Amazon Elastic Cloud (EC2): An Easy Route to Your Own Dedicated Web-server

The Amazon Elastic Cloud is a service which allows anyone to create their own dedicated web-servers which Aamazon manage & run. The whole process can be done from a web-browser, and the resulting machines are fully functional servers. You have to set up an account with Amazon, which is straightforward, and involves various steps which I won't detail here.

The simple, great thing about Amazon cloud servers is that they allow you to create "WAN" or internet accessed service and manage them surprisingly easily, once you have understood a few key aspects to how Amazon operate. The only complexity is the horrific barrage of Amazon jargon you have to cope with en-route.

Once your account is set up you can access the AWS Management Console, which is the on-line management tool, it gives total control of all your servers. (https://console.aws.amazon.com/ec2/home).

I would emphasize that I am a long-time solo programmer in the Delphi / Object Oriented space, I am not linked to or employed by Amazon in any way.

Typical Uses for Amazon EC2

- To create a server on a Wide Area Network to host a database accessed by customers from a variety of offices in different locations. The ElevateDB databases are a good example of a database which works best on a dedicated server as many public web-servers are unwilling to open the ports it requires.
- To create a dedicated web-server to host a set of websites which cannot easily be placed in more conventional web-hosting. For example websites with large data-access requirements or websites built with tools like Morfik (www.morfik.com).
- To use as a repository for a Subversion (or other) version control system.
- To create a server for testing purposes, which might only run for a few hours before being shut down.

Key powerful features of the Amazon Cloud:

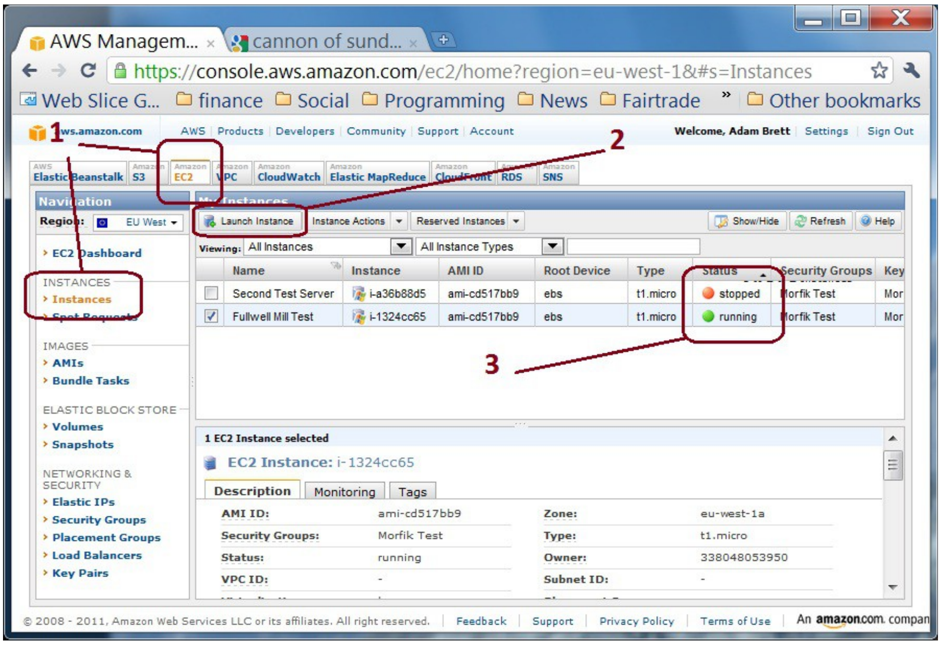
- You can create as many web-servers as you like, with Linux or Windows Operating systems in any / all possible flavours and sizes.
- You can switch these servers on and off from the management console, as well as being able to change various firewall and security permissions from the same place.
- You are only charged per hour **while a server is actually running**, if you switch a server off you are not charged. Full pricing is detailed at aws.amazon.com/ec2/pricing/, and consists of a per hour charge plus a per terabyte charge. There are a number of pricing options (depending on size of server, RAM, processor etc.) but from my perspective costs are very cheap. The server my business uses consistently costs us less then £30 per month.
- If you have a "standard set up" for a server which you need to re-use you can create it once and then save it. Creating a new server identical to this saved server set up is then extremely easy. These saved servers are called "AMI's" (Amazon-Machine-Instances, in the confusing jargon of the Amazon cloud). You can create as many as you want. They are very similar in concept to saved machine-images in VMWare.
- You can allocate yourself IP addresses and then associate these with a server.
- If you need to maintain a server you can switch the IP address to a different server and take the first server off-line for maintenance, and this can be done from within your browser.
- I do not use Amazon cloud for testing, but I imagine it could be incredibly powerful, as it is possible to set up sets of servers and potentially have automated testing systems running on all or many, mimicking the operation of a conventional LAN.
- You can “scale up” and “scale down” machines on demand, to cope with peaks in use. Note that the costs for larger Amazon servers are considerably higher than for their entry-level “micro” instances, but are still

highly competitive.

Creating a new "Instance"

The Amazon Cloud is full of jargon. "Instance" is probably the first piece of jargon you need to understand. It is Amazon's term for your virtual-server, i.e. your "computer in the cloud". Amazon Instances are not "real" dedicated servers, rather they are some sort of virtual machine running a larger computer. You set up and Instance and can then switch it on and off at will. You are only charged for the service while the instance is running.

Instances can be various flavours of Linux or Windows, with a wide variety of scales from "micro" (single processor with 600 meg of RAM) up to really massive multi-core processors. Pricing rises with machine-size.



	Name	Instance	AMI ID	Root Device	Type	Status	Security Groups	Key
□	Second Test Server	i-a36b88d5	ami-cd517bb9	ebs	t1.micro	● stopped	Morfik Test	Mor
☑	Fullwell Mill Test	i-1324cc65	ami-cd517bb9	ebs	t1.micro	● running	Morfik Test	Mor

AMI ID:	ami-cd517bb9	Zone:	eu-west-1a
Security Groups:	Morfik Test	Type:	t1.micro
Status:	running	Owner:	338048053950
VPC ID:	-	Subnet ID:	-

fig 1: The Amazon Management Console "Instances" page.

Once you have set up an account, open the Amazon Management Console. When you first get there it is bloody confusing. There are lots of tabs with strange names like "Elastic MapReduce", and most of the pages under the tabs don't contain any information until you fill out various details and get an account for that particular sub-service.

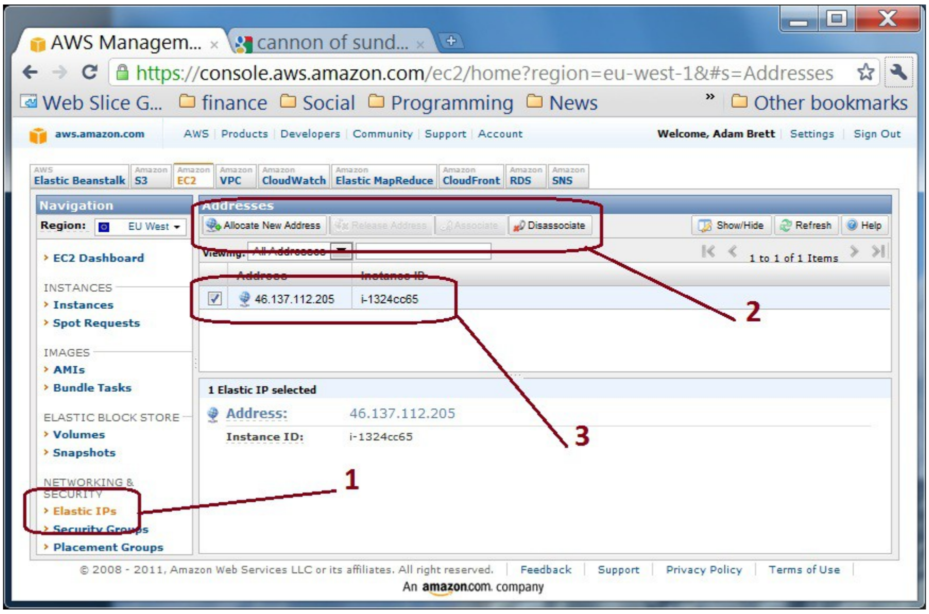
To manage your own servers the tab you want is "EC2" (which stands for "Elastic Cloud 2" ...) Click on this, then click on "Instances" (marked 1. in the figure above).

