Auto Scaling

Getting Started Guide

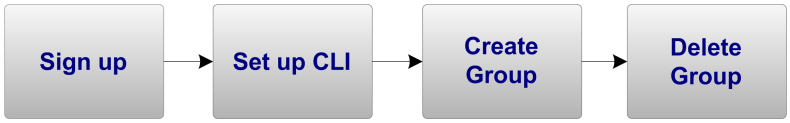
API Version 2011-01-01

Auto Scaling: Getting Started Guide

Table of Contents

Get Started with Auto Scaling

Auto Scaling is a web service designed to launch or terminate EC2 instances automatically based on user-defined policies, schedules, and health checks. You can get started with Auto Scaling by performing the tasks shown in the following diagram. You'll use the Auto Scaling Command Line Interface (CLI) to complete these tasks.

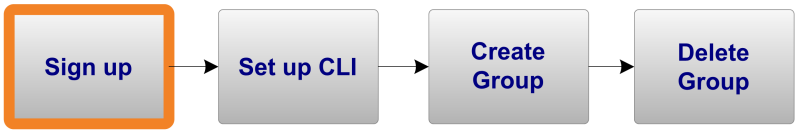


This guide walks you through a few basic Auto Scaling procedures, including setting up the CLI, creating an Auto Scaling group, and deleting it.

Note

Use of this guide assumes that you have completed the Amazon Elastic Cloud Computing Getting Started Guide.

Sign Up for Auto Scaling



To use Auto Scaling, sign up for Amazon Elastic Compute Cloud (EC2) and create an AWS account if you don't have one.

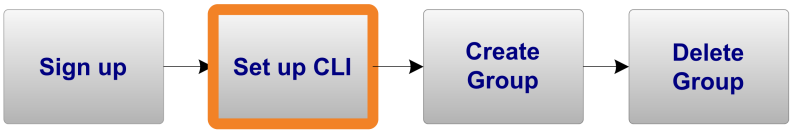
To sign up for Auto Scaling

1. Go to http://aws.amazon.com, and then click **Sign Up Now**.
2. Follow the on-screen instructions.

   Part of the sign-up procedure involves receiving a phone call and entering a PIN using the phone keypad.

Now you are ready to download the command-line tools.

Set Up the Command Line Interface



Topics

Install the Command-Line Tools

To use the commands described in this guide, you must install and configure the command-line tools.

To install the Auto Scaling command-line tools

1. Download the command-line tools from Auto Scaling Tools.
2. Save and unpack the archive to a convenient location on your workstation.
3. Verify that you have Java version 1.5 or newer:

   a. Enter the command `java -version` to determine the version of Java on your workstation. The following is an example response to this command:

   ```
   java version "1.6.0_21"
   Java(TM) SE Runtime Environment (build 1.6.0_21-b07)
   Java HotSpot(TM) 64-Bit Server VM (build 17.0-b17, mixed mode)
   ```

   b. If you don't have version 1.5 or newer, download and install a newer version from http://www.java.com/en/download/index.jsp.

4. Set the `JAVA_HOME` environment variable to point to your Java installation.

The following example shows how to set this environment variable in Linux and UNIX.

```
$ export JAVA_HOME=<PATH>
```

The following is an example of the syntax in Windows.

```
C:\> set JAVA_HOME=<PATH>
```

5. Add `$JAVA_HOME/bin` to your `$PATH` environment variable (`%JAVA_HOME%\bin` on Windows):

   - On Linux and UNIX, you can update your `PATH` as follows:

     ```
     $ export PATH=$PATH:$JAVA_HOME/bin
     ```

   - On Windows the syntax is slightly different:

     ```
     C:\> set PATH=%PATH%;%JAVA_HOME%\bin
     ```

6. Set the `AWS_AUTO_SCALING_HOME` environment variable to the location on your computer where you unpacked the Auto Scaling archive.

