

Linux KVM による仮想化環境の構築

○早川正人^{A)}、若松 進^{A)}、大下 弘^{A)}、野崎公隆^{A)}、

千代谷一幸^{A)}、雨宮尚範^{A)}、松岡 孝^{B)}

^{A)} 工学系技術支援室 情報通信技術系

^{B)} 共通基盤技術支援室 情報通信技術系

概要

仮想化技術の1つである KVM は「Kernel-based Virtual Machine」の略であり、Linux カーネルに仮想マシンの管理機能を統合したものである。完全仮想化（フル・バーチャライゼーション）による仮想マシン環境を提供し、動作中の仮想マシンを別のホストに無停止で移動させる「ライブマイグレーション」もサポートされていて、仮想マシンが提供中のサービスを阻害することなく、物理マシンの保守作業等が可能になる。

今回は工学研究科技術部研修において、KVM ベースの仮想化サーバの構築を行い、ライブマイグレーションの動作確認を実行したので報告する。

1 KVM について

一般的に OS の仮想化は、ほかの OS の上で仮想マシンを動かすもの（ホスト OS 型）と、ハードウェアの上で直接仮想マシンを動かすもの（ハイパーバイザ型）の大きく2種類に分かれる。KVM は、ハードウェアのエミュレーションやゲスト OS の管理用のフロントエンドとして「QEMU」を使い、Linux の上でゲスト OS を動かすので、一見前者のホスト OS 型に見えるが、Linux 自体をハイパーバイザにしてしまうことを考慮すると、後者のハイパーバイザ型といえる。

KVM は Linux Kernel にマージされ、kernel 2.6.20 から標準機能として利用でき、Red Hat Enterprise Linux 6 では仮想化技術の中心に据えている。なお、CPU の仮想化支援技術を利用しているので、Inte VT や AMD-V といった仮想化支援機能に対応した CPU と、それらの機能を有効にできる BIOS が必要となる。



図 1. KVM のアーキテクチャ

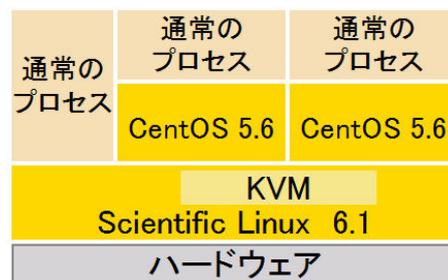


図 2. 構築した仮想化サーバの仕様

2 実験環境の構築

仮想化サーバ構築用には、表 1 に示すスペックの PC を 2 台用意し、NAS には Thecus N7700 Pro を使用した。また、サーバ用の PC には、オンボード NIC (eth1) の他に Intel NIC (eth0) を追加、ネットワークアドレスとして 10.10/16 (NET2)、192.168.1/24 (NET1) を想定し、ネットワークデバイスの設定を行った。

表 1. 仮想化サーバ用 PC の部品構成

PC パーツ	詳細
CPU	AMD PhenomII (4Core, 2.5GHz)
Memory	Transcend (PC3-10600, 8GB) ×2
HDD	Seagate (7200rpm, 1TB)
Motherboard	Biostar (Graphics, GbE, PCI-E)

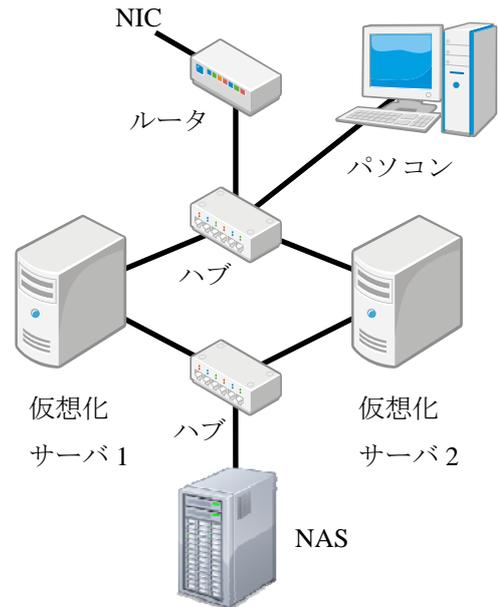


図 3. 実験環境の構成

2.1 ホスト OS のインストール

ホスト OS となる Linux ディストリビューションは、Red Hat Enterprise Linux をベースとして高い互換性を持ち、無償で配布されている Scientific Linux (サイエンティフィック・リナックス) を選択し、Scientific Linux の公式ページ (<http://www.scientificlinux.org/>) 等から Scientific Linux 6.1 の 64bit 版の ISO イメージをダウンロードした。ISO イメージからインストール用ディスク (DVD2 枚) を作成した後、DVD から起動して OS のインストール作業を行った。なお、インストール時の各サーバの設定内容を表 2 に示す。

