

クラウド基盤構築演習

第二部: Eucalyptusによるクラウド基盤構築

第10回: Eucalyptus環境の構築

ver1.3 2012/07/07



目次

- Eucalyptusのインストール
- Eucalyptusの設定
- Eucalyptusの動作確認
- ツールの準備
- マシンイメージの登録
- インスタンスの起動と接続

本講で学ぶこと、実施すること -1-

- CentOS環境にEucalyptusをインストールしましょう
 - 実際にインストールすることによって、どんなソフトウェアが必要かを理解します
 - もしインストールでつまづいてしまったら、他の人も同じところでつまづく可能性を知ることができます
- インストールしたEucalyptusを設定しましょう
 - どこに何を設定すべきかを理解します
 - 設定を通してEucalyptusの各コンポーネントの仕組みを理解する前段階に知識をステップアップさせます

本講で学ぶこと、実施すること -2-

- 起動したEucalyptusの動作確認をしましょう
 - 各コンポーネントがどのような状態であることが正常なのかを理解できます
 - 正常な状態を知っておくことで、それ以外の状態は何かが起きていることを認識できることになります
- Eucalyptusの利用環境を用意しましょう
 - インストールして設定して起動して、ではIaaSを使えるようにするためには他に何が必要かを把握できます
 - インストールや設定よりも、実際に使うための環境を準備するほうが大変であることが認識できます
 - Eucalyptusの仕組みの概要が理解できます



EUCALYPTUSのインストール

インストール前に準備するもの -1-

- CentOSのインストールメディア
 - もし、インターネットに接続できない環境や回線が細い環境でEucalyptusをインストールする場合には必要です
 - CentOSのリポジトリサーバに接続できる場合や、回線が細くない環境では必要ありません
 - 媒体としてではなくISOファイルとして用意してもOKです
 - CentOS 6.2のインストールDVDがあれば十分
- CentOSのリポジトリサーバのURL
 - CentOSのインストールメディアを利用しない場合に必要です
 - 日本国内の主なミラーサーバからローカルリポジトリを作成しておくが良い
- 今回はローカルにレポジトリサーバを用意したので、それを利用する。

インストール前に準備するもの -2-

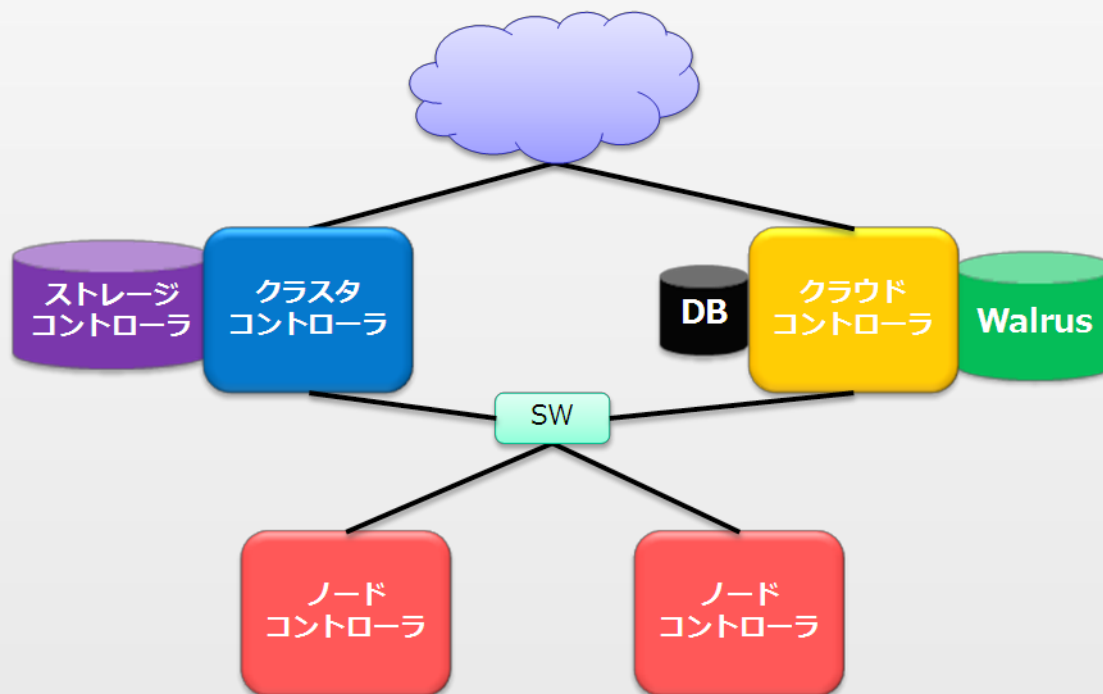
- Eucalyptus社が配布するバイナリ
 - Eucalyptusのパッケージ
 - euca2oolsのパッケージ
 - 以前はRPMをダウンロードしてインストールする形式だったが、最近ではyumやapt-getなどで簡単にインストールする方法が主流
- マシンイメージ
 - 例えば以下から入手
 - Eucalyptus社の配布しているマシンイメージ
 - マシンイメージ工房のマシンイメージ
 - 自分で作成
 - 作成方法は別途資料を参照

前提条件 -1-

- CentOS 6.2 x86_64がインストールされているマシン
 - 物理マシンのOSインストール時にはシンプルにBasic Serverのみ
 - 今回は時間の節約のため、予めインストールされた環境を利用する
- ネットワークはサービスセグメント用とバックエンド用の2回線
 - CLC/Walrus, CC/SCは利用者と接続するためのサービスセグメントとバックエンド用の2回線を設定
 - NCはCLC/Walrus, CC/SCと接続するバックエンド用の1回線のみあれば十分だが、セットアップ作業をやり易くするために2回線用意

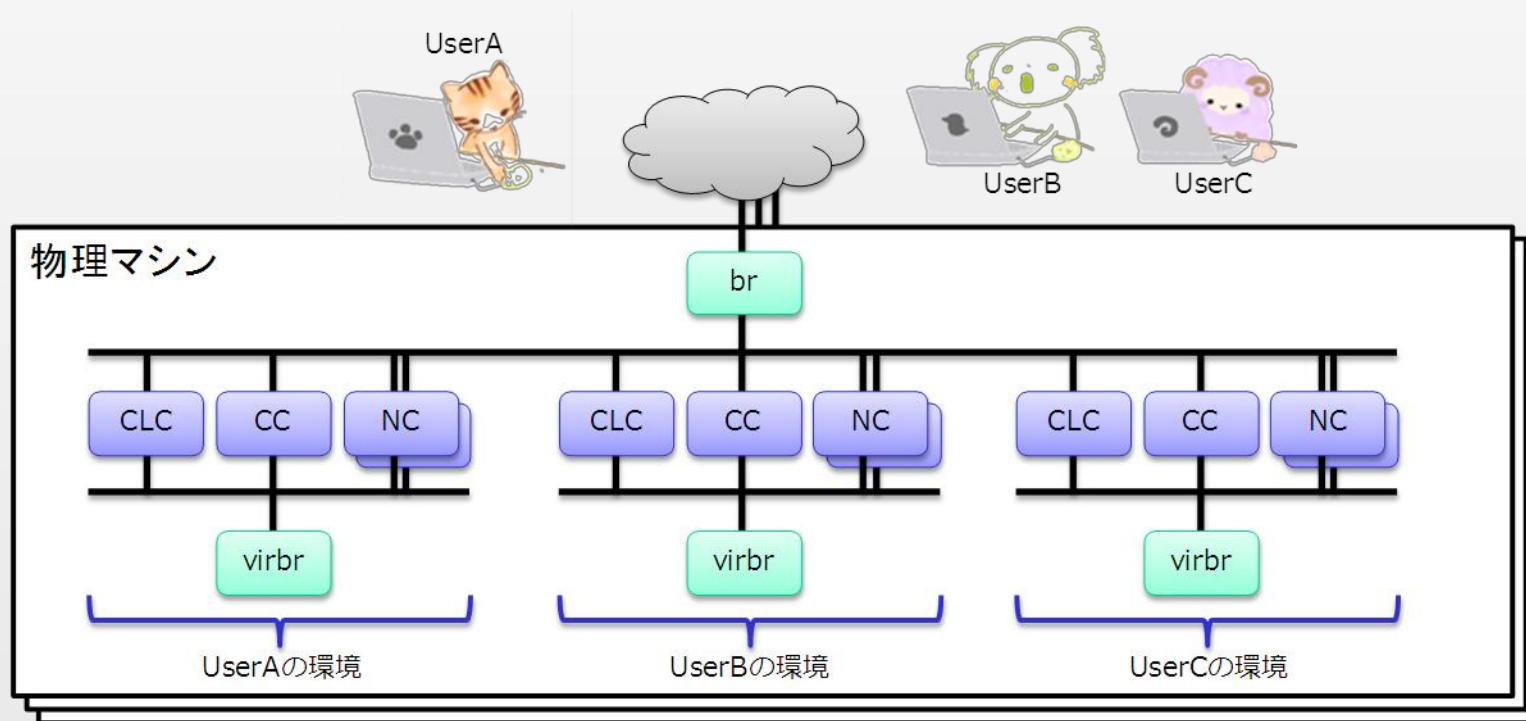
前提条件 -2-

- 今回は以下の構成を想定して構築します。



補足情報

- 実際には以下のような構成で演習環境を構築しています。





ではインストールをはじめましょう

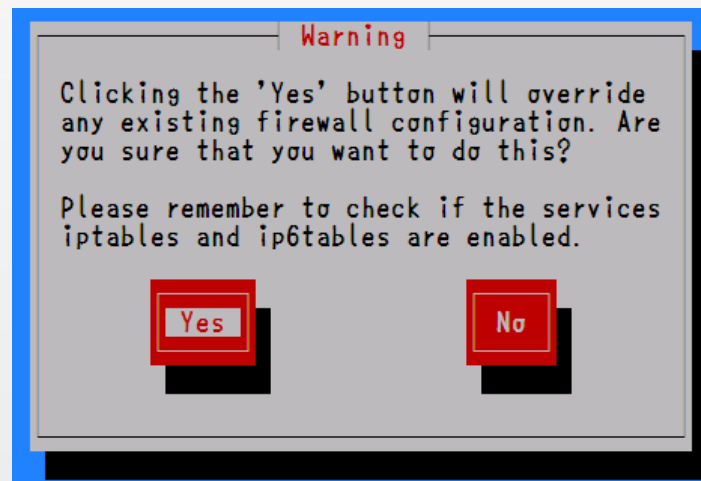
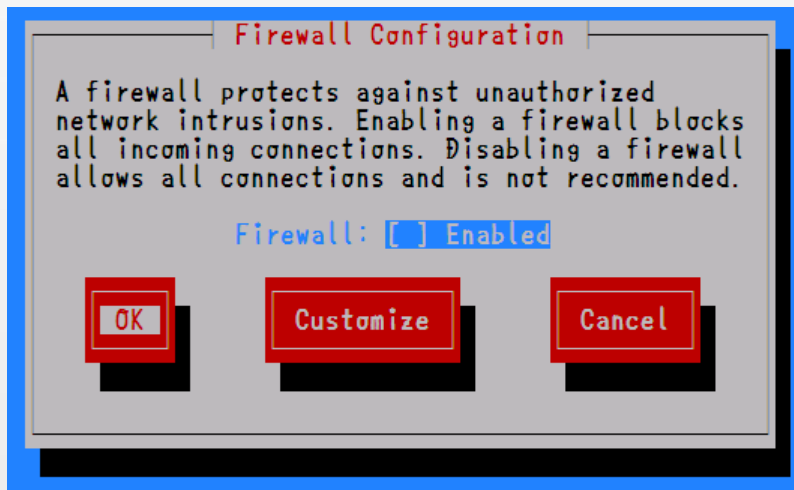
- 別紙「ユーザ情報」を参照し、自分の接続すべき各マシンを確認しておきます
- 各ホストにはsshで接続します。
 - 接続する際にはeth0のIPアドレスにアクセスしてください。
- インストール作業は大きくわけて以下の作業になります
 - インストール前の環境設定
 - インストール作業
 - インストール後の設定

インストール前のOS設定

- セキュリティ設定 -

■ ファイアウォールの設定

- 各マシンにログインし、「LANG=C setup」コマンドを実行し、[Firewall configuration]を選択し以下の図のように[Firewall:]をDisabledに変更して、「OK」を選択
- 既存の設定を上書きしてもよいか確認してくるので「Yes」を選択