   - On Linux and UNIX, you can set your `AWS_AUTO_SCALING_HOME` as follows:

     ```
     $ export AWS_AUTO_SCALING_HOME=<PATH>
     ```

   - On Windows the syntax is slightly different:

     ```
     C:\> set AWS_AUTO_SCALING_HOME=<PATH>
     ```

7. Add `$AWS_AUTO_SCALING_HOME/bin` to your `$PATH` environment variable (`%AWS_AUTO_SCALING_HOME%\bin` on Windows):

   - On Linux and UNIX, you can update your `PATH` as follows:

     ```
     $ export PATH=$PATH:$AWS_AUTO_SCALING_HOME/bin
     ```

   - On Windows the syntax is slightly different:

     ```
     C:\> set PATH=%PATH%;%AWS_AUTO_SCALING_HOME%\bin
     ```

Configure the Tools for Your Credentials

Before you use Auto Scaling, you must provide your AWS credentials to the command-line tools. You can use your AWS access keys.

To use access keys with the command-line tools

1. Navigate to the AWS security credentials page. You will be prompted to log in if you haven't already.
2. Retrieve an access key and its corresponding secret key.
   a. Scroll down to the **Access Credentials** section and select the **Access Keys** tab.
   b. Locate an active Access Key in the **Your Access Keys** list.
   c. To display the Secret Access Key, click **Show** in the **Secret Access Key** column.
   d. Write down the keys or save them.
   e. If no Access Keys appear in the list, click **Create a New Access Key** and follow the on-screen prompts.
3. Add your access key ID and secret access key to the file named `credential-file-path.template`:
   a. Open the file `credential-file-path.template` included in your command-line tools archive.
   b. Copy and paste your access key ID and secret access key into the file.
   c. Rename the file and save it to a convenient location on your computer.
   d. If you are using Linux, set the file permissions as follows:

```
$ chmod 600 [credential file name]
```

4. Set the `AWS_CREDENTIAL_FILE` environment variable to the fully qualified path of the credential file you just created.
   - On Linux and UNIX, set the AWS_CREDENTIAL_FILE environment variable as follows:

```
$ export AWS_CREDENTIAL_FILE=<PATH>
```

   - On Windows the syntax is slightly different:

```
C:\> set AWS_CREDENTIAL_FILE=<PATH>
```

Test Your Configuration

To test your Auto Scaling installation and configuration

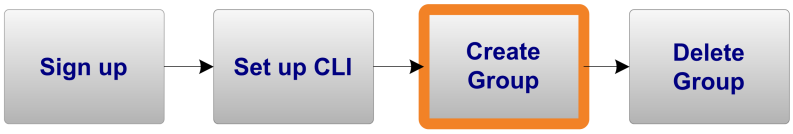
1. On your Linux or Windows workstation, open a new command prompt.
2. Type the command **`as-cmd`**.

3. You should see output similar to the following:

```
Command Name                                  Description

------------                                  -----------

as-create-auto-scaling-group                 Create a new auto scaling group
as-create-launch-config                       Create a new launch config
as-create-trigger                             Prints commands to create ...
as-delete-auto-scaling-group                  Delete ... auto scaling group
as-delete-launch-config                      Delete ... launch configuration
as-delete-policy                              Delete the specified policy
as-delete-scheduled-action                    Delete ... scheduled action
as-describe-adjustment-types                  Describes all policy ... types
as-describe-auto-scaling-groups               Describes... scaling group(s)
as-describe-auto-scaling-instances            Describes ... instance(s)
as-describe-launch-configs                    Describe ... launch configs
as-describe-metric-collection-types          Describes ... granularity types
as-describe-policies                          Describes ... policy/policies
as-describe-process-types                     Describes ... process types.
as-describe-scaling-activities               Describe ... scaling activities
as-describe-scheduled-actions                 Describes ... scheduled action
as-disable-metrics-collection                 Disable collection of metrics
as-enable-metrics-collection                  Enable collection of metrics
as-execute-policy                             Executes the specified policy

as-put-scaling-policy                        Creates or updates a ... policy
as-put-scheduled-update-group-action          Creates or updates ... action
as-resume-processes                          Resumes all suspended activities
as-set-desired-capacity                       Set the desired capacity of ...
as-set-instance-health                        Set the health of the instance

as-suspend-processes                          Suspends all scaling proce...
as-terminate-instance-in-auto-scaling-group Terminate a given instance.
as-update-auto-scaling-group                  Update ... auto scaling group
help
version                                       Prints the version of ...

For help on a specific command, type '<commandname> --help'
```

This completes your installation and configuration of the Auto Scaling command-line tools.

