRBD等を扱っています。

Linux-HA Japan 成果物ダウンロード
RHEL/CentOS向けPacemaker RPMパッケージ(yumのリポジトリ形式)や設定ファイル(crm)作成支援ツール、ディスク監視機能などをダウンロードできます。とりあえずRHELもしくはCentOS等のRHEL互換OSにインストールしてみたい場合はこちら。インストール後にとりあえず何か動かしてみたい場合はこちらを参考にしてみてください。

マニュアル
本家コミュニティ提供の公式マニュアルやLinux-HA Japan提供の翻訳マニュアル。
マニュアル読んでもよくわからない場合は、過去のカンファレンスや勉強会等の発表資料も参考に。

メーリングリスト
インストール方法や設定方法等の質問はMLまで。
※投稿するにはメールアドレスの登録が必要です。

イベント情報
カンファレンスへの出展や講演、勉強会開催情報、講演時のスライド公開など。

開発者向けサイト
Linux-HA Japan開発者向けサイトです。Linux-HA Japan独自開発機能のソースコードやバイナリやbugレポート等

最近の投稿

- OSC2014 Hokkaido セミナー資料公開 2014/06/18
- 製品呼称に関する注意喚起 2014/04/01
- OSC2014 Tokyo/Spring セミナー資料・アンケート結果公開 2014/03/05
- 動かして理解する Pacemaker ~CRM設定編~ その3 2014/01/14
- Pacemaker-1.0.13-1.1 + RHEL5におけるIPaddr2の不具合について 2013/12/27
- Pacemakerリポジトリパッケージ1.0.13-1.2リリース 2013/12/26
- OSC2013 Tokyo/Fall セミナー資料・アンケート結果公開 2013/10/22
- 動かして理解する Pacemaker ~CRM設定



④お知らせ



LPIC 304 技術解説セミナー



LPIC304とLPICレベル1が改定予定。
そして、OpenStack認定試験登場の予感。



お知らせ

[2015年](#) [2014年](#) [2013年](#) [2012年](#) [2011年](#) [2010年](#) [2009年](#) [2008年](#) [2006年](#) [2005年](#)

◀ 3 4 5 6 7 8 9 10 11 12 ▶

お知らせ

2014.06.05

[Linux道場Linux学習環境構築編「【第14回】ハードディスクについて」をアップしました。](#)

イベント・セミナー

2014.06.03

[ビジネスパートナー「テクノバン株式会社」様にてLPI-Japan理事長 成井が講演を行いました。\(5月16日\)](#)

イベント開催日:2014年5月16日(金)

プレスリリース

2014.06.03

[LPI-Japanが、『日本電子専門学校』を高等教育機関第一号のLPI-Japan HTML5アカデミック認定校に認定](#)

マスコミ掲載

2014.06.02

[ITpro記事のご紹介\(2014/06/02\)『OpenStack認定試験を開始へ、LPIC 1とLPIC 304の内容も更新』](#)



LPIC 304 技術解説セミナー



CloudStack認定試験、配信間近！

LPI-Japan、『CloudStack技術者認定試験』を2015年2月初旬に配信開始

報道関係各位



LPI-Japan、『CloudStack技術者認定試験』を2015年2月初旬に配信開始

ITプロフェッショナル認定機関として「LPIC」、「OSS-DB技術者認定制度」及び「HTML5プロフェッショナル技術者認定制度」を実施する特定非営利活動法人エルピーアイジャパン（以下：LPI-Japan、東京都港区、理事長 成井 弦、<http://lpi.or.jp>）は、2015年2月初旬に『CloudStack技術者認定試験（CloudStack Professional Certification Exam）』（以下、本試験）の配信開始を発表しました。

近年、急速にクラウド化が進展していく中、プライベート・クラウドとしてOSSのIaaS基盤を活用する事例が増えており、これに対応した技術者が求められてきています。日本市場においては、早くからCloudStackを用いてプライベート・クラウドが構築されており、そのサービスを提供する企業も多く、それに合わせた技術者の育成が求められていることからLPI-Japanでは本試験を開発し、提供することとしました。