Managing "Instances"

Click on the "launch instance" button (marked 2. in the figure above) to create a new cloud-computer. You are presented with a multi-page form (Wizard) which takes you through the steps. I will go through this process in more detail below, it is fairly straightforward, though several of the steps involve Amazon-centric jargon which is a bit hard to understand at first.

Once you have set up an Instance you can control it by right clicking on it with the mouse to access a context menu which allows you to undertake a variety of actions on that Instance including start, stop, terminate & retrieve password.

Setting IP addresses and Security Groups for your Instance



Address	Instance ID
☑ 46.137.112.205	i-1324cc65

1 Elastic IP selected

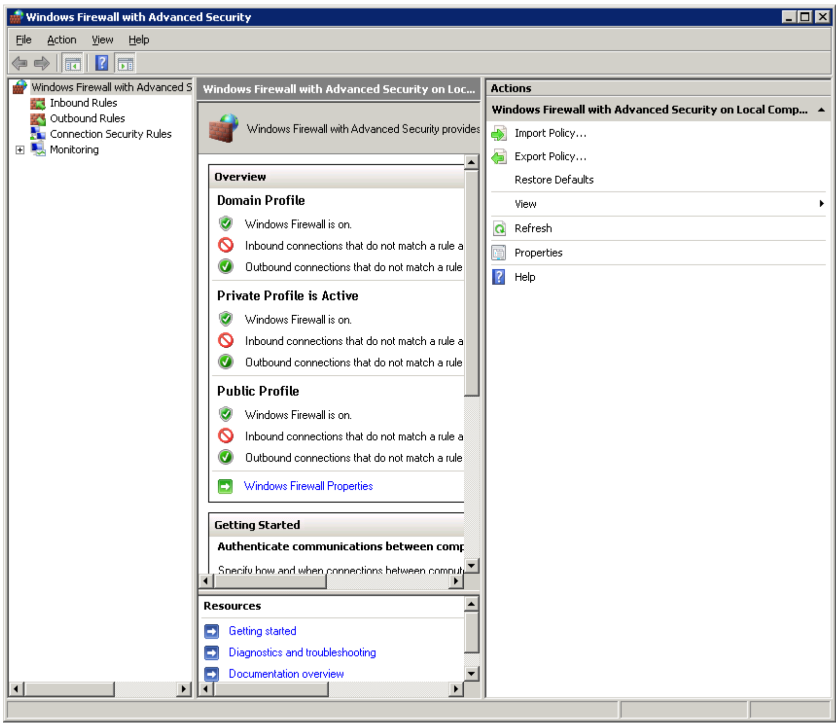
Address: 46.137.112.205

Instance ID: i-1324cc65

fig 2 Amazon Elastic IPs

In the Management Console there is an "Elastic IP" heading (marked 1 in the figure above). Click on this then click on "Allocate new Address" (marked 2 above) to be allocated a new dedicated IP address you can use. Once you have the address right click on the address in the grid (marked 3 above) and choose "associate" from the context-menu to actually link your address to a particular server.

Firewall Access and Permissions


Windows Firewall with Advanced Security
File
Action
View
Help
Windows Firewall with Advanced S
Inbound Rules
Outbound Rules
Connection Security Rules
Monitoring
Windows Firewall with Advanced Security on Loc...
Actions
Windows Firewall with Advanced Security on Local Comp...
Import Policy...
Export Policy...
Restore Defaults
View
Refresh
Properties
Help
Overview
Getting Started
Resources
Getting started
Diagnostics and troubleshooting
Documentation overview
Windows Firewall Properties

Windows Firewall with Advanced Security provides

Domain Profile

Windows Firewall is on.

Inbound connections that do not match a rule a

Outbound connections that do not match a rule

Private Profile is Active

Windows Firewall is on.

Inbound connections that do not match a rule a

Outbound connections that do not match a rule

Public Profile

Windows Firewall is on.

Inbound connections that do not match a rule a

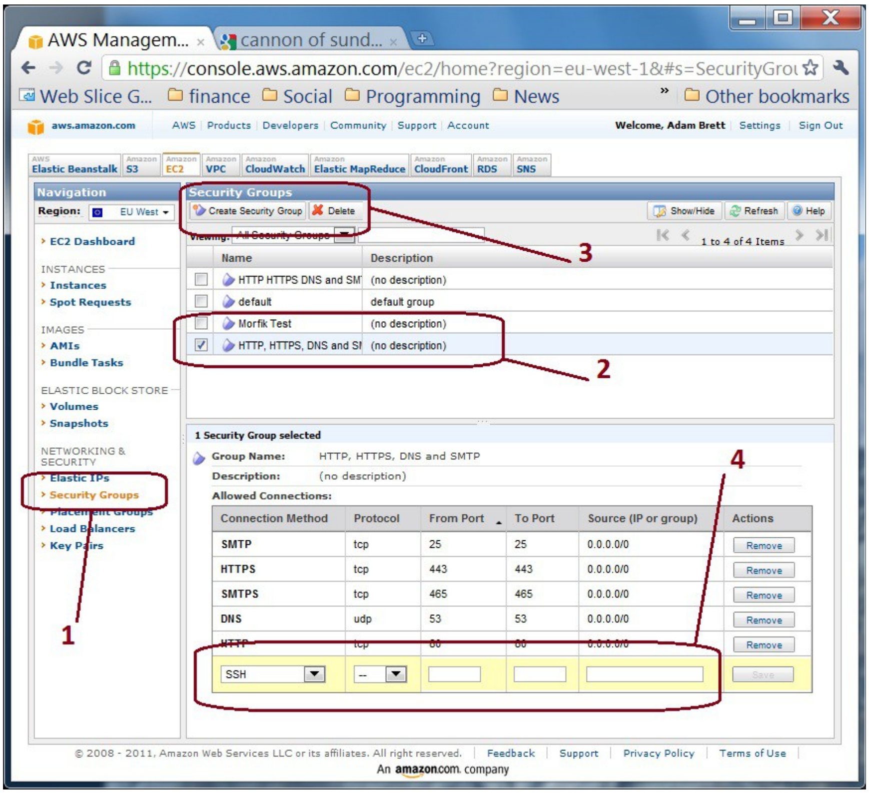
Outbound connections that do not match a rule

Authenticate communications between comp

Specify how and when connections between comput

fig 3: Windows Firewall settings on your server

The way Amazon manages Firewalls is a tiny bit confusing. You can of course manage the firewall of your Instance yourself. You do this with Windows Advanced Firewall on the machine (shown in fig 3, above). How to set up and manage this is beyond the scope of this article, but there is plenty of information on the web.


