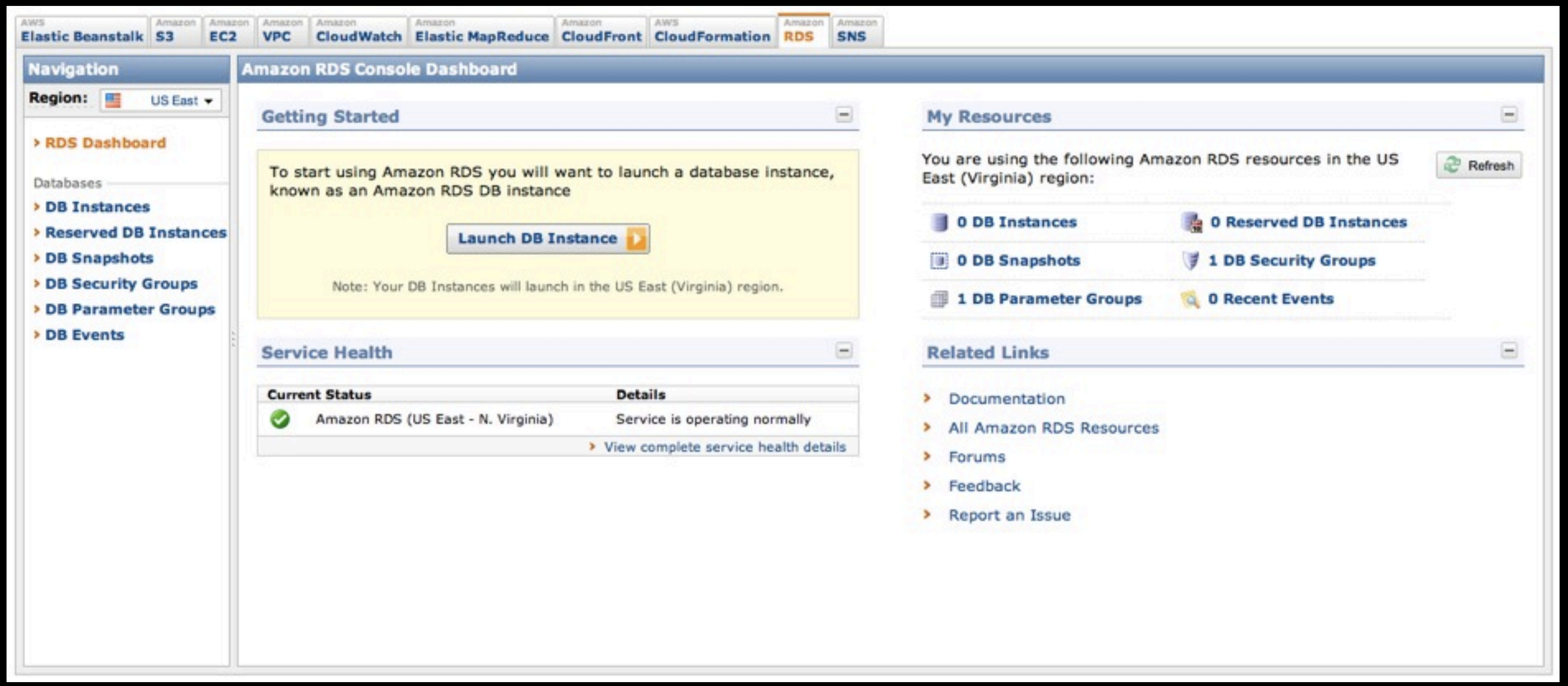
Deploying Web Applications to the Cloud

ThoughtWorks®

Cosmin Stejerean

@offbytwo

AWS Console


Elastic Beanstalk
S3
EC2
VPC
CloudWatch
Elastic MapReduce
CloudFront
CloudFormation
RDS
SNS
Navigation
Region:
US East
RDS Dashboard
Databases
DB Instances
Reserved DB Instances
DB Snapshots
DB Security Groups
DB Parameter Groups
DB Events
Amazon RDS Console Dashboard
Getting Started
To start using Amazon RDS you will want to launch a database instance, known as an Amazon RDS DB instance
Launch DB Instance
Note: Your DB Instances will launch in the US East (Virginia) region.
Service Health
Current Status
Details
Amazon RDS (US East - N. Virginia)
Service is operating normally
View complete service health details
My Resources
You are using the following Amazon RDS resources in the US East (Virginia) region:
Refresh
0 DB Instances
0 Reserved DB Instances
0 DB Snapshots
1 DB Security Groups
1 DB Parameter Groups
0 Recent Events
Related Links
Documentation
All Amazon RDS Resources
Forums
Feedback
Report an Issue

Today's goals

- Deploy sample Django application to EC2
- Serve static assets from CloudFront CDN
- Move database to RDS
- Load balance using ELB

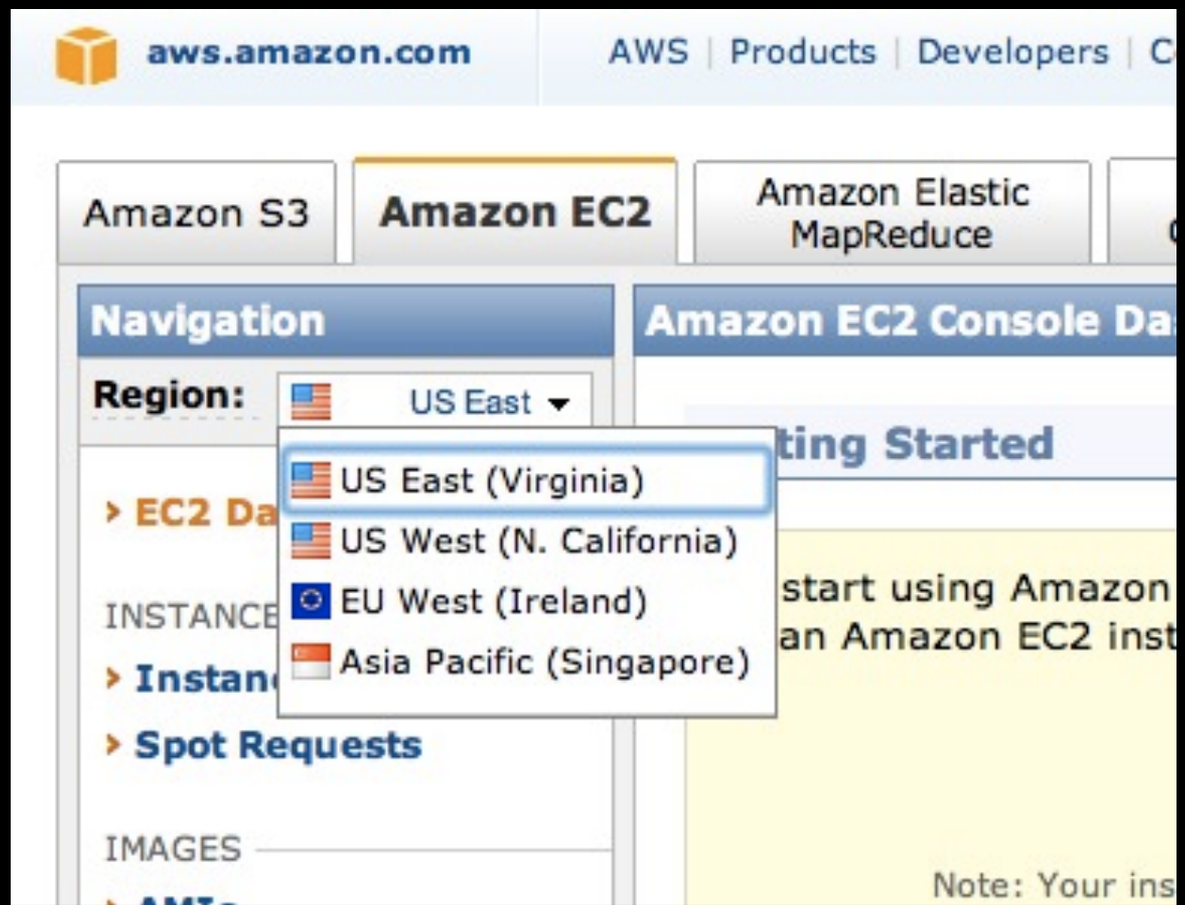
Before we begin

- Download http://bit.ly/pycon-type-this

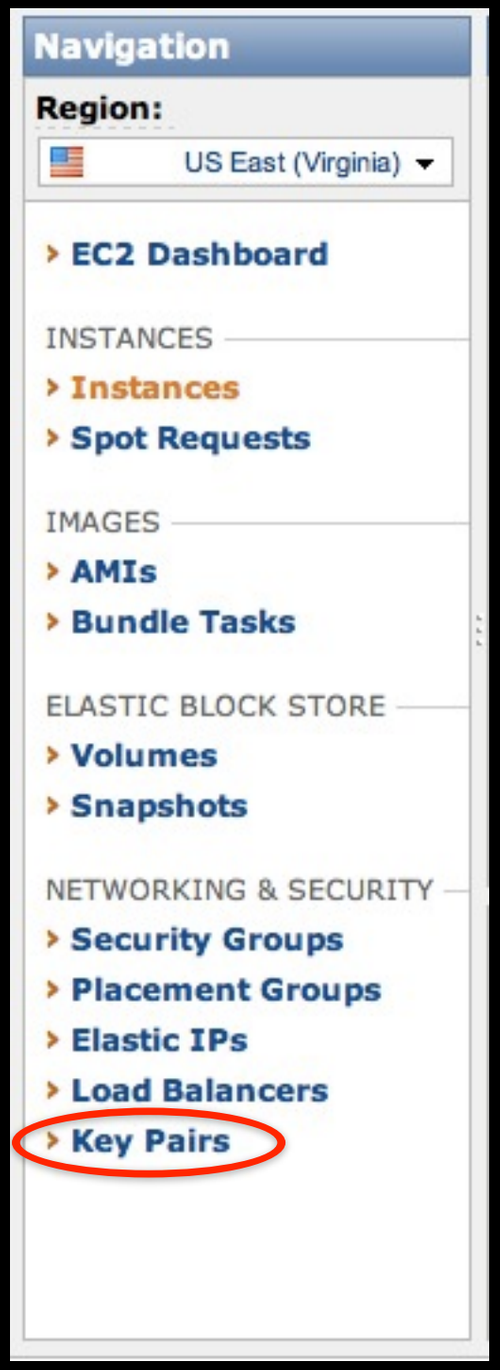
Exercise 1

- Create EBS volume
- Configure Database
- Deploy Django app

Select Region



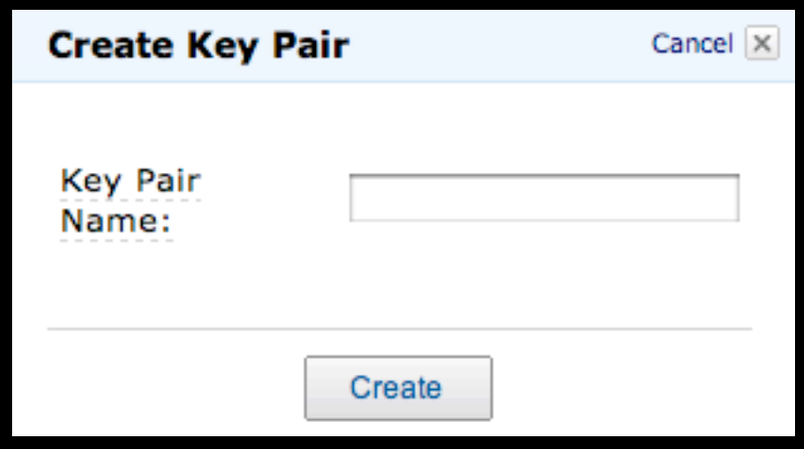
Key Pairs


Navigation
Region:
US East (Virginia)
EC2 Dashboard
INSTANCES
Instances
Spot Requests
IMAGES
AMIs
Bundle Tasks
ELASTIC BLOCK STORE
Volumes
Snapshots
NETWORKING & SECURITY
Security Groups
Placement Groups
Elastic IPs
Load Balancers
Key Pairs

or


My Resources
You are using the following Amazon EC2 resources in the US East (Virginia) region:
Refresh
1 Running Instance
1 Elastic IP
2 EBS Volumes
1 EBS Snapshot
1 Key Pair
1 Security Group
0 Load Balancers
0 Placement Groups

Generate Key Pair



Windows Users

- http://bit.ly/awsputty

For Windows on Intel x86

PuTTY:	putty.exe	(or by FTP)	(RSA sig)	(DSA sig)
PuTTYtel:	puttytel.exe	(or by FTP)	(RSA sig)	(DSA sig)
PSCP:	pscp.exe	(or by FTP)	(RSA sig)	(DSA sig)
PSFTP:	psftp.exe	(or by FTP)	(RSA sig)	(DSA sig)
Plink:	plink.exe	(or by FTP)	(RSA sig)	(DSA sig)
Pageant:	pageant.exe	(or by FTP)	(RSA sig)	(DSA sig)
PuTTYgen:	puttygen.exe	(or by FTP)	(RSA sig)	(DSA sig)

A .ZIP file containing all the binaries (except PuTTYtel), and also the help files

Zip file:	putty.zip	(or by FTP)	(RSA sig)	(DSA sig)

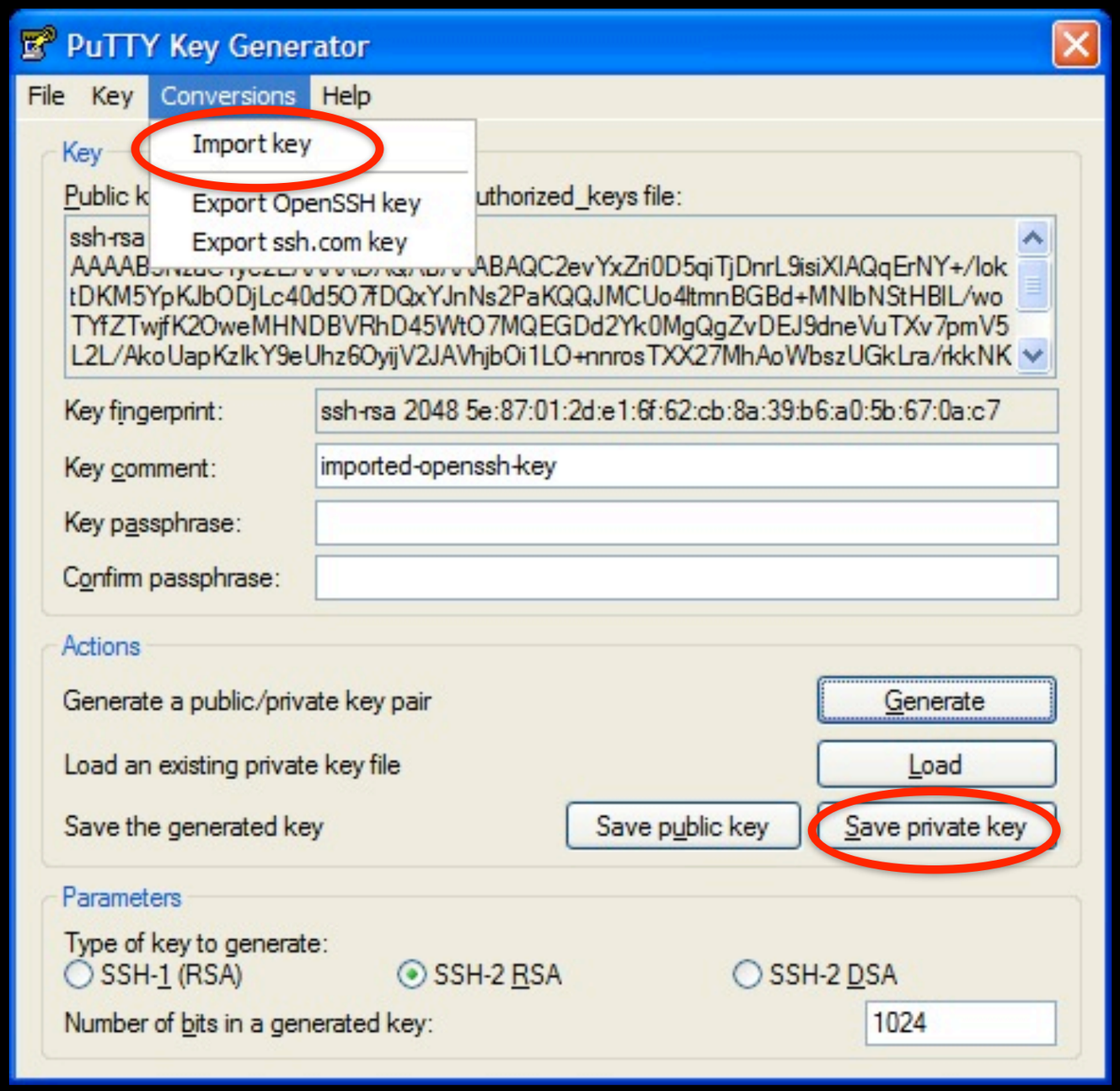
A Windows installer for everything except PuTTYtel

Installer:	putty-0.60-installer.exe	(or by FTP)	(RSA sig)	(DSA sig)

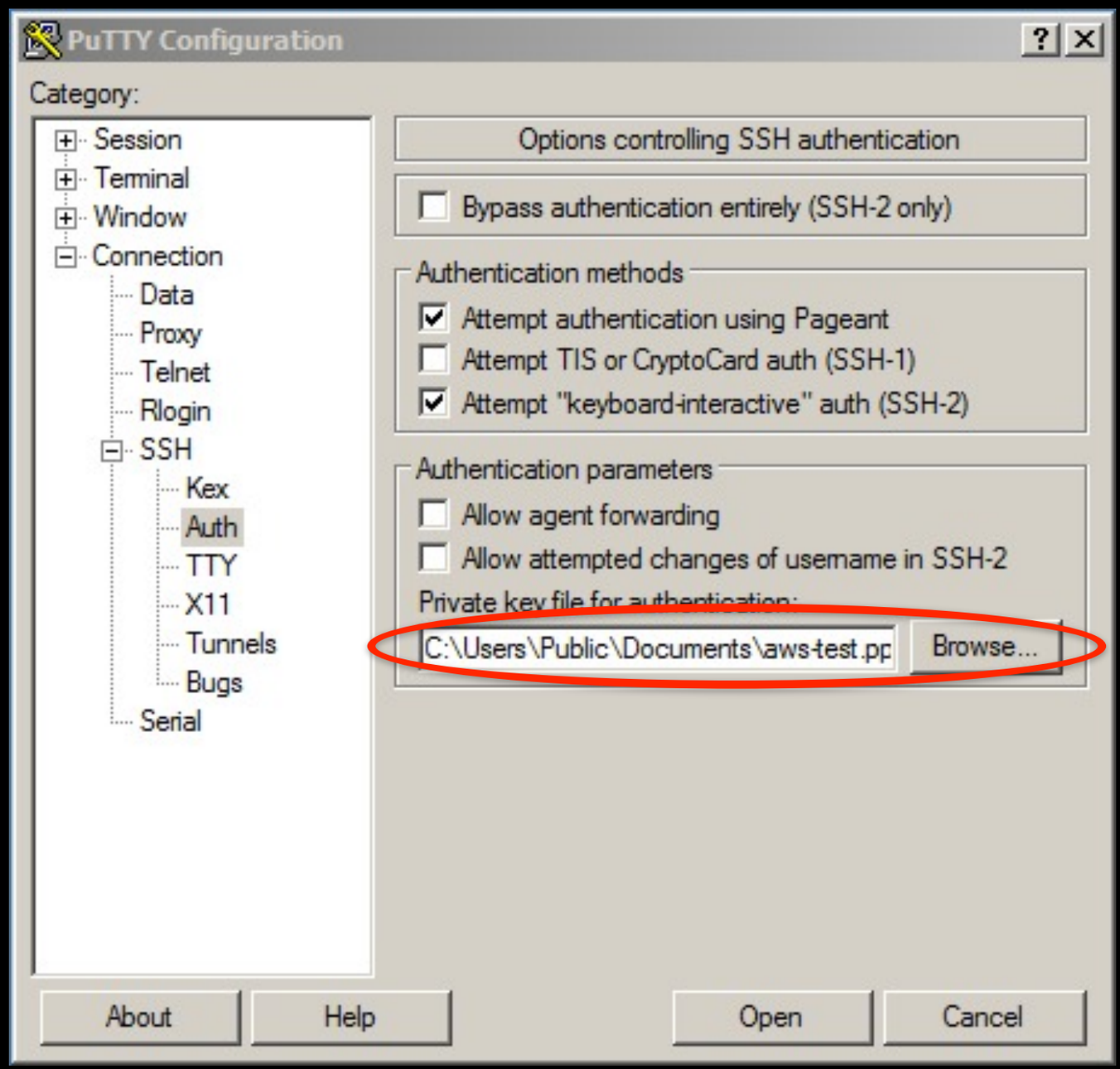
MD5 checksums for all the above files

MD5sums:	md5sums	(or by FTP)	(RSA sig)	(DSA sig)

