Eucalyptus (1.0)

Elastic Computing, Utility Computing, and Cloud Computing are (possibly synonymous) terms referring to a popular SLA-based computing paradigm that allows users to "rent" Internet-accessible computing capacity on a for-fee basis. While a number of commercial enterprises currently offer Elastic/Utility/Cloud hosting services and several proprietary software systems exist for deploying and maintaining a computing Cloud, standards-based open-source systems have been few and far between.

EUCALYPTUS -- Elastic Utility Computing Architecture for Linking Your Programs To Useful Systems -- is an open-source software infrastructure for implementing Elastic/Utility/Cloud computing using computing clusters and/or workstation farms. The current interface to EUCALYPTUS is interface-compatible with Amazon.com's EC2 (arguably the most commercially successful Cloud computing service), but the infrastructure is designed to be modified and extended so that multiple client-side interfaces can be supported. In addition, EUCALYPTUS is implemented using commonly-available Linux tools and basic web service technology making it easy to install and maintain.

Overall, the goal of the EUCALYPTUS project is to foster community research and development of Elastic/Utility/Cloud service implementation technologies, resource allocation strategies, service level agreement (SLA) mechanisms and policies, and usage models. The current release is version 1.0 and it includes the following features:

- Interface compatibility with EC2
- Simple installation and deployment using Rocks cluster-management tools
- Simple set of extensible cloud allocation policies
- Overlay functionality requiring no modification to the target Linux environment
- Basic "Cloud Administrator" tools for system management and user accounting
- The ability to configure multiple clusters, each with private internal network addresses, into a single Cloud.

The initial version of EUCALYPTUS requires Xen to be installed on all nodes that can be allocated, but no modifications to the "dom0" installation or to the hypervisor itself.

Eucalyptus Features (1.0)

Supported

- Easy installation

    - Rocks-based "almost-one-button" binary install

- Administration
    - Adding users, adding/listing images, adding/deleting nodes, adding/deleting clusters

- EC2 compatibility:
    - In terms of command-line tools:
        - ec2-describe-images (currently lists admin-provided images)
        - ec2-add-keypair (key pair generation & upload) and ec2-describe-keypair
        - ec2-run-instances and ec2-terminate-instances
        - ec2-describe-instances

Unsupported

- Installation
    - Rocks-independent binary install
    - Experts-only source install

- Administration
    - Listing/deleting users, deleting images

- Amazon EC2 incompatibilities:
    - User-selectable images (bundle & upload)
    - User-selectable kernels
    - Elastic IP addresses
    - In terms of command-line tools:
        - ec2-authorize (firewall control)
        - ec2-bundle-vol (image creation)
        - ec2-register and ec2-deregister (image registration with EC2)
        - ec2-upload-bundle & ec2-delete-bundle (image upload/removal to S3)

Known Eucalyptus Bugs (1.0)

- Some EC2 command-line utilities will return a 'Read error' to the user, regardless of the actual internal error
    - what you should do: try the command again

- There is a condition where eucalyptus instances are started, but the system loses track of their existence.
    - what you should do: when users are done using their instances, the admin can forcefully shutdown ALL running instances by running '/opt/eucalyptus-1.0/usr/sbin/euca killall'

- Rarely, an instance will appear to be running even though the actual VM has been terminated
  - what you should do: when users are done using their instances, forcefully shutdown all instances and restart eucalyptus on the front-end
- Restarting eucalyptus while instances are running will result in those instances being unknown to the eucalyptus system
  - what you should do: when users are done using their instances, forcefully shutdown all instances and restart eucalyptus on the front-end
- Rarely, eucalyptus processes may be left running even after the init script has finished stopping the system (eucalyptus stop).
  - what you should do: search for eucalyptus processes and kill them manually (anything process with the string 'euca' contained in the commandline)
- Our log-files can grow rather large (we estimate several hundred megabytes per week).
  - what you should do: stop eucalyptus, remove or back up the log files (/opt/eucalyptus-1.0/var/log/eucalyptus/*), and restart eucalyptus

Frequently Asked Questions (1.0)

Is Eucalyptus a precise implementation of Amazon's EC2?

No. Eucalyptus supports Amazon's interface syntactically and it implements the same functionality (with a few exceptions), but internally it is almost certainly different. Eucalyptus is designed to be extensible and easy to install and maintain, particularly in a research environment, where system administrator time is the most expensive commodity. While we can't be certain, Amazon's main design goal almost has to be scalability. Put another way, if we were to design a commercial software venue for cloud services where scalability is paramount and we could mandate how all clusters within the cloud were initially configured (instead of an open-source software tool for community distribution) we would have designed Eucalyptus differently.