AWS Managem...
cannon of sund...
https://console.aws.amazon.com/ec2/home?region=eu-west-1&#s=SecurityGrou
Web Slice G... finance Social Programming News Other bookmarks
aws.amazon.com AWS | Products | Developers | Community | Support | Account
Welcome, Adam Brett | Settings | Sign Out
Elastic Beanstalk S3 EC2 VPC CloudWatch Elastic MapReduce CloudFront RDS SNS
Navigation
Region: EU West
EC2 Dashboard
INSTANCES
Instances
Spot Requests
IMAGES
AMIs
Bundle Tasks
ELASTIC BLOCK STORE
Volumes
Snapshots
NETWORKING & SECURITY
Elastic IPs
Security Groups
Placement Groups
Load Balancers
Key Pairs
Security Groups
Create Security Group
Delete
Show/Hide
Refresh
Help
Viewing: All Security Groups
1 to 4 of 4 Items
1
2
3
4
© 2008 - 2011, Amazon Web Services LLC or its affiliates. All right reserved. | Feedback | Support | Privacy Policy | Terms of Use
An amazon.com. company

	Name	Description
□	HTTP HTTPS DNS and SM	(no description)
□	default	default group
□	Morfik Test	(no description)
☑	HTTP, HTTPS, DNS and SI	(no description)

1 Security Group selected

Group Name: HTTP, HTTPS, DNS and SMTP

Description: (no description)

Allowed Connections:

Connection Method	Protocol	From Port	To Port	Source (IP or group)	Actions
SMTP	tcp	25	25	0.0.0.0/0	Remove
HTTPS	tcp	443	443	0.0.0.0/0	Remove
SMTPS	tcp	465	465	0.0.0.0/0	Remove
DNS	udp	53	53	0.0.0.0/0	Remove
HTTP	tcp	80	80	0.0.0.0/0	Remove
SSH	--				Save

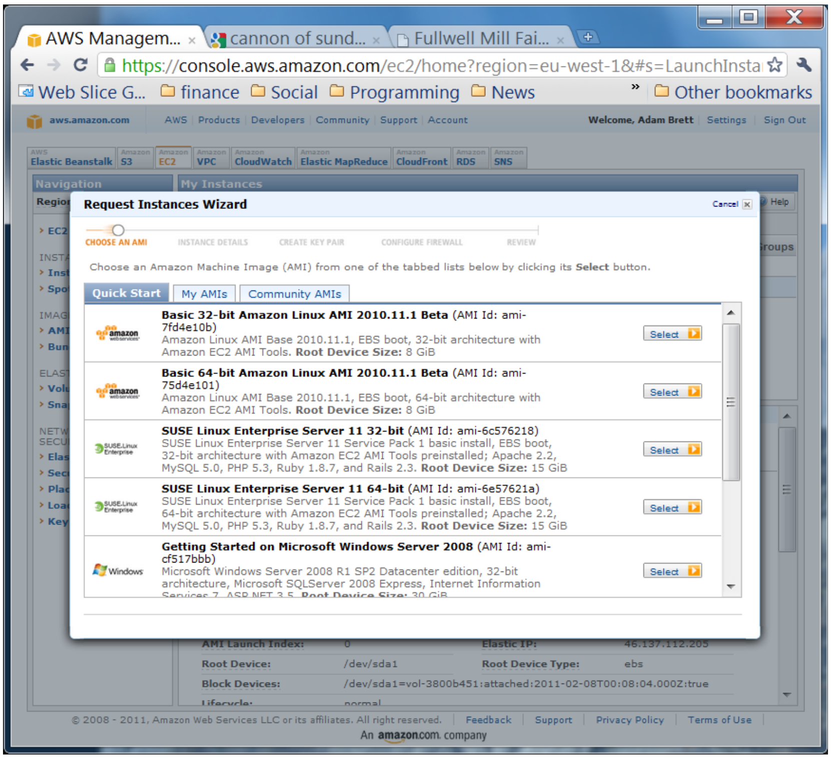
fig 4: Security Groups

The important thing to realise is that Amazon have a secondary external firewall set up for all cloud computers. They call this is their "Security Groups" feature. This lists the permissions which have been granted by Amazon for access to your Instance.

Each named security group represents a selection of permissions for access through Amazon's firewall into your server. You can open the port on your server manually using Windows Advanced Firewall on the machine, but no-one will be able to use this port unless it is also open on the Security Group associated with that Instance.

Security Groups can be edited and changed using the Management Console as shown in the figure above.

The "Launch Instance" Wizard


AWS Managem...
cannon of sund...
Fullwell Mill Fai...
https://console.aws.amazon.com/ec2/home?region=eu-west-1&#s=LaunchInsta
Web Slice G... finance Social Programming News Other bookmarks
aws.amazon.com AWS | Products | Developers | Community | Support | Account
Welcome, Adam Brett | Settings | Sign Out
Elastic Beanstalk S3 EC2 VPC CloudWatch Elastic MapReduce CloudFront RDS SNS
Navigation
My Instances
Cancel
Help
CHOOSE AN AMI
INSTANCE DETAILS
CREATE KEY PAIR
CONFIGURE FIREWALL
REVIEW
Quick Start
My AMIs
Community AMIs
Select
© 2008 - 2011, Amazon Web Services LLC or its affiliates. All right reserved. | Feedback | Support | Privacy Policy | Terms of Use
An amazon.com. company

Request Instances Wizard

Choose an Amazon Machine Image (AMI) from one of the tabbed lists below by clicking its **Select** button.

Basic 32-bit Amazon Linux AMI 2010.11.1 Beta (AMI Id: ami-7fd4e10b)
Amazon Linux AMI Base 2010.11.1, EBS boot, 32-bit architecture with Amazon EC2 AMI Tools. **Root Device Size:** 8 GiB

Basic 64-bit Amazon Linux AMI 2010.11.1 Beta (AMI Id: ami-75d4e101)
Amazon Linux AMI Base 2010.11.1, EBS boot, 64-bit architecture with Amazon EC2 AMI Tools. **Root Device Size:** 8 GiB

SUSE Linux Enterprise Server 11 32-bit (AMI Id: ami-6c576218)
SUSE Linux Enterprise Server 11 Service Pack 1 basic install, EBS boot, 32-bit architecture with Amazon EC2 AMI Tools preinstalled; Apache 2.2, MySQL 5.0, PHP 5.3, Ruby 1.8.7, and Rails 2.3. **Root Device Size:** 15 GiB

SUSE Linux Enterprise Server 11 64-bit (AMI Id: ami-6e57621a)
SUSE Linux Enterprise Server 11 Service Pack 1 basic install, EBS boot, 64-bit architecture with Amazon EC2 AMI Tools preinstalled; Apache 2.2, MySQL 5.0, PHP 5.3, Ruby 1.8.7, and Rails 2.3. **Root Device Size:** 15 GiB

Getting Started on Microsoft Windows Server 2008 (AMI Id: ami-cf517bbb)
Microsoft Windows Server 2008 R1 SP2 Datacenter edition, 32-bit architecture, Microsoft SQLServer 2008 Express, Internet Information Services 7, ASP.NET 3.5. **Root Device Size:** 30 GiB

AMI Launch Index: 0 Elastic IP: 46.137.112.205
Root Device: /dev/sda1 Root Device Type: ebs
Block Devices: /dev/sda1=vol-3800b451:attached:2011-02-08T00:08:04.000Z:true
Lifecycle: normal

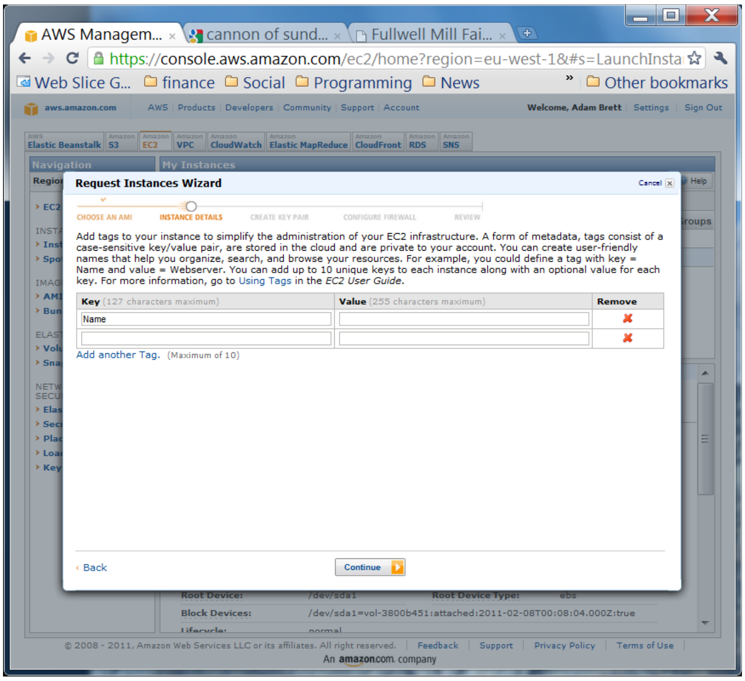
fig 5: Step 1: Pick your base operating system

Key things to note in the above:

- There is a "My AMIs" tab, and a "Community AMIs" tab. These tabs show lists of pre-configured servers. Literally machine-images of the full operating system, installed programmes and hard-disk for virtual machines which you can call into existence. Very similar to the concept of a virtual machine in VMWare.
- The "Quick start" gives a small selection of commonly used operating systems, note that pricing varies according to the set up you choose!
- Of course you can create any server you want, but if you want to create a server to use with Morfik & their auto-deployment it should be Windows Server 2008, either 32 or 64 bit.

