Instance Dynamic Registration and Dashboards

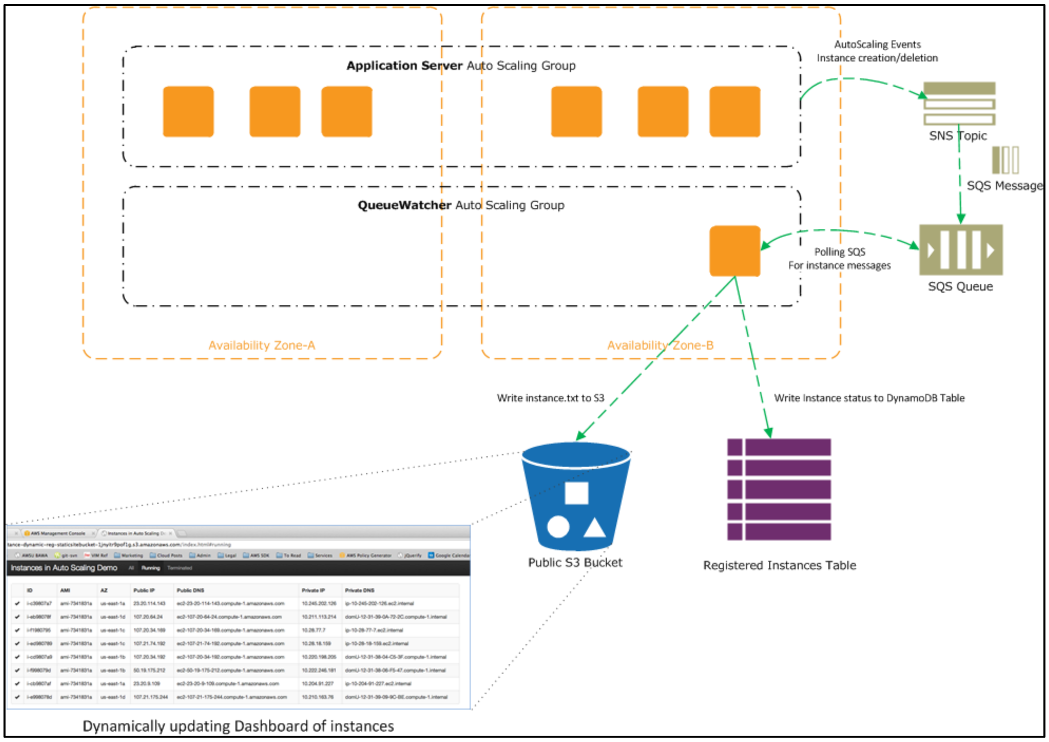
Table of Contents

Introduction

Dynamic Registration

Dynamic registration is the process whereby a resource is automatically registered with a service after it is provisioned. Examples of registration services may include Active Directory, LDAP, DNS, or DynamoDB.

Automatically registering Amazon EC2 instances after they are provisioned can make it easier to dynamically discover those Amazon EC2 instances in your environment. Instances that are created as part of an Auto Scaling group can take advantage of Auto Scaling Notifications. When Amazon EC2 instances are added or removed Auto Scaling publishes notifications to an SNS topic. Since an SNS topic can publish message to Simple Queue Service queues, it is possible for Auto Scaling to publish notifications about new or terminated instances to a durable queue like Amazon SQS. These messages in Amazon SQS can then be processed and registered with services like DynamoDB.


Application Server Auto Scaling Group
QueueWatcher Auto Scaling Group
Availability Zone-A
Availability Zone-B
AutoScaling Events
Instance creation/deletion
SNS Topic
SQS Message
Polling SQS
For instance messages
SQS Queue
Write instance.txt to S3
Write Instance status to DynamoDB Table
Public S3 Bucket
Registered Instances Table
AWS Management Console
Instances in Auto Scaling De
tance-dynamic-reg-staticsitebucket-1jnyitr9pof1g.s3.amazonaws.com/index.html#running
AWSU BAWA
git-svn
VIM Ref
Marketing
Cloud Posts
Admin
Legal
AWS SDK
To Read
Services
AWS Policy Generator
jQuerify
Google Calenda
Instances in Auto Scaling Demo
All
Running
Terminated
Dynamically updating Dashboard of instances

	ID	AMI	AZ	Public IP	Public DNS	Private IP	Private DNS
✓	i-c39807a7	ami-7341801a	us-east-1a	23.20.114.143	ec2-23-20-114-143.compute-1.amazonaws.com	10.245.202.126	ip-10-245-202-126.ec2.internal
✓	i-eb98078f	ami-7341801a	us-east-1d	107.20.64.24	ec2-107-20-64-24.compute-1.amazonaws.com	10.211.113.214	domU-12-31-39-0A-72-2C.compute-1.internal
✓	i-f1980795	ami-7341801a	us-east-1c	107.20.34.169	ec2-107-20-34-169.compute-1.amazonaws.com	10.28.77.7	ip-10-28-77-7.ec2.internal
✓	i-ed980789	ami-7341801a	us-east-1c	107.21.74.192	ec2-107-21-74-192.compute-1.amazonaws.com	10.28.18.159	ip-10-28-18-159.ec2.internal
✓	i-cd9807a9	ami-7341801a	us-east-1b	107.20.34.192	ec2-107-20-34-192.compute-1.amazonaws.com	10.220.198.205	domU-12-31-38-04-C5-3F.compute-1.internal
✓	i-f998079d	ami-7341801a	us-east-1b	50.19.175.212	ec2-50-19-175-212.compute-1.amazonaws.com	10.222.246.181	domU-12-31-38-06-F5-47.compute-1.internal
✓	i-cb9807af	ami-7341801a	us-east-1a	23.20.9.109	ec2-23-20-9-109.compute-1.amazonaws.com	10.204.91.227	ip-10-204-91-227.ec2.internal
✓	i-e998078d	ami-7341801a	us-east-1d	107.21.175.244	ec2-107-21-175-244.compute-1.amazonaws.com	10.210.163.76	domU-12-31-39-09-9C-BE.compute-1.internal

Assessment

- What is Dynamic Registration?[i]
- How can you be notified about instances being created in an Auto Scaling group?[ii]
- What is bootstrapping?[iii]
- What is instance meta-data and user-data?[iv]
- What is CloudInit?[v]

Lab Learning Objectives

This lab explores how to create discoverable endpoints in a scalable architecture. In this lab session, you will learn how to:

- Use Amazon SNS with Auto Scaling to send notifications when scaling occurs.
- Push Amazon SNS notifications into Amazon SQS
- Manually inspect the contents of an Amazon SQS queue in the console
- Use Amazon S3 to create simple low-cost administration interfaces by hosting a static website with dynamic data and updates from DynamoDB

At the end of this lab you'll have an administration console that displays the end-points that have been registered by the instances in your Auto Scaling group. This console gives a viewport on the DynamoDB table that contains this information, and this table could be used by other instances in your application to discover end-points dynamically.

Technical Knowledge Prerequisites

To successfully complete this lab, you will need to be able to have some experience and understanding of:

- Amazon EC2 (Amazon Elastic Compute Cloud) (http://aws.amazon.com/ec2/)
- Amazon S3 (Simple Storage Service) (http://aws.amazon.com/s3/)

You do not need to have hands on experience of these services, as this lab follows a guided approach – with a series of steps for you to complete using the AWS Web Console.

Overview and Essentials

Elasticity and Scalability

There are two major concepts to understand in this lab:

- An **elastic** environment is **highly utilized** all the time, by **just-in-time provisioning** of required resources.
- A **scalable** architecture **accommodates on-demand elastic growth** without changing the design.

An example of applying these principles can be the web tier within an architecture:

- **Elasticity**: add instances to the web tier as some metric increases, for example CPU. Remove instances from the web tier as these metrics decrease. This gives us an architecture that scales with demand and minimizes cost.
- **Scalable**: use horizontal scaling – adding and removing instances of the **same size,** rather than varying the size of instance. This gives an architecture that scales in a consistent manner, and should not require redesign as workloads increase.