表 2. 仮想化サーバの設定内容

ホスト名	SV1	SV2	
IP アドレス	NET1(eth0)	192.168.1.11	192.168.1.12
	NET2(eth1)	10.10.1.11	10.10.1.12
パーティション構成	/ (ルート) : 100GB swap : 16GB /guest : 残り全部 (最大容量まで使用)		
インストール パッケージの選択	Desktop を選択し、ソフトウェアのカスタマイズ で仮想化関連のパッケージを選択する		

2.2 ネットワークデバイスの設定

仮想化サーバのネットワークデバイスとして Intel NIC を eth0、オンボード NIC を eth1 に割り当て、IP アドレスを設定し、ホスト名でコンピューターを同定できるように「/etc/hosts」ファイルに記載した。また、ゲストマシンをサーバとして利用するためには、ネットワークブリッジでゲストマシンと外部コンピューターの相互アクセスを可能にする必要があるため、ブリッジインターフェース br0 を作成し、eth0 をブリッジに参加させた。設定はネットワークスクリプト ifcfg-br0、ifcfg-eth0、ifcfg-eth1 を編集することで行った。

- /etc/hosts への追記
iscsi1 10.10.1.10

sv1 192.168.1.11

sv2 192.168.1.12

sv10 10.10.1.11

sv20 10.10.1.12

- /etc/sysconfig/network-scripts/ifcfg-br0 の作成

DEVICE=br0

ONBOOT=yes

BOOTPROTO=none

IPADDR=192.168.1.11 (仮想化サーバ 2 は 192.168.1.12)

NETMASK=255.255.255.0

GATEWAY=192.168.1.1

TYPE=Bridge

- /etc/sysconfig/network-scripts/ifcfg-eth0 への追記

BRIDGE=br0

- /etc/sysconfig/network-scripts/ifcfg-eth1 への追記 (同名の設定項目は置き換えた)

BOOTPROTO=none

IPADDR=10.10.1.11 (仮想化サーバ 2 は 10.10.1.12)

NETMASK=255.255.255.0

GATEWAY=192.168.1.1

2.3 NAS の設定

設定は NAS をネットワークに接続した後、NAS の機能である Web 管理インターフェース (図 4) を用いて行い、iSCSI ターゲットとして 5TB の領域を作成した。なお、設定項目の詳細を表 3 に示す。



図 4. Web 管理インターフェース

表 3. NAS の設定詳細

	設定項目	設定値
Network 1	IP	10.10.1.10
	Netmask	255.255.0.0
RAID	RAID Level	RAID 6
	RAID ID	RAID
	Data Percentage	50%
iSCSI Volume	Allocation	50%
	iSCSI Target Volume	Enable
	Target Name	iscsi1

2.4 iSCSI デバイスのマウント

iSCSI デバイスを利用できる様にするために下記の設定を行った。

- 1) iSCSI イニシエータ パッケージをインストールする

```
# yum install iscsi-initiator-utils
```

- 2) サービスを起動する

```
# service iscsi start
```

- 3) 自動起動の設定


```
# chkconfig iscsid on
# chkconfig iscsi on
```
- 4) iSCSI ターゲットに接続する (iSCSI ターゲットの IP は 10.10.1.10)


```
# iscsiadm --mode discovery --type sendtargets --portal 10.10.1.10
10.10.1.10:3260,1 iqn.2006-01.com.openfiler:tsn.b5ffc134d692
```
- 5) iSCSI ターゲットに接続できているかを確認する


```
# iscsiadm -m node
10.10.1.10:3260,1 iqn.2006-01.com.openfiler:tsn.b5ffc134d692
```
- 6) ノードに接続する


```
# iscsiadm --mode node --targetname iqn.2006-01.com.openfiler:tsn.b5ffc134d692 --login
```
- 7) fdisk でデバイスとしての認識を確認


```
# fdisk -l
```
- 8) パーティションの作成 (/dev/sda を使用)


```
# parted /dev/sda
(parted) mklabel gpt
(parted) mkpart
パーティションの名前? [ ]?
ファイルシステムの種類? [ext2]? ext3
(parted) mkpartfs
パーティションの名前? [ ]?
ファイルシステムの種類? [ext2]? ext3
(parted) quit
```
- 9) ファイルシステムの作成 (ファイルシステムの容量により相当な時間を要する場合があります)


```
# mkfs.ext3 /dev/sda1
```
- 10) マウントと確認


```
# mkdir /iscsi1
# mount -t ext3 /dev/sda1 /iscsi1
# df -h /dev/sda1
```