Subsequent pages of the wizard allow you to choose the size of the instance and its location. There are advanced instance options, which I am guessing only matter once you have many machines working simultaneously.

Then come some sections which confused me at first, so I think deserve a bit of clarification



Request Instances Wizard

CHOOSE AN AMI · INSTANCE DETAILS · CREATE KEY PAIR · CONFIGURE FIREWALL · REVIEW

Add tags to your instance to simplify the administration of your EC2 infrastructure. A form of metadata, tags consist of a case-sensitive key/value pair, are stored in the cloud and are private to your account. You can create user-friendly names that help you organize, search, and browse your resources. For example, you could define a tag with key = Name and value = Webserver. You can add up to 10 unique keys to each instance along with an optional value for each key. For more information, go to Using Tags in the *EC2 User Guide*.

Key (127 characters maximum)	Value (255 characters maximum)	Remove
Name		✗
		✗

Add another Tag. (Maximum of 10)

‹ Back

Continue

Root Device: /dev/sda1 Root Device Type: ebs

Block Devices: /dev/sda1=vol-3800b451:attached:2011-02-08T00:08:04.000Z:true

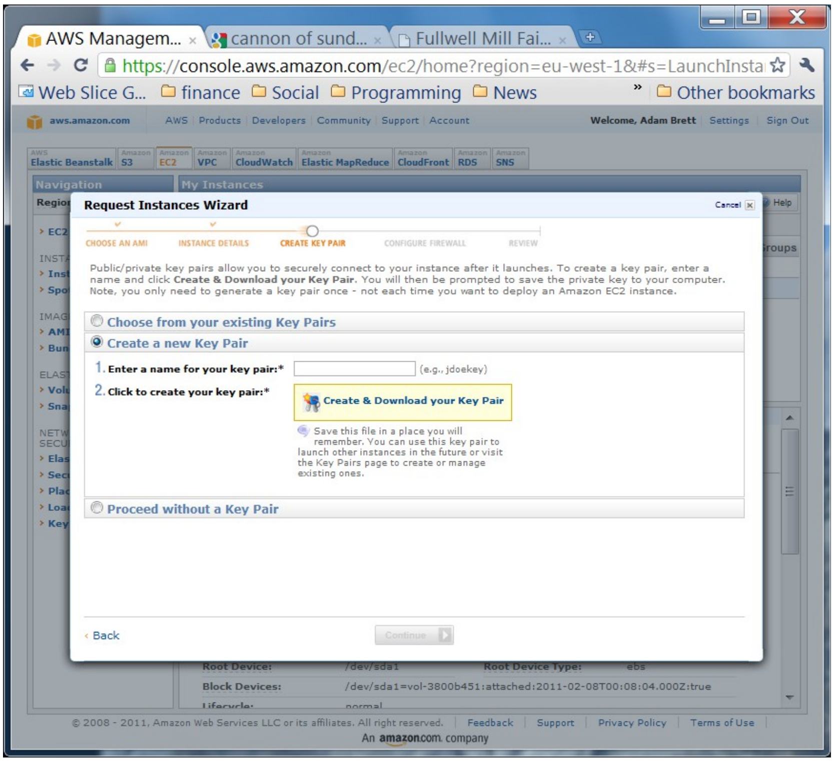
Lifecycle: normal

© 2008 - 2011, Amazon Web Services LLC or its affiliates. All right reserved. Feedback | Support | Privacy Policy | Terms of Use

An amazon.com. company

fig 6: Instance Details Key Value dialogue

You come to this & think "what on earth is it for, it is really serious" ... well no it isn't. All it is is a space which you can write down some meta-data which will be stored with your instance to identify it for your own management purposes. You don't have to add anything here, and you can edit it later.



Request Instances Wizard

Public/private key pairs allow you to securely connect to your instance after it launches. To create a key pair, enter a name and click **Create & Download your Key Pair**. You will then be prompted to save the private key to your computer. Note, you only need to generate a key pair once - not each time you want to deploy an Amazon EC2 instance.

○ Choose from your existing Key Pairs

● 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.

○ Proceed without a Key Pair

Root Device: /dev/sda1 Root Device Type: ebs

Block Devices: /dev/sda1=vol-3800b451:attached:2011-02-08T00:08:04.000Z:true

© 2008 - 2011, Amazon Web Services LLC or its affiliates. All right reserved. | Feedback | Support | Privacy Policy | Terms of Use

An amazon.com company

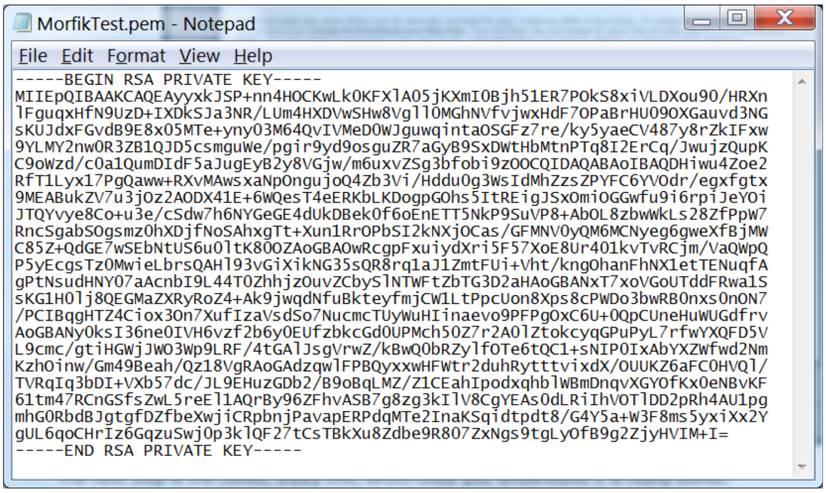
fig 7: Security Key Pair Creation

Instance Key Pairs

The next step is the clever, tricky one, which once you understand it is really useful.

Your server has to be secure, but you have to be able to access it. It is hard to square off these 2 conflicting requirements. Amazon have created a security key process which allows you to access your server safely.