Auto Scaling

Auto Scaling allows you to scale your Amazon EC2 capacity up or down automatically according to conditions you define. With Auto Scaling, you can ensure that the number of Amazon EC2 instances you're using increases seamlessly during demand spikes to maintain performance, and decreases automatically during demand lulls to minimize costs.

Auto Scaling can be configured to send a message (http://docs.amazonwebservices.com/AutoScaling/latest/DeveloperGuide/ASGettingNotifications.html) to an Amazon SNS topic whenever an instance is launched or terminated.

Bootstrapping

Bootstrapping is the process of automatically setting up your servers after they boot. This can be to add additional software components to an instance, deploy the latest version of a binary, to acquire resources (attach volumes or EIPs) or to register the instance with a central registry.

Meta-Data and User-Data

Amazon EC2 instances can access instance-specific metadata (http://docs.amazonwebservices.com/AWSEC2/latest/UserGuide/AESDG-chapter-instancedata.html#instancedata-data-categories), as well as data supplied when launching the instances (User-Data), using a specific URI such as http:// 169.254.169.254/ latest/meta-data/ and http:// 169.254.169.254/ latest/user-data/).

You can script your instance through bootstrapping to retrieve this meta-data and use it to configure itself. For example the user-data could contain the location in Amazon S3 of the latest build of a website, your script could download this onto your instance. Thus you can change which build the instance uses by changing its startup user-data.

CloudInit

To give greater flexibility you can combine bootstrapping with meta-data: a script can be passed to the instance via meta-data which can then be run as the bootstrap action. A standard way to do this is via CloudInit (https://help.ubuntu.com/community/CloudInit) an open source package from Canonical.

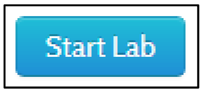
AWS CloudFormation

AWS CloudFormation enables you to create and delete related AWS resources (Instances, Auto Scaling groups, DynamoDB tables etc) together as a unit called a stack. You define the characteristics of a stack using a template (a JSON text file). You can write your template from scratch, or start with one of the example templates. http://docs.amazonwebservices.com/AWSCloudFormation/latest/UserGuide/example-templates.html

AWS CloudFormation stacks can then be executed to create or update your stack, and it also provides **cfn-init** (https://s3.amazonaws.com/cloudformation-examples/BoostrappingApplicationsWithAWSCloudFormation.pdf) the Amazon CloudFormation version of CloudInit for instances launched from Amazon CloudFormation stacks.

Start your *qwikLAB™*

1. Start your *qwikLAB™*
   Use the 'Start Lab' button to start your lab.
   (Hint: If you are prompted for a token, please use one you've been given or have purchased.)



   You will see the lab creation in progress.

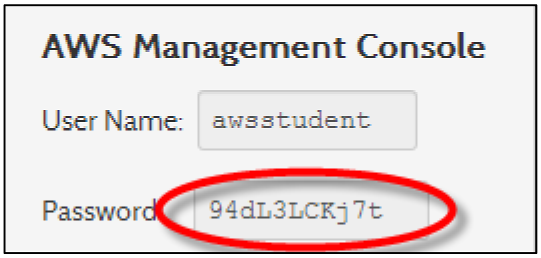
Create in progress...

2. Note a few properties of the lab.

   a. **Duration -** The time the lab will run for before shutting itself down.
   b. **Setup Time -** The estimated lab creation time on starting the lab.
   c. **AWS Region** - The AWS Region the lab resources are being created in.

3. Copy the Password provided.

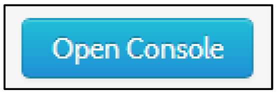
   d. Hint: selecting the value shown and using Ctrl+C works best


AWS Management Console

User Name: awsstudent

Password 94dL3LCKj7t

4. Click the 'Open Console' button.

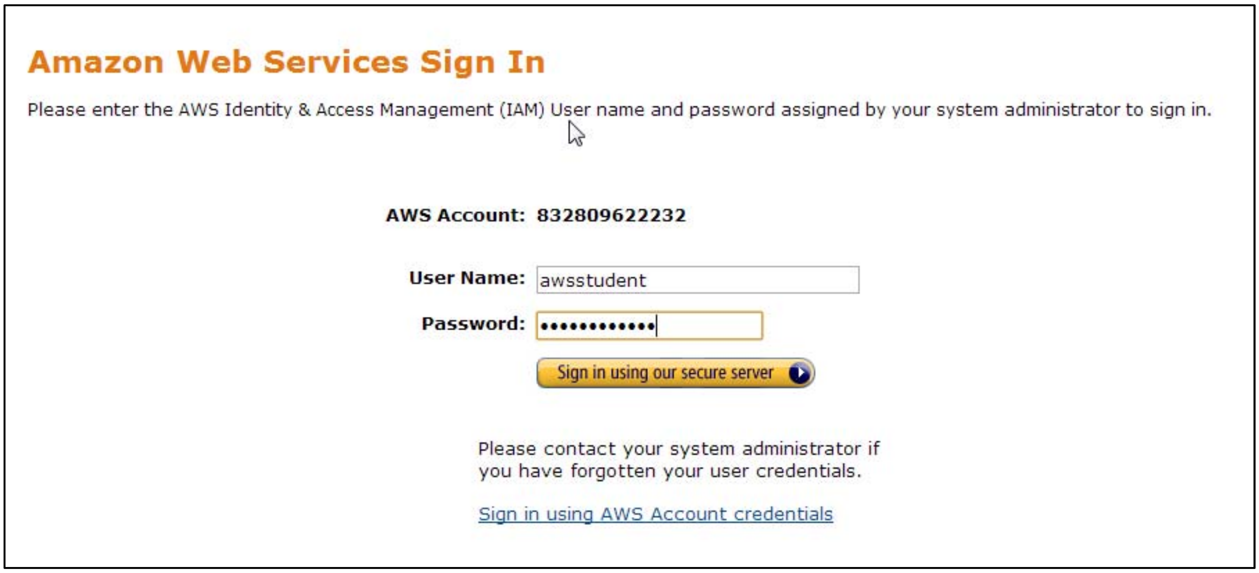


5. Login to the AWS Management Console

   Enter the User Name '**awsstudent**' and paste the password you copied from the lab details in *qwikLAB™* into the Password field.

   Click on the 'Sign in using our secure server' button.

   In this step you logged into the AWS Management Console using login credentials for a user provisioned via AWS Identity Access Management in an AWS account by *qwikLAB™*.



Amazon Web Services Sign In

Please enter the AWS Identity & Access Management (IAM) User name and password assigned by your system administrator to sign in.

AWS Account: 832809622232

User Name: awsstudent

Password: ••••••••••••

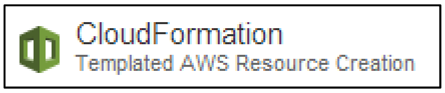
Sign in using our secure server

Please contact your system administrator if you have forgotten your user credentials.

Sign in using AWS Account credentials

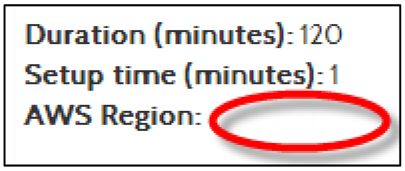
AWS Management Console

6. Select "CloudFormation" from the Console Home.



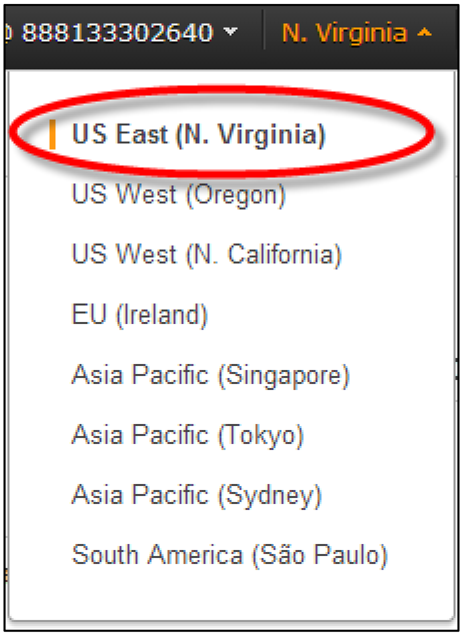
Confirm your AWS Region

7. Note the AWS Region set for your lab in *qwikLAB™*



Duration (minutes): 120
Setup time (minutes): 1
AWS Region:

8. Select or confirm that the same AWS Region is already set in the AWS Management Console



Hands-on

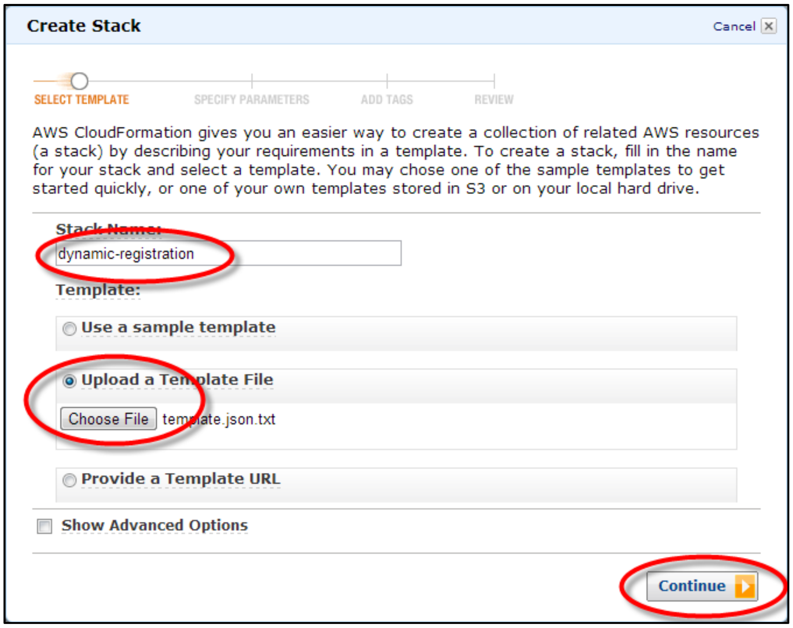
Launch Amazon EC2 Instances in an Auto Scaling Group

This exercise uses a AWS CloudFormation template to launch a stack that contains an Auto Scaling Group. The Auto Scaling group is configured to publish notifications to an Amazon SQS queue (via an Amazon SNS topic) when Amazon EC2 instances are added or removed to the Group.

1. Download the Amazon CloudFormation template https://us-east-1-aws-training.s3.amazonaws.com/self-paced-lab-8/template.json for this exercise to your computer. Remember the path to where this file is downloaded.

2. Click on "Create New Stack".

   a. In the Stack Name field, enter "`dynamic-registration`"
   b. In the Stack Template Source field, click Upload a Template File and choose the file you downloaded in the first step
   c. Click the "Continue" button



Create Stack Cancel ☒

SELECT TEMPLATE SPECIFY PARAMETERS ADD TAGS REVIEW

AWS CloudFormation gives you an easier way to create a collection of related AWS resources (a stack) by describing your requirements in a template. To create a stack, fill in the name for your stack and select a template. You may chose one of the sample templates to get started quickly, or one of your own templates stored in S3 or on your local hard drive.

Stack Name: dynamic-registration

Template:

○ Use a sample template

● Upload a Template File

Choose File template.json.txt

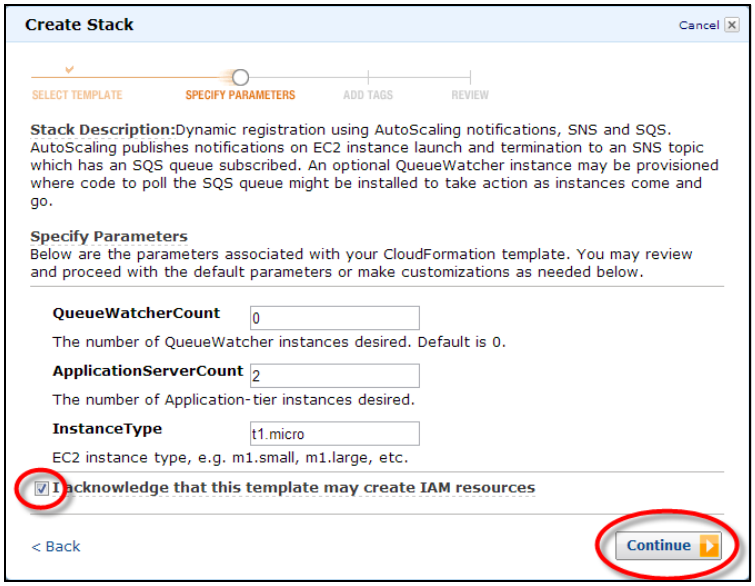
○ Provide a Template URL

□ Show Advanced Options

Continue

3. Leave the `QueueWatcherCount` field with value `0`

4. Tick the “I acknowledge that this template may create IAM resources” checkbox, then click “Continue”



Stack Description: Dynamic registration using AutoScaling notifications, SNS and SQS. AutoScaling publishes notifications on EC2 instance launch and termination to an SNS topic which has an SQS queue subscribed. An optional QueueWatcher instance may be provisioned where code to poll the SQS queue might be installed to take action as instances come and go.

Specify Parameters
Below are the parameters associated with your CloudFormation template. You may review and proceed with the default parameters or make customizations as needed below.

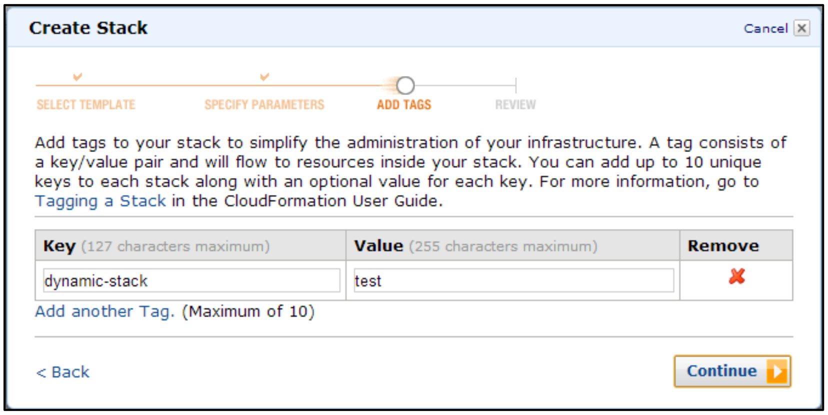
QueueWatcherCount 0
The number of QueueWatcher instances desired. Default is 0.

ApplicationServerCount 2
The number of Application-tier instances desired.

InstanceType t1.micro
EC2 instance type, e.g. m1.small, m1.large, etc.

☑ **I acknowledge that this template may create IAM resources**

5. When creating a stack, you can add arbitrary key/value tags to identify it for purposes such as cost allocation. For complete information about what tags are and how they can be used, see Tagging Your Resources in the *Amazon EC2 User Guide:* (*http://docs.amazonwebservices.com/AWSEC2/latest/UserGuide/Using_Tags.html*)
In this example we made a stack variable and identified it as test. Click Continue



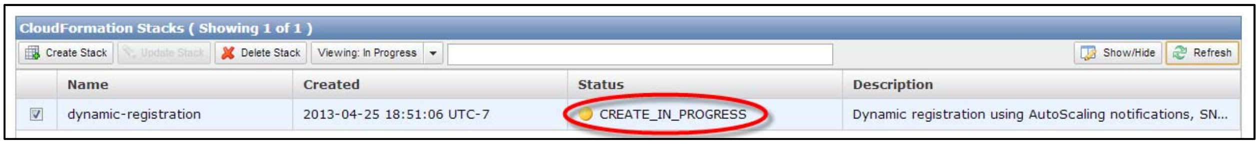
Add tags to your stack to simplify the administration of your infrastructure. A tag consists of a key/value pair and will flow to resources inside your stack. You can add up to 10 unique keys to each stack along with an optional value for each key. For more information, go to Tagging a Stack in the CloudFormation User Guide.