Filesystem	サイズ	使用	残り	使用%	マウント位置
/dev/sda1	88G	184M	83G	1%	/iscsi1
- 11) 再起動後に自動的にマウントされるよう fstab を編集する(末尾に一行追加)


```
# tail -1 /etc/fstab
/dev/sda1          /iscsi1          ext3          _netdev          0 0
```

2.5 仮想化サーバの設定

仮想化サーバ用の OS (ゲスト OS) は CentOS5.6 の 32bit 版を使用し、ホストとなるマシンのローカルディスク上にインストールし動作確認を実行した後、ライブマイグレーションの動作テストのために iSCSI 機能を有する NAS のディスク領域にインストールした。

まず、仮想マシンの管理ツールである仮想マシンマネージャーを起動する (図 5)。この仮想マシンマネージャーには現在インストールされている仮想マシンが表示されており、稼働状況を確認することができる。

また、仮想マシンの起動、停止もここで行うことができる。新たに仮想マシンを追加する場合は、“新しい仮想マシンの作成” ボタンを押して「新しい仮想マシン」ウインドウを表示し、必要事項を入力していく。



図 5. 仮想マシンマネージャー



図 6. 新しい仮想マシンの作成



図 7. ローカルディスクへ作成

ステップ 1 で新規に作成する仮想マシンの名前を入力したあとインストール方法を選択する。今回は予めダウンロードしておいた ISO イメージを使用するため、ローカルのインストールメディアを選択した。

ステップ 2 (図 6) では、インストールメディアの場所として“ISO イメージを使用”を選び、参照ボタンを押して ISO メディアの検索画面からイメージファイルを指定する。次に OS の種類を“Linux”、バージョンを“Red Hat Enterprise Linux 5.4 or later”と選択した。

ステップ 3 で仮想マシンに割り当てるメモリ容量 (512MB) と CPU の数(1)を指定する。

ステップ 4 (図 7) ではインストールするストレージを指定するが、ホストマシンのディスク上にインストールするには“コンピューターのハードディスク上にディスクイメージを作成”を選択して、必要なディスク容量を指定する。また、NAS へインストールするには“管理しているストレージか、他の既存のストレージを選択する”を選択し、参照ボタンを押してストレージボリュームを指定する。「ストレージボリュームの検索または作成」画面 (図 8) で iSCSI に対して新規ボリュームボタンを押して図 9 の作成画面に移り、ボリュームの名前と使用容量(20GB)を指定した。新しく作成したボリュームを選択して元の画面に戻る (図 10)。



図 8. ストレージボリューム選択

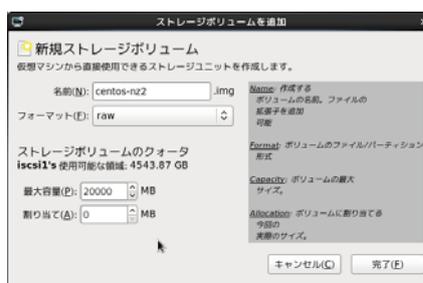


図 9. ストレージボリューム作成



図 10. NAS へインストール



図 11. ネットワーク設定



図 12. 仮想マシンマネージャー



図 13. ホスト OS のデスクトップ

ステップ 5 (図 11) では接続するネットワークなどを指定する。ネットワークにはブリッジ接続を利用するため“ホストデバイス eth0 (Bridge 'br0')”を選択した。そして仮想化の種類として“kvm”、アーキテクチャとして“i686”を選択した。ここで“完了”ボタンを押すとゲスト OS のインストールが始まる。

インストールが終わると、仮想マシンマネージャー (図 12) に新たに仮想マシンが追加され、ホストマシン上で実行中であることが確認できる。(図 13)

3 ライブマイグレーション

ライブマイグレーションとは稼働中の仮想マシンを停止させずに別のホストマシンに移動する技術のことであり、ホストマシンを停止させる計画メンテナンス時に、あらかじめ仮想マシンを別のホストマシンに移行させておけば、仮想マシンで提供するサービスの無停止でのメンテナンス実施が可能になる。