■ SELinuxの設定

- 各マシンの/etc/selinux/configを編集し、以下のパラメータを変更します。

SELINUX=enforcing



SELINUX=disabled

インストール前のOS設定

- 不要なサービスの停止 -

- Eucalyptusを動かすために不要となるサービスを停止するように設定します。

- 各物理マシンにて実行

```
# chkconfig iptables off  
# chkconfig ip6tables off  
# /etc/init.d/iptables stop
```

- 上記以外にも特にEucalyptusでは使用しないサービスは停止しても構いません

- 例えばcupsとか

インストール前のOS設定

-ループバックデバイスの設定-

- 何も設定しない場合、利用可能なループバックデバイスの最大数は8個までとなっているため、利用可能な最大数を増やす必要があります。
- ループバックデバイスを増やす必要があるのは、SCとNCです。
 - SCのストレージに外部ストレージを利用する場合はSCのループバックデバイスは増やさなくても大丈夫です
- CentOS 6では以下のように実行することで、256個まで利用可能になります。

```
# for i in `seq 8 255`; do  
> echo loop${i} >> /etc/udev/makedev.d/50-udev.nodes;  
> done
```

- 256個以上を利用したい場合は中井さんのブログを参照してください。
<http://d.hatena.ne.jp/enakai00/20110307/1299464439>

インストール前のOS設定 -ブリッジの設定-

- NCにブリッジ「br0」を設定し、NIC「eth1」を接続します。
 - 以下を実行し、ブリッジのためのソフトウェアがNCにインストールされていることを確認します。

```
# rpm -qa bridge-utils  
bridge-utils-1.2-9.el6.x86_64
```

- eth1の設定ファイル「/etc/sysconfig/network-scripts/ifcfg-eth1」を以下のように変更します。

```
IPADDR="192.168.120.222"  
NETMASK="255.255.255.0"  
↓  
#IPADDR="192.168.120.222"  
#NETMASK="255.255.255.0"  
BRIDGE="br0"
```

インストール前のOS設定 -ブリッジの設定-

- br0の設定ファイル「/etc/sysconfig/network-scripts/ifcfg-br0」を以下の内容で作成します

```
DEVICE="br0"  
TYPE="Bridge"  
BOOTPROTO="static"  
IPADDR="別紙のeth1のIPアドレス"  
NETMASK="255.255.255.0"  
NM_CONTROLLED="no"  
ONBOOT="yes"
```

- ネットワークのサービスを再起動します

```
# /etc/init.d/network restart      # 以下のように出力されればOK  
Shutting down interface eth0:      [ OK ]  
Shutting down interface eth1:  bridge br0 does not exist!  
  
[ OK ]  
Shutting down loopback interface: [ OK ]  
Bringing up loopback interface:   [ OK ]  
Bringing up interface eth0:       [ OK ]  
Bringing up interface eth1:       [ OK ]  
Bringing up interface br0:        [ OK ]
```


インストール前のOS設定 -ブリッジの設定-

- ネットワークを再起動するとifconfigのeth1の結果が以下のように変化し、新しいブリッジデバイス「br0」が作成されたことが確認できます

```
eth1    Link encap:Ethernet  HWaddr 52:54:00:02:00:32
        inet addr:192.168.120.223  Bcast:192.168.120.255  Mask:255.255.255.0
        inet6 addr: fe80::5054:ff:fe02:32/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:21079 errors:0 dropped:0 overruns:0 frame:0
        TX packets:1761 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:31635220 (30.1 MiB)  TX bytes:124225 (121.3 KiB)

        ↓

br0      Link encap:Ethernet  HWaddr 52:54:00:02:00:32
        inet addr:192.168.120.223  Bcast:192.168.120.255  Mask:255.255.255.0
        inet6 addr: fe80::5054:ff:fe02:32/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:11 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
        RX bytes:0 (0.0 b)  TX bytes:650 (650.0 b)

eth1     Link encap:Ethernet  HWaddr 52:54:00:02:00:32
        inet6 addr: fe80::5054:ff:fe02:32/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:21082 errors:0 dropped:0 overruns:0 frame:0
        TX packets:1769 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:31635394 (30.1 MiB)  TX bytes:124821 (121.8 KiB)
```

インストール前のOS設定

-カーネルのアップデート-

- Walrusをインストールするホストのカーネルを最新版にアップデートします。
 - CentOS 6.2のリリース版に含まれているカーネルを使用した場合、Walrusは登録されたマシンイメージの復号処理が行えないという問題があります。(Ubuntu 10.04にも同様の問題あり)
 - 以下のコマンドをWalrusのインストール先であるCLCのホストにて実行します

```
# wget http://[VirbrのIPアドレス]/centos/6.2/kernel-2.6.32-220.23.1.el6.x86_64.rpm
# wget http://[VirbrのIPアドレス]/centos/6.2/kernel-firmware-2.6.32-220.23.1.el6.noarch.rpm
# rpm -Uvh kernel-2.6.32-220.23.1.el6.x86_64.rpm ¥
> kernel-firmware-2.6.32-220.23.1.el6.noarch.rpm
```

- 今までの作業を反映させるために、一度CLC,CC,NCの各ホストを再起動します

```
# reboot
```

インストール前のOS設定

- yumレポジトリの設定 -

- /etc/yum.repos.d/CentOS-Base.repo を編集し以下の編集を行なう

- [base]の設定

```
mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=os
#baseurl=http://mirror.centos.org/centos/$releasever/os/$basearch/
↓
#mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=os
baseurl=http://[VirbrのIPアドレス]/centos/$releasever/os/$basearch/
```

- [updates]の設定

```
mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=updates
#baseurl=http://mirror.centos.org/centos/$releasever/updates/$basearch/
↓
#mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=updates
baseurl=http://[VirbrのIPアドレス]/centos/$releasever/updates/$basearch/
```

- [extra]の設定。末尾に以下を追記

```
enabled=0
```

インストール前のOS設定

- yumレポジトリの設定1 -

■ eucalyptusのレポジトリ登録

- 以下を各マシンで実行してレポジトリを登録する

```
# yum install http://[VirbrのIPアドレス]/eucalyptus/3.1/rhel/6/x86_64/eucalyptus-release-3.1.noarch.rpm
```

- /etc/yum.repos.d/eucalyptus.repo を編集し、[eucalyptus] を設定

```
baseurl=http://downloads.eucalyptus.com/software/eucalyptus/3.1/rhel/$releasever/$basearch  
↓  
baseurl=http://[VirbrのIPアドレス]/eucalyptus/3.1/rhel/$releasever/$basearch
```

■ euca2oolsのレポジトリ登録

- 以下をCLCとNCで実行してレポジトリを登録する

```
# yum install http://[VirbrのIPアドレス]/euca2ools/2.1/rhel/6/x86_64/euca2ools-release-2.1.noarch.rpm
```

- /etc/yum.repos.d/euca2ools.repo を編集し、[euca2ools] を設定

```
baseurl=http://downloads.eucalyptus.com/software/euca2ools/2.1/rhel/$releasever/$basearch  
↓  
baseurl=http://[VirbrのIPアドレス]/euca2ools/2.1/rhel/$releasever/$basearch
```

インストール前のOS設定

- yumレポジトリの設定2 -

- epelのレポジトリ登録

- 以下を各マシンで実行してレポジトリを登録する

```
# yum install http://[VirbrのIPアドレス]/eucalyptus/3.1/rhel/6/x86_64/epel-release-6.noarch.rpm
```

- /etc/yum.repos.d/epel.repo を編集し、[epel] を設定

```
#baseurl=http://download.fedoraproject.org/pub/epel/6/$basearch
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-6&arch=$basearch
↓
baseurl=http://[VirbrのIPアドレス]/epel/6/$basearch
#mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-6&arch=$basearch
```

- Walrus を構築するホストでのみ実行

- 以下を実行してレポジトリを登録する

```
# yum install http://[VirbrのIPアドレス]/eucalyptus/3.1/rhel/6/x86_64/elrepo-release-6.noarch.rpm
```

- /etc/yum.repos.d/elrepo.repo を編集し、[elrepo] を設定

```
baseurl=http://elrepo.org/linux/elrepo/el6/$basearch/
mirrorlist=http://elrepo.org/mirrors-elrepo.el6
↓
baseurl=http://[VirbrのIPアドレス]/elrepo/el6/$basearch/
#mirrorlist=http://elrepo.org/mirrors-elrepo.el6
```

インストール前のOS設定

-KVMとlibvirtのアップデート-

- NCをインストールするホストのKVMとlibvirtを最新版にアップデートします。
 - CentOS 6.2のリリース版に含まれているKVMとlibvirtを使用した場合、インスタンスの起動時に/dev/kvmを使用する権限がない旨のエラーを吐いて起動処理が失敗します
 - 以下のコマンドを各NCのホストにて実行します

```
# yum update libvirt libvirt-client qemu-kvm
```

インストール前のOS設定

-ntpの設定 その1-

- Eucalyptusの各コンポーネントは一定以上時刻がズレると正しく通信できないため、各コンポーネントでntpの設定をする必要があります。
- 今回のような構成の場合はCLC/WalrusはCC/SCにサービスを提供し、CC/SCはNCにサービスを提供するような設定にします。
 - 今回はシングルクラスタ構成なため、ntpサーバはCLCのみでも良いと思えるかもしれませんが、マルチクラスタ構成にした場合はNCが所属するCCに問い合わせに行くほうが構造的に綺麗です
- ただし、今回はKVM上のゲストOSを物理マシンとしているため、ntpによる時刻同期はKVMの仕組みとバッティングしてしまうため、**時刻同期の設定はスキップします**。物理マシンで実際にEucalyptus環境を構築する際に時刻同期のことを思い出してください。



インストール前のOS設定

-ntpの設定 その2-

- 各物理マシンの/etc/ntp.confを設定（この演習では設定しません）
 - CLC/Walrusでの設定
 - 上位サーバがある場合は上位サーバに問い合わせにいくよう設定
 - CC/SCに対して問い合わせを許可する
 - 上位サーバがある場合、設定後に「ntpddate -b 上位サーバ」を2～3回実行し、上位サーバと時刻を合わせてから「/etc/init.d/ntpd start」を実行する
 - CC/SCでの設定
 - CLC/Walrusに問い合わせにいくよう設定
 - NCに対して問い合わせを許可する
 - 設定後に「ntpddate -b CLC/Walrus」を2～3回実行し、時刻を合わせてから「/etc/init.d/ntpd start」を実行する
 - NCでの設定
 - CC/SCに問い合わせにいくよう設定
 - 設定後に「ntpddate -b CC/SC」を2～3回実行し、時刻を合わせてから「/etc/init.d/ntpd start」を実行する

Eucalyptusのインストール

- 各コンポーネントのインストール1 -

■ CLCにて以下を実行し、Cloud ControllerとWalrusをインストール

Cloud Controller を構成するパッケージ群のインストール

```
# yum groupinstall eucalyptus-cloud-controller
```

途中で GPG キーのインポート確認が出たら y を押下する

```
warning: rpmts_HdrFromFdno: Header V3 RSA/SHA1 Signature, key ID c105b9de: NOKEY
```

```
Retrieving key from file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6
```

```
Importing GPG key 0xC105B9DE:
```

```
  Userid : CentOS-6 Key (CentOS 6 Official Signing Key) <centos-6-key@centos.org>
```

```
  Package: centos-release-6-2.el6.centos.7.x86_64 (@anaconda-CentOS-201112091719.x86_64/6.2)
```

```
  From    : /etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6
```

```
Is this ok [y/N]: y
```

```
warning: rpmts_HdrFromFdno: Header V3 RSA/SHA1 Signature, key ID c1240596: NOKEY
```

```
Retrieving key from file:///etc/pki/rpm-gpg/RPM-GPG-KEY-eucalyptus-release
```

```
Importing GPG key 0xC1240596:
```

```
  Userid : Eucalyptus Systems, Inc. (release key) <security@eucalyptus.com>
```

```
  Package: eucalyptus-release-3.1-1.el6.noarch (@/eucalyptus-release-3.1.noarch)
```

```
  From    : /etc/pki/rpm-gpg/RPM-GPG-KEY-eucalyptus-release
```

```
Is this ok [y/N]: y
```

Walrus をインストール

```
# yum install eucalyptus-walrus
```

Eucalyptusのインストール

- 各コンポーネントのインストール2 -

■ CCにて以下を実行し、Cluster ControllerとStorage Controllerをインストール

Cluster ControllerとStorage Controllerをインストール

```
# yum install eucalyptus-cc eucalyptus-sc
```