Key (127 characters maximum)	**Value** (255 characters maximum)	**Remove**
dynamic-stack	test	✖

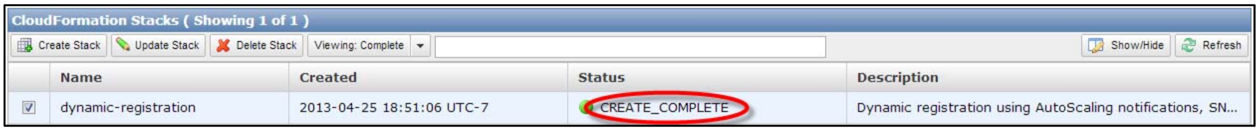
Add another Tag. (Maximum of 10)

6. In the Review step, click “Continue”, then click “Close”. A stack named “dynamic-registration” should appear with a yellow CREATE_IN_PROGRESS status.

7. Click the console Refresh button (top-right of console) occasionally until stack status changes to a CREATE_COMPLETE. This should complete in less than 10 minutes. You may monitor progress in the Events Tab in the lower half.



	Name	Created	Status	Description
☑	dynamic-registration	2013-04-25 18:51:06 UTC-7	● CREATE_IN_PROGRESS	Dynamic registration using AutoScaling notifications, SN...

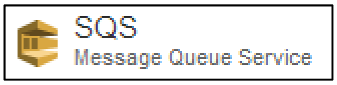


	Name	Created	Status	Description
☑	dynamic-registration	2013-04-25 18:51:06 UTC-7	● CREATE_COMPLETE	Dynamic registration using AutoScaling notifications, SN...

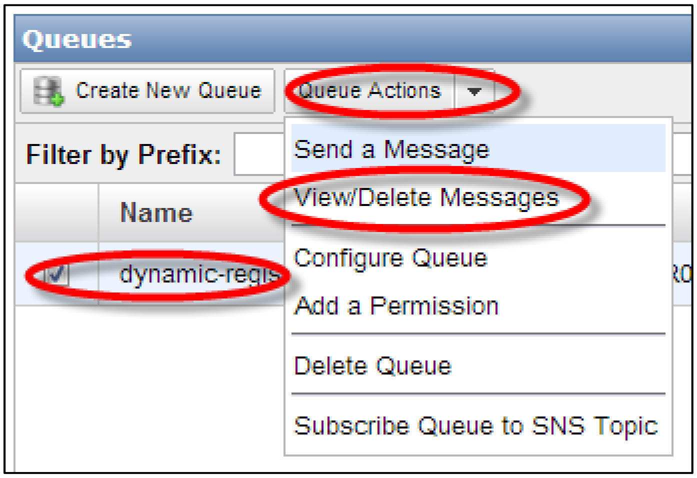
Poll an Amazon SQS Queue for Messages

As the Auto Scaling Group you created in the first task adds and removes Amazon EC2 instances, it will publish notifications to an Amazon SNS topic. In addition to sending notifications of such activity to your e-mail address, Amazon SNS also publishes messages to a queue, where they are durably stored and accessible via other services.

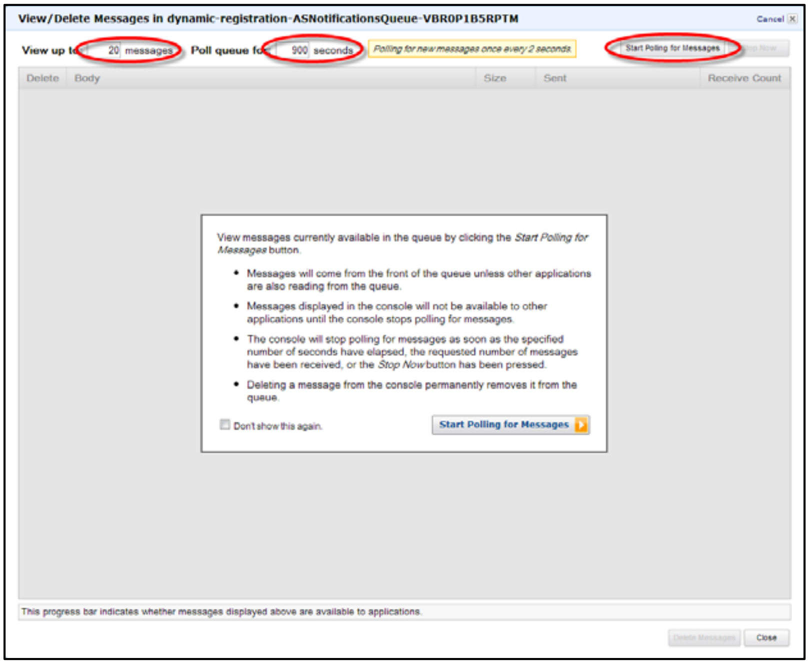
1. Navigate to the SQS console



2. Locate and select the queue created by the Amazon CloudFormation stack in the first task. The queue name should begin with “dynamic-registration”. Click Queue Actions, then View/Delete Messages.



3. A popup appears. Set the following values in the window that appears:
   a. View up to 20 messages
   b. Poll queue for 900 seconds
   c. Click Start Polling for Messages


View/Delete Messages in dynamic-registration-ASNotificationsQueue-VBR0P1B5RPTM

View up to 20 messages Poll queue for 900 seconds

Polling for new messages once every 2 seconds.

Delete	Body	Size	Sent	Receive Count

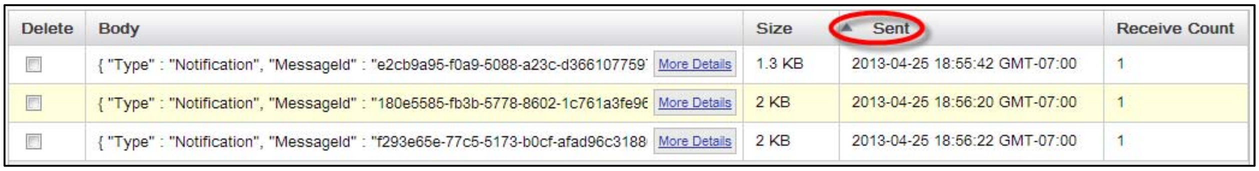
View messages currently available in the queue by clicking the *Start Polling for Messages* button.

- Messages will come from the front of the queue unless other applications are also reading from the queue.
- Messages displayed in the console will not be available to other applications until the console stops polling for messages.
- The console will stop polling for messages as soon as the specified number of seconds have elapsed, the requested number of messages have been received, or the *Stop Now* button has been pressed.
- Deleting a message from the console permanently removes it from the queue.

□ Don't show this again.

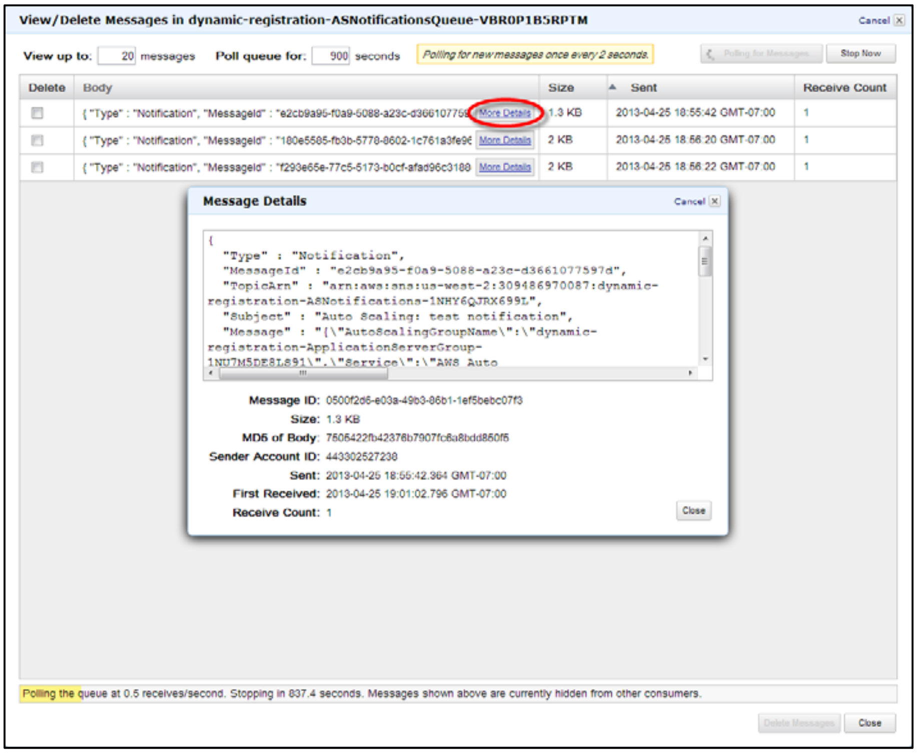
This progress bar indicates whether messages displayed above are available to applications.

