
Auto Scaling

用户指南



Auto Scaling: 用户指南

Table of Contents

什么是 Auto Scaling ?	1
Auto Scaling 组件	1
入门	2
访问 Auto Scaling	2
Auto Scaling 定价	3
PCI DSS 合规性	3
相关服务	3
Auto Scaling 的优势	3
示例：覆盖可变需求	3
示例：Web 应用程序架构	5
示例：在可用区之间分配实例	5
Auto Scaling 生命周期	7
扩大	7
已投入使用的实例	7
缩小	8
附加实例	8
分离实例	8
生命周期挂钩	8
进入和退出备用状态	8
Auto Scaling 限制	9
设置	10
注册 AWS	10
准备使用 Amazon EC2	10
入门	11
第 1 步：创建启动配置	11
步骤 2：创建 Auto Scaling 组	12
步骤 3：验证 Auto Scaling 组	13
步骤 4：（可选）删除您的 Auto Scaling 基础设施	14
教程：设置具有扩展和负载均衡功能的应用程序	15
先决条件	15
使用 AWS 管理控制台配置扩展和负载均衡功能	15
创建或选择启动配置	16
创建 Auto Scaling 组	16
（可选）验证您的负载均衡器是否已附加到您的 Auto Scaling 组	17
使用 AWS CLI 配置扩展和负载均衡功能	17
创建启动配置	18
创建包含负载均衡器的 Auto Scaling 组	18
启动配置	19
创建启动配置	19
使用 EC2 实例创建启动配置	20
使用 EC2 实例创建启动配置	21
从实例创建启动配置以及覆盖块储存设备	22
创建启动配置和覆盖实例类型	23
更改启动配置	24
在 VPC 中启动 Auto Scaling 实例	24
默认 VPC	25
VPC 中的 IP 寻址	25
实例部署租期	25
将 EC2-Classic 实例链接到 VPC	26
示例	28
在 Auto Scaling 组中启动竞价型实例	28
使用 AWS 管理控制台启动竞价型实例	28
使用 AWS CLI 启动竞价型实例	31
Auto Scaling 组	35

创建 Auto Scaling 组	36
使用 EC2 实例创建 Auto Scaling 组	37
通过控制台使用 EC2 实例创建 Auto Scaling 组	37
使用 AWS CLI 从 EC2 实例创建 Auto Scaling 组	38
使用 Amazon EC2 启动向导创建 Auto Scaling 组	39
标记 Auto Scaling 组和实例	39
标签限制	40
标记生命周期	40
对 Auto Scaling 组添加或修改标签	40
删除标签	42
将负载均衡器与 Auto Scaling 组结合使用	42
附加负载均衡器	43
添加 ELB 运行状况检查	45
添加可用区	45
合并 Auto Scaling 组	47
使用 AWS CLI 合并区域	48
删除 Auto Scaling 基础设施	49
删除 Auto Scaling 组	49
(可选) 删除启动配置	49
(可选) 删除负载均衡器	50
(可选) 删除 CloudWatch 警报	50
扩展组	51
扩展计划	51
多个扩展策略	52
保持 Auto Scaling 组的大小	52
确定实例运行状况	53
替换运行状况不佳的实例	53
手动扩展	53
使用控制台更改 Auto Scaling 组的大小	53
使用 AWS CLI 更改 Auto Scaling 组的大小	54
将 EC2 实例附加到 Auto Scaling 组	55
从 Auto Scaling 组中分离 EC2 实例	58
计划的扩展	60
计划的操作的注意事项	61
使用控制台创建计划的操作	61
更新计划的操作	61
使用 AWS CLI 创建或更新计划的操作	62
删除计划的操作	62
动态扩展	63
扩展策略类型	63
简单扩展和步进扩展	63
目标跟踪扩展	69
向现有的 Auto Scaling 组添加扩展策略	72
基于 Amazon SQS 进行扩展	73
Auto Scaling 冷却时间	75
示例：Auto Scaling 冷却时间	76
默认冷却时间	76
特定于扩展的冷却时间	77
冷却时间和多个实例	77
冷却时间和生命周期挂钩	77
冷却时间和竞价型实例	77
Auto Scaling 实例终止	78
默认终止策略	78
自定义终止策略	80
实例保护	81
生命周期挂钩	83
生命周期挂钩的工作方式	83

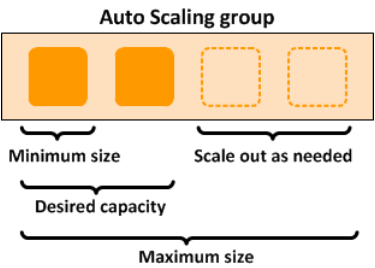
使用生命周期挂钩时的注意事项	84
准备通知	85
添加生命周期钩子	87
完成生命周期钩子	87
测试通知	88
临时删除实例	88
备用状态的工作方式	89
处于备用状态的实例的运行状况	89
使用 AWS 管理控制台临时删除实例	90
使用 AWS CLI 临时删除实例	90
暂停重新开始流程	92
Auto Scaling 流程	92
使用控制台暂停和恢复流程	93
使用 AWS CLI 暂停和恢复流程	94
监控您的 Auto Scaling 实例和组	95
运行状况检查	95
实例运行状况	96
运行状况检查宽限期	96
自定义运行状况检查	96
Amazon CloudWatch 指标	97
Auto Scaling 组指标	97
Auto Scaling 组指标的维度	98
启用 Auto Scaling 组指标	98
为 Auto Scaling 实例配置监控	99
查看 CloudWatch 指标	99
创建 Amazon CloudWatch 警报	100
Amazon CloudWatch Events	101
Auto Scaling 事件	101
创建 Lambda 函数	105
将事件路由到 Lambda 函数	105
Amazon SNS 通知	106
SNS 通知	106
配置 Amazon SNS	107
配置 Auto Scaling 组以发送通知	108
测试通知配置	108
验证是否收到扩展事件的通知	109
删除通知配置	110
AWS CloudTrail 日志记录	110
CloudTrail 中的 Auto Scaling 信息	111
了解 Auto Scaling 日志文件条目	111
控制对您 Auto Scaling 资源的访问权	114
Auto Scaling 操作	114
Auto Scaling 资源	115
Auto Scaling 条件键	116
支持的资源级权限	116
预定义的 AWS 托管策略	118
客户管理的策略	118
示例：创建并管理启动配置	118
示例：创建并管理 Auto Scaling 组和扩展策略	119
示例：使用标签控制访问	120
示例：更改 Auto Scaling 组的容量	121
使用 IAM 角色启动 Auto Scaling 实例	122
先决条件	122
创建启动配置	123
故障排除	124
检索错误消息	124
实例启动失败	126

安全组 <该安全组的名称> 不存在。启动 EC2 实例失败。	126
密钥对 <与您的 EC2 实例相关联的密钥对> 不存在。启动 EC2 实例失败。	127
当前不支持请求的配置。	127
AutoScalingGroup <Auto Scaling 组名称> 未找到。	127
不再支持请求的可用区域。请重新尝试请求.....	127
请求的实例类型 (<实例类型>) 在请求的可用区域 (<实例可用区域>) 中不受支持.....	127
您没有订阅此项服务。请参阅 http://www.amazonaws.cn 。	127
设备名称上载无效。启动 EC2 实例失败。	128
用于参数 virtualName 的值 (<与实例存储设备相关联的名称>) 无效... ..	128
实例存储 AMI 不支持 EBS 块存储设备映射。	128
置放群组可能无法与类型为“m1.large”的实例一起使用。启动 EC2 实例失败。	128
AMI 问题	128
AMI ID <您的 AMI 的 ID> 不存在。启动 EC2 实例失败。	129
AMI <AMI ID> 正在等待，无法运行。启动 EC2 实例失败。	129
虚拟化类型为“hvm”的非 Windows AMI 当前可能仅适用于群集计算实例类型。启动 EC2 实例失 败。	129
用于参数 virtualName 的值 (<AMI ID>) 无效。	129
请求的实例类型的架构 (i386) 与用于 ami-6622f00f (x86_64) 的清单中的架构不匹配。启动 EC2 实例失败。	130
负载均衡器问题	130
找不到负载均衡器 <您的启动环境>。验证负载均衡器配置失败。	130
名为 <负载均衡器名称> 的活动负载均衡器不存在。更新负载均衡器配置失败。	130
EC2 实例 <实例 ID> 不在 VPC 内。更新负载均衡器配置失败。	131
EC2 实例 <实例 ID> 在 VPC 中。更新负载均衡器配置失败。	131
请求中包含的安全令牌无效。验证负载均衡器配置失败。	131
容量限制	131
您请求的可用区域 (<请求的可用区域>) 中当前没有足够的 <实例类型> 容量.....	131
已运行 <实例数量> 个实例。启动 EC2 实例失败。	132
Resources	133
文档历史记录	134

什么是 Auto Scaling ？

Auto Scaling 可帮助确保您拥有适量的 Amazon EC2 实例来处理您的应用程序负载。您可创建 EC2 实例的集合，称为 Auto Scaling 组。您可以指定每个 Auto Scaling 组中最少的实例数量，Auto Scaling 会确保您的组中的实例永远不会低于这个数量。您可以指定每个 Auto Scaling 组中最大的实例数量，Auto Scaling 会确保您的组中的实例永远不会高于这个数量。如果您在创建组的时候或在创建组之后的任何时候指定了所需容量，Auto Scaling 会确保您的组一直具有此数量的实例。如果您指定了扩展策略，则 Auto Scaling 可以在您的应用程序的需求增加或降低时启动或终止实例。


例如，以下 Auto Scaling 组的最小容量为 1 个实例，所需容量为 2 个实例，最大容量为和 4 个实例。您制定的扩展策略是按照您指定的条件，在最大最小实例数范围内调整实例的数量。



有关 Auto Scaling 优点的更多信息，请参阅 [Auto Scaling 的优势 \(p. 3\)](#)。

Auto Scaling 组件

下表描述 Auto Scaling 的关键组件。

	<p>组</p> <p>您的 EC2 实例整理到组 中，从而当作一个逻辑单位进行扩展和管理。当您创建一个组时，您可以指定其中 EC2 实例的最小数量、最大数量以及</p>
---	--

	所需数量。有关更多信息，请参阅 Auto Scaling 组 (p. 35) 。
	启动配置 组使用启动配置 作为其 EC2 实例的模板。创建启动配置时，您可以为实例指定诸如 AMI ID、实例类型、密钥对、安全组和块储存设备映射等信息。关于更多信息，请参阅 启动配置 (p. 19) 。
	扩展计划 扩展计划 告知 Auto Scaling 进行扩展的时间和方式。例如，您可以根据指定条件的发生（动态扩展）或根据时间表来制定扩展计划。有关更多信息，请参阅 扩展计划 (p. 51) 。

入门

如果您刚刚接触 Auto Scaling，建议您在开始工作之前查看 [Auto Scaling 生命周期 \(p. 7\)](#)。

若要开始使用，请学完 [Auto Scaling 入门 \(p. 11\)](#) 教程，以创建一个 Auto Scaling 组并了解它在该组中的实例终止时如何进行响应。如果您已有 EC2 实例正在运行，则可用一个现有 EC2 实例创建 Auto Scaling 组，并可随时从该组中删除实例。

访问 Auto Scaling

AWS 提供基于 Web 的用户界面，即 AWS 管理控制台。如果您已注册 AWS 账户，则可通过登录 AWS 管理控制台访问 Auto Scaling。首先，从控制台主页选择 EC2，然后从导航窗格选择 Launch Configurations。

如果倾向于使用命令行界面，您可使用以下选项：

AWS 命令行界面 (CLI)

提供大量 AWS 产品的相关命令，同时被 Windows、Mac 和 Linux 支持。要了解其用法，请参阅 [AWS Command Line Interface 用户指南](#)。有关 Auto Scaling 命令的更多信息，请参阅 AWS Command Line Interface Reference 中的 [autoscaling](#)。

适用于 Windows PowerShell 的 AWS 工具

为在 PowerShell 环境中编写脚本的用户提供大量 AWS 产品的相关命令。要开始使用，请参阅 [适用于 Windows PowerShell 的 AWS 工具 用户指南](#)。有关 Auto Scaling 的 Cmdlet 的更多信息，请参阅 [适用于 Windows PowerShell 的 AWS 工具 Reference](#)。

Auto Scaling 提供 Query API。这些请求属于 HTTP 或 HTTPS 请求，需要使用 HTTP 动词 GET 或 POST 以及一个名为 `Action` 的查询参数。有关 Auto Scaling 的 API 操作的更多信息，请参阅 Auto Scaling API 参考 中的 [Actions](#)。

如果您倾向于使用特定语言的 API 而非通过 HTTP 或 HTTPS 提交请求来构建应用程序，AWS 为软件开发人员提供了库文件、示例代码、教程和其他资源。这些库文件提供可自动执行任务的基本功能，例如以加密方式对请求签名、重试请求和处理错误响应，因此您可以更轻松地上手。有关更多信息，请参阅 [AWS SDKs and Tools](#)。

有关用于访问 AWS 的证书的信息，请参阅 Amazon Web Services 一般参考 中的 [AWS Security Credentials](#)。

Auto Scaling 定价

Auto Scaling 不产生额外费用，因此可方便地试用它并了解它如何使您的 AWS 架构获益。

PCI DSS 合规性

Auto Scaling 支持由商家或服务提供商处理、存储和传输信用卡数据，而且已经验证符合支付卡行业 (PCI) 数据安全标准 (DSS)。有关 PCI DSS 的更多信息，包括如何请求 AWS PCI Compliance Package 的副本，请参阅 [PCI DSS 第 1 级](#)。

相关服务

使用 Elastic Load Balancing 在 Auto Scaling 组的多个实例之间自动分配应用程序的传入流量。有关更多信息，请参阅 [Elastic Load Balancing 用户指南](#)。

要监控您的实例和 Amazon EBS 卷的基本统计数据，可使用 Amazon CloudWatch。有关更多信息，请参阅 [Amazon CloudWatch 用户指南](#)。

要监控对您的账户的 Auto Scaling API 的调用（包括由 AWS 管理控制台、命令行工具和其他服务进行的调用），请使用 AWS CloudTrail。有关更多信息，请参阅 [AWS CloudTrail User Guide](#)。

Auto Scaling 的优势

向应用程序架构添加 Auto Scaling 是最大程度利用 AWS 云的一种方式。当您使用 Auto Scaling 时，您的应用程序将获得以下优势：

- 提高容错能力。Auto Scaling 可以检测到实例何时运行状况不佳并终止实例，然后启动新实例以替换它。您还可以配置 Auto Scaling 以使用多个可用区。如果一个可用区变得不可用，则 Auto Scaling 可以在另一个可用区中启动实例以进行弥补。
- 提高可用性。Auto Scaling 组可帮助确保应用程序始终拥有合适的容量以满足当前流量需求。
- 加强成本管理。Auto Scaling 可以按需要动态地增加或降低容量。您只需为使用的 EC2 实例付费，在实际需要的时候启动实例，在不需要的时候终止实例，从而节约成本。

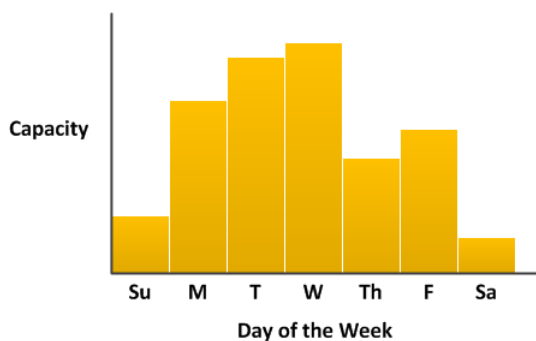
内容

- [示例：覆盖可变需求 \(p. 3\)](#)
- [示例：Web 应用程序架构 \(p. 5\)](#)
- [示例：在可用区之间分配实例 \(p. 5\)](#)

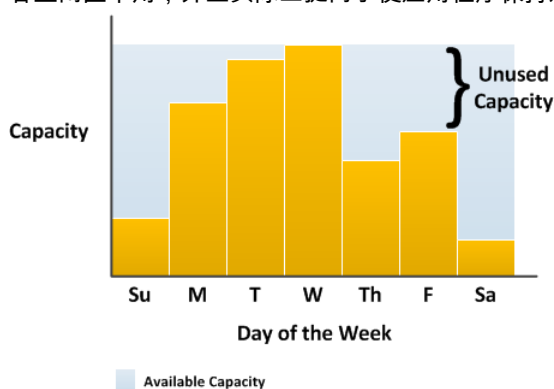
示例：覆盖可变需求

为了更好地演示 Auto Scaling 的一些优势，请设想有一个在 AWS 上运行的基本 Web 应用程序。此应用程序允许员工搜索可用于开会的会议室。每周开始和结束时段，此应用程序的使用率最低。每周中期，有更多的员工安排会议，因此对此应用程序的需求会显著提高。

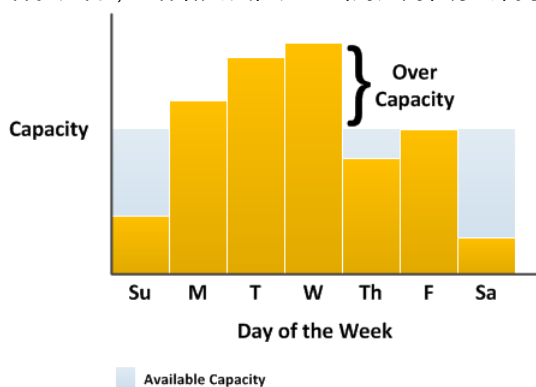
下图显示此应用程序的容量在一周中的使用情况。



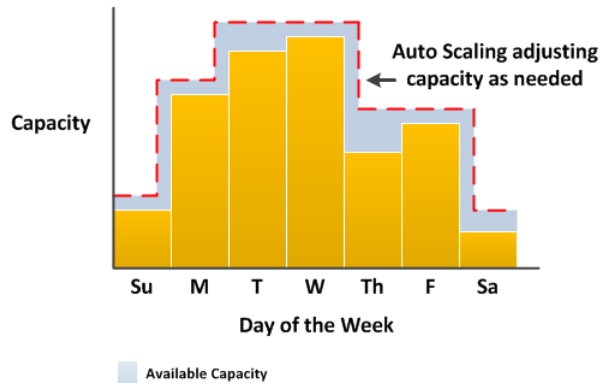
按照传统做法，可通过两种方式对这些容量变化做好规划。第一种选择是添加足够多的服务器，以便应用程序始终具有足够的容量来满足需求。但是，这种做法的缺点是应用程序在某些天并不需要这么多容量。额外容量闲置不用，并且实际上提高了使应用程序保持运行的成本。



第二种选择是采用处理应用程序平均需求所需的容量。这种做法成本更低，因为不用购买仅仅偶尔使用的设备。然而，这样做的风险是：当对应用程序的需求超过其容量时，可能造成糟糕的客户体验。

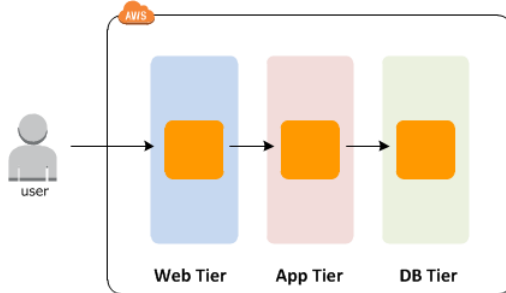


通过向此应用程序添加 Auto Scaling，您就有了第三种选择。您可以仅在需要时才向应用程序添加新实例，并在不再需要这些实例时终止它们。由于 Auto Scaling 使用 EC2 实例，因此您只需在使用时为使用的实例付费。您现在有了一个具有成本效益的架构，可在尽量减少支出的同时提供最佳客户体验。

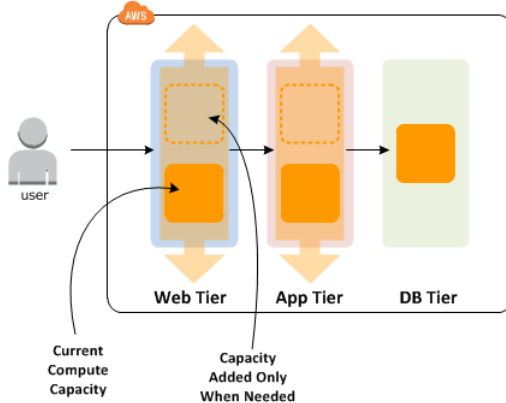


示例：Web 应用程序架构

在常见的 Web 应用程序场景中，您同时运行应用程序的多个副本来满足客户流量。将在相同的 EC2 实例（云服务器）上托管应用程序的多个副本，每个实例都可以处理客户请求。



Auto Scaling 代表您管理 EC2 实例的启动和终止。您定义一组条件（如 Amazon CloudWatch 警报），用于确定 Auto Scaling 组何时启动或终止 EC2 实例。向网络架构添加 Auto Scaling 组可帮助提高应用程序的可用性和容错能力。



您可以根据需要创建任意数量的 Auto Scaling 组。例如，您可以为每个层创建一个 Auto Scaling 组。

要在您的 Auto Scaling 组的各实例之间分配流量，可在您的架构中引入一个负载均衡器。有关更多信息，请参阅 [将负载均衡器与 Auto Scaling 组结合使用](#) (p. 42)。

示例：在可用区之间分配实例

AWS 资源，如 EC2 实例，置于高度可用的数据中心之内。为了提供额外的扩展性和可靠性，这些数据中心位于不同的物理位置。地区是指大型、分布范围广泛的地理位置。每个区域含有多个不同的位置，被称为可

用区，是设计用于隔离其他可用区内故障的位置，可向相同区域中的其他可用区提供低成本、低延迟的网络连接。有关更多信息，请参阅 Amazon Web Services 一般参考 中的 [区域和终端节点：Auto Scaling](#)。

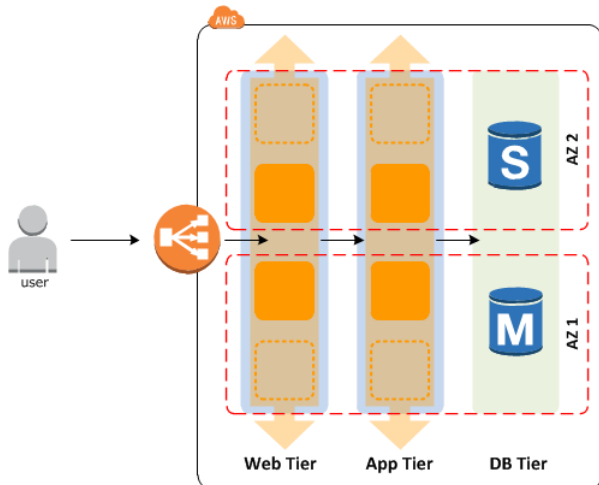
通过使 Auto Scaling 组跨越区域中的多个可用区，Auto Scaling 使您可以利用地理冗余的安全性和可靠性。当一个可用区运行状况不佳或无法使用时，Auto Scaling 将在不受影响的可用区中启动新实例。当运行状况不佳的可用区恢复运行状况时，Auto Scaling 会自动在所有指定的可用区中重新均匀分配应用程序实例。

Auto Scaling 组可以包含来自同一区域的一个或多个可用区的 EC2 实例。但是，Auto Scaling 组不能跨多个区域。

对于 VPC 中的 Auto Scaling 组，在子网中启动 EC2 实例。在您创建或更新 Auto Scaling 组时，要为您的 EC2 实例选择子网。您可为每个可用区选择一个或多个子网。有关更多信息，请参阅 [在 VPC 中启动 Auto Scaling 实例 \(p. 24\)](#)。

实例分配

Auto Scaling 尝试在为您的 Auto Scaling 组启用的可用区之间均匀分配实例。Auto Scaling 通过尝试在实例最少的可用区中启动新实例来执行此操作。但是，如果尝试失败，Auto Scaling 将尝试在另一可用区中启动实例，直到成功。对于 VPC 中的 Auto Scaling 组，如果可用区内有多个子网，Auto Scaling 将从可用区中随机选择一个子网。



再平衡活动

在某些操作发生后，您的 Auto Scaling 组可能会在不同可用区之间变得不平衡。Auto Scaling 将通过重新平衡可用区来进行补偿。以下操作可能导致重新平衡活动：

- 更改组的可用区。
- 您明确地终止或断开实例，组变得不平衡。
- 之前没有足够容量的可用区已恢复，并提供了额外容量。
- 之前竞价市场价格超出您的竞价出价的可用区现在的市场价格低于您的出价。

再平衡时，Auto Scaling 在终止旧实例之前启动新实例，所以再平衡不会损害应用程序的性能或可用性。

因为 Auto Scaling 尝试在终止旧实例之前启动新实例，所以达到或接近指定的最大容量可能会阻碍或完全停止再平衡活动。为避免此问题，在再平衡活动期间，系统可以暂时超出某组的指定最大容量的 10%（或超出 1 个实例，以较大者为准）。仅当该组达到或接近最大容量，并需要重新平衡时，才可超出容量限制；此类情况的原因是用户请求重新分区，或者是为了弥补区域可用性问题。该超出状态仅持续重新平衡该组所需的时间（通常为几分钟）。

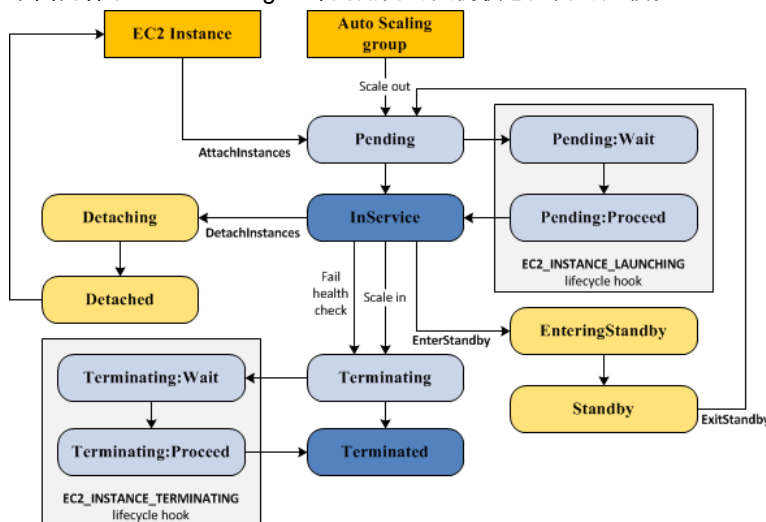
Auto Scaling 生命周期

Auto Scaling 组中的 EC2 实例具有的路径或生命周期不同于其他 EC2 实例中的路径或生命周期。生命周期从 Auto Scaling 组启动实例并将其投入使用时开始。生命周期在您终止实例或 Auto Scaling 组禁用实例并将其终止时结束。

Note

一旦启动实例，您就需要为实例付费，包括尚未将实例投入使用的时间。

下图阐释了 Auto Scaling 生命周期内的实例状态之间的过渡。



扩大

以下扩大事件指示 Auto Scaling 组启动 EC2 实例并将其附加到组：

- 手动增大组的大小。有关更多信息，请参阅 [手动扩展 \(p. 53\)](#)。
- 您创建一个扩展策略来自动根据指定的所需增量来增大组的大小。有关更多信息，请参阅 [动态扩展 \(p. 63\)](#)。
- 您可以通过安排在某个特定时间增大组的大小来设置扩展。有关更多信息，请参阅 [计划的扩展 \(p. 60\)](#)。

在发生扩大事件时，Auto Scaling 组将使用分配的启动配置来启动所需数目的 EC2 实例。这些实例最初处于 Pending 状态。如果您向 Auto Scaling 组添加生命周期钩子，则可在此处执行自定义操作。有关更多信息，请参阅 [生命周期钩子 \(p. 8\)](#)。

在每个实例完全配置并通过 Amazon EC2 运行状况检查后，该实例将附加到 Auto Scaling 组并进入 InService 状态。针对 Auto Scaling 组的所需容量对实例进行计数。

已投入使用的实例

实例将保持 InService 状态，直至出现下列情况之一：

- 发生缩小事件，并且 Auto Scaling 选择终止此实例来减小 Auto Scaling 组的大小。有关更多信息，请参阅 [控制 Auto Scaling 在缩小过程中终止哪些实例 \(p. 78\)](#)。
- 将实例置于 standby 状态。有关更多信息，请参阅 [进入和退出备用状态 \(p. 8\)](#)。
- 您从 Auto Scaling 组分离实例。有关更多信息，请参阅 [分离实例 \(p. 8\)](#)。

- 实例未通过所需数目的运行状况检查，因此将从 Auto Scaling 组中删除实例、终止实例和替换实例。有关更多信息，请参阅 [Auto Scaling 实例的运行状况检查 \(p. 95\)](#)。

缩小

重要的是，您要为您创建的每个扩展事件创建一个相应的缩小事件。这有助于确保分配给您的应用程序的资源与对这些资源的需求尽可能相符。

以下缩小事件指示 Auto Scaling 组从组中分离 EC2 实例并将其终止：

- 手动减小组的大小。
- 您创建一个扩展策略，自动根据指定的所需减少量来减小组的大小。
- 您可以通过安排在某个特定时间减小组的大小来设置扩展。

发生缩小事件时，Auto Scaling 组分离一个或多个实例。Auto Scaling 组使用其终止策略来确定要终止的实例。正在从 Auto Scaling 组中分离和关闭的实例将进入 `Terminating` 状态，且无法重新将其投入使用。如果您向 Auto Scaling 组添加生命周期挂钩，则可在此处执行自定义操作。最后，实例将完全终止并进入 `Terminated` 状态。

附加实例

您可以将符合特定条件的正在运行的 EC2 实例附加到您的 Auto Scaling 组。在附加实例后，将该实例作为 Auto Scaling 组的一部分进行管理。

有关更多信息，请参阅 [将 EC2 实例附加到 Auto Scaling 组 \(p. 55\)](#)。

分离实例

您可以从 Auto Scaling 组中分离实例。分离实例后，您可以独立于 Auto Scaling 组管理实例或者将实例附加到其他 Auto Scaling 组。

有关更多信息，请参阅 [从 Auto Scaling 组中分离 EC2 实例 \(p. 58\)](#)。

生命周期挂钩

您可以将生命周期挂钩添加到 Auto Scaling 组，以便在实例启动或终止时执行自定义操作。

当 Auto Scaling 响应扩大事件时，它将启动一个或多个实例。这些实例最初处于 `Pending` 状态。如果您已将一个 `autoscaling:EC2_INSTANCE_LAUNCHING` 生命周期挂钩添加到您的 Auto Scaling 组，则实例将从 `Pending` 状态转换为 `Pending:Wait` 状态。完成生命周期操作后，实例将进入 `Pending:Proceed` 状态。在完全配置实例后，实例将附加到 Auto Scaling 组并进入 `InService` 状态。

当 Auto Scaling 响应缩小事件时，它将终止一个或多个实例。这些实例将从 Auto Scaling 组中分离并进入 `Terminating` 状态。如果您已将一个 `autoscaling:EC2_INSTANCE_TERMINATING` 生命周期挂钩添加到您的 Auto Scaling 组，则实例将从 `Terminating` 状态转换为 `Terminating:Wait` 状态。完成生命周期操作后，实例将进入 `Terminating:Proceed` 状态。在完全终止实例后，实例将进入 `Terminated` 状态。