■ 各NCにて以下を実行し、Node Controllerをインストール

まずは以下を実行してインストール対象のホストでKVMが動作可能かを確認する

例) 今回の環境のようなインテルCPUの場合

```
# modprobe kvm_intel
```

```
# modprobe vhost_net
```

例) AMD CPUの場合

```
# modprobe kvm_amd
```

```
# modprobe vhost_net
```

もし対応していない場合には、以下のような出力が返る

```
# modprobe kvm_intel
```

```
FATAL: Error inserting kvm_intel (/lib/modules/2.6.32-220.el6.x86_64/kernel/arch/x86/kvm/kvm-intel.ko): Operation not supported
```

NC のインストール

```
# yum install eucalyptus-nc
```

Eucalyptusの設定

- eucalyptus.confの設定1 -

- CCの/etc/eucalyptus/eucalyptus.confの以下のパラメータを変更します。

行	パラメータ	デフォルト値	設定値
157	VNET_MODE	SYSTEM	MANAGED-NOVLAN
163	VNET_PRIVINTERFACE	eth0	eth1
174	VNET_PUBINTERFACE	eth0	eth0
191	VNET_PUBLICIPS	(コメント) your-free-public-ip-1 your-free-public-ip-2 ...	(コメント外す) 別紙を参照
196	VNET_SUBNET	(コメント) 192.168.0.0	(コメント外す) 172.16.0.0
197	VNET_NETMASK	(コメント) 255.255.0.0	(コメント外す) 255.255.0.0
202	VNET_ADDRSPERNET	(コメント) 32	(コメント外す) 128
206	VNET_DNS	(コメント) your-dns-server-ip	(コメント外す) 157.1.200.66

```

VNET_MODE="SYSTEM"
VNET_PRIVINTERFACE="eth0"
VNET_PUBINTERFACE="eth0"
#VNET_PUBLICIPS="your-free-public-ip-1 your-free-public-ip-2 ..."
#VNET_SUBNET="192.168.0.0"
#VNET_NETMASK="255.255.0.0"
#VNET_ADDRSPERNET="32"
#VNET_DNS="your-dns-server-ip"

```

↓

```

VNET_MODE="MANAGED-NOVLAN"
VNET_PRIVINTERFACE="eth1"
VNET_PUBINTERFACE="eth0"
VNET_PUBLICIPS="157.1.207.14-157.1.207.19"
VNET_SUBNET="172.16.0.0"
VNET_NETMASK="255.255.0.0"
VNET_ADDRSPERNET="128"
VNET_DNS="157.1.200.66"

```

Eucalyptusの設定

- eucalyptus.confの設定2 -

- 各NCの/etc/eucalyptus/eucalyptus.confの以下のパラメータを変更します。

行	パラメータ	デフォルト値	設定値
157	VNET_MODE	SYSTEM	MANAGED-NOVLAN
174	VNET_PUBINTERFACE	eth0	eth1

```
VNET_MODE="SYSTEM"  
VNET_PUBINTERFACE="eth0"
```

↓

```
VNET_MODE="MANAGED-NOVLAN"  
VNET_PUBINTERFACE="eth1"
```

Eucalyptusの設定

- DB初期化とコンポーネントの登録1 -

- 以下をCLCで実行してCloud ControllerのDBを初期化します

```
# euca_conf --initialize
```

以下のように「succeeded」と出力されれば初期化成功です
Initializing Database...
Initialize command succeeded

- 以下をCLCで実行してCloud ControllerとWalrusを起動します

```
# service eucalyptus-cloud start
```

- 以下をCCで実行してCluster ControllerとStorage Controllerを起動します

```
# service eucalyptus-cc start  
# service eucalyptus-cloud start
```

- 以下をNCで実行してNode Controllerを起動します

```
# service eucalyptus-nc start
```

Eucalyptusの設定

- DB初期化とコンポーネントの登録2 -

- 以下をCLCで実行してCloud ControllerとWalrusが起動していることを確認します

以下を実行してWeb管理画面のサービスがポート8443/tcpで起動していることを確認

```
# lsof -i:8443
```

COMMAND	PID	USER	FD	TYPE	DEVICE	SIZE/OFF	NODE	NAME
eucalyptu	2347	eucalyptus	304u	IPv4	72754	0t0	TCP	*:pcsync-https (LISTEN)

以下を実行して、Cloud Controller/Walrusのサービスがポート8773/tcpで起動していることを確認

```
# lsof -i:8773
```

COMMAND	PID	USER	FD	TYPE	DEVICE	SIZE/OFF	NODE	NAME
eucalyptu	2347	eucalyptus	197u	IPv4	63914	0t0	UDP	157.1.207.220:8773
eucalyptu	2347	eucalyptus	198u	IPv4	63915	0t0	UDP	228.7.7.3:8773
eucalyptu	2347	eucalyptus	221u	IPv4	65597	0t0	TCP	*:8773 (LISTEN)

- 以下をCCで実行してCluster Controllerが起動していることを確認します

以下を実行してCCのサービスがポート8774/tcpで起動していることを確認

```
# lsof -i:8774
```

COMMAND	PID	USER	FD	TYPE	DEVICE	SIZE/OFF	NODE	NAME
httpd	1703	root	4u	IPv6	10337	0t0	TCP	*:8774 (LISTEN)
httpd	1705	eucalyptus	4u	IPv6	10337	0t0	TCP	*:8774 (LISTEN)
httpd	1706	eucalyptus	4u	IPv6	10337	0t0	TCP	*:8774 (LISTEN)
httpd	1707	eucalyptus	4u	IPv6	10337	0t0	TCP	*:8774 (LISTEN)
httpd	1708	eucalyptus	4u	IPv6	10337	0t0	TCP	*:8774 (LISTEN)
httpd	1709	eucalyptus	4u	IPv6	10337	0t0	TCP	*:8774 (LISTEN)

Eucalyptusの設定

- DB初期化とコンポーネントの登録3 -

- 以下を各NCで実行してNode Controllerが起動していることを確認します

以下を実行してNode Controllerのサービスがポート8775/tcpで起動していることを確認

```
# lsof -i:8775
```

COMMAND	PID	USER	FD	TYPE	DEVICE	SIZE/OFF	NODE	NAME
httpd	1970	root	4u	IPv6	11958	0t0	TCP	*:8775 (LISTEN)
httpd	2003	eucalyptus	4u	IPv6	11958	0t0	TCP	*:8775 (LISTEN)

- 以下をCLCで実行してCloud ControllerにWalrusのコンポーネントを登録します

```
# euca_conf --register-walrus --partition walrus --host [CLCのeth0のIP] --component walrus0
```

--hostにはCLCのeth0のIPアドレス(別紙参照)を指定する

--componentには任意のコンポーネント名を指定する

実行後、以下のように出力される

```
No credentials found, attempting local authentication
```

```
RESPONSE true
```

```
Trying rsync to sync keys with 157.1.207.10
```

```
The authenticity of host '157.1.207.10 (157.1.207.10)' can't be established.
```

```
RSA key fingerprint is 46:10:b8:72:08:8e:13:80:28:8f:17:2b:74:61:28:3a.
```

```
Are you sure you want to continue connecting (yes/no)? yes
```

```
root@157.1.207.10's password: ← パスワードを入力
```

```
done
```

Eucalyptusの設定

- DB初期化とコンポーネントの登録4 -

- 以下をCLCで実行してCloud ControllerにCluster ControllerとStorage Controllerのコンポーネントを登録します

```
# euca_conf --register-cluster --partition jp-east-1a --host [CCのeth0のIP] --component cluster0
--partitionには任意のゾーン名を指定する。AWS形式を模すならば上記のような文字列を指定する
--hostにはCCのeth0のIPアドレス(別紙参照)を指定する
--componentには任意のコンポーネント名を指定する
```

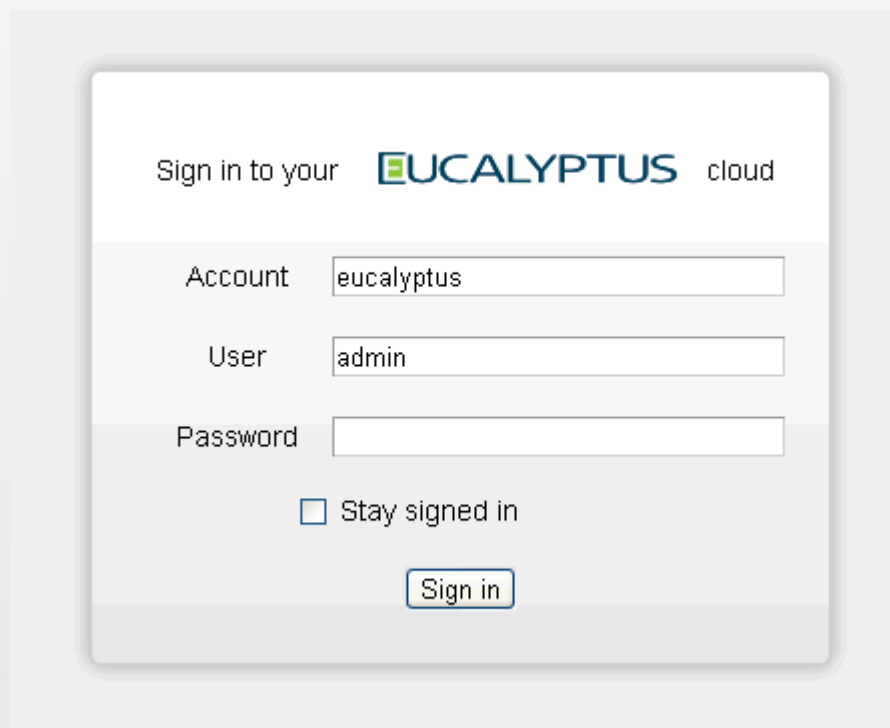
```
# euca_conf --register-sc --partition jp-east-1a --host [CCのeth0のIP] --component storage0
--partitionには--register-cluster時に指定したゾーン名を指定する
--hostにはCCのeth0のIPアドレス(別紙参照)を指定する
--componentには任意のコンポーネント名を指定する
```

- 以下をCCで実行してCluster ControllerにNode Controllerを登録します

```
# euca_conf --register-nodes "[NC1のeth1のIP] [NC2のeth1のIP]"
各NCのIPアドレスをスペース区切りで指定する
```


Web管理画面へのログイン

- [https://\[CLCのeth0のIPアドレス\]:8443/](https://[CLCのeth0のIPアドレス]:8443/)にアクセスすると以下の画面が表示されます
- Passwordには初期状態のパスワード「admin」を入力し、[Sign in]をクリックします



Sign in to your **EUCALYPTUS** cloud

Account

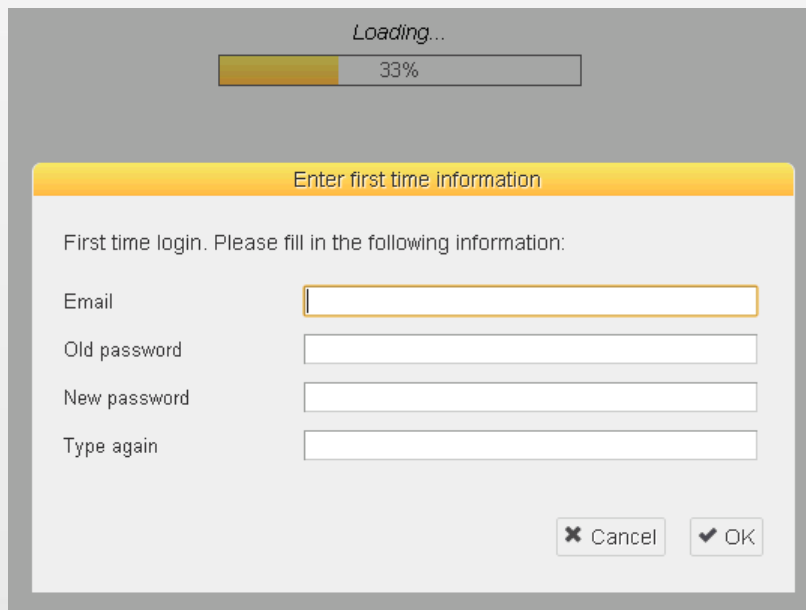
User

Password

☐ Stay signed in

初期パスワードの変更

- 以下の各フィールドを入力し、[OK]をクリックします。
 - Email → 今回はSMTPの設定をしていないので、「root@localhost」でもOKですが、実際の環境では管理者のメールアドレスを入力したほうがよいです
 - Old passwordには前のページで入力した「admin」を入力します。
 - New passwordとType againには新しく設定するパスワードを入力します