4. Click the Sent column header to sort incoming messages according to when they were sent. Several messages should appear in your queue.



Delete	Body	Size	▲ Sent	Receive Count
□	{ "Type" : "Notification", "MessageId" : "e2cb9a95-f0a9-5088-a23c-d366107759 More Details	1.3 KB	2013-04-25 18:55:42 GMT-07:00	1
□	{ "Type" : "Notification", "MessageId" : "180e5585-fb3b-5778-8602-1c761a3fe96 More Details	2 KB	2013-04-25 18:56:20 GMT-07:00	1
□	{ "Type" : "Notification", "MessageId" : "f293e65e-77c5-5173-b0cf-afad96c3188 More Details	2 KB	2013-04-25 18:56:22 GMT-07:00	1

5. Click More Details on several of the messages. What is contained in the messages? Essentially, information about the instance(s) that were launched. Could we write a simple service to parse these messages? Yes



View/Delete Messages in dynamic-registration-ASNotificationsQueue-VBR0P1B5RPTM

View up to: 20 messages Poll queue for: 900 seconds *Polling for new messages once every 2 seconds.*

Delete	Body	Size	▲ Sent	Receive Count
□	{ "Type" : "Notification", "MessageId" : "e2cb9a95-f0a9-5088-a23c-d366107759 More Details	1.3 KB	2013-04-25 18:55:42 GMT-07:00	1
□	{ "Type" : "Notification", "MessageId" : "180e5585-fb3b-5778-8602-1c761a3fe96 More Details	2 KB	2013-04-25 18:56:20 GMT-07:00	1
□	{ "Type" : "Notification", "MessageId" : "f293e65e-77c5-5173-b0cf-afad96c3188 More Details	2 KB	2013-04-25 18:56:22 GMT-07:00	1

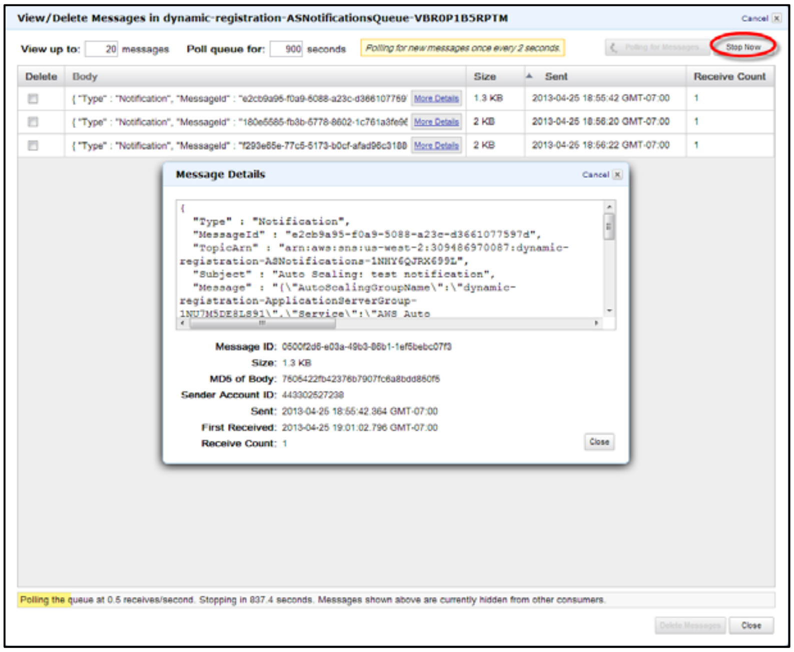
Message Details

```
{
  "Type" : "Notification",
  "MessageId" : "e2cb9a95-f0a9-5088-a23c-d3661077597d",
  "TopicArn" : "arn:aws:sns:us-west-2:309486970087:dynamic-
registration-ASNotifications-1NHY6QJRX699L",
  "Subject" : "Auto Scaling: test notification",
  "Message" : "{\"AutoScalingGroupName\":\"dynamic-
registration-ApplicationServerGroup-
1NU7M5DE8L891\",\"Service\":\"AWS Auto
```

Message ID: 0500f2d6-e03a-49b3-86b1-1ef5bebc07f3
Size: 1.3 KB
MD5 of Body: 7505422fb42376b7907fc8a8bdd850f5
Sender Account ID: 443302527238
Sent: 2013-04-25 18:55:42.364 GMT-07:00
First Received: 2013-04-25 19:01:02.796 GMT-07:00
Receive Count: 1

Polling the queue at 0.5 receives/second. Stopping in 837.4 seconds. Messages shown above are currently hidden from other consumers.

6. After viewing the messages, stop polling and return the messages to the queue by clicking “Stop Now” and then “Close”



View/Delete Messages in dynamic-registration-ASNotificationsQueue-VBR0P1B5RPTM

View up to: 20 messages Poll queue for: 900 seconds *Polling for new messages once every 2 seconds.*

Delete	Body	Size	Sent	Receive Count
□	{ "Type" : "Notification", "MessageId" : "e2cb9a95-f0a9-5088-a23c-d366107769' More Details	1.3 KB	2013-04-25 18:55:42 GMT-07:00	1
□	{ "Type" : "Notification", "MessageId" : "180e5585-fb3b-5778-8602-1c761a3fe9€ More Details	2 KB	2013-04-25 18:56:20 GMT-07:00	1
□	{ "Type" : "Notification", "MessageId" : "f293e65e-77c5-5173-b0cf-afad96c3188 More Details	2 KB	2013-04-25 18:56:22 GMT-07:00	1

Message Details

```
{
  "Type" : "Notification",
  "MessageId" : "e2cb9a95-f0a9-5088-a23c-d3661077597d",
  "TopicArn" : "arn:aws:sns:us-west-2:309486970087:dynamic-
registration-ASNotifications-1NHY6QJRX699L",
  "Subject" : "Auto Scaling: test notification",
  "Message" : "{\"AutoScalingGroupName\":\"dynamic-
registration-ApplicationServerGroup-
1NU7M5DE8L891\",\"Service\":\"AWS Auto
```

Message ID: 0500f2d6-e03a-49b3-86b1-1ef5bebc07f3
Size: 1.3 KB
MD5 of Body: 7605422fb42376b7907fc6a8bdd860f6
Sender Account ID: 443302527238
Sent: 2013-04-25 18:55:42.364 GMT-07:00
First Received: 2013-04-25 19:01:02.796 GMT-07:00
Receive Count: 1

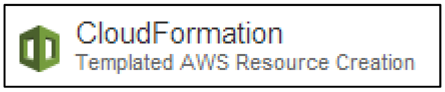
Polling the queue at 0.5 receives/second. Stopping in 837.4 seconds. Messages shown above are currently hidden from other consumers.

Launch a QueueWatcher to Process Notifications

At this point we've seen how Auto Scaling publishes notifications when Amazon EC2 instances are launched or terminated in an Auto Scaling Group. The Amazon SNS Topic that Auto Scaling is publishing messages to is storing those messages in an Amazon SQS queue. This makes it possible to compose a simple script using one of the AWS SDKs to poll Amazon SQS for those messages and store them in a queryable location (e.g., Dynamo DB, DNS, etc...)