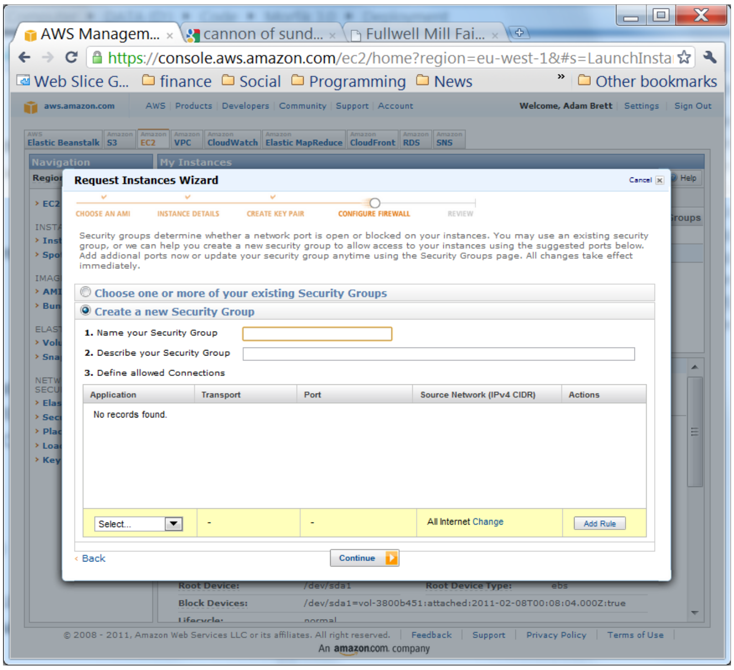
First you name the security key (this name is not critical). Then you down-load a long key file, which will look something like this:



```
-----BEGIN RSA PRIVATE KEY-----
MIIEpQIBAAKCAQEAyyxkJSP+nn4HOCKwLk0KFXlA05jKXmI0Bjh51ER7POkS8xiVLDXou90/HRXn
lFguqxHfN9UzD+IXDkSJa3NR/LUm4HXDVwSHw8Vgll0MGhNVfvjwxHdF7OPaBrHU09OXGauvd3NG
sKUJdxFGvdB9E8x05MTe+yny03M64QvIVMeD0WJguwqintaOSGFz7re/ky5yaeCV487y8rZkIFxw
9YLMY2nw0R3ZB1QJD5csmguWe/pgir9yd9osguZR7aGyB9SxDWtHbMtnPTq8I2ErCq/JwujzQupK
C9oWzd/c0a1QumDIdF5aJugEyB2y8VGjw/m6uxvZSg3bfobi9zOOCQIDAQABAoIBAQDHiwu4Zoe2
RfT1Lyx17PgQaww+RXvMAwsxaNpOngujoQ4Zb3Vi/Hddu0g3WsIdMhZzsZPYFC6YVOdr/egxfgtx
9MEABukZV7u3jOz2AODX41E+6WQesT4eERKbLKDogpGOhs5ItREigJSxOmiOGGwfu9i6rpiJeYOi
JTQYvye8Co+u3e/cSdw7h6NYGeGE4dUkDBek0f6oEnETT5NkP9SuVP8+AbOL8zbwWkLs28ZfPpW7
RncSgabSOgsmz0hXDjfNoSAhxgTt+Xun1RrOPbSI2kNXjOCas/GFMNV0yQM6MCNyeg6gweXfBjMW
C85Z+QdGE7wSEbNtUS6u0ltK80OZAoGBAOwRcgpFxuiydXri5F57XoE8Ur401kvTvRCjm/VaQWpQ
P5yEcgsTz0MwieLbrsQAHl93vGiXikNG35sQR8rq1aJ1ZmtFUi+Vht/kngOhanFhNX1etTENuqfA
gPtNsudHNY07aAcnbI9L44T0ZhhjzOuvZCbySlNTWFtZbTG3D2aHAoGBANxT7xoVGoUTddFRwa1S
sKG1H0lj8QEGMaZXRyRoZ4+Ak9jwqdNfuBkteyfmjCW1LtPpcUon8Xps8cPWDo3bwRB0nxs0nON7
/PCIBqgHTZ4Ciox3On7XufIzaVsdSo7NucmcTUyWuHIinaevo9PFPgOxC6U+0QpCUneHuWUGdfrv
AoGBANy0ksI36ne0IVH6vzf2b6y0EUfzbkcGd0UPMch50Z7r2A0lZtokcyqGPuPyL7rfwYXQFD5V
L9cmc/gtiHGWjJWO3Wp9LRF/4tGAlJsgVrwZ/kBwQ0bRZylf0Te6tQC1+sNIP0IxAbYXZWfwd2Nm
KzhOinw/Gm49Beah/Qz18VgRAoGAdzqwlFPBQyxxwHFWtr2duhRytttvixdX/OUUKZ6aFC0HVQl/
TVRqIq3bDI+VXb57dc/JL9EHuzGDb2/B9oBqLMZ/Z1CEahIpodxqhblWBmDnqvXGYOfKx0eNBvKF
61tm47RCnGSfsZwL5reEl1AQrBy96ZFhvASB7g8zg3kIlV8CgYEAs0dLRiIhVOTlDD2pRh4AU1pg
mhG0RbdBJgtgfDZfbeXwjiCRpbnjPavapERPdqMTe2InaKSqidtpdt8/G4Y5a+W3F8ms5yxiXx2Y
gUL6qoCHrIz6GqzuSwj0p3klQF27tCsTBkXu8Zdbe9R807ZxNgs9tgLyOfB9g2ZjyHVIM+I=
-----END RSA PRIVATE KEY-----
```

fig 8: A "PEM" key file

You will use this key later (just once) to access your server to connect to it and retrieve your log-on password.



Request Instances Wizard

CHOOSE AN AMI · INSTANCE DETAILS · CREATE KEY PAIR · CONFIGURE FIREWALL · REVIEW

Security groups determine whether a network port is open or blocked on your instances. You may use an existing security group, or we can help you create a new security group to allow access to your instances using the suggested ports below. Add addional ports now or update your security group anytime using the Security Groups page. All changes take effect immediately.

○ Choose one or more of your existing Security Groups

● Create a new Security Group

1. Name your Security Group
2. Describe your Security Group
3. Define allowed Connections

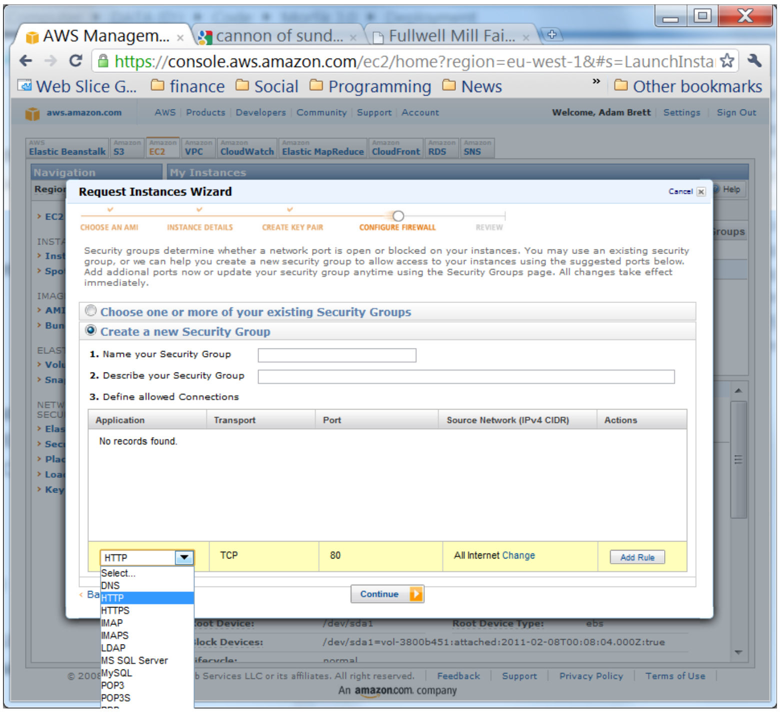
Application	Transport	Port	Source Network (IPv4 CIDR)	Actions
No records found.				
Select...	-	-	All Internet Change	Add Rule

https://console.aws.amazon.com/ec2/home?region=eu-west-1&#s=LaunchInsta
Cancel
Back
Continue

fig 9: Firewall Configuration

Create Security Group

This was probably the most confusing step for me the first time I did it. Here you are creating "Security Groups". The problem is that if you are new to the whole process you have no idea of the purpose of this step. Remember from the start of the article, Security Groups (once you have set them up) are the External Firewall Amazon create between your web-server ("Instance") and the outside world. For you or any other users to access the server (for example as a server for a website) you have to open Ports on the machine to allow communication. The purpose of this step is therefore to pick the ports you want to have open through the external Amazon firewall into your Instance. These allow the Instance to do things like access the web, act as a web-server, mail-server etc.