Loading... 33%

Enter first time information

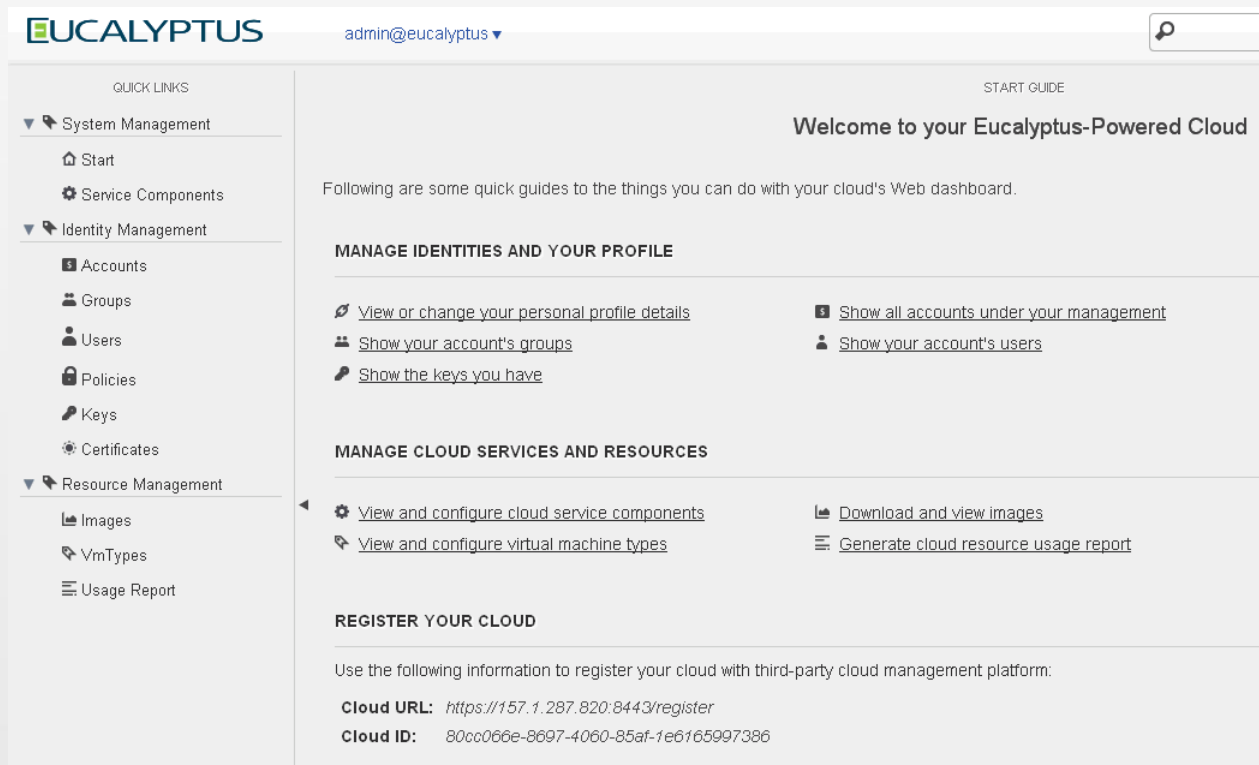
First time login. Please fill in the following information:

Email	<input type="text"/>
Old password	<input type="password"/>
New password	<input type="password"/>
Type again	<input type="password"/>

✕ Cancel ✓ OK

Web管理画面のトップページ

- ログインが成功すると以下のようなWeb管理画面のトップページに遷移します





ツールの準備

証明書類の取得

- CLCのマシンにて以下を実行し、euca2oolsを使用する環境を準備します

```
# mkdir -p ~/.euca/admin/keypairs
# euca_conf --get-credentials ~/.euca/euca2-admin-x509.zip
# cd ~/.euca/admin/
# unzip ../euca2-admin-x509.zip
# source eucarc
```

- CLCのマシンにて以下を実行し、利用可能なリソースの情報が取得できることを確認します

```
# euca-describe-availability-zones verbose
AVAILABILITYZONE      jp-east-1a      157.1.206.61 arn:euca:eucalyptus:jp-east-
1a:cluster:cluster0/
AVAILABILITYZONE      | - vm types    free / max    cpu    ram    disk
AVAILABILITYZONE      | - m1.small    0006 / 0006   1      128    2
AVAILABILITYZONE      | - c1.medium   0006 / 0006   1      256    5
AVAILABILITYZONE      | - m1.large    0002 / 0002   2      512    10
AVAILABILITYZONE      | - m1.xlarge   0002 / 0002   2      1024   20
AVAILABILITYZONE      | - c1.xlarge   0000 / 0000   4      2048   20
```

- NCの合計コア数とmaxの値が一致しない点については後程説明します。



マシンイメージの登録

マシンイメージの登録 -1-

- CLCのマシンにて以下を実行し、マシンイメージをCLCにダウンロードします

```
# cd ~/
# wget http://[VirbrのIPアドレス]/euca-centos-2011.07.02-x86_64.tgz
```

- ダウンロードしたマシンイメージを展開します

```
# tar -xvzf euca-centos-2011.07.02-x86_64.tgz
# cd euca-centos-2011.07.02-x86_64/
```

マシンイメージの登録 -2-

- まず最初にカーネルイメージを登録します。以下を実行して、登録するイメージをバンドル化します

```
# euca-bundle-image -i kvm-kernel/vmlinuz-2.6.28-11-generic --kernel true
Checking image
Compressing image
Encrypting image
Splitting image...
Part: vmlinuz-2.6.28-11-generic.part.00
Generating manifest /tmp/vmlinuz-2.6.28-11-generic.manifest.xml
```

- 次にWalrusにイメージをアップロードします

```
# euca-upload-bundle -b kernel.001 -m /tmp/vmlinuz-2.6.28-11-generic.manifest.xml
Checking bucket: kernel.001
Creating bucket: kernel.001
Uploading manifest file
Uploading part: vmlinuz-2.6.28-11-generic.part.00
Uploaded image as kernel.001/vmlinuz-2.6.28-11-generic.manifest.xml
```

- 次にWalrusにアップロードしたイメージを登録します

```
# euca-register kernel.001/vmlinuz-2.6.28-11-generic.manifest.xml
IMAGE eki-804E375E
```


マシンイメージの登録 -3-

- 次にRAMディスクイメージを登録します。以下を実行して、登録するイメージをバンドル化します

```
# euca-bundle-image -i kvm-kernel/initrd.img-2.6.28-11-generic --ramdisk true
Checking image
Compressing image
Encrypting image
Splitting image...
Part: initrd.img-2.6.28-11-generic.part.00
Generating manifest /tmp/initrd.img-2.6.28-11-generic.manifest.xml
```

- 次にWalrusにイメージをアップロードします

```
# euca-upload-bundle -b ramdisk.001 -m /tmp/initrd.img-2.6.28-11-generic.manifest.xml
Checking bucket: ramdisk.001
Creating bucket: ramdisk.001
Uploading manifest file
Uploading part: initrd.img-2.6.28-11-generic.part.00
Uploaded image as ramdisk.001/initrd.img-2.6.28-11-generic.manifest.xml
```

- 次にWalrusにアップロードしたイメージを登録します

```
# euca-register ramdisk.001/initrd.img-2.6.28-11-generic.manifest.xml
IMAGE eri-E2B23EAE
```

マシンイメージの登録 -4-

- 最後にマシンイメージを登録します。以下を実行して、登録するイメージをバンドル化します
 - オプション「--kernel」にはカーネルイメージを登録した際に出力されたekiを指定します
 - オプション「--ramdisk」にはRAMディスクイメージを登録した際に出力されたeriを指定します

```
# # euca-bundle-image -i euca-centos-2011.07.02-x86_64.img --kernel eki-804E375E --ramdisk eri-
E2B23EAE
Checking image
Compressing image
Encrypting image
Splitting image...
Part: euca-centos-2011.07.02-x86_64.img.part.00
Part: euca-centos-2011.07.02-x86_64.img.part.01
(中略)
Part: euca-centos-2011.07.02-x86_64.img.part.32
Part: euca-centos-2011.07.02-x86_64.img.part.33
Generating manifest /tmp/euca-centos-2011.07.02-x86_64.img.manifest.xml
```

マシンイメージの登録 -5-

■ Walrusにイメージをアップロードします

```
# euca-upload-bundle -b centos.001 -m /tmp/euca-centos-2011.07.02-x86_64.img.manifest.xml
Checking bucket: centos.001
Creating bucket: centos.001
Uploading manifest file
Uploading part: euca-centos-2011.07.02-x86_64.img.part.00
Uploading part: euca-centos-2011.07.02-x86_64.img.part.01
(中略)
Uploading part: euca-centos-2011.07.02-x86_64.img.part.32
Uploading part: euca-centos-2011.07.02-x86_64.img.part.33
Uploaded image as centos.001/euca-centos-2011.07.02-x86_64.img.manifest.xml
```

■ Walrusにアップロードしたイメージを登録します

```
# euca-register centos.001/euca-centos-2011.07.02-x86_64.img.manifest.xml
IMAGE emi-F33637CC
```

■ 以下のコマンドで登録されたイメージの一覧が確認できます

```
# euca-describe-images emi-F33637CC
IMAGE emi-E7F839FA centos.001/euca-centos-2011.07.02-x86_64.img.manifest.xml
523082990801 available public i386 machine eki-17DB39A2 eri-5635389F
instance-store
```



インスタンスの起動と接続

キーペアの登録とインスタンスの起動

- まずインスタンスを起動する前に、接続用のキーペアを作成します。以下をCLCで実行してください。

```
# euca-add-keypair key01 > ~/.euca/admin/keypairs/key01.pem  
# chmod 400 ~/.euca/admin/keypairs/key01.pem
```

- 登録したイメージを以下のように起動します

```
# euca-run-instances -k key01 emi-F33637CC  
RESERVATION      r-8EFE44B1      413900018582      default  
INSTANCE         i-31EB3D6B      emi-F33637CC      0.0.0.0 0.0.0.0 pending key01    0  
m1.small         2012-07-01T16:59:34.123Z  jp-east-1a      eki-804E375E      eri-E2B23EAE  
monitoring-disabled  0.0.0.0 0.0.0.0      instance-store
```

- 以下のようにWatchコマンドで流しておくことで、インスタンスの状態を確認し続けることが可能です

```
# watch -d euca-describe-instances i-31EB3D6B
```

セキュリティグループの設定と接続

- 起動したインスタンスに接続するためにはセキュリティグループを適切に設定する必要があります。
- 大雑把な設定になってしまいますが、例えば以下のように実行しSSH接続が可能な状態にします

```
# euca-authorize -P tcp -p 22 -s 0.0.0.0/0 default
GROUP    default
PERMISSION  default  ALLOWS  tcp      22      22      FROM    CIDR    0.0.0.0/0
```

- pingに応答させるようにするためには、以下のように実行します

```
# euca-authorize -P icmp -t 1:3 -s 0.0.0.0/0 default
GROUP    default
PERMISSION  default  ALLOWS  icmp     1       3       FROM    CIDR    0.0.0.0/0
```

- インスタンスが起動したら、以下のように実行してインスタンスに接続できるか確認します

```
# ssh -i ~/.euca/admin/keypairs/key01.pem 157.1.206.64
```



参考資料 - CENTOS 5にインストールする-

インストール前のOS設定 -セキュリティ設定-

- ファイアウォールとSELinuxの設定
 - 各物理マシンにログインし、setupコマンドを実行して以下の図のように「Security Level:」をDisabledに変更し、「SELinux:」をDisabledに変更して、「OK」を選択



- 設定後、再起動してもいいですが、インストール前のOS設定が終わった後で再起動するので、ここは再起動せずに次の設定を行ないます

インストール前のOS設定

-不要なサービスの停止-

- Eucalyptusを動かすために不要となるサービスを停止するように設定します。

- 各物理マシンにて実行

```
chkconfig iptables off  
chkconfig ip6tables off  
/etc/init.d/iptables stop
```

- NCにて実行

```
chkconfig libvirtd off
```

- 上記以外にも特にEucalyptusでは使用しないサービスは停止しても構いません

インストール前のOS設定

-ループバックデバイスの設定-

- 何も設定しない場合、利用可能なループバックデバイスの最大数は8個までとなっているため、利用可能な最大数を増やす必要があります。
- なお、ループバックデバイスを増やす必要があるのは、SCとNCです。
- CentOSでは以下のように実行することで、256個まで利用可能になります。

```
echo 'options loop max_loop=256' > /etc/modprobe.d/loop  
for i in `seq 8 255`; do echo loop${i} >> /etc/udev/makedev.d/50-udev.nodes; done
```