有关更多信息，请参阅 [Auto Scaling 生命周期挂钩 \(p. 83\)](#)。

进入和退出备用状态

可以将任何处于 `InService` 状态的实例置于 `Standby` 状态。这使您能够终止对实例的使用，排查实例的问题或对实例进行更改，然后重新将实例投入使用。

处于 `standby` 状态的实例继续由 Auto Scaling 组管理。但是，在将这些实例重新投入使用前，它们不是您的应用程序的有效部分。

有关更多信息，请参阅 [临时从 Auto Scaling 组中删除实例 \(p. 88\)](#)。

Auto Scaling 限制

要查看 Auto Scaling 资源的当前限制，请使用 Amazon EC2 控制台的 Limits 页或使用 [describe-account-limits](#) (AWS CLI) 命令。要请求增大限制，请使用 [Auto Scaling 限制表](#)。

下表列出与您的 Auto Scaling 资源相关的默认限制。

资源	默认限制
每个区域的启动配置	100
每个区域的 Auto Scaling 组	20
每个 Auto Scaling 组的扩展策略	50
每个 Auto Scaling 组的计划操作	125
每个 Auto Scaling 组的生命周期挂钩	50
每个 Auto Scaling 组的 SNS 主题	10
每个 Auto Scaling 组的 Classic 负载均衡器	50*
每个 Auto Scaling 组的目标组	50*
每个扩展策略的步进调整	20

* 请注意，您一次最多可以附加或分离 10 个项。

有关其他服务的限制的信息，请参阅 Amazon Web Services 一般参考 中的 [AWS 服务限制](#)。

设置 Auto Scaling

开始使用 Auto Scaling 前，请完成以下任务。

任务

- [注册 AWS \(p. 10\)](#)
- [准备使用 Amazon EC2 \(p. 10\)](#)

注册 AWS

创建 AWS 账户时，我们会自动为所有 AWS 服务注册您的账户。您只需为使用的服务付费。除了要为 EC2 实例支付费用，您使用 Auto Scaling 无需任何额外费用。

如果您没有 AWS 账户，请按照以下方式注册 AWS。

如需注册 AWS 账户

1. 打开 <http://www.amazonaws.cn/>，然后选择 Create an AWS Account。
2. 按照屏幕上的说明进行操作。

作为注册流程的一部分，您会收到一个电话，需要您使用电话键盘输入一个 PIN 码。

注册过程完成后，AWS 会向您发送确认电子邮件。

准备使用 Amazon EC2

如果您以前未使用过 Amazon EC2，请完成 Amazon EC2 文档中所述的任务。有关更多信息，请参阅 Amazon EC2 用户指南（适用于 Linux 实例）中的 [Amazon EC2 中的设置](#)，或者 Amazon EC2 用户指南（适用于 Windows 实例）中的 [Amazon EC2 中的设置](#)，这取决于您计划对 EC2 实例使用的操作系统。

Auto Scaling 入门

当您每次计划使用 Auto Scaling 时，都必须从特定构建块入手。本教程将分步为您介绍为 Auto Scaling 设置基本基础设施的过程。

以下的分步说明帮助您创建一个模板，该模板用来定义您的 EC2 实例，创建 Auto Scaling 组以始终保持健康的实例数量，并（可选）删除这一基本 Auto Scaling 基础设施。本教程假定您熟悉 EC2 实例的启动并已创建密钥对和安全组。

任务

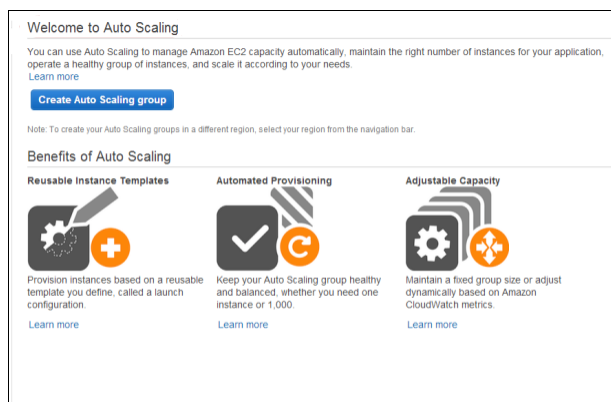
- [第 1 步：创建启动配置 \(p. 11\)](#)
- [步骤 2：创建 Auto Scaling 组 \(p. 12\)](#)
- [步骤 3：验证 Auto Scaling 组 \(p. 13\)](#)
- [步骤 4：（可选）删除您的 Auto Scaling 基础设施 \(p. 14\)](#)

第 1 步：创建启动配置

启动配置会指定 Auto Scaling 为您创建的 EC2 实例类型。您创建启动配置时需包括以下信息：用于启动 EC2 实例的 Amazon 系统映像 (AMI) ID、实例类型、键前缀、安全组、块储存设备映射以及其他配置设置。

创建启动配置

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航栏上，选择区域。您创建的 Auto Scaling 资源与您指定的区域绑定，不会在各区域之间重复。有关更多信息，请参阅 [示例：在可用区之间分配实例 \(p. 5\)](#)。
3. 在导航窗格上的 Auto Scaling 下，选择 Launch Configurations。
4. 在 Welcome to Auto Scaling 页面上，选择 Create Auto Scaling group。



5. 在 Create Auto Scaling Group 页面上，选择 Create launch configuration。
6. 在 Choose AMI 页面上，有一组称为 Amazon 系统映像 (AMI) 的基本配置，它们充当您的实例的模板。选择 64 位 Amazon Linux AMI。
7. 在 Choose Instance Type 页面上，为您的实例选择硬件配置。建议您保留默认值，即 t2.micro 实例。选择 Next: Configure details。

Note

必须将 T2 实例启动到 VPC 的子网中。如果选择 t2.micro 实例，但是没有 VPC，系统会为您创建一个。此 VPC 在区域的每个可用区中均包含一个公有子网。

8. 在 Configure Details (配置实例详细信息) 页面中，执行以下操作：
 - a. 对于 Name，为启动配置键入一个名称（例如，my-first-lc）。
 - b. 对于 Advanced Details，选择 IP 地址类型。如果要连接至 VPC 中的一个实例，您必须选择一个指定了公有 IP 地址的选项。如果您想连接到实例，但是不确定您是否有默认 VPC，请选择 Assign a public IP address to every instance。
 - c. 选择 Skip to review。
9. 在 Review 页面上，选择 Edit security groups。按照说明操作来选择现有安全组，然后选择 Review。
10. 在 Review 页上选择 Create launch configuration。
11. 在 Select an existing key pair or create a new key pair 页面上，选择所列的选项之一。注意，在本教程中，您不会连接到自己的实例。因此，您可以选择 Proceed without a key pair，除非您打算连接到您的实例。
12. 选择 Create launch configuration。

步骤 2：创建 Auto Scaling 组

Auto Scaling 组是 EC2 实例的集合，并且是 Auto Scaling 服务的核心。通过指定您要用于启动实例的启动配置，并指定您的组必须始终保持的实例数，创建 Auto Scaling 组。您还可以指定要在其中启动实例的可用区。

创建 Auto Scaling 组

1. 在 Configure Auto Scaling group details 页面上，执行以下操作：
 - a. 对于 Group name，键入您的 Auto Scaling 组的名称（例如，my-first-asg）。
 - b. 在本教程中，将 Group size 保留为默认值，即 1 个实例。
 - c. 如果启动的是 t2.micro 实例，则必须在 Network 中选择一个 VPC。否则，如果您的账户支持 EC2-Classic 而且您准备启动的实例类型不需要 VPC，则选择 Launch into EC2-Classic 或 VPC 均可。

- d. 如果您在上一步中选择了 VPC，请从 Subnet 中选择一个或多个子网。如果您在上一步中已选择 EC2-Classic，请从 Availability Zone(s) 中选择一个或多个可用区。
- e. 选择 Next: Configure scaling policies。
2. 在 Configure scaling policies 页面上，选择 Keep this group at its initial size 和 Review。
3. 在 Review 页面上，选择 Create Auto Scaling group。
4. 在 Auto Scaling group creation status 页面上，选择 Close。

步骤 3：验证 Auto Scaling 组

现在，您已创建您的 Auto Scaling 组，可验证该组是否已启动 EC2 实例。

验证您的 Auto Scaling 组是否已启动 EC2 实例

1. 在 Auto Scaling Groups 页面上，选择您刚创建的 Auto Scaling 组。
2. Details 选项卡提供了有关 Auto Scaling 组的信息。

Details	Activity History	Scaling Policies	Instances	Notifications	Tags	Scheduled Actions
Launch Configuration my-first-lc						
Load Balancers						
Desired	1	Availability Zone(s)		us-west-2a		
Min	1	Subnet(s)		subnet-cb663da2		
Max	5	Default Cooldown		300		
Health Check Type	EC2	Placement Group				
Health Check Grace Period	300	Suspended Processes				
Termination Policies	Default	Enabled Metrics				
Creation Time	Tue Jan 26 13:20:17 GMT-800 2016	Instance Protection				

3. 在 Activity History 选项卡上，Status 列显示实例的当前状态。当您的实例启动时，状态列将显示 In progress。该实例启动后，状态会变为 Successful。您还可以使用刷新按钮来查看您的实例的当前状态。
4. 在 Instances 选项卡上，Lifecycle 列显示实例的状态。您可以看到 Auto Scaling 组已启动 EC2 实例，并且该实例处于 InService 生命周期状态。Health Status 列显示列在您的实例上进行的 EC2 实例健康检查的结果。

Details	Activity History	Scaling Policies	Instances	Notifications	Tags	Scheduled Actions
Actions ▾						
Filter: Any Health Status ▾ Any Lifecycle State ▾ <input type="text" value="Filter instances..."/>						
<input type="checkbox"/>	Instance ID ▾	Lifecycle ▾	Launch Configuration Name ▾	Availability Zone ▾	Health Status ▾	Protected from ▾
<input type="checkbox"/>	i-cca22415	InService	my-first-lc	us-west-2a	Healthy	

5. (可选) 如果需要，您可以尝试以下实验，以进一步了解 Auto Scaling 的情况。您的 Auto Scaling 组的最小容量为 1 个实例。因此，如果您终止了正在运行的实例，Auto Scaling 必须启动一个新实例替换它。
 - a. 在 Instances 选项卡上，选择实例的 ID。这将在 Instances 页面上显示实例。
 - b. 依次选择 Actions、Instance State 和 Terminate。当系统提示您确认时，选择 Yes, Terminate。
 - c. 在导航窗格上，依次选择 Auto Scaling Groups 和 Activity History。Auto Scaling 组的默认冷却时间是 300 秒（5 分钟），因此您大约需要等 5 分钟才能看到扩展活动。扩展活动开始执行时，您会看到有关第一个实例终止的条目，以及一个有关启动新实例的条目。Instances 选项卡仅显示新实例。
 - d. 在导航窗格中，选择 Instances。此页面同时显示已终止的实例和正在运行的实例。

如果要删除您的 Auto Scaling 设置，请转到下一步。否则，您可以使用此 Auto Scaling 基础设施作为基础，并尝试以下一个或多个操作：

- 保持 Auto Scaling 组中的实例数量 (p. 52)
- 手动扩展 (p. 53)
- 动态扩展 (p. 63)
- 在 Auto Scaling 组扩展时获取 SNS 通知 (p. 106)

步骤 4：(可选) 删除您的 Auto Scaling 基础设施

您可以删除 Auto Scaling 设置，或者只删除 Auto Scaling 组而保留启动配置以供将来使用。

删除 Auto Scaling 组

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择您的 Auto Scaling 组 (例如，`my-first-asg`)。
4. 依次选择 Actions 和 Delete。当系统提示进行确认时，选择 Yes, Delete。

Name 列指示 Auto Scaling 组正在被删除。Desired、Min 以及 Max 列显示 Auto Scaling 组有 0 个实例。

如果您要保留启动配置，请跳过这一步。

删除启动配置

1. 在导航窗格上的 Auto Scaling 下，选择 Launch Configurations。
2. 选择您的启动配置 (例如，`my-first-lc`)。
3. 依次选择 Actions 和 Delete launch configuration。当系统提示进行确认时，选择 Yes, Delete。

教程：设置具有扩展和负载均衡功能的应用程序

可以将负载均衡器附加到 Auto Scaling 组。该负载均衡器在组中的所有实例间自动分配传入流量。有关结合 Auto Scaling 使用 Elastic Load Balancing 的好处的更多信息，请参阅 [将负载均衡器与 Auto Scaling 组结合使用](#) (p. 42)。

在本教程中，在创建 Auto Scaling 组时，可以向该组附加负载均衡器。要将负载均衡器附加到现有 Auto Scaling 组，请参阅[将负载均衡器附加到 Auto Scaling 组](#) (p. 43)。

内容

- [先决条件](#) (p. 15)
- [使用 AWS 管理控制台配置扩展和负载均衡功能](#) (p. 15)
- [使用 AWS CLI 配置扩展和负载均衡功能](#) (p. 17)

先决条件

- (可选) 创建一个 IAM 角色，该角色将为您的应用程序授予对所需 AWS 的访问权限。
- 启动一个实例；务必指定 IAM 角色 (如果已创建) 并将所需的任何配置脚本指定为用户数据。连接到实例并对其进行自定义。例如，您可以安装软件 and 应用程序并复制数据。测试您的实例上的应用程序以确保您的实例配置正确。从您的实例创建自定义 Amazon 系统映像 (AMI)。如果您不再需要实例，可终止它。
- 创建负载均衡器。Elastic Load Balancing 支持两类负载均衡器：传统负载均衡器 和 应用程序负载均衡器。您可以创建任一类型的负载均衡器以附加到 Auto Scaling 组。有关更多信息，请参阅 [Elastic Load Balancing 用户指南](#)。

通过 传统负载均衡器，实例注册到负载均衡器，通过 应用程序负载均衡器，实例作为目标注册到目标组。当您计划将负载均衡器与 Auto Scaling 组配合使用时，无需将 EC2 实例注册到负载均衡器或目标组。在将负载均衡器或目标组附加到 Auto Scaling 组后，Auto Scaling 会在启动实例时将其注册到负载均衡器或目标组。

使用 AWS 管理控制台配置扩展和负载均衡功能

在创建您的 Auto Scaling 组时，请完成以下任务以设置具有扩展和负载均衡功能的应用程序。

任务

- [创建或选择启动配置](#) (p. 16)
- [创建 Auto Scaling 组](#) (p. 16)
- (可选) [验证您的负载均衡器是否已附加到您的 Auto Scaling 组](#) (p. 17)

创建或选择启动配置

如果您已拥有要使用的启动配置，请使用以下过程选择该启动配置。

选择现有的启动配置

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在屏幕顶部的导航栏上，选择在创建负载均衡器时使用的区域。
3. 在导航窗格上的 Auto Scaling 下，选择 Launch Configurations。
4. 选择一个启动配置。
5. 选择 Create Auto Scaling group。

或者，使用下列过程创建新的启动配置。

创建启动配置

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在屏幕顶部的导航栏上，选择在创建负载均衡器时使用的区域。
3. 在导航窗格上的 Auto Scaling 下，选择 Launch Configurations。如果您没有任何 Auto Scaling 资源，则会看到一个欢迎页面；选择 Create Auto Scaling group。
4. 选择 Create launch configuration。
5. 在 Choose AMI 页面上，选择您的自定义 AMI。
6. 在 Choose Instance Type 页面上，选择实例的硬件配置，然后选择 Next: Configure details。
7. 在 Configure Details (配置实例详细信息) 页面中，执行以下操作：
 - a. 对于 Name，为启动配置键入一个名称。
 - b. (可选) 要安全地将凭证分配给您的 EC2 实例，请选择您的 IAM 角色。
 - c. (可选) 如果需要连接到非默认 VPC 内的实例，则对于 Advanced Details 和 IP Address Type，选择 Assign a public IP address to every instance。
 - d. (可选) 要为实例指定用户数据或配置脚本，则对于 Advanced Details 和 User data，粘贴您的配置脚本。
 - e. 选择 Skip to review。
8. 在 Review 页面上，选择 Edit security groups。按照说明操作来选择现有安全组，然后选择 Review。
9. 在 Review 页上选择 Create launch configuration。
10. 在 Select an existing key pair or create a new key pair 页面上，选择所列的选项之一。选中确认复选框，然后选择 Create launch configuration。

Warning

如果需要连接到您的实例，不要选择 Proceed without a key pair。

11. Launch configuration creation status 页面将显示您最近创建的启动配置的状态。选择 Create an Auto Scaling group using this launch configuration。

创建 Auto Scaling 组

选择或创建启动配置后，使用以下步骤从中断的地方继续操作。

创建 Auto Scaling 组的步骤

1. 在 Configure Auto Scaling group details 页面上，执行以下操作：
 - a. 对于 Group name，键入 Auto Scaling 组的名称。
 - b. 对于 Group size，键入 Auto Scaling 组的初始实例数。
 - c. 如果为启动配置选择了需要 VPC 的实例类型（例如 T2 实例），则必须为 Network 选择 VPC。否则，如果您的账户支持 EC2-Classic 而且您已选择不需要 VPC 的实例类型，则可以选择 Launch into EC2-Classic 或 VPC。
 - d. 如果您在上一步中选择了 VPC，请从 Subnet 中选择一个或多个子网。如果您改为选择 EC2-Classic，请从 Availability Zone(s) 中选择一个或多个可用区。
 - e. 对于 Advanced Details，请选择 Receive traffic from Elastic Load Balancer(s)，然后执行下列操作之一：
 - [Classic 负载均衡器] 从 Load Balancers 中选择您的负载均衡器。
 - [目标组] 从 Target Groups 中选择您的目标组。
 - f. 对于 Advanced Details，选择 Receive traffic from Elastic Load Balancer(s)，然后从 Load Balancers 中选择您的负载均衡器。
 - g. （可选）要使用 Elastic Load Balancing 运行状况检查，请为 Advanced Details 和 Health Check Type 选择 ELB。
 - h. 选择 Next: Configure scaling policies。
2. 在 Configure scaling policies 页面上，选择 Keep this group at its initial size，然后选择 Review。

如果您要为您的 Auto Scaling 组配置扩展策略，请参阅 [创建具有目标跟踪扩展策略的 Auto Scaling 组 \(p. 70\)](#)。
3. 审核 Auto Scaling 组的详细信息。您可以选择 Edit 进行更改。完成后，选择 Create Auto Scaling group。

(可选) 验证您的负载均衡器是否已附加到您的 Auto Scaling 组

验证您的负载均衡器是否已附加到您的 Auto Scaling 组

1. 选择您的 Auto Scaling 组。
2. 在 Details 选项卡上，Load Balancers 显示任何附加的负载均衡器，Target Groups 显示任何附加的目标组。
3. 在 Details 选项卡上，Load Balancers 任何附加的负载均衡器。
4. 在 Activity History 选项卡上，Status 列显示 Auto Scaling 实例的状态。实例启动时，其状态是 In progress。该实例启动后，状态会变为 Successful。
5. 在 Instances 选项卡上，Lifecycle 列显示 Auto Scaling 实例的状态。在实例准备好接收流量时，其状态为 InService。

Health Status 列显示了对您的实例执行健康检查的结果。

使用 AWS CLI 配置扩展和负载均衡功能

完成以下任务以设置具有扩展和负载均衡功能的应用程序。

任务

- [创建启动配置 \(p. 18\)](#)

- [创建包含负载均衡器的 Auto Scaling 组 \(p. 18\)](#)

创建启动配置

如果您已经有了要使用的启动配置，请跳过这一步。

创建启动配置的步骤

使用以下 `create-launch-configuration` 命令：

```
aws autoscaling create-launch-configuration --launch-configuration-name my-lc \  
--image-id ami-514ac838 --instance-type m1.small
```

创建包含负载均衡器的 Auto Scaling 组

在创建 Auto Scaling 组时，您可以为该组挂载一个现有的负载均衡器。

创建带附加 传统负载均衡器 的 Auto Scaling 组

将以下 `create-auto-scaling-group` 命令和 `--load-balancer-names` 选项结合使用可创建一个带附加 传统负载均衡器 的 Auto Scaling 组：

```
aws autoscaling create-auto-scaling-group --auto-scaling-group-name my-lb-asg \  
--launch-configuration-name my-lc \  
--availability-zones "us-west-2a" "us-west-2b" \  
--load-balancer-names "my-lb" \  
--max-size 5 --min-size 1 --desired-capacity 2
```

创建带附加的目标组的 Auto Scaling 组

将以下 `create-auto-scaling-group` 命令和 `--target-group-arns` 选项结合使用可创建一个带附加的目标组的 Auto Scaling 组。

```
aws autoscaling create-auto-scaling-group --auto-scaling-group-name my-lb-asg \  
--launch-configuration-name my-lc \  
--vpc-zone-identifier "subnet-41767929" \  
--vpc-zone-identifier "subnet-b7d581c0" \  
--target-group-arns "arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-  
targets/1234567890123456" \  
--max-size 5 --min-size 1 --desired-capacity 2
```


启动配置

启动配置是 Auto Scaling 组用于启动 EC2 实例的模板。创建启动配置时，您可以指定实例的信息，例如 Amazon 系统映像 (AMI) 的 ID、实例类型、一个密钥对、一个或多个安全组和块储存设备映射。如果之前已启动过 EC2 实例，可以指定相同的信息来启动实例。

创建 Auto Scaling 组时必须指定启动配置。您可以为多个 Auto Scaling 组指定启动配置。但是一次只能为一个 Auto Scaling 组指定一个启动配置，而且启动配置在创建后不能修改。因此，如果要更改 Auto Scaling 组的启动配置，必须先创建启动配置，然后使用新的启动配置来更新 Auto Scaling 组。

内容

- [创建启动配置 \(p. 19\)](#)
- [使用 EC2 实例创建启动配置 \(p. 20\)](#)
- [更改 Auto Scaling 组的启动配置 \(p. 24\)](#)
- [在 VPC 中启动 Auto Scaling 实例 \(p. 24\)](#)
- [在 Auto Scaling 组中启动竞价型实例 \(p. 28\)](#)

创建启动配置

在创建启动配置时，必须指定与要启动的 EC2 实例相关的信息，例如 Amazon 系统映像 (AMI)、实例类型、密钥对、安全组和块储存设备映射。

或者，您可以使用正在运行的 EC2 实例的属性来创建启动配置。有关更多信息，请参阅 [使用 EC2 实例创建启动配置 \(p. 20\)](#)。

在创建启动配置后，您可创建 Auto Scaling 组。有关更多信息，请参阅 [创建 Auto Scaling 组 \(p. 36\)](#)。

Auto Scaling 组一次与一个启动配置关联，并且启动配置在创建后不能修改。因此，如果您要更改现有 Auto Scaling 组的启动配置，则必须使用新的启动配置更新它。有关更多信息，请参阅 [更改 Auto Scaling 组的启动配置 \(p. 24\)](#)。

使用控制台创建启动配置

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在屏幕顶部的导航栏上，将显示当前区域。为 Auto Scaling 组选择一个符合您需求的区域。
3. 在导航窗格上的 Auto Scaling 下，选择 Launch Configurations。如果您是初次使用 Auto Scaling，则会看到一个欢迎页面；选择 Create Auto Scaling group。
4. 选择 Create launch configuration。

5. 在 Choose AMI 页面中，选择 AMI。
6. 在 Choose Instance Type 页面上，为您的实例选择硬件配置。选择 Next: Configure details。

Note

必须将 T2 实例启动到 VPC 的子网中。如果选择 `t2.micro` 实例，但是没有 VPC，系统会为您创建一个。此 VPC 在区域的每个可用区中均包含一个公有子网。

7. 在 Configure Details (配置实例详细信息) 页面中，执行以下操作：
 - a. 对于 Name，为启动配置键入一个名称。
 - b. (可选) 对于 IAM role，选择要与实例关联的角色。有关更多信息，请参阅 [使用 IAM 角色启动 Auto Scaling 实例 \(p. 122\)](#)。
 - c. (可选) 默认情况下，已为 Auto Scaling 实例启用基本监控。要为 Auto Scaling 实例启用详细监控，请选择 Enable CloudWatch detailed monitoring。
 - d. 对于 Advanced Details 和 IP Address Type，选择一个选项。要连接至 VPC 中的实例，您必须选择一个分配公有 IP 地址的选项。如果您要连接到实例，但是不确定您是否有默认 VPC，请选择 Assign a public IP address to every instance。
 - e. 选择 Skip to review。
8. 在 Review 页面上，选择 Edit security groups。按照说明操作来选择现有安全组，然后选择 Review。
9. 在 Review 页上选择 Create launch configuration。
10. 对于 Select an existing key pair or create a new key pair，选择所列的选项之一。选中确认复选框，然后选择 Create launch configuration。

Warning

如果需要连接到您的实例，请不要选择 Proceed without a key pair。

使用命令行创建启动配置

您可以使用以下任一命令：

- [create-launch-configuration](#) (AWS CLI)
- [New-ASLaunchConfiguration](#) (适用于 Windows PowerShell 的 AWS 工具)

使用 EC2 实例创建启动配置

Auto Scaling 提供了一个选项，可使用正在运行的 EC2 实例中的属性创建启动配置。使用此选项时，Auto Scaling 将指定实例中的属性复制到一个模板中，这个模板可用来启动一个或多个 Auto Scaling 组。

Tip

您可以[直接从 EC2 实例创建 Auto Scaling 组 \(p. 37\)](#)。如果使用此功能，Auto Scaling 还自动为您创建启动配置。

如果指定的实例有 Auto Scaling 当前不支持的属性，则由 Auto Scaling 使用从确定的实例创建的启动配置启动的实例可能不同于该确定的实例。

从头创建启动配置和从现有 EC2 实例创建启动配置之间存在差异。如果从头创建启动配置，需要指定映像 ID、实例类型、可选资源（如存储设备）和可选设置（如监控）。如果从运行的实例创建启动配置，默认情况下 Auto Scaling 从指定实例派生启动配置的属性，以及该实例启动时所在的 AMI 的块储存设备映射（忽略在启动后添加到该实例的任何附加的块设备）。

如果使用正在运行的实例创建启动配置，通过指定作为相同请求一部分的属性，您可以覆盖以下属性：AMI、块储存设备、密钥对、实例配置文件、实例类型、内核、监控、部署租用、虚拟磁盘、安全组、竞价价格、用户数据、该实例是否有关联的公有 IP 地址，以及该实例是否经过了 EBS 优化。

以下示例介绍如何从 EC2 实例创建启动配置。

示例

- [使用 EC2 实例创建启动配置 \(p. 21\)](#)
- [从实例创建启动配置以及覆盖块储存设备 \(p. 22\)](#)
- [创建启动配置和覆盖实例类型 \(p. 23\)](#)

使用 EC2 实例创建启动配置

要使用现有 EC2 实例的属性创建启动配置，请指定该实例的 ID。

Important

用于启动指定实例的 AMI 必须仍然存在。

使用 AWS 管理控制台从 EC2 实例创建启动配置

您可以使用该控制台从运行中的 EC2 实例创建启动配置和 Auto Scaling 组，还可以将该实例添加到新的 Auto Scaling 组。有关更多信息，请参阅 [将 EC2 实例附加到 Auto Scaling 组 \(p. 55\)](#)。

使用 AWS CLI 从 EC2 实例创建启动配置

使用以下 `create-launch-configuration` 命令，从使用与该实例（启动后添加的任何块储存设备除外，它们将被忽略）具有相同属性的实例创建启动配置：

```
aws autoscaling create-launch-configuration --launch-configuration-name my-lc-from-instance
--instance-id i-a8e09d9c
```

您可以使用以下 `describe-launch-configurations` 命令来描述启动配置，以及验证其属性是否与实例的属性相匹配：

```
aws autoscaling describe-launch-configurations --launch-configuration-names my-lc-from-instance
```

以下为响应示例：

```
{
  "LaunchConfigurations": [
    {
      "UserData": null,
      "EbsOptimized": false,
      "LaunchConfigurationARN": "arn",
      "InstanceMonitoring": {
        "Enabled": false
      },
      "ImageId": "ami-05355a6c",
      "CreatedTime": "2014-12-29T16:14:50.382Z",
      "BlockDeviceMappings": [],
      "KeyName": "my-key-pair",
      "SecurityGroups": [
        "sg-8422d1eb"
      ],
      "LaunchConfigurationName": "my-lc-from-instance",
      "KernelId": "null",
      "RamdiskId": null,
      "InstanceType": "t1.micro",
      "AssociatePublicIpAddress": true
    }
  ]
}
```

```
}  
]  
}
```

从实例创建启动配置以及覆盖块储存设备

默认情况下，Auto Scaling 使用您指定的 EC2 实例中的属性来创建启动配置，除非这些块储存设备来自用于启动该实例的 AMI，而不是该实例。要将块储存设备添加到启动配置，请覆盖该启动配置的块储存设备映射。

Important

用于启动指定实例的 AMI 必须仍然存在。

使用 AWS CLI 创建启动配置和覆盖块储存设备

使用以下 `create-launch-configuration` 命令来创建使用 EC2 实例但具有自定义块储存设备映射的启动配置：

```
aws autoscaling create-launch-configuration --launch-configuration-name my-lc-from-  
instance-bdm --instance-id i-a8e09d9c  
--block-device-mappings "[{\"DeviceName\":\"/dev/sda1\", \"Ebs\":{\"SnapshotId\":  
\"snap-3decf207\"}}, {\"DeviceName\":\"/dev/sdf\", \"Ebs\":{\"SnapshotId\":\"snap-  
eed6ac86\"}}]"
```

使用以下 `describe-launch-configurations` 命令来描述启动配置，并且验证它是否使用您的自定义块储存设备映射：

```
aws autoscaling describe-launch-configurations --launch-configuration-names my-lc-from-  
instance-bdm
```

下面的示例响应描述了该启动配置：

```
{  
  "LaunchConfigurations": [  
    {  
      "UserData": null,  
      "EbsOptimized": false,  
      "LaunchConfigurationARN": "arn",  
      "InstanceMonitoring": {  
        "Enabled": false  
      },  
      "ImageId": "ami-c49c0dac",  
      "CreatedTime": "2015-01-07T14:51:26.065Z",  
      "BlockDeviceMappings": [  
        {  
          "DeviceName": "/dev/sda1",  
          "Ebs": {  
            "SnapshotId": "snap-3decf207"  
          }  
        },  
        {  
          "DeviceName": "/dev/sdf",  
          "Ebs": {  
            "SnapshotId": "snap-eed6ac86"  
          }  
        }  
      ],  
      "KeyName": "my-key-pair",  
      "SecurityGroups": [  
        {  
          "GroupId": "sg-1a2b3c4d"  
        }  
      ]  
    }  
  ]  
}
```

```
        "sg-8637d3e3"
      ],
      "LaunchConfigurationName": "my-lc-from-instance-bdm",
      "KernelId": null,
      "RamdiskId": null,
      "InstanceType": "t1.micro",
      "AssociatePublicIpAddress": true
    }
  ]
}
```

创建启动配置和覆盖实例类型

默认情况下，Auto Scaling 使用您指定的 EC2 实例中的属性创建启动配置。根据具体要求，您可能需要更改其中某些属性。Auto Scaling 为您提供从实例覆盖属性以及使用所需值的选项。例如，您可以覆盖实例类型。

Important

用于启动指定实例的 AMI 必须仍然存在。

使用 AWS CLI 创建启动配置和覆盖实例类型

使用以下 `create-launch-configuration` 命令来创建使用 EC2 实例但实例类型（例如 `m1.small`）不同于该示例（例如 `t1.micro`）的启动配置：

```
aws autoscaling create-launch-configuration --launch-configuration-name my-lc-from-  
instance-changetype --instance-id i-a8e09d9c --instance-type m1.small
```