Request Instances Wizard

CHOOSE AN AMI · INSTANCE DETAILS · CREATE KEY PAIR · CONFIGURE FIREWALL · REVIEW

Security groups determine whether a network port is open or blocked on your instances. You may use an existing security group, or we can help you create a new security group to allow access to your instances using the suggested ports below. Add addional ports now or update your security group anytime using the Security Groups page. All changes take effect immediately.

○ Choose one or more of your existing Security Groups

● Create a new Security Group

1. Name your Security Group
2. Describe your Security Group
3. Define allowed Connections

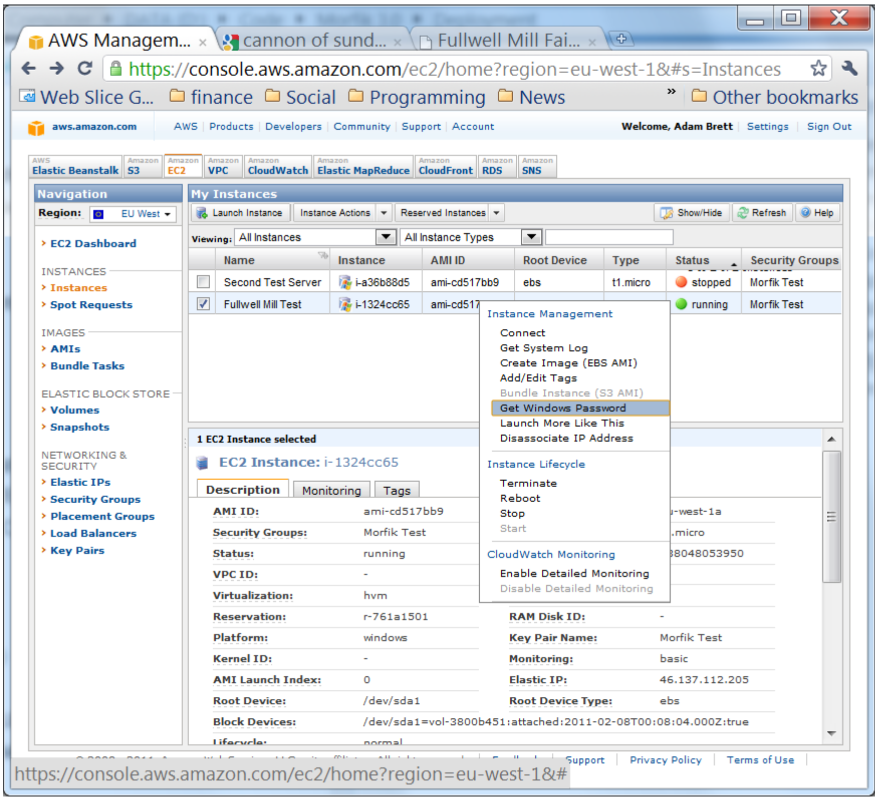
Application	Transport	Port	Source Network (IPv4 CIDR)	Actions
No records found.				
HTTP	TCP	80	All Internet Change	Add Rule

fig 10: Adding a security rule to open a port

Once you understand the process it is quite easy: Name your Security Group, and add a number of open ports with their respective rules. Amazon have created a number of entries on the list for commonly used ports. If you plan to use the Instance as a web-server to host a web-site the main port you want to open is 80, which is the second one on the list. To connect to your server remotely (very important for later!) you should also add permission for "RDP" (Port 3389) at this step. If you have something like a custom database you want to be able to access on the server you will need to add Custom rules on this screen detailing the ports you need to open.

Explaining the purpose of different open ports is beyond the scope of this article, but there is lots of information about it on the web. This is the last step in creating an Instance, the wizard now shows a "review" page, which allows you to double check what you have done & make changes prior to clicking the "launch" button.

Actually connecting to your Instance



https://console.aws.amazon.com/ec2/home?region=eu-west-1&#s=Instances

Welcome, Adam Brett

Region: EU West

Viewing: All Instances | All Instance Types

	Name	Instance	AMI ID	Root Device	Type	Status	Security Groups
□	Second Test Server	i-a36b88d5	ami-cd517bb9	ebs	t1.micro	stopped	Morfik Test
☑	Fullwell Mill Test	i-1324cc65	ami-cd517			running	Morfik Test

1 EC2 Instance selected

EC2 Instance: i-1324cc65

AMI ID:	ami-cd517bb9		-west-1a
Security Groups:	Morfik Test		.micro
Status:	running		8048053950
VPC ID:	-		
Virtualization:	hvm		
Reservation:	r-761a1501	RAM Disk ID:	-
Platform:	windows	Key Pair Name:	Morfik Test
Kernel ID:	-	Monitoring:	basic
AMI Launch Index:	0	Elastic IP:	46.137.112.205
Root Device:	/dev/sda1	Root Device Type:	ebs
Block Devices:	/dev/sda1=vol-3800b451:attached:2011-02-08T00:08:04.000Z:true		

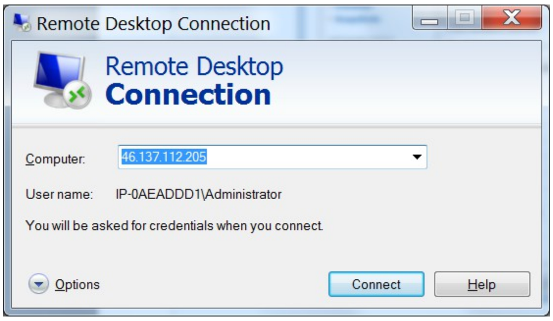
https://console.aws.amazon.com/ec2/home?region=eu-west-1&#

fig 11: Connecting

Once the instance is created it takes a few minutes to set up & get working, so go & make a cup of coffee. Then, back in the main Management Console, right click on the Instance you have just created and click "get windows password" you will be prompted to paste in the "PEM Key" (shown in fig 8, above). The system will work for a while & then give you an Administrator Password. **Note this down carefully somewhere.**

Your server now exists, and is running. You can start and stop it in the Management Console, you can also Terminate it (which permanently erases the whole machine!). As a small aside, you can now also click on the "Monitoring" tab to see activity on the machine, and allocate it an IP address to the Instance if it is going to need one.

But what you really want to do is actually log on to it. This process is fairly easy with newer versions of Windows. I know it is a bit more complicated with earlier versions, I only explain it for Windows 7 here.



Computer: 46.137.112.205

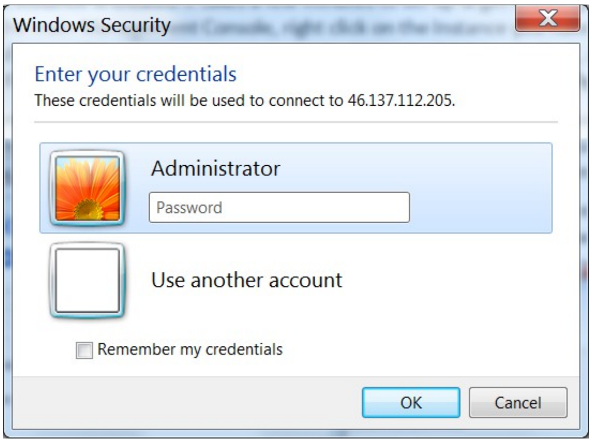
User name: IP-0AEADDD1\Administrator

You will be asked for credentials when you connect.

fig 11: Remote Connection log-in

First run "Remote Desktop Connection". Type this into the "start" menu if you don't know where it is hiding. The only thing to note here is that I have typed the Elastic IP address I have allocated to the machine into the "computer" field.

