The Definitive Step-by-Step Guide for Installing ColdFusion 8 on Ubuntu 7.10 using Amazon EC2 from WindowsXP or Vista

Original: April 15, 2008; Updated: September 16, 2008
by David Smith (http://awsconsole.riaforge.org)

Prerequisites:

- Establish an EC2/S3 account. http://aws.amazon.com/ec2
- Download the install the EC2 and S3 plugins for Firefox:
  http://developer.amazonwebservices.com/connect/entry.jspa?externalID=609
  http://developer.amazonwebservices.com/connect/entry.jspa?externalID=771
- Download and install PuTTY and PuTTYgen:
  http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html
- Download and install WinSCP (choose the "Explorer-like" interface at setup):
  http://winscp.net/eng/download.php
- Download and install TightVNC for Windows:
  http://www.tightvnc.com/download.html

Goals:

- Configure the Firefox EC2 interface (Elasticfox) to stop/start instances and configure ports
- Generate a keypair, and enable command-line access via Putty and WinSCP
- Build, save and register your own Amazon Machine Image (AMI) consisting of ColdFusion 8, Apache2 and MySQL 5 for running on an Ubuntu OS with GUI desktop installed

Setup Credentials and Start Your Base Instance:

1. **Start this base Ubuntu 7.10 instance:**
   http://developer.amazonwebservices.com/connect/entry.jspa?externalID=1065&categoryID=101
   1.1. Open Firefox, install and/or navigate to Elasticfox under the Tools menu
   1.2. Setup your credentials:
      1.2.1. Account name = your username on the forums where it says "Welcome, [username]" after you login. It is *not* your "Name" or your "UserID".
      1.2.2. AWS Access key = "Your Access Key ID"
      1.2.3. AWS Secret key = "Your Secret Access Key" (note: when copying and pasting this key the AWS site may add a blank space to the very end, which you cannot see if you just copy and paste directly into Elasticfox. Instead, paste it to Notepad, strip the extra space, then copy and paste that into Elasticfox.)
      1.2.4. Click the "Add" button to save your profile credentials, then close the dialog
      1.2.5. Hit the blue refresh button to bring up the list of public AMIs
      1.2.6. Now click the "keypairs" tab and create a new keypair. You will need this to interact with your instance via Putty and WinSCP.
      1.2.7. Create a keypair name (arbitrary) and save it to your hard drive wherever you are saving your other certificate files generated for you when you first established your EC2 account (just to keep everything in one place). You'll convert your keypair file to a Putty-compatible version in Step 2.

1.2.8. Now go back to the list of public AMIs and find the Ubuntu 7.10 instance that matches the description and/or AMI ID from the link above and click the green launch button to start it
1.2.9. When launching a new instance, you can leave all the default alone if it is your intention to start a 32-bit "small" instance, EXCEPT you must select your keypair from the drop down list. Click "Launch" when done with this dialog.
1.2.10. Your instance will now boot up and you'll be provided with your DNS in the lower window. You may have to hit the blue Refresh button a couple times until it shows "running". Note, you can right-click on top of any of your instance data to copy them to clipboard (handy!).
1.2.11. You must now open up port 22 and 80 on the public DNS by clicking the Security Groups tab, clicking the green Grant Permissions button in the lower frame and entering "from port 80 to 80" in the Add Permission dialog box. Click Add. Then do it again for port 22 to 22, click Add again. All of this can be found in the EC2 documentation here http://docs.amazonwebservices.com/AWSEC2/2007-08-29/DeveloperGuide/ami-from-existing-image.html

2. Convert your previously-saved keypair file to a Putty-compatible version to can gain command-line access:

2.1. Start PuTTYgen
2.2. Click load, and browse to your keypair file saved in Step 1.2.7. By default PuTTYgen will look for a files with a .ppk file extension (which yours doesn't have yet), so set the filetype to "all" when browsing.
2.3. Click "Save private key file", and click "Yes" when asked if you want to save it without a password. Save it wherever you want.