使用以下 `describe-launch-configurations` 命令来描述启动配置以及验证实例类型是否被覆盖：

```
aws autoscaling describe-launch-configurations --launch-configuration-names my-lc-from-  
instance-changetype
```

下面的示例响应描述了该启动配置：

```
{
  "LaunchConfigurations": [
    {
      "UserData": null,
      "EbsOptimized": false,
      "LaunchConfigurationARN": "arn",
      "InstanceMonitoring": {
        "Enabled": false
      },
      "ImageId": "ami-05355a6c",
      "CreatedTime": "2014-12-29T16:14:50.382Z",
      "BlockDeviceMappings": [],
      "KeyName": "my-key-pair",
      "SecurityGroups": [
        "sg-8422d1eb"
      ],
      "LaunchConfigurationName": "my-lc-from-instance-changetype",
      "KernelId": "null",
      "RamdiskId": null,
      "InstanceType": "m1.small",
      "AssociatePublicIpAddress": true
    }
  ]
}
```

```
}
```

更改 Auto Scaling 组的启动配置

Auto Scaling 组一次与一个启动配置关联，并且启动配置在创建后不能修改。要更改 Auto Scaling 组的启动配置，可以使用现有的启动配置作为基础来创建新的启动配置，然后将 Auto Scaling 组更新为使用新的启动配置。

更改 Auto Scaling 组的启动配置后，将使用新的配置选项启动所有新实例，但现有实例不受影响。

使用控制台更改 Auto Scaling 组的启动配置

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格中，选择 Launch Configurations。
3. 选择启动配置，选择 Actions，然后单击 Copy launch configuration。这将设置与原启动配置选项相同的新启动配置，但在名称中会增加“Copy”文本。
4. 在 Copy Launch Configuration 页面上，根据需要编辑配置选项，然后选择 Create launch configuration。
5. 在确认页面上，选择 View your Auto Scaling groups。
6. 选择 Auto Scaling 组，然后依次选择 Details 和 Edit。
7. 从 Launch Configuration 中选择新的启动配置，然后选择 Save。

使用 AWS CLI 更改 Auto Scaling 组的启动配置

1. 使用 [describe-launch-configurations](#) 命令描述当前的启动配置。
2. 使用 [create-launch-configuration](#) 命令创建新的启动配置。
3. 带 `--launch-configuration-names` 参数使用 [update-auto-scaling-group](#) 命令更新 Auto Scaling 组的启动配置。

使用 Windows PowerShell 工具 更改 Auto Scaling 组的启动配置

1. 使用 [Get-ASLaunchConfiguration](#) 命令描述当前的启动配置。
2. 使用 [New-ASLaunchConfiguration](#) 命令创建新的启动配置。
3. 带 `-LaunchConfigurationName` 参数使用 [Update-ASAutoScalingGroup](#) 命令更新 Auto Scaling 组的启动配置。

在 VPC 中启动 Auto Scaling 实例

Amazon Virtual Private Cloud (Amazon VPC) 使您能够在私有、隔离的 AWS 云区域定义虚拟的联网环境。您的虚拟联网环境完全由您控制。有关更多信息，请参阅 [Amazon VPC 用户指南](#)。

在 Virtual Private Cloud (VPC) 内，您可启动 AWS 资源，如 Auto Scaling 组。VPC 中 Auto Scaling 组的工作原理与它在 Amazon EC2 中的工作原理基本相同，支持同样的功能集。

Amazon VPC 中的子网是可用区的一个部分，由 VPC 的 IP 地址范围段定义。您可以根据您的安全和运行需求，使用子网对您的实例进行分组。子网完全位于其最初创建时所在的可用区内。在子网中启动 Auto Scaling 实例。

要支持 Internet 与子网中的实例之间的通信，您必须创建 Internet 网关并将它挂载到 VPC。通过 Internet 网关，子网中的资源可通过 Amazon EC2 网络边界连接到 Internet。如果子网的流量路由到 Internet 网关，则

该子网就叫做公有子网。如果子网的流量不路由到 Internet 网关，则该子网就叫做私有子网。请为必须连接到 Internet 的资源使用公有子网，为不需要连接到 Internet 的资源使用私有子网。

先决条件

在 VPC 中启动 Auto Scaling 实例之前，您必须先创建 VPC 环境。创建了 VPC 和子网后，在子网内启动 Auto Scaling 实例。创建带单个公有子网的 VPC 的最简单方法就是使用 VPC 向导。有关更多信息，请参阅 [Amazon VPC 入门指南](#)。

内容

- [默认 VPC \(p. 25\)](#)
- [VPC 中的 IP 寻址 \(p. 25\)](#)
- [实例部署租期 \(p. 25\)](#)
- [将 EC2-Classic 实例链接到 VPC \(p. 26\)](#)
- [示例 \(p. 28\)](#)

默认 VPC

如果您在 2013 年 12 月 4 日以后创建了 AWS 账户，或者您要在新区域中创建 Auto Scaling 组，则我们会为您创建默认 VPC。默认 VPC 随每个可用区中的默认子网提供。如果您有默认 VPC，则在默认情况下，将在默认 VPC 中创建 Auto Scaling 组。

有关默认 VPC 以及检查您的账户是否附带默认 VPC 的信息，请参阅 Amazon VPC 开发人员指南 中的 [您的默认 VPC 和子网](#)。

VPC 中的 IP 寻址

在 VPC 中启动 Auto Scaling 实例时，将自动为这些实例分配子网地址范围内的私有 IP 地址。这样，这些实例就能够与 VPC 中的其他实例通信。

您可以进行启动配置，为您的实例分配公有 IP 地址。将公有 IP 地址分配给实例，可让这些实例与 Internet 或 AWS 中的其他服务通信。

当您为实例启用公有 IP 地址，且将这些实例启动到子集（该子集配置为向实例自动分配 IPv6 地址）时，这些实例可以接收 IPv4 和 IPv6 地址。否则，这些实例只接收 IPv4 地址。有关更多信息，请参阅 Amazon EC2 用户指南（适用于 Linux 实例）中的 [IPv6 地址](#)。

实例部署租期

专用实例在主机硬件级别上与您的非专用实例和属于其他 AWS 账户的实例实现物理隔离。当您创建 VPC 时，默认情况下它的租期属性设置为 `default`。在这样一个 VPC 中，您可启动租期值为 `dedicated` 的实例，以便它们作为单一租期实例运行。否则，默认情况下，这些实例作为共享租期实例运行。如您将 VPC 的租期属性设置为 `dedicated`，则在 VPC 内启动的所有实例都作为单一租期实例运行。有关更多信息，请参阅 Amazon VPC 用户指南 中的 [Dedicated Instances](#)。有关定价信息，请参阅 [Amazon EC2 Dedicated Instances](#)。

在创建启动配置后，实例部署租期的默认值为 `null`，实例租期由 VPC 的租期属性控制。下表总结了在 VPC 中启动的 Auto Scaling 实例的实例部署租期。

启动配置租期	VPC 租期 = 默认	VPC 租期 = 专用
未指定	共享租期实例	专用实例
<code>default</code>	共享租期实例	专用实例

启动配置租期	VPC 租期 = 默认	VPC 租期 = 专用
dedicated	专用实例	专用实例

您可使用 [create-launch-configuration](#) 命令和 `--placement-tenancy` 选项，将您的启动配置的实例部署租期指定为 `default` 或 `dedicated`。例如，以下命令将启动配置租期设置为 `dedicated`：

```
aws autoscaling create-launch-configuration --launch-configuration-name my-launch-config --placement-tenancy dedicated --image-id ...
```

您可以使用以下 [describe-launch-configurations](#) 命令验证启动配置的实例部署租期：

```
aws autoscaling describe-launch-configurations --launch-configuration-names my-launch-config
```

以下是创建专用实例的启动配置的示例输出。请注意 `PlacementTenancy` 不是此命令输出的一部分，除非您已明确设置了实例部署租期。

```
{
  "LaunchConfigurations": [
    {
      "UserData": null,
      "EbsOptimized": false,
      "PlacementTenancy": "dedicated",
      "LaunchConfigurationARN": "arn",
      "InstanceMonitoring": {
        "Enabled": true
      },
      "ImageId": "ami-b5a7ea85",
      "CreatedTime": "2015-03-08T23:39:49.011Z",
      "BlockDeviceMappings": [],
      "KeyName": null,
      "SecurityGroups": [],
      "LaunchConfigurationName": "my-launch-config",
      "KernelId": null,
      "RamdiskId": null,
      "InstanceType": "m3.medium"
    }
  ]
}
```

将 EC2-Classic 实例链接到 VPC

如果您在 EC2-Classic 的 Auto Scaling 组中启动实例，您可使用 `ClassicLink` 将这些实例链接到 VPC。`ClassicLink` 让您能够将 VPC 的一个或多个安全组关联到 Auto Scaling 组中的 EC2-Classic 实例，从而让这些链接的 EC2-Classic 实例与 VPC 中的实例通过私有 IP 地址进行通信。有关更多信息，请参阅 Amazon EC2 用户指南（适用于 Linux 实例）中的 [ClassicLink](#)。

如果您的 Auto Scaling 组中存在运行的 EC2-Classic 实例，您可将它们链接到启用 `ClassicLink` 的 VPC。有关更多信息，请参阅 Amazon EC2 用户指南（适用于 Linux 实例）中的 [将实例链接到 VPC](#)。或者，您可以更新 Auto Scaling 组来使用一个启动配置，该启动配置在启动时将 EC2-Classic 实例自动链接到一个 VPC，然后终止运行的实例并让 Auto Scaling 启动链接到该 VPC 的新实例。

使用 AWS 管理控制台链接到 VPC

使用以下程序创建启动配置，通过该启动配置将 EC2-Classic 实例链接到指定的 VPC，并更新现有 Auto Scaling 组来使用启动配置。

使用控制台，将 Auto Scaling 组中的 EC2-Classic 实例链接到 VPC。

1. 确认 VPC 已启用 ClassicLink。有关更多信息，请参阅 Amazon EC2 用户指南（适用于 Linux 实例）中的[查看启用了 ClassicLink 的 VPC](#)。
2. 为 EC2-Classic 实例将要链接到的 VPC 创建一个安全组，并设定规则，管理链接的 EC2-Classic 实例与 VPC 中的实例之间的通信。
3. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
4. 在导航窗格上，选择 Launch Configurations。如果您是首次接触 Auto Scaling，您将看到欢迎页面。选择 Create Auto Scaling group。
5. 选择 Create launch configuration。
6. 在 Choose AMI 页面中，选择 AMI。
7. 在 Choose an Instance Type 页面上，选择实例类型，然后选择 Next: Configure details。
8. 在 Configure details 页面上，执行以下操作：
 - a. 为您的启动配置键入一个名称。
 - b. 展开 Advanced Details，选择您需要的 IP Address Type，然后选择 Link to VPC。
 - c. 对于 VPC，选择在步骤 1 中启用了 ClassicLink 的 VPC。
 - d. 对于 Security Groups，选择在步骤 2 中创建的安全组。
 - e. 选择 Skip to review。
9. 在 Review 页面上，进行所需的更改，然后选择 Create launch configuration。对于 Select an existing key pair or create a new key pair，选择一个选项，选中确认复选框（如果出现），然后选择 Create launch configuration。
10. 当出现提示时，请遵循说明创建使用新启动配置的 Auto Scaling 组。务必为 Network 选择 Launch into EC2-Classic。否则，请选择 Cancel 并将您的启动配置添加到一个现有 Auto Scaling 组，如下所示：
 - a. 在导航窗格上，选择 Auto Scaling Groups。
 - b. 选择您的 Auto Scaling 组，然后选择 Actions 和 Edit。
 - c. 对于 Launch Configuration，选择新的启动配置，然后选择 Save。

使用 AWS CLI 链接到 VPC

使用以下程序创建启动配置，通过该启动配置将 EC2-Classic 实例链接到指定的 VPC，并更新现有 Auto Scaling 组来使用该启动配置。

使用 AWS CLI 将 Auto Scaling 组中的 EC2-Classic 实例链接到 VPC。

1. 确认 VPC 已启用 ClassicLink。有关更多信息，请参阅 Amazon EC2 用户指南（适用于 Linux 实例）中的[查看启用了 ClassicLink 的 VPC](#)。
2. 为 EC2-Classic 实例将要链接到的 VPC 创建一个安全组，并设定规则，管理链接的 EC2-Classic 实例与 VPC 中的实例之间的通信。
3. 使用以下 [create-launch-configuration](#) 命令创建一个启动配置，其中 vpc_id 是步骤 1 中启用了 ClassicLink 的 VPC 的 ID，而 group_id 是步骤 2 中创建的安全组。

```
aws autoscaling create-launch-configuration --launch-configuration-name classiclink-config
--image-id ami_id --instance-type instance_type
--classic-link-vpc-id vpc_id --classic-link-vpc-security-groups group_id
```

4. 使用在上一步中创建的启动配置来更新您的现有 Auto Scaling 组，例如 my-asg。这一 Auto Scaling 组中启动的任何新 EC2-Classic 实例都是已链接的 EC2-Classic 实例。使用以下 [update-auto-scaling-group](#) 命令：

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg
```

```
--launch-configuration-name classiclink-config
```

或者，您也可以为通过 [create-auto-scaling-group](#) 创建的新 Auto Scaling 组使用此启动配置。

示例

有关示例，请参阅以下教程：

- [Auto Scaling 入门 \(p. 11\)](#)
- [在 Amazon Web Services 上托管 Web 应用程序](#)
- [在 Amazon Web Services 上托管 NET Web 应用程序](#)

在 Auto Scaling 组中启动竞价型实例

如果您能灵活控制应用程序的运行时间并且应用程序可以中断，那么相对于按需实例，竞价型实例是经济实惠之选。您可以将 Auto Scaling 设置为启动竞价型实例，而不是按需实例。

在使用 Auto Scaling 启动竞价型实例之前，我们建议您熟悉使用 Amazon EC2 启动和管理竞价型实例的操作。有关更多信息，请参阅 Amazon EC2 用户指南（适用于 Linux 实例）中的[竞价型实例](#)。

以下是 Auto Scaling 使用竞价型实例的方式：

- 设置您的出价。使用 Auto Scaling 启动竞价型实例时，需要在启动配置中设置出价。您不能使用单个启动配置同时启动按需实例和竞价型实例。
- 更改您的出价。要更改竞价出价，您必须使用新出价创建一个启动配置，然后将其与 Auto Scaling 组相关联。请注意，只要现有实例所用的启动配置中指定的出价高于当前竞价市场价格，这些实例就会继续运行。
- 竞价市场价格和您的出价。如果竞价型实例的市场价格超过 Auto Scaling 组中正在运行的实例的竞价出价，Amazon EC2 会终止实例。如果竞价出价恰好与竞价市场价格一样，您的出价是否履行取决于几个因素，如可用的竞价型实例容量。
- 维护您的竞价型实例。竞价型实例终止后，Auto Scaling 会尝试启动替换实例来保持该组的所需容量。如果出价高于市场价格，则启动竞价型实例。否则，Auto Scaling 会不断尝试。
- 在可用区之间平衡。如果您指定多个可用区，Auto Scaling 会在这些可用区之间分配出价。如果您的竞价出价在一个可用区中过低而无法执行任何出价，Auto Scaling 会检查出价是否在其他可用区中被执行。如果是，Auto Scaling 会取消失败的出价并在执行了出价的可用区之间重新分配它们。如果没有执行出价的可用区中的价格降得足够低以使未来的出价能够成功，Auto Scaling 会在所有可用区之间重新平衡。有关更多信息，请参阅[再平衡活动 \(p. 6\)](#)。
- Auto Scaling 和竞价型实例终止。Auto Scaling 能够终止或替代竞价型实例，就像它能够终止或替代按需实例一样。有关更多信息，请参阅[控制 Auto Scaling 在缩小过程中终止哪些实例 \(p. 78\)](#)。

内容

- [使用 AWS 管理控制台启动竞价型实例 \(p. 28\)](#)
- [使用 AWS CLI 启动竞价型实例 \(p. 31\)](#)

使用 AWS 管理控制台启动竞价型实例

要创建启动竞价型实例的 Auto Scaling 组，请完成以下任务：

任务

- [创建启动配置 \(p. 29\)](#)
- [创建 Auto Scaling 组 \(p. 29\)](#)
- [验证并检查您的实例 \(p. 29\)](#)
- [\(可选 \) 在 Auto Scaling 组发生变化时获取通知 \(p. 30\)](#)
- [\(可选 \) 更新出价 \(p. 30\)](#)
- [清除 \(p. 30\)](#)

创建启动配置

创建启动配置

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Launch Configurations。如果您是初次使用 Auto Scaling，则会看到一个欢迎页面；选择 Create Auto Scaling group。
3. 选择 Create launch configuration。
4. 在 Choose AMI 页面中，选择 AMI。
5. 在 Choose Instance Type 页面上，为您的实例选择硬件配置。选择 Next: Configure details。
6. 在 Configure Details (配置实例详细信息) 页面中，执行以下操作：
 - a. 对于 Name，为启动配置键入一个名称。考虑在该名称中包含“Spot”和出价。
 - b. 选择 Request Spot Instances。如果选择此选项，可以看到区域中可用区的当前价格。对于 Maximum price，键入您的出价。
 - c. 对于 Advanced Details，选择 IP 地址类型。如果要连接至 VPC 中的一个实例，您必须选择一个指定了公有 IP 地址的选项。如果您想连接到实例，但是不确定您是否有默认 VPC，请选择 Assign a public IP address to every instance。
 - d. 选择 Skip to review。
7. 在 Review 页面上，选择 Edit security groups。按照说明操作来选择现有安全组，然后选择 Review。
8. 在 Review 页面上，注意启动配置详细信息包含您的出价。选择 Create launch configuration。
9. 在 Select an existing key pair or create a new key pair 页面上，选择所列的选项之一。选中确认复选框，然后选择 Create launch configuration。

Warning

如果需要连接到您的实例，请不要选择 Proceed without a key pair。

创建 Auto Scaling 组

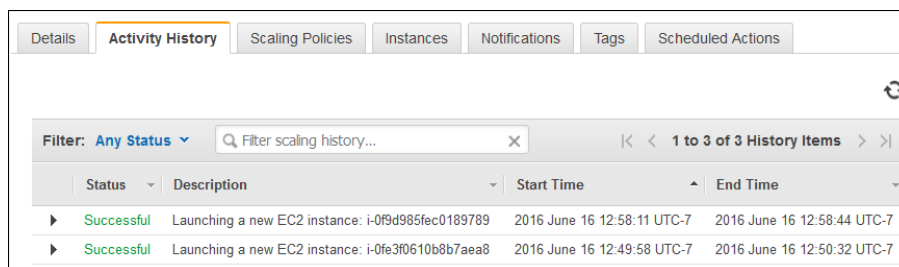
创建 Auto Scaling 组时，可指定您刚创建的启动配置。请注意，启动配置是实例的模板，它包括您的竞价型实例出价。

有关如何使用 AWS 管理控制台创建 Auto Scaling 组的更多信息和说明，请参阅[创建 Auto Scaling 组 \(p. 36\)](#)。

验证并检查您的实例

确认 Auto Scaling 正在启动竞价型实例

1. 选择您的新 Auto Scaling 组。
2. 在 Activity History 选项卡上，它显示 Auto Scaling 已成功启动您请求的竞价型实例。

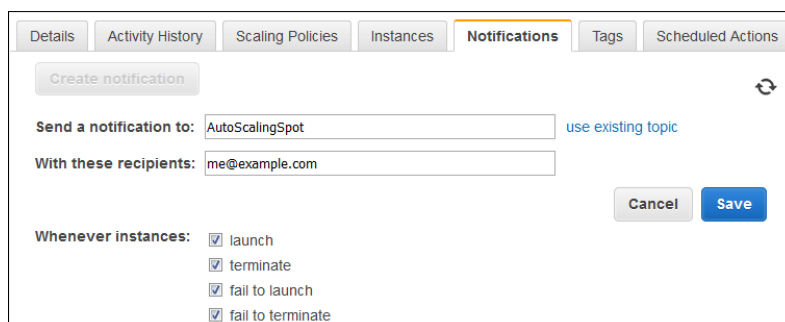


3. 在 Instances 选项卡上，它显示有关竞价型实例的详细信息。您将看到 Auto Scaling 正在启动您在指定的可用区中请求的实例。

(可选) 在 Auto Scaling 组发生变化时获取通知

设置通知

1. 选择 Auto Scaling 组。
2. 在 Notifications 选项卡上，选择 Create notification。
3. 选择 create topic，指定以下内容，然后选择 Save：
 - Send a notification to (发送通知到) – AutoScalingSpot
 - With these recipients (收件人如下) – 您的电子邮件账户
 - Whenever instances - 启动、终止、未能启动和未能终止状态中的一个或多个



在创建通知主题后，您指定的电子邮件账户将立即收到确认电子邮件。

(可选) 更新出价

更新竞价型实例的出价

1. 创建一个启动配置，其设置与[创建启动配置](#) (p. 29)中相同，但名称和最高价格不同。
2. 选择您的 Auto Scaling 组。
3. 在 Details 选项卡上，选择 Edit。
4. 选择您刚刚创建的启动配置，然后选择 Save。

清除

在使用实例和 Auto Scaling 组后，建议将其清除。删除 Auto Scaling 组时，将同时删除所有竞价型实例和对该组的未出价。

清除 Auto Scaling 组和实例

1. 选择您的 Auto Scaling 组。
2. 依次选择 Actions 和 Delete。
3. 当系统提示进行确认时，选择 Yes, Delete。

使用 AWS CLI 启动竞价型实例

要创建启动竞价型实例的 Auto Scaling 组，请完成以下任务：

任务

- [创建启动配置 \(p. 31\)](#)
- [创建 Auto Scaling 组 \(p. 31\)](#)
- [验证并检查您的实例 \(p. 31\)](#)
- (可选) [在 Auto Scaling 组发生变化时获取通知 \(p. 33\)](#)
- (可选) [更新竞价型实例的出价 \(p. 33\)](#)
- [清除 \(p. 34\)](#)

创建启动配置

要使用 Auto Scaling 为竞价型实例出价，请在 `create-launch-configuration` 命令中使用 `--spot-price` 选项，从而指定您愿意为实例支付的最高价格，如下所示：

```
aws autoscaling create-launch-configuration --launch-configuration-name spot-lc-5cents --image-id ami-1a2bc4d --instance-type m1.small --spot-price "0.05"
```

创建 Auto Scaling 组

使用 `create-auto-scaling-group` 命令与刚才创建的启动配置来创建 Auto Scaling 组。下面的命令启动 2 个竞价型实例：

```
aws autoscaling create-auto-scaling-group --auto-scaling-group-name spot-asg --launch-configuration-name spot-lc-5cents --availability-zones "us-west-2a" "us-west-2b" --max-size 5 --min-size 1 --desired-capacity 2
```

验证并检查您的实例

使用 `describe-scaling-activities` 命令列出 Auto Scaling 为您的 Auto Scaling 组执行的活动，如下所示：

```
aws autoscaling describe-scaling-activities --auto-scaling-group-name spot-asg
```

如果最初无法同时履行两个出价，则输出看起来类似于以下示例，其中一个出价成功，而 Auto Scaling 在等待另一个出价：

```
{
  "Activities": [
    {
      "Description": "Placing Spot instance request. Status Reason: Placed Spot instance request: sir-036wjsp9. Waiting for instance(s)",
      "AutoScalingGroupName": "spot-asg",
      "ActivityId": "28189e6b-e14f-4783-8d48-4d03b40b1354",
```

```
    "Details": "{\"Availability Zone\":\"us-west-2a\"}",
    "StartTime": "2015-03-01T16:21:41.578Z",
    "Progress": 20,
    "Cause": "At 2015-03-01T16:21:40Z a difference between desired and actual
capacity changing the desired capacity, increasing the capacity from 0 to 2.",
    "StatusMessage": "Placed Spot instance request: sir-036wjsp9. Waiting for
instance(s)",
    "StatusCode": "WaitingForSpotInstanceId"
  },
  {
    "Description": "Launching a new EC2 instance: i-d95eb0d4",
    "AutoScalingGroupName": "spot-asg",
    "ActivityId": "b987ab02-f7c3-4948-a0bc-5d1449de30ec",
    "Details": "{\"Availability Zone\":\"us-west-2b\"}",
    "StartTime": "2015-03-01T16:21:41.578Z",
    "Progress": 100,
    "EndTime": "2015-03-01T16:29:46Z",
    "Cause": "At 2015-03-01T16:21:40Z a difference between desired and actual
capacity changing the desired capacity, increasing the capacity from 0 to 2.",
    "StatusCode": "Successful"
  }
]
}
```

如果 `as-describe-scaling-activities` 的输出包括 `Failed` 活动，则查看响应了解详细信息。例如，可能 AMI ID 不再有效，或者它与您选择的实例类型不兼容。如果没有给出原因，请检查您的出价是否高于该可用区的竞价市场价格。

要查看有关 Auto Scaling 组的信息，请使用 `describe-auto-scaling-groups` 命令，如下所示：

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name spot-asg
```

在下面的示例输出中，Auto Scaling 按照您的指定，启动了两个实例，这两个实例都在运行：

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupARN": "arn",
      "HealthCheckGracePeriod": 0,
      "SuspendedProcesses": [],
      "DesiredCapacity": 2,
      "Tags": [],
      "EnabledMetrics": [],
      "LoadBalancerNames": [],
      "AutoScalingGroupName": "spot-asg",
      "DefaultCooldown": 300,
      "MinSize": 1,
      "Instances": [
        {
          "InstanceId": "i-d95eb0d4",
          "AvailabilityZone": "us-west-2b",
          "HealthStatus": "Healthy",
          "LifecycleState": "InService",
          "LaunchConfigurationName": "spot-lc-5cents"
        },
        {
          "InstanceId": "i-13d7dc1f",
          "AvailabilityZone": "us-west-2a",
          "HealthStatus": "Healthy",
          "LifecycleState": "InService",
          "LaunchConfigurationName": "spot-lc-5cents"
        }
      ]
    }
  ],
}
```

```
        "MaxSize": 5,
        "VPCZoneIdentifier": null,
        "TerminationPolicies": [
            "Default"
        ],
        "LaunchConfigurationName": "spot-1c-5cents",
        "CreateTime": "2015-03-01T16:12:35.608Z",
        "AvailabilityZones": [
            "us-west-2b",
            "us-west-2a"
        ],
        "HealthCheckType": "EC2"
    }
}
]
```

除了使用 `describe-auto-scaling-groups` 外，您还可以使用 `describe-auto-scaling-instances` 命令，如下所示：

```
aws autoscaling describe-auto-scaling-instances
```

下面是示例输出：

```
{
  "AutoScalingInstances": [
    {
      "AvailabilityZone": "us-west-2a",
      "InstanceId": "i-13d7dc1f",
      "AutoScalingGroupName": "spot-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService",
      "LaunchConfigurationName": "spot-1c-5cents"
    },
    {
      "AvailabilityZone": "us-west-2b",
      "InstanceId": "i-d95eb0d4",
      "AutoScalingGroupName": "spot-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService",
      "LaunchConfigurationName": "spot-1c-5cents"
    }
  ]
}
```

(可选) 在 Auto Scaling 组发生变化时获取通知

有关在 Auto Scaling 中设置电子邮件通知的信息，请参阅在 [Auto Scaling 组扩展时获取 SNS 通知](#) (p. 106)。

(可选) 更新竞价型实例的出价

更新竞价型实例出价的步骤

1. 创建一个启动配置，其设置与之前相同，但名称和最高价格不同，如下所示：

```
aws autoscaling create-launch-configuration --launch-configuration-name spot-1c-7cents
--image-id ami-1a2b3c4d --instance-type m1.small --spot-price "0.07"
```

2. 通过使用 `update-auto-scaling-group` 命令，修改您的 Auto Scaling 组，以使用新的启动配置，如下所示：

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name spot-asg --launch-configuration-name spot-lc-7cents
```

3. 使用 [describe-scaling-activities](#) 命令查看您的更改，如下所示：

```
aws autoscaling describe-scaling-activities --auto-scaling-group-name spot-asg
```

清除

在使用实例和 Auto Scaling 组后，建议将其清除。在如下 [delete-auto-scaling-group](#) 命令中使用可选 `--force-delete` 参数，该参数指定 Auto Scaling 组中的实例随 Auto Scaling 组一同终止，即使实例仍在运行也是如此。否则，您必须终止这些实例，然后才能删除 Auto Scaling 组。

```
aws autoscaling delete-auto-scaling-group --auto-scaling-group-name spot-asg --force-delete
```


Auto Scaling 组

Auto Scaling 组 包含具有相似特征的 EC2 实例集合，这些实例被视为一个逻辑组以便进行实例扩展和管理。例如，如果单个应用程序操作多个实例，可能需要增加该组中的实例数量来提高应用程序性能，或者在需求下降时减少实例数量来降低成本。您可以使用 Auto Scaling 组根据指定的条件自动扩展实例数量，或维持固定数量的实例，即使实例运行状况不佳也是如此。这种在 Auto Scaling 组中自动扩展和维护实例数量的功能是 Auto Scaling 服务的核心功能。

Auto Scaling 组首先启动足够的 EC2 实例以满足其所需容量。Auto Scaling 组通过对组中的实例执行定期运行状况检查来维持此实例数量。如果某个实例运行状况不佳，则该组终止运行状况不佳的实例，并启动其他实例来替换它。有关运行状况检查替换的更多信息，请参阅[保持 Auto Scaling 组中的实例数量 \(p. 52\)](#)。

您可以使用扩展策略自动增加或减少组中正在运行的 EC2 实例的数量，以符合不断变化的条件。扩展策略生效时，Auto Scaling 组调整组的所需容量并根据需要启动或终止实例。如果手动扩展或按计划进行扩展，则必须调整组的所需容量，才能使更改生效。有关更多信息，请参阅[扩展 Auto Scaling 组的大小 \(p. 51\)](#)。

开始规划之前，请花些时间全面考察应用程序在 AWS 云中运行时的情况。记录以下内容：

- 启动和配置服务器需要多长时间
- 哪些指标与应用程序的性能关系最密切
- 您希望 Auto Scaling 组跨多少个可用区
- 您希望 Auto Scaling 发挥什么作用。是否希望 Auto Scaling 进行扩展以增加或减少容量？是否只希望 Auto Scaling 确保始终运行特定数量的服务器？（请记住，Auto Scaling 可以同时实现这两个目标。）
- 您可以使用哪些现有资源（如 EC2 实例或 AMI）

您越了解您的应用程序，Auto Scaling 架构的效率就越高。

内容

- [创建 Auto Scaling 组 \(p. 36\)](#)
- [使用 EC2 实例创建 Auto Scaling 组 \(p. 37\)](#)
- [使用 Amazon EC2 启动向导创建 Auto Scaling 组 \(p. 39\)](#)
- [标记 Auto Scaling 组和实例 \(p. 39\)](#)
- [将负载均衡器与 Auto Scaling 组结合使用 \(p. 42\)](#)
- [将 Auto Scaling 组合并为单个多区组 \(p. 47\)](#)
- [删除 Auto Scaling 基础设施 \(p. 49\)](#)

创建 Auto Scaling 组

创建 Auto Scaling 组时，必须指定用于启动实例的启动配置，以及组必须始终保持的最小实例数。要充分利用 Auto Scaling 组，您还应指定以下项：