Why choose Amazon's EC2 interface for Eucalyptus?

The intention is to be able to support multiple cloud computing interfaces using the same "back end" infrastructure. EC2 seemed to be the best documented of the available choices at the time we began development and also the most commercially successful so we chose to implement it first. The interface module, however, can be replaced without changing the rest of the system (we hope).

Is Eucalyptus the product of a company?

No. Eucalyptus has been developed in the MAYHEM Lab within the Computer Science Department at the University of California, Santa Barbara primarily as a tool for cloud-computing research. It is distributed as open source with a FreeBSD-style license that does not restrict its usage much. We hope that it fosters further interest and development, both scientific and commercial, in cloud computing.

Do Eucalyptus users need a credit card to access the system?

No. In a commercial cloud, clients pay for their use with a credit card. Eucalyptus is designed to work in an environment where machines are available to a user community that accesses them via logins. Because user accountability usually must be ensured by the system administrators in such an environment, we have developed a "cloud administrator" interface for Eucalyptus that is more analogous to common practice in a setting without fees.

Cloud users must register with the cloud administrator using a Web-based sign-up page. The cloud administrator receives an email message requesting approval whenever a potential user requests access. The message contains links for approving or denying the request. The decision of the administrator is communicated to the requester via email. In the case of approval, the user will be able to obtain from a password-protected Web page the cryptographic certificates necessary for using Eucalyptus.

What software environments are supported?

Eucalyptus 1.0 targets Linux systems that use Xen (versions 3.*) for virtualization. It is packaged for Rocks V as a "roll" -- an ISO disk image with RPMs and some metadata that Rocks uses for cluster-wide installation. The RPM works as a stand-alone package as well, but installation outside Rocks is not supported in version 1.0. The client-side tools are those that are provided by Amazon for use with EC2.

What happens when all virtual machine "slots" are allocated?

The SLAs (Service Level Agreements) implemented by the cloud controller are bare-bones simple. Our intention is for Eucalyptus sites and development teams to come up with their own (we have a few in mind we'd like to try) but in the release of version 1.0, the simple SLAs supplied enforce no restrictions on occupancy duration. This type of SLA is comparable to the Amazon SLA with the exception that Amazon's will "time out" when charges to the registered credit card are no longer postable.

We have implemented no such time out analogy at present, so if users or the cloud administrator do not terminate running instances, eventually Eucalyptus will run out of VM slots to allocate. In this case, user requests fail and the EC2 tools report no instances available when the command "ec2-describe-availability-zones" is invoked.

What is an "Availability Zone" in Eucalyptus?

Amazon implements "availability zones" to allow users some degree of control over instance placement. Specifically, EC2 users can choose to host images in different availability zones if they wish to try and ensure independent failure probabilities. Amazon, presumably, takes steps to insulate instances in separate availability zones from correlated failure (e.g., a single power outage that takes out a data center).

Under Eucalyptus, the abstraction is slightly different. Each availability zone corresponds to a separate cluster within the Eucalyptus cloud. The advantage is that the networking within a single availability zone can be made much faster (i.e., it uses the cluster's private network in native mode). For allocations that span clusters, the technology Eucalyptus uses to implement a private network for each allocation imposes a substantial performance penalty.

Thus the two are similar in that cloud allocations to separate availability zones do reduce the chance of correlated failure. They are different, however, in that under Eucalyptus, each availability zone is restricted to a single "machine" (e.g., cluster) where at Amazon, the zones are much broader.

Does Eucalyptus require Rocks?

For the moment, the answer is probably "yes." There is nothing about the software itself that depends on Rocks. However configuring it for a particular installation is tricky. For Eucalyptus 1.0, we have written a set of Rocks scripts (that are included in the roll) that essentially query the local installation and generate the necessary configuration files for Eucalyptus. In some cases, the configuration files and where Eucalyptus puts persistent state it needs must be synchronized. By invoking the various scripts, Rocks handles the configuration automatically.

However, it can be done by hand and we are planning a stand-alone installation system in the near future. If you would like to use Eucalyptus and Rocks is not an option for you, please contact us and we'll try to help.

When will the source code be available?