Create an Auto Scaling Group



Topics

Now that you have installed the Auto Scaling CLI, you're ready to create an Auto Scaling group. You'll use the command-line tools to create a simple Auto Scaling group designed to contain exactly one EC2 instance.

First, you'll create a launch configuration that specifies the type of EC2 instance that you want Auto Scaling to create. Second, you'll create an Auto Scaling group that references your launch configuration. Finally, you'll verify that the Auto Scaling group is functioning.

Create a Launch Configuration

The launch configuration specifies the type of Amazon EC2 instance that Auto Scaling creates for you. To create a launch configuration with `as-create-launch-config`, you must specify an Amazon Machine Image (AMI) ID and an Amazon EC2 instance type.

For this launch configuration, use the following AMI ID and instance type from the *Amazon Elastic Compute Cloud Getting Started Guide*:

- AMI ID: `ami-2272864b`
- Instance type: `m1.large`

To create a launch configuration

- Open a command prompt and enter the `as-create-launch-config` command.

```
as-create-launch-config MyLC --image-id ami-2272864b --instance-type m1.large
```

Auto Scaling returns the following:

```
OK-Created launch config
```

Create an Auto Scaling Group

After you have defined your launch configuration, you are ready to create an Auto Scaling group. To create an Auto Scaling group with `as-create-auto-scaling-group`, you must specify a name for your group, a launch configuration, one or more Availability Zones, a minimum group size, and a maximum group size.

Name your Auto Scaling group "MyGroup" and use the launch configuration you created previously. The Availability Zones you choose determine the physical location of your Auto Scaling instances. For this example, specify a single zone: `us-east-1a`. Set the minimum and maximum size of your Auto Scaling group to `1`. Because the minimum size of the group is `1`, Auto Scaling begins creating an EC2 instance immediately after you call `as-create-auto-scaling-group`.

Important

The instance you're about to launch will be live (and not running in a sandbox). You will incur the standard Amazon EC2 usage fees for the instance until you terminate it as the last task in this tutorial. The total charges will be minimal (typically less than a dollar). For more information about Amazon EC2 usage rates, go to the Amazon EC2 product page.

To create an Auto Scaling group

- Enter the `as-create-auto-scaling-group` command.

```
as-create-auto-scaling-group MyGroup --launch-configuration MyLC --availab
ility-zones us-east-1a --min-size 1 --max-size 1
```

Auto Scaling returns the following:

```
OK-Created AutoScalingGroup
```

Verify Auto Scaling Group Creation

You can use the `as-describe-auto-scaling-groups` command to check whether the `MyGroup` Auto Scaling group exists. Use the `--headers` parameter to print headings that describe each value that the command returns.

To verify that the Auto Scaling group exists

- Enter the `as-describe-auto-scaling-groups` command.

```
as-describe-auto-scaling-groups --headers
```

Auto Scaling returns the following:

```
AUTO-SCALING-GROUP  GROUP-NAME   LAUNCH-CONFIG  AVAILABILITY-ZONES  MIN-SIZE
  MAX-SIZE  DESIRED-CAPACITY
AUTO-SCALING-GROUP  MyGroup      MyLC           us-east-1a          1
    1         1
```

You can also use the `as-describe-auto-scaling-instances` command to check whether the `MyGroup` Auto Scaling group contains any running instances. Use the `--headers` parameter to print headings that describe each value that the command returns.

To verify that MyGroup contains an EC2 instance

- Enter the `as-describe-auto-scaling-instances` command.