- 可用区或子网。构建或修改 AWS 中的应用程序以使用多个可用区通常是个好方法。如果您的 Auto Scaling 组在 VPC 中运行，则可以选择指定希望 Auto Scaling 使用的子网。
- 指标和运行状况检查。有效的 Auto Scaling 组利用指标确定应在何时启动或终止实例。此外，定义运行状况检查也很有用，Auto Scaling 可以通过检查确定实例是否运行良好，或者（如果运行不良）Auto Scaling 是否应终止并替换实例。

或者，您可以使用 EC2 实例（而非启动配置）创建 Auto Scaling 组。有关更多信息，请参阅 [使用 EC2 实例创建 Auto Scaling 组 \(p. 37\)](#)。

先决条件

创建启动配置。有关更多信息，请参阅 [创建启动配置 \(p. 19\)](#)。

使用控制台创建 Auto Scaling 组

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在屏幕顶部的导航栏中，选择在创建启动配置时使用的同一区域。
3. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
4. 选择 Create Auto Scaling group。
5. 在 Create Auto Scaling Group 页面上，选择 Create an Auto Scaling group from an existing launch configuration，选择一个启动配置，然后选择 Next Step。

Note

如果您没有任何启动配置，系统首先会提示您创建一个启动配置，然后才能继续创建 Auto Scaling 组的步骤。

6. 在 Configure Auto Scaling group details 页面上，执行以下操作：
 - a. 对于 Group name，键入 Auto Scaling 组的名称。
 - b. 对于 Group size，键入 Auto Scaling 组的初始实例数。
 - c. 如果为启动配置选择了需要 VPC 的实例类型（例如 T2 实例），则必须为 Network 选择 VPC。否则，如果您的账户支持 EC2-Classic 而且您已选择不需要 VPC 的实例类型，则可以选择 Launch into EC2-Classic 或 VPC。
 - d. 如果您在上一步中选择了 VPC，请从 Subnet 中选择一个或多个子网。如果您改为选择 EC2-Classic，请从 Availability Zone(s) 中选择一个或多个可用区。
 - e. 选择 Next: Configure scaling policies。
7. 在 Configure scaling policies 页面上，选择下列选项之一，然后选择 Review：
 - 要根据需要手动调整 Auto Scaling 组的大小，请选择 Keep this group at its initial size。有关更多信息，请参阅 [手动扩展 \(p. 53\)](#)。
 - 要根据您指定的条件自动调整 Auto Scaling 的大小，则选择 Use scaling policies to adjust the capacity of this group，然后按照指示操作。有关详细信息，请参阅 [配置扩展策略 \(p. 71\)](#)。
8. （可选）要立即添加标签，请选择 Edit tags，然后完成以下步骤。或者，您可以稍后添加标签。有关更多信息，请参阅 [标记 Auto Scaling 组和实例 \(p. 39\)](#)。
 - a. 对于 Key 和 Value，键入第一个标签的键和值。
 - b. 如果要使 Auto Scaling 将标签传播到 Auto Scaling 组启动的实例，则使 Tag New Instances 保持为选中状态。

- c. 选择 Add tag 以添加其他标签，然后键入标签的键和值。
 - d. 选择 Review。
9. 在 Review 页面上，选择 Create Auto Scaling group。
 10. 在 Auto Scaling group creation status 页面上，选择 Close。

使用命令行创建 Auto Scaling 组

您可以使用以下任一命令：

- [create-auto-scaling-group](#) (AWS CLI)
- [New-ASAutoScalingGroup](#) (适用于 Windows PowerShell 的 AWS 工具)

使用 EC2 实例创建 Auto Scaling 组

Auto Scaling 为您提供了一个选项，即通过指定 EC2 实例（而不是启动配置）并指定 Auto Scaling 组的属性（如 EC2 实例的最小、最大和所需数量）创建 Auto Scaling 组。

当您使用 EC2 实例创建 Auto Scaling 组时，Auto Scaling 自动创建启动配置并将其与 Auto Scaling 组关联。该启动配置的名称与 Auto Scaling 组相同，同时它从指定实例派生其属性，如 AMI ID、实例类型和可用区。

限制

以下是使用 EC2 实例创建 Auto Scaling 组的限制条件：

- 如果识别的实例有标签，标签不会复制到新 Auto Scaling 组的 `Tags` 属性。
- Auto Scaling 组包括 AMI 的块储存设备映射，用来启动实例；不包括实例启动后挂载的任何块储存设备。
- 如果识别的实例在一个以上的负载均衡器注册，负载均衡器名称不会复制到新 Auto Scaling 组的 `LoadBalancerNames` 属性。

先决条件

开始之前，请使用 Amazon EC2 控制台或者 [describe-instances](#) 命令 (AWS CLI) 查找 EC2 实例的 ID。

EC2 实例必须符合以下标准：

- 实例位于您要创建 Auto Scaling 组的可用区中。
- 实例不是其他 Auto Scaling 组的成员。
- 实例处于 `running` 状态。
- 用于启动实例的 AMI 必须仍然存在。

内容

- [通过控制台使用 EC2 实例创建 Auto Scaling 组 \(p. 37\)](#)
- [使用 AWS CLI 从 EC2 实例创建 Auto Scaling 组 \(p. 38\)](#)

通过控制台使用 EC2 实例创建 Auto Scaling 组

您可以使用该控制台从运行中的 EC2 实例创建一个 Auto Scaling 组并将该实例添加到新的 Auto Scaling 组。有关更多信息，请参阅 [将 EC2 实例附加到 Auto Scaling 组 \(p. 55\)](#)。

使用 AWS CLI 从 EC2 实例创建 Auto Scaling 组

使用以下 `create-auto-scaling-group` 命令，从 EC2 实例 `i-7f12e649` 创建 Auto Scaling 组 `my-asg-from-instance`。

```
aws autoscaling create-auto-scaling-group --auto-scaling-group-name my-asg-from-instance --instance-id i-7f12e649 --min-size 1 --max-size 2 --desired-capacity 2
```

使用以下 `describe-auto-scaling-groups` 命令可创建 Auto Scaling 组。

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg-from-instance
```

如以下示例响应所示，该组的所需容量为 2，有 2 个运行实例，启动配置同样命名为 `my-asg-from-instance`：

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupARN": "arn",
      "HealthCheckGracePeriod": 0,
      "SuspendedProcesses": [],
      "DesiredCapacity": 2,
      "Tags": [],
      "EnabledMetrics": [],
      "LoadBalancerNames": [],
      "AutoScalingGroupName": "my-asg-from-instance",
      "DefaultCooldown": 300,
      "MinSize": 1,
      "Instances": [
        {
          "InstanceId": "i-6bd79d87",
          "AvailabilityZone": "us-west-2a",
          "HealthStatus": "Healthy",
          "LifecycleState": "InService",
          "LaunchConfigurationName": "my-asg-from-instance"
        },
        {
          "InstanceId": "i-6cd79d80",
          "AvailabilityZone": "us-west-2a",
          "HealthStatus": "Healthy",
          "LifecycleState": "InService",
          "LaunchConfigurationName": "my-asg-from-instance"
        }
      ],
      "MaxSize": 2,
      "VPCZoneIdentifier": "subnet-6bea5f06",
      "TerminationPolicies": [
        "Default"
      ],
      "LaunchConfigurationName": "my-asg-from-instance",
      "CreatedTime": "2014-12-29T16:14:50.397Z",
      "AvailabilityZones": [
        "us-west-2a"
      ],
      "HealthCheckType": "EC2"
    }
  ]
}
```

使用以下 `describe-launch-configs` 命令描述启动配置 `my-asg-from-instance`。

```
aws autoscaling describe-launch-configurations --launch-configuration-names my-asg-from-instance
```

使用 Amazon EC2 启动向导创建 Auto Scaling 组

使用 Amazon EC2 启动向导，您可以在一个过程中创建启动配置和 Auto Scaling 组。如果要启动多个实例，并且希望从已在 Amazon EC2 启动向导中选择的设置创建新的启动配置和 Auto Scaling 组，这非常有用。您不能使用此选项使用现有启动配置创建 Auto Scaling 组。

从启动向导创建启动配置和 Auto Scaling 组

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在控制面板中，选择 Launch Instance。
3. 选择一个 AMI，然后在下一页面上选择一个实例类型，然后选择 Next: Configure Instance Details。
4. 在 Number of instances 中输入要启动的实例数，然后选择 Launch into Auto Scaling Group。无需在此页面上输入任何其他配置详细信息。
5. 在确认页面上，选择 Create Launch Configuration。
6. 您会切换到启动配置向导的步骤 3。已根据您在 Amazon EC2 启动向导中的选择选择了 AMI 和实例类型。为启动配置输入名称，根据需要配置所有其他设置，然后选择 Next: Add Storage。
7. 配置任何其他卷，然后选择 Next: Configure Security Group。
8. 创建一个新安全组或选择一个现有组，然后选择 Review。
9. 查看启动配置的详细信息，然后选择 Create launch configuration 来选择密钥对并创建启动配置。
10. 在 Configure Auto Scaling group details 页面上，已为您选定您创建的启动配置，并在 Group size 中填入了您在 Amazon EC2 启动向导中指定的实例数。为组输入名称，指定 VPC 和子网（如果需要），然后选择 Next: Configure scaling policies。
11. 在 Configure scaling policies 页面上，选择下列选项之一，然后选择 Review：
 - 要根据需要手动调整 Auto Scaling 组的大小，请选择 Keep this group at its initial size。有关更多信息，请参阅 [手动扩展 \(p. 53\)](#)。
 - 要根据您指定的条件自动调整 Auto Scaling 的大小，则选择 Use scaling policies to adjust the capacity of this group，然后按照指示操作。有关详细信息，请参阅 [配置扩展策略 \(p. 71\)](#)。
12. 在 Review 页面上，您可以选择添加标签或通知，并编辑其他配置详细信息。完成后，选择 Create Auto Scaling group。

标记 Auto Scaling 组和实例

通过以标签的形式将自己的元数据分配给每个组，您可以对 Auto Scaling 组进行组织和管理。可为每个标签指定键 和值。键可以是具有特定关联值的一般类别，例如“project”、“owner”或“environment”。例如，要区分测试环境和生产环境，您可以为每个 Auto Scaling 组分配一个标签，其键为“environment”，值为“test”（如果组是测试环境的一部分）或“production”（如果组是生产环境的一部分）。建议您使用一组具有一致性的标签来更轻松地跟踪您的 Auto Scaling 组。

您可以指定 Auto Scaling 还将 Auto Scaling 组的标签添加到它启动的 EC2 实例。当实例处于 Pending 状态时，Auto Scaling 将应用标签。请注意，如果您具有生命周期挂钩，则标签在实例进入 Pending:Wait 状态时可用。

通过标记 EC2 实例，您可以在 AWS 账单中按标签查看实例成本分配。有关更多信息，请参阅 AWS Billing and Cost Management 用户指南 中的 [使用成本分配标签](#)。

内容

- [标签限制 \(p. 40\)](#)
- [标记生命周期 \(p. 40\)](#)
- [对 Auto Scaling 组添加或修改标签 \(p. 40\)](#)
- [删除标签 \(p. 42\)](#)

标签限制

下面是适用于标签的基本限制：

- 每个资源的最大标签数是 50。
- 可以使用单个调用添加或删除的标签的最大数目为 25。
- 最大键长度为 127 个 Unicode 字符。
- 最大值长度为 255 个 Unicode 字符。
- 标签密钥和值要区分大小写。
- 请勿在标签名称或值中使用 `aws:` 前缀，因为它预留给 AWS 使用。您不能编辑或删除具有此前缀的标签名称或值，它们不计入每个 Auto Scaling 组的标签数限制。

创建或更新 Auto Scaling 组时，您可以创建标签并分配给 Auto Scaling 组。您可以随时删除 Auto Scaling 组标签。有关在您创建 Auto Scaling 组时分配标签的信息，请参阅[步骤 2：创建 Auto Scaling 组 \(p. 12\)](#)。

标记生命周期

如果您选择将标签传播到您的 Auto Scaling 实例，标签是通过以下方式进行管理的：

- Auto Scaling 在启动实例时，即为实例添加标签。此外，Auto Scaling 还添加关键字为 `aws:autoscaling:groupName`、值为 Auto Scaling 组名称的标签。
- 您在附加现有实例时，Auto Scaling 就会向这些实例添加标签，覆盖具有相同标签关键字的现有标签。此外，Auto Scaling 还添加关键字为 `aws:autoscaling:groupName`、值为 Auto Scaling 组名称的标签。
- 将实例从 Auto Scaling 组中分离时，Auto Scaling 仅删除 `aws:autoscaling:groupName` 标签。
- 在手动收缩或 Auto Scaling 自动收缩时，Auto Scaling 会删除将要终止的实例的所有标签。

对 Auto Scaling 组添加或修改标签

当您向 Auto Scaling 组添加标签时，可以指定是否应将其添加到在您的 Auto Scaling 组中启动的实例。如果修改标签，在更改后，标签的更新版本将添加到在 Auto Scaling 组中启动的实例。如果创建或修改 Auto Scaling 组的标签，不会对已经在 Auto Scaling 组中运行的实例进行这些更改。

内容

- [使用 AWS 管理控制台添加或修改标签 \(p. 40\)](#)
- [使用 AWS CLI 添加或修改标签 \(p. 41\)](#)

使用 AWS 管理控制台添加或修改标签

使用 Amazon EC2 控制台添加或修改标签。

添加或修改标签

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。

3. 选择您的 Auto Scaling 组。
4. 在 Tags 选项卡上，选择 Add/Edit tags。Add/Edit Auto Scaling Group Tags 页面列出了 Auto Scaling 组的任何现有标签。
5. 要修改现有标签，请编辑 Key 和 Value。
6. 要添加新标签，请选择 Add tag，然后编辑 Key 和 Value。您可以使 Tag New Instances 保持选中状态，以便自动将标签添加到在 Auto Scaling 组启动的实例，否则取消选中它。
7. 添加完标签后，选择 Save。

使用 AWS CLI 添加或修改标签

使用 `create-or-update-tags` 命令创建或修改标签。例如，以下命令添加键为“environment”、值为“test”的标签，在进行此更改后，该标签还将被添加到在 Auto Scaling 组中启动的实例。如果具有此键的标签已经存在，则会替换现有标签。

```
aws autoscaling create-or-update-tags --tags "ResourceId=my-asg,ResourceType=auto-scaling-group,Key=environment,Value=test,PropagateAtLaunch=true"
```

以下为响应示例：

```
OK-Created/Updated tags
```

使用以下 `describe-tags` 命令来列出用于指定 Auto Scaling 组的标签。

```
aws autoscaling describe-tags --filters Name=auto-scaling-group,Values=my-asg
```

以下为响应示例：

```
{
  "Tags": [
    {
      "ResourceType": "auto-scaling-group",
      "ResourceId": "my-asg",
      "PropagateAtLaunch": true,
      "Value": "test",
      "Key": "environment"
    }
  ]
}
```

或者，使用以下 `describe-auto-scaling-groups` 命令来验证是否已将标签添加到了 Auto Scaling 组。

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

以下为响应示例：

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupARN": "arn",
      "HealthCheckGracePeriod": 0,
      "SuspendedProcesses": [],
      "DesiredCapacity": 1,
      "Tags": [
        {
          "ResourceType": "auto-scaling-group",
```



```
        "ResourceId": "my-asg",
        "PropagateAtLaunch": true,
        "Value": "test",
        "Key": "environment"
    }
],
"EnabledMetrics": [],
"LoadBalancerNames": [],
"AutoScalingGroupName": "my-asg",
...
}
]
```

删除标签

您可以随时删除与 Auto Scaling 组关联的标签。

内容

- [使用 AWS 管理控制台删除标签 \(p. 42\)](#)
- [使用 AWS CLI删除标签 \(p. 42\)](#)

使用 AWS 管理控制台删除标签

使用控制台删除标签

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择您的 Auto Scaling 组。
4. 在 Tags 选项卡上，选择 Add/Edit tags。Add/Edit Auto Scaling Group Tags 页面列出了 Auto Scaling 组的任何现有标签。
5. 选择标签旁边的删除图标。
6. 选择 Save。

使用 AWS CLI删除标签

使用 `delete-tags` 命令删除标签。例如，以下命令删除键为“environment”的标签。

```
aws autoscaling delete-tags --tags "ResourceId=my-asg,ResourceType=auto-scaling-group,Key=environment"
```

注意，您必须指定标签键，但无需指定值。如果您指定了一个值，并且该值不正确，则不会删除标签。

将负载均衡器与 Auto Scaling 组结合使用

在使用 Auto Scaling 时，可以在需求增多时自动增大 Auto Scaling 组，并在需求减少时自动减小该组。在 Auto Scaling 添加和删除 EC2 实例时，您必须确保跨所有 EC2 实例分配应用程序的流量。Elastic Load Balancing 服务自动跨数量不断变化的 EC2 实例路由传入 Web 流量。您的负载均衡器将作为到您的 Auto Scaling 组中的实例的所有传入流量的单一接触点。有关更多信息，请参阅 [Elastic Load Balancing 用户指南](#)。

要将负载均衡器与 Auto Scaling 组结合使用，请创建负载均衡器，然后将它附加到该组。

内容

- 将负载均衡器附加到 Auto Scaling 组 (p. 43)
- 结合使用 ELB 运行状况检查和 Auto Scaling (p. 45)
- 将具有扩展和负载均衡功能的应用程序扩展到其他可用区 (p. 45)

将负载均衡器附加到 Auto Scaling 组

Auto Scaling 与 Elastic Load Balancing 集成，以便您可以将一个或多个负载均衡器附加到现有 Auto Scaling 组。附加负载均衡器后，它自动注册组中的实例，并将传入流量分布到这些实例。要对实例使用 Elastic Load Balancing 运行状况检查以确保流量仅路由到正常实例，请参阅[结合使用 ELB 运行状况检查和 Auto Scaling \(p. 45\)](#)。

当您附加负载均衡器时，它进入 Adding 状态，同时注册组中的实例。负载均衡器注册了组中所有实例后，它进入 Added 状态。在至少一个注册实例通过运行状况检查后，它进入 InService 状态。在负载均衡器进入 InService 状态后，Auto Scaling 可以终止和替换报告为运行状况不佳的所有实例。请注意，如果已注册实例均未通过运行状况检查（例如，由于误配置的运行状况检查），负载均衡器无法进入 InService 状态，因此 Auto Scaling 不会终止和替换这些实例。

当分离负载均衡器时，它进入 Removing 状态，同时取消注册组中的实例。请注意，实例在取消注册后仍保持运行。如果启用连接耗尽，则 Elastic Load Balancing 将等待动态请求完成或最大超时到期（以先到者为准），然后再取消注册实例。请注意，连接耗尽对应用程序负载均衡器是始终启用的，但对传统负载均衡器是必须启用的。有关更多信息，请参阅[传统负载均衡器指南](#)中的[连接耗尽](#)。

Elastic Load Balancing 将有关负载均衡器和 EC2 实例的数据发送给 Amazon CloudWatch。CloudWatch 收集资源的性能数据并以指标的方式呈现这些数据。有关更多信息，请参阅[使用 Amazon CloudWatch 监控 Auto Scaling 组和实例 \(p. 97\)](#)。在将负载均衡器附加到 Auto Scaling 组后，可以创建使用 Elastic Load Balancing 指标自动扩展应用程序的扩展策略。有关更多信息，请参阅[创建具有目标跟踪扩展策略的 Auto Scaling 组 \(p. 70\)](#)。

内容

- 先决条件 (p. 43)
- 使用控制台添加负载均衡器 (p. 43)
- 使用 AWS CLI 添加负载均衡器 (p. 44)

先决条件

开始前，在 Auto Scaling 组所在的区域中创建一个负载均衡器。Elastic Load Balancing 支持两类负载均衡器：传统负载均衡器和应用程序负载均衡器。您可以创建任一类型的负载均衡器以附加到 Auto Scaling 组。有关更多信息，请参阅[Elastic Load Balancing 用户指南](#)。

通过传统负载均衡器，实例注册到负载均衡器，通过应用程序负载均衡器，实例作为目标注册到目标组。当您计划将负载均衡器与 Auto Scaling 组配合使用时，无需将 EC2 实例注册到负载均衡器或目标组。在将负载均衡器或目标组附加到 Auto Scaling 组后，Auto Scaling 会在启动实例时将其注册到负载均衡器或目标组。

使用控制台添加负载均衡器

使用以下过程将负载均衡器附加到现有的 Auto Scaling 组。要在创建 Auto Scaling 组时向 Auto Scaling 组附加负载均衡器，请参阅[教程：设置具有扩展和负载均衡功能的应用程序 \(p. 15\)](#)。

将负载均衡器附加到组

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。

3. 选择您的组。
4. 在 Details 选项卡上，选择 Edit。
5. 执行以下任一操作：
 - a. [传统负载均衡器] 对于 Load Balancers，选择您的负载均衡器。
 - b. [应用程序负载均衡器] 对于 Target Groups，选择您的目标组。
6. 选择 Save。

如果不再需要负载均衡器，请使用以下步骤将其与 Auto Scaling 组分离。

将负载均衡器与组分离

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择您的组。
4. 在 Details 选项卡上，选择 Edit。
5. 执行以下任一操作：
 - a. [传统负载均衡器] 对于 Load Balancers，删除负载均衡器。
 - b. [应用程序负载均衡器] 对于 Target Groups，删除目标组。
6. 选择 Save。

使用 AWS CLI 添加负载均衡器

附加 传统负载均衡器

使用以下 [attach-load-balancers](#) 命令将指定的负载均衡器附加到 Auto Scaling 组：

```
aws autoscaling attach-load-balancers --auto-scaling-group-name my-asg --load-balancer-names my-lb
```

附加目标组

使用以下 [attach-load-balancer-target-groups](#) 命令将指定的目标组附加到 Auto Scaling 组：

```
aws autoscaling attach-load-balancer-target-groups --auto-scaling-group-name my-asg --target-group-arns my-targetgroup-arn
```

分离 传统负载均衡器

使用以下 [detach-load-balancers](#) 命令将负载均衡器与 Auto Scaling 组分离（如果不再需要）：

```
aws autoscaling detach-load-balancers --auto-scaling-group-name my-asg --load-balancer-names my-lb
```

分离目标组

使用以下 [detach-load-balancer-target-groups](#) 命令将目标组与 Auto Scaling 组分离（如果不再需要）：

```
aws autoscaling detach-load-balancer-target-groups --auto-scaling-group-name my-asg --target-group-arns my-targetgroup-arn
```

结合使用 ELB 运行状况检查和 Auto Scaling

Auto Scaling 组会定期检查每个实例的运行状况。它可以单独使用 EC2 状态检查，或使用 EC2 状态检查加上 Elastic Load Balancing 运行状况检查。如果 Auto Scaling 确定某实例运行状况不佳，将替换该实例。

如果您将一个 Auto Scaling 组配置为仅使用 EC2 状态检查来确定运行状态（这是默认设置），而实例未能通过 EC2 状态检查，则 Auto Scaling 会认为该实例运行状况不佳。但是，如果您已将一个或多个负载均衡器或目标组附加到 Auto Scaling 组，其中一个负载均衡器报告某实例运行状况不佳，则 Auto Scaling 不会认为该实例运行状况不佳，因此不会替换该实例。

如果您的 Auto Scaling 组配置为同时使用 EC2 状态检查和 Elastic Load Balancing 运行状况检查来确定运行状况，而某实例未能通过状态检查或运行状况检查，则 Auto Scaling 会认为该实例运行状况不佳。请注意，如果您将多个负载均衡器附加到一个 Auto Scaling 组，则只有在所有负载均衡器均报告某实例正常的情况下，Auto Scaling 才会认为该实例正常。只要有一个负载均衡器报告某实例运行状况不佳，Auto Scaling 就会替换该实例，即使其他负载均衡器报告该实例正常也是如此。

有关更多信息，请参阅 [Auto Scaling 实例的运行状况检查](#) (p. 95)。

内容

- [使用控制台添加运行状况检查](#) (p. 45)
- [使用 AWS CLI 添加运行状况检查](#) (p. 45)

使用控制台添加运行状况检查

使用以下过程将宽限期为 300 秒的 ELB 运行状况检查添加到已附加负载均衡器的 Auto Scaling 组。

使用控制台添加运行状况检查

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择您的组。
4. 在 Details 选项卡上，选择 Edit。
5. 对于 Health Check Type，选择 ELB。
6. 对于 Health Check Grace Period，输入 300。
7. 选择 Save。
8. 在 Instances 选项卡上，Health Status 列显示新添加的运行状况检查的结果。

使用 AWS CLI 添加运行状况检查

使用以下 `update-auto-scaling-group` 命令创建宽限期为 300 秒的运行状况检查：

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-lb-asg --health-check-type ELB --health-check-grace-period 300
```

将具有扩展和负载均衡功能的应用程序扩展到其他可用区

通过使 Auto Scaling 组跨区域中的多个可用区，然后附加负载均衡器以跨这些可用区分配传入流量，可以利用地理冗余的安全性和可靠性。在为负载均衡器启用的所有可用区之间均等分配传入流量。

Note

Auto Scaling 组可以包含来自同一区域中的多个可用区的 EC2 实例。不过，Auto Scaling 组无法包含来自多个区域的 EC2 实例。

当一个可用区运行状况不佳或无法使用时，Auto Scaling 在不受影响的可用区中启动新实例。当运行状况不佳的可用区恢复正常状态时，Auto Scaling 自动在 Auto Scaling 组的所有可用区中重新均匀分配应用程序实例。Auto Scaling 通过尝试在实例最少的可用区中启动新实例来执行此操作。但是，如果尝试失败，Auto Scaling 将尝试在其他可用区中启动，直到成功为止。

您可以通过将可用区添加到您的 Auto Scaling 组中，然后为您的负载均衡器启用该可用区，来扩展具有扩展和负载均衡功能的应用程序的可用性。在您启用新的可用区域后，负载均衡器会开始在所有已经启用的可用区域间均匀地路由流量。

内容

- [使用控制台添加可用区 \(p. 46\)](#)
- [使用 AWS CLI 添加可用区 \(p. 46\)](#)

使用控制台添加可用区

使用以下步骤将 Auto Scaling 组扩展到其他子网 (EC2-VPC) 或可用区 (EC2-Classical)。

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择您的组。
4. 在 Details 选项卡上，选择 Edit。
5. 执行以下任一操作：
 - [EC2-VPC] 在 Subnet(s) 中，选择与可用区相应的子网。
 - [EC2-Classical] 在 Availability Zones(s) 中，选择可用区。
6. 选择 Save。
7. 在导航窗格上的 NETWORK & SECURITY 下，选择 Load Balancers。
8. 选择您的负载均衡器。
9. 执行以下任一操作：
 - [传统负载均衡器 in EC2-Classical] 在 Instances 选项卡上，选择 Edit Availability Zones。在 Add and Remove Availability Zones 页面上，选择要添加的可用区。
 - [传统负载均衡器 in a VPC] 在 Instances 选项卡上，选择 Edit Availability Zones。在 Add and Remove Subnets 页面上，对于 Available subnets，为要添加的子网选择加号图标 (+)。该子网将移到 Selected subnets 下。
 - [应用程序负载均衡器] 在 Description 选项卡上，为 Availability Zones 选择 Edit。为要添加的可用区的子网选择加号图标 (+)。该子网将移到 Selected subnets 下。
10. 选择 Save。

使用 AWS CLI 添加可用区

您将使用的命令取决于您的负载均衡器是 VPC 中的 传统负载均衡器、EC2-Classical 中的 传统负载均衡器，还是 应用程序负载均衡器。

对于在 VPC 中具有 传统负载均衡器 的 Auto Scaling 组

1. 使用以下 `update-auto-scaling-group` 命令将子网添加到 Auto Scaling 组：

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg --vpc-zone-identifier subnet-41767929 subnet-cb663da2 --min-size 2
```

2. 使用以下 [describe-auto-scaling-groups](#) 命令验证新子网中的实例是否准备好接受来自负载均衡器的流量：

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

3. 使用以下 [attach-load-balancer-to-subnets](#) 命令为 传统负载均衡器 启用新子网：

```
aws elb attach-load-balancer-to-subnets --load-balancer-name my-lb --subnets subnet-41767929
```

对于在 EC2-Classical 中具有 传统负载均衡器 的 Auto Scaling 组

1. 使用以下 [update-auto-scaling-group](#) 命令将可用区添加到 Auto Scaling 组：

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg --availability-zones us-west-2a us-west-2b us-west-2c --min-size 3
```

2. 使用以下 [describe-auto-scaling-groups](#) 命令验证新可用区中的实例是否准备好接受来自负载均衡器的流量：

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

3. 使用以下 [enable-availability-zones-for-load-balancer](#) 命令为 传统负载均衡器 启用新的可用区：

```
aws elb enable-availability-zones-for-load-balancer --load-balancer-name my-lb --availability-zones us-west-2c
```

对于具有 应用程序负载均衡器 的 Auto Scaling 组

1. 使用以下 [update-auto-scaling-group](#) 命令将子网添加到 Auto Scaling 组：

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg --vpc-zone-identifier subnet-41767929 subnet-cb663da2 --min-size 2
```

2. 使用以下 [describe-auto-scaling-groups](#) 命令验证新子网中的实例是否准备好接受来自负载均衡器的流量：

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

3. 使用以下 [set-subnets](#) 命令为 应用程序负载均衡器 启用新的子网：

```
aws elbv2 set-subnets --load-balancer-arn my-lb-arn --subnets subnet-41767929 subnet-cb663da2
```

将 Auto Scaling 组合并为单个多区组

要将独立的单区 Auto Scaling 组合并为单个跨多可用区的 Auto Scaling 组，请将其中一个单区组重新分区，成为多区组，然后删除其他组。只要新的多区组所在的可用区与原始单区组所在的其中一个可用区相同，无论组有没有负载均衡器，此流程均适用。

以下示例假设您在两个不同可用区 `us-west-2c` 和 `us-west-2a` 中有两个相同的组。这两个组都具有以下规格：

- 最小大小 = 2
- 最大大小 = 5
- 所需容量 = 3

使用 AWS CLI 合并区域

使用以下过程将 `my-group-a` 和 `my-group-c` 合并到一个同时涵盖 `us-west-2a` 和 `us-west-2c` 的组中。

将独立的单区 组合并为单个多区组的步骤

1. 使用以下 `update-auto-scaling-group` 命令将 `us-west-2c` 可用区添加到 `my-group-a` 的支持的可用区中，并增加此组的最大大小以允许来自这两个单区组的实例：

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-group-a --availability-zones "us-west-2a" "us-west-2c" --max-size 10 --min-size 4
```

2. 使用以下 `set-desired-capacity` 命令增加 `my-group-a` 的大小：

```
aws autoscaling set-desired-capacity --auto-scaling-group-name my-group-a --desired-capacity 6
```

3. （可选）使用以下 `describe-auto-scaling-groups` 命令验证 `my-group-a` 是新的尺寸：

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-group-a
```

4. 使用以下 `update-auto-scaling-group` 命令从 `my-group-c` 中移除实例：

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-group-c --min-size 0 --max-size 0
```