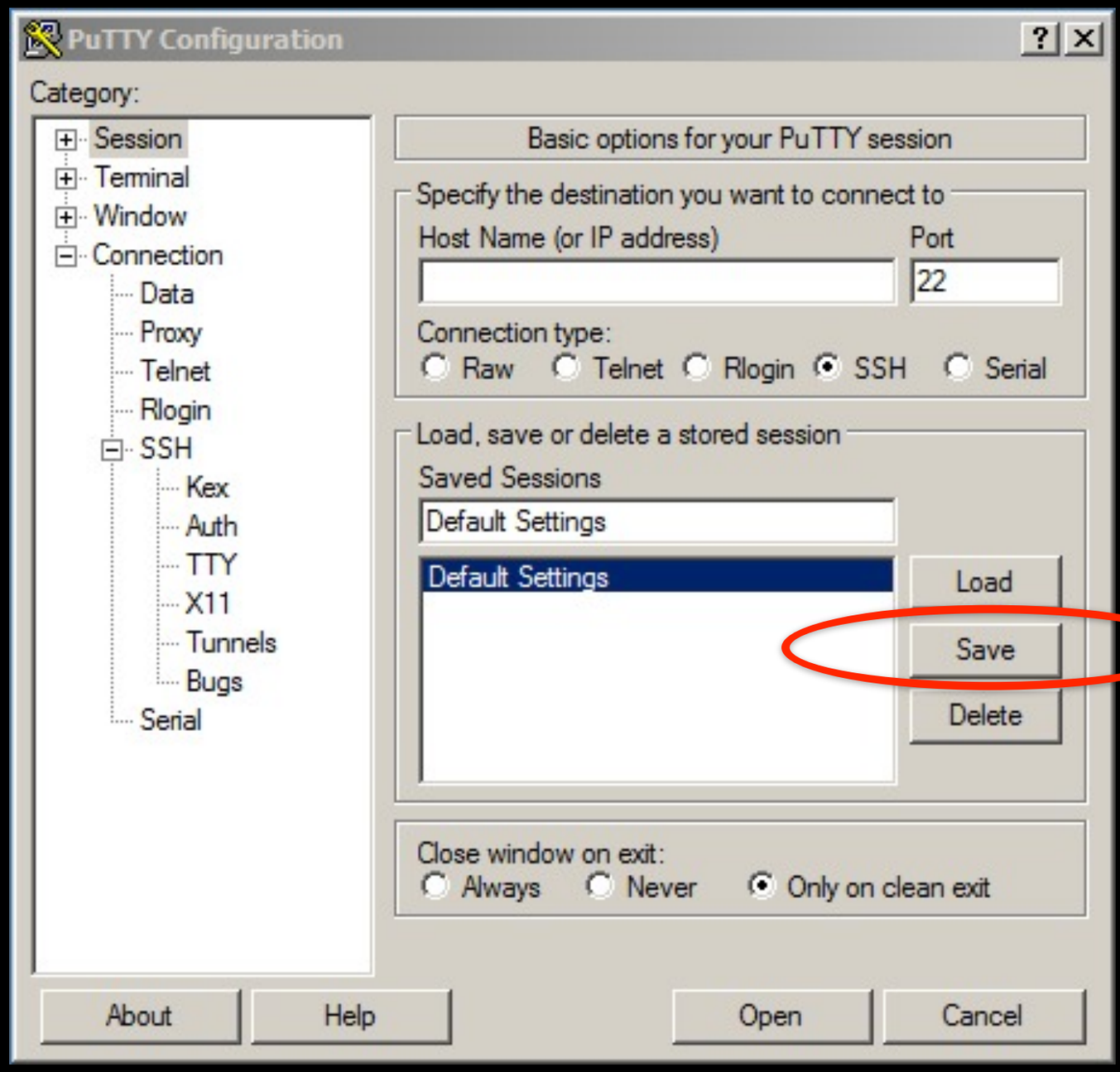
Key Conversion puttygen.exe



PuTTY Configuration



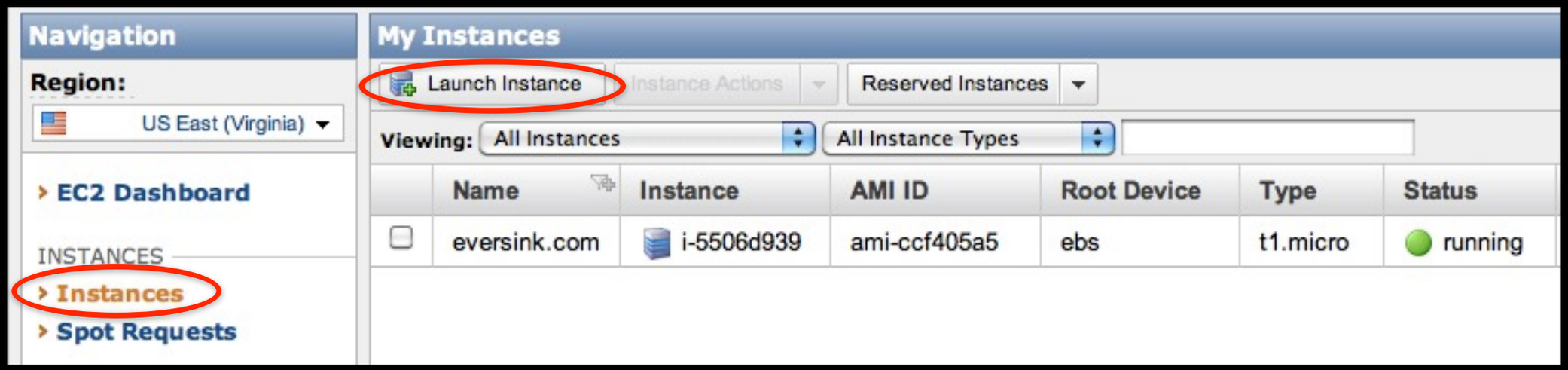
PuTTY Configuration



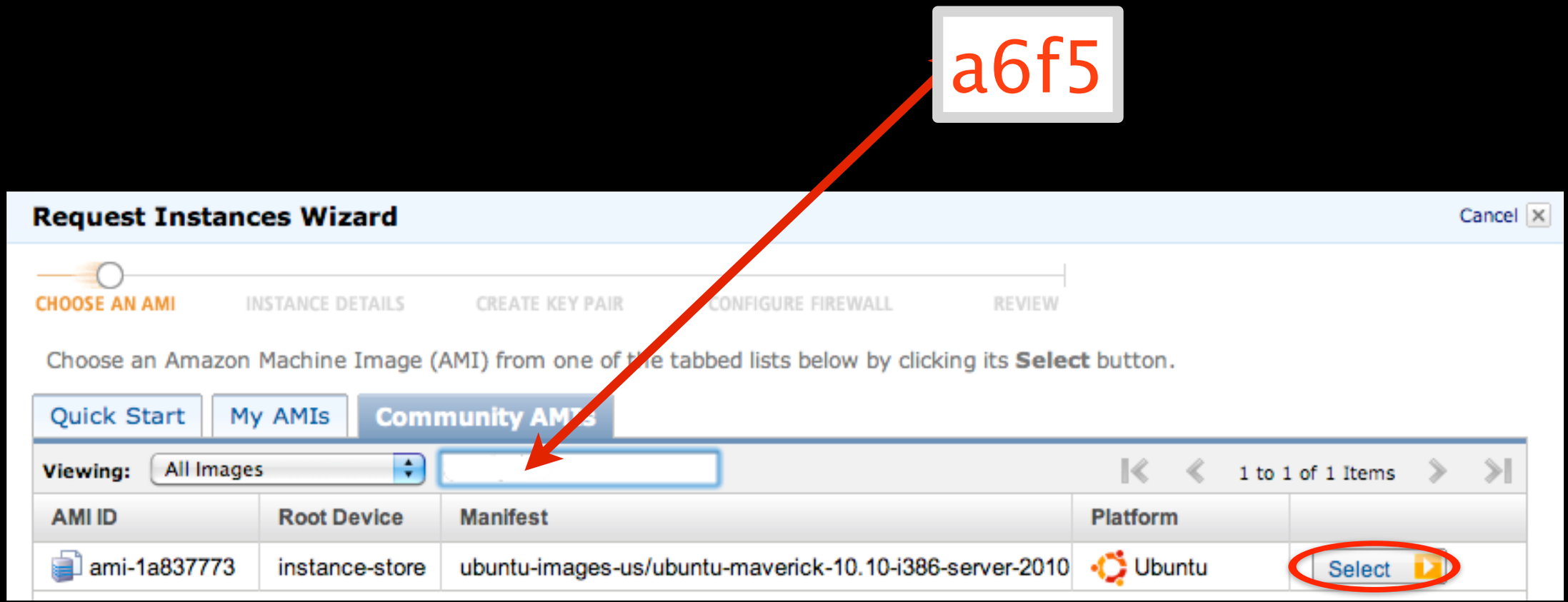
Database Instance

- Ubuntu 10.10 with MySQL on EBS

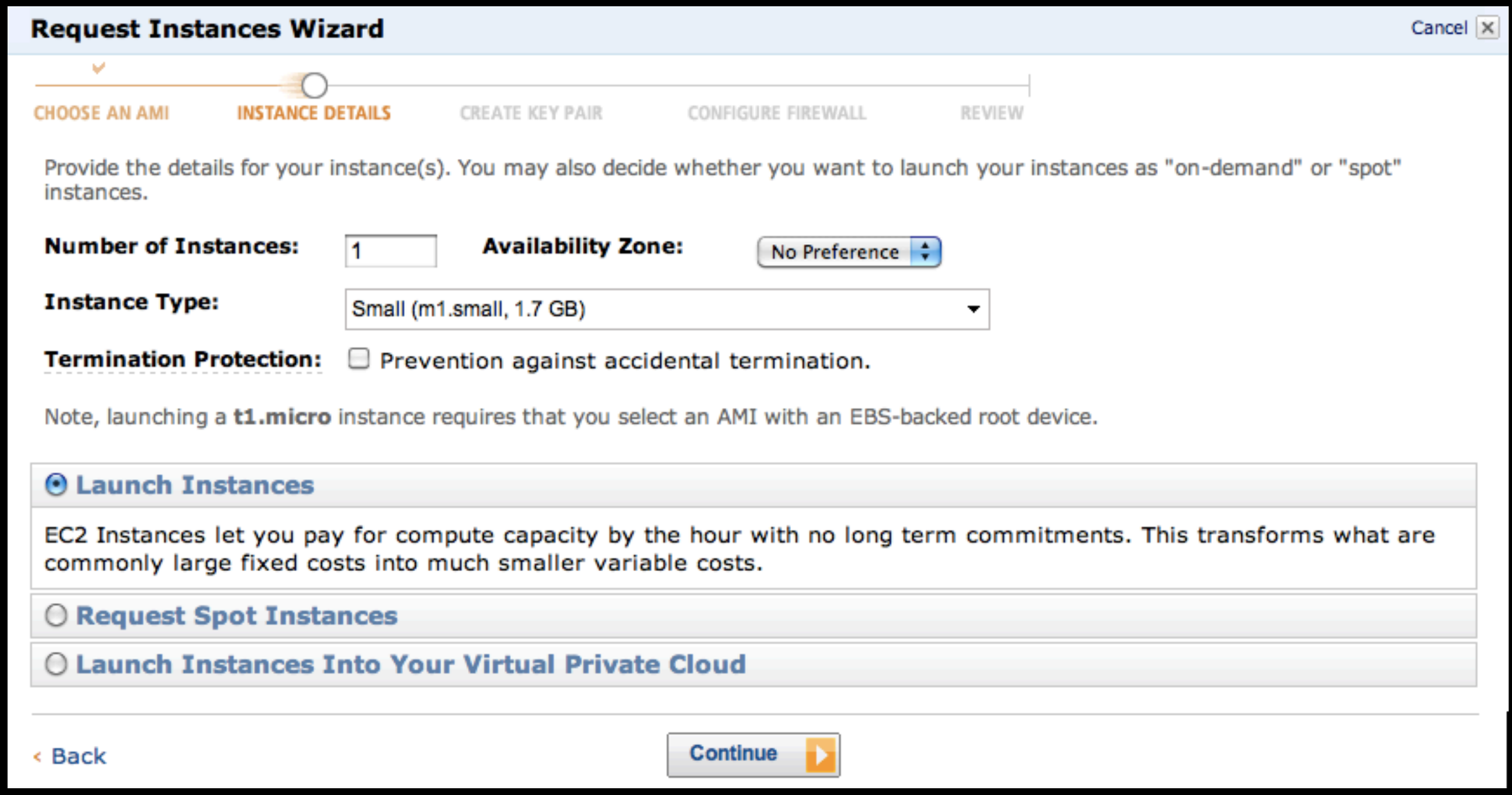
Launch Instance



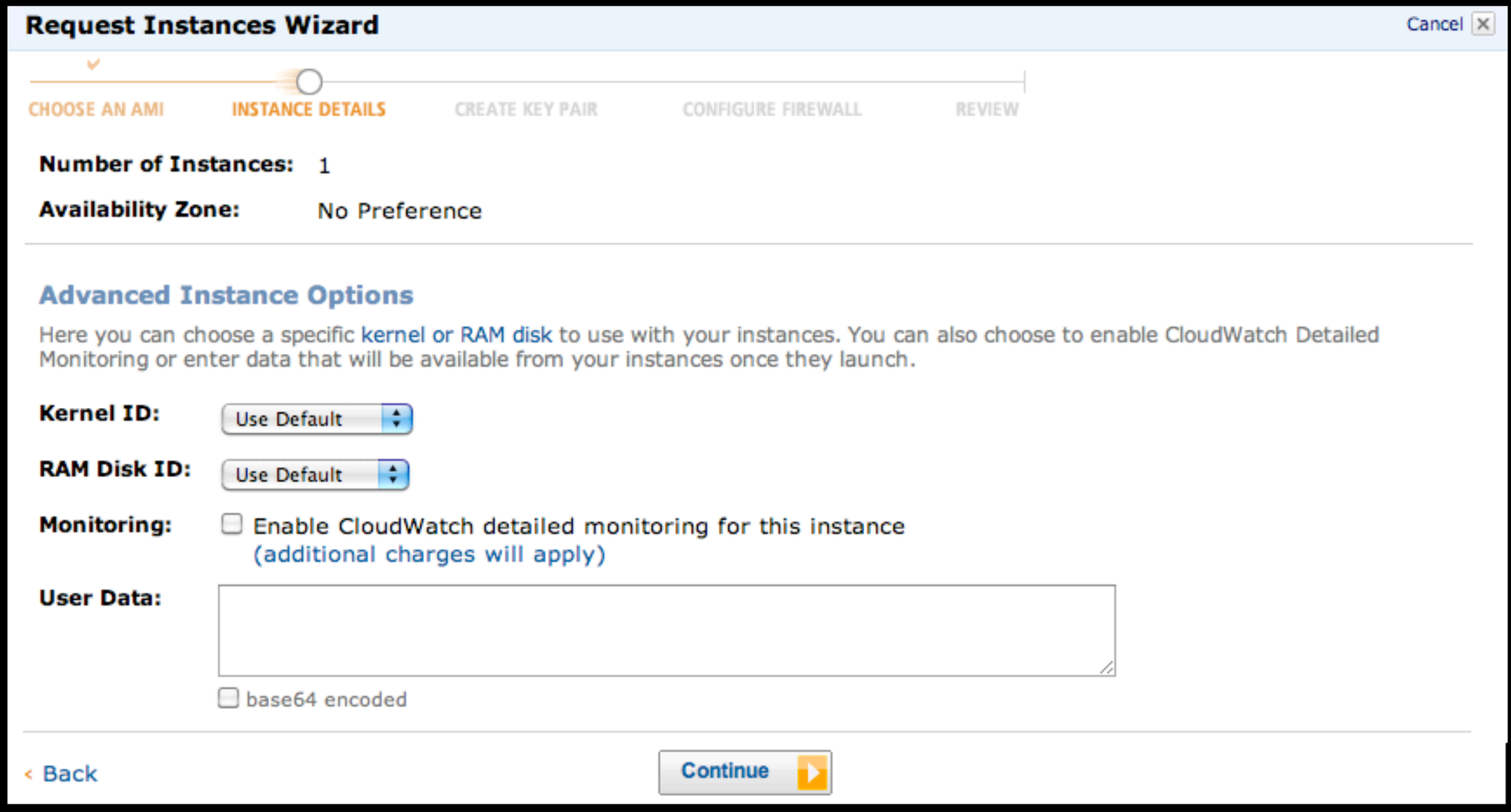
Find the AMI


a6f5
Request Instances Wizard
Cancel
CHOOSE AN AMI
INSTANCE DETAILS
CREATE KEY PAIR
CONFIGURE FIREWALL
REVIEW
Choose an Amazon Machine Image (AMI) from one of the tabbed lists below by clicking its Select button.
Quick Start
My AMIs
Community AMIs
Viewing:
All Images
1 to 1 of 1 Items
AMI ID
Root Device
Manifest
Platform
ami-1a837773
instance-store
ubuntu-images-us/ubuntu-maverick-10.10-i386-server-2010
Ubuntu
Select

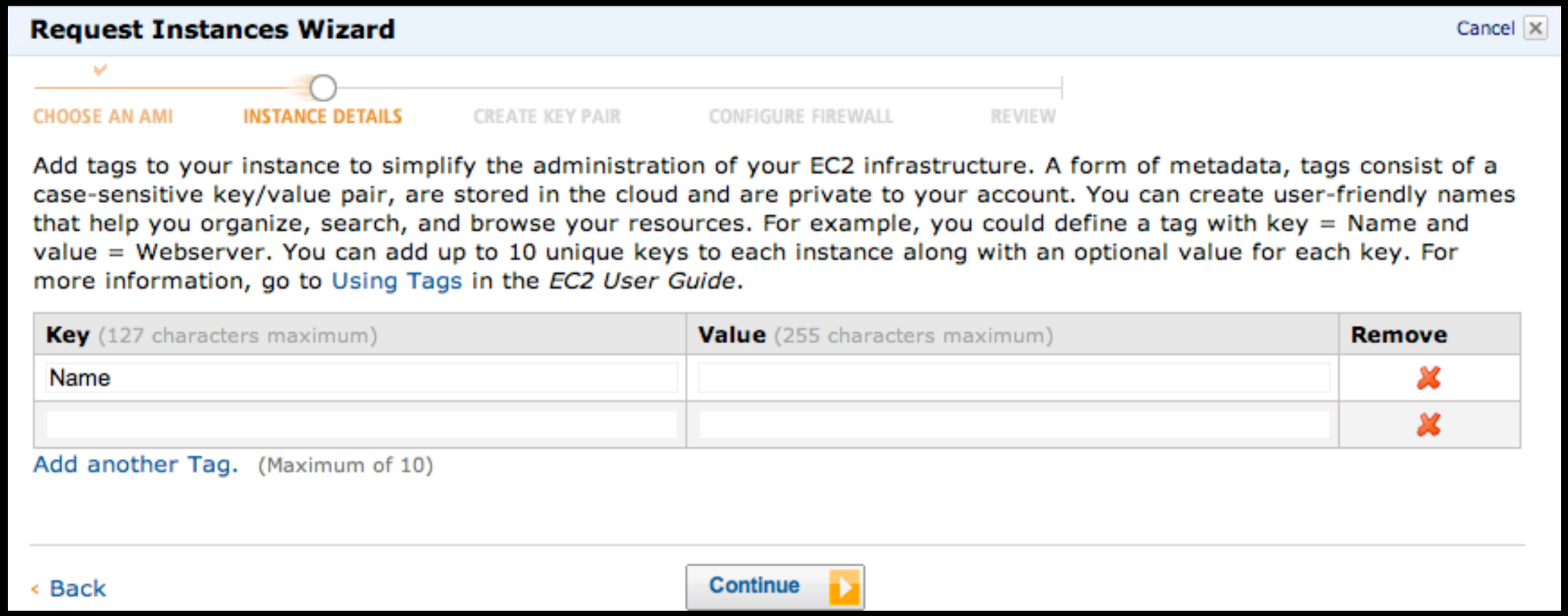
Instance Details



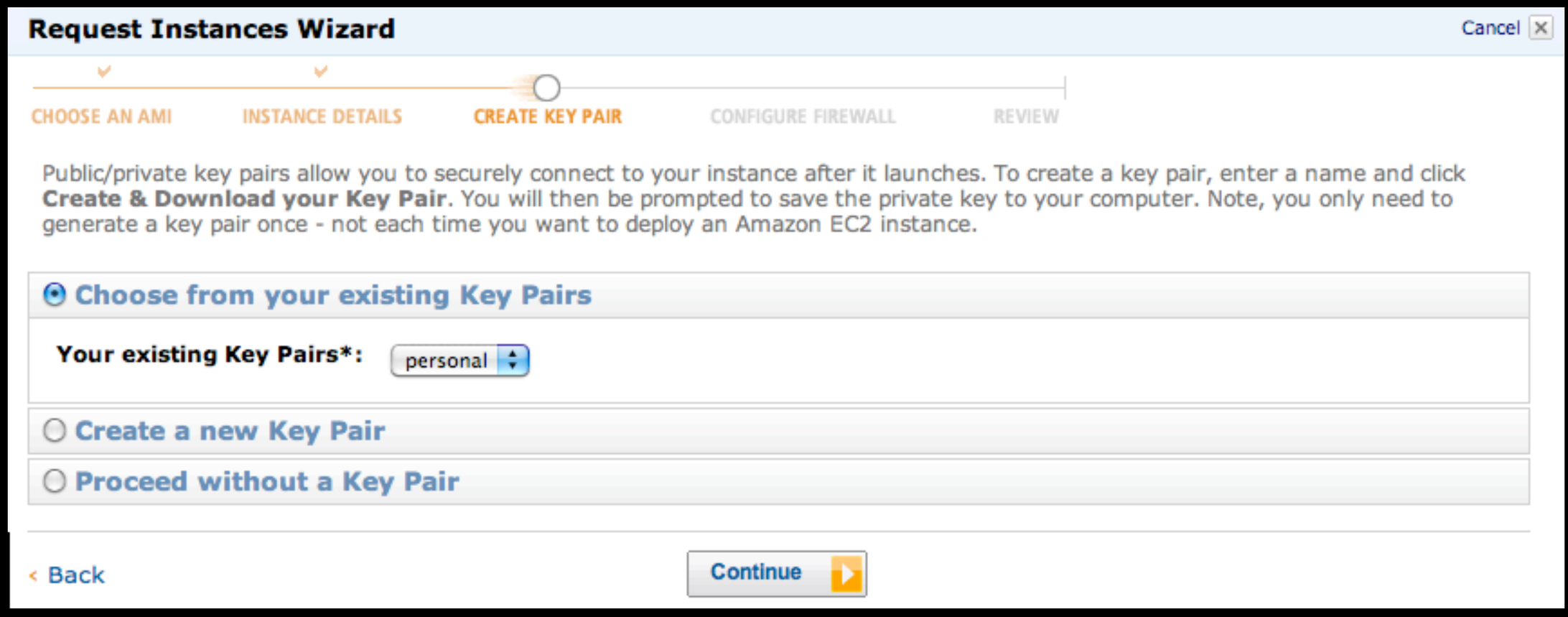
Instance Details



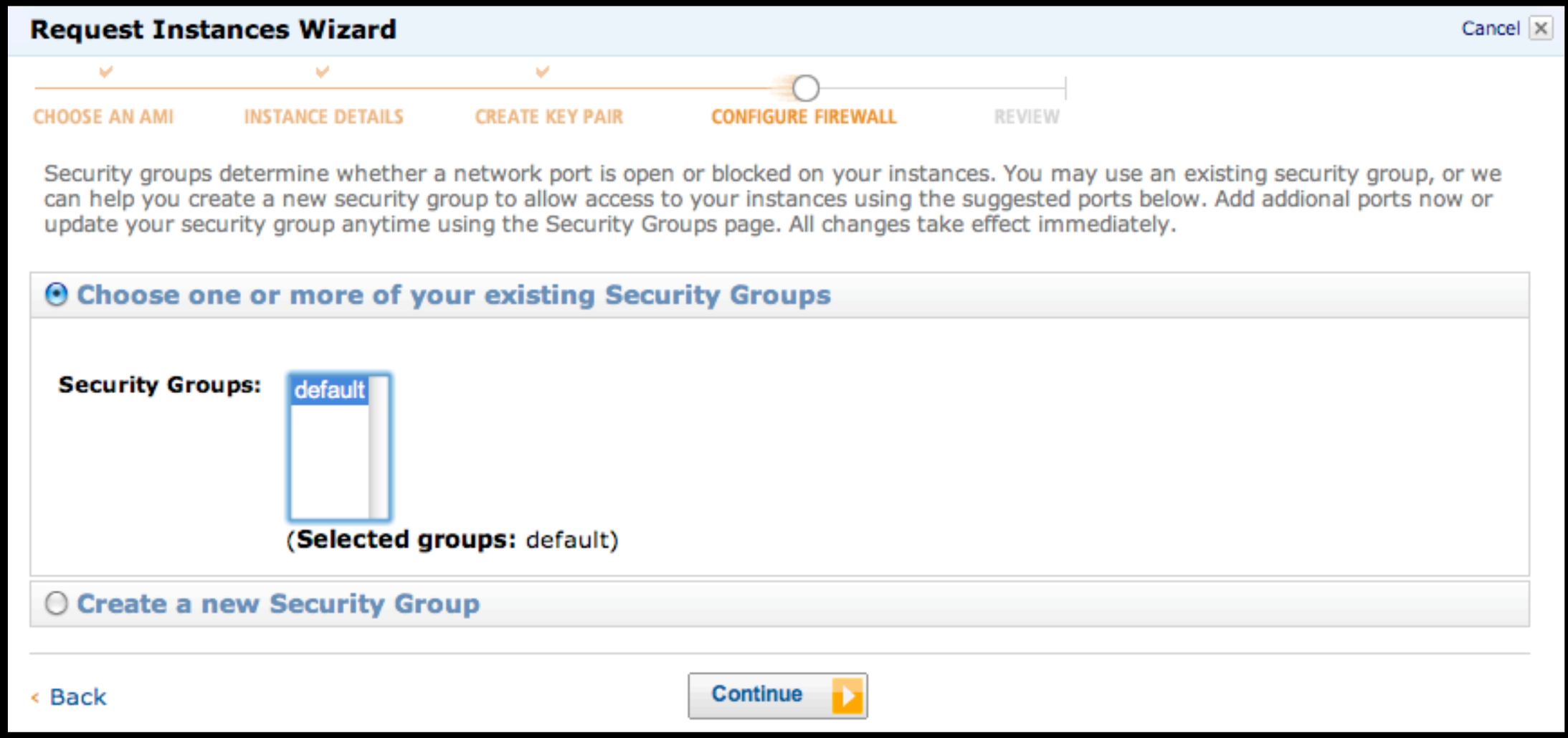
Tags for Identification



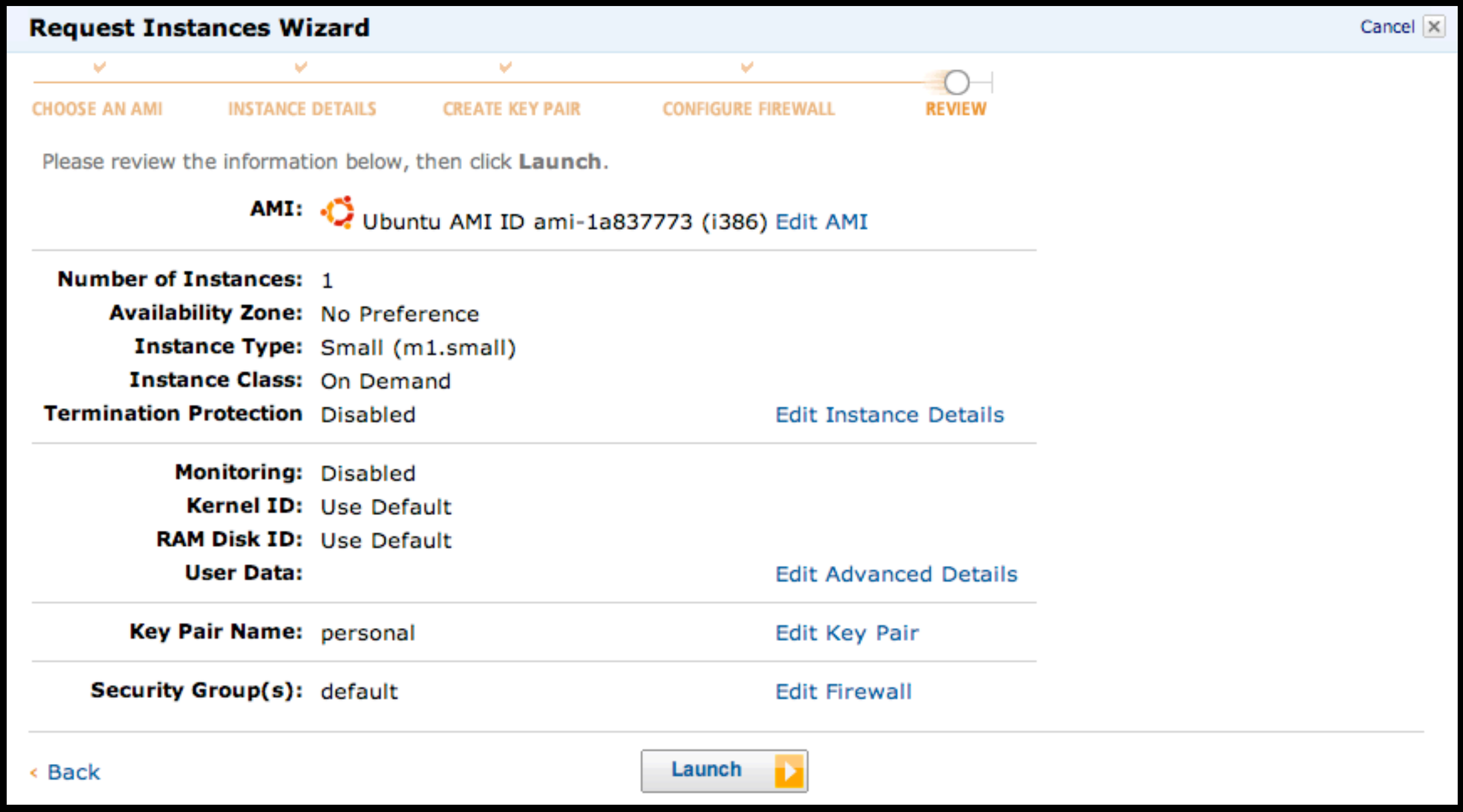
Select Key Pair



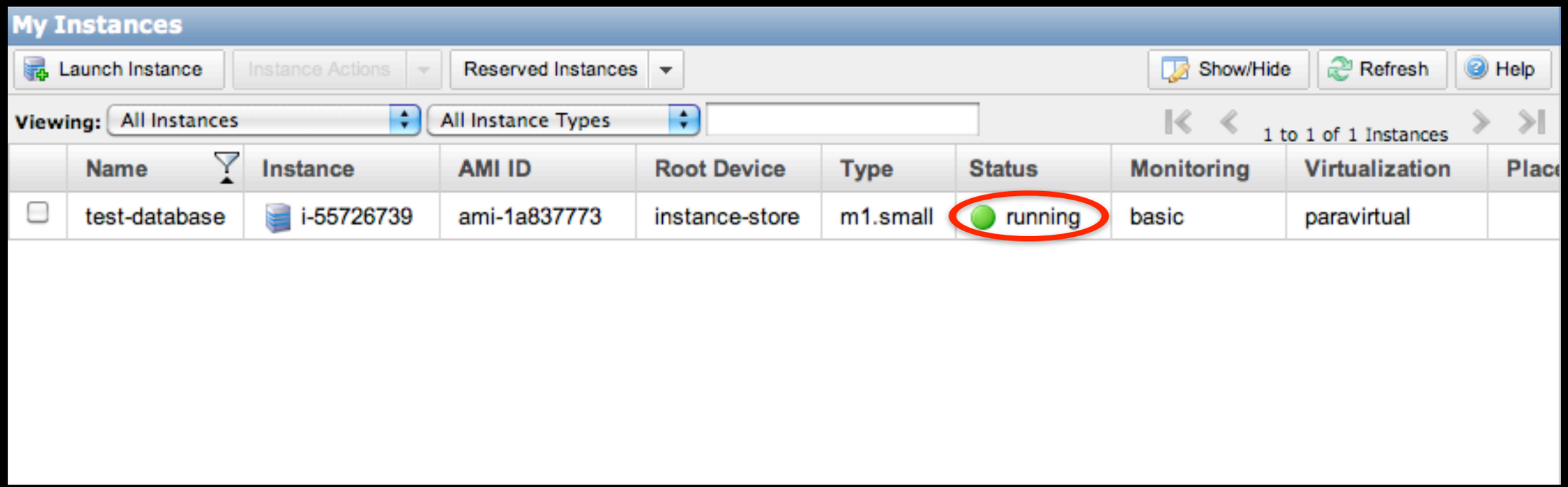
Configure Firewall



Launch


Request Instances Wizard
Cancel
CHOOSE AN AMI
INSTANCE DETAILS
CREATE KEY PAIR
CONFIGURE FIREWALL
REVIEW
Please review the information below, then click Launch.
AMI: Ubuntu AMI ID ami-1a837773 (i386) Edit AMI
Number of Instances: 1
Availability Zone: No Preference
Instance Type: Small (m1.small)
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: personal
Edit Key Pair
Security Group(s): default
Edit Firewall
Back
Launch

Wait until it is running



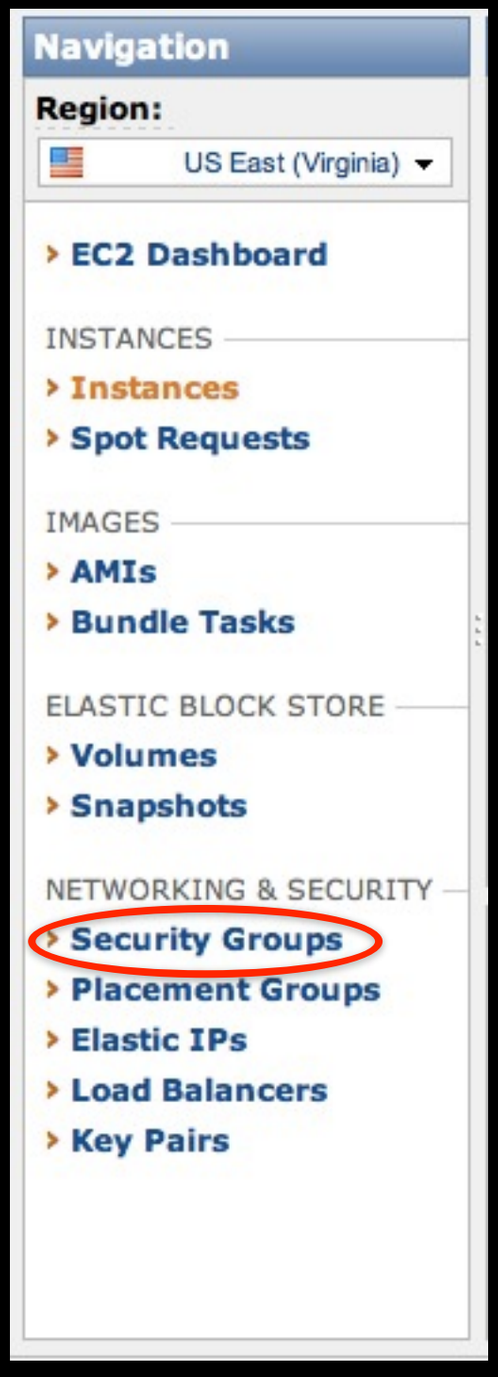
Get DNS name

EC2 Instance: i-55726739

Description | Monitoring | Tags

AMI ID:	ami-1a837773	**Zone:**	us-east-1c
Security Groups:	default	**Type:**	m1.small
Status:	running	**Owner:**	479107200387
VPC ID:	-	**Subnet ID:**	-
Source/Dest. Check:		**Virtualization:**	paravirtual
Placement Group:		**Reservation:**	r-24c25249
RAM Disk ID:	-	**Platform:**	-
Key Pair Name:	personal	**Kernel ID:**	aki-407d9529
Monitoring:	basic	**AMI Launch Index:**	0
Elastic IP:	-	**Root Device:**	-
Root Device Type:	instance-store		
Block Devices:	N/A - Instance Store		
Lifecycle:	normal		
Public DNS:	ec2-50-17-22-1.compute-1.amazonaws.com		
Private DNS:	ip-10-245-197-221.ec2.internal		
Private IP Address:	10.245.197.221		
Launch Time:	2011-03-08 13:58 CST		
State Transition Reason:			
Termination Protection:	Disabled		

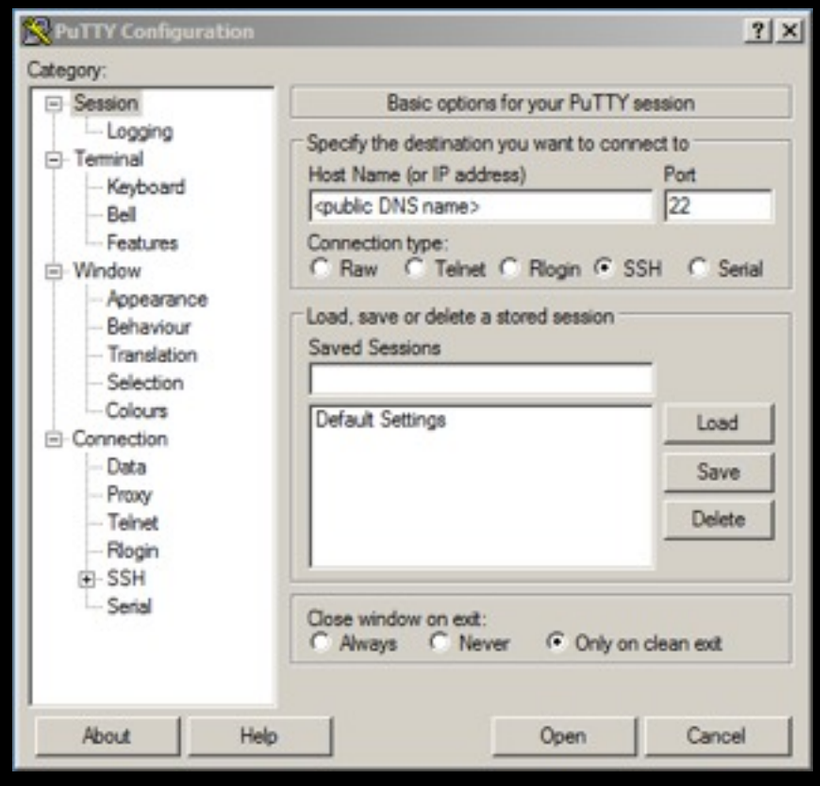
Configure Firewall



Connection Method	Protocol	From Port	To Port	Source (IP or group)	Actions
All	tcp	0	65535	default group	Remove
All	udp	0	65535	default group	Remove
SSH	tcp	22	22	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

