

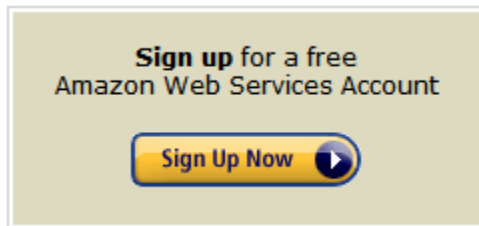
ICAPS IPPC 2011

Amazon EC2 Setup Instructions, v3-9-11

Questions/comments to Scott Sanner (ssanner@gmail.com)

Step 1: Create an Amazon EC2 account

<http://aws.amazon.com/ec2/>



Signing up involves receiving an automated phone call, but this is a trivial step.

You will also need to enter a credit card.

You should **not need to pay for usage as the EC2 credits you will receive should cover up to 98 hours of large instance usage – more than you will need for the competition. Also, a single EC2 micro-instance is free. Your account details can be found here:*

<http://aws.amazon.com/account/>

Step 2: Login to the AWS Management Console (if not already)

<http://aws.amazon.com/console/>

Already have an AWS account?

Sign in to get started.



Amazon EC2


Save this as your default console

**Make sure to select "Amazon EC2" from the drop down list before clicking.*

Step 3: Launch a test instance






<https://console.aws.amazon.com/ec2/>

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

Launch Instance 

Note: Your instances will launch in the US East (Virginia) region.

A. Choose an AMI: This will require you to choose an instance type, for uniformity of machine type for the competition, please use either:

	Basic 64-bit Amazon Linux AMI 2010.11.1 Beta (AMI Id: ami-74f0061d) Amazon Linux AMI Base 2010.11.1, EBS boot, 64-bit architecture with Amazon EC2 AMI Tools. Root Device Size: 8 GB		Select 
	Basic 64-bit Microsoft Windows Server 2008 (AMI Id: ami-d9e40db0) Microsoft Windows 2008 R1 SP2 Datacenter edition and 64-bit architecture. Root Device Size: 30 GB		Select 

The AMI ID may change, but in short you should choose either **Basic 64-bit Amazon Linux** or **Basic 64-bit Windows Server 2008**.

B. Instance Details: This is where you choose the amount of memory. For uniformity in the competition, you are required to use the **Large (m1.large, 7.5GB)** instance type:

Number of Instances: **Availability Zone:**

Instance Type:

Termination Protection: Prevention ation.

Launch Instances

EC2 Instances let you pay for compute capacity by the hour with no long term commitments. This transforms what are commonly large fixed costs into much smaller variable costs.

However, for **free experimentation** you can choose instead **Micro (t1.micro, 613 MB)**:

Instance Type:

For *Advanced Instance Options*, just use the defaults:

Advanced Instance Options

Here you can choose a specific **kernel** or **RAM disk** to use with your instances. You can also choose to enable CloudWatch Detailed Monitoring or enter data that will be available from your instances once they launch.

Kernel ID:

RAM Disk ID:

Monitoring: Enable CloudWatch detailed monitoring for this instance
(additional charges will apply)

User Data:

base64 encoded

It is not necessary to *add tags to your instance*, but you can if you want, e.g.,

Key (127 characters maximum)	Value (255 characters maximum)	Remove
Name	IPPC-Client	

[Add another Tag.](#) (Maximum of 10)

C. Create Key Pair: for secure access to a running instance, you need to create a key pair. Just choose a name you'll remember, click the button, and save the file; you'll need this file later to access your instance via *ssh* or *remote desktop*.

Create a new Key Pair

1. Enter a name for your key pair:* (e.g., jdoekey)

2. Click to create your key pair:*

Save this file in a place you will remember. You can use this key pair to launch other instances in the future or visit the Key Pairs page to create or manage existing ones.

D. Configure Firewall: you will need to access your instance externally via *ssh* and perhaps other tools (like *remote desktop* for Windows via the RDP protocol). So for now you need to open up these ports.

Later you will also need to open up **custom** ports for the **competition server**... this is easy to do in a later configuration window, but for some reason this initial configuration does not permit custom ports, so don't worry about this now.