5. （可选）使用以下 `describe-auto-scaling-groups` 命令验证 `my-group-c` 中未保留任何实例：

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-group-c
```

下面是示例输出：

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupARN": "arn",
      "HealthCheckGracePeriod": 300,
      "SuspendedProcesses": [],
      "DesiredCapacity": 0,
      "Tags": [],
      "EnabledMetrics": [],
      "LoadBalancerNames": [],
      "AutoScalingGroupName": "my-group-c",
      "DefaultCooldown": 300,
      "MinSize": 0,
      "Instances": [],
      "MaxSize": 0,
      "VPCZoneIdentifier": "null",
      "TerminationPolicies": [
        "Default"
      ]
    }
  ]
}
```

```
    ],  
    "LaunchConfigurationName": "my-lc",  
    "CreatedTime": "2015-02-26T18:24:14.449Z",  
    "AvailabilityZones": [  
        "us-west-2c"  
    ],  
    "HealthCheckType": "EC2"  
  }  
]  
}
```

6. 使用 `delete-auto-scaling-group` 命令删除 `my-group-c` :

```
aws autoscaling delete-auto-scaling-group --auto-scaling-group-name my-group-c
```

删除 Auto Scaling 基础设施

要完全删除您的 Auto Scaling 基础设施，请完成以下任务。

任务

- 删除 Auto Scaling 组 (p. 49)
- (可选) 删除启动配置 (p. 49)
- (可选) 删除负载均衡器 (p. 50)
- (可选) 删除 CloudWatch 警报 (p. 50)

删除 Auto Scaling 组

当您删除 Auto Scaling 组时，其所需值、最小值和最大值设置为 0。因此，Auto Scaling 实例将会终止。或者，您可以在删除 Auto Scaling 组之前终止或分离实例。

使用控制台删除 Auto Scaling 组

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 在 Auto Scaling 组页面上，选择您的 Auto Scaling 组，然后选择 Actions 和 Delete。
4. 当系统提示进行确认时，选择 Yes, Delete。

使用 AWS CLI 删除 Auto Scaling 组

使用以下 `delete-auto-scaling-group` 命令删除 Auto Scaling 组：

```
aws autoscaling delete-auto-scaling-group --auto-scaling-group-name my-asg
```

(可选) 删除启动配置

请注意，如果要保留启动配置以备将来使用，可跳过此步骤。

使用控制台删除启动配置

1. 在导航窗格上的 Auto Scaling 下，选择 Launch Configurations。
2. 在 Launch Configurations 页面上，选择启动配置，然后选择 Actions 和 Delete launch configuration。

3. 当系统提示进行确认时，选择 Yes, Delete。

使用 AWS CLI 删除启动配置

使用以下 `delete-launch-configuration` 命令：

```
aws autoscaling delete-launch-configuration --launch-configuration-name my-lc
```

(可选) 删除负载均衡器

请注意，如果 Auto Scaling 组未向 Elastic Load Balancing 负载均衡器注册，或者您要保留负载均衡器以备将来使用，可跳过此步骤。

删除您的负载均衡器

1. 在导航窗格上的 LOAD BALANCING 下，选择 Load Balancers。
2. 选择负载均衡器，然后选择 Actions 和 Delete。
3. 当系统提示进行确认时，选择 Yes, Delete。

使用 AWS CLI 删除与 Auto Scaling 组关联的负载均衡器

对于 应用程序负载均衡器，使用以下 `delete-load-balancer` 命令：

```
aws elbv2 delete-load-balancer --load-balancer-arn my-load-balancer-arn
```

对于 传统负载均衡器，使用以下 `delete-load-balancer` 命令：

```
aws elb delete-load-balancer --load-balancer-name my-load-balancer
```

(可选) 删除 CloudWatch 警报

请注意，如果 Auto Scaling 组未与任何 CloudWatch 警报关联，或者您要保留该警报以备将来使用，可跳过此步骤。

使用控制台删除 CloudWatch 警报

1. 通过以下网址打开 CloudWatch 控制台：<https://console.amazonaws.cn/cloudwatch/>。
2. 在导航窗格上，选择 Alarms。
3. 选择警报，然后选择 Delete。
4. 当系统提示进行确认时，选择 Yes, Delete。

使用 AWS CLI 删除 CloudWatch 警报

使用 `delete-alarms` 命令。例如，使用以下命令可删除 AddCapacity 和 RemoveCapacity 警报：

```
aws cloudwatch delete-alarms --alarm-name AddCapacity RemoveCapacity
```


扩展 Auto Scaling 组的大小

扩展是增加或减少应用程序的计算容量的能力。扩展从事件或扩展操作开始，事件或扩展操作指示 Auto Scaling 启动或终止 EC2 实例。

Auto Scaling 为满足您的应用程序需求提供了多种调整扩展的方式。因此，您需要很好地了解您的应用程序，这十分重要。请注意以下事项：

- 您希望 Auto Scaling 在您的应用程序架构中发挥什么作用？通常的想法是将 Auto Scaling 作为增加和减少容量的方式，但它在维持稳定的服务器数量方面也十分有用。
- 哪些成本约束对您比较重要？因为 Auto Scaling 使用 EC2 实例，所以您只需为使用的资源付费。了解成本约束可以帮助您确定何时扩展应用程序以及扩展量。
- 哪些指标对应用程序比较重要？CloudWatch 支持一些可用于 Auto Scaling 组的不同指标。我们建议您查看这些指标，以了解其中哪些与您的应用程序最有意义。

内容

- [扩展计划 \(p. 51\)](#)
- [多个扩展策略 \(p. 52\)](#)
- [保持 Auto Scaling 组中的实例数量 \(p. 52\)](#)
- [手动扩展 \(p. 53\)](#)
- [计划的扩展 \(p. 60\)](#)
- [动态扩展 \(p. 63\)](#)
- [Auto Scaling 冷却时间 \(p. 75\)](#)
- [控制 Auto Scaling 在缩小过程中终止哪些实例 \(p. 78\)](#)
- [Auto Scaling 生命周期挂钩 \(p. 83\)](#)
- [临时从 Auto Scaling 组中删除实例 \(p. 88\)](#)
- [暂停和恢复 Auto Scaling 流程 \(p. 92\)](#)

扩展计划

Auto Scaling 提供了多种扩展 Auto Scaling 组的方式。

始终保持当前实例等级

您可以配置 Auto Scaling 组，使其始终保持最小数量或指定的所需数量的运行实例。为了保持当前实例等级，Auto Scaling 将对 Auto Scaling 组内运行的实例执行定期运行状况检查。如果 Auto Scaling 发现运行状况不佳的实例，它将终止该实例，并启动新实例。有关配置 Auto Scaling 组以保持当前实例等级的信息，请参阅[保持 Auto Scaling 组中的实例数量 \(p. 52\)](#)。

手动扩展

手动扩展是扩展资源的最基本方法。只指定 Auto Scaling 组的最大容量、最小容量或所需容量的变化。Auto Scaling 可以管理创建或终止实例的流程来维护更新的容量。有关更多信息，请参阅 [手动扩展 \(p. 53\)](#)。

按计划扩展

有时候，您确切地知道何时要增加或减少组中的实例数量，因为该需求遵循可预测的计划。按计划扩展意味着扩展操作作为时间和日期的函数自动执行。有关更多信息，请参阅 [计划的扩展 \(p. 60\)](#)。

根据需求进行扩展

按策略扩展是一种更高级的资源扩展方式，允许您定义控制 Auto Scaling 流程的参数。例如，您可以创建策略，要求在平均 CPU 使用率连续 15 分钟保持在 90% 以上时扩展 EC2 实例。如果您知道如何根据条件变化来进行扩展，但却不知道条件何时改变，那么可使用这种方法。您可以通过设置 Auto Scaling 来进行响应。

您应该为每个要监控的事件设置两个策略，一个用于收缩 (终止实例)，另一个用于扩展 (启动实例)。例如，如果需要在网络带宽达到特定等级时进行扩展，需要创建策略以指定 Auto Scaling 应启动特定数量的实例来帮助您应对流量。但是可能还需要一个相应的策略，在网络带宽水平降低时缩小特定数量的实例。有关更多信息，请参阅 [动态扩展 \(p. 63\)](#)。

多个扩展策略

一个 Auto Scaling 组在任何给定时间可附有多个扩展策略。实际上，我们建议每个 Auto Scaling 组至少使用两个策略：一个用于扩大架构，一个用于缩小架构。您还可以合并扩展策略以发挥 Auto Scaling 组的最大性能。

为了说明多个策略如何协同工作，请考虑使用一个 Auto Scaling 组和一个 Amazon SQS 队列的应用程序，该队列用于向组中的 EC2 实例发送请求。为了帮助确保应用程序性能达到最佳级别，有两个策略用于控制何时扩展 Auto Scaling 组。一个策略使用 Amazon CloudWatch 指标 `CPUUtilization` 检测实例何时达到 90% 的容量。另一个策略使用 `NumberOfMessagesVisible` 检测 SQS 队列何时填满消息。

Note

在生产环境中，这两个策略都有补充策略用于控制 Auto Scaling 何时应缩小 EC2 实例数。

将多个策略附加到 Auto Scaling 组时，两个策略有可能同时指示 Auto Scaling 进行扩大 (或缩小)。在前例中，有可能 EC2 实例触发 `CPUUtilization` 指标的 CloudWatch 警报，而 SQS 队列触发 `NumberOfMessagesVisible` 指标的警报。

如果发生这些情况，Auto Scaling 选择对 Auto Scaling 组影响最大的策略。例如，假设针对 CPU 使用率的策略指示 Auto Scaling 启动一个实例，而针对 SQS 队列的策略提示 Auto Scaling 启动两个实例。如果同时满足两个策略的扩展条件，则 Auto Scaling 优先选择 SQS 队列策略，因为它对 Auto Scaling 组影响最大。这样，Auto Scaling 将两个实例启动到该组中。即使策略对扩展使用不同的条件，此优先顺序也适用。例如，如果一个策略指示 Auto Scaling 启动三个实例，而另一个策略指示 Auto Scaling 将容量增加 25%，则 Auto Scaling 优先选择当时对组影响最大的策略。

保持 Auto Scaling 组中的实例数量

创建启动配置和 Auto Scaling 组后，Auto Scaling 组首先启动最小数量 (或指定的所需容量) 的 EC2 实例。如果 Auto Scaling 组没有附加其他扩展条件，Auto Scaling 组始终保持这一最小数量的运行实例。

为了保持相同数量的实例，Auto Scaling 对 Auto Scaling 组内运行的实例执行定期运行状况检查。如果发现实例运行状况不佳，它将终止该实例，并启动新实例。

Auto Scaling 组中所有实例在开始启动时都处于良好运行状态。除非 Auto Scaling 收到实例运行状况不佳的通知，否则将认为它们运行状况良好。此通知可能来自以下一个或多个来源：Amazon EC2、Elastic Load Balancing 或自定义运行状况检查。

确定实例运行状况

默认情况下，Auto Scaling 组通过定期检查 EC2 实例状态检查结果来确定每个实例的运行状况。如果实例状态是除 `running` 以外的其他任何状态，或者系统状态为 `impaired`，则 Auto Scaling 认为实例运行状况不佳并启动替换实例。有关 EC2 实例状态检查的更多信息，请参阅 Amazon EC2 用户指南（适用于 Linux 实例）中的[监控实例状态](#)。

如果您已将 Auto Scaling 组与负载均衡器或目标组相关联，并已选择使用 ELB 运行状况检查，Auto Scaling 将通过查看实例状态检查和 ELB 运行状况检查的结果来确定实例的运行状况。如果实例处于除 `running` 之外的其他状态、系统状态为 `impaired`，或者 Elastic Load Balancing 报告实例未能通过运行状况检查，则 Auto Scaling 将实例标记为运行状况不佳。

您可以通过指定额外检查来自定义 Auto Scaling 组进行的运行状况检查。或者，如果您有自己的运行状况检查系统，可以直接将实例的运行状况信息从系统发送到 Auto Scaling。

替换运行状况不佳的实例

在 Amazon EC2 或 Elastic Load Balancing 运行状况检查中将实例标记为运行状况不佳之后，将立即安排对它进行替换。它不会自动恢复良好的运行状况。您可以通过调用 `SetInstanceHealth` 操作（或 `as-set-instance-health` 命令）进行手动干预，将该实例重新设置为良好的运行状况。但是如果该实例已终止，将导致错误。因为将实例标记为运行状况不佳和实际终止该实例之间的间隔非常短，尝试使用 `SetInstanceHealth` 操作（或 `as-set-instance-health` 命令）将该实例重新设置为良好运行状况可能仅适用于已暂停的组。有关更多信息，请参阅[暂停和恢复 Auto Scaling 流程](#)（p. 92）。

Auto Scaling 将创建用于终止运行状况不佳的实例的新扩展活动，然后终止该实例。随后，另一个扩展活动将启动新实例以替代终止的实例。

实例终止后，任何关联的弹性 IP 地址都会取消关联，并且不会自动与新实例关联。必须手动将这些弹性 IP 地址与新实例关联。同样，实例终止后，其附加的 EBS 卷也会分离。必须手动将这些 EBS 卷附加到新实例。

手动扩展

您可以随时更改现有 Auto Scaling 组的大小。更新 Auto Scaling 组的所需容量或更新附加到 Auto Scaling 组的实例。

内容

- [使用控制台更改 Auto Scaling 组的大小](#)（p. 53）
- [使用 AWS CLI 更改 Auto Scaling 组的大小](#)（p. 54）
- [将 EC2 实例附加到 Auto Scaling 组](#)（p. 55）
- [从 Auto Scaling 组中分离 EC2 实例](#)（p. 58）

使用控制台更改 Auto Scaling 组的大小

在更改 Auto Scaling 组的大小时，Auto Scaling 将管理启动或终止实例的过程以维护新的组大小。

以下示例假设您创建了一个 Auto Scaling 组，该组的最小大小为 1，最大大小为 5。因此，该组目前正在运行的实例是 1 个。

更改 Auto Scaling 组的大小

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。

2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择您的 Auto Scaling 组。
4. 在 Details 选项卡上，选择 Edit。
5. 对于 Desired，将所需容量增加一。例如，如果当前值是 1，请键入 2。

理想容量必须小于或等于组的最大容量。如果您的 Desired 新值大于 Max，则必须更新 Max。

完成后，选择 Save。

接下来，验证 Auto Scaling 组是否已另外启动一个实例。

验证 Auto Scaling 组的容量是否已更改

1. 在 Activity History 选项卡上，Status 列显示实例的当前状态。您可以使用刷新按钮，直至实例的状态更改为 Successful，这表示您的 Auto Scaling 组已成功启动新实例。
2. 在 Instances 选项卡上，Lifecycle 列显示实例的状态。启动实例只需很短的时间。实例启动后，其状态将更改为 InService。您可以看到 Auto Scaling 组启动了 1 个新实例，并且实例处于 InService 状态。

使用 AWS CLI 更改 Auto Scaling 组的大小

在更改 Auto Scaling 组的大小时，Auto Scaling 将管理启动或终止实例的过程以维护新的组大小。

以下示例假设您创建了一个 Auto Scaling 组，该组的最小大小为 1，最大大小为 5。因此，该组目前正在运行的实例是 1 个。

使用 `set-desired-capacity` 命令更改 Auto Scaling 组的大小，如下例所示：

```
aws autoscaling set-desired-capacity --auto-scaling-group-name my-asg --desired-capacity 2
```

默认情况下，此命令不等待为该组指定的冷却时间完成。您可以通过指定 `--honor-cooldown` 选项来覆盖默认行为并等待冷却时间完成，如下示例所示。有关更多信息，请参阅 [Auto Scaling 冷却时间 \(p. 75\)](#)。

```
aws autoscaling set-desired-capacity --auto-scaling-group-name my-asg --desired-capacity 2 --honor-cooldown
```

使用 `describe-auto-scaling-groups` 命令确认已更改 Auto Scaling 组的大小，如下例所示：

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

作为响应，Auto Scaling 提供关于组和已启动实例的详细信息。该响应类似于以下示例：

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupARN": "arn",
      "HealthCheckGracePeriod": 300,
      "SuspendedProcesses": [],
      "DesiredCapacity": 2,
      "Tags": [],
      "EnabledMetrics": [],
      "LoadBalancerNames": [],
      "AutoScalingGroupName": "my-asg",
      "DefaultCooldown": 300,
      "MinSize": 1,
      "Instances": [
```

```
{
  {
    "InstanceId": "i-33388a3f",
    "AvailabilityZone": "us-west-2a",
    "HealthStatus": "Healthy",
    "LifecycleState": "InService",
    "LaunchConfigurationName": "my-lc"
  },
  "MaxSize": 5,
  "VPCZoneIdentifier": "subnet-e4f33493",
  "TerminationPolicies": [
    "Default"
  ],
  "LaunchConfigurationName": "my-lc",
  "CreatedTime": "2014-12-12T23:30:42.611Z",
  "AvailabilityZones": [
    "us-west-2a"
  ],
  "HealthCheckType": "EC2"
}
]
```

注意 DesiredCapacity 显示了新值。Auto Scaling 组已启动实例。

将 EC2 实例附加到 Auto Scaling 组

Auto Scaling 为您提供了一个选项，可通过将一个或多个 EC2 实例附加到现有 Auto Scaling 组来为这些实例启用 Auto Scaling。附加实例后，它们将成为 Auto Scaling 组的一部分。

要挂载的实例必须满足以下条件：

- 实例处于 running 状态。
- 用于启动实例的 AMI 必须仍然存在。
- 实例不是其他 Auto Scaling 组的成员。
- 实例与 Auto Scaling 组位于同一可用区。
- 如果 Auto Scaling 组具有附加的传统负载均衡器，则实例和该负载均衡器必须都位于 EC2- 或同一 VPC 中。如果 Auto Scaling 组具有附加的目标组，则实例和 应用程序负载均衡器 必须都位于同一 VPC 中。

当您附加实例时，Auto Scaling 会将该组的所需容量与要附加的实例数相加。如果要附加的实例数加上所需容量超过组的最大大小，请求将失败。

如果您将实例附加到已具有附加负载均衡器的 Auto Scaling 组，则实例将注册到该负载均衡器。如果您将实例附加到已具有附加目标组的 Auto Scaling 组，则实例将注册到该目标组。

内容

- [使用 AWS 管理控制台挂载实例 \(p. 56\)](#)
- [使用 AWS CLI 挂载实例 \(p. 56\)](#)

这些示例使用的 Auto Scaling 组具有以下配置：

- Auto Scaling 组名称 = **my-asg**
- 最小大小 = **1**
- 最大大小 = **5**
- 所需容量 = **2**
- 可用区 = **us-west-2a**

使用 AWS 管理控制台挂载实例

您可以将现有实例挂载到现有 Auto Scaling 组或在创建新 Auto Scaling 组时挂载到新建的组。

使用控制台将实例挂载到新 Auto Scaling 组

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格中，选择 Instances。
3. 选择实例。
4. 依次选择 Actions、Instance Settings 和 Attach to Auto Scaling Group。
5. 在 Attach to Auto Scaling Group 页面上，选择 a new Auto Scaling group，键入组名，然后选择 Attach。

将使用新的启动配置和为 Auto Scaling 组指定的相同名称创建新的 Auto Scaling 组。启动配置从所挂载的实例获取其设置（例如，安全组和 IAM 角色）。Auto Scaling 组从所挂载的实例获取设置（例如，可用区和子网），并且有所需的容量和最大大小 1。

6. （可选）要编辑 Auto Scaling 组的设置，请在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。选择新 Auto Scaling 组，再选择 Edit，根据需要更改设置，然后选择 Save。

使用控制台将实例挂载到现有 Auto Scaling 组

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. （可选）在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。选择 Auto Scaling 组并验证 Auto Scaling 组的最大大小足够大，可以再添加一个实例。否则，请选择 Edit，增加最大大小，然后选择 Save。
3. 在导航窗格中，选择 Instances。
4. 选择实例。
5. 依次选择 Actions、Instance Settings 和 Attach to Auto Scaling Group。
6. 在 Attach to Auto Scaling Group 页面上，选择 an existing Auto Scaling group，选择实例，然后选择 Attach。
7. 如果实例不符合条件（例如，与 Auto Scaling 组不在同一可用区中），则系统会提供带详细信息的错误消息。选择 Close 并使用符合条件的实例重试。

使用 AWS CLI 挂载实例

使用 AWS CLI 将实例挂载到 Auto Scaling 组

1. 使用下面的 `describe-auto-scaling-groups` 命令描述一个特定的 Auto Scaling 组：

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-names my-asg
```

下面的示例响应显示所需容量为 2 并且组有两个正在运行的实例：

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupARN": "arn",
      "HealthCheckGracePeriod": 300,
      "SuspendedProcesses": [],
      "DesiredCapacity": 2,
      "Tags": [],
      "EnabledMetrics": [],
      "LoadBalancerNames": [],
```



```
"AutoScalingGroupName": "my-asg",
"DefaultCooldown": 300,
"MinSize": 1,
"Instances": [
  {
    "InstanceId": "i-a5e87793",
    "AvailabilityZone": "us-west-2a",
    "HealthStatus": "Healthy",
    "LifecycleState": "InService",
    "LaunchConfigurationName": "my-lc"
  },
  {
    "InstanceId": "i-a4e87792",
    "AvailabilityZone": "us-west-2a",
    "HealthStatus": "Healthy",
    "LifecycleState": "InService",
    "LaunchConfigurationName": "my-lc"
  }
],
"MaxSize": 5,
"VPCZoneIdentifier": "subnet-e4f33493",
"TerminationPolicies": [
  "Default"
],
"LaunchConfigurationName": "my-lc",
"CreatedTime": "2014-12-12T23:30:42.611Z",
"AvailabilityZones": [
  "us-west-2a"
],
"HealthCheckType": "EC2"
}
]
```

2. 使用下面的 `attach-instances` 命令将一个实例挂载到 Auto Scaling 组：

```
aws autoscaling attach-instances --instance-ids i-a8e09d9c --auto-scaling-group-name my-asg
```

3. 要验证实例是否已挂载，请使用下面的 `describe-auto-scaling-groups` 命令：

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-names my-asg
```

下面的示例响应显示所需容量已从 1 增加到 3，并且有一个新实例 `i-a8e09d9c`：

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupARN": "arn",
      "HealthCheckGracePeriod": 300,
      "SuspendedProcesses": [],
      "DesiredCapacity": 3,
      "Tags": [],
      "EnabledMetrics": [],
      "LoadBalancerNames": [],
      "AutoScalingGroupName": "my-asg",
      "DefaultCooldown": 300,
      "MinSize": 1,
      "Instances": [
        {
          "InstanceId": "i-a8e09d9c",
          "AvailabilityZone": "us-west-2a",
          "HealthStatus": "Healthy",
```

```
        "LifecycleState": "InService",
        "LaunchConfigurationName": "my-lc"
    },
    {
        "InstanceId": "i-a5e87793",
        "AvailabilityZone": "us-west-2a",
        "HealthStatus": "Healthy",
        "LifecycleState": "InService",
        "LaunchConfigurationName": "my-lc"
    },
    {
        "InstanceId": "i-a4e87792",
        "AvailabilityZone": "us-west-2a",
        "HealthStatus": "Healthy",
        "LifecycleState": "InService",
        "LaunchConfigurationName": "my-lc"
    }
],
"MaxSize": 5,
"VPCZoneIdentifier": "subnet-e4f33493",
"TerminationPolicies": [
    "Default"
],
"LaunchConfigurationName": "my-lc",
"CreateTime": "2014-12-12T23:30:42.611Z",
"AvailabilityZones": [
    "us-west-2a"
],
"HealthCheckType": "EC2"
}
]
```

从 Auto Scaling 组中分离 EC2 实例

可以从 Auto Scaling 组中移除实例。分离实例之后，可以独立于 Auto Scaling 组的其余部分管理这些实例。通过分离实例，您可以：

- 将实例从一个 Auto Scaling 组移出，然后将其附加到其他组。有关更多信息，请参阅 [将 EC2 实例附加到 Auto Scaling 组 \(p. 55\)](#)。
- 通过使用正在运行应用程序的现有实例创建 Auto Scaling 组，以测试该组，然后在测试完成后将这些实例与 Auto Scaling 组分离。

分离实例时，您可以选择按分离的实例数减少 Auto Scaling 组的所需容量。如果您选择不减少容量，Auto Scaling 会启动新实例以替换分离的实例。如果减少容量，但将多个实例从同一可用区分离，则除非暂停 AZRebalance 流程，否则 Auto Scaling 可以重新平衡可用区。有关更多信息，请参阅 [Auto Scaling 流程 \(p. 92\)](#)。

如果将实例从具有附加负载均衡器的 Auto Scaling 组分离，则该实例将从负载均衡器取消注册。如果您从具有附加目标组的 Auto Scaling 组中分离实例，则实例将从目标组取消注册。如果为负载均衡器启用了连接耗尽，则 Auto Scaling 会等待未送达请求完成。

内容

- [使用 AWS 管理控制台分离实例 \(p. 59\)](#)
- [使用 AWS CLI 分离实例 \(p. 59\)](#)

这些示例使用的 Auto Scaling 组具有以下配置：

- Auto Scaling 组名称 = my-asg
- 最小容量 = 1
- 最大容量 = 5
- 所需容量 = 4
- 可用区域 = us-west-2a

使用 AWS 管理控制台分离实例

使用以下步骤将实例与 Auto Scaling 组分离。

使用控制台将实例与现有 Auto Scaling 组分离

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择您的 Auto Scaling 组。
4. 在 Instances 选项卡上，选择实例，然后选择 Actions 和 Detach。
5. 在 Detach Instance 页面上，选中该复选框可让 Auto Scaling 启动替换实例，取消选中该复选框可减少所需容量。选择 Detach Instance。

使用 AWS CLI 分离实例

使用以下步骤将实例与 Auto Scaling 组分离。

使用 AWS CLI 将实例与现有 Auto Scaling 组分离

1. 使用以下 `describe-auto-scaling-instances` 命令列出当前实例：

```
aws autoscaling describe-auto-scaling-instances
```

以下示例响应显示该组有 4 个运行的实例：

```
{
  "AutoScalingInstances": [
    {
      "AvailabilityZone": "us-west-2a",
      "InstanceId": "i-2a2d8978",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService",
      "LaunchConfigurationName": "my-lc"
    },
    {
      "AvailabilityZone": "us-west-2a",
      "InstanceId": "i-5f2e8a0d",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService",
      "LaunchConfigurationName": "my-lc"
    },
    {
      "AvailabilityZone": "us-west-2a",
      "InstanceId": "i-a52387f7",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService",
      "LaunchConfigurationName": "my-lc"
    }
  ]
}
```

```
    }
    {
      "AvailabilityZone": "us-west-2a",
      "InstanceId": "i-f42d89a6",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService",
      "LaunchConfigurationName": "my-lc"
    }
  ]
}
```

2. 使用以下 `detach-instances` 命令分离实例并减小所需容量：

```
aws autoscaling detach-instances --instance-ids i-2a2d8978 --auto-scaling-group-name my-asg --should-decrement-desired-capacity
```

3. 使用以下 `describe-auto-scaling-instances` 命令验证是否已分离该实例：

```
aws autoscaling describe-auto-scaling-instances
```

以下示例响应显示现在有 3 个运行的实例：

```
{
  "AutoScalingInstances": [
    {
      "AvailabilityZone": "us-west-2a",
      "InstanceId": "i-5f2e8a0d",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService",
      "LaunchConfigurationName": "my-lc"
    },
    {
      "AvailabilityZone": "us-west-2a",
      "InstanceId": "i-a52387f7",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService",
      "LaunchConfigurationName": "my-lc"
    },
    {
      "AvailabilityZone": "us-west-2a",
      "InstanceId": "i-f42d89a6",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService",
      "LaunchConfigurationName": "my-lc"
    }
  ]
}
```

计划的扩展

按时间表定期扩展，使您得以按照可预测的负载变化来扩展应用程序。例如，您的 Web 应用程序的流量会在每周的星期三开始增加，并在星期四保持高流量状态，然后在星期五开始下降。您可以根据 Web 应用程序的可预测流量模式来计划扩展活动。

要配置您的 Auto Scaling 组来根据计划进行扩展，您需要创建计划操作，用以告知 Auto Scaling 在指定的时间执行扩展操作。要创建计划的扩展操作，请指定希望扩展操作生效的开始时间，以及用于扩展操作的新的

最小大小、最大大小和所需大小。在指定的时间，Auto Scaling 将用扩展操作指定的最小大小、最大大小和所需大小值来更新组。

您可以创建仅用于一次扩展的预先计划操作，或者创建用于按经常性计划进行扩展的预先计划操作。

内容

- [计划的操作的注意事项 \(p. 61\)](#)
- [使用控制台创建计划的操作 \(p. 61\)](#)
- [更新计划的操作 \(p. 61\)](#)
- [使用 AWS CLI 创建或更新计划的操作 \(p. 62\)](#)
- [删除计划的操作 \(p. 62\)](#)

计划的操作的注意事项

创建计划的操作时，请记住以下内容。

- Auto Scaling 可以保证同组内计划操作的执行顺序正确，但不保证跨组的计划操作的执行顺序正确。
- 计划操作的执行时间一般为几秒钟。然而，该操作可能会比计划的开始时间延迟最多两分钟。因为 Auto Scaling 将按照指定操作的顺序来执行 Auto Scaling 组内的操作，所以计划开始时间彼此接近的计划操作可能需要更长时间才能执行。
- 您最多可以为每个 Auto Scaling 组创建 125 个计划的操作。
- 计划的操作必须具有唯一时间值。如果您尝试计划在某个时间进行活动，但另一个扩展活动已计划占用该时间，系统将拒绝该请求，并向您发送关于该冲突的出错消息。
- 不支持冷却时间。

使用控制台创建计划的操作

完成以下过程可创建计划的操作以扩展您的 Auto Scaling 组。

创建计划的操作

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择您的 Auto Scaling 组。
4. 在 Scheduled Actions 选项卡上，选择 Create Scheduled Action。
5. 在 Create Scheduled Action 页面上，执行以下操作：
 - 使用 Min、Max 和 Desired Capacity 中的至少一项来指定组的大小。
 - 为 Recurrence 选择一个选项。如果您选择 Once，则 Auto Scaling 将在指定的时间执行操作。如果您选择 Cron，请键入一个 Cron 表达式，此表达式指定 Auto Scaling 执行操作的时间（采用 UTC 时间）。如果您选择以 Every 开头的选项，则将为您创建 Cron 表达式。
 - 如果您为 Recurrence 选择了 Once，则在 Start Time 中指定操作的时间。
 - 如果您指定了重复计划，则可以为 Start Time 和 End Time 指定值。如果您指定开始时间，Auto Scaling 会在该时间执行操作，然后根据重复计划执行操作。如果您指定结束时间，Auto Scaling 在该时间后不执行操作。
6. 选择 Create。

更新计划的操作

如果您的需求发生变化，则可更新计划的操作。

更新计划的操作

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择您的 Auto Scaling 组。
4. 在 Scheduled Actions 选项卡上，选择计划的操作。
5. 选择 Actions 和 Edit。
6. 在 Edit Scheduled Action 页面上，执行以下操作：
 - 使用 Min、Max 或 Desired Capacity 根据需要更新组的大小。
 - 根据需要更新指定的重复。
 - 根据需要更新开始时间和结束时间。
 - 选择 Save。

使用 AWS CLI 创建或更新计划的操作

您可以创建仅用于一次扩展的计划，或者按重复的时间表进行扩展。

仅计划一次扩展的步骤

要在特定时间（使用 UTC 的“YYYY-MM-DDThh:mm:ssZ”格式表示）增加您的 Auto Scaling 组中正在运行的实例数量，请使用以下 `put-scheduled-update-group-action` 命令：