Connect via SSH

- chmod 600 <privatekey>
- ssh -i <privatekey> **ubuntu**@<publicdnsname>



Update the system

- `sudo aptitude update`
- `sudo aptitude -y safe-upgrade`

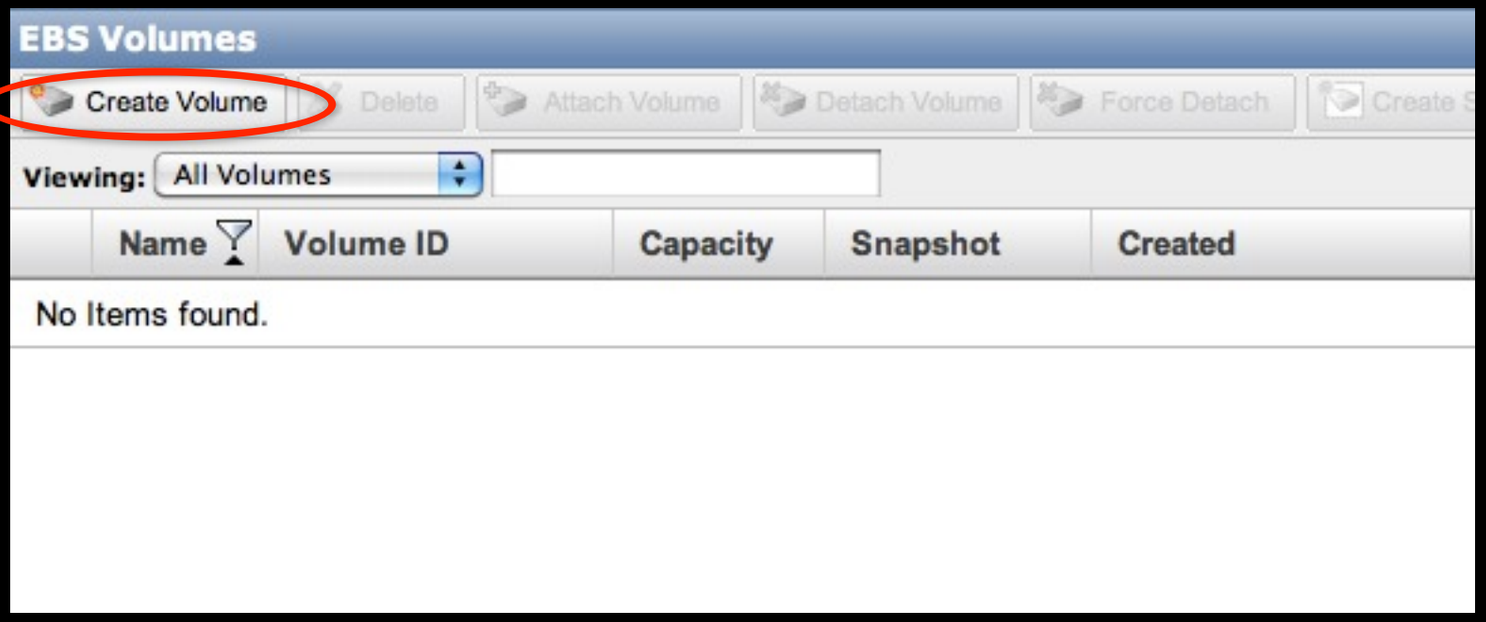
Availability Zone

EC2 Instance: i-55726739

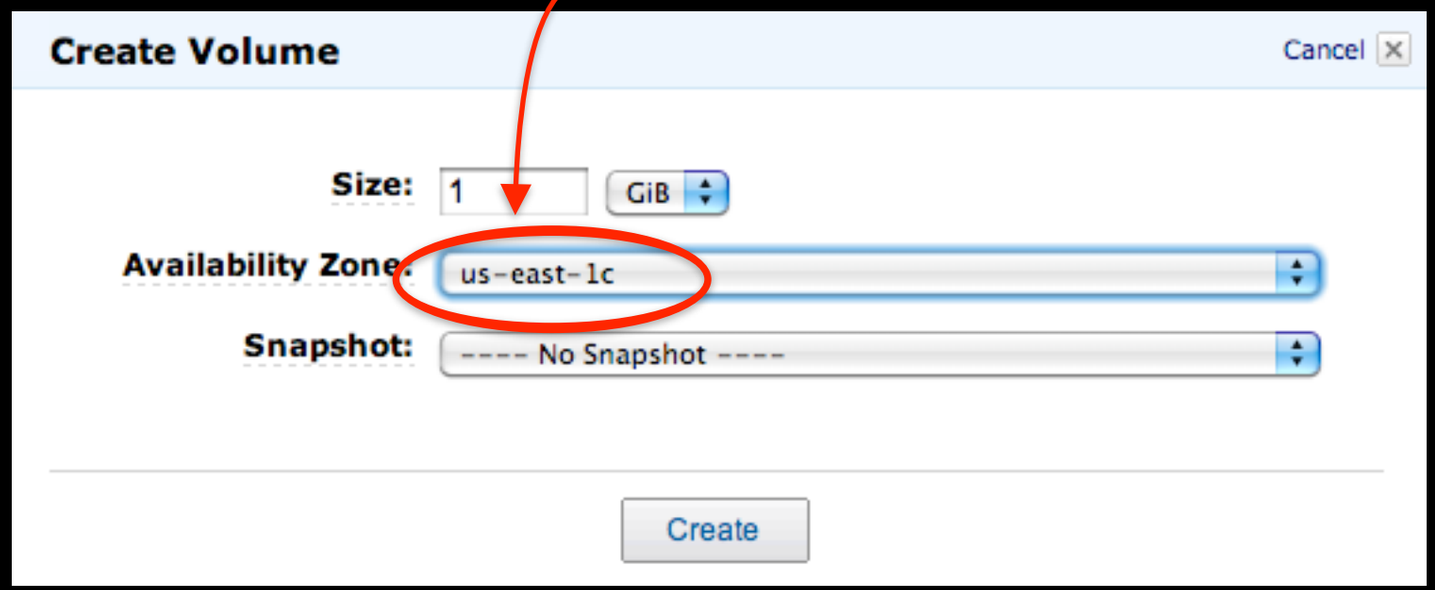
Description | Monitoring | Tags

AMI ID:	ami-1a837773	**Zone:**	us-east-1c
Security Groups:	default	**Type:**	m1.small
Status:	running	**Owner:**	479107200387
VPC ID:	-	**Subnet ID:**	-
Source/Dest. Check:		**Virtualization:**	paravirtual
Placement Group:		**Reservation:**	r-24c25249
RAM Disk ID:	-	**Platform:**	-
Key Pair Name:	personal	**Kernel ID:**	aki-407d9529
Monitoring:	basic	**AMI Launch Index:**	0
Elastic IP:	-	**Root Device:**	-
Root Device Type:	instance-store		
Block Devices:	N/A - Instance Store		
Lifecycle:	normal		
Public DNS:	ec2-50-17-22-1.compute-1.amazonaws.com		
Private DNS:	ip-10-245-197-221.ec2.internal		
Private IP Address:	10.245.197.221		
Launch Time:	2011-03-08 13:58 CST		
State Transition Reason:			
Termination Protection:	Disabled		

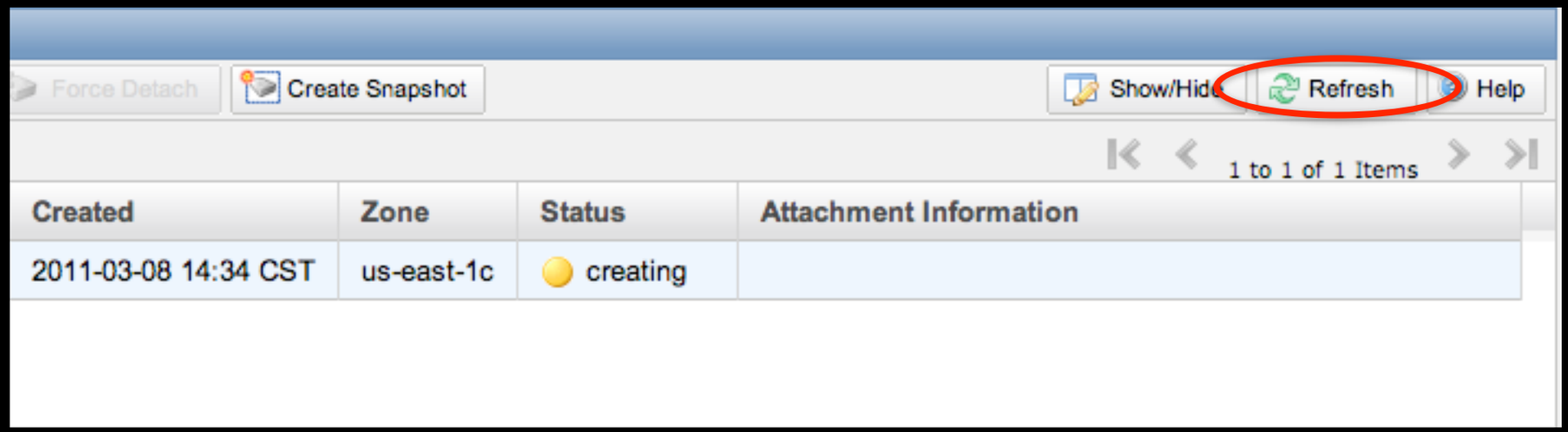
Create EBS Volume



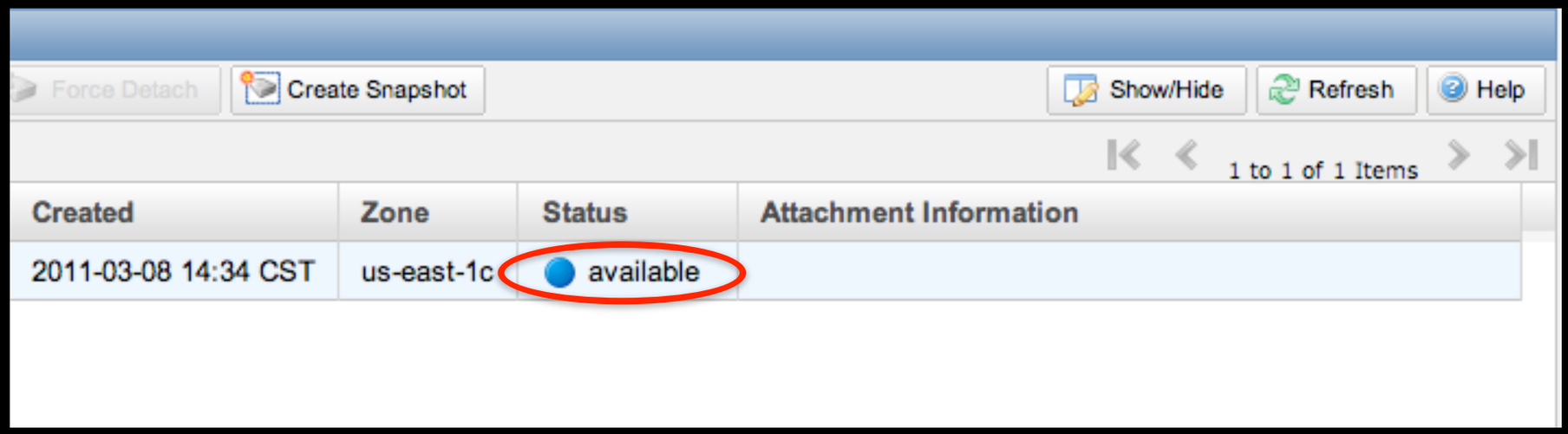
Select the correct Availability Zone



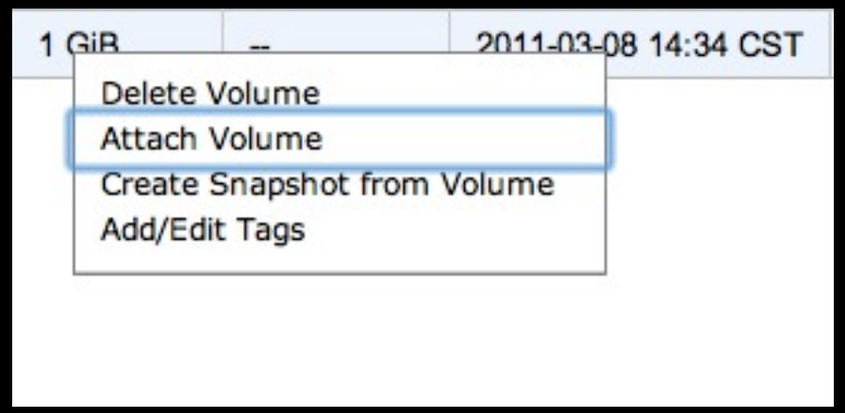
Might need to refresh



Wait for available



Attach Volume



Install MySQL on EBS

- `wget http://bit.ly/pycon-mysql-ebs`
- `bash pycon-mysql-ebs /dev/sdf`
- `mysqlshow --user=root`

Create Application DB

- `wget http://bit.ly/pycon-init-db`
- `bash pycon-init-db`
  `-D clktc`
  `-l http://bit.ly/pycon-db-dump`

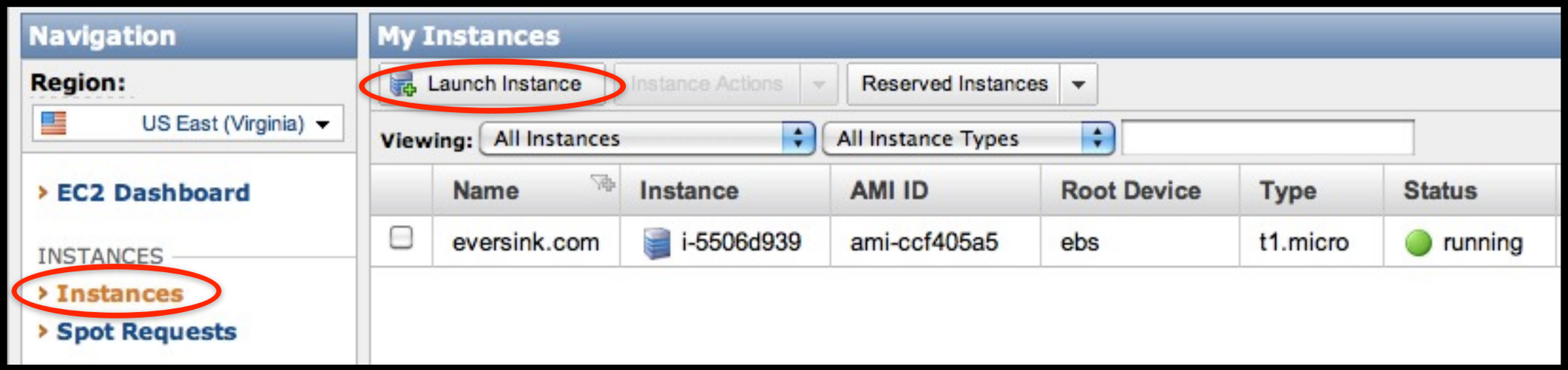
Save DB output

- `-H ec2-50-17-77-46.compute-1.amazonaws.com -D demo -U demo -P b513a23d199c848`

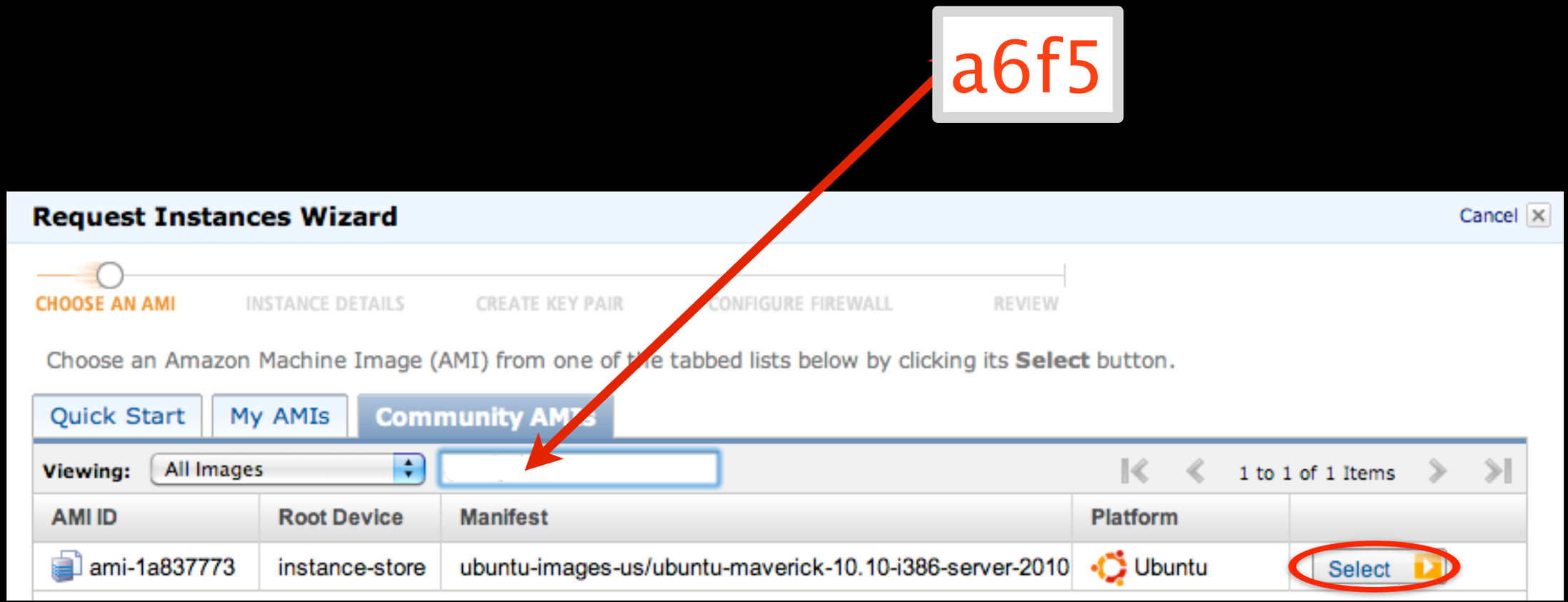
Django Instance

- Ubuntu 10.10 with Apache/mod_wsgi

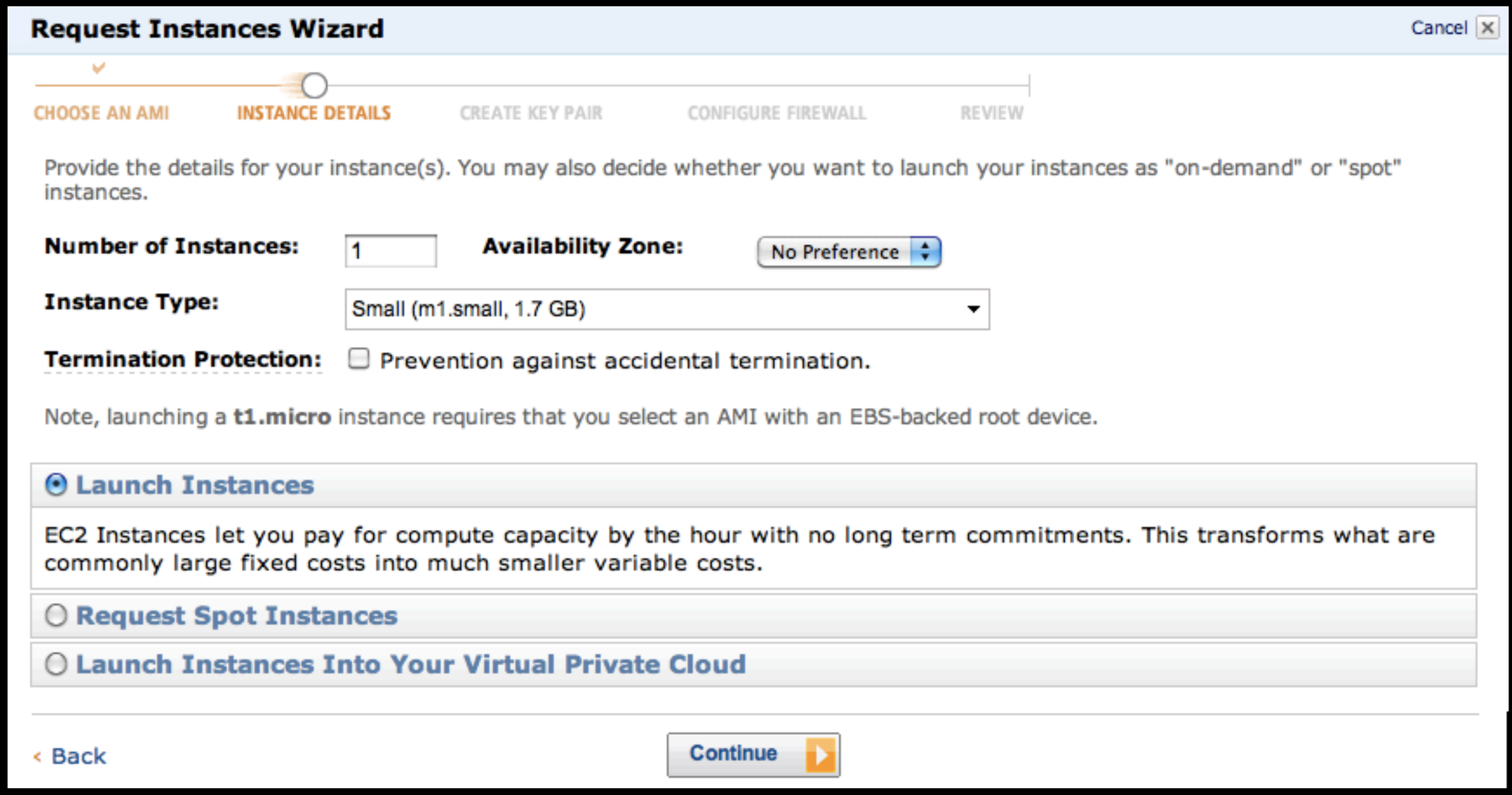
Launch Instance



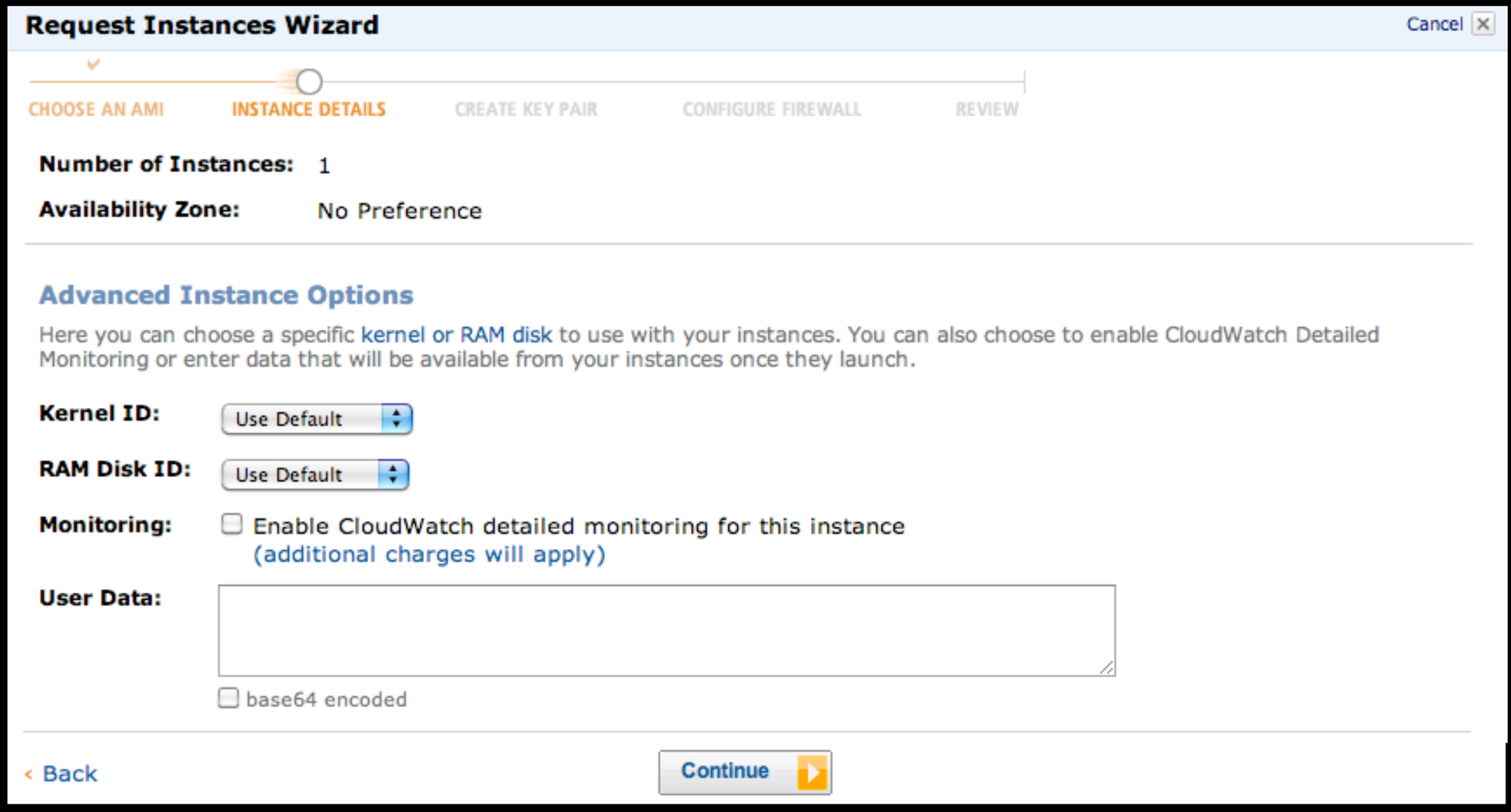
Find the AMI


a6f5
Request Instances Wizard
Cancel
CHOOSE AN AMI
INSTANCE DETAILS
CREATE KEY PAIR
CONFIGURE FIREWALL
REVIEW
Choose an Amazon Machine Image (AMI) from one of the tabbed lists below by clicking its Select button.
Quick Start
My AMIs
Community AMIs
Viewing:
All Images
1 to 1 of 1 Items
AMI ID
Root Device
Manifest
Platform
ami-1a837773
instance-store
ubuntu-images-us/ubuntu-maverick-10.10-i386-server-2010
Ubuntu
Select