インストール前のOS設定

-Xenの設定-

- NCはlibvirtを使用してXenを制御するため、以下のように実行してXenがlibvirt経由で操作されるように設定します。

```
sed -i -e 's/#¥((xend-http-server ¥)no)/¥1yes)/' /etc/xen/xend-config.sxp  
sed -i -e 's/#¥((xend-address localhost)¥)/¥1/' /etc/xen/xend-config.sxp
```

インストール前のOS設定

-ntpの設定 その1-

- Eucalyptusの各コンポーネントは一定以上時刻がズレると正しく通信できないため、各コンポーネントでntpの設定をする必要があります。
- 今回はCLC/WalrusはCC/SCにサービスを提供し、CC/SCはNCにサービスを提供するような設定にします。
 - 今回はシングルクラスタ構成のため、ntpサーバはCLCのみでも良いと思えるかもしれないが、マルチクラスタ構成にした場合はNCが所属するCCに問い合わせに行くほうが構成的に綺麗

インストール前のOS設定

-ntpの設定 その2-

- 各物理マシンの/etc/ntp.confを設定
 - CLC/Walrusでの設定
 - 上位サーバがある場合は上位サーバに問い合わせにいくよう設定
 - CC/SCに対して問い合わせを許可する
 - 上位サーバがある場合、設定後に「ntpddate -b 上位サーバ」を2～3回実行し、上位サーバと時刻を合わせてから「/etc/init.d/ntpd start」を実行する
 - CC/SCでの設定
 - CLC/Walrusに問い合わせにいくよう設定
 - NCに対して問い合わせを許可する
 - 設定後に「ntpddate -b CLC/Walrus」を2～3回実行し、時刻を合わせてから「/etc/init.d/ntpd start」を実行する
 - NCでの設定
 - CC/SCに問い合わせにいくよう設定
 - 設定後に「ntpddate -b CC/SC」を2～3回実行し、時刻を合わせてから「/etc/init.d/ntpd start」を実行する



インストール前のOS設定

-最終確認と再起動-

- 最後に各設定項目の再確認を実施します
 - ファイアウォールとSELinuxの設定は行ったか？
 - 不要なサービスを停止する設定は行なったか？
 - libvirtが利用できるようXenの設定は行なったか？
 - ntpの設定は行なったか？
 - 各物理マシンにてntpq -pを実行し、上位サーバと同期ができていることを確認
- 再確認をし、問題がなければOSを再起動します。

依存ライブラリや関連パッケージのインストール -1-

- yumコマンドを利用して以下の関連パッケージをインストールします。

- CLC/Walrusにて

```
yum install -y java-1.6.0-openjdk ant ant-nodeps httpd perl-Convert-ASN1
```

- CC/SCにて

```
yum install -y ant ant-nodeps dhcp bridge-utils httpd scsi-target-utils ¥  
perl-Convert-ASN1
```

- NCにて

```
yum install -y bridge-utils httpd perl-Convert-ASN1
```

依存ライブラリや関連パッケージのインストール -2-

- 各物理マシンにて以下を実行し、tar玉を展開します

```
tar xzvf eucalyptus-2.0.3-centos-x86_64.tar.gz  
cd eucalyptus-2.0.3-centos-x86_64/eucalyptus-2.0.3-rpm-deps-x86_64/
```

- rpmコマンドを利用して以下のように関連パッケージをインストールします。

- CLC/WalrusやCC/SCにて

```
rpm -Uvh aoetools-21-1.el4.x86_64.rpm ¥  
euca-axis2c-1.6.0-1.x86_64.rpm ¥  
euca-rampartc-1.3.0-1.x86_64.rpm ¥  
vblade-14-1mdv2008.1.x86_64.rpm ¥  
vtun-3.0.2-1.el5.rf.x86_64.rpm ¥  
lzo2-2.02-3.el5.rf.x86_64.rpm ¥  
perl-Crypt-OpenSSL-Random-0.04-1.el5.rf.x86_64.rpm ¥  
perl-Crypt-OpenSSL-RSA-0.25-1.el5.rf.x86_64.rpm ¥  
perl-Crypt-X509-0.32-1.el5.rf.noarch.rpm ¥  
python25-2.5.1-bashton1.x86_64.rpm ¥  
python25-devel-2.5.1-bashton1.x86_64.rpm ¥  
python25-libs-2.5.1-bashton1.x86_64.rpm  
cd ..
```




依存ライブラリや関連パッケージのインストール -3-

■ NCにて

```
rpm -Uvh aoetools-21-1.el4.x86_64.rpm ¥  
euca-axis2c-1.6.0-1.x86_64.rpm ¥  
euca-rampartc-1.3.0-1.x86_64.rpm ¥  
perl-Crypt-OpenSSL-Random-0.04-1.el5.rf.x86_64.rpm ¥  
perl-Crypt-OpenSSL-RSA-0.25-1.el5.rf.x86_64.rpm ¥  
perl-Crypt-X509-0.32-1.el5.rf.noarch.rpm ¥  
python25-2.5.1-bashton1.x86_64.rpm ¥  
python25-devel-2.5.1-bashton1.x86_64.rpm ¥  
python25-libs-2.5.1-bashton1.x86_64.rpm  
cd ..
```

Eucalyptusのインストール

■ CLC/Walrusにて以下を実行

```
rpm -Uvh eucalyptus-2.0.3-1.x86_64.rpm ¥  
eucalyptus-common-java-2.0.3-1.x86_64.rpm ¥  
eucalyptus-cloud-2.0.3-1.x86_64.rpm ¥  
eucalyptus-walrus-2.0.3-1.x86_64.rpm
```

■ CC/SCにて以下を実行

```
rpm -Uvh eucalyptus-2.0.3-1.x86_64.rpm ¥  
eucalyptus-common-java-2.0.3-1.x86_64.rpm ¥  
eucalyptus-sc-2.0.3-1.x86_64.rpm ¥  
eucalyptus-cc-2.0.3-1.x86_64.rpm ¥  
eucalyptus-gl-2.0.3-1.x86_64.rpm
```

■ NCにて以下を実行

```
rpm -Uvh eucalyptus-2.0.2-1.x86_64.rpm ¥  
eucalyptus-gl-2.0.2-1.x86_64.rpm ¥  
eucalyptus-nc-2.0.2-1.x86_64.rpm
```

euca2oolsのインストール

- 本講ではCLC/Walrus環境にてeuca2oolsを使用しつつ説明するため、euca2oolsのインストール先はCLC/Walrusの物理マシンにします
- euca2oolsで必要となる依存パッケージのインストール

```
yum -y install swig rsync
```

■ euca2oolsをインストール

```
tar xzvf euca2ools-1.3.1-centos-x86_64.tar.gz  
cd euca2ools-1.3.1-centos-x86_64/  
rpm -ivh euca2ools-1.3.1-1.x86_64.rpm
```



EUCALYPTUSの設定

MANAGEDモードの設定例

■ CC/SCのeucalyptus.confを以下のように設定します

```
VNET_PUBINTERFACE="eth0"  
VNET_PRIVINTERFACE="eth1"  
VNET_MODE="MANAGED"  
VNET_SUBNET="10.1.0.0"  
VNET_NETMASK="255.255.0.0"  
VNET_ADDRSPERNET="64"  
VNET_PUBLICIPS="192.168.1.120-192.168.1.220"  
VNET_DNS="192.168.1.10"
```

■ NCのeucalyptus.confを以下のように設定します

```
VNET_PUBINTERFACE="eth1"  
VNET_MODE="MANAGED"
```

各パラメータの説明 -1-

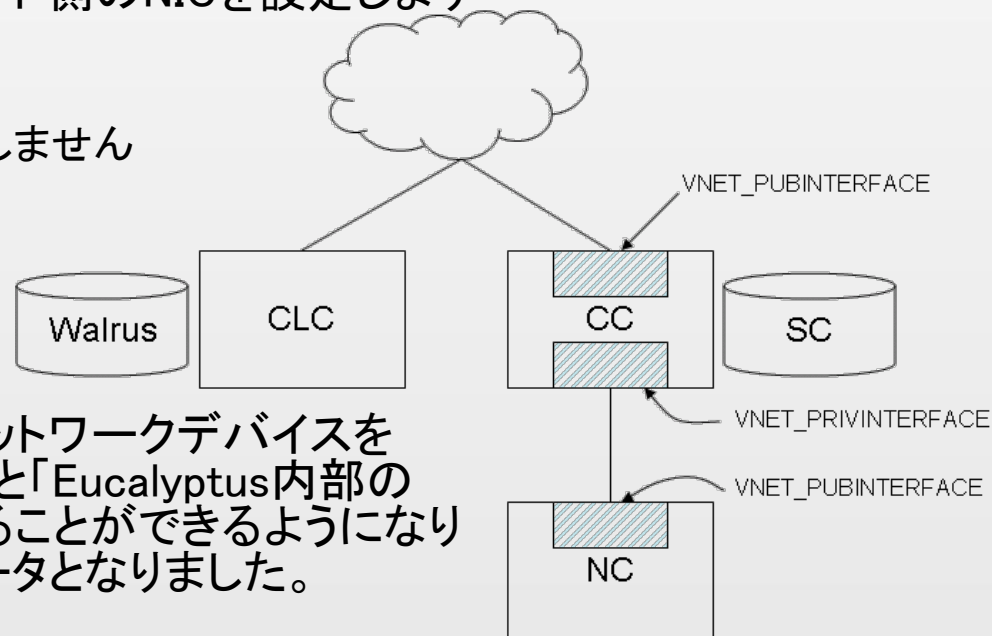
■ VNET_PUBINTERFACE

- CC/SCにおけるサービスセグメント側のNICを設定します
- NCにおけるバックエンド側——つまりCC/SCと繋がるNICを設定します

■ VNET_PRIVINTERFACE

- CC/SCにおけるバックエンド側のNICを設定します

■ NCではVNET_PRIVINTERFACEは使用しません



- この2つのパラメータによりCCのネットワークデバイスを「Eucalyptus外部のネットワーク用」と「Eucalyptus内部のネットワーク用」と明示的に指定することができるようになりましたが、やや理解しにくいパラメータとなりました。

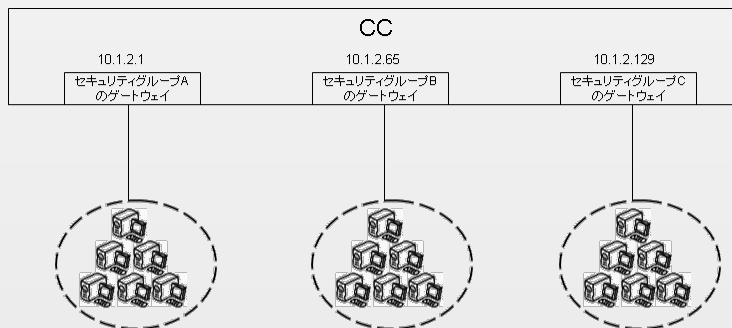
各パラメータの説明 -2-

■ VNET_MODE

- 前述したネットワークモードを設定します

■ VNET_SUBNET, VNET_NETMASK, VNET_ADDRSPERNET

- これらはいずれも仮想マシンに割り当てられるプライベートIP用のネットワークを表します。
- 下記の様に設定した場合、仮想マシンのプライベートIPは、セキュリティグループごとに「10.1.2.0～10.1.2.63」、「10.1.2.65～10.1.2.127」のように64個ずつ(うち2個はネットワークアドレスとブロードキャスト)が使われます。



	セキュリティグループA	セキュリティグループB	セキュリティグループC
インスタンスに使用されるIPアドレス	10.1.2.2～10.1.2.62	10.1.2.66～10.1.2.126	10.1.2.130～10.1.2.190
ネットワークアドレス	10.1.2.0	10.1.2.64	10.1.2.128
ブロードキャスト	10.1.2.63	10.1.2.127	10.1.2.191

各パラメータの説明 -3-

■ VNET_PUBLICIPS

- インスタンスに割り当てるパブリックIPアドレスを設定します。これらは利用者端末からインスタンスを利用する場合のIPアドレスとなります。ハイフンを使用して区切ることで範囲を設定することができます。

```
VNET_PUBLICIPS="192.168.1.120-192.168.1.220"
```

- 連続していないIPアドレスを設定する場合は下記のようにスペースで区切ることで1つずつ設定することもできます

```
VNET_PUBLICIPS="192.168.1.120 192.168.1.121 192.168.1.122"
```