```
aws autoscaling put-scheduled-update-group-action --scheduled-action-name ScaleUp --auto-scaling-group-name my-asg --start-time "2013-05-12T08:00:00Z" --desired-capacity 3
```

要在特定时间（使用 UTC 的“YYYY-MM-DDThh:mm:ssZ”格式表示）减少您的 Auto Scaling 组中正在运行的实例数量，请使用以下 `put-scheduled-update-group-action` 命令：

```
aws autoscaling put-scheduled-update-group-action --scheduled-action-name ScaleDown --auto-scaling-group-name my-asg --start-time "2013-05-13T08:00:00Z" --desired-capacity 1
```

按照重复的时间表来计划扩展

您可以使用 Cron 格式以 UTC 时间指定重复时间表。有关更多信息，请参阅 [Cron Wikipedia](#) 条目。

使用以下 `put-scheduled-update-group-action` 命令创建一个计划的操作，该计划的操作在每年的 1 月、6 月和 12 月的第一天的 00:30 运行。

```
aws autoscaling put-scheduled-update-group-action --scheduled-action-name scaleup-schedule-year --auto-scaling-group-name my-asg --recurrence "30 0 1 1,6,12 0" --desired-capacity 3
```

删除计划的操作

完成计划的操作后，可以将其删除。

使用控制台删除计划的操作

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。

3. 选择您的 Auto Scaling 组。
4. 在 Scheduled Actions 选项卡上，选择计划的操作。
5. 依次选择 Actions 和 Delete。
6. 当系统提示进行确认时，选择 Yes, Delete。

使用 AWS CLI 删除计划的操作

使用以下 `delete-scheduled-action` 命令：

```
aws autoscaling delete-scheduled-action --scheduled-action-name ScaleUp
```

动态扩展

当您使用 Auto Scaling 动态扩展时，必须定义如何按照不断变化的需求进行扩展。例如，假设有一个 Web 应用程序当前在两个实例上运行，您不希望该 Auto Scaling 组的 CPU 使用率超过 70%。您可以将 Auto Scaling 组配置为自动扩展来满足这一需求。策略类型决定扩展操作如何执行。

内容

- [扩展策略类型 \(p. 63\)](#)
- [简单扩展和步进扩展 \(p. 63\)](#)
- [目标跟踪扩展 \(p. 69\)](#)
- [向现有的 Auto Scaling 组添加扩展策略 \(p. 72\)](#)
- [基于 Amazon SQS 进行扩展 \(p. 73\)](#)

扩展策略类型

- Simple scaling—根据一个扩展调整值增加或减小当前组容量。
- Step scaling—根据一组扩展调整值增加或减小当前组容量，这组扩展称为步进调整，根据警报违反情况变化。
- Target tracking scaling - 根据特定指标的目标值增加或减少组的当前容量。它的工作方式类似于您家中负责控制温度的恒温器 - 您只需选择一个温度，恒温器会替您完成所有其他工作。

简单扩展和步进扩展

Auto Scaling 起初只支持简单扩展策略。如果您在引入目标跟踪和步进策略类型前创建了自己的扩展策略，您的策略将被视为简单扩展策略。

即使只有一个步进调整，我们也建议您使用步进扩展策略，因为我们会持续评估警报，在扩展活动或健康检查替换期间不会锁定组。

简单扩展策略

扩展活动启动后，该策略必须等待扩展活动或健康检查替换完成并且冷却时间到期，然后才能响应其他警报。冷却时间有助于防止 Auto Scaling 在先前活动产生明显影响前启动其他扩展活动。您可以使用与 Auto Scaling 组关联的默认冷却时间，也可以通过为策略指定冷却时间覆盖默认值。有关更多信息，请参阅 [Auto Scaling 冷却时间 \(p. 75\)](#)。

步进扩展策略

扩展活动启动后，该策略继续响应其他警报，甚至在进行扩展活动或健康检查替换时也是如此。因此，Auto Scaling 会在接收警报消息时评估任何突破的警报。如果要创建扩展策略，可以指定影响聚合指标的新启动实例就绪所需的估计预热时间。有关更多信息，请参阅 [实例预热 \(p. 65\)](#)。

Note

步进扩展策略不支持冷却时间。因此，对此类策略不能指定冷却时间，组的默认冷却时间也不适用。

扩展调整类型

执行步进扩展或简单扩展策略时，它将使用策略中指定的扩展调整来更改 Auto Scaling 组的当前容量。扩展调整不能将组容量更改为超出最大组大小或低于最小组大小。

对于步进扩展和简单扩展，Auto Scaling 支持以下调整类型：

- **ChangeInCapacity**—将当前组容量增加或减小指定的实例数。正值将增加容量，负调整值将减小容量。

示例：如果当前组容量为 3 个实例，调整值为 5，则当此策略执行时，Auto Scaling 会向组添加 5 个实例，总数为 8 个实例。

- **ExactCapacity**—将当前组容量更改为指定的实例数量。为该调整类型指定一个正值。

示例：如果当前组容量为 3 个实例，调整值为 5，则当此策略执行时，Auto Scaling 会将容量更改为 5 个实例。

- **PercentChangeInCapacity**—将当前组容量增加或减小指定的百分比。正值将增加容量，负值将减小容量。如果得到的值不是整数，Auto Scaling 会进行舍入，如下所示：
 - 大于 1 的值向下取整。例如，12.7 取整为 12。
 - 0 和 1 之间的值取整为 1。例如，.67 取整为 1。
 - 0 和 -1 之间的值取整为 -1。例如，-.58 取整为 -1。
 - 小于 -1 的值向上取整。例如，-6.67 取整为 -6。

示例：如果当前容量为 10 个实例，调整值为 10%，则当此策略执行时，Auto Scaling 会向组添加 1 个实例，总数为 11 个实例。

借助 **PercentChangeInCapacity**，您还可以指定要扩展的最少实例数量 (使用 **MinAdjustmentMagnitude** 参数或使用控制台中的 **Add instances in increments of at least**)。例如，假定您创建一个增加 25% 的策略，并且您指定最小增量为 2 个实例。如果您有一个 Auto Scaling 组包含 4 个实例，而要执行该扩展策略，则 4 的 25% 就是 1 个实例。但是，因为您指定了最小增量 2，则 Auto Scaling 将添加 2 个实例。

步进调整

创建步进扩展策略时，将添加一个或多个步进调整，让您可以根据警报违例的严重程度进行扩展。每个步进调整都根据扩展调整类型指定一个指标值下限、一个指标值上限和扩展量。

策略步进调整有一些规则：

- 步进调整范围不能重叠或有间隙。
- 只有一个步进调整可以有空下限 (负无穷)。如果一个步进调整有负下限，则必须有一个步进调整有空下限。
- 只有一个步进调整可以有空上限 (正无穷)。如果一个步进调整有正上限，则必须有一个步进调整有空上限。
- 同一步进调整中的上限和下限不能为空。
- 如果指标值高于违例阈值，则含下限而不含上限。如果指标值低于违例阈值，则不含下限而含上限。

Auto Scaling 将该聚合类型应用于所有实例的指标数据点。它将聚合指标值与步进调整定义的上限和下限进行比较，以确定执行哪个步进调整。

如果您使用 AWS 管理控制台，则以绝对值的形式指定上限和下限。如果您使用 API 或 CLI，则以违例阈值相对值的形式指定上限和下限。例如，假设您有一个警报的违例阈值为 50，扩展调整类型为 `PercentChangeInCapacity`。还有具有以下步进调整的扩展和收缩策略：

扩大策略			
下限	上限	调整	指标值
0	10	0	50 <= 值 < 60
10	20	10	60 <= 值 < 70
20	null	30	70 <= 值 < +无穷
缩小策略			
下限	上限	调整	指标值
-10	0	0	40 < 值 <= 50
-20	-10	-10	30 < 值 <= 40
null	-20	-30	-无穷 < 值 <= 30

您的组的当前容量和所需容量均为 10 个实例。当聚合指标值大于 40 小于 60 时，该组维持其当前容量和所需容量。

如果指标值达到 60，根据扩展策略的第二个步进调整（增加 10 个实例的 10%），Auto Scaling 会将组的所需容量增加 1 个实例，成为 11 个实例。在新实例运行并且其指定预热时间过期后，Auto Scaling 会将组的当前容量增加到 11 个实例。如果在容量增加后指标值又增加到 70，根据扩展策略的第三个步进调整（增加 11 个实例的 30%，即 3.3 个实例，向下取整为 3 个实例），Auto Scaling 会将组的所需容量增加另外 3 个实例，成为 14 个实例。

如果指标值降到 40，根据收缩策略的第二个步进调整（删除 14 个实例的 10%，向下取整为 1 个实例），Auto Scaling 会将组的所需容量减小 1 个实例，成为 13 个实例。如果在容量减小后指标值又降到 30，根据收缩策略的第三个步进调整（删除 13 个实例的 30%，即 3.9 个实例，向下取整为 3 个实例），Auto Scaling 会将组的所需容量减少另外 3 个实例，成为 10 个实例。

实例预热

通过步进扩展策略可以指定新启动实例的预热时间（秒数）。在指定预热时间过期前，实例不会计入 Auto Scaling 组的聚合指标。

在扩展时，Auto Scaling 不会将正在预热的实例看作当前组容量的一部分。因此，落入同一步进调整中的多个警报违例只会导致一个扩展活动。这可确保添加的实例不会超出您的需要。使用上一部分中的示例，假设指标值增加到 60，而在新实例仍在预热时又增加到 62。当前容量仍为 10 个实例，所以 Auto Scaling 应添加 1 个实例（10 个实例的 10%），但该组的所需容量已经是 11 个实例了，因此 Auto Scaling 不会进一步增加所需容量。但是，如果指标在新实例仍在预热时增长到 70，Auto Scaling 应添加 3 个实例（10 个实例的 30%），但该组的所需容量已经是 11 了，所以 Auto Scaling 只添加 2 个实例，因为新的所需容量为 13 个实例。

在收缩时，Auto Scaling 将正在终止的实例视为当前组容量的一部分。因此不会从 Auto Scaling 组中删除更多实例。

正在进行扩展活动时不能启动收缩活动。

创建带有步进扩展策略的 Auto Scaling 组

您可以创建一个使用 CloudWatch 警报来确定 Auto Scaling 组何时应扩大或缩小的扩展策略。每个 CloudWatch 警报均监控一个指标，并在该指标超过您在策略中指定的阈值时向 Auto Scaling 发送消息。使用警报可以监控所用的 AWS 服务发送给 CloudWatch 的任何指标，也可以创建和监控自己的自定义指标。

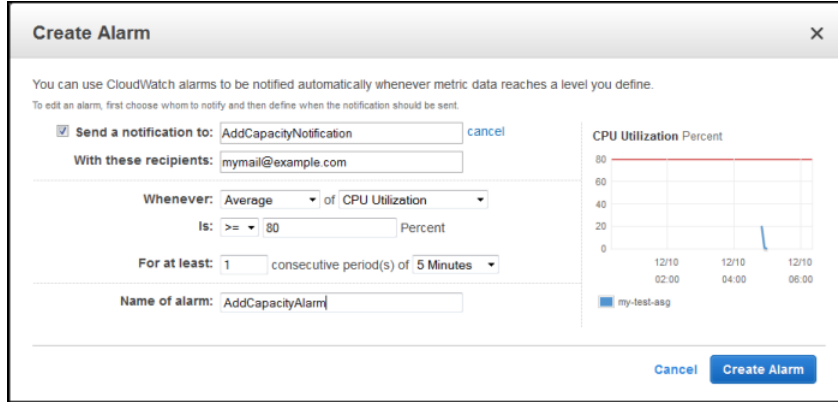
创建 CloudWatch 警报后，可以指定 Amazon SNS 主题，以在警报状态改变时发送电子邮件通知。有关更多信息，请参阅 [创建 Amazon CloudWatch 警报 \(p. 100\)](#)。

使用控制台可以创建具有两个扩展策略的 Auto Scaling 组：将组容量增加 30% 的扩展策略，以及将组容量减小到两个实例的收缩策略。

根据指标创建具有扩展的 Auto Scaling 组

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择 Create Auto Scaling group。
4. 在 Create Auto Scaling Group 页面上，执行以下操作之一：
 - 选择 Create an Auto Scaling group from an existing launch configuration，选择现有的启动配置，然后选择 Next Step。
 - 如果您没有想要使用的启动配置，请选择 Create a new launch configuration，然后按照指示操作。有关更多信息，请参阅 [创建启动配置 \(p. 19\)](#)。
5. 在 Configure Auto Scaling group details 页面上，执行以下操作：
 - a. 对于 Group name，键入 Auto Scaling 组的名称。
 - b. 对于 Group size，键入 Auto Scaling 组所需的容量。
 - c. 如果启动配置指定需要 VPC 的实例（如 T2 实例），则必须从 Network 选择一个 VPC。另外，如果您的 AWS 账户支持 EC2-Classic 并且这些实例不需要 VPC，您可以选择 Launch info EC2-Classic 或 VPC。
 - d. 如果您在上一步中选择了 VPC，请从 Subnet 中选择一个或多个子网。如果您在上一步中已选择 EC2-Classic，请从 Availability Zone(s) 中选择一个或多个可用区。
 - e. 选择 Next: Configure scaling policies。
6. 在 Configure scaling policies (配置扩展策略) 页面上，执行以下操作：
 - a. 选择 Use scaling policies to adjust the capacity of this group (使用扩展策略调整此组的容量)。
 - b. 使用以 Scale between 开头的行指定 Auto Scaling 组的最小大小和最大大小。例如，如果组已经是最大大小，则需要指定一个新的最大值才能进行扩大。

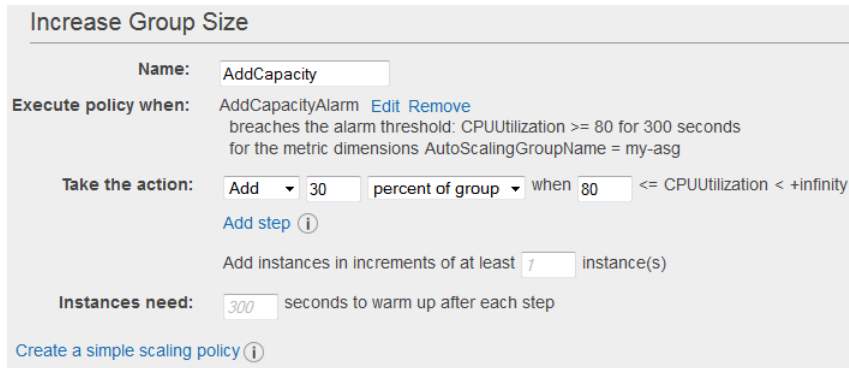
Scale between and instances. These will be the minimum and maximum size of your group.
 - c. 在 Increase Group Size 下指定扩展策略。您可以选择为该策略指定名称，然后选择 Add new alarm。
 - d. 在 Create Alarm 页面上，选择 create topic。对于 Send a notification to，键入 SNS 主题的名称。对于 With these recipients，键入一个或多个用于接收通知的电子邮件地址。如果需要，可以使用自定义名称替换默认警报名称。接下来，为策略指定指标和标准。例如，您可以保留 Whenever 的默认设置 (CPU 使用率平均值)。对于 Is，选择 >= 并键入 80%。对于 For at least，键入 1 个连续的 5 Minutes 时间段。选择 Create Alarm。



- e. 对于 Take the action, 选择 Add, 在下一个字段中键入 30, 然后选择 percent of group。默认情况下, 此步骤调整的下限为警报阈值, 上限为 null (正无穷)。

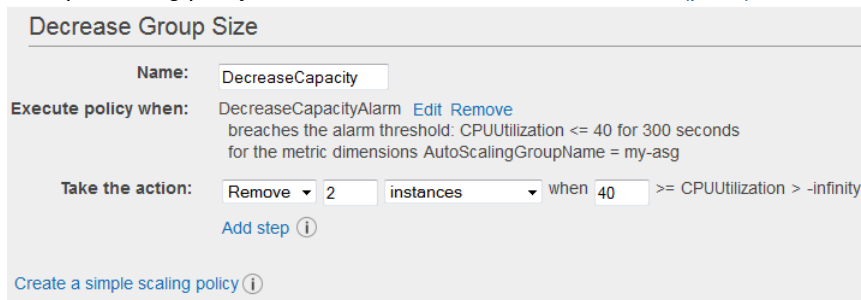
要添加其他步骤调整, 请选择 Add step。要设置可扩展的最少实例数, 请更新 Add instances in increments of at least 1 instance(s) 中的数量字段。

(可选) 建议您采用默认值通过几个步骤创建两个扩展策略。要创建简单扩展策略, 请选择 Create a simple scaling policy。有关更多信息, 请参阅 [扩展策略类型 \(p. 63\)](#)。



- f. 在 Decrease Group Size 下指定收缩策略。您可以选择为该策略指定名称, 然后选择 Add new alarm。
- g. 在 Create Alarm 页面上, 您可以选择为扩展策略创建的通知, 也可以为收缩策略创建一个新通知。如果需要, 可以使用自定义名称替换默认警报名称。保留 Whenever 的默认设置 (CPU 使用率平均值)。对于 Is, 选择 <= 并键入 40%。对于 For at least, 键入 1 个连续的 5 Minutes 时间段。选择 Create Alarm。
- h. 对于 Take the action, 选择 Remove, 在下一个字段中键入 2, 然后选择 instances。默认情况下, 此步骤调整的上限为警报阈值, 下限为 null (负无穷)。要添加其他步骤调整, 请选择 Add step。

(可选) 建议您采用默认值通过几个步骤创建两个扩展策略。要创建简单扩展策略, 请选择 Create a simple scaling policy。有关更多信息, 请参阅 [扩展策略类型 \(p. 63\)](#)。



- i. 选择 Review。
 - j. 在 Review 页面上，选择 Create Auto Scaling group。
7. 使用以下步骤验证您的 Auto Scaling 组的扩展策略。
- a. Auto Scaling Group creation status 页面确认您的 Auto Scaling 组已成功创建。选择 View your Auto Scaling Groups。
 - b. 在 Auto Scaling Groups 页面上，选择您刚创建的 Auto Scaling 组。
 - c. 在 Activity History 选项卡上，Status 列显示 Auto Scaling 组是否具有已成功启动的实例。
 - d. 在 Instances 选项卡上，Lifecycle 列包含实例的状态。启动实例只需很短的时间。实例启动后，其生命周期状态更改为 InService。
- Health Status (运行状态) 列显示对您的实例进行 EC2 实例运行状况检查的结果。
- e. 在 Scaling Policies 选项卡上，可以查看为 Auto Scaling 组创建的策略。

使用 AWS CLI 配置扩展策略

按如下方式使用 AWS CLI 为 Auto Scaling 组配置步进扩展策略。

任务

- [步骤 1：创建 Auto Scaling 组](#) (p. 68)
- [第 2 步：创建扩展策略](#) (p. 68)
- [第 3 步：创建 CloudWatch 警报](#) (p. 69)

步骤 1：创建 Auto Scaling 组

使用以下 [create-auto-scaling-group](#) 命令并利用启动配置 `my-lc` 创建名为 `my-asg` 的 Auto Scaling 组。如果您没有要使用的启动配置，可以创建一个。有关更多信息，请参阅 [create-launch-configuration](#)。

```
aws autoscaling create-auto-scaling-group --auto-scaling-group-name my-asg --launch-configuration-name my-lc --max-size 5 --min-size 1 --availability-zones "us-west-2c"
```

第 2 步：创建扩展策略

您可以创建扩展策略，告诉 Auto Scaling 组在指定条件改变时要执行什么操作。

示例：my-scaleout-policy

使用以下 [put-scaling-policy](#) 命令创建名为 my-scaleout-policy 的扩展策略，调整类型为 PercentChangeInCapacity，它将组容量增加 30%：

```
aws autoscaling put-scaling-policy --policy-name my-scaleout-policy --auto-scaling-group-name my-asg --scaling-adjustment 30 --adjustment-type PercentChangeInCapacity
```

Auto Scaling 返回作为此策略唯一名称的 ARN。随后，您可以使用 ARN 或者策略名称和组名称的组合来指定策略。将此 ARN 保存在安全位置。您需要使用它来创建 CloudWatch 警报。

```
{
  "PolicyARN": "arn:aws:autoscaling:us-west-2:123456789012:scalingPolicy:ac542982-cbeb-4294-891c-a5a941dfa787:autoScalingGroupName/my-asg:policyName/my-scaleout-policy"
}
```

示例：my-scalein-policy

使用以下 `put-scaling-policy` 命令创建名为 `my-scalein-policy` 的扩展策略，调整类型为 `ChangeInCapacity`，它将组容量减少两个实例：

```
aws autoscaling put-scaling-policy --policy-name my-scalein-policy --auto-scaling-group-name my-asg --scaling-adjustment -2 --adjustment-type ChangeInCapacity
```

Auto Scaling 返回策略的 ARN。将此 ARN 保存在安全位置。您需要使用它来创建 CloudWatch 警报。

```
{
  "PolicyARN": "arn:aws:autoscaling:us-west-2:123456789012:scalingPolicy:4ee9e543-86b5-4121-b53b-aa4c23b5bbcc:autoScalingGroupName/my-asg:policyName/my-scalein-policy"
}
```

第 3 步：创建 CloudWatch 警报

在步骤 2 中，您已创建扩展策略，这些策略可以指示 Auto Scaling 组在指定条件发生变化时如何扩大和缩小。在此步骤中，您将通过标识要监控的指标、定义用于扩展的条件来创建警报，然后将警报与扩展策略相关联。

示例：AddCapacity

使用以下 CloudWatch `put-metric-alarm` 命令创建警报，该警报在指定指标的值超过 80 时会增加 Auto Scaling 组的大小。例如，您可以在所有实例的平均 CPU 使用率 (CPUUtilization) 增加到 80% 时添加容量。要使用您自己的自定义指标，请在 `--metric-name` 中指定其名称，并在 `--namespace` 指定其命名空间。

```
aws cloudwatch put-metric-alarm --alarm-name AddCapacity --metric-name CPUUtilization --namespace AWS/EC2 --statistic Average --period 120 --threshold 80 --comparison-operator GreaterThanOrEqualToThreshold --dimensions "Name=AutoScalingGroupName,Value=my-asg" --evaluation-periods 2 --alarm-actions PolicyARN
```

示例：RemoveCapacity

使用以下 CloudWatch `put-metric-alarm` 命令创建警报，该警报在指定指标的值超过 40 时会减小 Auto Scaling 组的大小。例如，您可以在所有实例的平均 CPU 使用率 (CPUUtilization) 减少到 40% 时删除容量。要使用您自己的自定义指标，请在 `--metric-name` 中指定其名称，并在 `--namespace` 中指定其命名空间。

```
aws cloudwatch put-metric-alarm --alarm-name RemoveCapacity --metric-name CPUUtilization --namespace AWS/EC2 --statistic Average --period 120 --threshold 40 --comparison-operator LessThanOrEqualToThreshold --dimensions "Name=AutoScalingGroupName,Value=my-asg" --evaluation-periods 2 --alarm-actions PolicyARN
```

目标跟踪扩展

目标跟踪扩展策略可简化配置动态扩展的过程。您可以选择预定义的指标，也可以配置自定义指标并设置一个目标值。Auto Scaling 会创建和管理触发扩展策略的 CloudWatch 警报，并根据指标和目标值计算扩展调整。扩展策略根据需要增加或减少容量，将指标保持在指定的目标值或接近指定的目标值。除了将指标保持在目标值附近以外，目标跟踪扩展策略还能根据负载波动模式调节指标波动，最大限度地减少 Auto Scaling 组容量的快速波动。

例如，您可以使用目标跟踪扩展进行以下操作：

- 配置目标跟踪扩展策略，使 Auto Scaling 组的平均聚合 CPU 利用率保持在 50%。
- 为 Auto Scaling 组配置目标跟踪扩展策略，使 应用程序负载均衡器 目标组的每目标请求数保持在 1000。

指定自定义指标时，请注意，有些指标并不适用于目标跟踪。指标必须是有效的使用率指标，它用于描述实例的繁忙程度。指标值必须与 Auto Scaling 组中的实例数成比例地增加或减少，以便能够利用指标数据按比例地扩展或缩减实例数量。例如，如果某个 Auto Scaling 组的负载分布在各个实例上，则 CPU 使用率指标 (即维度为 `AutoScalingGroupName` 的 Amazon EC2 指标 `CPUUtilization`) 能够正常工作。以下指标不起作用：

- 作为 Auto Scaling 组前端的负载均衡器收到的请求数 (即 Elastic Load Balancing 指标 `RequestCount`)，因为负载均衡器收到的请求数不随 Auto Scaling 组的使用率而改变。
- 负载均衡器请求延迟 (即 Elastic Load Balancing 指标 `Latency`)，因为请求延迟可能会随使用率增加而增加，但不一定会按比例变化。
- SQS 指标 `ApproximateNumberOfMessagesVisible`，因为队列中的消息数量可能不会与处理该队列消息的 Auto Scaling 组的大小成比例变化。但是，如果消息分布在各个实例上，则用于衡量 Auto Scaling 组每实例队列消息数的自定义指标能够正常工作。

您可以为 Auto Scaling 组提供多个目标跟踪扩展策略，前提是它们各自使用不同的指标。对于扩展和收缩，Auto Scaling 组根据在组中提供最大容量的策略进行扩展。这样可以更灵活地覆盖多种场景，确保始终有足够的容量来处理应用程序工作负载。

您还可以选择禁用目标跟踪扩展策略的收缩部分。该功能为您提供了更大的灵活性：您可以使用目标跟踪扩展策略扩展 Auto Scaling 组，同时使用其他的方法 (例如不同的扩展策略类型) 收缩 Auto Scaling 组。

请注意以下事项：

- 目标跟踪扩展策略假设它应该在指定指标高于目标值时扩展 Auto Scaling 组。因此，不能使用目标跟踪扩展策略在指定指标低于目标值时收缩 Auto Scaling 组。
- 当指定指标数据不足时，目标跟踪扩展策略不会扩展 Auto Scaling 组。它不会将数据不足解读为使用率低，因此不会收缩 Auto Scaling 组。要在指定指标数据不足时收缩 Auto Scaling 组，请创建一个简单或步进扩展策略，并设置警报在 Auto Scaling 组状态变为 `INSUFFICIENT_DATA` 时调用该扩展策略。例如，应用程序负载均衡器 指标 `RequestCountPerTarget`，这是一个预定义指标，当没有请求被路由到目标组时，它没有数据点。要在没有请求路由到目标组时收缩 Auto Scaling 组，请创建一个简单或步进扩展策略，为该指标创建警报，并将警报设置为在 Auto Scaling 组状态变为 `INSUFFICIENT_DATA` 时调用该扩展策略。
- 您可能会看到目标值与实际指标数据点之间存在差距。这是因为 Auto Scaling 在确定需要添加或删除多少个实例时始终通过向上或向下舍入来保守地进行操作，以免 Auto Scaling 添加的实例数量不足或删除的实例数量过多。但是，对于实例较少的 Auto Scaling 组，组的使用率可能偏离目标值较远。例如，假设将 `CPUUtilization` 的目标值设置为 50%，则当 Auto Scaling 组超出目标值时，Auto Scaling 确定添加 1.5 个实例可将 CPU 使用率降至 50% 左右。由于不可能添加 1.5 个实例，Auto Scaling 将该值向上舍入，添加两个实例。这可能会将 CPU 使用率降至 50% 以下，但可确保应用程序具有充足的支持资源。同样，如果 Auto Scaling 确定删除 1.5 个实例可使 CPU 使用率提高到 50% 以上，它将只删除一个实例。对于实例较多的 Auto Scaling 组，使用率分布在更多的实例上。添加或删除实例会导致目标值与实际指标数据点之间的差距缩小。
- 建议将随指标扩展的频率设置为 1 分钟，这可更快地响应使用率变化。如果将随指标扩展的频率设置为 5 分钟，可能会导致响应时间变慢，并且可能导致系统依据陈旧的指标数据进行扩展。默认情况下，Amazon EC2 实例是针对基本监控启用的，也就是说，实例的指标数据以 5 分钟的间隔提供。您可以启用详细监控，从而以 1 分钟的频率获取实例的指标数据。有关更多信息，请参阅 [Auto Scaling 实例配置监控](#)。
- 不要编辑或删除 Auto Scaling 为目标跟踪扩展策略管理的 CloudWatch 警报。当您删除 Auto Scaling 策略时，Auto Scaling 会自动删除相应的警报。

创建具有目标跟踪扩展策略的 Auto Scaling 组

使用控制台创建具有目标跟踪扩展策略的 Auto Scaling 组。

根据指标创建具有扩展的 Auto Scaling 组

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择 Create Auto Scaling group。
4. 在 Create Auto Scaling Group 页面上，执行以下操作之一：
 - 选择 Create an Auto Scaling group from an existing launch configuration，选择现有的启动配置，然后选择 Next Step。
 - 如果您没有想要使用的启动配置，请选择 Create a new launch configuration，然后按照指示操作。有关更多信息，请参阅 [创建启动配置 \(p. 19\)](#)。
5. 在 Configure Auto Scaling group details 页面上，执行以下操作：
 - a. 对于 Group name，键入 Auto Scaling 组的名称。
 - b. 对于 Group size，键入 Auto Scaling 组所需的容量。
 - c. 如果启动配置指定需要 VPC 的实例（如 T2 实例），则必须从 Network 选择一个 VPC。另外，如果您的 AWS 账户支持 EC2-Classic 并且这些实例不需要 VPC，您可以选择 Launch into EC2-Classic 或 VPC。
 - d. 如果您在上一步中选择了 VPC，请从 Subnet 中选择一个或多个子网。如果您在上一步中已选择 EC2-Classic，请从 Availability Zone(s) 中选择一个或多个可用区。
 - e. 选择 Next: Configure scaling policies。
6. 在 Configure scaling policies (配置扩展策略) 页面上，执行以下操作：
 - a. 选择 Use scaling policies to adjust the capacity of this group (使用扩展策略调整此组的容量)。
 - b. 使用以 Scale between 开头的行指定 Auto Scaling 组的最小大小和最大大小。例如，如果组已经是最大大小，则指定一个新的最大值才能进行扩展。
 - c. 对于 Scale Group Size，指定扩展策略。您可以选择为该策略指定名称，然后为 Metric type 选择一个值。
 - d. 为指标指定 Target value。
 - e. 为 Instances need 指定实例预热值，用于控制新启动的实例在多长时间后开始作用于 CloudWatch 指标。
 - f. 选中 Disable scale-in 选项则只创建扩展策略。这样，可以根据需要创建独立的其他类型的收缩策略。
 - g. 选择 Review。
 - h. 在 Review 页面上，选择 Create Auto Scaling group。

实例预热

通过目标跟踪扩展策略可以指定新启动实例的预热时间（秒数）。在指定预热时间过期前，实例不会计入 Auto Scaling 组的聚合指标。

在扩展时，Auto Scaling 不会将正在预热的实例看作当前组容量的一部分。这可确保添加的实例不会超出您的需要。

在收缩时，Auto Scaling 将正在终止的实例视为当前组容量的一部分。因此不会从 Auto Scaling 组中删除更多实例。

正在进行扩展活动时不能启动收缩活动。

使用 AWS CLI 配置扩展策略

按如下方式使用 AWS CLI 为 Auto Scaling 组配置目标跟踪扩展策略。

任务

- [步骤 1：创建 Auto Scaling 组 \(p. 72\)](#)
- [第 2 步：创建扩展策略 \(p. 72\)](#)

步骤 1：创建 Auto Scaling 组

使用以下 [create-auto-scaling-group](#) 命令并利用启动配置 `my-lc` 创建名为 `my-asg` 的 Auto Scaling 组。如果您没有要使用的启动配置，可以创建一个。有关更多信息，请参阅 [create-launch-configuration](#)。