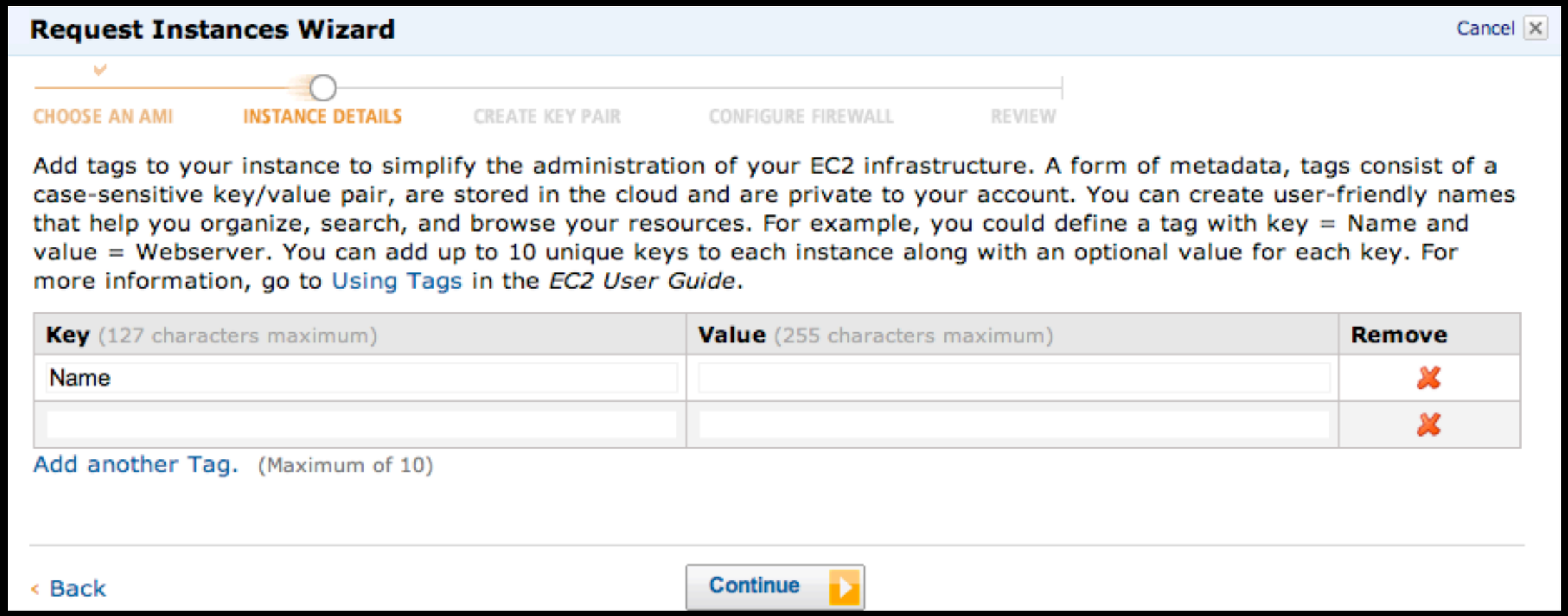
Instance Details



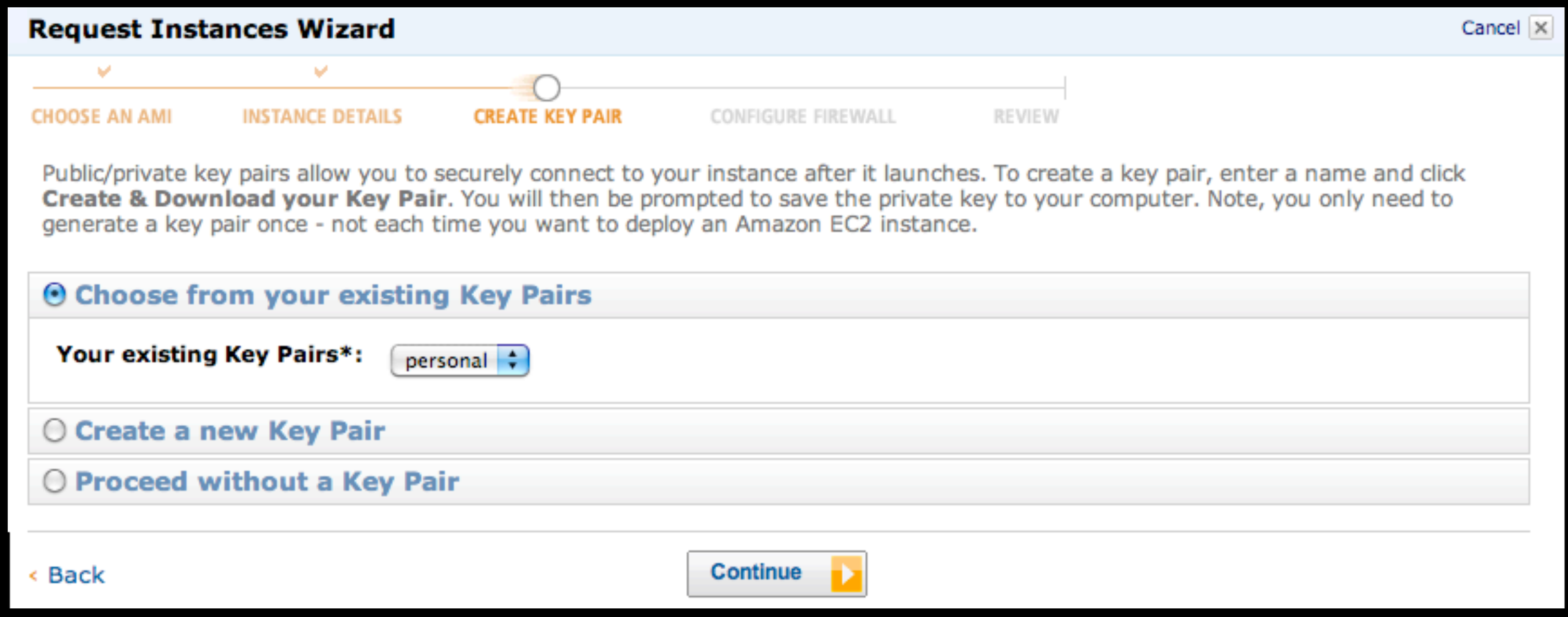
Instance Details



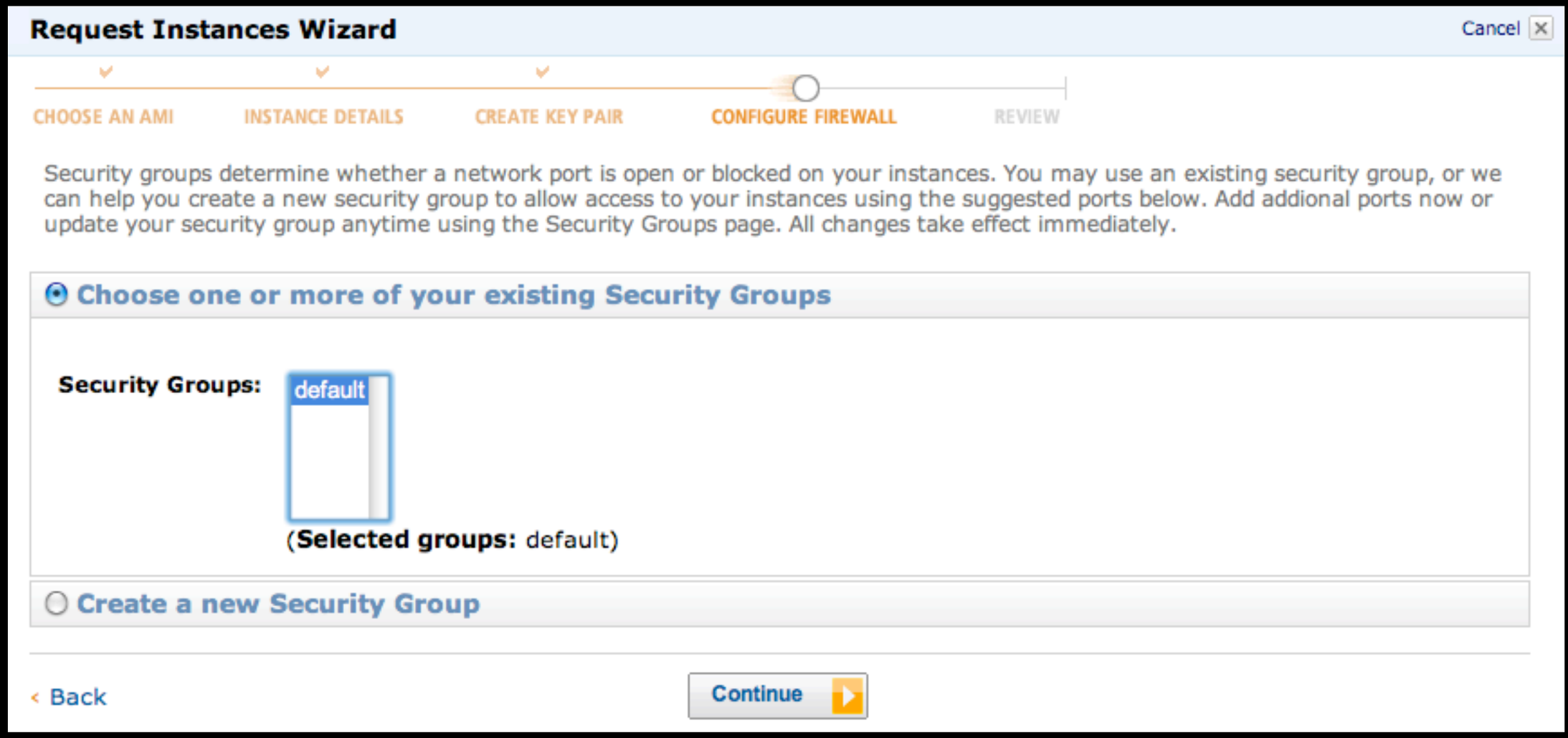
Tags for Identification



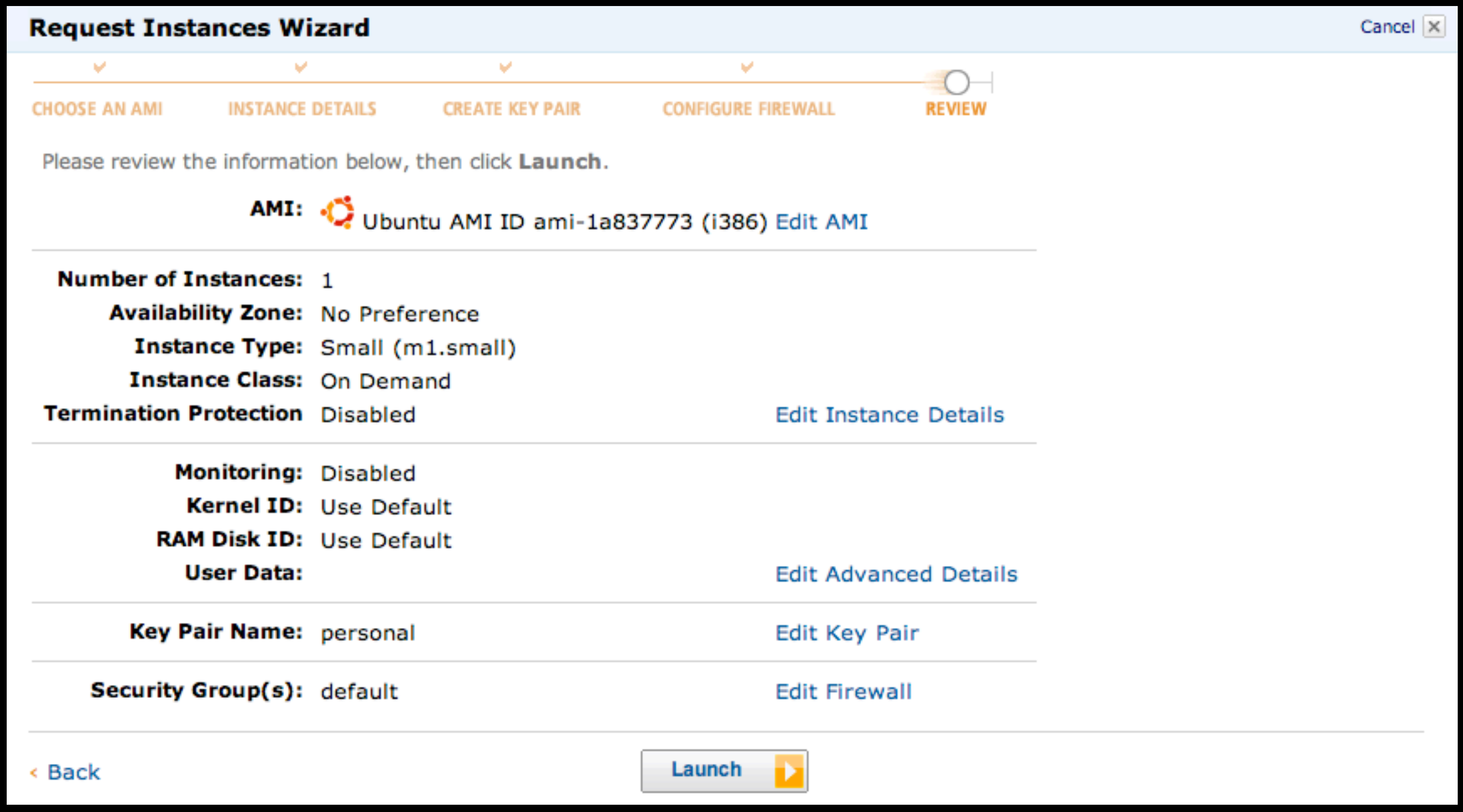
Select Key Pair



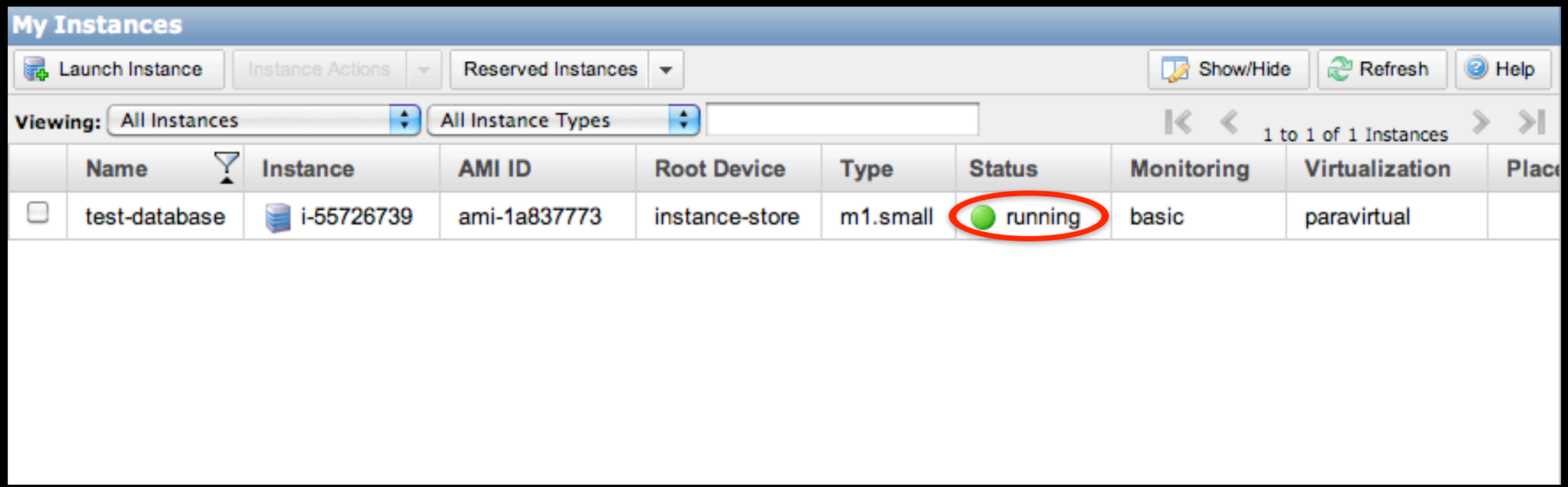
Configure Firewall



Launch


Request Instances Wizard
Cancel
CHOOSE AN AMI
INSTANCE DETAILS
CREATE KEY PAIR
CONFIGURE FIREWALL
REVIEW
Please review the information below, then click Launch.
AMI: Ubuntu AMI ID ami-1a837773 (i386) Edit AMI
Number of Instances: 1
Availability Zone: No Preference
Instance Type: Small (m1.small)
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: personal
Edit Key Pair
Security Group(s): default
Edit Firewall
Back
Launch

Wait until it is running



Get DNS name

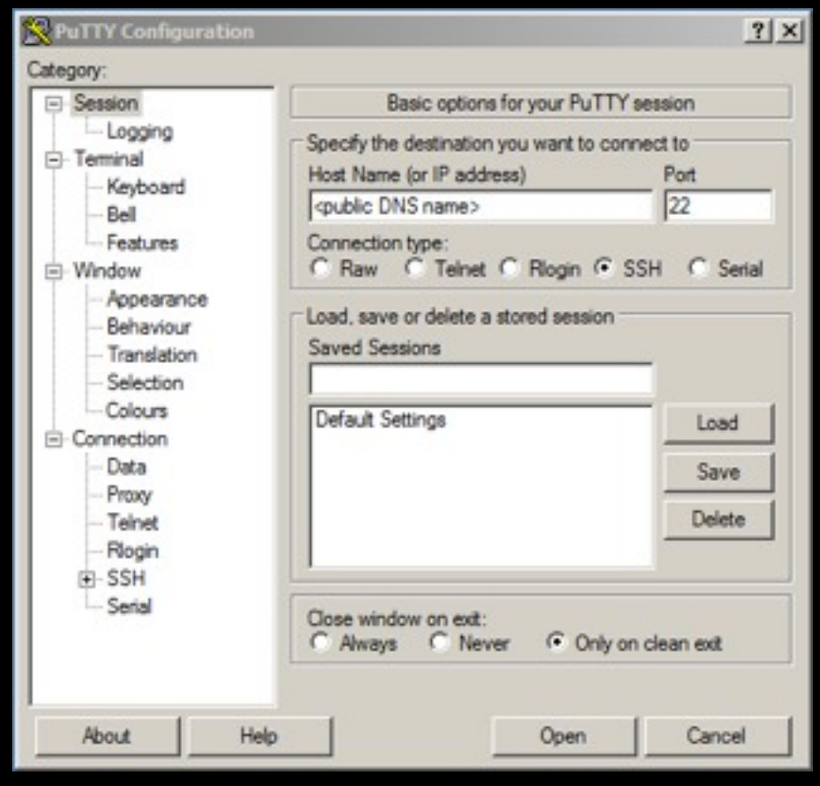
EC2 Instance: i-55726739

Description | Monitoring | Tags

AMI ID:	ami-1a837773	**Zone:**	us-east-1c
Security Groups:	default	**Type:**	m1.small
Status:	running	**Owner:**	479107200387
VPC ID:	-	**Subnet ID:**	-
Source/Dest. Check:		**Virtualization:**	paravirtual
Placement Group:		**Reservation:**	r-24c25249
RAM Disk ID:	-	**Platform:**	-
Key Pair Name:	personal	**Kernel ID:**	aki-407d9529
Monitoring:	basic	**AMI Launch Index:**	0
Elastic IP:	-	**Root Device:**	-
Root Device Type:	instance-store		
Block Devices:	N/A - Instance Store		
Lifecycle:	normal		
Public DNS:	ec2-50-17-22-1.compute-1.amazonaws.com		
Private DNS:	ip-10-245-197-221.ec2.internal		
Private IP Address:	10.245.197.221		
Launch Time:	2011-03-08 13:58 CST		
State Transition Reason:			
Termination Protection:	Disabled		

Connect via SSH

- chmod 600 <privatekey>
- ssh -i <privatekey> **ubuntu**@<publicdnsname>



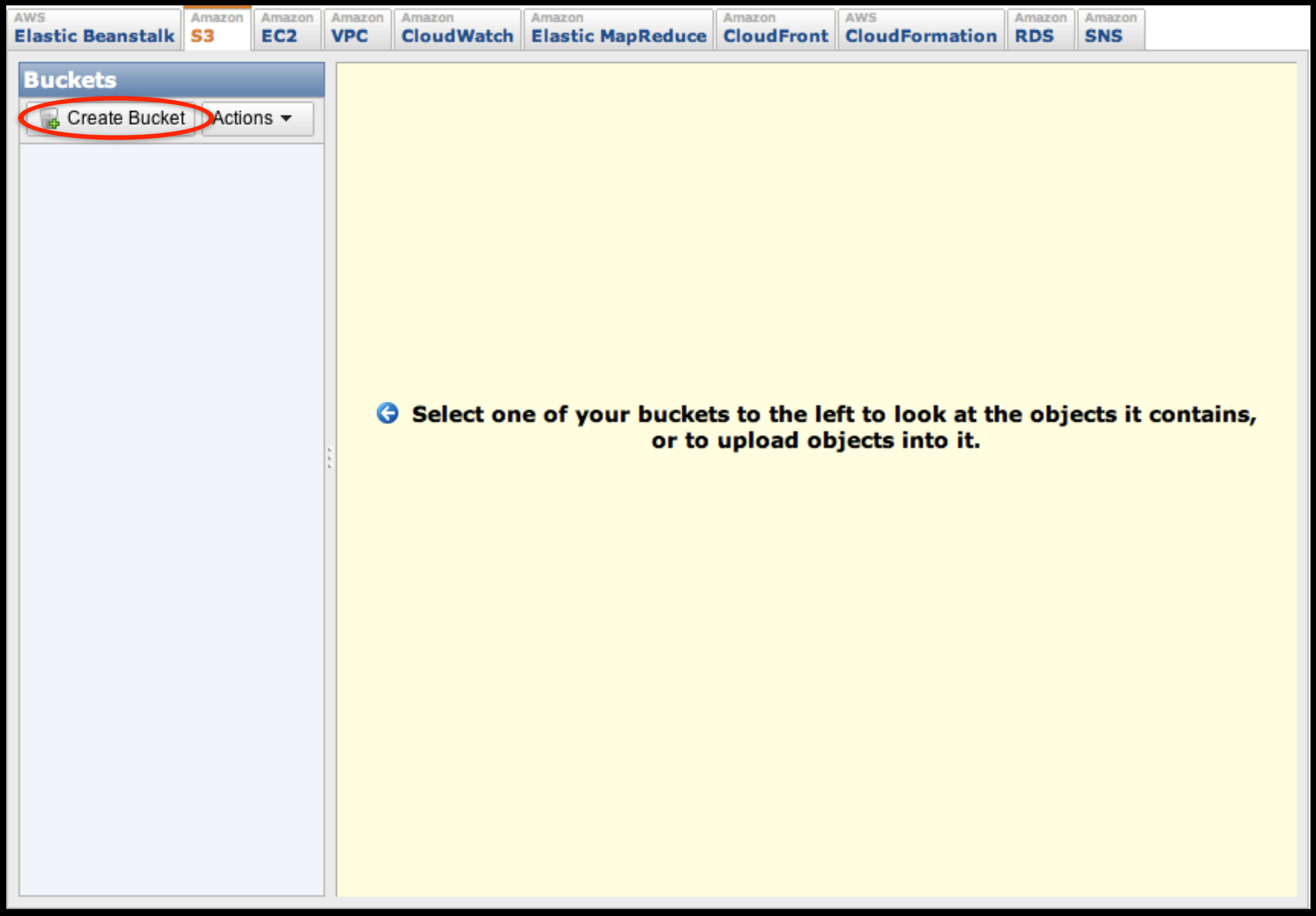
Install Django

- `wget http://bit.ly/pycon-aws-django`
- `sudo bash pycon-aws-django`
  `-n clktc`
  `-d http://bit.ly/pycon-clktc`
  `-s “/s/”`
  `<output from database>`

Exercise 2

- Create S3 bucket
- Create CF distribution
- Upload static assets
- Deploy with new STATIC_URL

Create S3 bucket



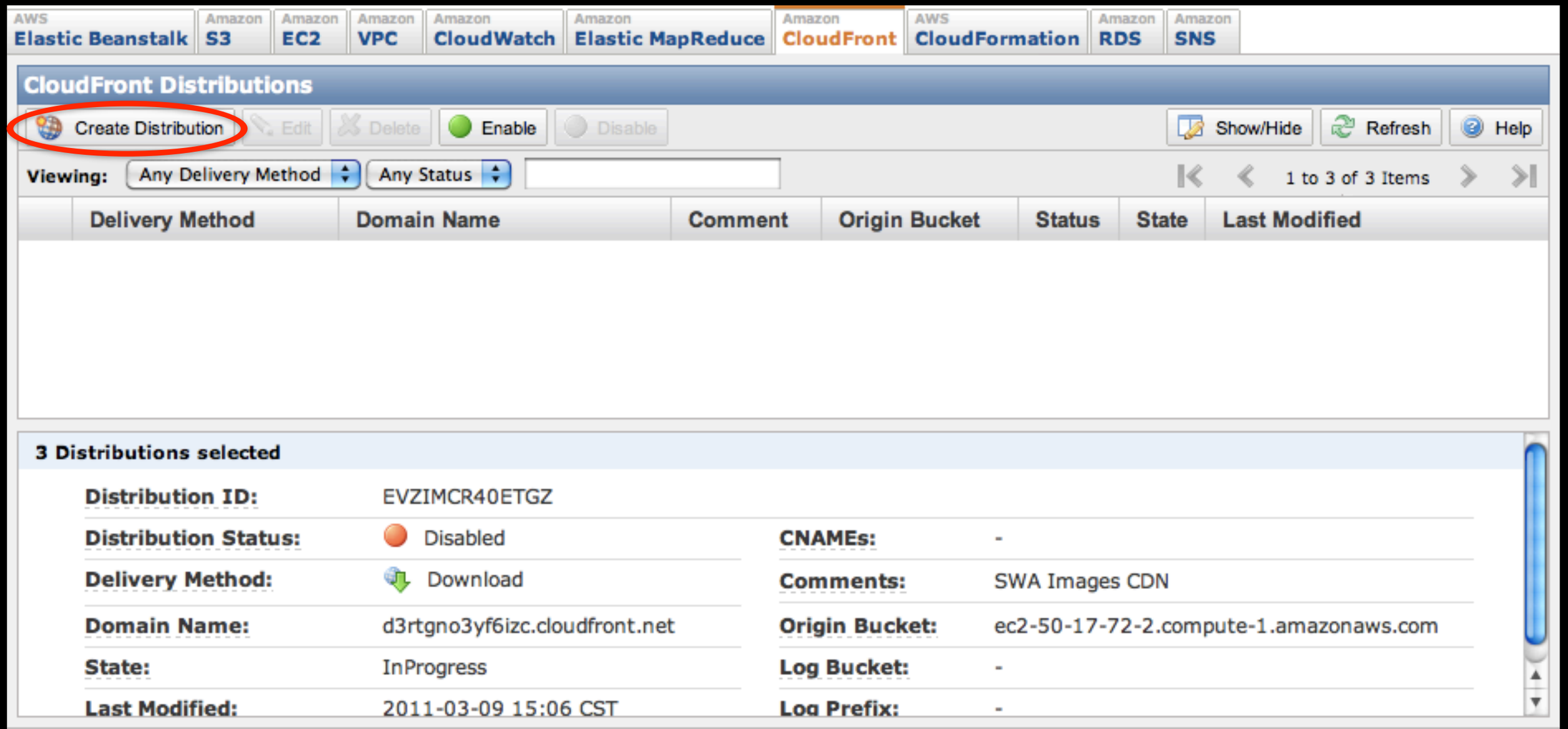
Pick a name

DNS friendly name

DNS friendly name

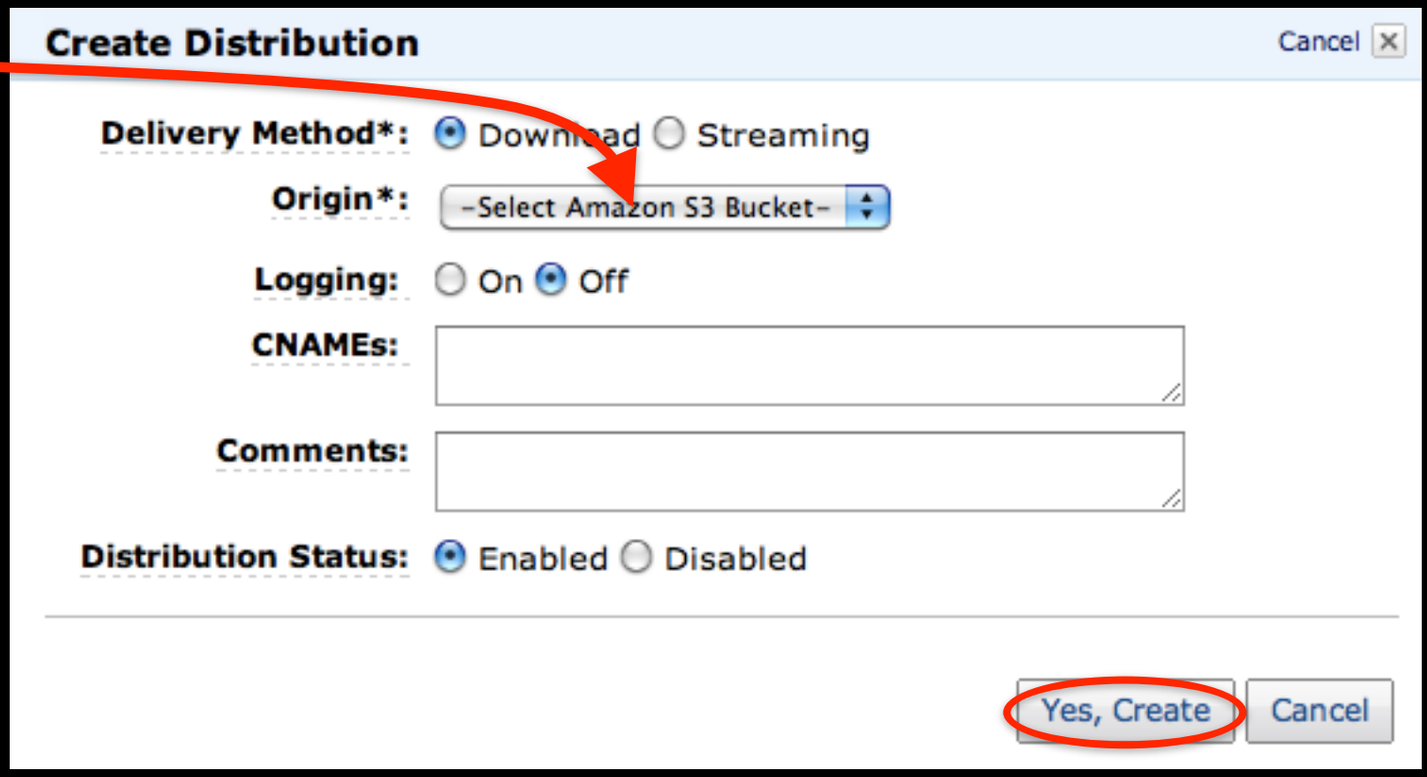
- Start with a letter or number
- Can also contain dashes and periods
  - but
- No dashes next to periods
- No leading or trailing periods
- No underscores