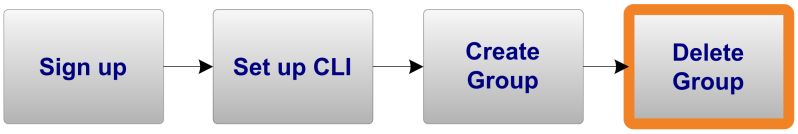
```
as-describe-auto-scaling-instances --headers
```

Auto Scaling returns the following (your `INSTANCE-ID` will differ):

```
INSTANCE  INSTANCE-ID  GROUP-NAME   AVAILABILITY-ZONE  STATE      STATUS
 LAUNCH-CONFIG
INSTANCE  i-bcdd63d1   MyGroup      us-east-1a         InService  HEALTHY
 MyLC
```

You have a number of other options when you use `as-create-auto-scaling-group`, `as-describe-auto-scaling-groups`, and `as-describe-auto-scaling-instances`. For more information, go to the Auto Scaling API Reference.

Delete the Group



Topics

In this section you will delete the Auto Scaling group. You will first remove the EC2 instance, then delete the Auto Scaling group, and finally delete the launch configuration.

Remove All EC2 Instances from the Group

You must terminate all EC2 instances in an Auto Scaling group before you can delete the group. A simple way to terminate all instances in a group is to update the group so that both the minimum size and maximum size are set to zero.

To remove the EC2 instance from the Auto Scaling group

1. Call `as-update-auto-scaling-group` to update the `MyGroup` Auto Scaling group that you created in the previous procedure.

   ```
   as-update-auto-scaling-group MyGroup --min-size 0 --max-size 0
   ```

   Auto Scaling returns the following:

   ```
   OK-Updated AutoScalingGroup
   ```

2. Call `as-describe-auto-scaling-groups` to check whether Auto Scaling has removed the instance from `MyGroup`.

   It can take a few minutes for the instance to terminate, so you might have to check the status more than once.

```
as-describe-auto-scaling-groups MyGroup --headers
```

   Auto Scaling returns the following if the instance termination is still in progress (your `INSTANCE-ID` will differ):

```
AUTO-SCALING-GROUP  GROUP-NAME  LAUNCH-CONFIG  AVAILABILITY-ZONES  MIN-SIZE
  MAX-SIZE  DESIRED-CAPACITY
AUTO-SCALING-GROUP  MyGroup     MyLC           us-east-1a          0
   0         0

INSTANCE  INSTANCE-ID  AVAILABILITY-ZONE  STATE       STATUS   LAUNCH-CONFIG
INSTANCE  i-bcdd63d1   us-east-1a         Terminating  Healthy  MyLC
```

   Auto Scaling returns the following after the instance is terminated:

```
AUTO-SCALING-GROUP  GROUP-NAME  LAUNCH-CONFIG  AVAILABILITY-ZONES  MIN-SIZE
  MAX-SIZE  DESIRED-CAPACITY
AUTO-SCALING-GROUP  MyGroup     MyLC           us-east-1a          0
   0         0
```

Delete the Auto Scaling Group

When no instances exist in `MyGroup`, you can delete the group.

To delete the Auto Scaling group

- Call `as-delete-auto-scaling-group`, specifying `MyGroup` as the only argument.

```
as-delete-auto-scaling-group MyGroup
```

  Auto Scaling returns the following:

```
Are you sure you want to delete this AutoScalingGroup? [Ny]
```

  Enter `Y` to confirm the deletion.

  Auto Scaling returns the following:

```
OK-Deleted AutoScalingGroup
```

Delete the Launch Configuration

Delete the launch configuration you created for this Auto Scaling group.

To delete the launch configuration

- Call `as-delete-launch-config`, specifying `MyLC` as the only argument.

```
as-delete-launch-config MyLC
```

Auto Scaling returns the following:

```
Are you sure you want to delete this launch configuration? [Ny]
```

Enter `Y` to confirm the deletion.

Auto Scaling returns the following:

```
OK-Deleted launch configuration
```

Your input is important to us. Help make our documentation helpful and easy to use. Please take a minute to provide feedback on your getting started experience with Auto Scaling. To begin the survey, see Please Provide Feedback (p. 13). Thank you.

Please Provide Feedback

Your input is important to help make our documentation helpful and easy to use. Please tell us about your experience getting started with Auto Scaling by completing our Getting Started Survey.

Thank you.

Where Do I Go from Here?