```
aws autoscaling create-auto-scaling-group --auto-scaling-group-name my-asg --launch-configuration-name my-lc --max-size 5 --min-size 1 --availability-zones "us-west-2c"
```

第 2 步：创建扩展策略

您可以创建扩展策略，告诉 Auto Scaling 组在指定条件改变时要执行什么操作。

示例：目标跟踪配置文件

下面是一个目标跟踪配置文件示例 (应将其另存为 `config.json`)：

```
{
  "TargetValue": "40.0",
  "PredefinedMetricSpecification": {
    {
      "PredefinedMetricType": "ASGAverageCPUUtilization"
    }
  }
}
```

示例：my-scaleout-policy

使用下面的 [put-scaling-policy](#) 命令及之前创建的 `config.json` 文件创建一个名为 `cpu40` 的扩展策略，用于将 Auto Scaling 组的 CPU 平均使用率保持在 40%：

```
aws autoscaling put-scaling-policy --policy-name cpu40 --auto-scaling-group-name my-asg --policy-type TargetTrackingScaling --target-tracking-configuration file://config.json
```

向现有的 Auto Scaling 组添加扩展策略

使用控制台将扩展策略添加到现有的 Auto Scaling 组。

根据指标更新具有扩展的 Auto Scaling 组

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择 Auto Scaling 组。
4. 在 Scaling Policies 选项卡上，选择 Add policy。
5. 如果使用的是目标跟踪扩展策略，请按照以下步骤操作。如果使用的是简单或步进扩展策略，请跳到下一步。
 - a. 在 Name 中键入策略的名称。
 - b. 选择 Metric type 并为该指标指定 Target value。
 - c. 为 Instances need 指定实例预热值，用于控制新启动的实例在多长时间后开始作用于 CloudWatch 指标。
 - d. 如果只想创建扩展策略，请选中 Disable scale-in 选项。这样，可以根据需要创建独立的收缩策略。

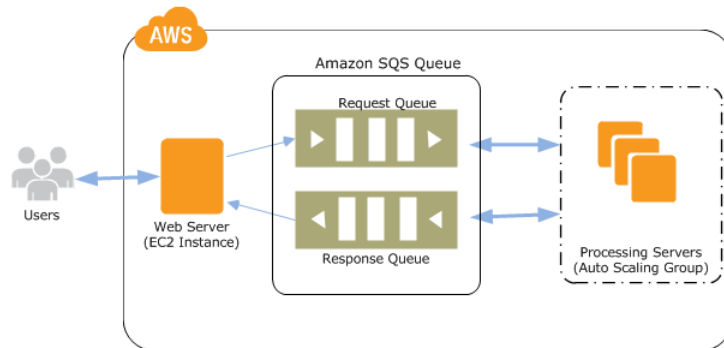
- e. 选择 Create。
6. 如果使用的是步进扩展策略，请执行以下操作：
 - a. 对于 Name，键入策略的名称，然后选择 Create new alarm。
 - b. 在 Create Alarm 页面上，选择 create topic。对于 Send a notification to，键入 SNS 主题的名称。对于 With these recipients，键入一个或多个用于接收通知的电子邮件地址。如果需要，可以使用自定义名称替换默认警报名称。下一步，使用 Whenever、Is 和 For at least 为该警报指定指标和标准。选择 Create Alarm。
 - c. 使用 Take the action 为策略指定扩展活动。默认情况下，此步骤调整的下限为警报阈值，上限为 null（正无穷）。要添加其他步骤调整，请选择 Add step。

(可选) 建议您采用默认值通过几个步骤创建两个扩展策略。要创建简单扩展策略，请选择 Create a simple scaling policy。有关更多信息，请参阅 [扩展策略类型 \(p. 63\)](#)。
 - d. 选择 Create。

基于 Amazon SQS 进行扩展

Amazon Simple Queue Service (Amazon SQS) 是可扩展的消息排队系统，当消息在应用程序架构的各个组件之间传输时，该系统可以存储消息。Amazon SQS 可以让 Web 服务应用程序快速可靠地对一个组件生成、由另一个组件使用的消息进行排队。队列是等待处理的消息的临时储存库。有关更多信息，请参阅 [Amazon Simple Queue Service 开发人员指南](#)。

例如，假设您有一个用于接收客户订单的 Web 应用程序。该应用程序在已配置为处理典型数量的订单的 Auto Scaling 组中的 EC2 实例上运行。该应用程序将订单放置在 Amazon SQS 队列中，选取订单以备处理，处理订单，然后将已处理的订单返回给客户。下图将演示此示例的架构。



如果您的订单水平始终保持不变，该架构将运作良好。如果订单水平发生变化，会发生什么？当订单增加时，需要启动更多 EC2 实例，当订单减少时，需要终止多余的 EC2 实例。如果订单按照可预见的计划增长和降低，您可以指定执行扩展活动的时间和日期。有关更多信息，请参阅 [计划的扩展 \(p. 60\)](#)。另外，您可以根据条件进行扩展，例如 SQS 队列中的消息数。有关更多信息，请参阅 [动态扩展 \(p. 63\)](#)。

队列可以提供用于方便地确定应用程序上的负载的机制。您可以使用队列长度（可从队列中检索的消息数量）来确定负载。因为队列中的每个消息都代表来自用户的请求，所以测量的队列长度十分近似于应用程序上的负载。CloudWatch 集成了 Amazon SQS，可以收集、查看和分析来自 SQS 队列的指标。您可以使用 Amazon SQS 发送的指标来确定任何时间点的 SQS 队列长度。有关 Amazon SQS 发送到 CloudWatch 的所有指标的列表，请参阅 Amazon Simple Queue Service 开发人员指南 中的 [Amazon SQS 指标](#)。

以下示例创建的 Auto Scaling 策略将 Auto Scaling 组配置为根据 SQS 队列中的消息数进行扩展。

使用 AWS CLI 通过 Amazon SQS 进行扩展

以下示例介绍如何创建扩大和缩小策略，以及如何为扩展策略创建、验证和确认 CloudWatch 警报。假设您已经有一个 SQS 队列、一个 Auto Scaling 组，以及运行使用 SQS 队列的应用程序的 EC2 实例。

创建扩展策略

您可以创建扩展策略，告诉 Auto Scaling 组在指定条件改变时要执行什么操作。

创建扩展策略的步骤

1. 使用以下 `put-scaling-policy` 命令创建一个为 Auto Scaling 组增加一个 EC2 实例的扩展策略：

```
aws autoscaling put-scaling-policy --policy-name my-sqs-scaleout-policy --auto-scaling-group-name my-asg --scaling-adjustment 1 --adjustment-type ChangeInCapacity
```

Auto Scaling 返回新策略的 Amazon 资源名称 (ARN)。将 ARN 保存在安全位置。在创建 CloudWatch 警报时，会需要它。

2. 使用以下 `put-scaling-policy` 命令创建一个将 Auto Scaling 组减少一个 EC2 实例的收缩策略：

```
aws autoscaling put-scaling-policy --policy-name my-sqs-scalein-policy --auto-scaling-group-name my-asg --scaling-adjustment -1 --adjustment-type ChangeInCapacity
```

Auto Scaling 返回新策略的 ARN。将 ARN 保存在安全位置。在创建 CloudWatch 警报时，会需要它。

创建 CloudWatch 警报

下一步，通过识别要监控的指标、定义扩展条件，然后将警报与您在先前任务中创建的扩展策略相关联，来创建警报。

Note

所有活动的 SQS 队列每 5 分钟向 CloudWatch 发送一次指标。我们建议您将警报 Period 设置为至少 300 秒。将警报 Period 设置为低于 300 秒将导致在等待指标时警报进入 `INSUFFICIENT_DATA` 状态。

创建 CloudWatch 警报

1. 使用以下 `put-metric-alarm` 命令来创建警报，如果队列中可处理的消息的数量 (`ApproximateNumberOfMessagesVisible`) 增加到 3，并持续 5 分钟保持大于等于 3，则该警报增加 Auto Scaling 组的大小。

```
aws cloudwatch put-metric-alarm --alarm-name AddCapacityToProcessQueue --metric-name ApproximateNumberOfMessagesVisible --namespace "AWS/SQS" --statistic Average --period 300 --threshold 3 --comparison-operator GreaterThanOrEqualToThreshold --dimensions Name=QueueName,Value=my-queue --evaluation-periods 2 --alarm-actions arn
```

2. 使用以下 `put-metric-alarm` 命令来创建警报，如果队列中可处理的消息的数量 (`ApproximateNumberOfMessagesVisible`) 减少到 1，并持续 5 分钟保持小于等于 1，则该警报减小 Auto Scaling 组的大小。

```
aws cloudwatch put-metric-alarm --alarm-name RemoveCapacityFromProcessQueue --metric-name ApproximateNumberOfMessagesVisible --namespace "AWS/SQS" --statistic Average --period 300 --threshold 1 --comparison-operator LessThanOrEqualToThreshold --dimensions Name=QueueName,Value=my-queue --evaluation-periods 2 --alarm-actions arn
```

验证您的扩展策略和 CloudWatch 警报

您可以验证是否已创建 CloudWatch 警报和扩展策略。

验证 CloudWatch 警报

使用以下 [describe-alarms](#) 命令：

```
aws cloudwatch describe-alarms --alarm-  
names AddCapacityToProcessQueue RemoveCapacityFromProcessQueue
```

验证扩展策略的步骤

使用以下 [describe-policies](#) 命令：

```
aws autoscaling describe-policies --auto-scaling-group-name my-asg
```

测试扩大和缩小策略

您可以通过以下方式测试扩展策略：增加 SQS 队列中的消息数，然后验证 Auto Scaling 组是否已启动更多 EC2 实例。同样，您可以通过以下方式测试收缩策略：减少 SQS 队列中的消息数，然后验证 Auto Scaling 组是否终止了一个 EC2 实例。

测试扩展策略

1. 请按照 [Amazon SQS 入门](#) 中的步骤，向 SQS 队列添加消息。确保队列中至少有 3 条消息。

SQS 队列指标 `ApproximateNumberOfMessagesVisible` 调用 CloudWatch 警报需要几分钟的时间。调用 CloudWatch 警报后，该警报会通知 Auto Scaling 策略启动 1 个 EC2 实例。

2. 使用以下 [describe-auto-scaling-groups](#) 命令验证该组是否已启动了实例：

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

测试收缩策略

1. 请按照 [Amazon SQS 入门](#) 中的步骤，从 SQS 队列删除消息。请确保队列中仅有 1 条消息。

SQS 队列指标 `ApproximateNumberOfMessagesVisible` 调用 CloudWatch 警报需要几分钟的时间。调用 CloudWatch 警报后，该警报会通知 Auto Scaling 策略终止 1 个 EC2 实例。

2. 使用以下 [describe-auto-scaling-groups](#) 命令验证该组是否已终止了实例：

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

Auto Scaling 冷却时间

Auto Scaling 冷却时间是 Auto Scaling 组的一个可配置设置，可以帮助确保在上一扩展活动生效前 Auto Scaling 不会启动或终止其他实例。在 Auto Scaling 组使用简单的扩展策略动态扩展后，Auto Scaling 会等待冷却时间完成，然后再继续扩展活动。在手动扩展 Auto Scaling 组时，默认设置是不等待冷却时间，但您可以覆盖默认设置并采用冷却时间。如果实例运行状况不佳，Auto Scaling 不会等待冷却时间结束才替换运行状况不佳的实例。

Important

步进扩展策略或计划的扩展不支持冷却时间。

Auto Scaling 支持默认冷却时间和特定于扩展的冷却时间。

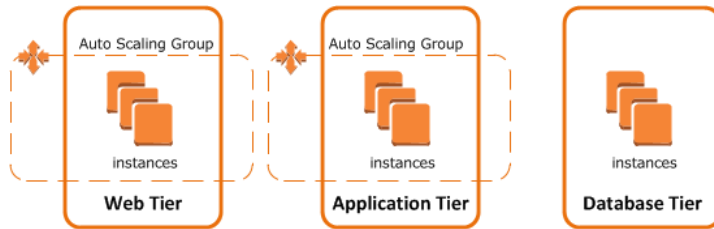
内容

- [示例：Auto Scaling 冷却时间 \(p. 76\)](#)

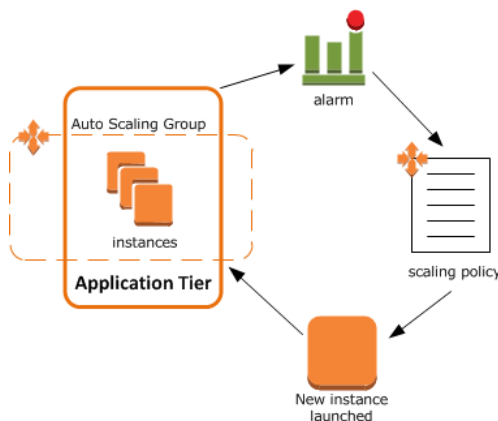
- [默认冷却时间 \(p. 76\)](#)
- [特定于扩展的冷却时间 \(p. 77\)](#)
- [冷却时间和多个实例 \(p. 77\)](#)
- [冷却时间和生命周期挂钩 \(p. 77\)](#)
- [冷却时间和竞价型实例 \(p. 77\)](#)

示例：Auto Scaling 冷却时间

考虑以下情景：有一个在 AWS 中运行的 Web 应用程序。此 Web 应用程序包含三个基本层：Web、应用程序和数据库。为了确保应用程序始终拥有足够的资源来满足流量需求，您创建了两个 Auto Scaling 组：一个用于 Web 层，一个用于应用程序层。



为了帮助确保用于应用程序层的 Auto Scaling 组具有适当数量的 EC2 实例，请[创建一个 CloudWatch 警报 \(p. 100\)](#)，每当实例的 CPUUtilization 指标超过 90% 时便会扩展。当警报发生时，Auto Scaling 会启动并配置其他实例。



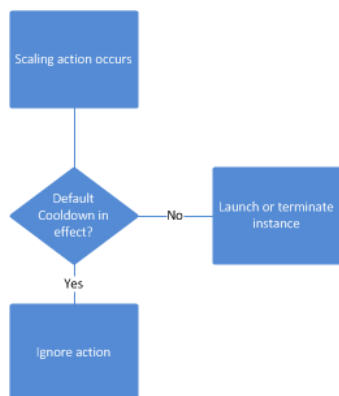
在实例投入使用之前，这些实例使用配置脚本安装和配置软件。因此，实例从启动到投入使用大约需要两到三分钟的时间。实际时间取决于诸多因素，如实例大小和是否有启动脚本要完成等。

现在出现了流量高峰，从而导致 CloudWatch 警报触发。该警报触发时，Auto Scaling 会启动一个实例来帮助处理增加的需求。但是存在一个问题：该实例需要几分钟的时间才能启动。在此期间，CloudWatch 警报可能会继续触发，从而导致 Auto Scaling 在警报每次出现时都另外启动一个实例。

但是，有了冷却时间，Auto Scaling 在启动一个实例后，将暂停所有简单扩展策略或手动扩展引起的扩展活动，直至经过了指定时间量。（默认值为 300 秒。）这样，新启动的实例有时间开始处理应用程序流量。冷却时间过后，所有暂停的扩展操作都会恢复。如果 CloudWatch 警报再次触发，则 Auto Scaling 将启动另一个实例，而冷却时间也会再次生效。不过，如果新增的实例足以将 CPU 使用率降为正常水平，则该组会保持其当前大小。

默认冷却时间

创建 Auto Scaling 组时会应用默认冷却时间。其默认值为 300 秒。此冷却时间自动应用于简单扩展策略的任何[动态扩展 \(p. 63\)](#)活动，您可以选择请求将此冷却时间应用于您的[手动扩展 \(p. 53\)](#)活动。



创建 Auto Scaling 组时，您可以使用 AWS 管理控制台、[create-auto-scaling-group](#) 命令 (AWS CLI) 或 [CreateAutoScalingGroup](#) API 操作配置默认冷却时间。

您可以随时使用 AWS 管理控制台、[update-auto-scaling-group](#) 命令 (AWS CLI) 或 [UpdateAutoScalingGroup](#) API 操作更改默认冷却时间。

特定于扩展的冷却时间

除了指定 Auto Scaling 组的默认冷却时间外，您还可以创建适用于特定简单扩展策略或手动扩展的冷却时间。特定于扩展的冷却时间会覆盖默认冷却时间。

特定于扩展的冷却时间常用于收缩策略，即根据特定条件或指标终止实例的策略。由于此策略是要终止实例，Auto Scaling 需要较短的时间来确定是否终止其他实例。而 300 秒的默认冷却时间就太长了，您可以通过对收缩策略应用特定于扩展的 180 秒冷却时间来降低成本。

您可以使用 AWS 管理控制台、[put-scaling-policy](#) 命令 (AWS CLI) 或 [PutScalingPolicy](#) API 操作创建特定于扩展的冷却时间。

冷却时间和多个实例

前面部分提供的示例演示了冷却时间在启动或终止单个实例时对 Auto Scaling 组的影响方式。但是，Auto Scaling 组一次启动多个实例并不少见。例如，您可能会选择让 Auto Scaling 在达到特定指标阈值时启动三个实例。

对于多个实例，冷却时间（无论是默认冷却时间还是特定于扩展的冷却时间）会在最后一个实例启动时开始生效。

冷却时间和生命周期挂钩

Auto Scaling 支持向 Auto Scaling 组添加生命周期挂钩。通过这些挂钩，您可以控制如何在 Auto Scaling 组中启动和终止实例，从而可以在实例投入使用之前或终止之前对实例执行操作。

生命周期挂钩可以影响针对 Auto Scaling 组、手动扩展或简单扩展策略配置的任何冷却时间的行为。冷却时间在实例退出等待状态之后才会开始。

冷却时间和竞价型实例

您可以创建 Auto Scaling 组以使用[竞价型实例](#) (p. 28)，而不使用按需实例或预留实例。当所有竞价型实例出价成功时，冷却时间开始。

控制 Auto Scaling 在缩小过程中终止哪些实例

对于每个 Auto Scaling 组，您可以控制 Auto Scaling 何时向网络架构添加实例（称为扩展）或从中删除实例（称为缩小）。您可以通过附加和分离实例手动扩展组大小，或者可以使用扩展策略自动执行该过程。

让 Auto Scaling 自动缩小时，您必须确定哪些实例 Auto Scaling 应首先终止。您可以使用终止策略对此进行配置。

在缩小时，您也可以使用实例保护防止 Auto Scaling 选择终止特定的实例。

内容

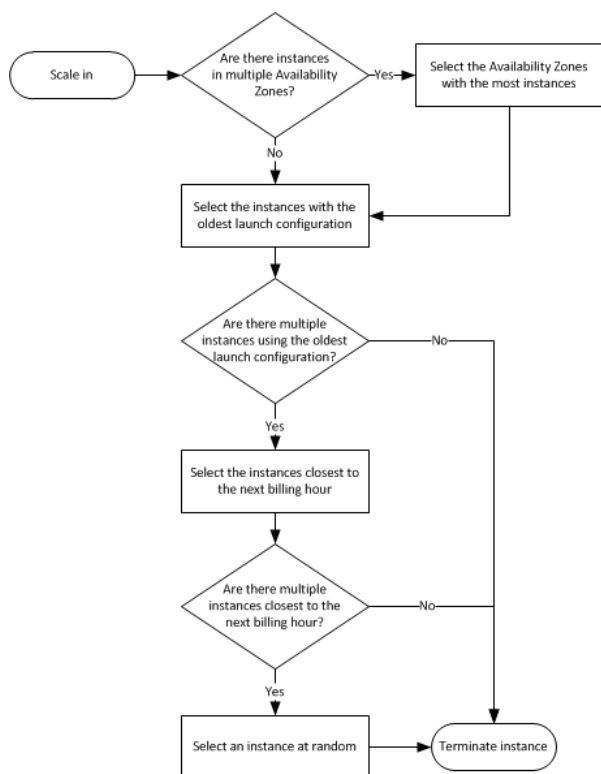
- [默认终止策略 \(p. 78\)](#)
- [自定义终止策略 \(p. 80\)](#)
- [实例保护 \(p. 81\)](#)

默认终止策略

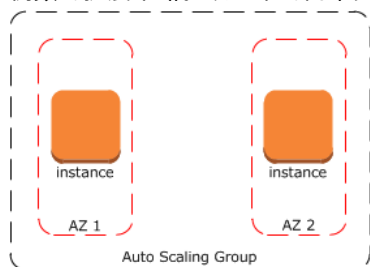
此默认终止策略可帮助确保您的网络架构均匀分布到多个可用区。当使用默认终止策略时，Auto Scaling 按如下方式选择要终止的实例：

1. Auto Scaling 确定在多个可用区中是否有实例。如果是，它会选择有最多实例并且至少一个实例不受缩小保护的可用区。如果有多个可用区有此数目的实例，则 Auto Scaling 选择使用最旧启动配置的实例所在的可用区。
2. Auto Scaling 确定所选可用区中的哪些不受保护的实例使用最旧启动配置。如果有一个此类实例，它会终止该实例。
3. 如果有多个实例使用最旧启动配置，则 Auto Scaling 确定哪些不受保护的实例最接近下个计费小时。（这有助于您最大程度使用您的 EC2 实例，同时最大程度减少对 Amazon EC2 用量的计费时数。）如果有一个此类实例，Auto Scaling 会终止该实例。
4. 如果有多个不受保护的实例最接近下个计费小时，则 Auto Scaling 随机选择其中一个实例。

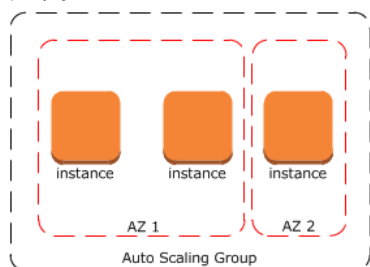
以下流程图说明默认终止策略的工作方式。



考虑一个有两个可用区的 Auto Scaling 组，所需容量为两个实例，采用在达到特定阈值时按 1 增加和减少实例数的扩展策略。该组中的两个实例分布如下。



当达到扩展策略的阈值时，策略生效，Auto Scaling 启动新实例。Auto Scaling 组中现在有三个实例，分布如下。



当达到收缩策略的阈值时，策略生效，Auto Scaling 终止其中一个实例。如果该组没有分配特定的终止策略，则 Auto Scaling 使用默认终止策略。Auto Scaling 选择有两个实例的可用区，终止从最旧启动配置启动的实例。如果这些实例是从同一启动配置启动的，则 Auto Scaling 选择最接近下个计费小时的实例并终止该实例。

自定义终止策略

为 Auto Scaling 组分配的默认终止策略通常足以适用于大多数情况。但是，您可以选择将默认策略替换为自定义策略。

当您自定义终止策略时，Auto Scaling 首先评估可用区是否存在不平衡。如果某个可用区的实例数多于该组使用的其他可用区，则 Auto Scaling 应用您对不平衡可用区中的实例指定的终止策略。如果该组使用的可用区是平衡的，则 Auto Scaling 应用您指定的终止策略。

Auto Scaling 当前支持以下自定义终止策略：

- **OldestInstance**。Auto Scaling 终止组中最旧的实例。当您把 Auto Scaling 组中的实例升级为新的 EC2 实例类型，可以逐渐将较旧类型的实例替换为较新类型的实例时，此选项十分有用。
- **NewestInstance**。Auto Scaling 终止组中最新的实例。如果要测试新的启动配置但不想在生产中保留它时，此策略非常有用。
- **OldestLaunchConfiguration**。Auto Scaling 终止采用最旧启动配置的实例。如果要更新某个组并且逐步淘汰先前配置中的实例时，此策略非常有用。
- **ClosestToNextInstanceHour**。Auto Scaling 终止最接近下个计费小时的实例。此策略帮助您最大程度使用您的实例并管理成本。
- **Default**。Auto Scaling 使用默认终止策略。如果您有多个扩展策略与该组关联，此策略非常有用。

使用控制台自定义终止策略

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上，选择 Auto Scaling Groups。
3. 选择 Auto Scaling 组。
4. 对于 Actions，选择 Edit。
5. 在 Details 选项卡上，找到 Termination Policies。选择一个或多个终止策略。如果您选择多个策略，请按照您希望应用这些策略的顺序将其列出。如果您使用 Default 策略，请将该策略设为列表中的最后一项。
6. 选择 Save。

使用 AWS CLI 自定义终止策略

使用以下命令之一：

- [create-auto-scaling-group](#)
- [update-auto-scaling-group](#)

您可以单独使用这些策略，或者将它们合并到 Auto Scaling 在终止实例时使用的策略的列表中。例如，使用以下命令更新 Auto Scaling 组以首先使用 **OldestLaunchConfiguration** 策略，然后使用 **ClosestToNextInstanceHour** 策略：

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg --termination-policies "OldestLaunchConfiguration,ClosestToNextInstanceHour"
```

如果您使用 Default 终止策略，请将该策略设为终止策略列表中的最后一项。例如：`--termination-policies "OldestLaunchConfiguration,Default"`。

实例保护

要控制 Auto Scaling 在缩小时是否可以终止特定实例，请使用实例保护。您可以对一个 Auto Scaling 组或单个 Auto Scaling 实例启用实例保护设置。当 Auto Scaling 启动实例时，实例会继承 Auto Scaling 组的实例保护设置。您可以随时更改 Auto Scaling 组或 Auto Scaling 实例的实例保护设置。

当实例状态为 `InService` 时，实例保护启动。当您分离受终止保护的实例时，其实例保护设置就会失效。再次将实例挂载到组时，它会继承组的当前实例保护设置。

如果缩小期间 Auto Scaling 组中的所有实例都受终止保护并且发生缩小事件，则 Auto Scaling 会减少所需容量。不过，Auto Scaling 不能终止所需数量的实例，直到其实例保护设置被禁用。

实例保护并不能针对以下情况保护 Auto Scaling 实例：

- 通过 Amazon EC2 控制台、`terminate-instances` 命令或 `TerminateInstances` 操作手动终止。要保护 Auto Scaling 实例不被手动终止，请启用终止保护。有关更多信息，请参阅 Amazon EC2 用户指南（适用于 Linux 实例）中的 [启用终止保护](#)。
- 实例未通过运行状况检查的情况下的运行状况检查更换。
- 竞价型实例中断。

任务

- [启用组的实例保护 \(p. 81\)](#)
- [修改组的实例保护设置 \(p. 82\)](#)
- [修改实例的实例保护设置 \(p. 82\)](#)

启用组的实例保护

您可以在创建 Auto Scaling 组时启用实例保护。默认情况下，实例保护是禁用的。

使用控制台启用实例保护

在创建 Auto Scaling 组时，在 `Configure Auto Scaling group details` 页上，在 `Advanced Details` 下从 `Instance Protection` 中选择 `Protect From Scale In` 选项。

▼ **Advanced Details**

Load Balancing ⓘ	You currently don't have any load balancers Learn about Elastic Load Balancing
Health Check Grace Period ⓘ	<input type="text" value="300"/> seconds
Monitoring ⓘ	Amazon EC2 Detailed Monitoring metrics, which are provided at 1 minute frequency, are not enabled for the launch configuration <code>my-lc</code> . Instances launched from it will use Basic Monitoring metrics, provided at 5 minute frequency. Learn more
Instance Protection ⓘ	<div><input type="text" value="Protect From Scale In"/></div>

使用 AWS CLI 启用实例保护

使用以下 `create-auto-scaling-group` 命令启用实例保护：

```
aws autoscaling create-auto-scaling-group --auto-scaling-group-name my-asg --new-instances-protected-from-scale-in ...
```

修改组的实例保护设置

您可以启用或禁用 Auto Scaling 组的实例保护设置。

使用控制台更改组的实例保护设置

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上，选择 Auto Scaling Groups。
3. 选择 Auto Scaling 组。
4. 在 Details 选项卡上，选择 Edit。
5. 对于 Instance Protection，选择 Protect From Scale In。



6. 选择 Save。

使用 AWS CLI 更改组的实例保护设置

使用以下 `update-auto-scaling-group` 命令可对指定的 Auto Scaling 组启用实例保护：

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg --new-instances-protected-from-scale-in
```

使用以下命令可对指定的组禁用实例保护：

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg --no-new-instances-protected-from-scale-in
```

修改实例的实例保护设置

默认情况下，实例从其 Auto Scaling 组获取其实例保护设置。但是，您随时可以对实例启用或禁用实例保护。

使用控制台更改实例的实例保护设置

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上，选择 Auto Scaling Groups。
3. 选择 Auto Scaling 组。
4. 在 Instances 选项卡上，选择实例。
5. 要启用实例保护，请依次选择 Actions、Instance Protection 和 Set Scale In Protection。系统提示时，选择 Set Scale In Protection。
6. 要禁用实例保护，请依次选择 Actions、Instance Protection 和 Remove Scale In Protection。系统提示时，选择 Remove Scale In Protection。

使用 AWS CLI 更改实例的实例保护设置

使用以下 `set-instance-protection` 命令可对指定实例启用实例保护：

```
aws autoscaling set-instance-protection --instance-ids i-5f2e8a0d --auto-scaling-group-name my-asg --protected-from-scale-in
```


使用以下命令可对指定实例禁用实例保护：

```
aws autoscaling set-instance-protection --instance-ids i-5f2e8a0d --auto-scaling-group-name my-asg --no-protected-from-scale-in
```

Auto Scaling 生命周期挂钩

Auto Scaling 生命周期挂钩 使您能够在 Auto Scaling 启动或终止实例时执行自定义操作。例如，您可以在新启动的实例上安装或配置软件，也可以在实例终止前从实例中下载日志文件。

每个 Auto Scaling 组均可具有多个生命周期挂钩。但是，每个 Auto Scaling 组的挂钩数存在限制。有关更多信息，请参阅 [Auto Scaling 账户限制](#)。

内容

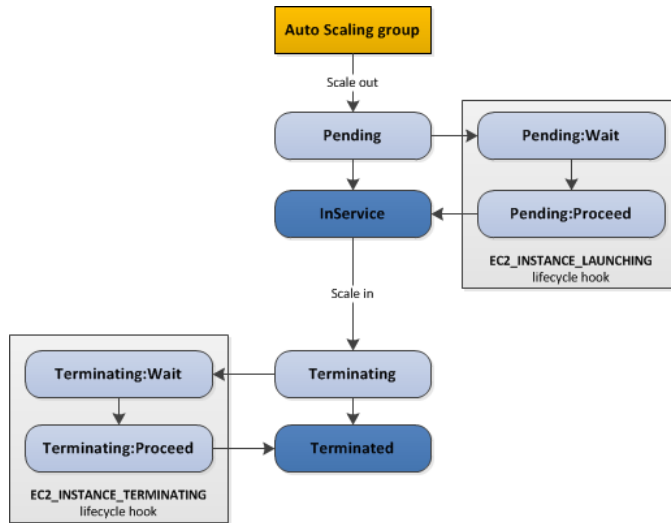
- [生命周期挂钩的工作方式 \(p. 83\)](#)
- [使用生命周期挂钩时的注意事项 \(p. 84\)](#)
- [准备通知 \(p. 85\)](#)
- [添加生命周期钩子 \(p. 87\)](#)
- [完成生命周期钩子 \(p. 87\)](#)
- [测试通知 \(p. 88\)](#)