Create CF Distribution

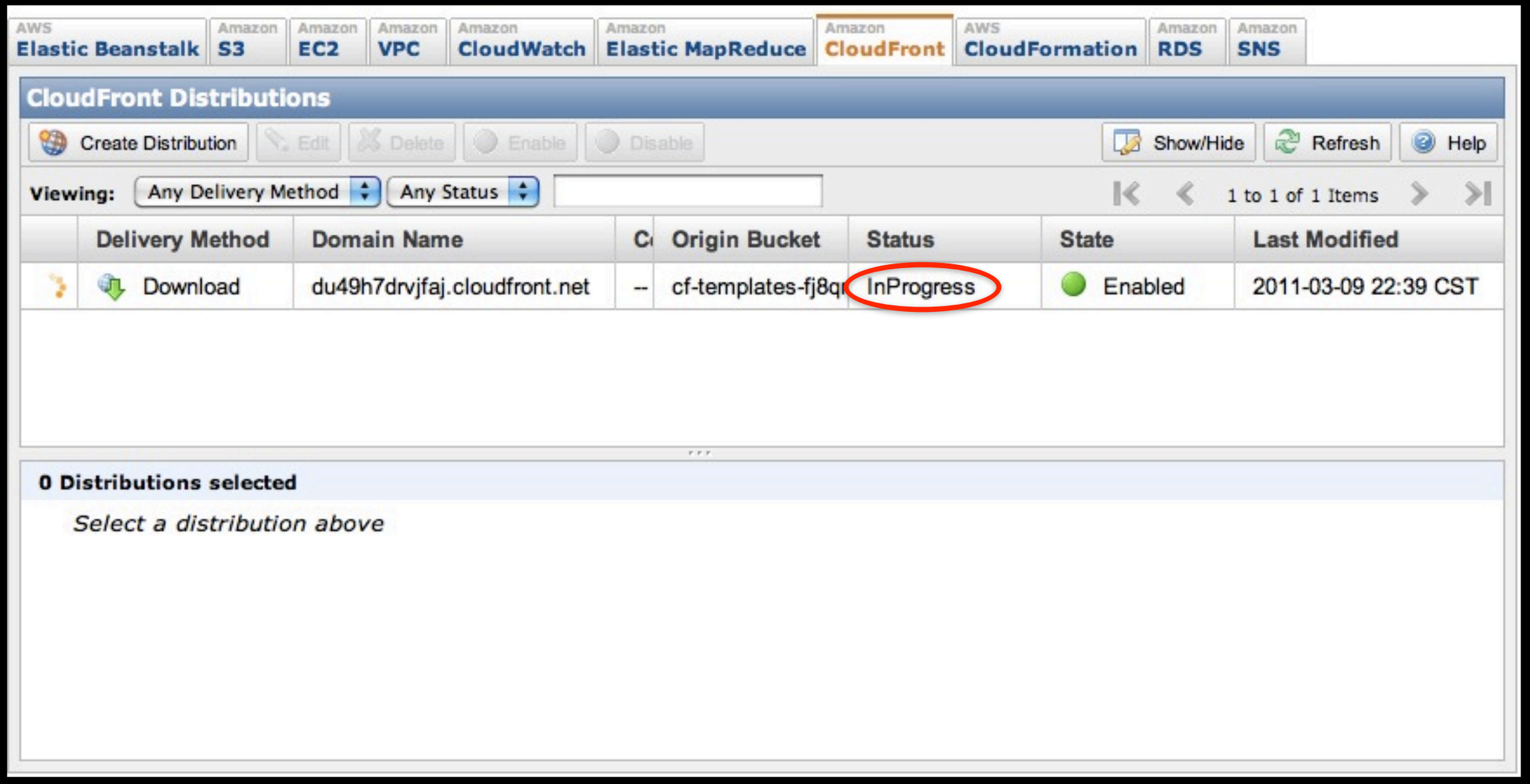


Pick source bucket

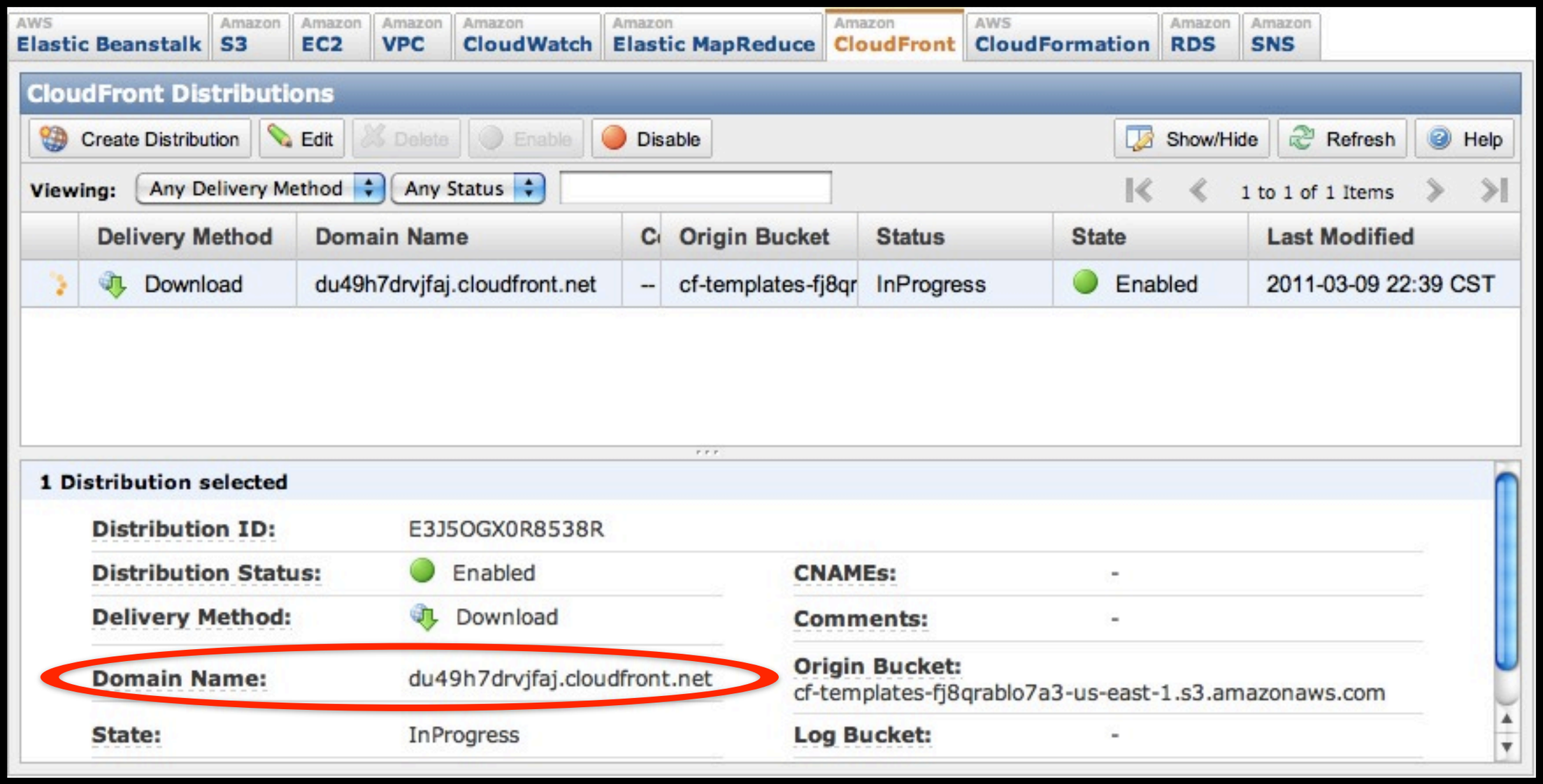
S3 bucket



In Progress


AWS Elastic Beanstalk
Amazon S3
Amazon EC2
Amazon VPC
Amazon CloudWatch
Amazon Elastic MapReduce
Amazon CloudFront
AWS CloudFormation
Amazon RDS
Amazon SNS
CloudFront Distributions
Create Distribution
Edit
Delete
Enable
Disable
Show/Hide
Refresh
Help
Viewing:
Any Delivery Method
Any Status
1 to 1 of 1 Items
Delivery Method
Domain Name
Origin Bucket
Status
State
Last Modified
Download
du49h7drvjfaj.cloudfront.net
--
cf-templates-fj8qr
InProgress
Enabled
2011-03-09 22:39 CST
0 Distributions selected
Select a distribution above

Get Domain Name



Security Credentials

Your Account

Account Activity

View current charges and account activity, itemized by service and by usage type. Previous months' billing statements are also available.

Usage Reports

Download usage reports for each service you are subscribed to. Reports can be customized by specifying usage types, timeframe, service operations, and more.

Security Credentials

Amazon Web Services uses access identifiers to authenticate requests to AWS and to identify the sender of a request. Three types of identifiers are available: (1) AWS Access Key Identifiers, (2) X.509 Certificates, and (3) Key pairs

Personal Information

View and edit personal contact information, such as address and phone number. Set communication preferences for email subscriptions.

Get Access Keys

Access Credentials

There are three types of access credentials used to authenticate your requests to AWS services: (a) access keys, (b) X.509 certificates, and (c) key pairs. Each access credential type is explained below.

Access Keys | X.509 Certificates | Key Pairs

Use access keys to make secure REST or Query protocol requests to any AWS service API. We create one for you when your account is created — see your access key below.

Your Access Keys

Created	Access Key ID	Secret Access Key	Status
November 7, 2010	AKIAIBZ57QE6YLAOWS5A	Show	Active (Make Inactive)

Create a new Access Key

View Your Deleted Access Keys

For your protection, you should never share your secret access keys with anyone. In addition, industry best practice recommends frequent key rotation.

Learn more about Access Keys

Checkpoint

- Browe to public DNS name of last instance

Upload static assets

- `wget http://bit.ly/pycon-to-s3`
- ```
  bash pycon-to-s3
  -b BUCKET_NAME
  -l /home/clktc/clktc/media
  -k ACCESS_KEY -s SECRET_KEY
  ```
  ```

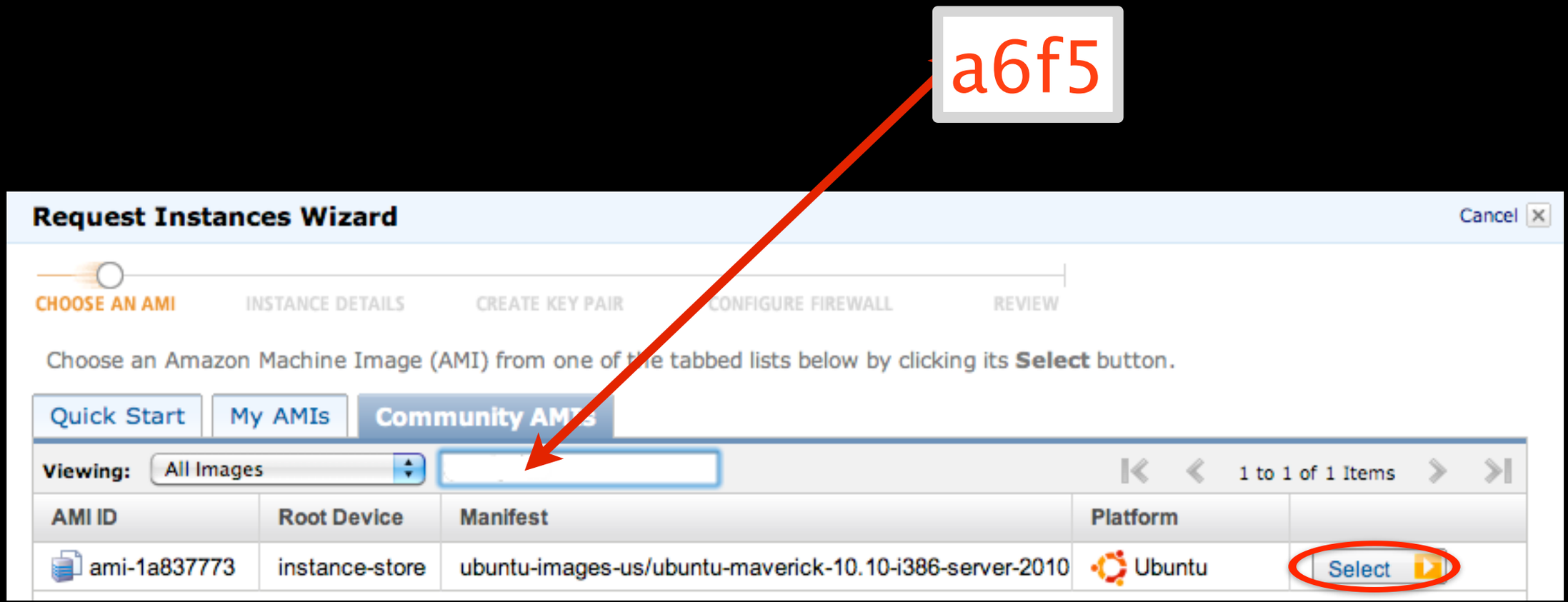
Checkpoint

- Verify everything worked by browsing to
  - http://<CLOUDFRONTDOMAIN>/css/common.css

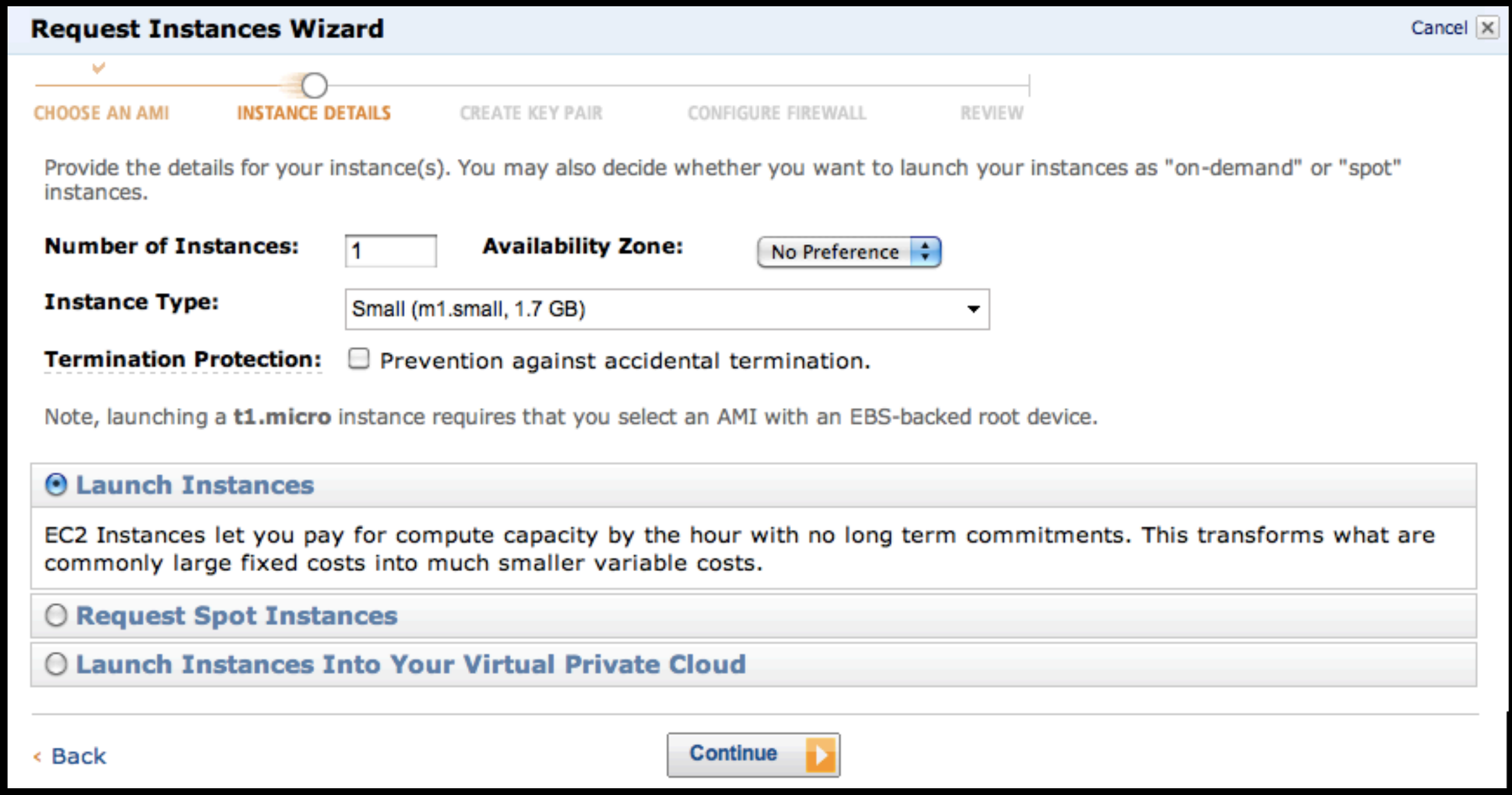
Launch New Instance

- Use user data to run a script on boot
- Automate deployment of the application

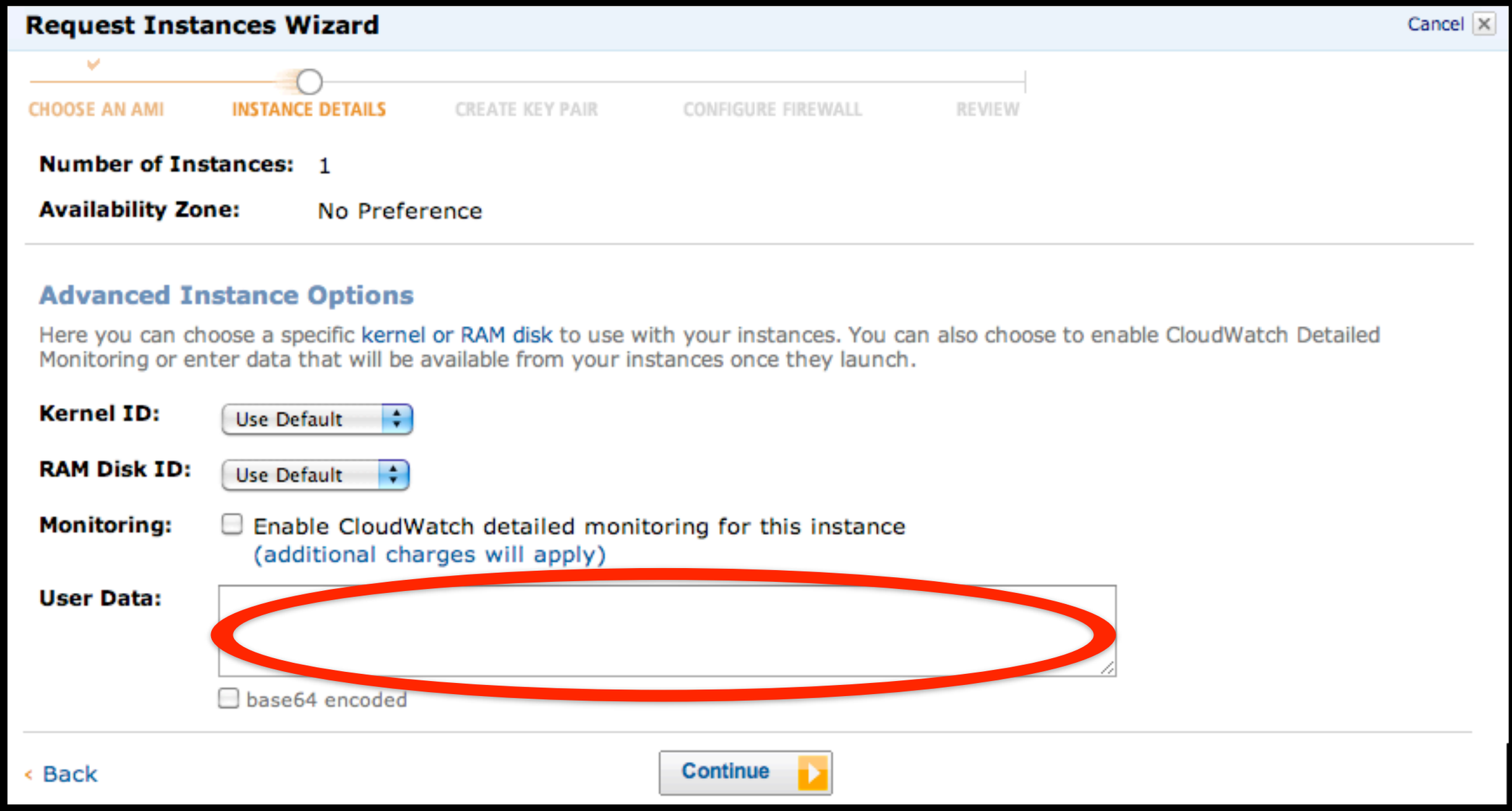
Find the AMI


a6f5
Request Instances Wizard
Cancel
CHOOSE AN AMI
INSTANCE DETAILS
CREATE KEY PAIR
CONFIGURE FIREWALL
REVIEW
Choose an Amazon Machine Image (AMI) from one of the tabbed lists below by clicking its Select button.
Quick Start
My AMIs
Community AMIs
Viewing:
All Images
1 to 1 of 1 Items
AMI ID
Root Device
Manifest
Platform
ami-1a837773
instance-store
ubuntu-images-us/ubuntu-maverick-10.10-i386-server-2010
Ubuntu
Select

Instance Details



Instance Details

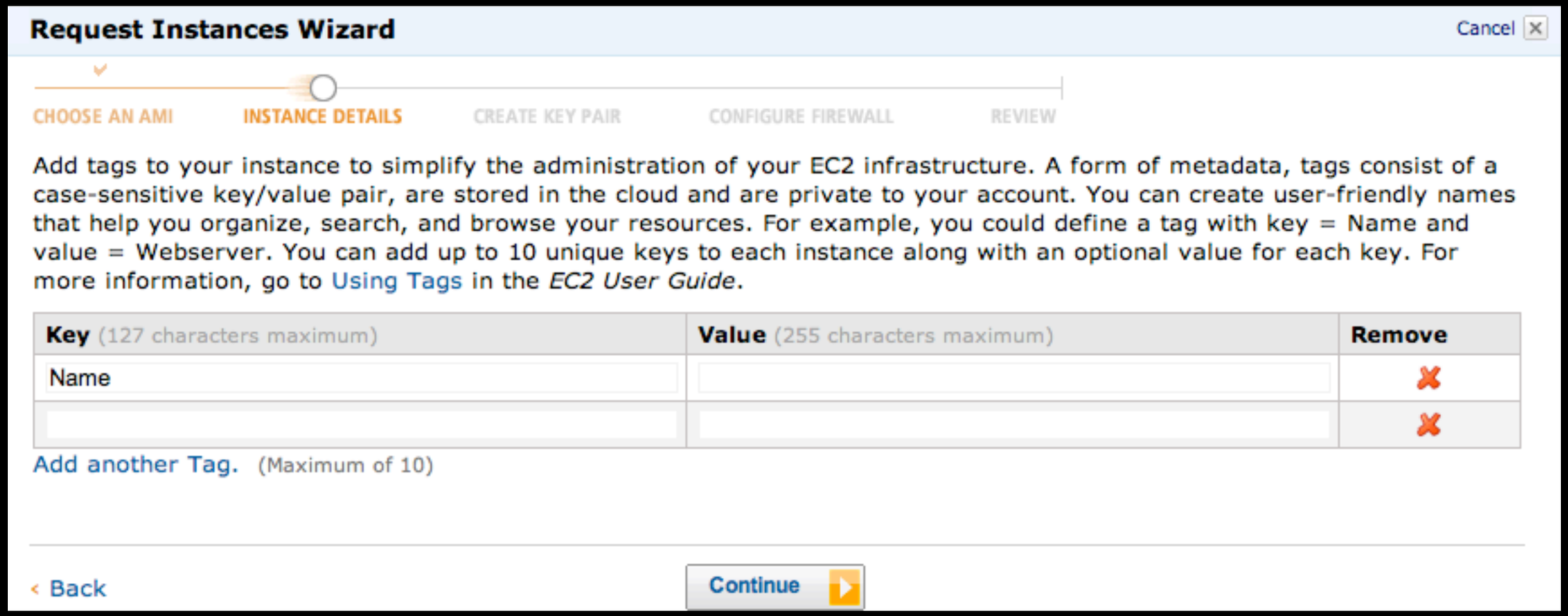


Paste in User Data

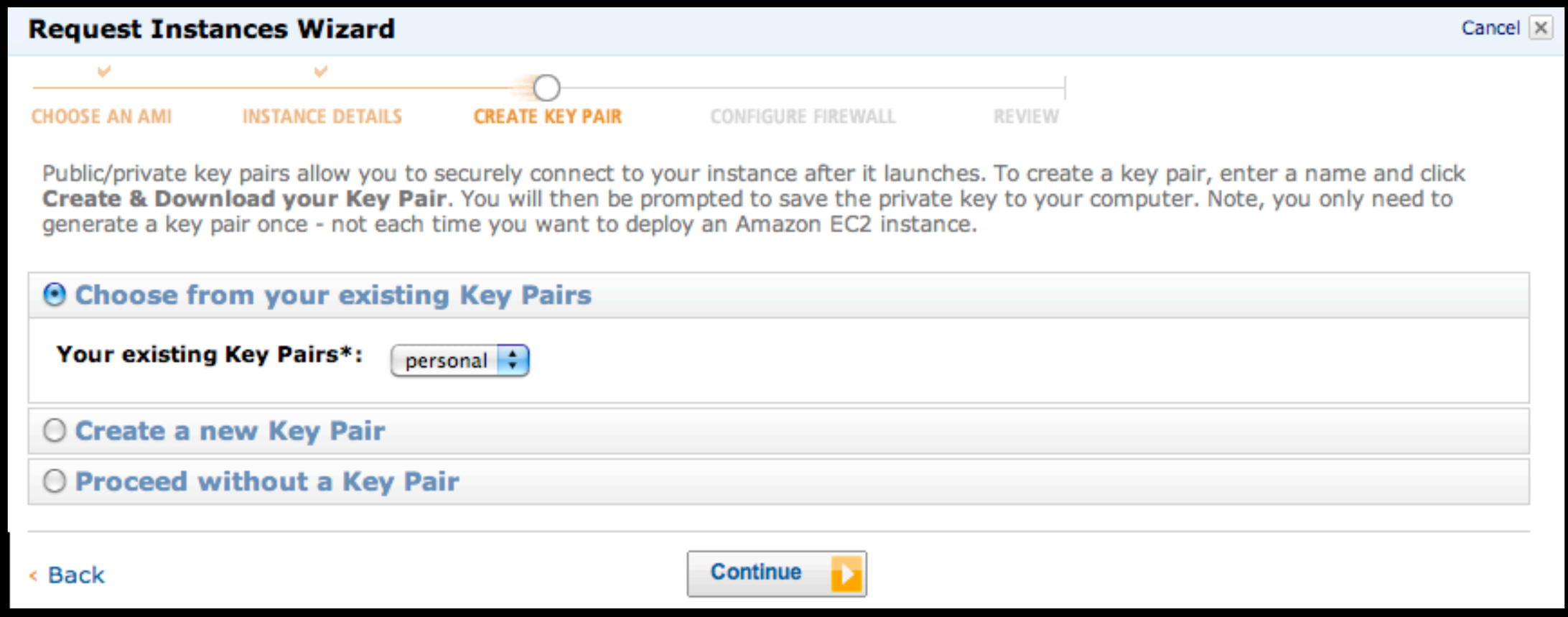
- `#!/bin/sh`