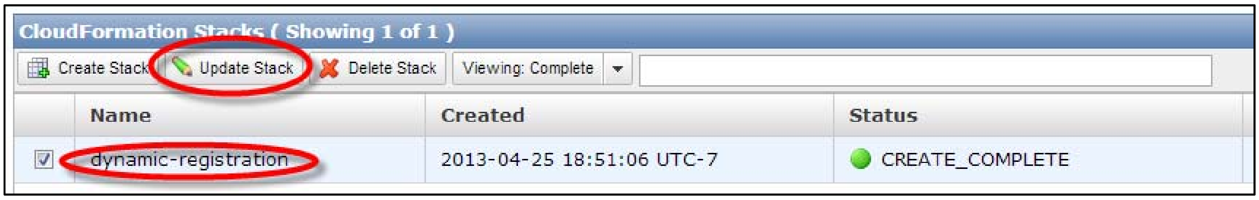
For this exercise, we will update the stack created in the first section to launch a t1.micro Amazon EC2 instance that we will refer to as a QueueWatcher. The QueueWatcher instance contains a small Python script that polls Amazon SQS for messages about launched or terminated instances. It then stores important information about each instance (including ID, Availability Zone, and public/private IP addresses) in a DynamoDB table. Finally, the QueueWatcher periodically writes a simple text file to Amazon S3 that contains information about the instances from the DynamoDB table. This text file can be viewed in a web browser to see a simple presentation of the activity in the Auto Scaling Group.

1. To launch a QueueWatcher instance, first navigate to the Amazon CloudFormation section of the AWS Management Console



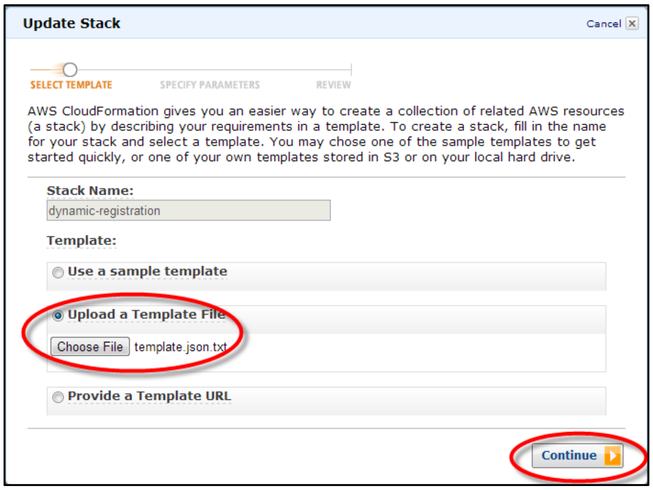
CloudFormation
Templated AWS Resource Creation

2. Select your stack (dynamic-registration) and click Update Stack



	Name	Created	Status
☑	dynamic-registration	2013-04-25 18:51:06 UTC-7	● CREATE_COMPLETE

3. Select Upload a Template File, then choose the Amazon CloudFormation template downloaded at the beginning of this exercise. Click Continue.



AWS CloudFormation gives you an easier way to create a collection of related AWS resources (a stack) by describing your requirements in a template. To create a stack, fill in the name for your stack and select a template. You may chose one of the sample templates to get started quickly, or one of your own templates stored in S3 or on your local hard drive.

Stack Name: dynamic-registration

Template:

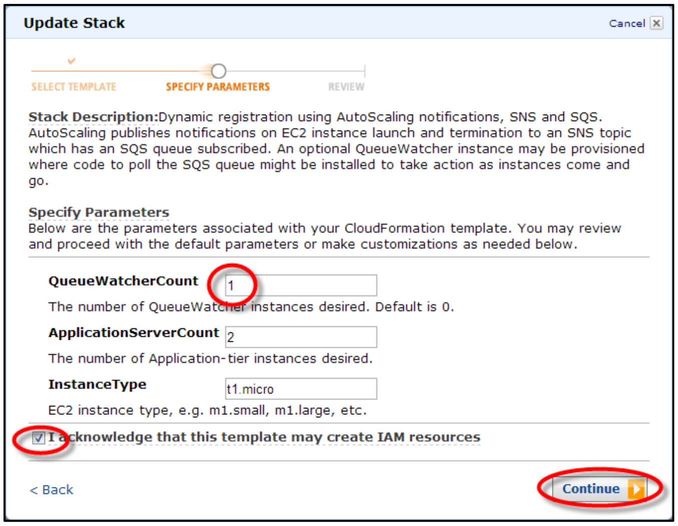
○ Use a sample template

● Upload a Template File

Choose File template.json.txt

○ Provide a Template URL

4. Change the value of QueueWatcherCount from 0 to 1. Tick the I acknowledge that this template may create IAM resources checkbox, then click Continue



Stack Description:Dynamic registration using AutoScaling notifications, SNS and SQS. AutoScaling publishes notifications on EC2 instance launch and termination to an SNS topic which has an SQS queue subscribed. An optional QueueWatcher instance may be provisioned where code to poll the SQS queue might be installed to take action as instances come and go.

Specify Parameters

Below are the parameters associated with your CloudFormation template. You may review and proceed with the default parameters or make customizations as needed below.

QueueWatcherCount 1

The number of QueueWatcher instances desired. Default is 0.

ApplicationServerCount 2

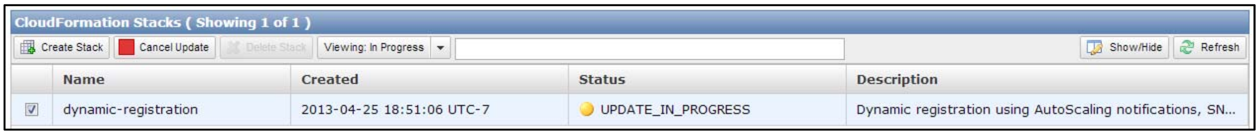
The number of Application-tier instances desired.

InstanceType t1.micro

EC2 instance type, e.g. m1.small, m1.large, etc.

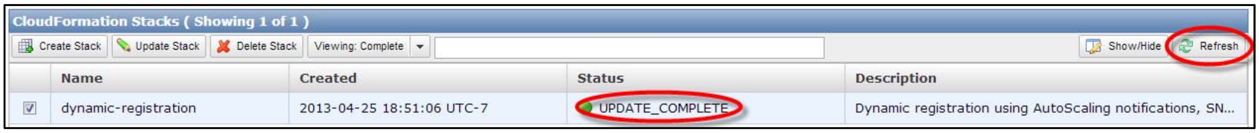
☑ I acknowledge that this template may create IAM resources

5. In the Review step, click Continue, then click Close. Your dynamic-registration stack should appear with a yellow UPDATE_IN_PROGRESS status.



	Name	Created	Status	Description
☑	dynamic-registration	2013-04-25 18:51:06 UTC-7	● UPDATE_IN_PROGRESS	Dynamic registration using AutoScaling notifications, SN...

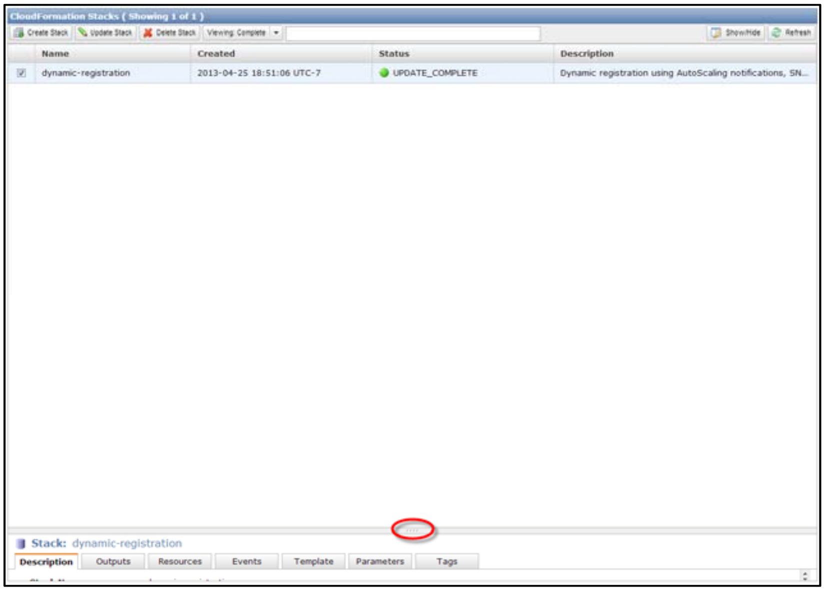
6. Click the console Refresh button (top-right of console) until stack status changes to UPDATE_COMPLETE. You may monitor progress in the Events tab in the lower half of the screen.