Click on Connect, and you will be prompted for log on credentials and password



Windows Security

Enter your credentials

These credentials will be used to connect to 46.137.112.205.

fig 12: Log on screen

Just fill in the password retrieved in step 10 above, click OK & in a few moments you will be connected to the server. Once you have the main screen of your server open you can start to use it as if it was your own machine.

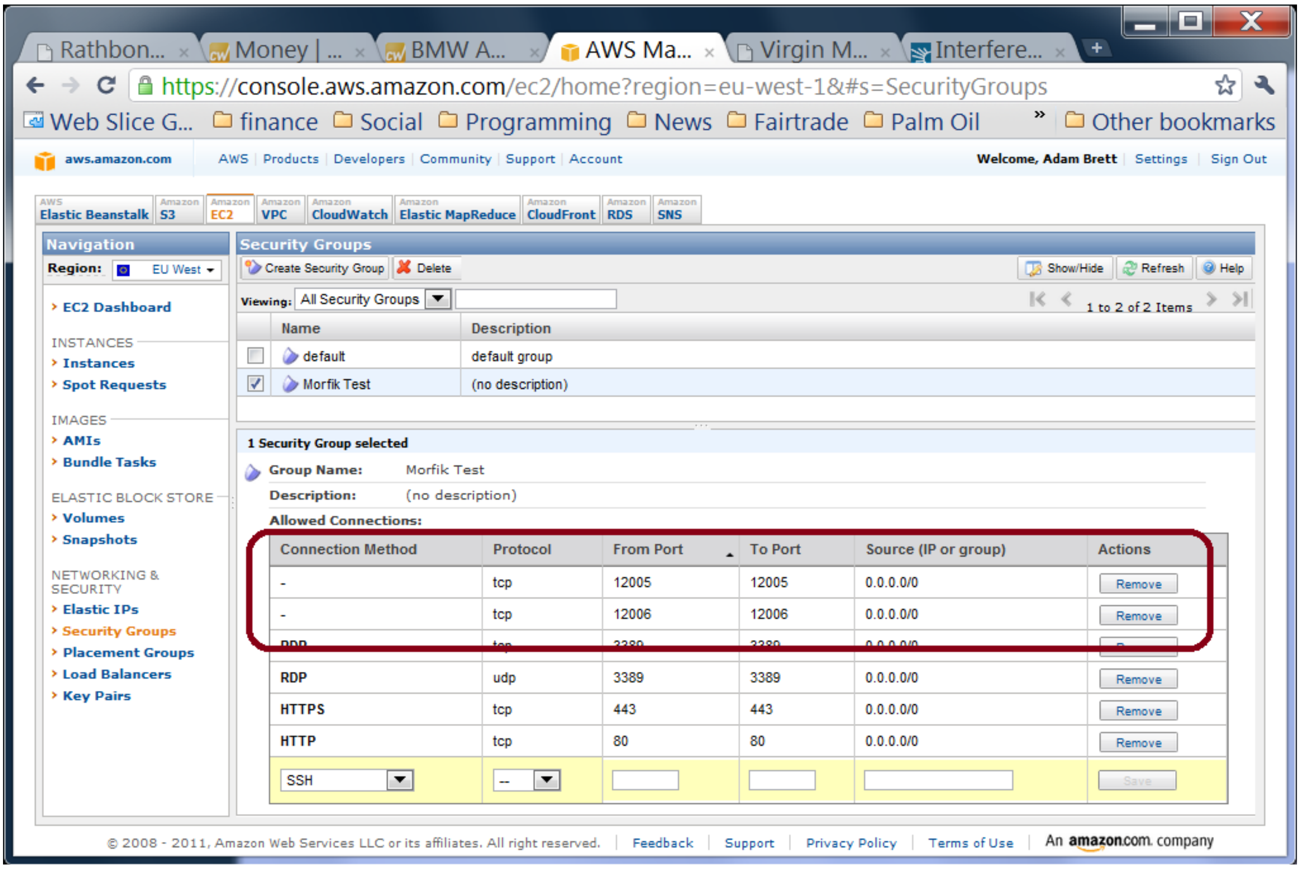
A few extra things to think about:

- If you haven't used Remote Connection before, spend a while looking at the "Options" (small button on fig 11 above).
- If you want to transfer files up & down to this server from your own computer via FTP you will have to open Pots you are using for FTP by adding these to the security group (fig 4 above) & Firewall permissions.

Setting up an Elevate or DBISAM Database on an Amazon Instance

If anything this is even easier than setting up a Morfik server. Again, remember that once you have set up the machine you can save it (either with or without the database files added to it).

- First set up a Security Group which opens the necessary ports (12010/12011 for EDB in its standard setting) needed by the server. Link this security group to your Instance (see fig 13 below).
- Create a new instance, going through the steps already detailed above.
- Request a new Elastic IP from Amazon, and assign it to your new instance.
- Get remote access to your Instance and copy EDBSRVR.exe and EDBMgr.exe onto it.
- Set up this computer as a server: Set EDBSRVR.exe running, ensure that it starts whenever the computer re-starts, use EDBMgr.exe to set up the databases, passwords, users etc. that your application requires.
- Set up a new Firewall Rule on the server's Internal Firewall (fig 3 above, and fig 16., below) Step 1, select "Inbound Rule", Step 2 select "Create New Rule". This will open a wizard. Follow through the steps. The new Rule is for a Port, it is a TCP rule, for a Specific Local Port, you should allow the connection, for Domain, Private and Public connections, and then name it as you wish. Do this twice, once for port 12010 and once for 12011.
- The rule will now appear in the "Inbound Rules" grid. Right click on it and select "Properties". In the window which appears. Under the General tab ensure that "Allow Connections" is ticked, and under the Programs and Services tab click on "This Program" and Browse to the location where EDBSRVR.exe is stored on the Instance.
- Finally set any extra rules you may want to such as limitations on Remote IP addresses allowed to access the server.
- On your own computer it should now be possible to connect to the remote server using EDBMgr or our own Delphi application. You simply have to get the IP Address for the Remote Database correct, and fill in Username and Password correctly.



https://console.aws.amazon.com/ec2/home?region=eu-west-1&#s=SecurityGroups

Region: EU West

Viewing: All Security Groups

	Name	Description
□	default	default group
☑	Morfik Test	(no description)