3. Start Putty and gain command-line access to your instance; set a root password

3.1. Start Putty
3.2. Paste your instance's public DNS into the Host Name box. Save the session info if you want.
3.3. In Putty, navigate to Connection > SSH > Auth and browse for your private key saved in 2.2.3.
3.4. Click Open to create a console session. Note: the first time you open this ssh connection in Putty you may get a message about the server host key is not cached in the registry. Just ignore and hit "Yes". You will get this message any time you ssh into a new instance.
3.5. If everything went smoothly up to this point you will be asked to login; enter "root" when prompted. You now have command line access to your Ubuntu instance with full admin rights.
3.6. You should set a root password at this point. To do that type "sudo passwd" at the command prompt and enter a password. You can type "sudo passwd" any time to change it.
3.7. Leave the session running (minimized) while you move on to the next step.

4. Start WinSCP to gain file control to your instance (equivalent of Windows Explorer):

4.1. Start WinSCP; use the "Explorer-like" interface (under Preferences if you need to reset it)
4.2. Click the "New" button if required, or else simply enter your EC2 public DNS as host name.
4.3. Enter "root" as username and whatever password you created in step 3.7.
4.4. Navigate to your *.ppk private key file from step 2.3.
4.5. Click "Save", then "Login".
4.6. Click "Yes" when the warning box pops up.
4.7. You can now see your instance's file structure. You can drag and drop files here, edit files, etc.
4.8. You can keep this open or close it. You will need it again later on for step 15 and possibly others.

Build the Machine!

Summary of actions you'll perform against the base Ubuntu image (32-bit):

- Install the Ubuntu desktop GUI
- Install Apache2
- Install MySQL server
- Install ColdFusion 8 Enterprise trial version (not developer edition)
- Update the Apache default file to "index.cfm"
- Install a start-up script to auto-boot ColdFusion 8 (a fix for this is coming out in 8.1)
- Create a MySQL datasource under ColdFusion
- Add an Apache connector to ColdFusion
- Upload an index.cfm file to create a database table and insert test records
- Create a J2EE EAR file to allow future ColdFusion deployment options, e.g., setting up multiple instances and clustering for load-balance and fail-over
- Bundle and upload the image as your own so you can terminate and use it later
- Register the image (will be private)

NOTE: items highlighted are the exact text you enter at the command line. To save time you can copy and paste rather than re-type. Since "Ctrl-V" doesn't work in console mode, you have to right-click to paste.

5. **Go back to Putty and install the Ubuntu desktop.** ***This may take 10+ minutes on a small instance.***
   5.1. sudo apt-get update
   5.2. sudo apt-get install x-window-system-core xserver-xorg gnome-desktop-environment

   Note: for the purposes of development, it is useful to have the full desktop on this Ubuntu server for several reasons: 1) when using up the ColdFusion web server config tool you can call it up in a GUI from the desktop rather than doing it all via command line; 2) there are some performance monitoring utilities in Ubuntu that might be useful to watch; 3) you can run Webmin for GUI access to Apache; 4) Sometimes it is just useful to "see" what's going on. In theory, however, a server should be as streamlined and free of unnecessary software as possible due to resource hogging and potential security holes.

   5.3. Choose 1600x1200 resolution or whatever you prefer (select with spacebar, then tab to OK).

6. **Install the VNC remote desktop server on Ubuntu**
   6.1. sudo apt-get install tightvncserver
   6.2. vncserver (starts the remote access server so you can establish a client connection)
   6.3. create a password
   6.4. Further down we will connect to the desktop, but for now, let's continue installing software

7. **Install Apache web server**
   7.1. sudo apt-get install apache2

8. **Install MySQL database server. This will take a minute.**
   8.1. sudo apt-get install mysql-server
   8.2. Create a password for MySQL

9. ***Optional: install Webmin GUI for Apache control:***
   - *9.1. wget http://garr.dl.sourceforge.net/sourceforge/webadmin/webmin_1.380_all.deb*
   - *9.2. sudo apt-get install libnet-ssleay-perl libauthen-pam-perl libio-pty-perl libmd5-perl*
   - *9.3. sudo dpkg -i webmin_1.380_all.deb*
   - *9.4. VCN to Ubuntu desktop, open Firefox and type the URL: https://[ec2 internal dns]:10000/ You can use your external address instead if you want so long as you open port 10000 through the EC2UI security group (see Step 1.2.11)*
   - *9.5. Log in to URL above using username "root" with password established in step 3.7.*

10. Configure Putty for VNC access, and use TightVNC Viewer to access your Ubuntu desktop

10.1. Close your current Putty console session and start up Putty again. Load or re-enter your EC2 public DNS and private key file, then navigate in Putty to Connection > SSH > Tunnels.