Topics

Auto Scaling offers features we haven't covered in this guide, such as setting the desired capacity, creating policies, and integrating with Amazon CloudWatch, Elastic Load Balancing, and Amazon Simple Notification Service. This section provides links to additional resources, which will help you deepen your understanding and use of Auto Scaling.

AWS Account and Security Credentials

So far you signed up for the service, got an AWS account and security credentials, and then completed a short exercise covering the essential product functions. Now that you're finished with the exercise, we recommend that you check with an administrator or coworker in your organization to determine if he or she already has an AWS account and security credentials for you to use in future interactions with AWS.

If you're an account owner or administrator and want to know more about AWS Identity and Access Management, go to the product description at http://aws.amazon.com/iam or to the technical documentation at Using AWS Identity and Access Management.

Other Ways to Access Auto Scaling

This Getting Started Guide has shown you how to use the CLI to create a simple Auto Scaling group. You can continue using Auto Scaling through the CLI, or try one of the other interfaces.

Continue using the Command Line Interface

The Command Line Interface includes functions not only for creating launch configurations and Auto Scaling groups, but also for creating policies and scheduled actions. These command-line tools are a fast way to execute all the Auto Scaling functions without coding to the API or using a library.

Use an Existing Library

If you prefer to use Auto Scaling through a programmatic interface, libraries and resources are available for the following languages:

- Java
- PHP
- Python
- Ruby
- Windows and .NET

For libraries and sample code in all languages, go to Amazon EC2 Sample Code & Libraries.

Code Directly to the Web Service API

If you want to write code directly to the Auto Scaling Query API, go to the Auto Scaling Developer Guide. The guide describes how to create and authenticate API requests, and how to use Auto Scaling through the API. For a complete description of all the API actions, go to the Auto Scaling API Reference.

Learn More about Auto Scaling

This section lists additional features of Auto Scaling and where to get more information.

Creating Health Checks

A health check is a call to check on the health status of each instance in an Auto Scaling group. If an instance reports degraded performance, Auto Scaling terminates the instance and launches another one to take its place. For more information, go to the Auto Scaling Developer Guide.

Creating Triggers

A trigger is a combination of an Auto Scaling policy and an Amazon CloudWatch alarm. You can create alarms that monitor specific metrics gathered from an EC2 instance. When paired with an alarm, a policy can initiate an Auto Scaling action when a CloudWatch metric breaches a specific threshold. For more information, go to the Auto Scaling Developer Guide.

Creating Scheduled Updates

A scheduled update is a call to Auto Scaling that is scheduled for a future time. For more information, go to the Auto Scaling Developer Guide.

Suspending and Resuming Processes

You can suspend scaling processes at any time. When you're ready, you can resume any or all of the suspended processes. When an Auto Scaling group's scaling processes are suspended, Auto Scaling creates no new scaling activities for that group for any reason. Scaling activities that were already in progress before the group was suspended continue until complete. For more information, go to the Auto Scaling Developer Guide.

Auto Scaling Resources

The following table lists related resources that you'll find useful as you work with the Auto Scaling service.

Resource	Description
Auto Scaling Technical FAQ	Covers the top questions developers have asked about this product.
Release Notes	Give a high-level overview of the current release. They specifically note any new features, corrections, and known issues.
AWS Developer Resource Center	A central starting point to find documentation, code samples, release notes, and other information to help you build innovative applications with AWS.
Discussion Forums	A community-based forum for developers to discuss technical questions related to Amazon Web Services.
AWS Support Center	The home page for AWS Technical Support, including access to our Developer Forums, Technical FAQs, Service Status page, and Premium Support.
AWS Premium Support	The primary web page for information about AWS Premium Support, a one-on-one, fast-response support channel to help you build and run applications on AWS infrastructure services.
Auto Scaling product information	The primary web page for information about Auto Scaling.
Contact Us	A central contact point for inquiries concerning AWS billing, account, events, abuse, etc.
Conditions of Use	Detailed information about the copyright and trademark usage at Amazon.com and other topics.

About This Guide

This is the *Auto Scaling Getting Started Guide*. It was last updated on February 09, 2012.