We have not released the code of version 1.0 because building and deploying this distributed system from source is too error-prone in its current state. That is likely to frustrate our early adopters, which we would like to avoid. A binary-only Rocks-based install eliminates many of the potential pitfalls, which is why we felt comfortable releasing 1.0 in that form.

But we are committed to making the deployment process more robust, cleaning up the code a bit, and publishing it soon, hopefully by early July 2008. Whether or not we will support "build from source and deploy" installation method, the code **will** be open. Stay tuned for the announcement on the mailing list.

Eucalyptus Installation Quick Start (1.0)

Assuming you have

- a cluster with Rocks V (including **xen** and **java** rolls),
- a few machines that you are willing to dedicate to Eucalyptus, and
- familiarity with Rocks commands,

installation of Eucalyptus, proceeds roughly as follows:

1. Add Eucalyptus to Rocks

Download Eucalyptus rolls : we have both 32-bit and 64-bit versions. Then:

```
# rocks add roll clean=1 /path/to/eucalyptus-5.0-0.*.disk1.iso
# rocks enable roll eucalyptus
# cd /home/install && rocks-dist dist
```

For more information try section 2.2 of Administrator's Guide

2. Install Eucalyptus on nodes

- **OPTION A:** Add new nodes (choose 'VM Container' and wait for nodes to pop up):

```
# insert-ethers
```

- **OPTION B:** Re-target existing nodes:

```
# rocks set host pxeboot vm-container-0-0 action=install
# rocks set host pxeboot vm-container-0-1 action=install
...
# ssh vc0-0 reboot
# ssh vc0-1 reboot
...
```

For more information try section 2.3 of Administrator's Guide

3. Install Eucalyptus on the front-end

Once the nodes have rebooted, do:

```
# kroll eucalyptus | sh
```

For more information try section 2.4 of Administrator's Guide

4. Configure Eucalyptus

Install at least one runnable VM image that users can select:

```
$ tar zxvf euca-ttylinux.tgz
$ cd ttylinux
# /opt/eucalyptus-1.0/usr/sbin/euca add_image \
  --disk-image ttylinux.img \
  --kernel-image vmlinuz-2.6.16.33-xen \
  --image-name ttylinux
```

Last, but not least, log in to the Eucalyptus Web-site (login: **admin**, password: **admin**) to set the administrator's email:

https://your.front.end.hostname:8443/

That is it!

For more information try section 2.5 of Administrator's Guide

What next?

At this point you'd probably want to see if it works. As an administrator, you can use the system exactly as all other users would. So, go ahead and download your certificates and use them with EC2 command-line tools as described in the User's Guide.

If it works for you, tell others to sign up for an account using the "Apply" link on the front page of the Web site. When they do so, you will receive email messages asking you to either accept or reject the request. More about this in section 4 of Administrator's Guide.

Eucalyptus Administrator's Guide (1.0)

This is a feature-limited binary-only beta release that relies on the Rocks cluster management platform for deploying Eucalyptus on a cluster. In the future, we will additionally offer a source release along with other methods of deployment. The release is feature limited in the sense that some fundamental functionality, namely removing users and removing virtual machine images, is not supported. (This will be fixed in the next update.)

If you are familiar with Rocks and you don't care for superfluous explanations, you

may want to start with the Quick Start page instead of this Guide.

Table of contents

1. Prerequisites

Eucalyptus 1.0 minimally requires a Rocks V front-end system that has been configured to include the **xen** and **java** Rocks rolls. It is not necessary to have any compute nodes configured, but if compute nodes are configured they can be re-targeted as Eucalyptus nodes.

- Follow the instructions in Rocks User's Guide to install it on your cluster. (Keep in mind that managing cluster nodes using Rocks requires full re-installation of software on them.)
- Xen 3.0.2 is the only virtual machine platform currently supported. In Rocks, make sure the **xen** roll, which is a standard component of Rocks V installation, is included on the compute nodes.
- Java 5 (or greater) is needed by the Eucalyptus components running on the front end. Including the **java** roll, which is a standard component of Rocks V installation, should accomplish that.
- Eucalyptus *requires* the unlimited-strength policy files for JCE for the Java Virtual Machine. *The current installer installs them automatically.* Thus, if you are in a country where this is illegal, DO NOT install Eucalyptus. If you are unsure of your country's cryptography import restrictions, Sun's documentation may be able to answer that question: see "Other Downloads" at the bottom of
  - http://java.sun.com/javase/downloads/index_jdk5.jsp (for JDK 1.5) or
  - http://java.sun.com/javase/downloads/index.jsp (for JDK 1.6).
- You *must* be **root** to install and run Eucalyptus. This document assumes that all commands will be executed as root.