本試験の開発にあたりましては、CloudStackユーザー会の方々をはじめとする多くの有識者のサポートにより試験開発を行っています。

LPI-Japanは本試験の実施を通じて、OSSクラウドの構築・運用管理を行う技術者の技術力の向上と人材育成に貢献する所存でございます。また、日本のITビジネス推進と『クラウドファースト（※1）』時代に必要とされるITプロフェッショナルの育成にも貢献いたします。

なお、9月30日、10月15日、17日に本試験のベータ試験を実施致します。

（ベータ試験詳細 <http://www.lpi.or.jp/news/event/page/cloudstack201409/>）

■CloudStackとは

CloudStackはサーバ仮想化環境を基盤にして、IaaS（Infrastructure as a Service）型のクラウドを提供するオープンソースソフトウェアです。「XenServer」「KVM」「商用仮想化ソフトウェア」など、各種のサーバ仮想化ソフトウェアに対応します。WebベースのGUIにより、エンドユーザーがセルフサービスで仮想マシンを展開できる環境を構築できるほか、複数ユーザー間でネットワークを隔離してセキュリティを確保する仕組みや、ファイアウォール、負荷分散、VPNなどの機能も提供します。

■CloudStackの歴史背景

元Cloud.com社により開発されたソフトウェアです。2011年7月、Citrix社がCloud.com社を買収し、その後2012年4月にオープンソースソフトウェアの開発や普及推進を図る非営利団体Apache Software Foundationに寄贈し、現在は完全なオープンソースソフトウェアとして提供されています。



ゼウス・ラーニングパワーからのお知らせ

新コース「LPICレベル1 試験直前対策1日コース」

「試験本番を前に最終確認をして、
確実に合格しませんか？」

最終確認後は、

スクール内併設のテストセンターで受験も出来ます！

LPIC101試験対策(毎月第1・第3・第5土曜開催)

LPIC102試験対策(毎月第2・第4土曜開催)

10:00～17:00 30,000円(受験料込)

今ならなんと新コース設立キャンペーンとして、
入学金10,000円(税抜)が無料！





LPIC 304 技術解説セミナー



ゼウス・ラーニングパワーからのお知らせ



土日開講 無料体験 カードOK フリー教科書 カウンセリング相談無料 増税以降も受講料据え置き 対象講座あり 給付金

Linuxに強い！そのワケとは…？

Point 1
LPi-Japan 公認パートナーアイストラクターが講師を務めています。

当スクールは、Linux/OSS技術者の技術力の認定制度「LPIC」を運営する【LPi-Japan】のアカデミック認定校としてお墨付き。そして LPi-Japan 公認パートナーアイストラクターとして厳選された講師が授業を行っています。

だから、Linuxに強いんです。

...

**教育訓練給付金
指定講座認定記念特別キャンペーン!!**

雇用保険に加入し、勤務していた期間の合計が1年以上ある。

▶ **YES** 当てはまる！
…という方は **CLICK!**

LPi-Japan成井理事長との対談も掲載！

資料請求
WEB資料・紙資料ご用意しています

Linux in English コース



Carl Stevens
LPi-Japanパートナーアイストラクター

This is your chance to study Linux with a native speaker of English while in Japan. I have helped numerous students to pass the LPIC exams. Study with me and gain the English and IT skills you need to succeed!

プロフィール

ニュージーランドより来日し講師として16年のキャリアを持つベテラン講師。Linux in English のカリキュラムは他のスクールにはない新しいスタイルのコースです。

保有資格

LPIC 302 (Mixed Environment)
LPIC 301 (Core)
CCNA
Japanese Proficiency Test, Level 1 (日本語能力試験1級)



企業の貴重な**人財**育成をお手伝いします！

IT研修
お任せください

助成金のご相談承ります。

女性に人気の講座です

MS Office 基礎実践コース
パソコンを使ったお仕事を目指そう！

**日本で唯一、
ネイティブスピーカーが教えるLinux！！
(日本語もしゃべるので安心)**



⑤質疑応答