ライブマイグレーションを行なうと、あるホスト上で動作している仮想マシン上のメモリーイメージが丸ごと別のホスト上の仮想マシンに移し替えられ、稼働中の OS やアプリケーションソフト、ネットワーク接続などを一切停止・切断させることなく新しいホスト上で動作を継続することができる。厳密には切り替えの瞬間にミリ秒単位の瞬断が生じるが、ネットワークのセッションなどが切断されることは無く、ユーザからは移動が行われたことは分からない。

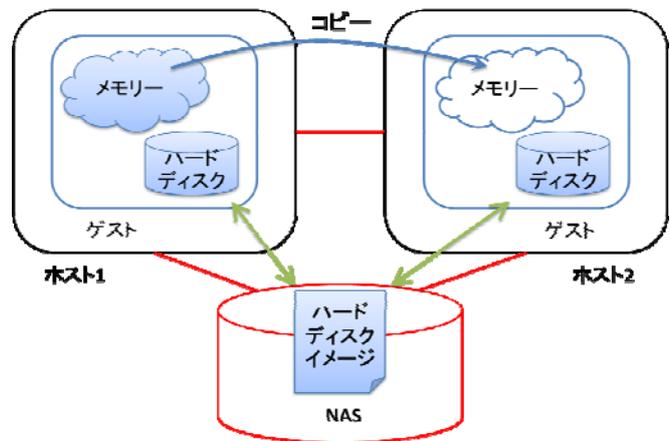


図 14. ライブマイグレーションとシステム構成

3.1 ライブマイグレーションの実行方法

ライブマイグレーションを実行するためには、ホスト 1 (sv1) でゲスト OS を動作させ。仮想マシンマネージャーでゲストにカーソルを合わせ、右クリックするとマイグレーションを選択 (図 15) することができる。その後でマイグレーション先のホスト (SV2) を選択 (図 16) して実行する。マイグレーションに要する時間は、アプリケーションが動作している状態で約 50 秒程度、誰もログインしていない場合は 36 秒程度であった。