	Name	Created	Status	Description
☑	dynamic-registration	2013-04-25 18:51:06 UTC-7	● UPDATE_COMPLETE	Dynamic registration using AutoScaling notifications, SN...

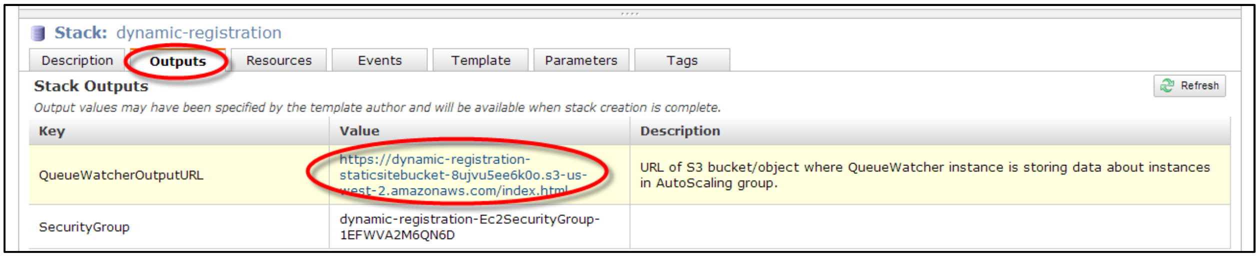
7. After the stack status changes to UPDATE_COMPLETE, wait for 2 minutes before proceeding to the next step.

8. You may need to make the bottom panel larger, by clicking and dragging the separator



	Name	Created	Status	Description
☑	dynamic-registration	2013-04-25 18:51:06 UTC-7	● UPDATE_COMPLETE	Dynamic registration using AutoScaling notifications, SN...

9. Once the update has completed, select the stack, click the Outputs tab in the bottom panel, and copy the value of the `QueueWatcherOutputURL` key.

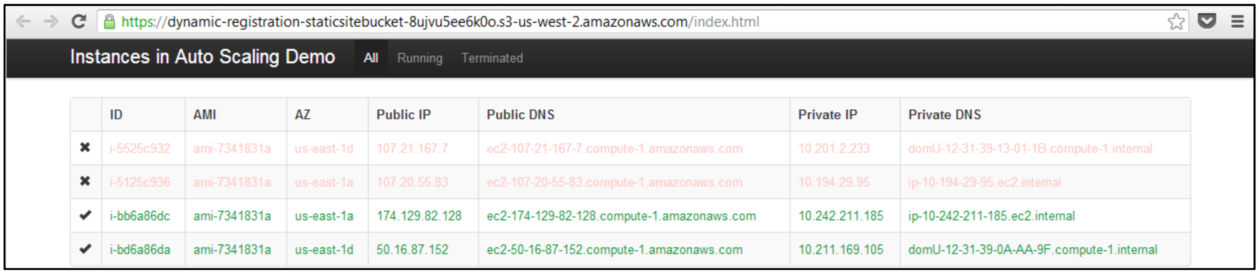


Stack Outputs

Output values may have been specified by the template author and will be available when stack creation is complete.

Key	Value	Description
QueueWatcherOutputURL	https://dynamic-registration-staticsitebucket-8ujvu5ee6k0o.s3-us-west-2.amazonaws.com/index.html	URL of S3 bucket/object where QueueWatcher instance is storing data about instances in AutoScaling group.
SecurityGroup	dynamic-registration-Ec2SecurityGroup-1EFWVA2M6QN6D	

10. Open the QueueWatcherOutputURLby pasting the value into a new browser tab or window. Keep this tab or window open for the next exercise. It is possible that the page at this URL will return an AccessDenied message. This indicates that the QueueWatcher instance hasn't yet published the first text file to Amazon S3. Refresh the page periodically until it loads.



https://dynamic-registration-staticsitebucket-8ujvu5ee6k0o.s3-us-west-2.amazonaws.com/index.html

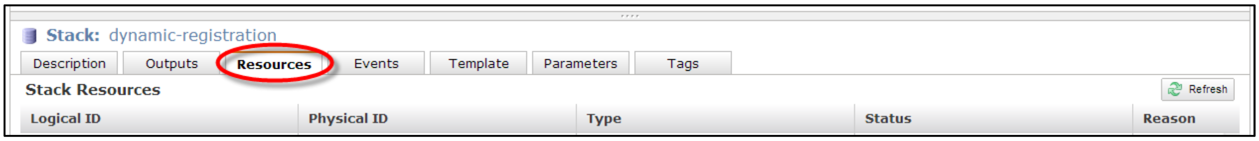
	ID	AMI	AZ	Public IP	Public DNS	Private IP	Private DNS
✖	i-5525c932	ami-7341831a	us-east-1d	107.21.167.7	ec2-107-21-167-7.compute-1.amazonaws.com	10.201.2.233	domU-12-31-39-13-01-1B.compute-1.internal
✖	i-5125c936	ami-7341831a	us-east-1a	107.20.55.83	ec2-107-20-55-83.compute-1.amazonaws.com	10.194.29.95	ip-10-194-29-95.ec2.internal
✔	i-bb6a86dc	ami-7341831a	us-east-1a	174.129.82.128	ec2-174-129-82-128.compute-1.amazonaws.com	10.242.211.185	ip-10-242-211-185.ec2.internal
✔	i-bd6a86da	ami-7341831a	us-east-1d	50.16.87.152	ec2-50-16-87-152.compute-1.amazonaws.com	10.211.169.105	domU-12-31-39-0A-AA-9F.compute-1.internal

Watch Auto Scaling, Amazon SNS and Amazon SQS in Action

Manually terminate 2 of the App Server Amazon EC2 instances that Auto Scaling launched in the first task and observe the output written by the QueueWatcher to Amazon S3. Remember, the QueueWatcher polls the Amazon SQS queue, registering and de-registering instances in the Auto Scaling Group with a DynamoDB table. It also outputs the instances registered with DynamoDB to a simple web page in Amazon S3.

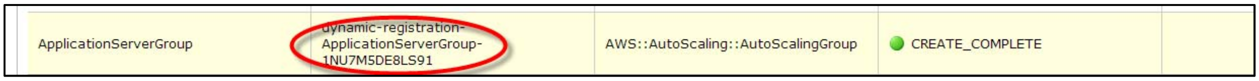
You will use the search feature of the Amazon EC2 Management Console to find the App Server instances to terminate. Amazon EC2 instances in an Auto Scaling are automatically tagged with the name of the Auto Scaling group they're in. By locating the name of your App Server Auto Scaling group, you can easily locate all of those instances in Management Console.

1. Switch back to the browser tab with the Amazon CloudFormation Management Console open
2. Single-click your stack to select it, then click the Resources tab at the bottom of the page



Logical ID	Physical ID	Type	Status	Reason

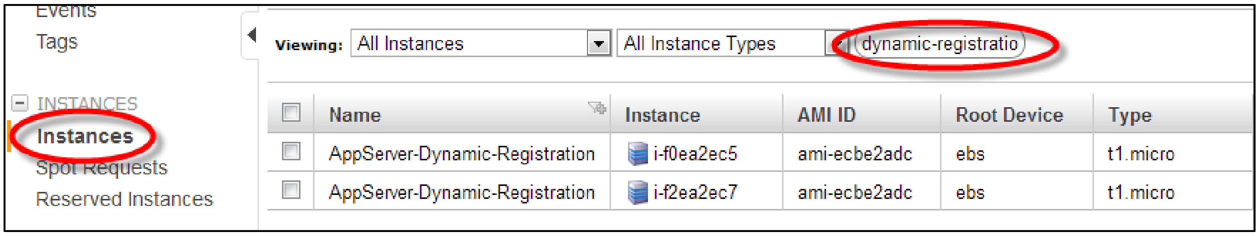
3. Find the resource with Logical ID value of ApplicationServerGroup, you may have to scroll down to find it. Copy its Physical ID to the clipboard. This is the ID of the Auto Scaling Group that your App Server instances are running in.