■ VNET_DNS

- インスタンスにプライベートIPアドレスを払い出す際に同時に知らせるDNSサーバのアドレスかホスト名を設定します。DNSサーバがない場合は下記のように空に設定します。(コメント化したままの状態だとCCが正しく起動できません)

```
VNET_DNS=""
```


各コンポーネントの起動

■ CLC/Walrusにて

```
/etc/init.d/eucalyptus-cloud start
```

■ CC/SCにて

```
/etc/init.d/eucalyptus-cloud start  
/etc/init.d/eucalyptus-cc start
```

■ NCにて

```
/etc/init.d/eucalyptus-nc start
```

コンポーネントの登録

- CLC/Walrusにて以下を実行し、フロントエンドのコンポーネントを登録します

```
# Walrusの登録 (IPアドレスはフロントエンドのVNET_PUBINTERFACEに設定した  
# ネットワークデバイスのIPアドレスを指定)  
euca_conf --register-walrus 192.168.1.100  
  
# CCの登録 ("cluster0"は任意の名称、IPアドレスはフロントエンドの  
# VNET_PRIVINTERFACEに設定したネットワークデバイスのIPアドレスを指定)  
euca_conf --register-cluster cluster0 192.168.2.100  
  
# SCの登録 ("cluster0"はCCの登録で指定した名称、IPアドレスはフロントエンドの  
# VNET_PRIVINTERFACEに設定したネットワークデバイスのIPアドレスを指定)  
euca_conf --register-sc cluster0 192.168.2.100
```

- CC/SCにて以下を実行し、NCのコンポーネントを登録します

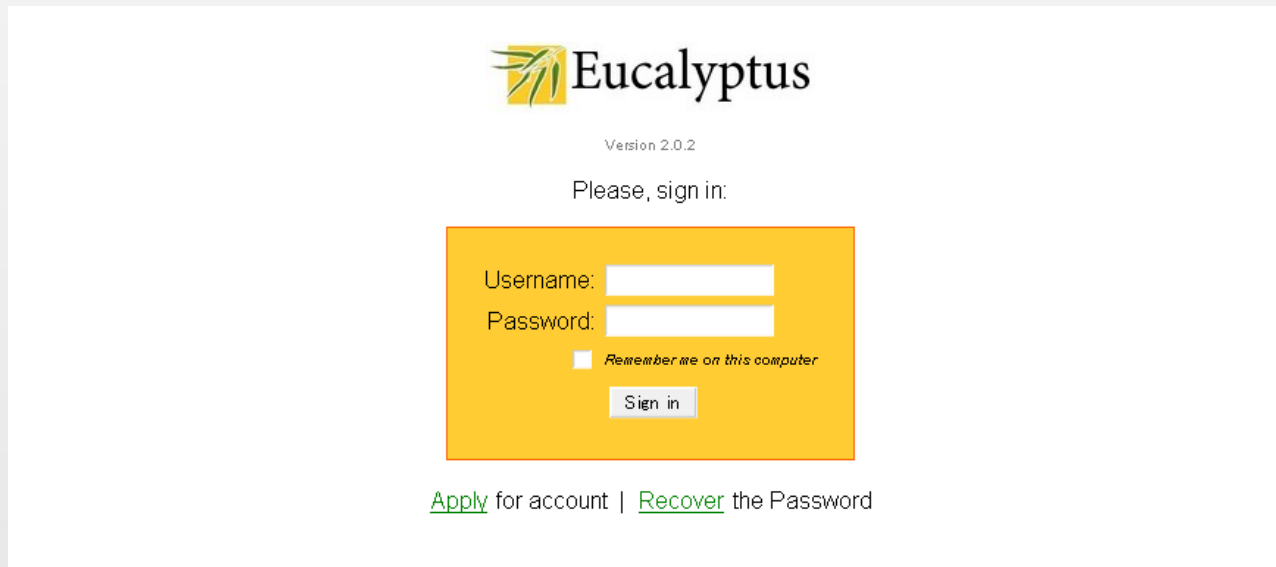
```
# NCの登録 (IPアドレスはNCのVNET_PUBINTERFACEに設定した  
# ネットワークデバイスのIPアドレスを指定)  
euca_conf --register-nodes "192.168.2.101"
```

WUIでの設定 -1-

- WEB管理画面から管理者アカウントの設定とEBSの設定をします。WEBブラウザで以下のURLにアクセスし、WEB管理画面を表示します。

`https://<FRONTEND_IP>:8443/`

- ログイン画面が表示されます。Username、Passwordともに、「admin」を入力し、ログインします。



Eucalyptus

Version 2.0.2

Please, sign in:

Username:

Password:

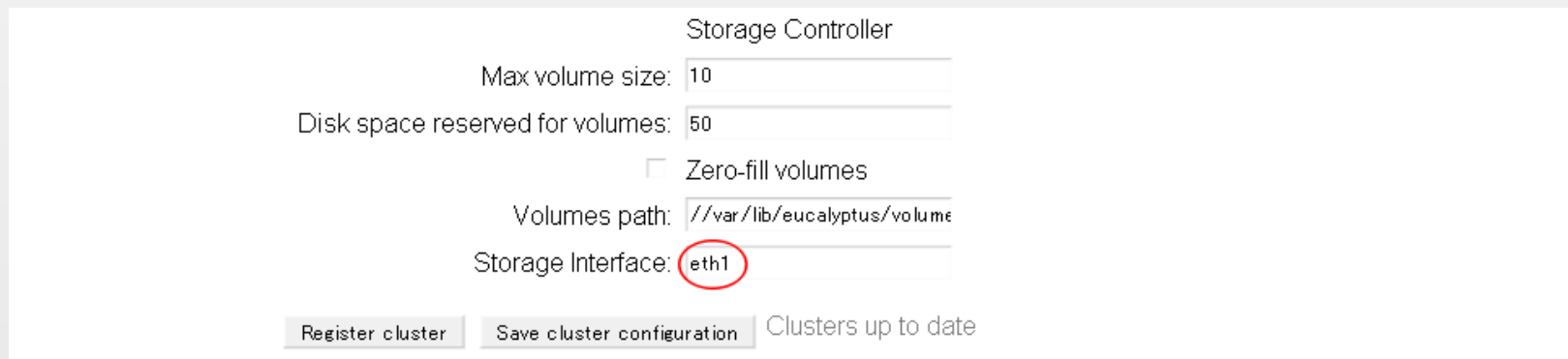
☐ Remember me on this computer

Sign in

[Apply](#) for account | [Recover](#) the Password

WUIでの設定 -2-

- ログイン後に表示される画面にてConfiguration タブをクリックすると、eucalyptus.confファイルやeuca_confコマンドだけでは設定できない項目が表示されます。
- そのなかで、「Clusters:」セクションにある「Storage Controller」の「Interface:」の値を「eth0」から「eth1」に変更し、「Save cluster configuration」ボタンをクリックして設定を保存します。これは、SCが提供するEBS機能をフロントエンドの「VNET_PRIVINTERFACE」である「eth1」のネットワークを通して提供するためです。

The screenshot shows a web interface for configuring the Storage Controller. The title 'Storage Controller' is at the top right. Below it are several input fields: 'Max volume size:' with the value '10', 'Disk space reserved for volumes:' with the value '50', a checkbox for 'Zero-fill volumes' which is unchecked, 'Volumes path:' with the value '//var/lib/eucalyptus/volume', and 'Storage Interface:' with the value 'eth1'. The 'eth1' value is circled in red. At the bottom, there are three buttons: 'Register cluster', 'Save cluster configuration', and 'Clusters up to date'.

Storage Controller

Max volume size: 10

Disk space reserved for volumes: 50

☐ Zero-fill volumes

Volumes path: //var/lib/eucalyptus/volume

Storage Interface: eth1

Register cluster Save cluster configuration Clusters up to date



EUCALYPTUSの動作確認

プロセスの確認

■ CLC/Walrusのプロセスを確認する

```
ps uww -C eucalyptus-cloud
```

■ CC/SCのプロセスを確認する

```
ps uww -C eucalyptus-cloud  
ps uww -C httpd
```

■ NCのプロセスを確認する

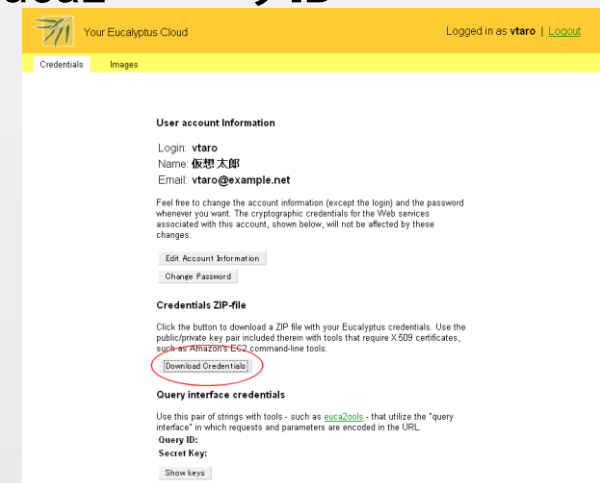
```
ps uww -C httpd
```



ツールの準備

証明書ファイルの取得 -1-

- CUI(euca2ools)を利用するには、X.509の証明書やユーザ情報などが入ったzipファイルを用意する必要があります。ユーザ「admin」はCLCでeuca_confコマンドあるいはeuca-get-credentialsコマンドを用いてこのzipファイルを取得することができますが、それ以外のユーザはWeb管理画面からダウンロードする必要があります。
- まず、Web管理画面にログインし、最初に表示される[Credentials]タブのページにある項目「Credentials ZIP-file」の[Download Credentials]ボタンをクリックし、当該ユーザのeuca2-ユーザID-x509.zipというファイルをダウンロードします。



証明書ファイルの取得 -2-

- ダウンロードしたeuca2-ユーザID-x509.zipを任意のLinux環境もしくはCLCにコピーし、以下のように展開します。(以下、ユーザIDは前章で作成したvtaroを用いて説明します)

```
# euca2-ユーザID-x509.zipファイルをHOMEディレクトリに置くことを前提とする
cd ~
# このあとの作業でキーペアを格納するディレクトリも一緒に作成する
mkdir -p .euca/vtaro/keypairs
cd .euca/vtaro/
# ~/ に置いてあるファイルをunzipコマンドで展開
unzip ../../euca2-vtaro-x509.zip
# 展開したファイル中にあるeucarcを読み込む
source eucarc
```



マシンイメージの登録

カーネルイメージの登録 -1-

- まず最初に、カーネルイメージを以下のように `euca-bundle-image` コマンドでバンドルします。

```
euca-bundle-image -i xen-kernel/vmlinuz-2.6.27.21-0.1-xen --kernel true
Checking image
Tarring image
Encrypting image
Splitting image...
Part: vmlinuz-2.6.27.21-0.1-xen.part.0
Generating manifest /tmp/vmlinuz-2.6.27.21-0.1-xen.manifest.xml
```

カーネルイメージの登録 -2-

- バンドルした結果生成されたmanifestファイルをeuca-upload-bundleコマンドでWalrusにアップロードします。

```
euca-upload-bundle -b kernel.001 -m /tmp/vmlinuz-2.6.27.21-0.1-xen.manifest.xml
Checking bucket: kernel.001
Creating bucket: kernel.001
Uploading manifest file
Uploading part: vmlinux-2.6.27.21-0.1-xen.part.0
Uploaded image as kernel.001/vmlinuz-2.6.27.21-0.1-xen.manifest.xml
```

- アップロードしたイメージのmanifestファイルをeuca-registerコマンドで登録します。

```
euca-register kernel.001/vmlinuz-2.6.27.21-0.1-xen.manifest.xml
IMAGE eki-5E491255
```



RAMディスクイメージの登録 -1-

- 次に、RAMディスクイメージを以下のようにeuca-bundle-imageコマンドでバンドルします。

```
euca-bundle-image -i xen-kernel/initrd-2.6.27.21-0.1-xen --ramdisk true
Checking image
Tarring image
Encrypting image
Splitting image...
Part: initrd-2.6.27.21-0.1-xen.part.0
Part: initrd-2.6.27.21-0.1-xen.part.1
Generating manifest /tmp/initrd-2.6.27.21-0.1-xen.manifest.xml
```