10.2. In Source port, put 5901; in Destination, put your instance DNS name followed by :5901, for example : *ec2-XXX-XXX-XXX-XXX.z-2.compute-1.amazonaws.com:5901*

10.3. Click the Add button for Putty to remember this setting. (At this point it is probably a good time to save all this info as a particular Putty profile you can load it any time without re-entering.)

10.4. Click "Open" to start the tunnel connection.

10.5. Login as "root". You can also hard-code "root" into Putty by going to Connection > Data and entering it at the top next to "auto log-in username". Saves time.

10.6. Leave this session window open/minimized and move on to Step 11.

11. Start the TightVNC Viewer program you downloaded or installed earlier

11.1. Next to "VNC Server" input "localhost:1" then hit Connect

11.2. When prompted enter the root password from step 3.7

11.3. If successful, a new window will open. When you see a command line prompt inside a white terminal window (this is on Ubuntu now) type "gnome-session".

11.4. Click "Continue" when asked if you want to run as a privileged user, and ignore all the subsequent warning messages and errors. When it appears everything has finished loading you are now ready to interact with the desktop. Minimize the white terminal window while working, do NOT close it or you will kill your remote session.

11.5. Note, if for some reason you get stuck in a full-screen mode, to exit full-screen type Ctrl-Alt-Shift-F or Ctrl-Esc Esc or right click on VNCviewer taskbar icon to see a menu.

11.6. Note, to shut down VNC type "vncserver -kill :1" from the Putty command line. To restart type "vncserver". Note there is a blank space after "kill".

12. **There is a problem with VNC's remote keyboard mapping by default. To correct it (so you can type things** into Ubuntu's desktop) do this from your Ubuntu desktop:

12.1. Right click on Applications (upper left-hand corner of the GUI) > click Edit Menus and enable "Configuration Editor" under "System Tools" and click Close. Now you can do everything with mouse clicks since the keyboard doesn't work yet.

12.2. Go to go Applications > System tools > Config editor

12.3. Go to desktop > gnome > peripherals > keyboard > kbd

12.4. Find the key name "layouts" and double click it. This will bring up an "Edit Key" window.

12.5. Click on add, then type "u" and "s" (will translate as "m" and "b") and click OK > OK again.

12.6. Close out of gconf editor, and then kill and restart your vnc session (see 13.7). You should now see the expected characters when you type anything. To test, open Firefox on Ubuntu and try to go somewhere.

13. **Download the ColdFusion 8 trial edition** for Linux if you have not already (not developer edition, or you won't be able to experiment with multi-server config): http://www.adobe.com/products/coldfusion/ .

    NOTE: you could save the download via your Ubuntu remote desktop connection, since it will need to live on your instance at some point. The only real benefit is that it would save you from having to upload a 300+ Mb file from your personal machine to your Ubuntu instance. If you think you'll use the file again, then save it up to S3 since data transfer between S3 and EC2 is extremely fast and free of bandwidth charges.

14. **Transfer the ColdFusion install file.** Assuming you store the file (ColdFusion-8-lin.bin) on S3, then you would transfer it to whatever folder on Ubuntu, for example "/root/downloads/", by running this command at a Putty console prompt (note, the folder "/downloads/" will be created automatically):
    14.1. Temporarily enable public access to the file on S3 using your S3 plugin for Firefox or the next step will fail (right-click the file and edit the ACL).
    14.2. wget -P /root/downloads/ [your bucket].s3.amazonaws.com/ColdFusion-8-lin.bin
    14.3. Note: [your bucket] is the name of the folder you created in the S3fox plugin to hold your ColdFusion file. You can right-click on the folder/bucket name to copy the URL into clipboard.

15. **Change the permission to allow execute of the following ColdFusion file.**
    15.1. Do this in WinSCP by right-clicking the file > Properties > checking all the "x" boxes to allow execution.

16. **Run the ColdFusion installer! Simply type the path below:**
    16.1. /root/downloads/ColdFusion-8-lin.bin

17. **Configure the ColdFusion install:**
    17.1. Select trial edition
    17.2. Select multi-server
    17.3. Don't install sub-components
    17.4. Blank serial number ok
    17.5. Use the built-in web server; do not associate with Apache yet
    17.6. Enter a password for CF admin
    17.7. Enable RDS and select a password
    17.8. Start ColdFusion when prompted
        17.8.1. cd /opt/jrun4/bin
        17.8.2. ./jrun -start cfusion

        NOTE: At this point, you will *not* be returned to a normal command line after running that, so you'll have to open a second terminal to continue. Don't kill the first terminal because it will stop CF. Until this bug fix comes out in CF 8.1 you'll have to create an auto-boot script (see below) first.