ApplicationServerGroup	dynamic-registration-ApplicationServerGroup-1NU7M5DE8LS91	AWS::AutoScaling::AutoScalingGroup	● CREATE_COMPLETE	

4. Navigate to the EC2 Management Console

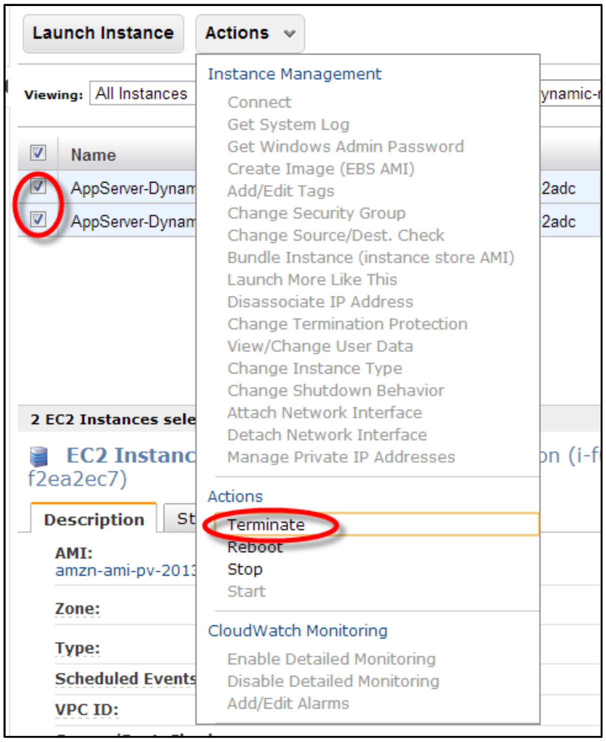
5. Click the Instances link in the left navigation column to view your running instances. In the My Instances view, paste the value you copied in the previous step into the Search field and hit return. The list of instances should be reduced, showing only the instances running in your App Server Auto Scaling Group.

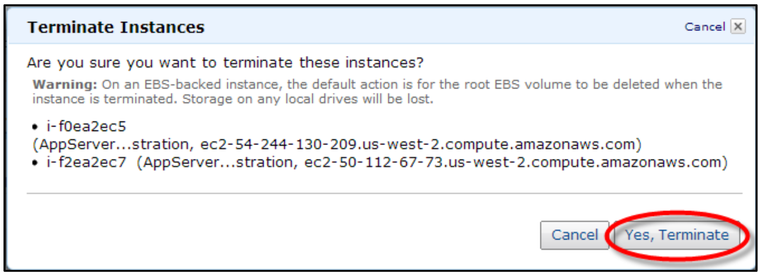


Viewing: All Instances ▾ All Instance Types ▾ dynamic-registratio

□	Name	Instance	AMI ID	Root Device	Type
□	AppServer-Dynamic-Registration	i-f0ea2ec5	ami-ecbe2adc	ebs	t1.micro
□	AppServer-Dynamic-Registration	i-f2ea2ec7	ami-ecbe2adc	ebs	t1.micro

6. Select the first 2 instances (there may only be two instances), click Instance Actions, choose Terminate, then click Yes, Terminate.





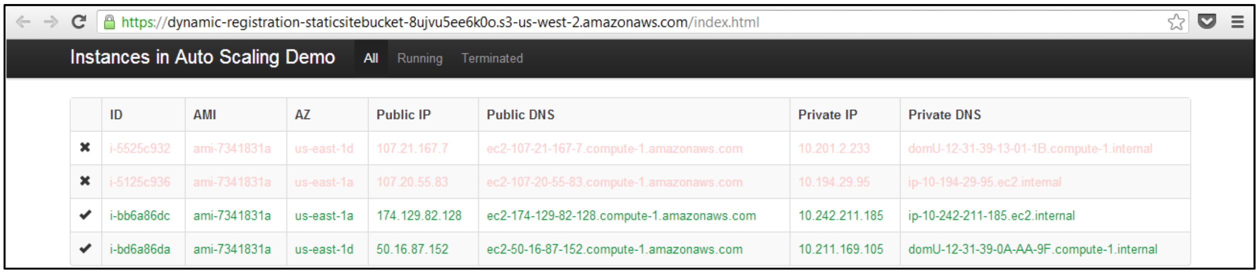
Terminate Instances Cancel ☒

Are you sure you want to terminate these instances?

Warning: On an EBS-backed instance, the default action is for the root EBS volume to be deleted when the instance is terminated. Storage on any local drives will be lost.

- i-f0ea2ec5 (AppServer...stration, ec2-54-244-130-209.us-west-2.compute.amazonaws.com)
- i-f2ea2ec7 (AppServer...stration, ec2-50-112-67-73.us-west-2.compute.amazonaws.com)

7. As the instances are terminating, switch back to the QueueWatcherOutputURL browser window or tab and observe the output (it will automatically update, so no need to hit refresh).



https://dynamic-registration-staticsitebucket-8ujvu5ee6k0o.s3-us-west-2.amazonaws.com/index.html

Instances in Auto Scaling Demo

	ID	AMI	AZ	Public IP	Public DNS	Private IP	Private DNS
✖	i-5525c932	ami-7341831a	us-east-1d	107.21.167.7	ec2-107-21-167-7.compute-1.amazonaws.com	10.201.2.233	domU-12-31-39-13-01-1B.compute-1.internal
✖	i-5125c936	ami-7341831a	us-east-1a	107.20.55.83	ec2-107-20-55-83.compute-1.amazonaws.com	10.194.29.95	ip-10-194-29-95.ec2.internal
✔	i-bb6a86dc	ami-7341831a	us-east-1a	174.129.82.128	ec2-174-129-82-128.compute-1.amazonaws.com	10.242.211.185	ip-10-242-211-185.ec2.internal
✔	i-bd6a86da	ami-7341831a	us-east-1d	50.16.87.152	ec2-50-16-87-152.compute-1.amazonaws.com	10.211.169.105	domU-12-31-39-0A-AA-9F.compute-1.internal

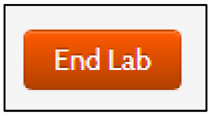
Conclusion

Congratulations! You now have successfully:

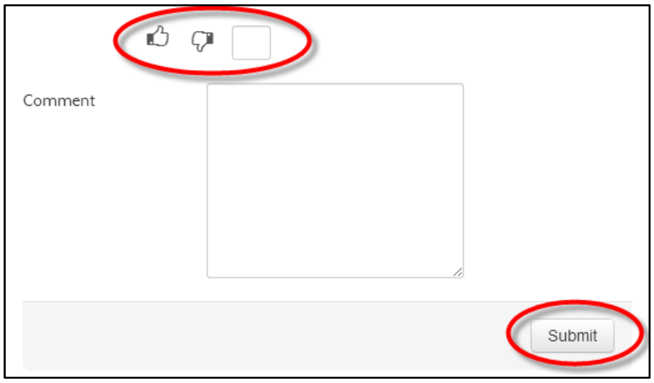
- Used Amazon CloudFormation to setup an Auto Scaling stack
- Used Auto Scaling events to send notifications when you stack scales
- Used the AWS Console to inspect the Amazon SQS queue
- Created a central repository of dynamic endpoints, allowing instances to be registered and discovered
- Seen how Amazon S3 can be used to create simple low-cost administration interfaces by hosting a static website with dynamic data and updates from DynamoDB

End Lab

1. Sign-out of the AWS Management Console.
2. Click the End Lab button in *qwikLAB™*.



3. Give the lab a thumbs-up/down, or enter a comment and click Submit



Any errors in this lab may be reported to aws-course-feedback@amazon.com.

[i] Dynamic registration is the process whereby a resource is automatically registered with a service after it is provisioned.
[ii] AutoScaling can be configured to send SNS messages when events occur.
[iii] Bootstrapping is the process of automatically setting up your servers after they boot
[iv] information about an instance (meta-data) or defined at launch (user-data), accessible from a HTTP end-point
[v] A way to pass scripts to be run when on an instance when it is launched