```
wget http://bit.ly/pycon-aws-django
bash pycon-aws-django
   -n clktc
   -d http://bit.ly/pycon-clktc
   -s “http://<cloudfront_dns_name>/”
   <output from database>
2>&1| tee /root/install.log
```

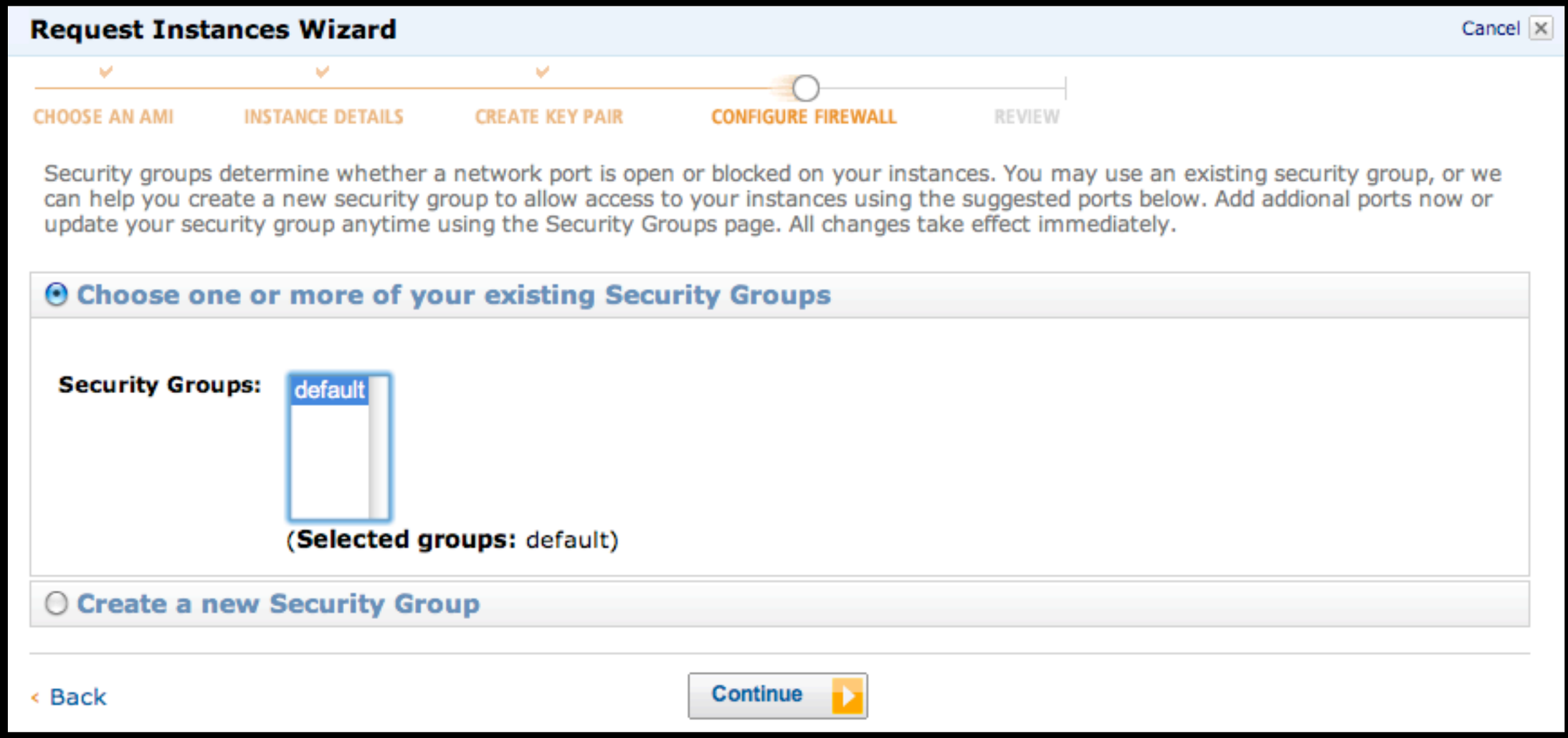
Tags for Identification



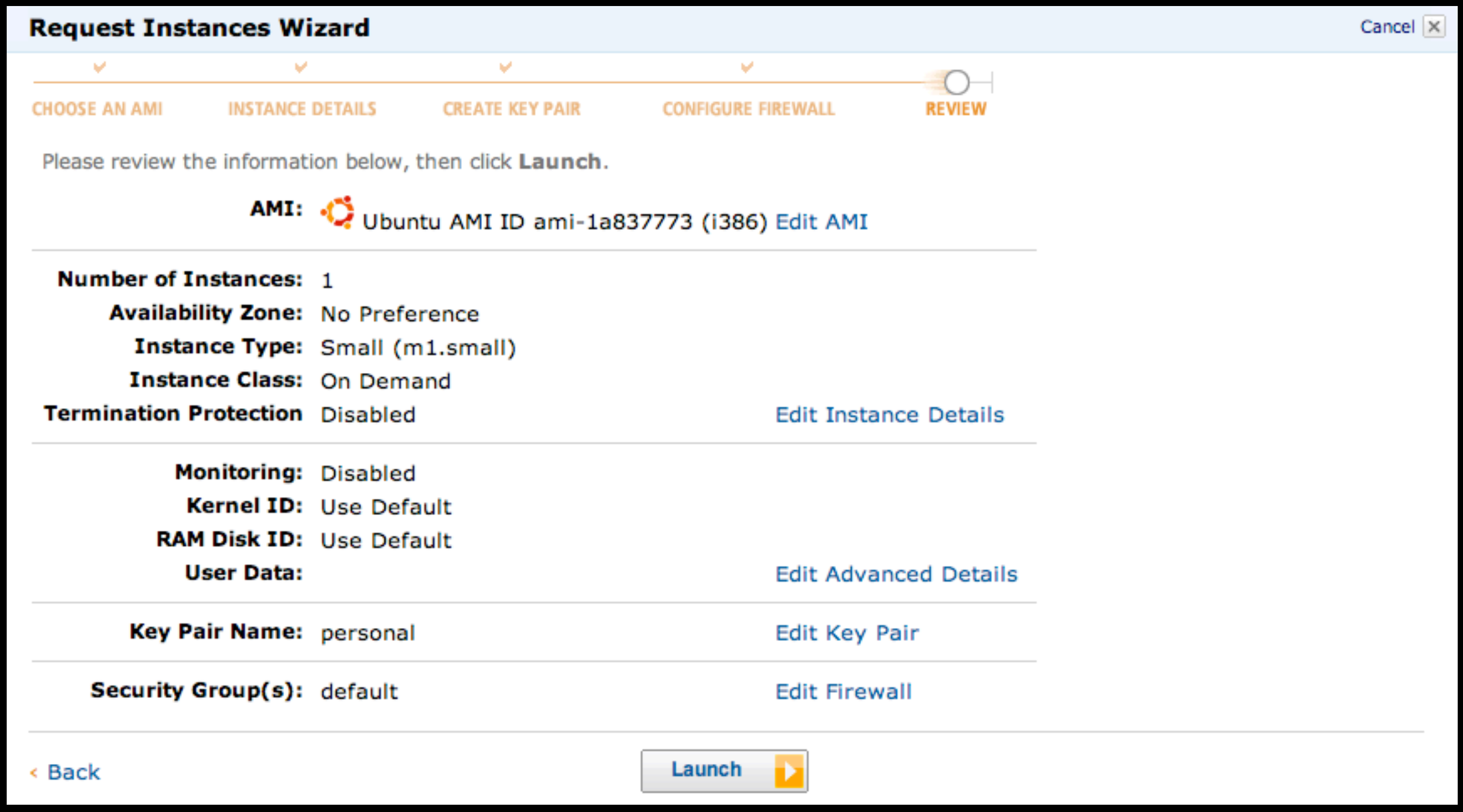
Select Key Pair



Configure Firewall



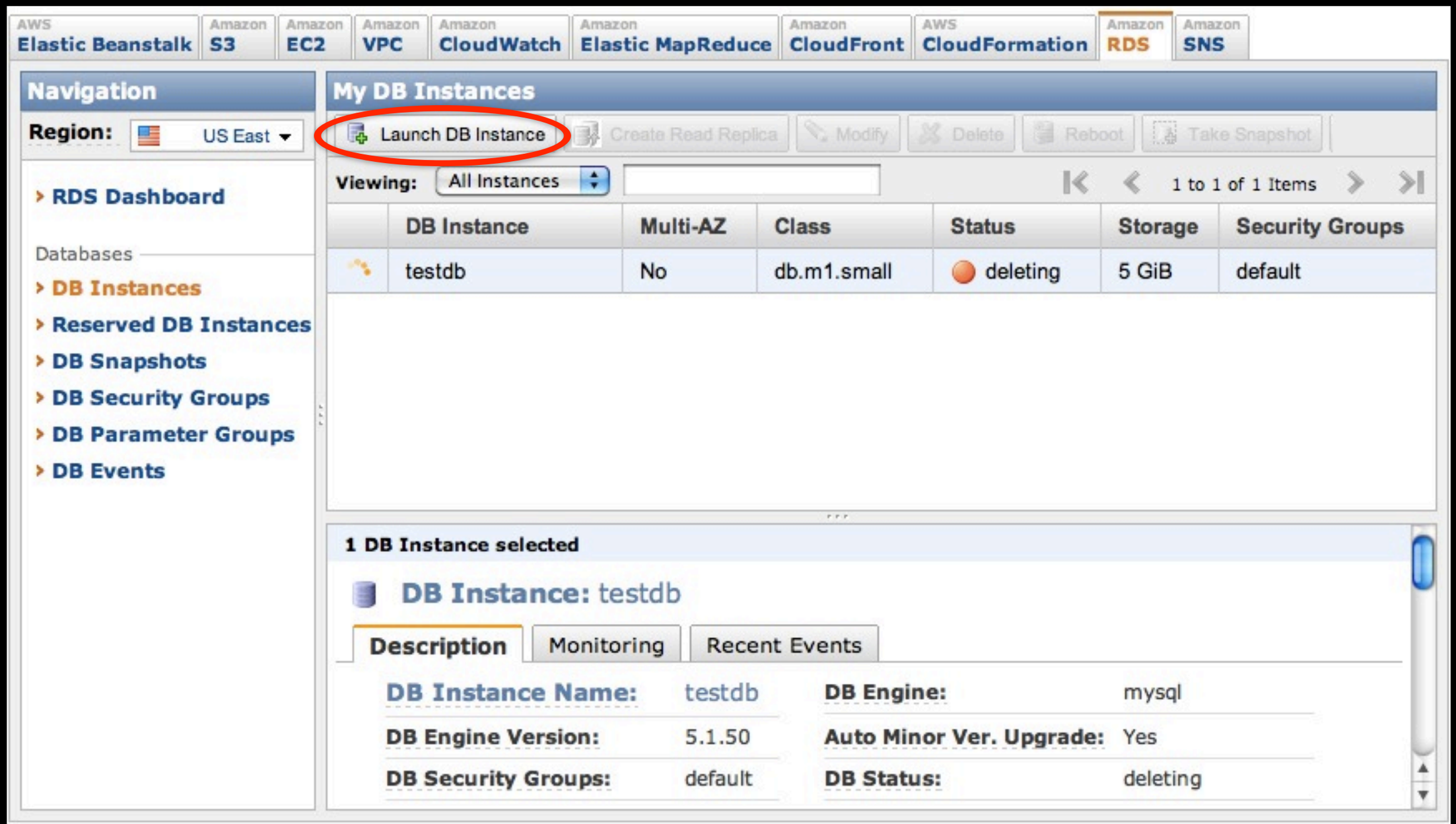
Launch


Request Instances Wizard
Cancel
CHOOSE AN AMI
INSTANCE DETAILS
CREATE KEY PAIR
CONFIGURE FIREWALL
REVIEW
Please review the information below, then click Launch.
AMI: Ubuntu AMI ID ami-1a837773 (i386) Edit AMI
Number of Instances: 1
Availability Zone: No Preference
Instance Type: Small (m1.small)
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: personal
Edit Key Pair
Security Group(s): default
Edit Firewall
Back
Launch

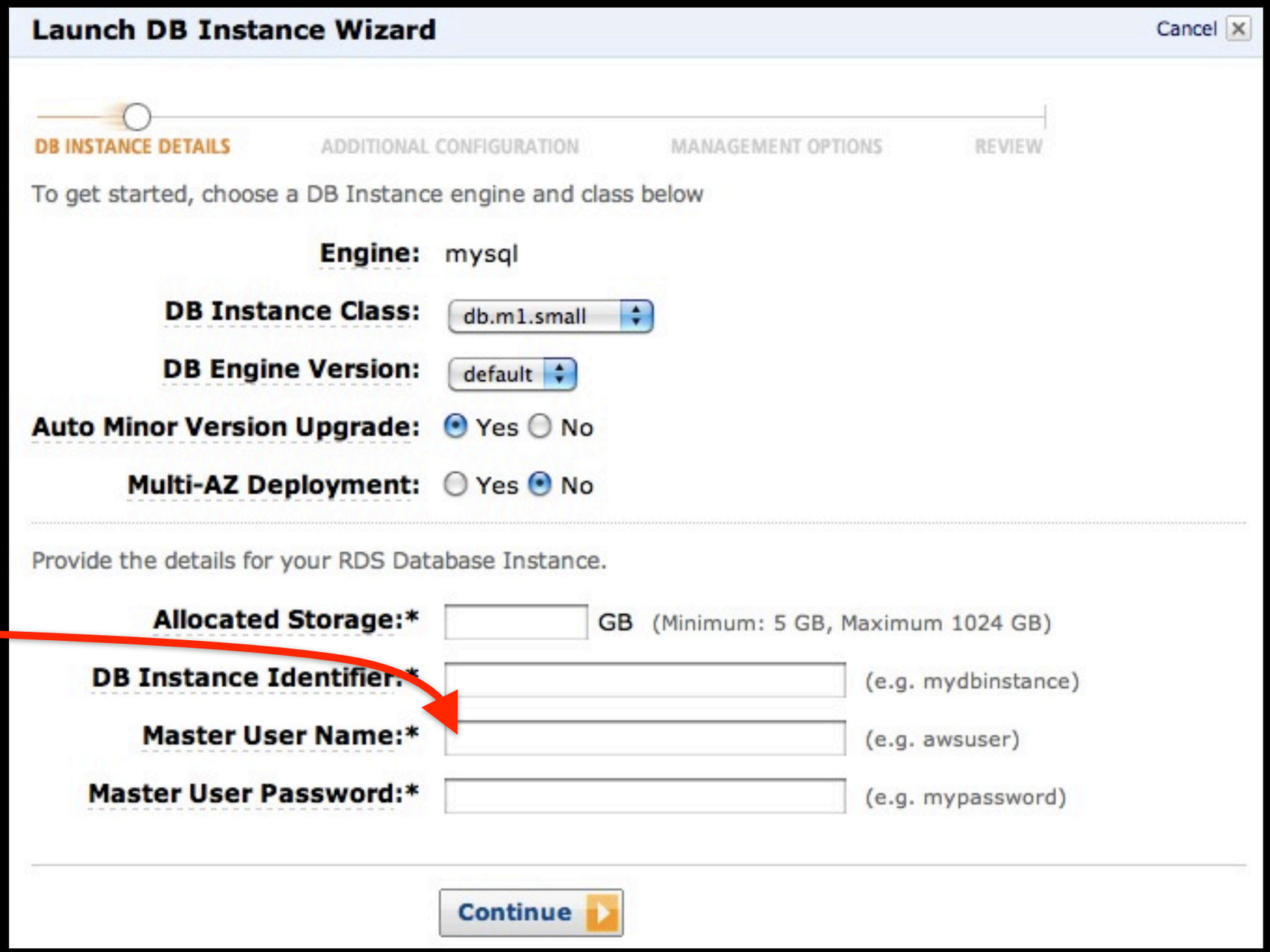
Exercise 3

- Create RDS database
- Move data from current MySQL instance
- Create read-replica
- Re-launch Django instance

Launch DB Instance



Username & Password

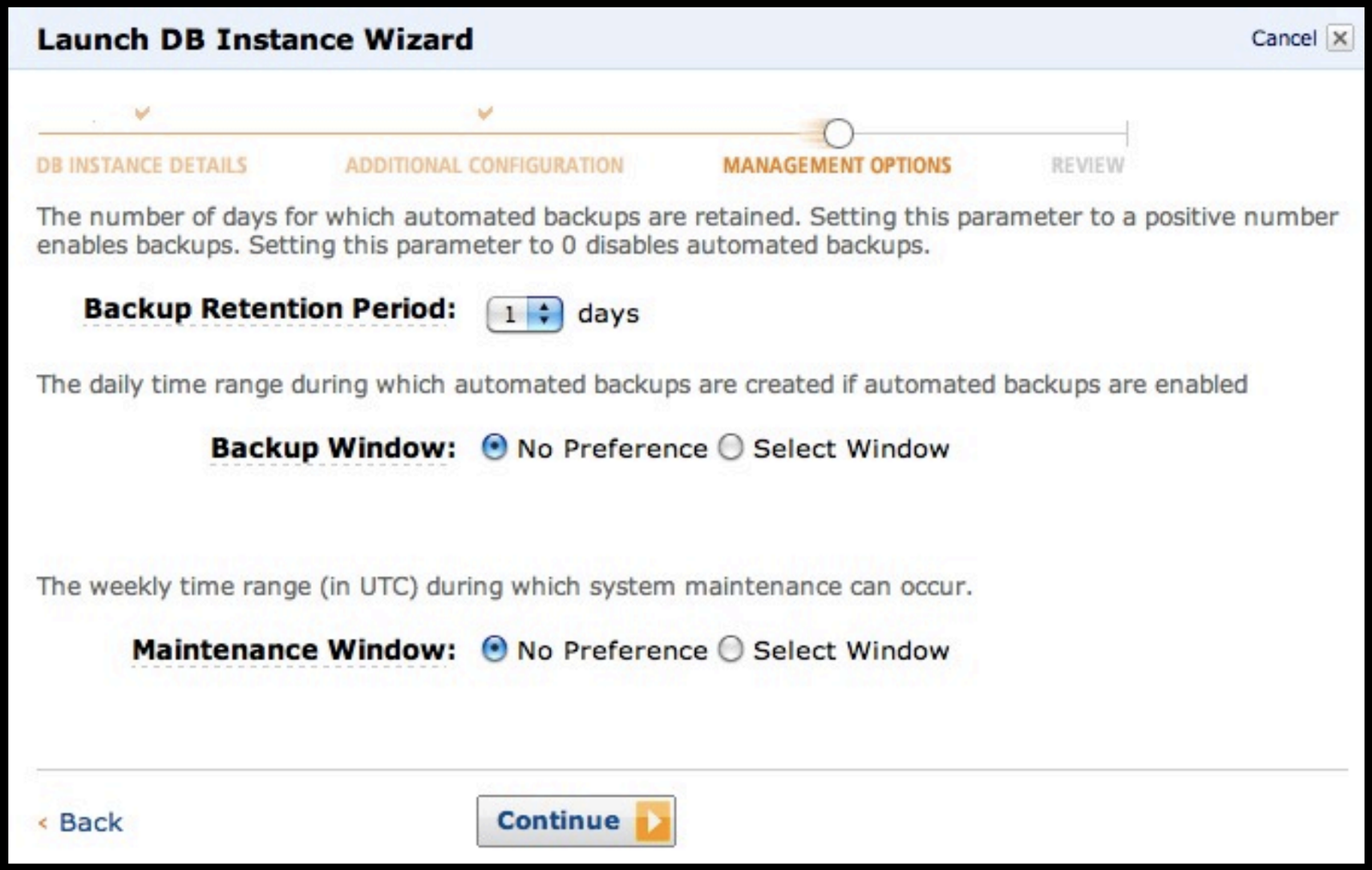


clktc

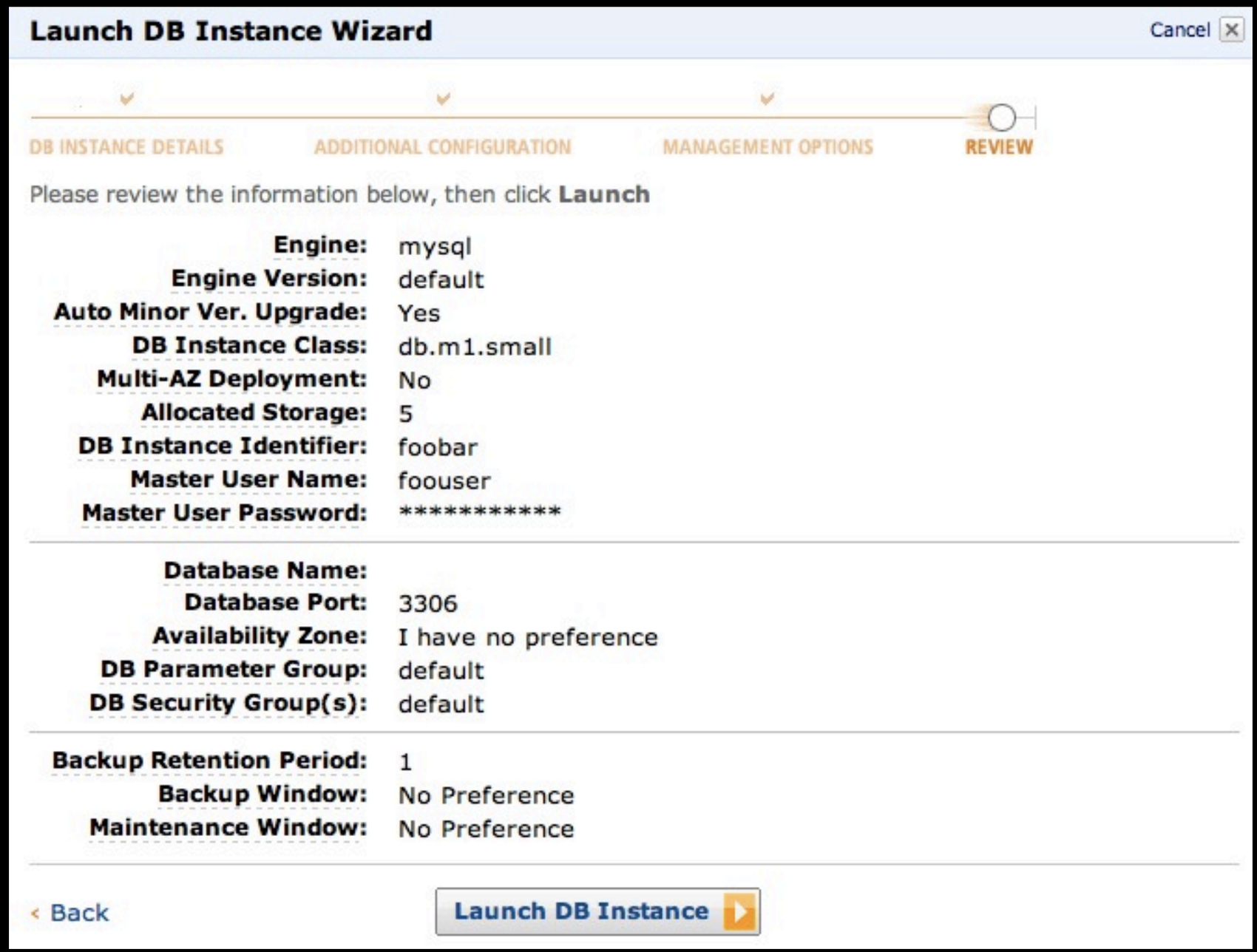
Database Name

clktc

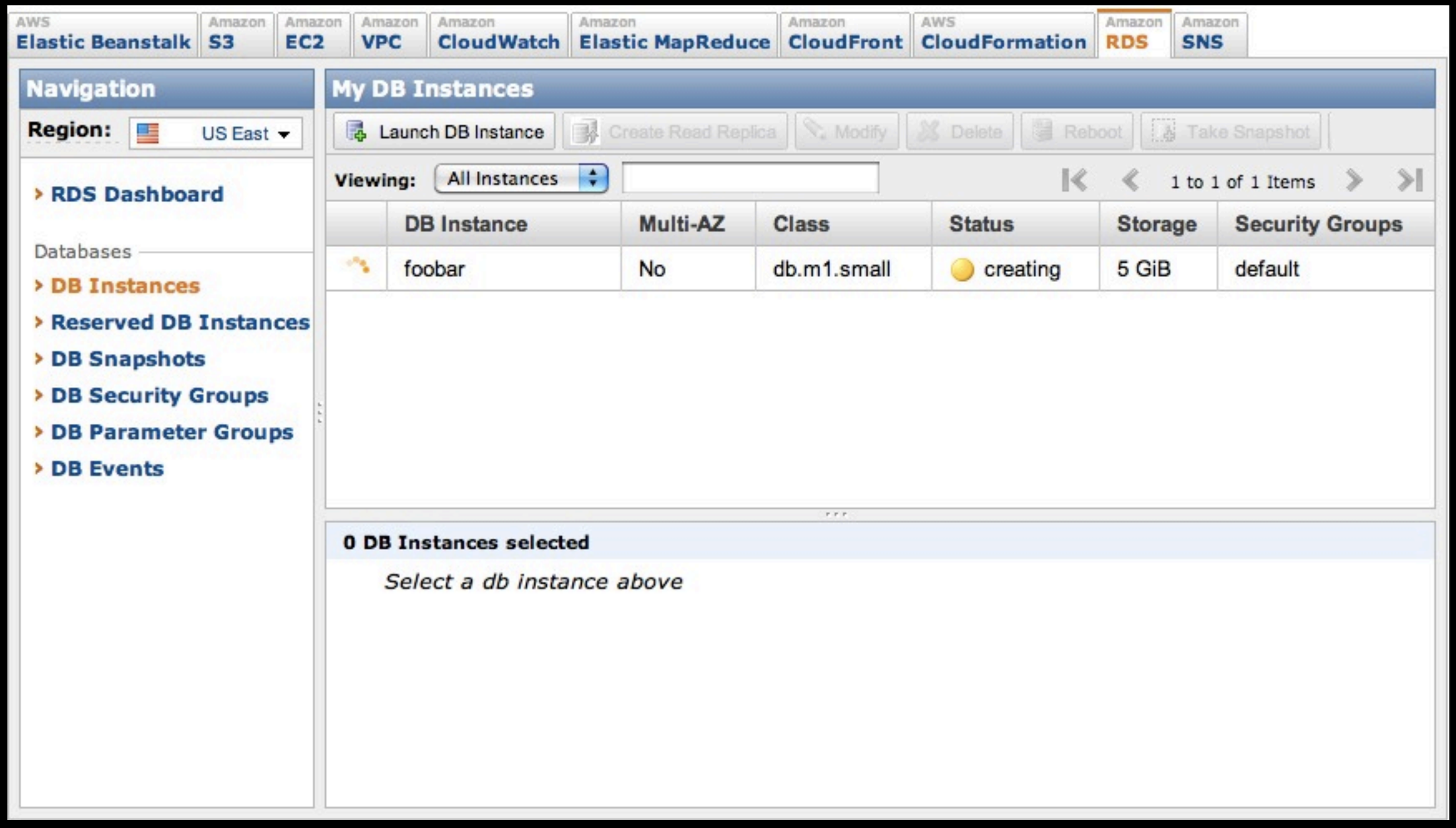
Management Options



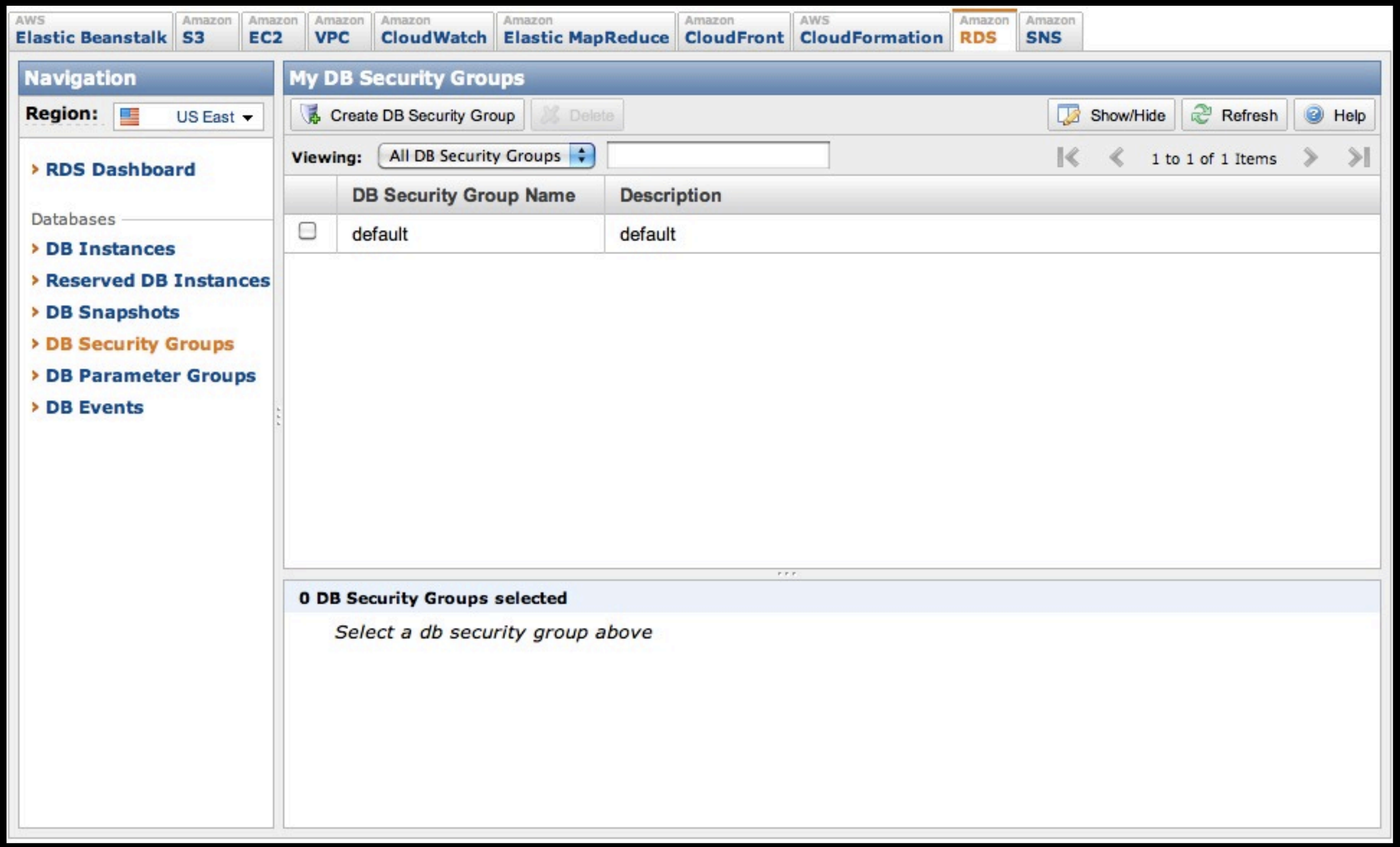
Review RDS details



Database is creating



Configure DB security



Security Credentials

Your Account

Account Activity

View current charges and account activity, itemized by service and by usage type. Previous months' billing statements are also available.

Usage Reports

Download usage reports for each service you are subscribed to. Reports can be customized by specifying usage types, timeframe, service operations, and more.

Security Credentials

Amazon Web Services uses access identifiers to authenticate requests to AWS and to identify the sender of a request. Three types of identifiers are available: (1) AWS Access Key Identifiers, (2) X.509 Certificates, and (3) Key pairs

Personal Information