1 Security Group selected

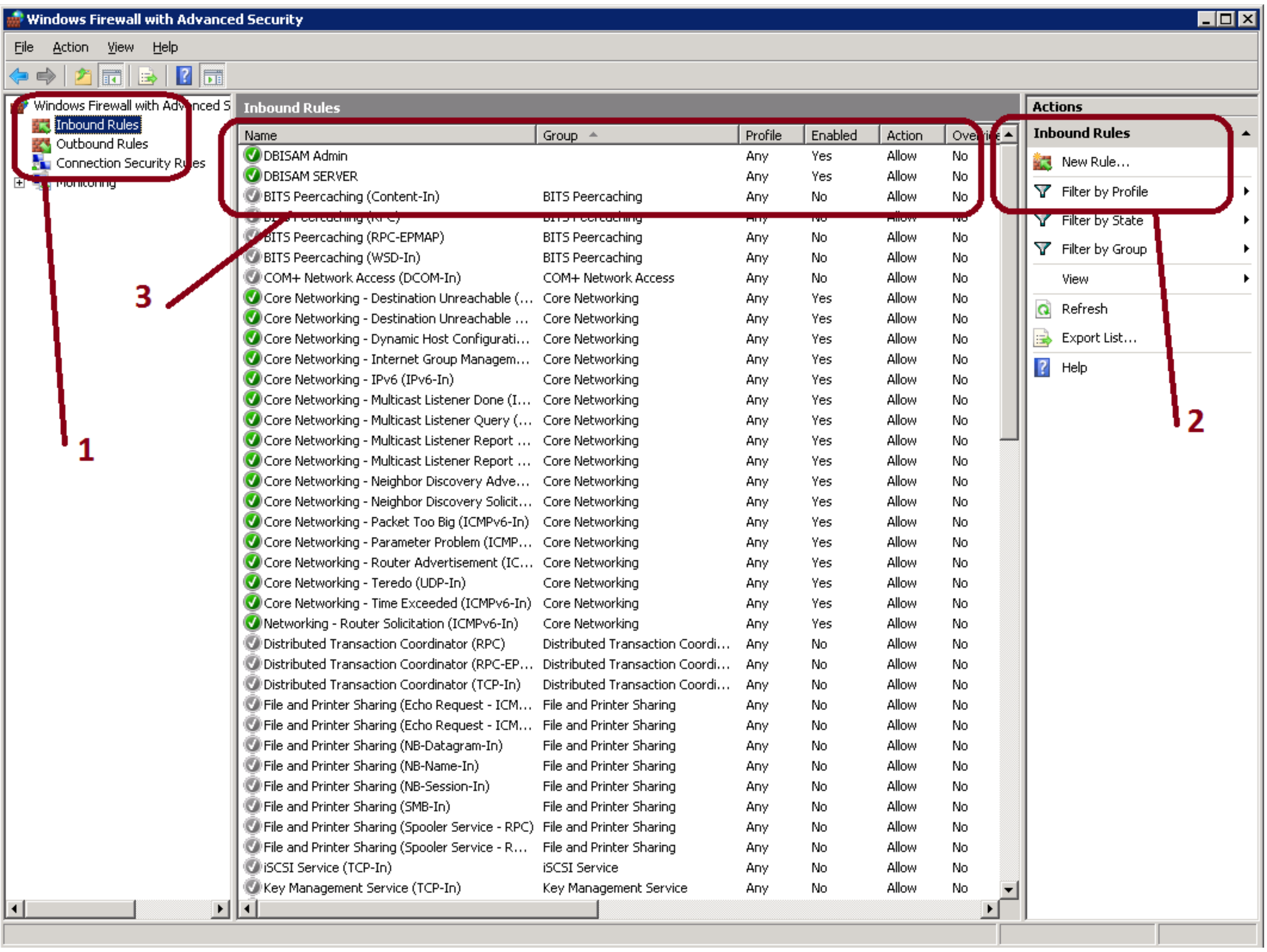
Group Name: Morfik Test

Description: (no description)

Allowed Connections:

Connection Method	Protocol	From Port	To Port	Source (IP or group)	Actions
-	tcp	12005	12005	0.0.0.0/0	Remove
-	tcp	12006	12006	0.0.0.0/0	Remove
RDP	tcp	3389	3389	0.0.0.0/0	Remove
RDP	udp	3389	3389	0.0.0.0/0	Remove
HTTPS	tcp	443	443	0.0.0.0/0	Remove
HTTP	tcp	80	80	0.0.0.0/0	Remove
SSH	--				Save

fig 13: Amazon console security group for Elevate DBISAM



Inbound Rules

Name	Group	Profile	Enabled	Action	Override
DBISAM Admin		Any	Yes	Allow	No
DBISAM SERVER		Any	Yes	Allow	No
BITS Peercaching (Content-In)	BITS Peercaching	Any	No	Allow	No
BITS Peercaching (RPC)	BITS Peercaching	Any	No	Allow	No
BITS Peercaching (RPC-EPMAP)	BITS Peercaching	Any	No	Allow	No
BITS Peercaching (WSD-In)	BITS Peercaching	Any	No	Allow	No
COM+ Network Access (DCOM-In)	COM+ Network Access	Any	No	Allow	No
Core Networking - Destination Unreachable (...	Core Networking	Any	Yes	Allow	No
Core Networking - Destination Unreachable ...	Core Networking	Any	Yes	Allow	No
Core Networking - Dynamic Host Configurati...	Core Networking	Any	Yes	Allow	No
Core Networking - Internet Group Managem...	Core Networking	Any	Yes	Allow	No
Core Networking - IPv6 (IPv6-In)	Core Networking	Any	Yes	Allow	No
Core Networking - Multicast Listener Done (I...	Core Networking	Any	Yes	Allow	No
Core Networking - Multicast Listener Query (...	Core Networking	Any	Yes	Allow	No
Core Networking - Multicast Listener Report ...	Core Networking	Any	Yes	Allow	No
Core Networking - Multicast Listener Report ...	Core Networking	Any	Yes	Allow	No
Core Networking - Neighbor Discovery Adve...	Core Networking	Any	Yes	Allow	No
Core Networking - Neighbor Discovery Solicit...	Core Networking	Any	Yes	Allow	No
Core Networking - Packet Too Big (ICMPv6-In)	Core Networking	Any	Yes	Allow	No
Core Networking - Parameter Problem (ICMP...	Core Networking	Any	Yes	Allow	No
Core Networking - Router Advertisement (IC...	Core Networking	Any	Yes	Allow	No
Core Networking - Teredo (UDP-In)	Core Networking	Any	Yes	Allow	No
Core Networking - Time Exceeded (ICMPv6-In)	Core Networking	Any	Yes	Allow	No
Networking - Router Solicitation (ICMPv6-In)	Core Networking	Any	Yes	Allow	No
Distributed Transaction Coordinator (RPC)	Distributed Transaction Coordi...	Any	No	Allow	No
Distributed Transaction Coordinator (RPC-EP...	Distributed Transaction Coordi...	Any	No	Allow	No
Distributed Transaction Coordinator (TCP-In)	Distributed Transaction Coordi...	Any	No	Allow	No
File and Printer Sharing (Echo Request - ICM...	File and Printer Sharing	Any	No	Allow	No
File and Printer Sharing (Echo Request - ICM...	File and Printer Sharing	Any	No	Allow	No
File and Printer Sharing (NB-Datagram-In)	File and Printer Sharing	Any	No	Allow	No
File and Printer Sharing (NB-Name-In)	File and Printer Sharing	Any	No	Allow	No
File and Printer Sharing (NB-Session-In)	File and Printer Sharing	Any	No	Allow	No
File and Printer Sharing (SMB-In)	File and Printer Sharing	Any	No	Allow	No
File and Printer Sharing (Spooler Service - RPC)	File and Printer Sharing	Any	No	Allow	No
File and Printer Sharing (Spooler Service - R...	File and Printer Sharing	Any	No	Allow	No
iSCSI Service (TCP-In)	iSCSI Service	Any	No	Allow	No
Key Management Service (TCP-In)	Key Management Service	Any	No	Allow	No

fig 16. Windows Firewall steps to create access permissions.

Pricing

I have been using a "Micro Instance" with Windows 2008.

As standard this comes in at $0.035 per hour (hosted in Ireland), which equated to roughly £195 per annum for an always-on machine. I will convert this to a "reserved" (always on) micro instance once I am sure I have it set up right. This only costs $0.016 per hour plus an annual fee equating to £124.00 annually.

If I were to upgrade this to larger reserved instances the annual fee would be: "Small" £485.00, "Large" £1,943.00. I am not certain why there is such a big jump between "micro" and "small", my guess is that some of the more memory-hungry large databases require the larger servers, making this extra charging possible.

If you were to run on Linux (not currently possible with Morfik Auto-deployment as far as I understand) the above costs fall by 33%.

The above costs do not include data-transfer. This is priced as follows: First 1gb is free. Thereafter prices start at roughly £0.10 per gb, and fall to £0.05 per gb in bands set in tens of Terabytes. Note that if you house several Amazon cloud servers together and transfer data between them there is no charge for these inter-machine transfers so long as they are in the same region (i.e. Eastern USA, Western USA or Europe).