RAMディスクイメージの登録 -2-

- バンドルした結果生成されたmanifestファイルをeuca-upload-bundleコマンドでWalrusにアップロードします。

```
euca-upload-bundle -b ramdisk.001 -m /tmp/initrd-2.6.27.21-0.1-xen.manifest.xml
Checking bucket: ramdisk.001
Creating bucket: ramdisk.001
Uploading manifest file
Uploading part: initrd-2.6.27.21-0.1-xen.part.0
Uploading part: initrd-2.6.27.21-0.1-xen.part.1
Uploaded image as ramdisk.001/initrd-2.6.27.21-0.1-xen.manifest.xml
```

- アップロードしたイメージのmanifestファイルをeuca-registerコマンドで登録します。

```
euca-register ramdisk.001/initrd-2.6.27.21-0.1-xen.manifest.xml
IMAGE eri-5E491255
```

マシンイメージの登録 -1-

- 最後に、マシンイメージを以下のようにeuca-bundle-imageコマンドでバンドルします。

```
euca-bundle-image -i centos.5-3.x86-64.img --kernel eki-5E491255 --ramdisk eri-596A123C
Checking image
Tarring image
Encrypting image
Splitting image...
Part: centos.5-3.x86-64.img.part.0
Part: centos.5-3.x86-64.img.part.1
(中略)
Part: centos.5-3.x86-64.img.part.60
Part: centos.5-3.x86-64.img.part.61
Generating manifest /tmp/ centos.5-3.x86-64.img.manifest.xml
```

マシンイメージの登録 -2-

- バンドルした結果生成されたmanifestファイルをeuca-upload-bundleコマンドでWalrusにアップロードします。

```
euca-upload-bundle -b machine.001 -m /tmp/centos.5-3.x86-64.img.manifest.xml
Checking bucket: machine.001
Creating bucket: machine.001
Uploading manifest file
Uploading part: centos.5-3.x86-64.img.part.0
Uploading part: centos.5-3.x86-64.img.part.1
(中略)
Uploading part: centos.5-3.x86-64.img.part.60
Uploading part: centos.5-3.x86-64.img.part.61
Uploaded image as machine.001/centos.5-3.x86-64.img.manifest.xml
```

- アップロードしたイメージのmanifestファイルをeuca-registerコマンドで登録します。

```
euca-register machine.001/centos.5-3.x86-64.img.manifest.xml
IMAGE emi-330411BE
```




ユーザの作成とツールの準備

ユーザの作成方法

- ユーザを作成するためには3つの方法があります
 - 管理コマンドでユーザを作成
 - バッチ処理や一括登録などでユーザを作成する場合はこの方法が便利です
 - 管理者がWebUIで作成
 - 初期ユーザや管理者権限を有するユーザを作成する場合などはこの方法が適しています
 - 利用者がWebUIで申請
 - 申請ベースでユーザ作成を行なうような場合はこの方法が適しています

管理コマンドでユーザを作成

- Eucalyptus 2.0系から実装された管理コマンドにはユーザを作成するコマンド `euca-add-user` があります。以下のように実行することでユーザを作成することができます。
- ただし、コマンドで作成したユーザは無効なユーザであるため、別途管理者によって有効化する必要があります。

```
# 実行前にeucarcによってEC2_で始まる環境変数が適宜設定されていること
euca-add-user -e vjiro@example.com vjiro
```

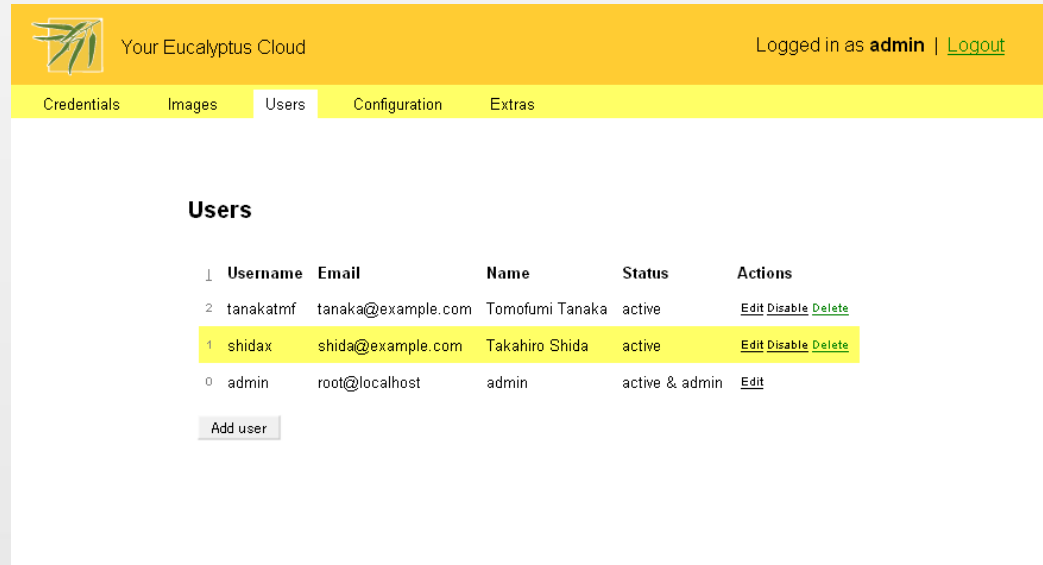
```
# 以下のコマンドでユーザを確認
```

```
euca-describe-users vjiro
```

USER	vjiro	n/a	DISABLED
USER-GROUP	vjiro	default	
USER-CERT	vjiro	None	None
USER-KEYS	vjiro	LQ1nA5MSy9fsZlaWh	QtT0SyuBKwiaIRRXa5 (略)
USER-CODE	vjiro	DgmxDmVqrzI63vUp7I7VBf58KA88WyuxyA	(略)
USER-WEB	vjiro	BwVhM9PvE0Rw0UC1RWU63j6oUGQyTBjsd	(略)

管理者がWebUIで作成 -1-

- WebUIでユーザを作成するためには、ユーザ「admin」が管理者権限を有するユーザがWebUIにログインし、[Users]タブをクリックしてユーザの一覧を表示します

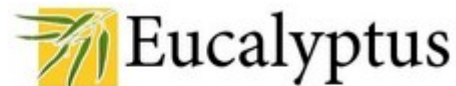
The screenshot shows the "Your Eucalyptus Cloud" interface. At the top, it says "Logged in as admin | Logout". Below this is a navigation bar with tabs: "Credentials", "Images", "Users" (which is selected), "Configuration", and "Extras". The main content area is titled "Users" and contains a table with the following data:

	Username	Email	Name	Status	Actions
2	tanakatmf	tanaka@example.com	Tomofumi Tanaka	active	Edit Disable Delete
1	shidax	shida@example.com	Takahiro Shida	active	Edit Disable Delete
0	admin	root@localhost	admin	active & admin	Edit

Below the table is a button labeled "Add user".

管理者がWebUIで作成 -2-

- ユーザー一覧画面の一番下にある[Add user]ボタンをクリックするとユーザ作成のページに遷移します。



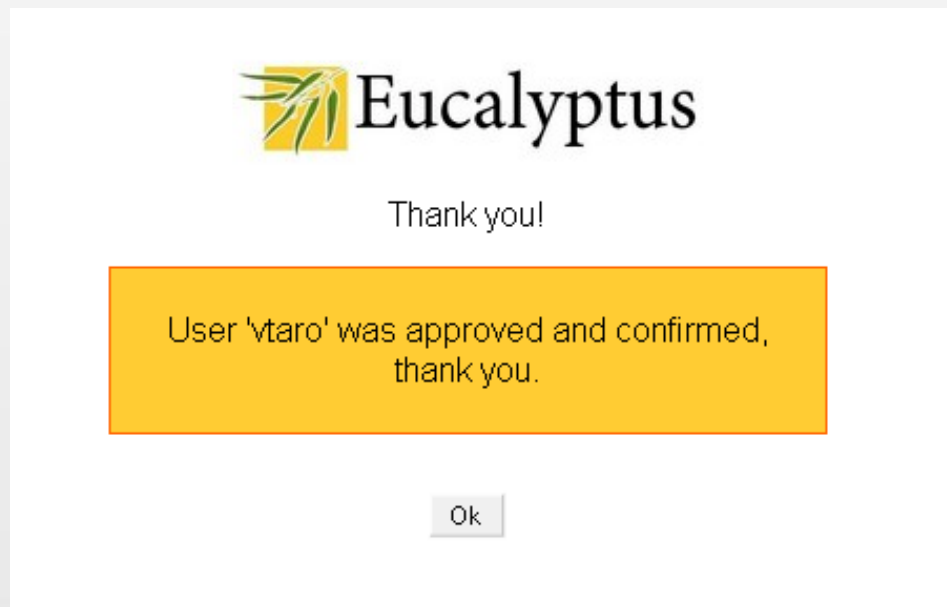
Please, fill out the form to add a user

Mandatory fields:

Username:	<input type="text" value="vtaro"/>	ログイン時に使用するためのユーザIDを入力(必須)
	<input type="checkbox"/> Administrator	管理者権限を付与する場合はチェックする
Password:	<input type="password" value="●●●●●●●●●●"/>	ログイン時に使用するパスワードを入力(必須)
Password, again:	<input type="password" value="●●●●●●●●●●"/>	上記で入力したパスワードを再入力(必須)
Full Name:	<input type="text" value="雲野 太郎"/>	フルネームを入力(必須)
Email address:	<input type="text" value="vtaro@example.com"/>	メールアドレスを入力(必須)
	<input checked="" type="checkbox"/> Skip email confirmation	メールによる確認を不要にする場合はチェック

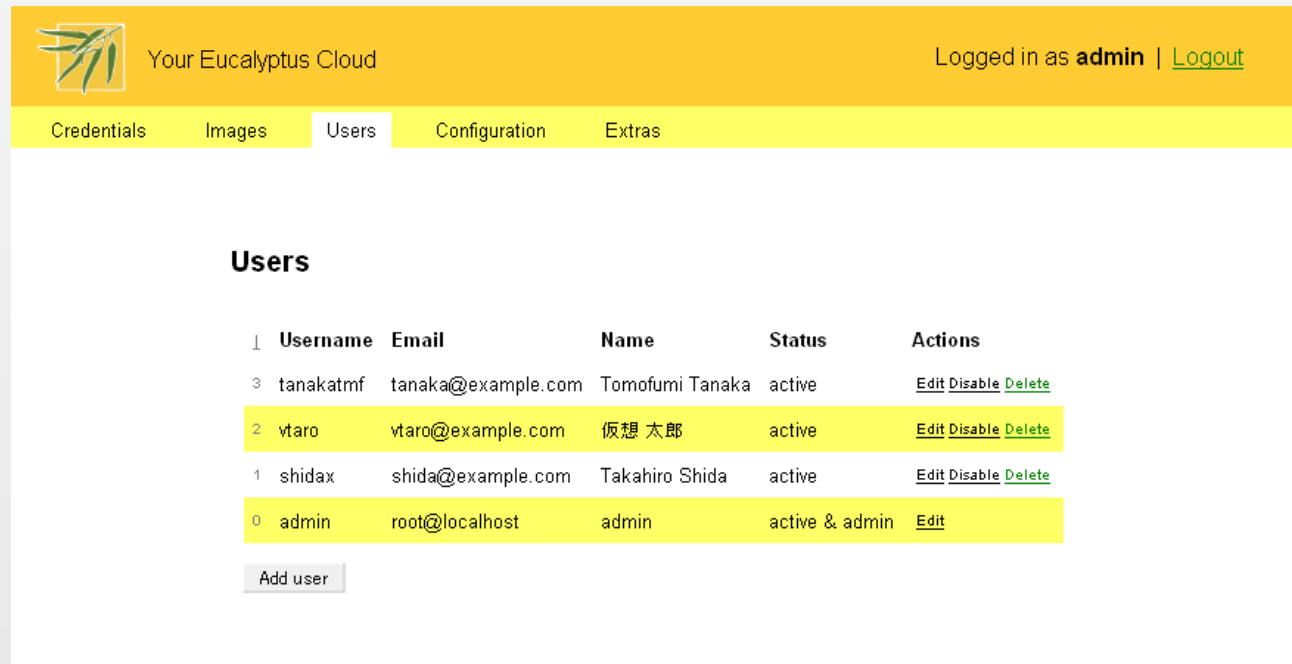
管理者がWebUIで作成 -3-

- 必要な値を入力したのち、[Add user]ボタンをクリックすると以下の画面に遷移し、ユーザが作成されます。



管理者がWebUIで作成 -4-

- ユーザの作成が完了するとユーザの一覧ページへと遷移し、作成したばかりのユーザも表示されていることが確認できます。



The screenshot shows the 'Your Eucalyptus Cloud' interface. At the top, it says 'Logged in as admin | Logout'. Below this is a navigation bar with tabs: 'Credentials', 'Images', 'Users' (selected), 'Configuration', and 'Extras'. The main content area is titled 'Users' and displays a table of users. The table has columns for 'Username', 'Email', 'Name', 'Status', and 'Actions'. There are four users listed: 'tanakatmf', 'vtaro', 'shidax', and 'admin'. The 'vtaro' row is highlighted in yellow. Below the table is an 'Add user' button.