For now I recommend just opening up the SSH, RDP, for now... maybe also HTTP or HTTPS if you need to access the web from your instance (e.g., installing software on your Windows image). Note that your instance incurs charges for Internet usage, but if your AWS account is new, you get *15Gb transfer in/out free per month for one year*.

Create a new Security Group

1. Name your Security Group:


2. Describe your Security Group:

3. Define allowed Connections

Application	Transport	Port	Source Network (IPv4 CIDR)	Actions
SSH	tcp	22	All Internet	Remove
RDP	TCP	3389	All Internet	Remove
Select...	-	-	All Internet Change	Add Rule

E. Review and Launch Instance:

Please review the information below, then click **Launch**.

AMI:  Amazon Linux AMI ID ami-74f0061d (x86_64)

Name: Basic 64-bit Amazon Linux AMI 2010.11.1 Beta

Description: Amazon Linux AMI Base 2010.11.1, EBS boot, 64-bit architecture with Amazon EC2 AMI Tools.

[Edit AMI](#)

Number of Instances: 1

Availability Zone: No Preference

Instance Type: Micro (t1.micro)

Instance Class: On Demand

Termination Protection: Disabled

[Edit Instance Details](#)

Monitoring: Disabled

Kernel ID: Use Default

RAM Disk ID: Use Default

User Data:

[Edit Advanced Details](#)

Key Pair Name: IPPC-Client

[Edit Key Pair](#)

Security Group(s): IPPC-Client-Security

[Edit Firewall](#)

[< Back](#)

[Launch](#) 

Step 4: Get your software running on the test instance

View your running instances by clicking on *Instances* in the navigation pane



on the main console page

<https://console.aws.amazon.com/ec2/>

On the instances page, click the checkbox for your running instance (in this case I had previously called it IPPC-Client). When you do this, the bottom pane will display information. What's important for making a connection is the **Public DNS** name:

Name	Instance	AMI ID	Root Device	Type	Status	Security Groups	Key Pair Name	Monitoring	
<input checked="" type="checkbox"/>	IPPC-Client	i-4377602f	ami-74f0061d	ebs	t1.micro	running	IPPC-Client-Securi	IPPC-Client	basic

Public DNS:	ec2-50-17-3-58.compute-1.amazonaws.com
Private DNS:	ip-10-117-73-121.ec2.internal
Private IP Address:	10.117.73.121

You connect to your running instance just as you would connect to any remote server. How you do this depends on what machine you are connecting from (Linux/Mac or Windows) and what EC2 instance you are connecting to (Linux or Windows). For simple instructions on each of these cases, please refer to pages 9-19 here:

<http://awsdocs.s3.amazonaws.com/EC2/latest/ec2-gsg.pdf>

Pages 9-16 tell you how to connect to a Linux EC2 instance from various local machines. Pages 17-19 tell you how to connect to a Windows EC2 instance from a local Windows machine. You can probably use *rdesktop* as well if your local machine is Linux.

I have not worked with the Windows system yet, but if you have a remote desktop connection and http/https firewall access on your EC2 instance, installing packages and transferring files should be the same as if it were your local computer.

On Linux, you can copy over your files with *scp*. You may also need to install packages (e.g., *javac* or *gcc* if you want to compile on the EC2); for the Amazon Linux instances, the package manager is *yum*. Here are quick examples of *yum* usage (output in yellow):

```
sudo yum install emacs
```

```
yum whatprovides */javac
```

```
1:java-1.6.0-openjdk-devel-1.6.0.0-44.1.9.1.18.amzn1.x86_64 : OpenJDK Development Environment
Repo : amzn
Matched from:
Filename : /usr/lib/jvm/java-1.6.0-openjdk-1.6.0.0.x86_64/bin/javac
```

```
sudo yum install 1:java-1.6.0-openjdk-devel-1.6.0.0-44.1.9.1.18.amzn1.x86_64
```

```
Running Transaction
  Updating      : 1:java-1.6.0-openjdk-1.6.0.0-44.1.9.1.18.amzn1.x86_64    1/3
  Installing    : 1:java-1.6.0-openjdk-devel-1.6.0.0-44.1.9.1.18.amzn1.x86_64  2/3
  Cleanup      : 1:java-1.6.0-openjdk-1.6.0.0-44.1.9.1.16.amzn1.x86_64    3/3
```

```
Installed:
  java-1.6.0-openjdk-devel.x86_64 1:1.6.0.0-44.1.9.1.18.amzn1
```

```
Dependency Updated:
  java-1.6.0-openjdk.x86_64 1:1.6.0.0-44.1.9.1.18.amzn1
```