View and edit personal contact information, such as address and phone number. Set communication preferences for email subscriptions.

Get Account ID

Account Identifiers

AWS uses two types of account identifiers — canonical user ID and AWS account ID. These account identifiers are used to share resources between accounts.

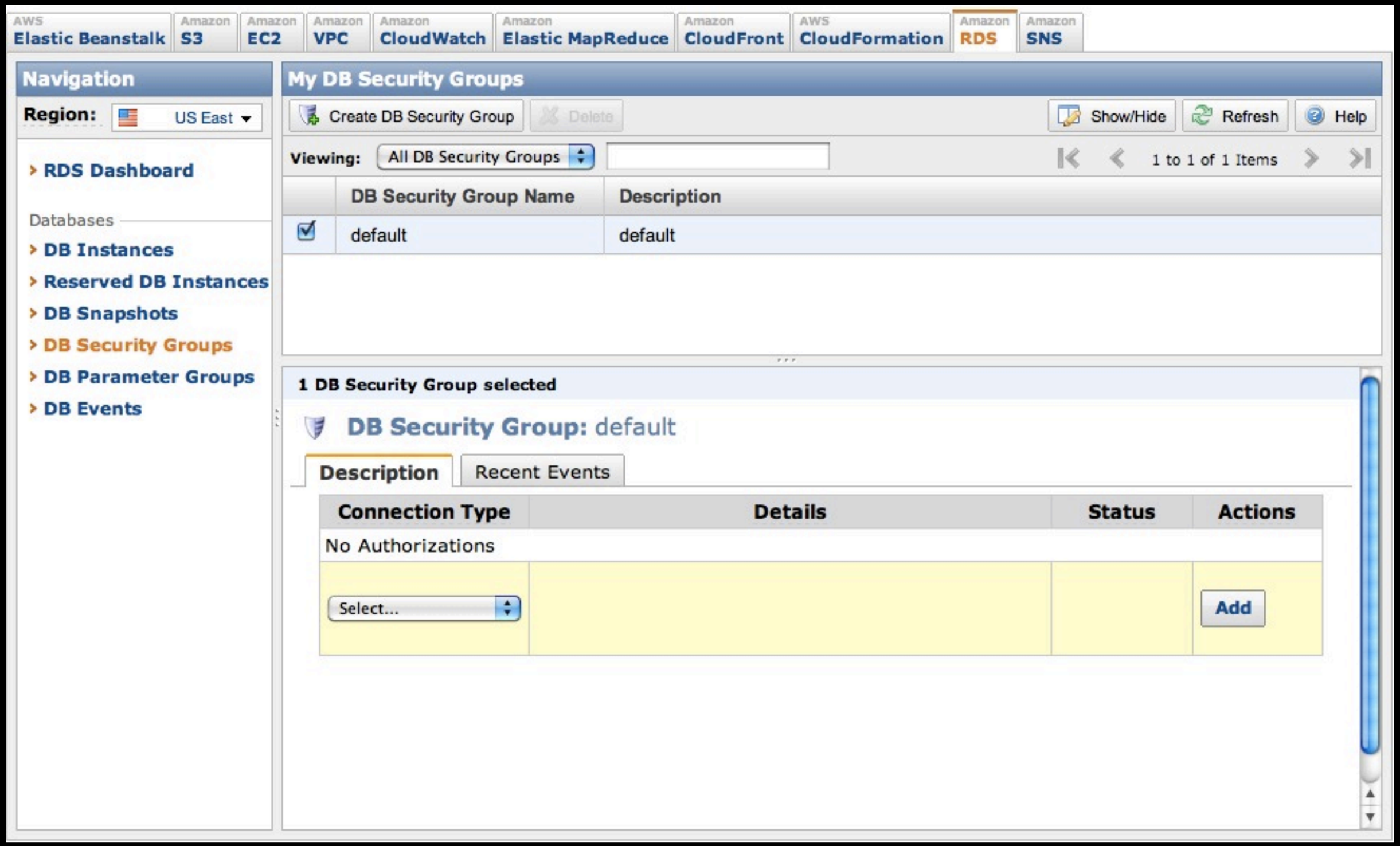
The canonical user ID can be used exclusively for Amazon S3 resources such as buckets or files.

The AWS account ID can be used for all AWS service resources except Amazon S3. These resources include Amazon EC2 AMIs, Amazon EBS snapshots, Amazon SQS queues, etc.

AWS Account ID: 4791-0720-0387
Canonical User ID: View canonical user ID

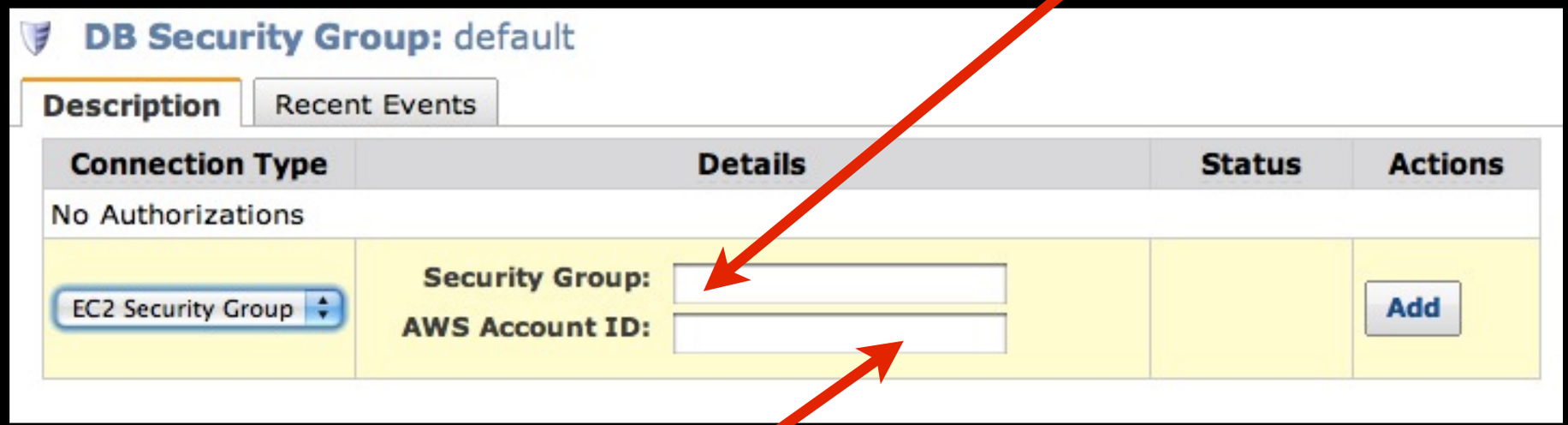
Learn more about Account Identifiers

Edit the default group



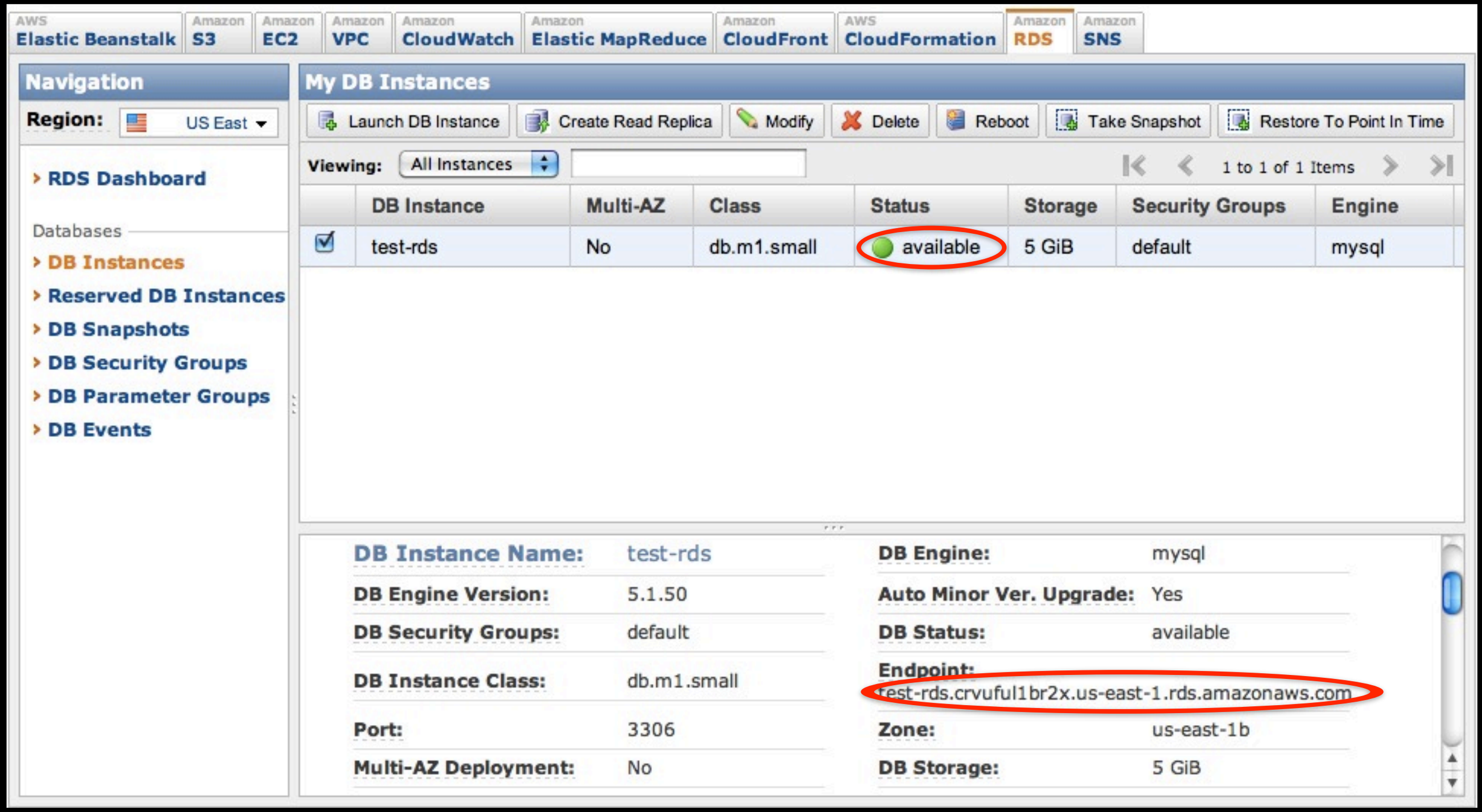
EC2 Security Group

default



account id (no dashes)

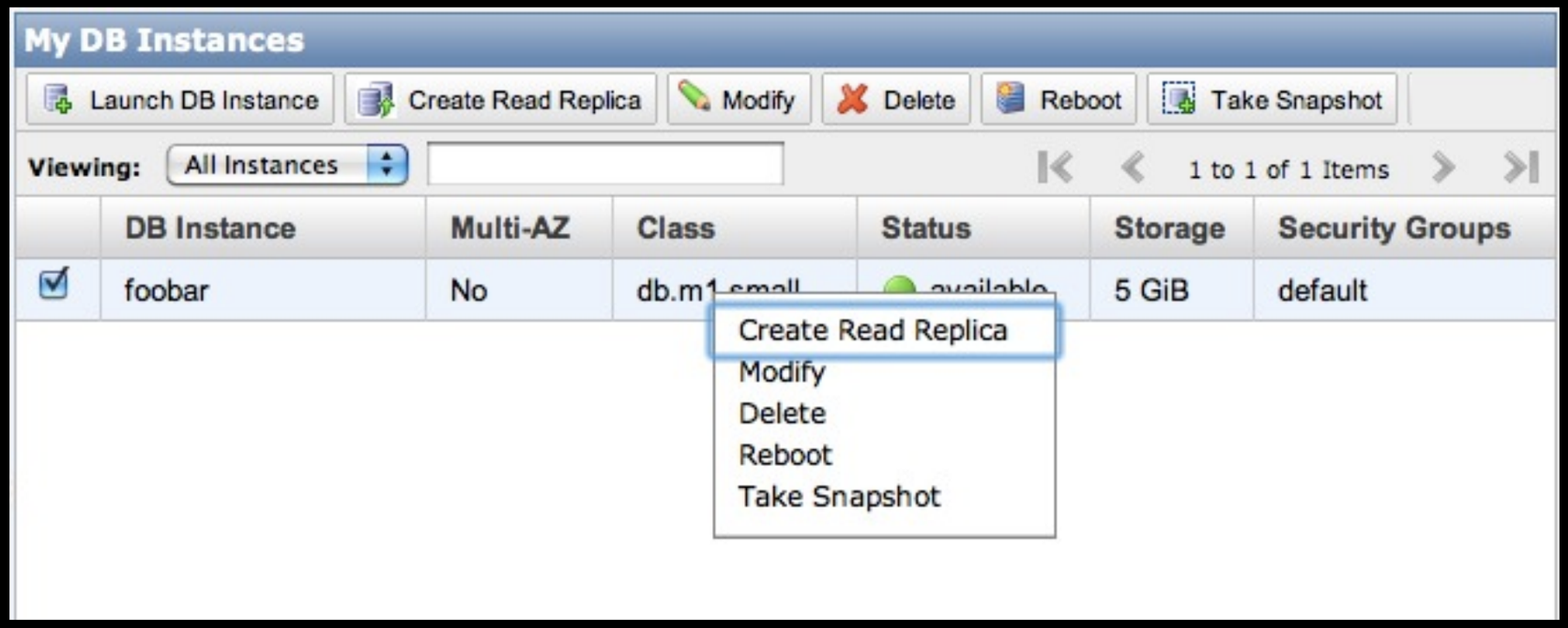
Wait for RDS to finish



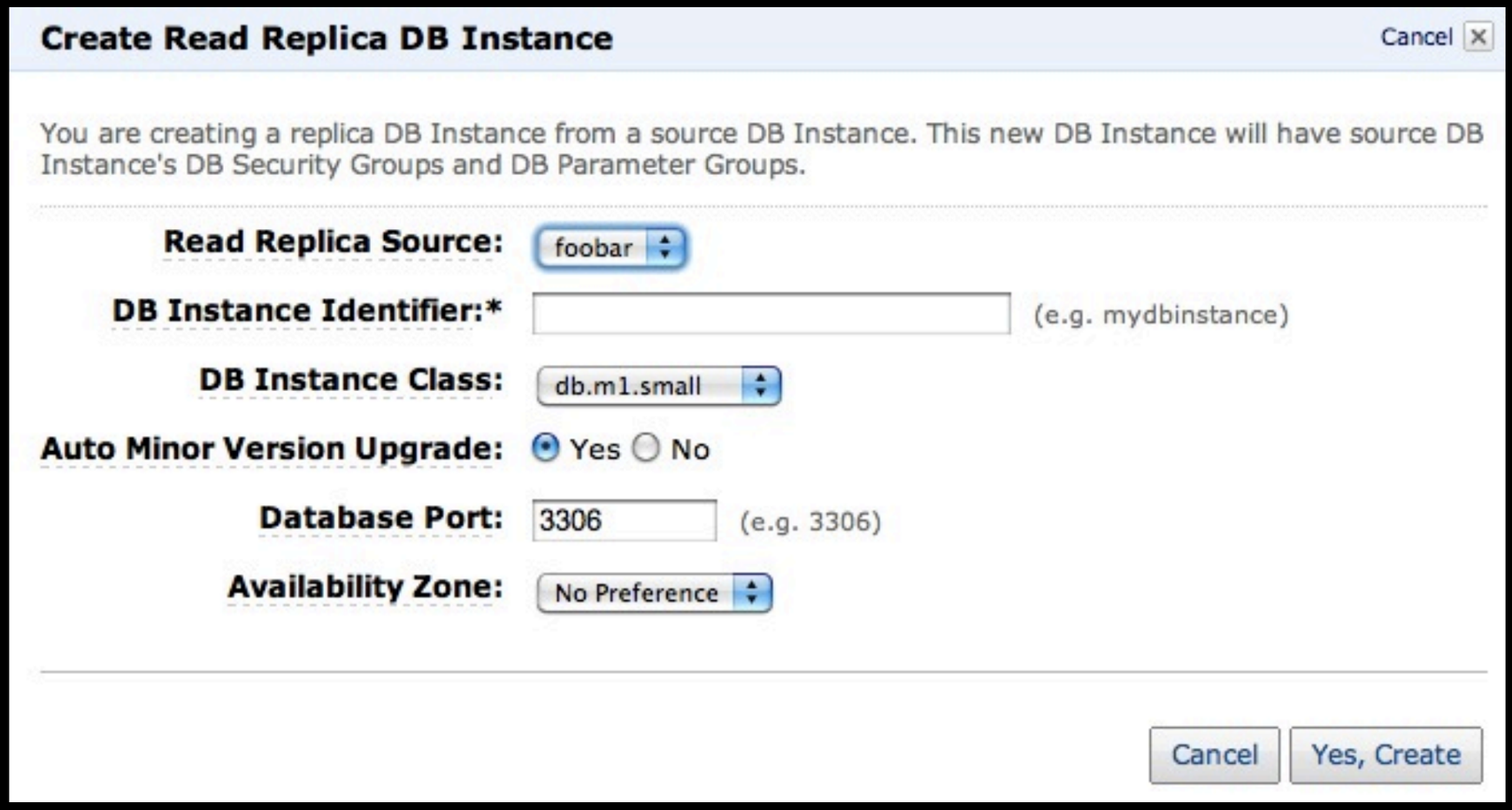
On MySQL instance

- `wget http://bit.ly/pycon-to-rds`
- `bash pycon-to-rds`
  `-U clktc`
  `-D clktc`
  `-P <password>`
  `-H <rds_endpoint>`

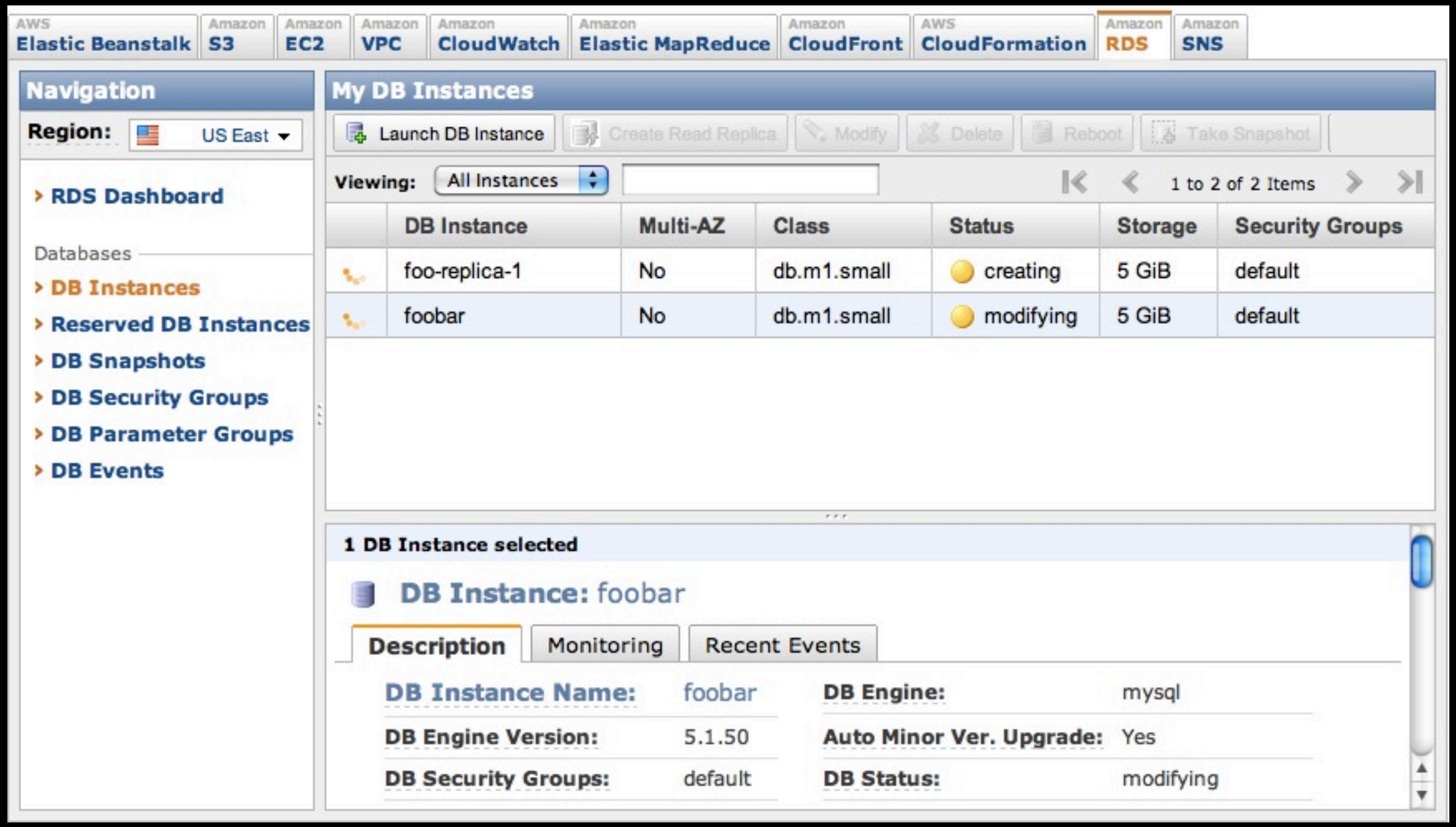
Create read replica



Read replica settings



Replica is now creating



Launch New Instances

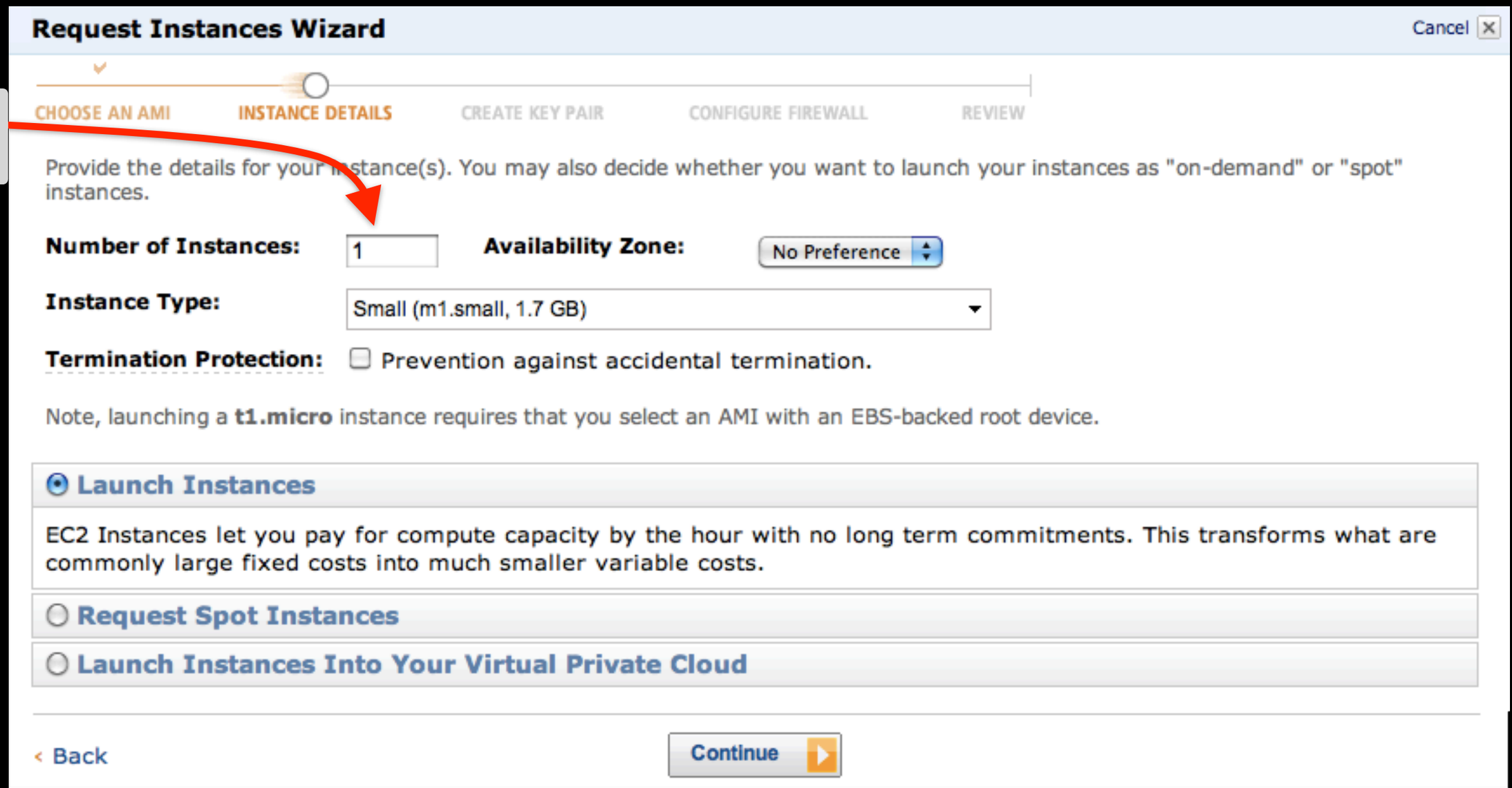
- Configured to point at RDS
- If we had more time we could also point it at the read replicas

Find the AMI

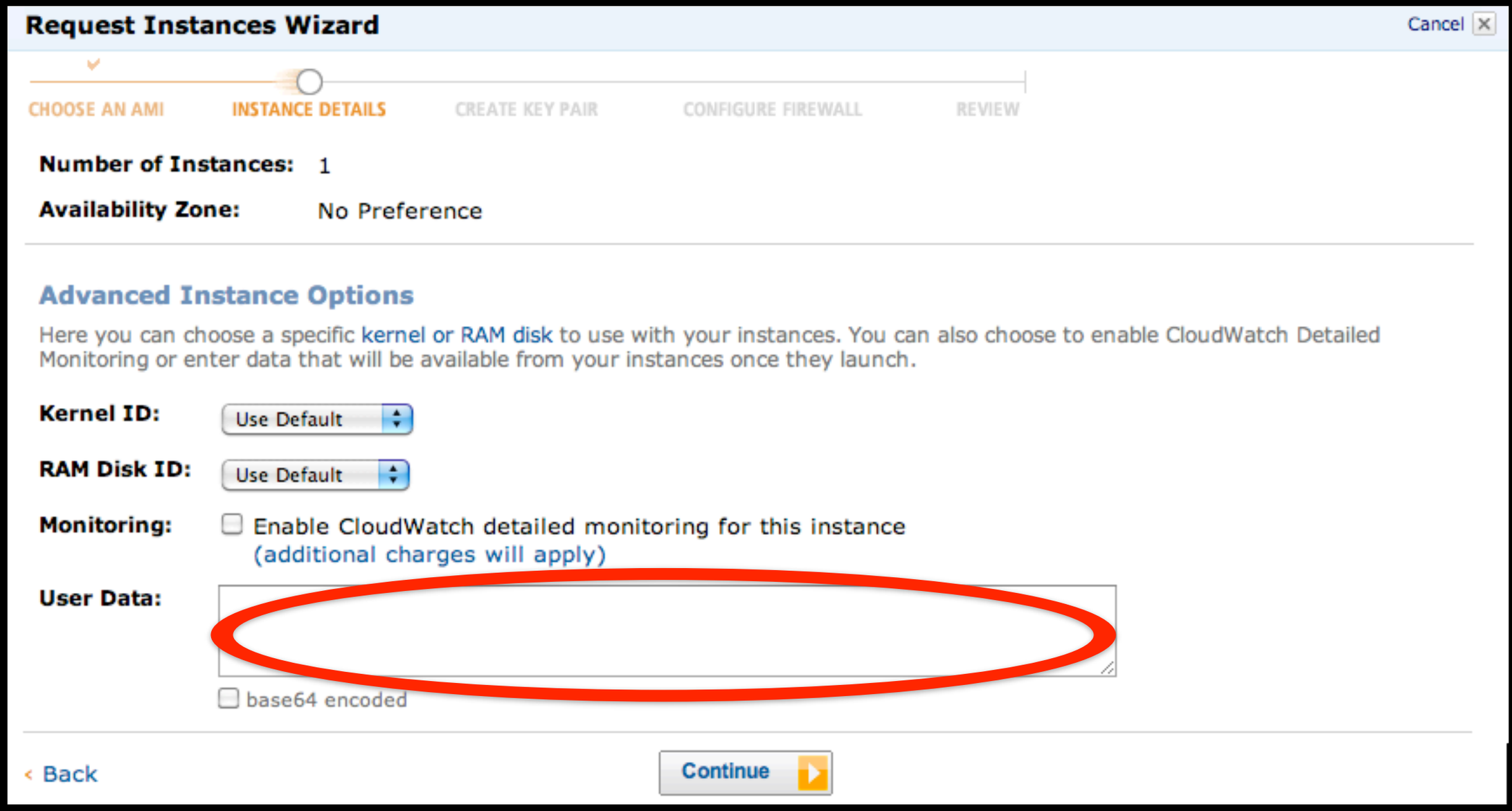

a6f5
Request Instances Wizard
Cancel
CHOOSE AN AMI
INSTANCE DETAILS
CREATE KEY PAIR
CONFIGURE FIREWALL
REVIEW
Choose an Amazon Machine Image (AMI) from one of the tabbed lists below by clicking its Select button.
Quick Start
My AMIs
Community AMIs
Viewing:
All Images
1 to 1 of 1 Items
AMI ID
Root Device
Manifest
Platform
ami-1a837773
instance-store
ubuntu-images-us/ubuntu-maverick-10.10-i386-server-2010
Ubuntu
Select

Instance Details

2



Instance Details

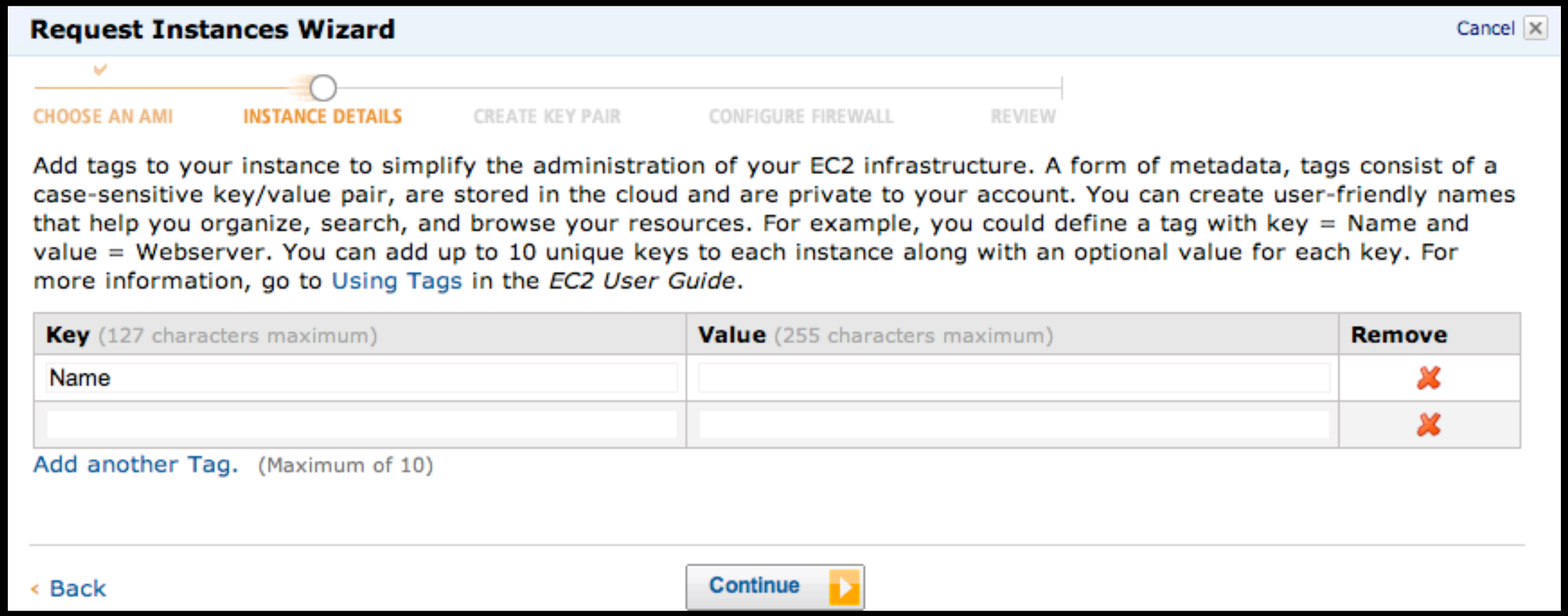


Paste in User Data

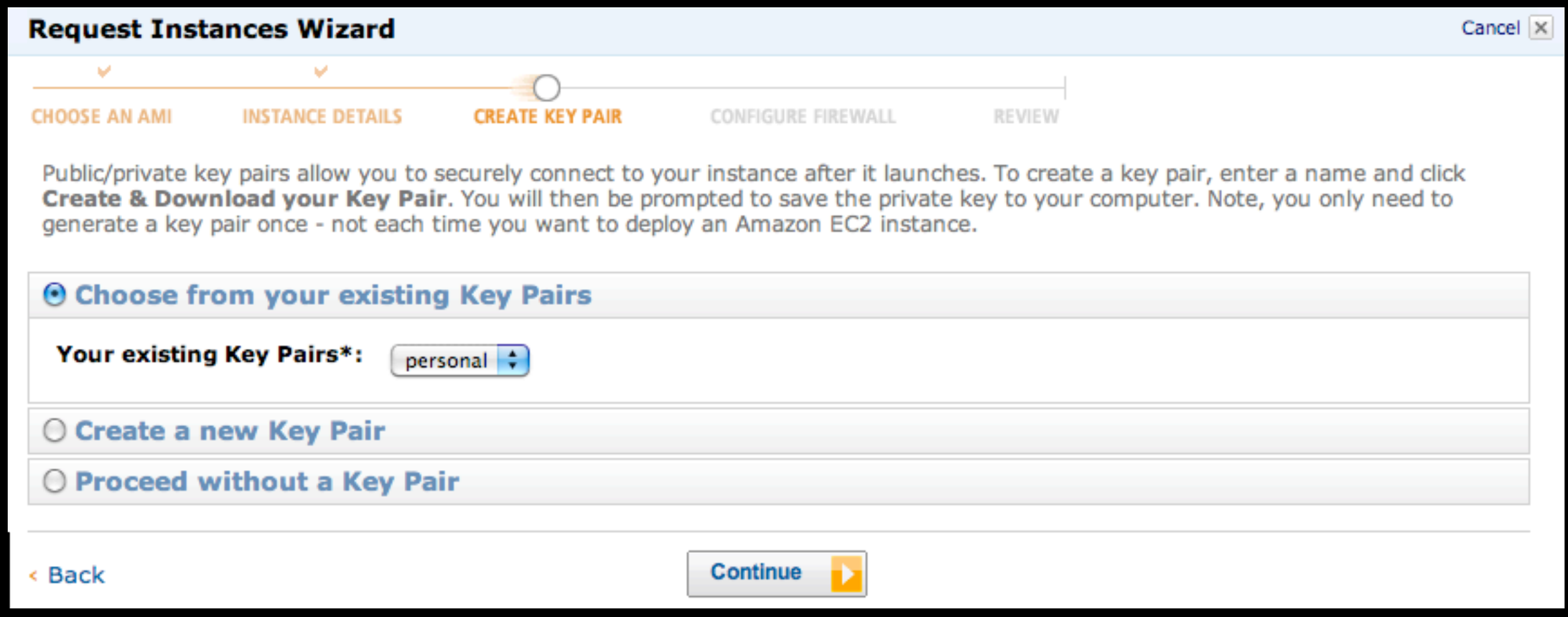
- `#!/bin/sh`