	Username	Email	Name	Status	Actions
3	tanakatmf	tanaka@example.com	Tomofumi Tanaka	active	Edit Disable Delete
2	vtaro	vtaro@example.com	仮想 太郎	active	Edit Disable Delete
1	shidax	shida@example.com	Takahiro Shida	active	Edit Disable Delete
0	admin	root@localhost	admin	active & admin	Edit

[Add user](#)

利用者がWebUIで申請 -1-

- 利用者がWebUIでユーザの作成申請を行なう場合は、WebUIにアクセスし、ログインページの下部にある「Apply for account」をクリックすると、ユーザの作成申請の画面に遷移します。

Please, sign in:

Username:

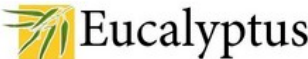
Password:

☐ *Remember me on this computer*

[Apply](#) for account | [Recover](#) the Password

利用者がWebUIで申請 -2-

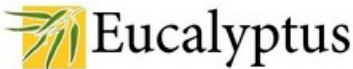
- 各項目に適宜情報を入力し、[Sign up]ボタンを押下するとユーザ作成の申請を受けつけた旨のメッセージが表示されます



Thank you!

Thank you for signing up! Your request has been forwarded to the cloud administrator. If your application is approved, you will receive an email message (at the address you specified) with instructions for activating your account.

Ok



Please, fill out the form:

Mandatory fields:

Username:

Password:

Password, again:

Full Name:

Email address:

Optional fields:

Telephone Number:

Project Leader:

Affiliation:

Project Description:

利用者がWebUIで申請 -3-

- 以下の内容のようなユーザ作成の申請が管理者にメールが届きます。

Someone has requested an account on the Eucalyptus system

Name: 仮想 次郎
Username: vjiro
Email address: vtaro@example.net
Telephone: 03-3000-0002
Affiliation: The V-Team
Project PI: vtaro
Project description:

=====
Something, something, something...
=====

To APPROVE this request, click on the following link:

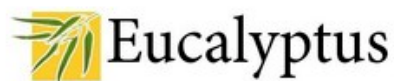
<https://192.168.1.10:8443/?action=approve&user=vjiro>

To REJECT this request, click on the following link:

<https://192.168.1.10:8443/?action=reject&user=vjiro>

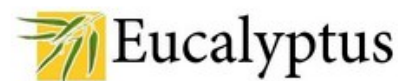
利用者がWebUIで申請 -4-

- 管理者はメールにあるリンクでユーザ作成の申請を承諾あるいは却下することができます。



Request to approve user 'vjiro' succeeded.

Ok



Request to reject user 'vjiro' succeeded.

Ok

利用者がWebUIで申請 -5-

- ユーザ作成の申請を承諾した場合、ユーザ作成の申請をした際に入力したユーザのメールアドレスに以下のようなメールが返ります。

Your Eucalyptus account was approved! Please, click on the following link to log into the system (with the login and password that you've chosen when signing up) and confirm your account:

<https://192.168.1.10:8443/?action=confirm&code=vLPCVdwRwyB9p8nZ1rBLKSmvwi47bg0BD0GrNBEV6UWwmGgySXTJL180h5pDmQkDLTyi80Stt7L1aqoxKZVM1w>

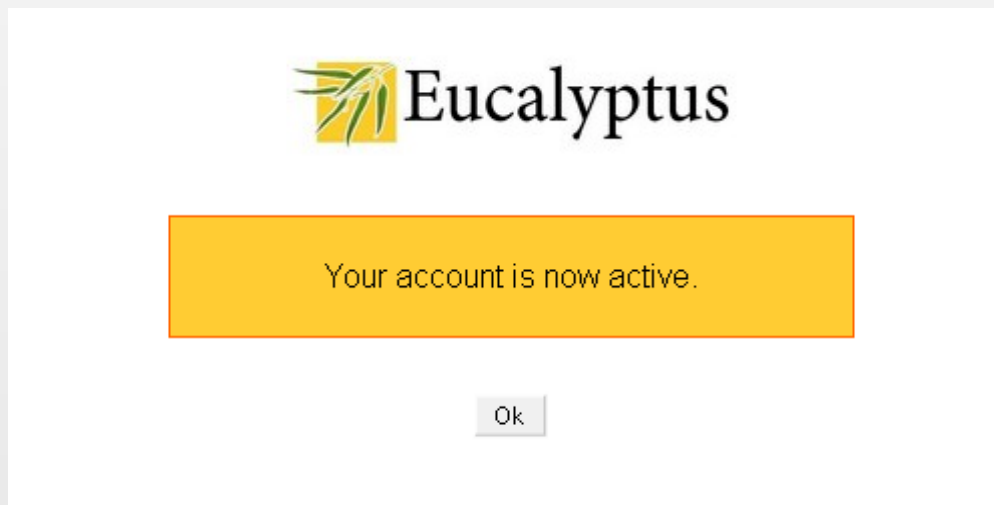
However, if you never requested a Eucalyptus account then, please, disregard this message.

- 却下した場合は以下のようなメールが返ります。

You or someone pretending to be you has requested an account on the Eucalyptus elastic cloud system. We are sorry to say that your request has been rejected by the cloud administrator. Please contact the administrator of the system for further information.

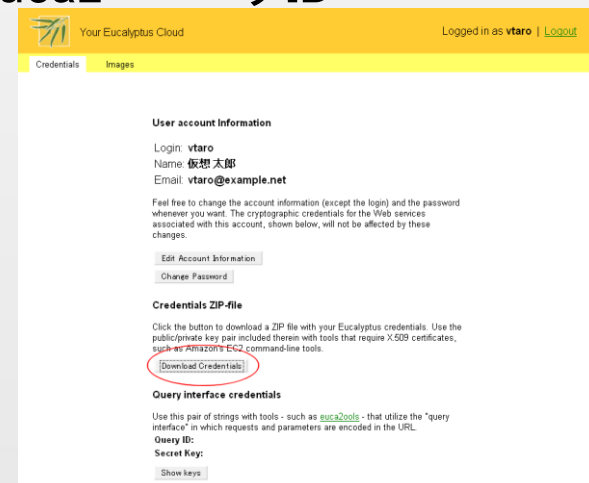
利用者がWebUIで申請 -6-

- 承諾された旨のメールがユーザに届いた場合、メールに記述されているURLにアクセスすることで、ユーザを活性化させることができます。



証明書ファイルの取得 -1-

- CUI(euca2ools)を利用するには、X.509の証明書やユーザ情報などが入ったzipファイルを用意する必要があります。ユーザ「admin」はCLCでeuca_confコマンドあるいはeuca-get-credentialsコマンドを用いてこのzipファイルを取得することができますが、それ以外のユーザはWeb管理画面からダウンロードする必要があります。
- まず、Web管理画面にログインし、最初に表示される[Credentials]タブのページにある項目「Credentials ZIP-file」の[Download Credentials]ボタンをクリックし、当該ユーザのeuca2-ユーザID-x509.zipというファイルをダウンロードします。



証明書ファイルの取得 -2-

- ダウンロードしたeuca2-ユーザID-x509.zipを任意のLinux環境もしくはCLCにコピーし、以下のように展開します。(以下、ユーザIDは前章で作成したvtaroを用いて説明します)

```
# euca2-ユーザID-x509.zipファイルをHOMEディレクトリに置くことを前提とする
cd ~
# このあとの作業でキーペアを格納するディレクトリも一緒に作成する
mkdir -p .euca/vtaro/keypairs
cd .euca/vtaro/
# ~/ に置いてあるファイルをunzipコマンドで展開
unzip ../../euca2-vtaro-x509.zip
# 展開したファイル中にあるeucarcを読み込む
source eucarc
```



インスタンスの起動と接続

キーペアの作成

- Linuxのインスタンスを起動してsshで接続するためにはキーペアと呼ばれるsshの公開鍵と秘密鍵のペアをEucalyptusに作成する必要があります。
- キーペアを作成するためにはeuca-add-keypairコマンドを利用して以下のように実行します。

```
# euca-add-keypairコマンドは実行結果を標準出力で返すため、  
# リダイレクトしてファイルに保存し、パーミッションを所有者だけが読み取れるよう変更します。
```

```
mkdir -p ~/.euca/vtaro/kaypairs  
euca-add-keypair key01 > ~/.euca/vtaro/kaypairs/key01.pk  
chmod 400 ~/.euca/vtaro/kaypairs/key01.pk
```

- 現在作成されているキーペアの一覧はeuca-describe-keypairsコマンドで確認できます。なお、この出力で表示されるフィンガープリントは前述のeuca-add-keypairで得られた出力の1行目と同じため、作成したキーペアと実際のファイルが同じキーペアか分からなくなった場合に比較することも可能です。

```
euca-describe-keypairs  
KEYPAIR key01 25:d9:00:a8:09:8e:03:3a:7a:1b:95:37:50:86:eb:94:36:54:1c:b3
```

セキュリティグループの作成

- インスタンスを起動してsshで接続するためにはセキュリティグループにルールを設定する必要があります。
 - セキュリティグループのルール設定はインスタンス後にも任意のタイミングで変更することが可能です
- ユーザごとにdefaultという名前のセキュリティグループが1つ用意されており、インスタンス起動時に特に指定しない場合はこのdefaultというセキュリティグループが使用されます
 - 用途によって使い分けられるよう、任意の名前でセキュリティグループを作成することも可能です。
- インスタンスを起動したあとからセキュリティグループ自体を別のセキュリティグループに変更することはできないため、あらかじめ用途別のセキュリティグループが必要な場合はインスタンスを起動する前に作成しておく必要があります。
- セキュリティグループを作成するためにはeuca-add-groupコマンドを利用して以下のように実行します。

```
euca-add-group -d "Web Server" web
```

インスタンスの起動

- インスタンスを起動するにはeuca-run-instancesコマンドを使用して以下のように実行します。

```
euca-run-instances -k key01 -t m1.small -n 1 emi-330411BE  
RESERVATION r-35CC06E6 vtaro vtaro-default  
INSTANCE i-410306FC emi-330411BE 0.0.0.0 0.0.0.0 pending key01 2010-06-  
24T13:46:41.218Z eki-5E491255 eri-596A123C
```

- インスタンスの起動には数十秒から数分ほど時間がかかりますので、euca-describe-instancesコマンドでインスタンスの状態を確認します。pendingがrunningになるとインスタンスが起動したことを示します。この状態はHypervisor上で起動処理が開始されたこと意味するため、実際にはrunningになってから更に仮想マシンのOS起動処理が終るまで数十秒ほど待つ必要があります。

```
euca-describe-instances emi-330411BE  
RESERVATION r-35CC06E6 vtaro default  
INSTANCE i-410306FC emi-330411BE 192.0.2.200 10.1.2.2 pending key01 0 m1.small  
2010-06-24T13:46:41.218Z cluster0 eki-5E491255 eri-596A123C
```

- 以下のようにwatchコマンドを用いて監視することもできます。

```
watch -d euca-describe-instances emi-330411BE
```



セキュリティグループの設定

- 前述したように、起動したインスタンスにssh接続するためにはセキュリティグループのルール設定に対して22/tcpを設定する必要があります。
- 例えば、192.168.1.0/24のセグメントから接続できるようにするためには以下のようにeuca-authorizeコマンドを利用して実行します。

```
euca-authorize -P tcp -p 22 -s 192.168.1.0/24 default  
GROUP default  
PERMISSION default ALLOWS tcp 22 22 FROM CIDR 192.168.1.0/24
```



インスタンスへのssh接続

- 起動したインスタンスにssh接続するためには、以下のようにsshコマンドを利用した一般的な接続方法で接続することができます。

```
ssh -i ~/.euca/vtaro/keypairs/key01.pk -o StrictHostKeyChecking=no 192.0.2.200
```

その他のインスタンス操作

■ インスタンスの再起動

```
euca-reboot-instances i-410306FC
```

■ インスタンスの停止 (EBS起動のみ)

```
euca-stop-instances i-410306FC
```

■ インスタンスの再開 (EBS起動のみ)

```
euca-start-instances i-410306FC
```

■ インスタンスの削除

```
euca-terminate-instances i-410306FC
```