2. Installation

2.1 Download Eucalyptus roll

Download Eucalyptus rolls : we have both 32-bit and 64-bit versions. Place that Eucalyptus Rocks ISO image on the front end. Those who prefer the command line can do:

```
$ wget http://eucalyptus.cs.ucsb.edu/releases/eucalyptus-5.0-0.x86_64.disk1.iso

# OR

$ wget http://eucalyptus.cs.ucsb.edu/releases/eucalyptus-5.0-0.i386.disk1.iso
```

Note: This ISO image is not very useful without a Rocks installation with **xen** and **java** rolls. Burning it onto a CD is not necessary.

2.2 Add Eucalyptus to Rocks

Install the Eucalyptus Roll and enable it for deployment on the cluster boot nodes.

```
$ rocks add roll clean=1 /path/to/eucalyptus-5.0-0.*.disk1.iso
#(warning: don't use a ../ path!)

$ rocks enable roll eucalyptus
$ cd /home/install && rocks-dist dist
$ rocks list roll
```

The output of the last command should include a line indicating that the eucalyptus roll is installed and enabled. It should also list "java" and "xen" rolls.

2.3 Install Eucalyptus on nodes

With the rocks roll now installed and enabled on the front-end, we can now start booting or rebooting nodes and instructing the system to rebuild them with eucalyptus installed and running, using standard "rocks" tools.

- **OPTION 1:** If you don't have all the compute nodes that you would like to use with Eucalyptus, run the following command on the front-end to add more of them:

```
$ insert-ethers
```

  At the prompt, select 'VM Container' and wait for the next screen. At this point, the front-end is waiting for nodes to boot. Boot your nodes, and wait for their MAC addresses to appear in the insert-ethers window.

- **OPTION 2:** If you have all the nodes that you want to use with Eucalyptus already registered in your system, perform the following command to re-target them for use with Eucalyptus:

```
$ rocks set host pxeboot vm-container-0-0 action=install
$ rocks set host pxeboot vm-container-0-1 action=install
#... (repeat this for each node you want to use with Eucalyptus)
```

  When done, reboot your nodes (e.g., by logging in with ssh into them and running "reboot").

```
$ ssh vc0-0 reboot
$ ssh vc0-1 reboot
#... (repeat this for each node you want to use with Eucalyptus)
```

With either option, the Eucalyptus Roll is being installed on your nodes as they boot. When the installation process is complete, your nodes will reboot (for the second time). At this point, the nodes are fully configured with the Eucalyptus node controller software running.

2.4 Install Eucalyptus on the front-end

As soon as all nodes have rebooted for the first time, you may proceed with front-end software installation on the Rocks front end (otherwise, you'll have to add the nodes "manually" later):

```
$ kroll eucalyptus | sh
```

If you get problems with ssh and rsync connecting to compute nodes, then you ran the command too early. That is OK, but you will have to propagate the keys to those nodes manually:

```
$ /opt/eucalyptus-1.0/usr/sbin/euca_sync_key
```

If this command also prints out connection errors, the nodes have not all rebooted, yet. Try again in a bit. (Incidentally, if you want to add more nodes into the Eucalyptus cloud later, you would have to add their host names to the `NODES=` in `/etc/default/eucalyptus` and re-run `euca_sync_key`.)

When this is complete (or when you have enough nodes that are working), the Eucalyptus system is fully installed and is ready to use once some first-time administrative tasks are complete.

2.5 Finalizing Eucalyptus Installation

To finalize the installation of Eucalyptus the administrator must login as `admin` and configure the system and (optionally) register an image.

2.5.1 Login as `admin`

Go to the following location (note the **https** prefix):

https://your.front.end.hostname:8443/

Your browser will flag the Web site as one using an untrusted (self-signed) certificate. Accept it.

You will be presented with a login screen, at which point you can log in using the username **admin** and password **admin**. The first time you log in, the system will

require you to change your password and enter a valid administrator email address. This address will be used whenever a new user requests an account on your Eucalyptus system.

2.5.2 Registering the test image (optional)

Please see the section Image Management.

2.6 Installation Complete

Now that you've finished the installation, the system is ready for users to sign up and start using your cloud. See the section on User Signup below for more information about that.