18. **Open TCP port 8300** otherwise you can only access port 8300 from the local Ubuntu desktop via VNC.
    18.1. Open your Firefox EC2UI and click on the Security Groups tab group.
    18.2. As in step 1.2.11, give permission to port 8300

19. **Finish the install** by going to the CF Admin URL when prompted:
    19.1. http://[ec2 dns]:8300/cfide/administrator/

20. **Edit /etc/apache2/mods-available/dir.conf** to include "index.cfm". Do this using WinSCP.

21. **Create a ColdFusion auto-boot script or else when you kill the current Putty session it will also kill ColdFusion** (CF 8.1 was supposed to fix this but didn't):
    21.1. Using WinSCP go to to /etc/init.d
    21.2. Create a new file called "coldfusion" with no file extension by going to File > New > File.
    21.3. Paste these parameters then Save the file:

```
#!/bin/bash
case "$1" in
  start)
    /opt/jrun4/bin/jrun -start cfusion
    ;;
  stop)
    /opt/jrun4/bin/jrun -stop cfusion
    ;;
esac
```

22. **Change the permissions to allow execute** on the file you just saved or else it won't boot!
    22.1. Right click the "coldfusion" file you just created and go to properties.
    22.2. Check off all the boxes under the "X" (for execute) column.

23. **Run the following command to update the Ubuntu bootstrap:**
    23.1. update-rc.d coldfusion defaults 99 (spot 99 makes sure everything else boots first)

24. **Now reboot Ubuntu** to see if everything comes up; make take a few moments so be patient
    24.1. reboot

25. ***Optional: delete the apache default web directory to keep things clean*** *(I have no idea why it's there)*
    25.1. Reconnect to your instance with WinSCP
    25.2. Navigate to /var/www/ (this is your web root, like Inetpub is on IIS)
    25.3. Right click on the apache2-default folder and delete it

26. **Create a test MySQL database**
    26.1. In step 8 you already installed MySQL. Now configure it.
    26.2. Open a putty command prompt. TIP: You can do this from inside WinSCP. Look for the Putty icon up at the top (note: you cannot create your VNC desktop tunnel this way.)
    26.3. Type mysql -u root -p then enter the password you created in step 8.2
    26.4. CREATE DATABASE database1;

27. **Create this datasource in ColdFusion now**
    27.1. Use MySQL version 4/5 in the CF admin to create a datasource called "MySQL"
    27.2. Database name is "database1" (or whatever you just created above)
    27.3. Username is "root" and password is whatever you created in step 8.2
    27.4. Note: there is no "server name". Leave it blank.

28. **Set up the Apache web connector** so ColdFusion works under the Apache root at /var/www/.
    28.1. Log into your Ubuntu desktop again with TightVNC viewer as in step 11.
    28.2. Once on the Ubuntu desktop, go to Applications > Accessories > Terminal
    28.3. Type cd /opt/jrun4/bin

28.4. Type ./wsconfig (not a typo, you must type the "." and "/" to execute the connector)
28.5. When the GUI pops up, click "Add"
28.6. Enter the required information:
  28.6.1. Leave Jrun Host as "localhost", and Jrun server as "cfusion"
  28.6.2. Select "Apache" web server obviously
  28.6.3. Configuration directory is "/etc/apache2"
  28.6.4. Hit the Advanced… button
  28.6.5. Server binary is located at "/usr/sbin/apache2"
  28.6.6. Control scripts are at "/usr/sbin/apache2ctl"
  28.6.7. Hit OK, then Ok again to exit
  28.6.8. Hit Yes when prompted to restart the web service
  28.6.9. Done.