```
wget http://bit.ly/pycon-aws-django
bash pycon-aws-django
   -n clktc
   -d http://bit.ly/pycon-clktc
   -s “http://<cloudfront_dns_name>/”
   -U clktc -D clktc -P <password>
   -H <rds_endpoint>
2>&1| tee /root/install.log
```

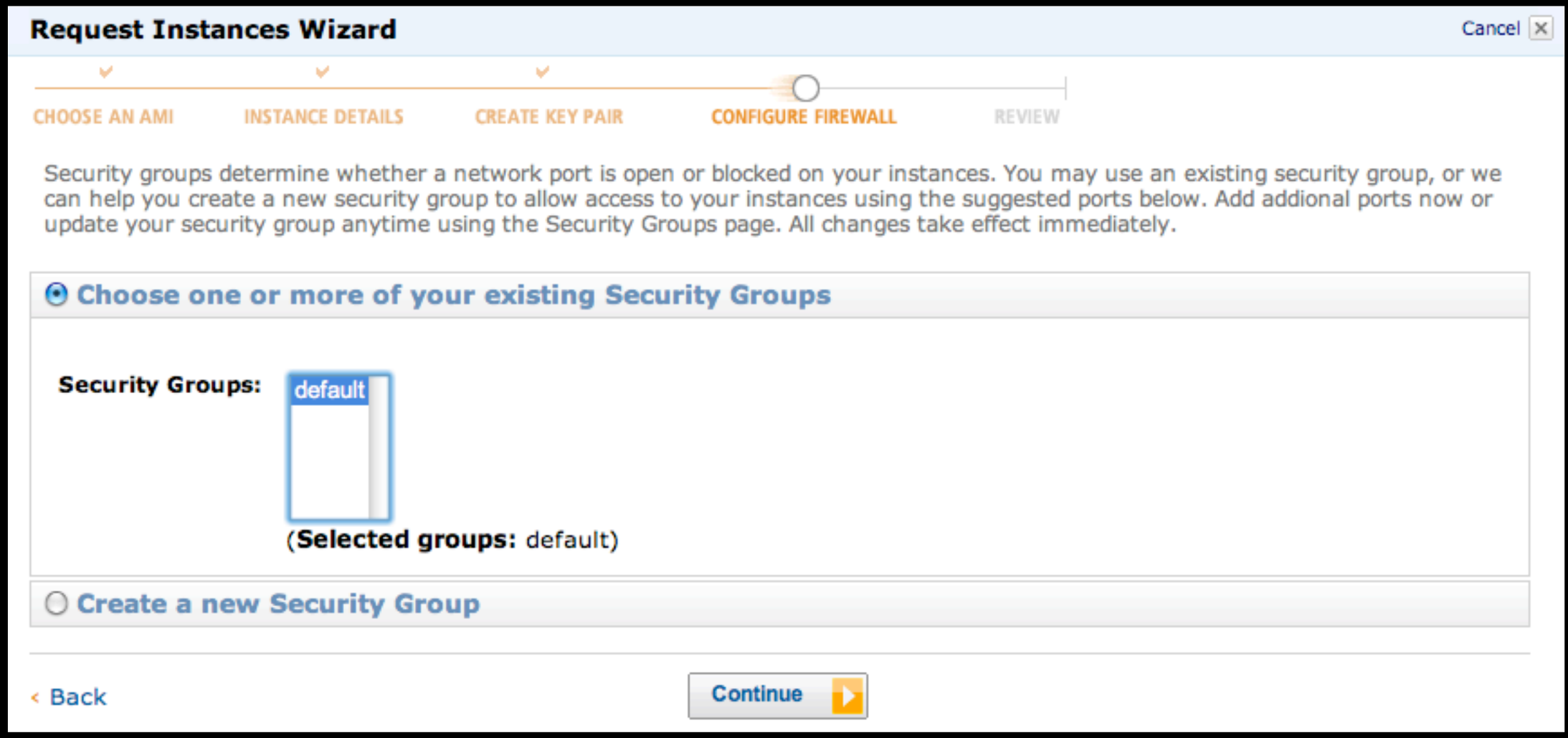
Tags for Identification



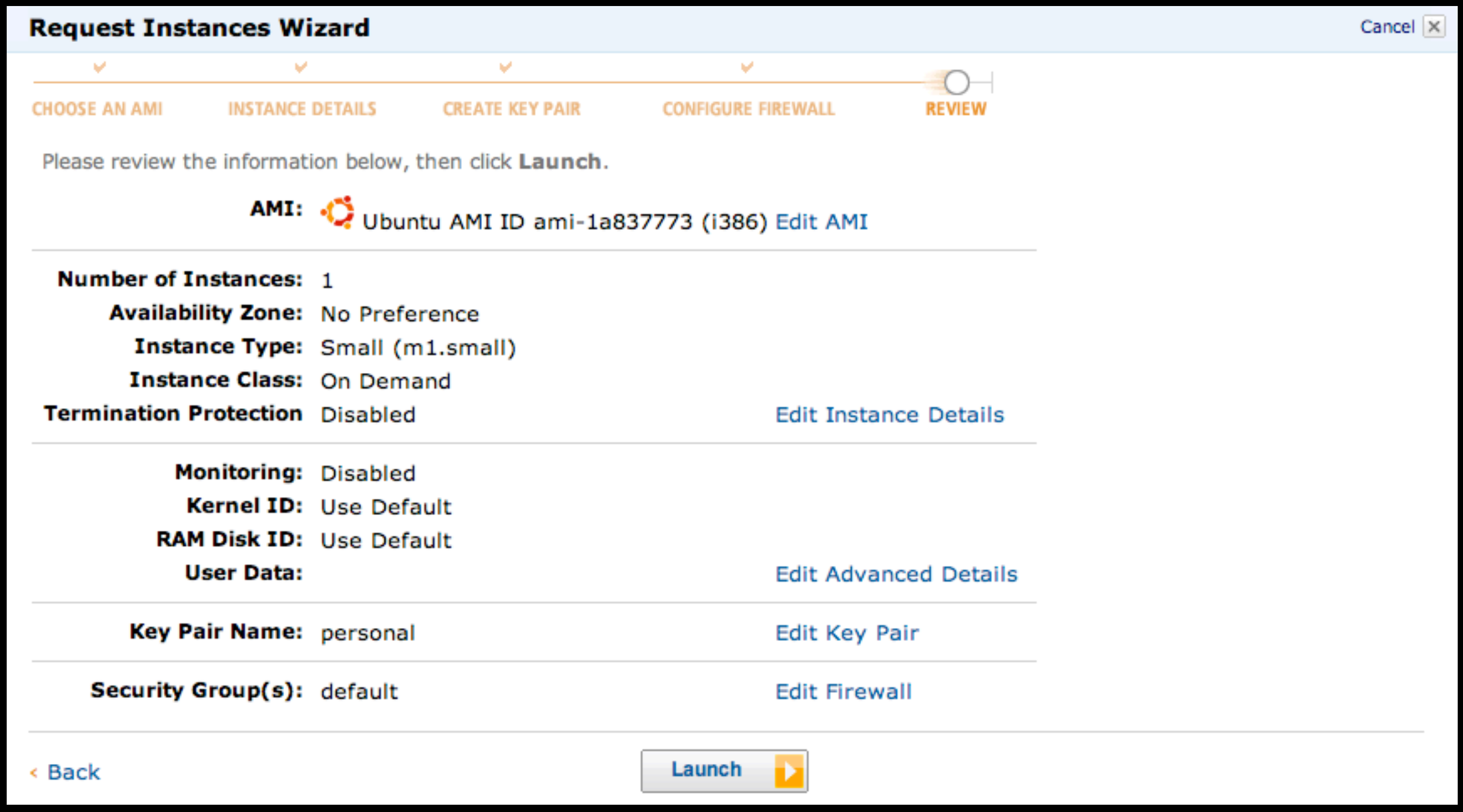
Select Key Pair



Configure Firewall



Launch


Request Instances Wizard
Cancel
CHOOSE AN AMI
INSTANCE DETAILS
CREATE KEY PAIR
CONFIGURE FIREWALL
REVIEW
Please review the information below, then click Launch.
AMI: Ubuntu AMI ID ami-1a837773 (i386) Edit AMI
Number of Instances: 1
Availability Zone: No Preference
Instance Type: Small (m1.small)
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: personal
Edit Key Pair
Security Group(s): default
Edit Firewall
Back
Launch

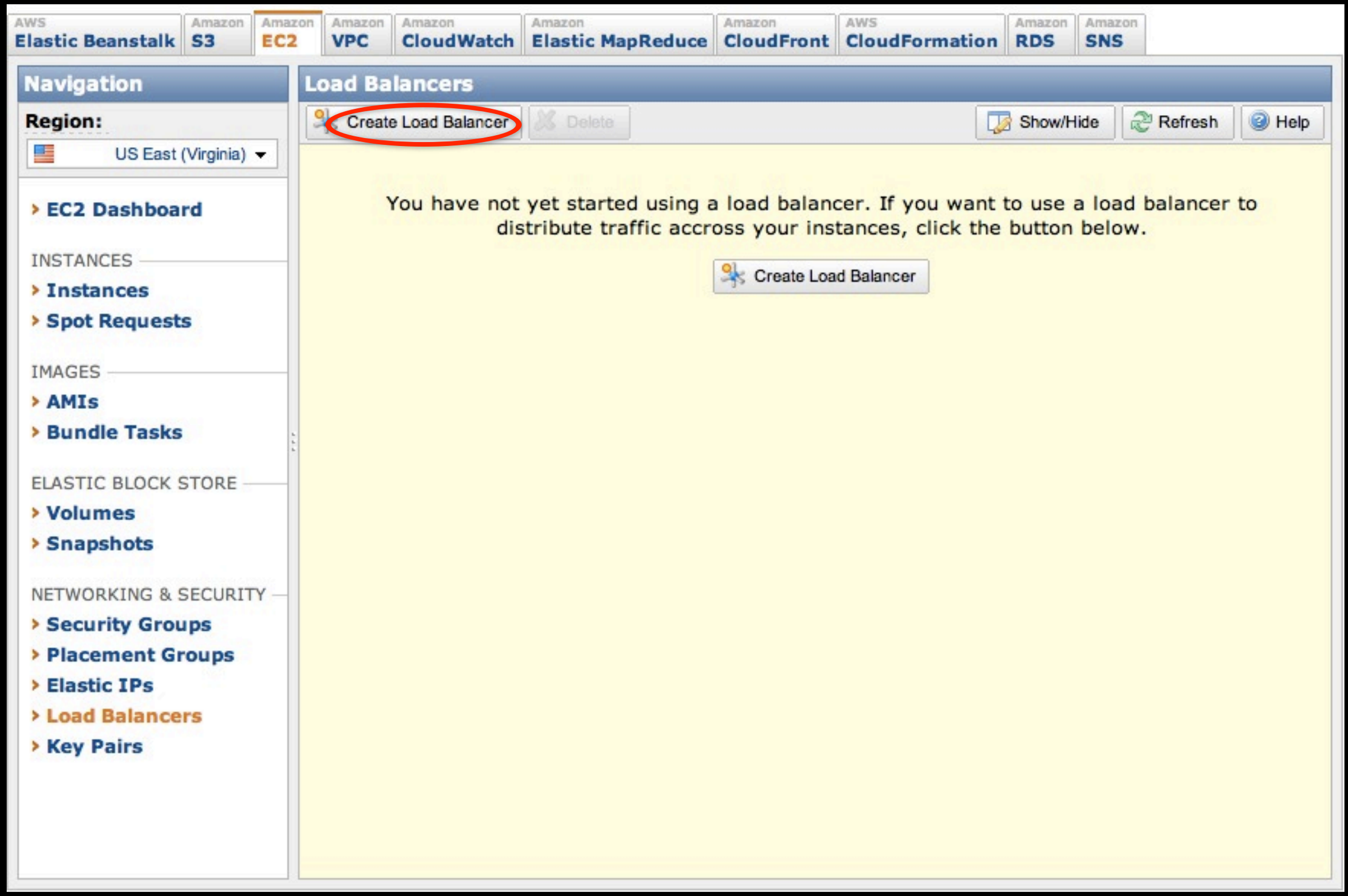
Checkpoint

- Save instance ids of new RDS instances
- Wait for RDS backed instance to launch
- Verify in browser that new instances works

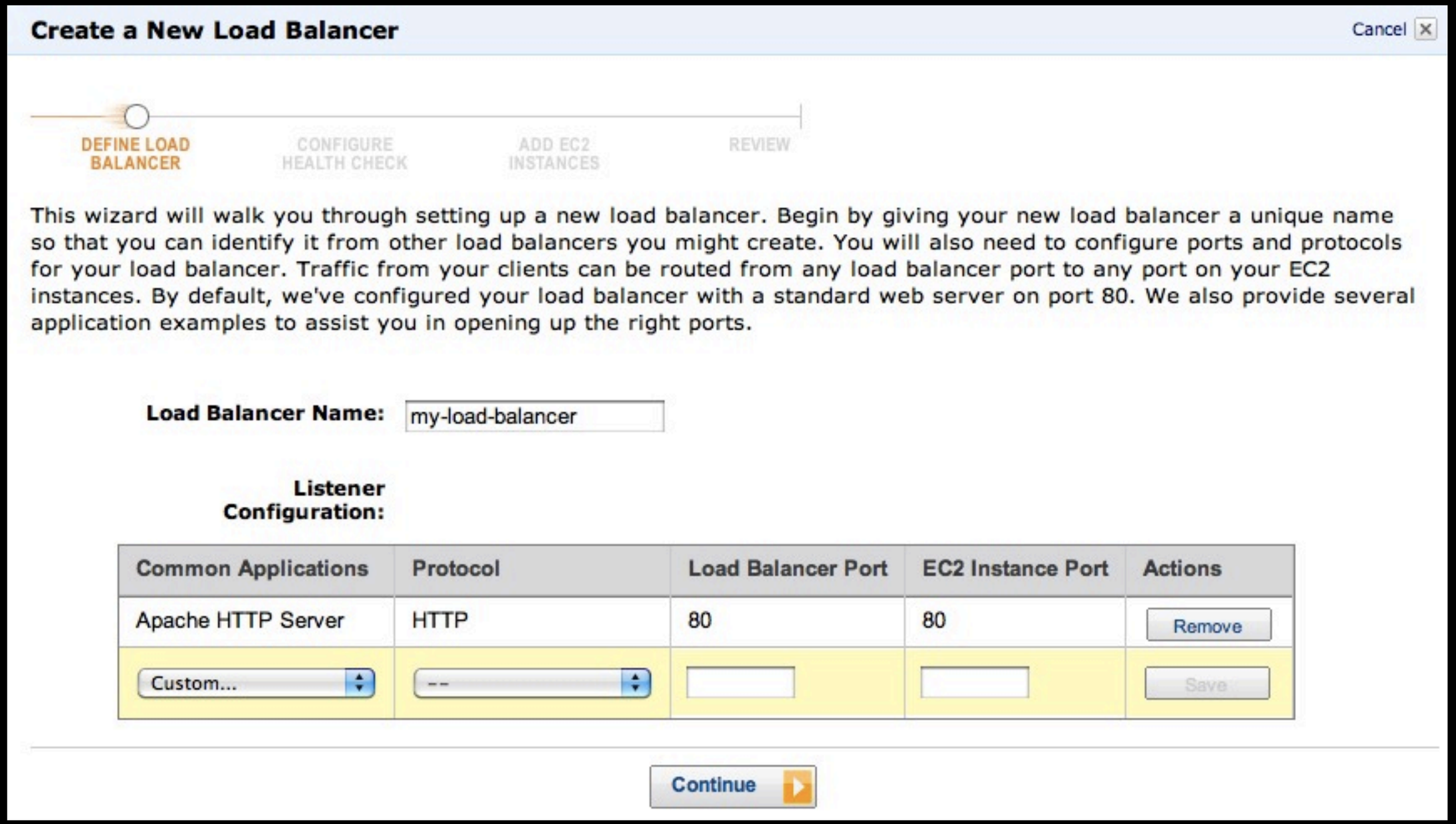
Exercise 4

- Create elastic load balancer
- Add instances to ELB

Create ELB


AWS Elastic Beanstalk
Amazon S3
Amazon EC2
Amazon VPC
Amazon CloudWatch
Amazon Elastic MapReduce
Amazon CloudFront
AWS CloudFormation
Amazon RDS
Amazon SNS
Navigation
Region:
US East (Virginia)
EC2 Dashboard
INSTANCES
Instances
Spot Requests
IMAGES
AMIs
Bundle Tasks
ELASTIC BLOCK STORE
Volumes
Snapshots
NETWORKING & SECURITY
Security Groups
Placement Groups
Elastic IPs
Load Balancers
Key Pairs
Load Balancers
Create Load Balancer
Delete
Show/Hide
Refresh
Help
You have not yet started using a load balancer. If you want to use a load balancer to distribute traffic accross your instances, click the button below.
Create Load Balancer

Name and ports



Health Check

Create a New Load Balancer Cancel

DEFINE LOAD BALANCER ✓ — CONFIGURE HEALTH CHECK — ADD EC2 INSTANCES — REVIEW

Your load balancer will automatically perform health checks on your EC2 instances and only route traffic to instances that pass the health check. If an instance fails the health check, it is automatically removed from the load balancer. Customize the health check to meet your specific needs.

Configuration Options:

Ping Protocol: HTTP

Ping Port: 80

Ping Path: /

Advanced Options:

Response Timeout:	5 Seconds	Time to wait when receiving a response from the health check (2 sec - 60 sec).
Health Check Interval:	0.1 Minutes	Amount of time between health checks (0.1 min - 5 min)
Unhealthy Threshold:	2 3 4 5 6 7 8 9 10	Number of consecutive health check failures before declaring an EC2 instance unhealthy.
Healthy Threshold:	2 3 4 5 6 7 8 9 10	Number of consecutive health check successes before declaring an EC2 instance healthy.

‹ Back Continue

Select Instances

Create a New Load Balancer Cancel

DEFINE LOAD BALANCER ✓ — CONFIGURE HEALTH CHECK ✓ — ADD EC2 INSTANCES ● — REVIEW

The table below lists all your running EC2 Instances that are not already behind another load balancer or part of an auto-scaling capacity group. Check the boxes in the Select column to add those instances to this load balancer.

Manually Add Instances to Load Balancer:

Select	Instance	State	Security Groups	Availability Zone
☐	i-5506d939	running	default	us-east-1c
☐	i-2771624b	running	default	us-east-1c
☐	i-2171624d	running	default	us-east-1c
☐	i-2371624f	running	default	us-east-1c
☐	i-3d716251	running	default	us-east-1c
☐	i-3f716253	running	default	us-east-1c

select all | select none

Availability Zone Distribution:

No instances selected

‹ Back Continue

Verify ELB details

Create a New Load Balancer Cancel

DEFINE LOAD BALANCER ✓ | CONFIGURE HEALTH CHECK ✓ | ADD EC2 INSTANCES ✓ | REVIEW

DEFINE LOAD BALANCER

Load Balancer Name: my-load-balancer
Port Configuration: 80 forwarding to 80 (HTTP)

Edit Load Balancer Definition

CONFIGURE HEALTH CHECK

Ping Target: HTTP:80:/
Timeout: 5
Interval: 0.1

Unhealthy Threshold: 2
Healthy Threshold: 3

Edit Health Check

ADD EC2 INSTANCES

EC2 Instances: No instances

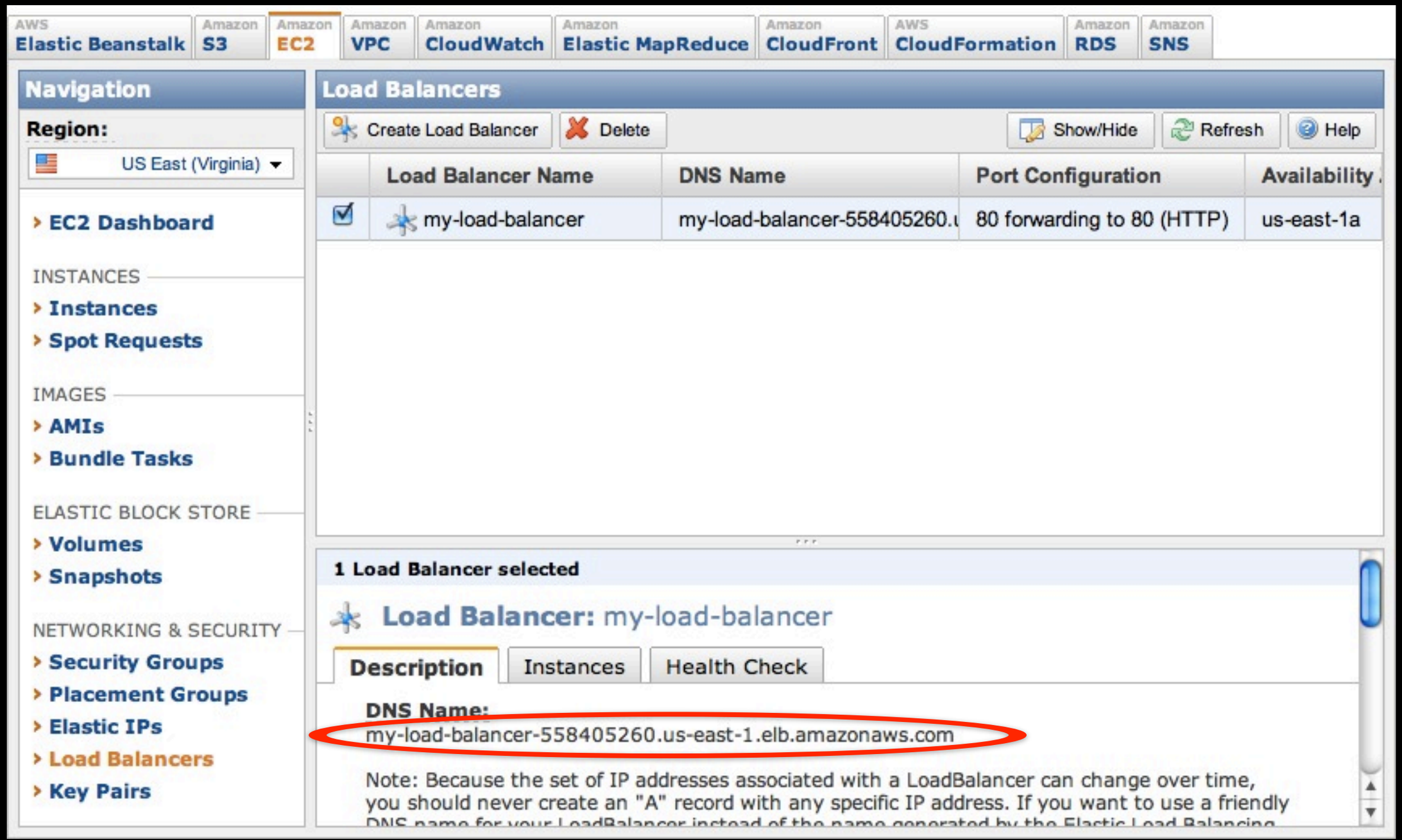
Edit EC2 Instance Selection

‹ Back

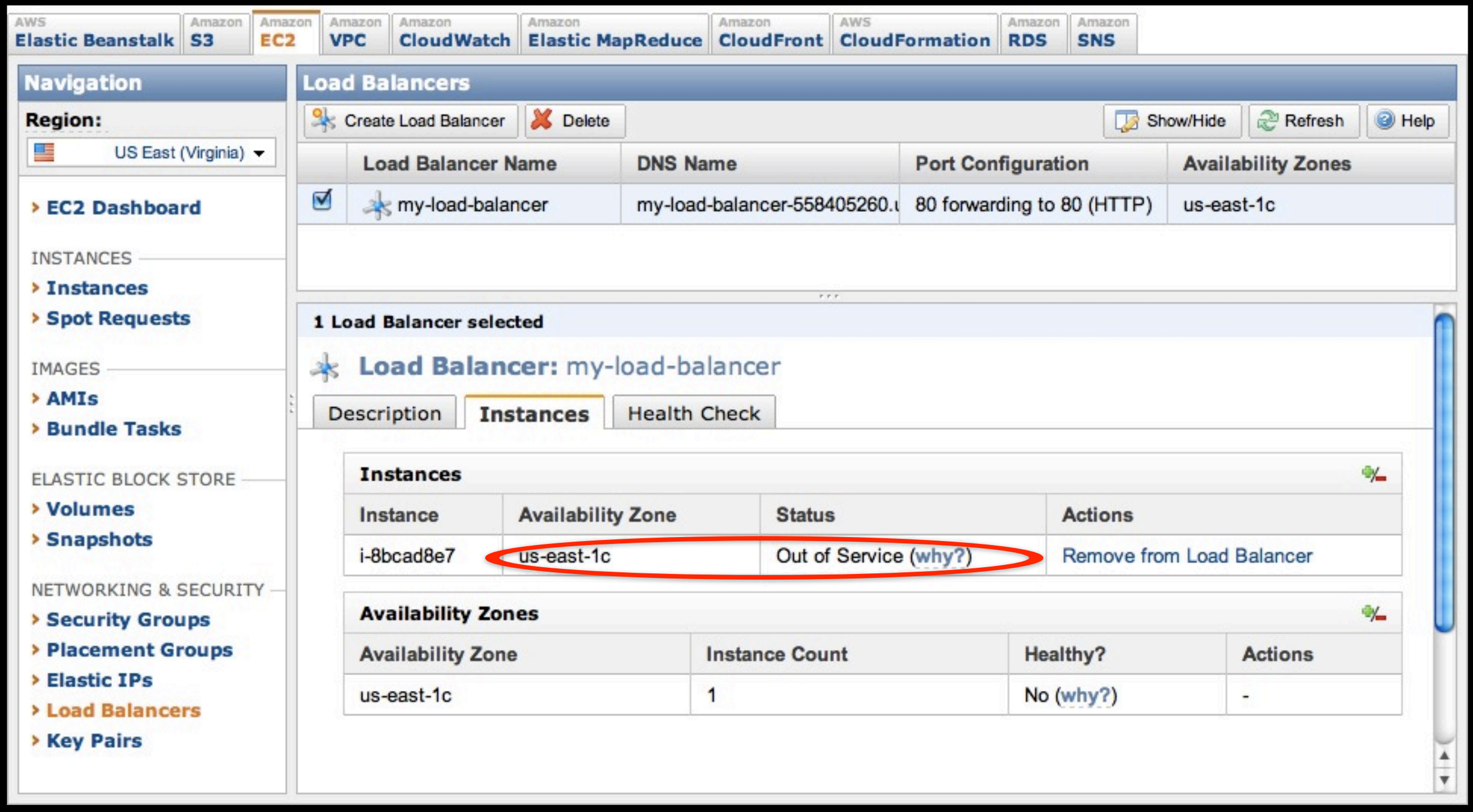
Create

Please review your selections on this page. Clicking "Create" will launch your load balancer. Check the Amazon EC2 product page for load balancer pricing info

Get DNS name


AWS Elastic Beanstalk
Amazon S3
Amazon EC2
Amazon VPC
Amazon CloudWatch
Amazon Elastic MapReduce
Amazon CloudFront
AWS CloudFormation
Amazon RDS
Amazon SNS
Navigation
Region:
US East (Virginia)
EC2 Dashboard
INSTANCES
Instances
Spot Requests
IMAGES
AMIs
Bundle Tasks
ELASTIC BLOCK STORE
Volumes
Snapshots
NETWORKING & SECURITY
Security Groups
Placement Groups
Elastic IPs
Load Balancers
Key Pairs
Load Balancers
Create Load Balancer
Delete
Show/Hide
Refresh
Help
Load Balancer Name
DNS Name
Port Configuration
Availability
my-load-balancer
my-load-balancer-558405260.
80 forwarding to 80 (HTTP)
us-east-1a
1 Load Balancer selected
Load Balancer: my-load-balancer
Description
Instances
Health Check
DNS Name:
my-load-balancer-558405260.us-east-1.elb.amazonaws.com
Note: Because the set of IP addresses associated with a LoadBalancer can change over time,
you should never create an "A" record with any specific IP address. If you want to use a friendly

Wait for health checks



Checkpoint

- Wait for load balancer to add instances
- Browse through load balancer URL

Shut down

EVERYTHING!

Questions?

- @offbytwo
- http://offbytwo.com