As administrator, you may interact with the system in precisely the same manner as your users, with the exception that your keys will allow you to inspect and terminate all instances running on the system, regardless of which user owns them. Next, you would probably want to generate keys and try using the system.

3. Configuration

3.1 Adding/Removing Nodes

Once you have a running Eucalyptus system you can add/remove vm-containers by editing `/etc/default/eucalyptus` and adding/removing your vm-container's hostname to/from the list of `NODES`:

```
# The list of Node Controllers the cluster controller will be in charge
# of. For example on the rocks cluster you can run rocks list host to
# find out the list of machines available to you (in our case we are
# interested in the VM Container kind).
NODES=" vm-container-0-0 vm-container-0-1"
```

Whenever you add new nodes into the system, propagate cryptographic keys to them as follows:

```
$ /opt/eucalyptus-1.0/usr/sbin/euca_sync_key
```

4. User Management

4.1 User sign-up

Users interested in joining the cloud should be directed to the front-end Web page (note the **https** prefix):

https://your.front.end.hostname:8443/

As soon as the administrator logs in for the first time and enters the email address to be used for application requests, thus activating the Web site for use by others, the login box of the Web site will have an "Apply for account" link underneath it. After a user fills out the application form, an email is sent to the administrator, containing two URLs, one for accepting and one for rejecting the user.

Note that there is no authentication performed on the people who fill out the form. It is up to the administrator to perform this authentication! The only "guarantee" the administrator has is that the account will not be active unless the person who requested the account (and, hence, knows the password) can read email at the submitted address. Therefore, if the administrator is willing to give the account to the person behind the email address, it is safe to approve the account. Otherwise, the administrator may use the additional information submitted (such as the telephone number, project PI, etc.) to make the decision.

Accepting or rejecting a signup request causes an email message to be sent to the user who made the request. In the case of an acceptance notification, the user will see a link for activating the account. Before activating the account, the user will have to log in with the username and password that they chose at signup.

Caveat: currently, we do not support removal of users from Eucalyptus :-) The next minor release will fix that.

5. Image Management

In order to use Eucalyptus, Xen images must be registered with the system. We have provided a small, simple version of linux that you can use to test. This section walks through the steps of registering the euca-ttylinux image with Eucalyptus.

5.1 Download the image

Download the image and place it on the front-end:

```
$ cd
$ wget http://eucalyptus.cs.ucsb.edu/releases/euca-ttylinux.tgz
```

Untar the image, and cd to the directory 'ttylinux'.

```
$ tar zxvf euca-ttylinux.tgz
ttylinux/
ttylinux/vmlinuz-2.6.16.33-xen
ttylinux/ttylinux.img
$ cd ttylinux
```

The two files are the Linux kernel and the root file system image, respectively. The default root password for the image is `root`. If you want to change or disable it, you need to modify the password file inside the root file system (by mounting it and

editing the files).

5.2 Register the Image

Provided with the Eucalyptus installation is the tool `euca` which is used to register images. To register an image you need to know the following information:

- Filesystem Image: used as the root filesystem.
- Kernel Image: kernel used to boot the system.
- Ramdisk Image: ramdisk supplied to the kernel at boot time. *(optional)*

Here is how to invoke `euca` to register the test image with Eucalyptus:

```
$ /opt/eucalyptus-1.0/usr/sbin/euca add_image \
  --disk-image ttylinux.img \
  --kernel-image vmlinuz-2.6.16.33-xen \
  --image-name ttylinux
# replace ttylinux.img with your disk image
# replace vmlinuz-2.6.16.33-xen with your kernel image
# replace ttylinux with the name of your choosing
```

This makes a copy of the image, so the files in `ttylinux` directory may now be deleted.

At this point, you may add addition Xen images that you may have. The 'euca' command supports installation of a ramdisk if your image requires one (see the 'euca --man' output for more information).

Caveat: currently, we do not support removal of images from Eucalyptus :-) The next minor release will fix that.

Eucalyptus User's Guide (1.0

This guide is meant for people interested in using an existing installation of Eucalyptus. (If you have a cluster that you would like to install Eucalyptus on, then take a look at the Administrator's Guide first.)

We will guide you through getting access to a Eucalyptus-based cloud, as well as installing and using tools for controlling virtual instances. Those familiar with Amazon's EC2 system will find most of these instructions familiar because Eucalyptus can be used with EC2's command-line tools.