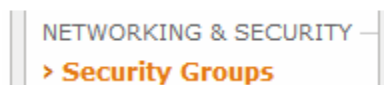
Step 5: Connect your IPPC client to the RDDLSim Server

We will have one server running on a *dedicated port for each competition entrant*. Please ensure that you connect to **your dedicated port**. If you believe the server for your port has crashed, email Scott (ssanner@gmail.com) to reset it.

Group Members	SERVER_PORT
Andrey Kolobov, Peng Dai, Mausam, Dan Weld	2300
Marek Grzes, Jesse Hoey	2301
Gabor Balazs, Szita Istvan, Csaba Szepesvari	2302
Jia-Hong Wu, Rajesh Kalyanam, Bob Givan	2303
Ari Weinstein, Michael Littman	2304
Thomas Keller, Patrick Eyerich	2305
Florent Teichteil, Guillaume Infantes, Vincent Vidal	2306
Tuna Toksoz, Kemal Ure, Josh Redding	2307
Kyle Morrison, Pascal Poupart	2308
Wee Sun Lee, Wu Kegui and group	2309
Kee-Eung Kim, Dongho Kim, Kang-hoon Lee and group	2310
Guy Shani, Diego Maniloff	2311
Marc Toussaint	2312
Shaowei Png, Sylvie Ong, Joelle Pineau	2313
Dan Bryce, Alan Olsen	2314
Eddy Borera, Arisoa Randrianasolo	2315
Saket Joshi, Prasad Tadepalli, Alan Fern	2316

A. Configure Firewall: to connect to the RDDLSim server you need to unblock the appropriate ports in the range 2300-2316.

View your security groups by clicking on *Instances* in the navigation pane



Select the security group assigned to your instance. Then add a *custom* TCP port corresponding to your assigned **SERVER_PORT** in the table above, for example:

Connection Method	Protocol	From Port	To Port	Source (IP or group)	Actions
SSH	tcp	22	22	0.0.0.0/0	Remove
RDP	tcp	3389	3389	0.0.0.0/0	Remove
-	tcp	2323	2323	0.0.0.0/0	Remove
Custom...	TCP	2323	2323	0.0.0.0/0	Save

B. Connect your client to the EC2 RDDLSim server: if you copy over the RDDLSim repository to EC2 (tgz or zip it first!), you should be able to execute the following sample client command (example for Linux):

```
./run rddl.competition.Client files/test_comp/rddl
  SERVER_NAME YourClientName
  rddl.policy.SPerseusSPUDDPolicy SERVER_PORT 123456
  sysadmin_inst_mdp__1
```

Here you need to replace **SERVER_NAME** with the name of the server sent by Scott via email and **SERVER_PORT** by the appropriate port assigned above. If running this is successful, then the Server will log all of your interactions and you are ready to compete!

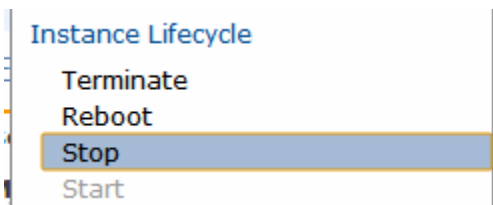
Step 6: Terminate your instance

If you are using anything other than the free Amazon Linux Micro-instance, you need to **stop** or **terminate** your instance to avoid using credits or paying for Amazon EC2 usage when not needed.

View your running instances by clicking on *Instances* in the navigation pane



Now right click on your instance and select either **Stop** or **Terminate**.



If you **Terminate**, **all modifications** to that machine such as package installations and all data on the virtual hard drive **will be lost**. Hence **Stopping** is better if you want to return to that particular instance and your saved data. When **Starting** again, note that the public DNS address you use to access the instance has probably changed.