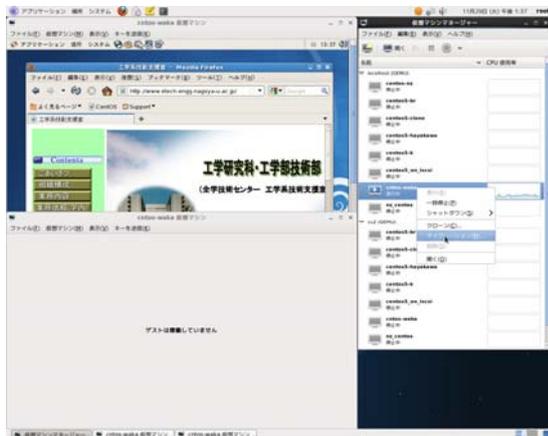


図 15. ライブマイグレーションの実行 1

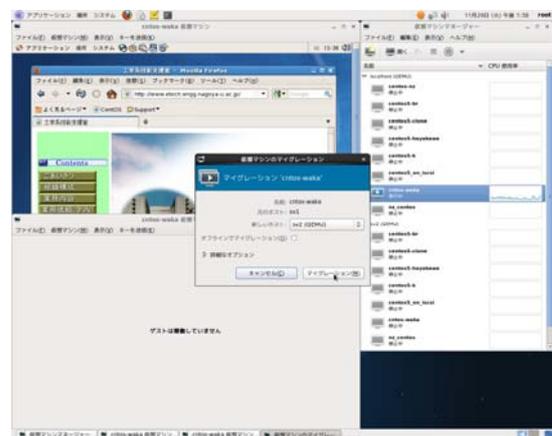


図 16. ライブマイグレーションの実行 2

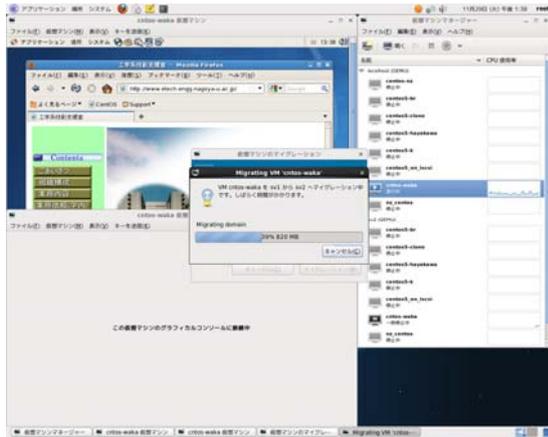


図 17. ライブマイグレーションの実行 3

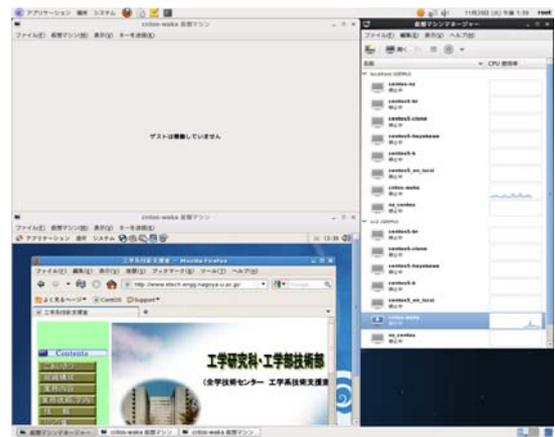


図 18. ライブマイグレーションの実行 4

3.2 ライブマイグレーションの動作確認

ライブマイグレーションの動作確認を次の方法で行った。ゲスト OS 上に DVDISO ファイル (約 4GB) を置き、それを Windows 端末上の ftp クライアントで取得するのに 6 分程度を要する。このデータ転送中にライブマイグレーションを実行して、正常にファイルが受け取れるかどうかを確認した。その結果、受信ファイルは元のものと同じだった。したがって、ライブマイグレーション中もとぎれることなくサービスが行えることが証明された。ただ、ライブマイグレーション中の転送速度は、数%程度に減少して (図 20) 応答が極端に悪くなった。次に、ゲスト OS のメモリ使用量を変更した場合とメモリ割り当て量を変更した場合で測定を実行した。

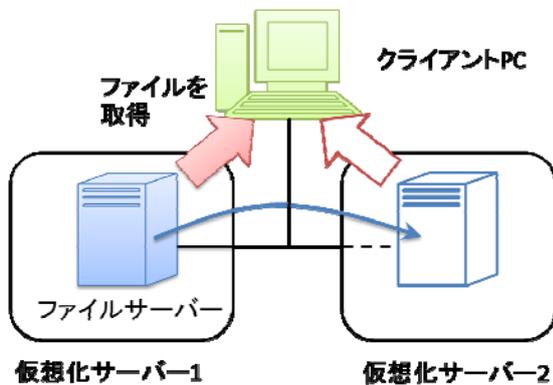


図 19. ファイルサーバの動作確認

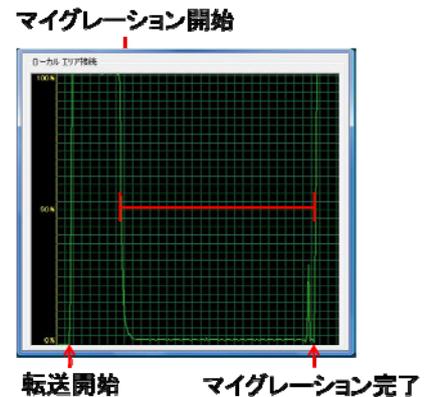


図 20. クライアント PC のネットワーク使用率

4 まとめ

今回の実験では、無償版のソフトを利用して仮想化サーバを構築し、実際に運用して動作確認できた。また、ライブマイグレーションを実行して、その動作を確認した。その結果、ライブマイグレーションの実行時間はメモリの最大使用量に依存する事がわかり、なるべくサーバ負荷が少ない時に実行するのが最善であるとわかった。また、ゲスト OS に割り当てるメモリ量を調節する事で、ライブマイグレーションの実行時間を最適化できる事がわかった。

参考文献

- [1] 野崎ほか: “プライベートクラウド用サーバの構築”, 名古屋大学工学研究科・工学部「技報」Vol.14, 2012年3月, PP. (2012年2月 現時点で不明)
- [2] クラウド Watch Linux のカーネルに入った仮想化技術「KVM」,
http://cloud.watch.impress.co.jp/docs/column/virtual/20110523_447267.html, Impress Group, 2012年2月
- [3] クラウド Watch Linux 標準のサーバー仮想化機能「KVM」を試す,
http://cloud.watch.impress.co.jp/docs/column/virtual/20110606_450502.html, Impress Group, 2012年2月
- [4] COMPUTERWORLD 「KVM」—Linux 標準の仮想化機能の得意領域を知る,
<http://www.computerworld.jp/contents/111429/>, IDG Interactive, 2012年2月
- [5] @IT Linux 標準の仮想化技術「KVM」の仕組み, <http://www.atmarkit.co.jp/flinux/rensai/kvm01/kvm01a.html>,
アイティメディア株式会社, 2012年2月
- [6] IT用語辞典, <http://e-words.jp/>, 株式会社インセプト, 2012年2月