1. Sign up

Load in your browser the Web page of the Eucalyptus cloud installation that you would like to use. Ask your system administrator for the URL if you don't know it. (The URL will be of the form *https://your.cloud.server:8443/*, where *your.cloud.server* is

likely to be the front-end of the cluster.)

Click the "Apply" link and fill out the form presented to you. You may not be able to use the system until the (human) administrator receives the notification of your application and approves it. The more information you supply the easier it may be for the administrator to make the decision.

Load the confirmation URL that you receive in the approval email message from the cloud administrator. **Log in** to the system with the login and password that you chose when filling out the application form.

2. Generate keys

Once you have logged in, you will see the 'Generate Certificate' button under the 'Credentials' tab. Generating a certificate for your account is necessary before you can use Amazon's EC2 command-line tools for querying and controlling Eucalyptus instances. Currently, the Web interface to Eucalyptus is limited and, hence, the use of command-line tools is practically inevitable.

Click the button to generate the certificate and save it. You can keep these keys in a secure place on any host. The following command-line instructions apply to any Unix-flavored machine with bash (not necessarily the cluster where Eucalyptus was installed). (See Amazon's Getting Started Guide for the similar instructions to use under Windows.)

Unzip the keys using the following command and **protect** them from exposure. The zip-file contains two files with the .pem extension; these are your public and private keys.

```
$ unzip name-of-the-key-zip.zip
$ mkdir ~/.euca
$ mv euca2-*.pem ~/.euca
$ chmod 0700 ~/.euca
$ chmod 0600 ~/.euca/*.pem
```

3. Install EC2 command-line tools

Download the EC2 command-line tools from Amazon.

```
$ wget http://s3.amazonaws.com/ec2-downloads/ec2-api-tools.zip
$ unzip ec2-api-tools.zip
```

Set the following environment variables (note that the base URL is different from the Web site's URL, both in terms of transport protocol - http instead of https - and the port number):

```
$ export EC2_HOME=/path/to/installed/ec2-commandline-tools
$ export PATH=$PATH:$EC2_HOME/bin
$ export EC2_PRIVATE_KEY=~/.euca/euca2-*-pk.pem
$ export EC2_CERT=~/.euca/euca2-*-cert.pem
$ export EC2_URL=http://your.cloud.server:8773/services/Eucalyptus
```

Now you *should* be ready to start using the tools. To test if the tools work (and if the cloud server is running properly), execute the following EC2 command:

```
$ ec2-describe-availability-zones
```

In the output of the above command, you should see the cluster's front-end hostname displayed along with the status of 'UP' and a short description of how many 'small' instance types the cloud can execute (002/002 means 2 available out of 2 total). If the response is different or if the number of available instances is 000/000, then contact your cloud administrator about the problem.

4. Quick Start

Now you can begin running VM instances on the Eucalyptus cloud. Using the EC2 command-line tools, you can learn about installed images, start VM instances using those images, describe the running instances, and terminate them when you're finished with them.

The following EC2 commands will allow you to query the system:

```
$ ec2-describe-images
IMAGE <emi-id> ...

$ ec2-describe-instances
(will be empty until you start an instance, as shown below)

$ ec2-describe-availability-zones

$ ec2-describe-keypairs
(will be empty until you add key pairs, as shown below)
```

Before starting a VM, you need to create at least one key pair. This key pair will be injected into the VM, allowing you to SSH into the instance. Below we will use *mykey* as a handle, but you may choose any string you like instead:

```
$ ec2-add-keypair mykey
KEYPAIR keyname ...
-----BEGIN RSA PRIVATE KEY-----
```

```
<key contents>
-----END RSA PRIVATE KEY-----

$ cat >mykey.private
(paste the key contents, including BEGIN and END lines, printed out by the previous command)

$ chmod 0600 mykey.private

$ ec2-run-instances <emi-id> -k mykey -n <number of instances to start>

$ ec2-describe-instances
(should now show the instance)
```

Once the instance is shown as 'Running', it will also show two IP addresses assigned to it. You may log into it with the SSH key that you created:

```
$ ssh -i mykey.private root@one-of-the-ip-addresses
```

To terminate instances, use:

```
$ ec2-terminate-instances <emi-id1> <emi-id2> ... <emi-idn>
```

Please, see Amazon's EC2 Getting Started Guide for more information about these command-line tools. Keep in mind that not all tools are currently supported.