29. **Now you can create a "hello world" index.cfm file** and save it to /var/www/
  29.1. Create the file locally and use WinSCP to drag and drop it into /var/www/
  29.2. Run the file by going to http://[ec2 dns] or http://[ec2 dns]/index.cfm (should work either way if Apache is properly configured

30. ***Optional: create an RDS connection to EC2*** *in Homesite, Dreamweaver, or Eclipse to modify the index.cfm file to create the database tables, save, and run it. This step assumes you know how to set up RDS on your editor. You can, of course, do this with notepad or WinSCP's built-in editor, but the point here is to learn how to connect to RDS since most ColdFusion developers will want to do this.*

31. **Write ColdFusion code to create and update the database.**

```
<cfif isdefined("submit")>
<CFQUERY datasource="MySQL" name="test">
INSERT INTO example (name, age)
VALUES ("#name#", #age#)
</CFQUERY>
</cfif>

<cfset thisServer = createObject( "java", "jrunx.kernel.JRun" ).getServerName()>

<cfset request.greeting = "Hello from Coldfusion 8 Enterprise" />
<cfoutput>#request.greeting# on #thisServer#.  If you see this message then CF, Apache and MySQL
are all running properly.</cfoutput>


<!--- <CFQUERY datasource="MySQL">
CREATE TABLE example(
id      INT NOT NULL AUTO_INCREMENT, PRIMARY KEY(id),
name    VARCHAR(30) NULL,
age     INT NULL,
created         TIMESTAMP DEFAULT NOW()
)
</CFQUERY>

<p>Done!</p>  --->

<H2>MySQL Insert & Select Test</H2>

<form action="index.cfm">
Name: <input type="text" name="name" size="30"><br>
Age: <input type="text" name="age" size="5"><br><br>
<input type="submit" name="submit"> <a href="index.cfm">Reset</a>
</form>

<CFQUERY datasource="MySQL" name="test">
SELECT * FROM example
ORDER BY ID
</CFQUERY>

<cfoutput>
```

```
<cfdump var="#test#">
</cfoutput>
```

31.1. Make sure to comment out the table creation code after the first run.

31.2. Now enter one piece of test data in the form, like name "Joe" age "34" and submit. When we bundle this image, we will have a working record to confirm everything is running when the server boots up.

32. ***Optional: go to CF Admin and create a J2EE archive file*** *(under Packaging & Deployment) that contains the MySQL datasource and the index.cfm file.*

32.1. Give it an Archive Name of "testSite1" and hit Add

32.2. Specify the application directory. In our case it's "/var/www"

32.3. Specify the distribution directory where the EAR file will sit. I can be anywhere, but

32.4. Select the EAR radio button

32.5. Archive the MySQL datasource

32.6. Hit Submit. It may take several moments.

NOTE: No other choices are required, but definitely investigate this area of the CF Admin. It's powerful. Essentially these packages let you deploy turn-key ColdFusion apps. If you've never understood the need for CF Enterprise, this might change your mind (this, and the power of clustering.) This is also where you can create application images and install them as additional ColdFusion instances under the "Instance Manager" menu (not to be confused with AWS instances). You would want to do this if interested in clustering.

33. **Everything is set! Now do a boot clean before saving your machine image**.

33.1. Type /etc/init.d/bootclean

34. **In order to prepare for bundling, move your cert-*.pem and pk-*.pem keys** into /mnt/. Amazon uses these keys in the command below for proper identification of your bundle manifest.

35. **Now bundle**: (this may take several minutes, and look like it's hanging, but it's not)

ec2-bundle-vol -d /mnt -k /mnt/pk-[fill in your pk info].pem -c /mnt/cert-[fill in your cert info].pem -u [your owner id] -r i386

Note: your "owner id" is your account number in the upper-left of the "**AWS Account Activity**" page on Amazon. Do not use hypens or spaces. I have no idea why they make this so difficult.

36. **Send bundle to S3**:

36.1. You must create a bucket first; recommend using Firefox S3Fox plugin. You'll have to set up your credentials for S3Fox like you did in step 1.2.1.

ec2-upload-bundle -b [your bucket name] -m /mnt/image.manifest.xml -a [your AWS Access Key ID] -s [your Secret Access Key]

37. **Finally, register your image** by opening the Firefox EC2UI, clicking the green "Register AMI" button, and pointing it to your manifest location:

37.1. [your bucket name]/image.manifest.xml

38. **Done!** Now don't forget to terminate your instances when finished for the day/night so that you won't be continually charged the hourly rate. When you boot up new instances of this image, you might see Server 500 errors until all the services have finished starting up. Instead of using EC2UI, try the ColdFusion-based AWS console at http://awsconsole.riaforge.org/