生命周期挂钩的工作方式

在将生命周期挂钩添加到 Auto Scaling 组后，其工作方式如下：

1. Auto Scaling 通过启动实例来响应扩大事件，并通过终止实例来响应缩小事件。
2. Auto Scaling 将实例置于等待状态 (Pending:Wait 或 Terminating:Wait)。实例将保持此状态，直至您告知 Auto Scaling 继续或者超时时段结束。
3. 您可以使用下面的一个或多个选项来执行自定义操作：
 - 定义一个 CloudWatch Events 目标以在生命周期操作发生时调用 Lambda 函数。在 Auto Scaling 将生命周期操作的事件提交到 CloudWatch Events 时，将调用 Lambda 函数。该事件包含有关实例是正在启动还是正在终止的信息，以及可用于控制生命周期操作的令牌。
 - 为生命周期钩子定义一个通知目标。Auto Scaling 将向该通知目标发送消息。该消息包含有关实例是正在启动还是正在终止的信息，以及可用于控制生命周期操作的令牌。
 - 创建一个脚本，当实例启动时，该脚本将在实例上运行。该脚本可使用运行时所在的实例的 ID 来控制生命周期操作。
4. 默认情况下，实例将保持等待状态 1 小时，然后 Auto Scaling 继续启动或终止过程 (Pending:Proceed 或 Terminating:Proceed)。如果您需要更长时间，可通过记录检测信号来重新开始超时时段。如果您在超时时段结束前已完成此操作，则可以完成生命周期操作，从而继续启动或终止过程。

下图阐释了此过程中实例状态之间的过渡：



有关 Auto Scaling 组中实例的完整生命周期的更多信息，请参阅[Auto Scaling 生命周期 \(p. 7\)](#)。

使用生命周期挂钩时的注意事项

向 Auto Scaling 组添加生命周期挂钩，使您可以对实例的启动和终止方式进行更多控制。下面是在向 Auto Scaling 添加生命周期挂钩时为确保组按预期方式继续执行而要考虑的一些事项。

注意事项

- [使实例保持等待状态 \(p. 84\)](#)
- [冷却时间和自定义操作 \(p. 84\)](#)
- [运行状况检查宽限期 \(p. 85\)](#)
- [生命周期操作结果 \(p. 85\)](#)
- [竞价型实例 \(p. 85\)](#)

使实例保持等待状态

实例可以在有限的时间里保持等待状态。默认值为 1 小时（3600 秒钟）。您可以通过以下方式调整该时长：

- 在创建生命周期挂钩时，设置生命周期挂钩的检测信号超时。对 `put-lifecycle-hook` 命令使用 `--heartbeat-timeout` 参数。对 `PutLifecycleHook` 操作使用 `HeartbeatTimeout` 参数。
- 如果您在超时时段结束前已使用 `complete-lifecycle-action` 命令或 `CompleteLifecycleAction` 操作完成操作，请继续下一状态。
- 通过使用 `record-lifecycle-action-heartbeat` 命令或 `RecordLifecycleActionHeartbeat` 操作记录检测信号来重新开始超时时段。这会将检测信号超时增加您创建生命周期挂钩时指定的超时值。例如，如果超时值为 1 小时，而您在 30 分钟后调用该命令，则实例将继续保持等待状态 1 个小时（总共为 90 分钟）。

您可以将实例保持在等待状态的最长时间是 48 小时或检测信号超时的 100 倍（以较小者为准）。

冷却时间和自定义操作

当 Auto Scaling 启动简单扩展策略引起的实例时，[冷却时间 \(p. 75\)](#)生效。冷却时间可帮助确保 Auto Scaling 组只根据需要启动或终止合适数量的实例。

考虑在实例启动时使用包含支持自定义操作的声明周期挂钩的 Auto Scaling 组。当应用程序需求增加时，Auto Scaling 会启动实例以增加容量。由于存在生命周期挂钩，实例会置于 `Pending:Wait` 状态，这意味着该实例尚无法用于处理流量。当实例进入等待状态时，简单扩展策略引起的扩展操作会暂停。当实例进入 `InService` 状态时，冷却时间开始。冷却时间过后，所有暂停的扩展操作都会恢复。

运行状况检查宽限期

如果您通过添加生命周期挂钩在实例启动时执行操作，在完成生命周期挂钩并且实例进入 `InService` 状态后，健康检查宽限期才会启动。

生命周期操作结果

生命周期挂钩结束时，结果为 `ABANDON` 或 `CONTINUE`。

如果实例正在启动，则 `CONTINUE` 指示您的操作已成功，并且 Auto Scaling 可将实例投入使用。否则，`ABANDON` 指示您的自定义操作未成功，并且 Auto Scaling 可终止实例。

如果实例正在终止，则 `ABANDON` 和 `CONTINUE` 都允许终止实例。不过，`ABANDON` 将停止任何剩余操作（例如，其他生命周期挂钩），而 `CONTINUE` 将允许完成任何其他生命周期挂钩。

竞价型实例

您可以对竞价型实例使用生命周期挂钩。但是，生命周期挂钩不会阻止实例因竞价价格更改（随时可能发生）而终止。此外，在竞价型实例终止时，您仍需完成生命周期操作（使用 `complete-lifecycle-action` 命令或 `CompleteLifecycleAction` 操作）。

准备通知

您可以选择配置当实例进入等待状态时的通知，这能让您执行自定义操作。您可以使用 Amazon CloudWatch Events、Amazon SNS 或 Amazon SQS 接收通知。选择您喜好的选项。

或者，如果您拥有在实例启动时配置实例的脚本，则无需在生命周期操作发生时接收通知。如果您尚未这样做，请更新脚本，从实例元数据中检索实例的实例 ID。有关更多信息，请参阅[检索实例元数据](#)。

选项

- [使用 CloudWatch Events 接收通知 \(p. 85\)](#)
- [使用 Amazon SNS 接收通知 \(p. 86\)](#)
- [使用 Amazon SQS 接收通知 \(p. 87\)](#)

使用 CloudWatch Events 接收通知

您可以使用 CloudWatch Events 设置一个目标以在生命周期操作发生时调用 Lambda 函数。

使用 CloudWatch Events 设置通知

1. 使用[创建 Lambda 函数 \(p. 105\)](#)中的步骤创建 Lambda 函数，并记下其 Amazon 资源名称 (ARN)。例如：`arn:aws:lambda:us-west-2:123456789012:function:my-function`。
2. 使用以下 `put-rule` 命令创建与生命周期操作匹配的 CloudWatch Events 规则：

```
aws events put-rule --name my-rule --event-pattern file://pattern.json --state ENABLED
```

实例启动生命周期操作的 `pattern.json` 为：

```
{
  "source": [ "aws.autoscaling" ],
```

```
"detail-type": [ "EC2 Instance-launch Lifecycle Action" ]
}
```

实例终止生命周期操作的 pattern.json 为：

```
{
  "source": [ "aws.autoscaling" ],
  "detail-type": [ "EC2 Instance-terminate Lifecycle Action" ]
}
```

3. 使用以下 [add-permission](#) 命令授予调用 Lambda 函数的规则权限。此命令信任 CloudWatch Events 服务委托人 (events.amazonaws.com)，并将权限范围划定为指定规则。

```
aws lambda add-permission \
--function-name LogScheduledEvent \
--statement-id my-scheduled-event \
--action 'lambda:InvokeFunction' \
--principal events.amazonaws.com \
--source-arn arn:aws:events:us-east-1:123456789012:rule/my-scheduled-rule
```

4. 使用以下 [put-targets](#) 命令创建一个目标，此目标在生命周期操作发生时调用 Lambda 函数：

```
aws events put-targets --rule my-rule --targets Id=1,Arn=arn:aws:lambda:us-west-2:123456789012:function:my-function
```

5. 当 Auto Scaling 响应扩展或收缩事件时，它将实例置于等待状态。当实例处于等待状态时，Lambda 函数将被调用。有关事件数据的更多信息，请参阅 [Auto Scaling 事件](#) (p. 101)。

使用 Amazon SNS 接收通知

您可以使用 Amazon SNS 设置通知目标以便在生命周期操作开始时接收通知。

使用 Amazon SNS 设置通知

1. 使用 Amazon SNS 创建目标。有关更多信息，请参阅 Amazon Simple Notification Service 开发人员指南中的 [创建主题](#)。记下目标的 ARN（例如 arn:aws:sns:us-west-2:123456789012:my-sns-topic）。
2. 使用 IAM 用户指南中的 [创建向 AWS 服务委托权限的角色](#) 中的步骤，创建向 Auto Scaling 授予访问通知目标的权限的 IAM 角色。当系统提示您选择角色类型时，请选择 AWS Service Roles 和 AutoScaling Notification Access。记下角色的 ARN。例如：arn:aws:iam::123456789012:role/my-notification-role。
3. 当 Auto Scaling 响应扩展或收缩事件时，它将实例置于等待状态。当实例处于等待状态时，Auto Scaling 会向通知目标发布消息。消息包含以下事件数据：
 - LifecycleActionToken — 生命周期操作令牌。
 - AccountId — AWS 账户 ID。
 - AutoScalingGroupName — Auto Scaling 组的名称。
 - LifecycleHookName — 生命周期钩子的名称。
 - EC2InstanceId — EC2 实例的 ID。
 - LifecycleTransition — 生命周期钩子类型。

例如：

```
Service: AWS Auto Scaling
Time: 2016-09-30T20:42:11.305Z
RequestId: 18b2ec17-3e9b-4c15-8024-ff2e8ce8786a
```

```
LifecycleActionToken: 71514b9d-6a40-4b26-8523-05e7ee35fa40
AccountId: 123456789012
AutoScalingGroupName: my-asg
LifecycleHookName: my-hook
EC2InstanceId: i-0598c7d356eba48d7
LifecycleTransition: autoscaling:EC2_INSTANCE_LAUNCHING
NotificationMetadata: null
```

使用 Amazon SQS 接收通知

您可以使用 Amazon SQS 设置通知目标以便在生命周期操作开始时接收通知。

使用 Amazon SQS 设置通知

1. 使用 Amazon SQS 创建目标。有关更多信息，请参阅 Amazon Simple Queue Service 开发人员指南 中的 [Amazon SQS 入门](#)。记下目标的 ARN。
2. 使用 IAM 用户指南 中的 [创建向 AWS 服务委托权限的角色](#) 中的步骤，创建向 Auto Scaling 授予访问通知目标的权限的 IAM 角色。当系统提示您选择角色类型时，请选择 AWS Service Roles 和 AutoScaling Notification Access。记下角色的 ARN。例如：`arn:aws:iam::123456789012:role/my-notification-role`。
3. 当 Auto Scaling 响应扩展或收缩事件时，它将实例置于等待状态。当实例处于等待状态时，Auto Scaling 会向通知目标发布消息。

添加生命周期钩子

您可以使用 `put-lifecycle-hook` 命令创建生命周期挂钩。

要执行扩大操作，请使用以下命令：

```
aws autoscaling put-lifecycle-hook --lifecycle-hook-name my-hook --auto-scaling-group-  
name my-asg \  
--lifecycle-transition autoscaling:EC2_INSTANCE_LAUNCHING
```

要执行缩小操作，则请使用以下命令：

```
aws autoscaling put-lifecycle-hook --lifecycle-hook-name my-hook --auto-scaling-group-  
name my-asg \  
--lifecycle-transition autoscaling:EC2_INSTANCE_TERMINATING
```

(可选) 要使用 Amazon SNS 或 Amazon SQS 接收通知，可使用其他选项。例如，添加以下选项可将 SNS 主题指定为通知目标：

```
--notification-target-arn arn:aws:sns:us-west-2:123456789012:my-sns-topic \  
--role-arn arn:aws:iam::123456789012:role/my-notification-role
```

Auto Scaling 将向使用以下键/值对的主题发送测试通知：

```
"Event": "autoscaling:TEST_NOTIFICATION"
```

完成生命周期钩子

当 Auto Scaling 响应扩大或缩小事件时，它会将实例置于等待状态并发送任意通知。在您完成生命周期钩子后，Auto Scaling 会继续启动或终止过程。

完成周期钩子

1. 当实例处于等待状态时，您可以执行自定义操作。有关更多信息，请参阅 [准备通知 \(p. 85\)](#)。
2. (可选) 如果需要更多时间来完成自定义操作，请使用 [record-lifecycle-action-heartbeat](#) 命令重新启动超时时段并将实例保持等待状态。您可以指定在上一步中接收的生命周期操作令牌，如以下命令所示：

```
aws autoscaling record-lifecycle-action-heartbeat --lifecycle-action-token bcd2f1b8-9a78-44d3-8a7a-4dd07d7cf635 --lifecycle-hook-name my-launch-hook --auto-scaling-group-name my-asg
```

或者，可以指定在上一步中接收的实例的 ID，如以下命令所示：

```
aws autoscaling record-lifecycle-action-heartbeat --instance-id i-1a2b3c4d --lifecycle-hook-name my-launch-hook --auto-scaling-group-name my-asg
```

3. 如果您在超时时段结束前已完成自定义操作，请使用 [complete-lifecycle-action](#) 命令，以便 Auto Scaling 能够继续启动或终止实例。您可以指定生命周期操作令牌，如以下命令所示：

```
aws autoscaling complete-lifecycle-action --lifecycle-action-result CONTINUE --lifecycle-action-token bcd2f1b8-9a78-44d3-8a7a-4dd07d7cf635 --lifecycle-hook-name my-launch-hook --auto-scaling-group-name my-asg
```

或者，您可以指定实例的 ID，如以下命令所示：

```
aws autoscaling complete-lifecycle-action --lifecycle-action-result CONTINUE --instance-id i-1a2b3c4d --lifecycle-hook-name my-launch-hook --auto-scaling-group-name my-asg
```

测试通知

要为启动事件生成通知，请通过将 Auto Scaling 组的所需容量增加 1 来更新 Auto Scaling 组。Auto Scaling 会启动 EC2 实例，您将在数分钟内收到通知。

使用控制台更改所需容量

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择您的 Auto Scaling 组。
4. 在 Details 选项卡上，选择 Edit。
5. 对于 Desired，将当前值增加 1。如果此值超过 Max，还必须将 Max 的值增加 1。
6. 选择 Save。
7. 在数分钟后，您将收到事件的通知。如果您不需要您为此测试启动的其他实例，则可以将 Desired 减少 1。在数分钟后，您将收到事件的通知。

临时从 Auto Scaling 组中删除实例

Auto Scaling 使您能够将处于 InService 状态的实例置于 Standby 状态，更新实例或排查实例的问题，然后将实例返回运行状态。处于备用状态的实例仍是 Auto Scaling 组的一部分，但它们不会主动处理应用程序流量。

Important

您需为处于备用状态的实例付费。

例如，您可以随时更改 Auto Scaling 组的启动配置，Auto Scaling 组启动的任何后续实例将使用此配置。不过，Auto Scaling 组不会更新当前正在运行的实例。您可以终止这些实例并让 Auto Scaling 组替换这些实例，也可以将其置于备用状态，更新软件，然后将实例重新置于运行状态。

内容

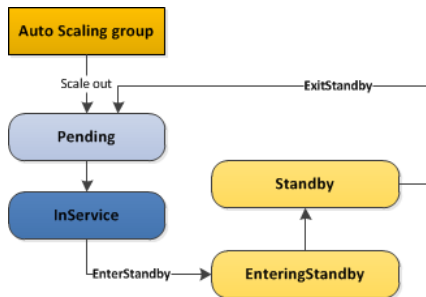
- [备用状态的工作方式 \(p. 89\)](#)
- [处于备用状态的实例的运行状况 \(p. 89\)](#)
- [使用 AWS 管理控制台临时删除实例 \(p. 90\)](#)
- [使用 AWS CLI 临时删除实例 \(p. 90\)](#)

备用状态的工作方式

备用状态按如下方式工作以帮助您临时从 Auto Scaling 组中删除实例：

1. 将实例置于备用状态。实例保持此状态，直至您退出备用状态。
2. 如果已向 Auto Scaling 组附加负载均衡器或目标组，则将从该负载均衡器或目标组取消注册实例。
3. 默认情况下，在将实例置于备用状态时，Auto Scaling 会减少 Auto Scaling 组的所需容量。这可防止 Auto Scaling 在您将此实例置于备用状态时启动附加实例。或者，您可以指定 Auto Scaling 不减少容量。这将导致 Auto Scaling 启动附加实例来替换处于备用状态的实例。
4. 您可以更新实例或排查实例的问题。
5. 可通过退出备用状态，将实例恢复运行状态。
6. 在将处于备用状态的实例恢复运行时，Auto Scaling 会自动增大所需容量。如果您在将实例置于备用状态时未减小容量，则 Auto Scaling 会检测到您拥有的实例数多于所需的实例数，并应用终止策略来减小 Auto Scaling 组的大小。有关更多信息，请参阅 [控制 Auto Scaling 在缩小过程中终止哪些实例 \(p. 78\)](#)。
7. 如果已向 Auto Scaling 组附加负载均衡器或目标组，则会将实例注册到该负载均衡器或目标组。

下图阐释了此过程中实例状态之间的过渡：



有关 Auto Scaling 组中实例的完整生命周期的更多信息，请参阅 [Auto Scaling 生命周期 \(p. 7\)](#)。

处于备用状态的实例的运行状况

Auto Scaling 不对处于备用状态的实例执行运行状况检查。当实例处于备用状态时，其运行状况将反映您将其置于备用状态之前，实例具有的状态。Auto Scaling 不对实例执行运行状况检查，直至您将实例恢复运行。

例如，如果您将运行正常的实例置于备用状态，然后终止该实例，则 Auto Scaling 会继续将实例报告为运行正常。如果您将终止的实例恢复运行，则 Auto Scaling 将对实例执行运行状况检查，确定它运行状况不佳，并启动替换实例。

使用 AWS 管理控制台临时删除实例

以下步骤演示更新当前正在运行的实例的一般过程。

使用控制台临时删除实例

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择 Auto Scaling 组。
4. 在 Instances 选项卡上，选择实例。
5. 选择 Actions 和 Set to Standby。
6. 在 Set to Standby 页面上，选中该复选框可让 Auto Scaling 启动替代实例，取消选中可减少所需容量。选择 Set to Standby。
7. 您可以根据需要更新实例或排查实例的问题。当您完成后，请继续下一步以将实例恢复运行。
8. 选择实例，再选择 Actions 和 Set to InService。在 Set to InService 页面上，选择 Set to InService。

使用 AWS CLI 临时删除实例

以下步骤演示更新当前正在运行的实例的一般过程。

使用 AWS CLI 临时删除实例

1. 使用下面的 `describe-auto-scaling-instances` 命令确定要更新的实例：

```
aws autoscaling describe-auto-scaling-instances
```

以下为响应示例：

```
{
  "AutoScalingInstances": [
    {
      "AvailabilityZone": "us-west-2a",
      "InstanceId": "i-5b73d709",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService",
      "LaunchConfigurationName": "my-lc"
    },
    ...
  ]
}
```

2. 使用以下 `enter-standby` 命令将实例置于 Standby 状态。`--should-decrement-desired-capacity` 选项将减少所需容量以使 Auto Scaling 不再启动替换实例。

```
aws autoscaling enter-standby --instance-ids i-5b73d709 --auto-scaling-group-name my-asg --should-decrement-desired-capacity
```

以下为响应示例：

```
{
  "Activities": [
    {
      "Description": "Moving EC2 instance to Standby: i-5b73d709",

```



```
    "AutoScalingGroupName": "my-asg",
    "ActivityId": "3b1839fe-24b0-40d9-80ae-bcd883c2be32",
    "Details": "{\"Availability Zone\":\"us-west-2a\"}",
    "StartTime": "2014-12-15T21:31:26.150Z",
    "Progress": 50,
    "Cause": "At 2014-12-15T21:31:26Z instance i-5b73d709 was moved to standby
in response to a user request, shrinking the capacity from 4 to 3.",
    "StatusCode": "InProgress"
  }
]
```

3. (可选) 使用以下 `describe-auto-scaling-instances` 命令验证实例是否处于 Standby 状态：

```
aws autoscaling describe-auto-scaling-instances --instance-ids i-5b73d709
```

以下是一项示范响应。注意，实例状态此时为 Standby。

```
{
  "AutoScalingInstances": [
    {
      "AvailabilityZone": "us-west-2a",
      "InstanceId": "i-5b73d709",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "Standby",
      "LaunchConfigurationName": "my-lc"
    }
  ]
}
```

4. 您可以根据需要更新实例或排查实例的问题。当您完成后，请继续下一步以将实例恢复运行。
5. 使用以下 `exit-standby` 命令使实例恢复运行：

```
aws autoscaling exit-standby --instance-ids i-5b73d709 --auto-scaling-group-name my-asg
```

以下为响应示例：

```
{
  "Activities": [
    {
      "Description": "Moving EC2 instance out of Standby: i-5b73d709",
      "AutoScalingGroupName": "my-asg",
      "ActivityId": "db12b166-cdcc-4c54-8aac-08c5935f8389",
      "Details": "{\"Availability Zone\":\"us-west-2a\"}",
      "StartTime": "2014-12-15T21:46:14.678Z",
      "Progress": 30,
      "Cause": "At 2014-12-15T21:46:14Z instance i-5b73d709 was moved out of
standby in
      response to a user request, increasing the capacity from 3 to 4.",
      "StatusCode": "PreInService"
    }
  ]
}
```

6. (可选) 使用以下 `describe-auto-scaling-instances` 命令验证实例是否已恢复运行。

```
aws autoscaling describe-auto-scaling-instances --instance-ids i-5b73d709
```

以下是一项示范响应。请注意，实例状态为 InService。

```
{
  "AutoScalingInstances": [
    {
      "AvailabilityZone": "us-west-2a",
      "InstanceId": "i-5b73d709",
      "AutoScalingGroupName": "my-asg",
      "HealthStatus": "HEALTHY",
      "LifecycleState": "InService",
      "LaunchConfigurationName": "my-lc"
    }
  ]
}
```

暂停和恢复 Auto Scaling 流程

Auto Scaling 允许您暂停然后恢复 Auto Scaling 组中的一个或多个 Auto Scaling 流程。如果需要调查配置问题或与 Web 应用程序相关的其他问题，然后在不触发 Auto Scaling 流程的前提下对应用程序进行更改，则此设置很有用。

Auto Scaling 可能会暂停多次启动实例失败的 Auto Scaling 组的流程。这称为管理暂停，最常用于符合以下条件的 Auto Scaling 组：连续尝试启动实例的时间超过 24 小时，但是未成功启动任何实例。您可以恢复由于管理原因暂停的流程。

内容

- [Auto Scaling 流程 \(p. 92\)](#)
- [使用控制台暂停和恢复流程 \(p. 93\)](#)
- [使用 AWS CLI 暂停和恢复流程 \(p. 94\)](#)

Auto Scaling 流程

Auto Scaling 支持以下流程：

Launch

将新的 EC2 实例添加到组，从而增加组的容量。

Warning

如果您暂停 Launch，这会中断其他流程。例如，如果暂停 Launch 流程，则无法使处于备用状态的实例恢复运行，因为组无法扩展。

Terminate

从组中删除 EC2 实例，从而减少组的容量。

Warning

如果您暂停 Terminate，这会中断其他流程。

HealthCheck

检查实例的运行状况。如果 Amazon EC2 或 Elastic Load Balancing 通知 Auto Scaling 实例运行状况不佳，Auto Scaling 会将该实例标记为运行状况不佳。此流程可覆盖您手动设置的实例运行状况状态。

ReplaceUnhealthy

终止被标记为运行状况不佳的实例，然后创建新实例将其替换。此流程与 HealthCheck 流程结合使用，并使用 Terminate 和 Launch 流程。

AZRebalance

在区域内的可用区之间均衡组中 EC2 实例的数量。如果从 Auto Scaling 组中删除可用区，或可用区运行状况不佳或无法使用，Auto Scaling 在终止运行状况不佳或无法使用的实例前，在不受影响的可用区中启动新实例。当运行状况不佳的可用区恢复正常状态时，Auto Scaling 自动在组的可用区中重新均匀分布实例。有关更多信息，请参阅 [再平衡活动 \(p. 6\)](#)。

如果暂停 AZRebalance 并且发生了扩展或收缩事件，Auto Scaling 仍会尝试均衡可用区。例如，在扩展期间，Auto Scaling 会在实例最少的可用区中启动实例。

如果您暂停 Launch 流程，AZRebalance 不会启动新实例，也不会终止现有实例。这是因为 AZRebalance 只在启动替换实例后才终止实例。如果您暂停 Terminate 流程，Auto Scaling 组容量可以增加超出最大大小百分之十，因为 Auto Scaling 允许这种情况在重新均衡活动期间临时发生。如果 Auto Scaling 不能终止实例，Auto Scaling 组可以保持超出其最大大小，直到您恢复 Terminate 流程。

AlarmNotification

接受来自与组关联的 CloudWatch 警报的通知。

如果您暂停 AlarmNotification，Auto Scaling 不会自动执行会被警报触发的策略。如果您暂停 Launch 或 Terminate，Auto Scaling 将分别不能执行扩大或缩小策略。

ScheduledActions

执行您创建的计划操作。

如果您暂停 Launch 或 Terminate，涉及启动或终止实例的计划操作会受到影响。

AddToLoadBalancer

在实例启动时，将其添加到附加的负载均衡器或目标组。

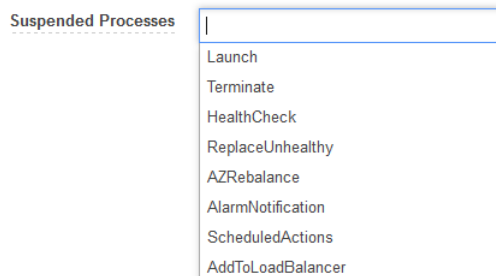
如果您暂停 AddToLoadBalancer，Auto Scaling 会启动实例，但不会将其添加到负载均衡器或目标组。如果您恢复 AddToLoadBalancer 流程，Auto Scaling 也会在启动实例时将其添加到负载均衡器或目标组。不过，Auto Scaling 不会添加在此流程暂停时启动的实例。您必须手动注册这些实例。

使用控制台暂停和恢复流程

您可以使用 AWS 管理控制台暂停和恢复单个流程。

使用控制台暂停和恢复流程

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择 Auto Scaling 组。
4. 在 Details 选项卡上，选择 Edit。
5. 对于 Suspended Processes，选择要暂停的流程。



要恢复已暂停的流程，请从 Suspended Processes 中删除该流程。

Suspended Processes

AlarmNotification x

6. 选择 Save。

使用 AWS CLI 暂停和恢复流程

您可以暂停和恢复单个流程或所有流程。

暂停一个流程

使用 `--scaling-processes` 选项使用 `suspend-processes` 命令，如下所示：

```
aws autoscaling suspend-processes --auto-scaling-group-name my-asg --scaling-processes AlarmNotification
```

暂停所有流程

使用 `suspend-processes` 命令（省略 `--scaling-processes` 选项），如下所示：

```
aws autoscaling suspend-processes --auto-scaling-group-name my-asg
```

恢复一个暂停的流程

使用 `resume-processes` 命令，如下所示：

```
aws autoscaling resume-processes --auto-scaling-group-name my-asg --scaling-processes AlarmNotification
```

恢复所有暂停的流程

使用 `resume-processes` 命令（省略 `--scaling-processes` 选项），如下所示：

```
aws autoscaling resume-processes --auto-scaling-group-name my-asg
```

监控您的 Auto Scaling 实例和组

您可以使用以下功能监控您的 Auto Scaling 实例和组。

运行状况检查

Auto Scaling 定期对 Auto Scaling 组中的实例执行运行状况检查并标识所有运行状况不佳的实例。可将 Auto Scaling 配置为使用 Amazon EC2 状态检查、Elastic Load Balancing 运行状况检查或自定义运行状况检查来确定实例的运行状况。有关更多信息，请参阅 [Auto Scaling 实例的运行状况检查 \(p. 95\)](#)。

CloudWatch 指标

Auto Scaling 将数据点发布到有关 Auto Scaling 组的 Amazon CloudWatch。利用 CloudWatch，您可以按一组有序的时间序列数据 (称为指标) 来检索关于这些数据点的统计数据。您可使用这些指标来验证您的系统是否按预期运行。有关更多信息，请参阅 [使用 Amazon CloudWatch 监控 Auto Scaling 组和实例 \(p. 97\)](#)。

CloudWatch Events

当您的 Auto Scaling 组启动或终止实例时或当生命周期操作发生时，Auto Scaling 会将事件提交到 Amazon CloudWatch Events 组。这使您能够在事件发生时调用 Lambda 函数。有关更多信息，请参阅 [在 Auto Scaling 组扩展时获取 CloudWatch Events \(p. 101\)](#)。

SNS 通知

当您的 Auto Scaling 组启动或终止实例时，Auto Scaling 会发送 Amazon SNS 通知。有关更多信息，请参阅 [在 Auto Scaling 组扩展时获取 SNS 通知 \(p. 106\)](#)。

CloudTrail 日志

AWS CloudTrail 可让您跟踪由您的 AWS 账户或代表您的 AWS 账户对 Auto Scaling API 进行的调用。CloudTrail 将信息存储在您指定的 Amazon S3 存储桶中的日志文件中。利用这些日志文件，您可以通过确定已发出的请求、请求的源 IP 地址、请求的发出方、请求的发出时间等来监控 Auto Scaling 组的活动。有关更多信息，请参阅 [使用 AWS CloudTrail 记录 Auto Scaling API 调用 \(p. 110\)](#)。

Auto Scaling 实例的运行状况检查

Auto Scaling 实例的运行状况有正常和不佳两种。一个实例在经过完全配置并通过初始运行状况检查后，将被 Auto Scaling 视为正常并进入 InService 状态。Auto Scaling 定期对 Auto Scaling 组中的实例执行运行状况检查并标识所有运行状况不佳的实例。Auto Scaling 在将一个实例标记为运行状况不佳后，会计划替换该实例。有关更多信息，请参阅 [替换运行状况不佳的实例 \(p. 53\)](#)。

实例运行状况

Auto Scaling 使用下列一项或多项检查来确定实例的运行状况：

- Amazon EC2 提供的状态检查 (系统状态检查和实例状态检查)。有关更多信息，请参阅 Amazon EC2 用户指南 (适用于 Linux 实例) 中的[实例的状态检查](#)。
- 由 Elastic Load Balancing 提供的运行状况检查。有关更多信息，请参阅 应用程序负载均衡器 指南 中的[目标组的运行状况检查](#)或 传统负载均衡器 指南 中的[为您的传统负载均衡器配置运行状况检查](#)。
- 自定义运行状况检查。

默认情况下，Auto Scaling 运行状况检查使用 EC2 状态检查的结果来确定实例的运行状况。如果一个实例未通过一个或多个状态检查，Auto Scaling 会将该实例标记为运行状况不佳。

如果您已将一个负载均衡器或目标组附加到 Auto Scaling 组，则可对 Auto Scaling 进行配置，让它在 Elastic Load Balancing 报告某个实例 `OutOfService` 时将该实例标记为运行状况不佳。如果为负载均衡器启用了连接耗尽，则在由于扩展事件或运行状况检查替换而终止实例之前，Auto Scaling 会等待进行中的请求完成或等待最大超时过期 (以先到者为准)。有关更多信息，请参阅[结合使用 ELB 运行状况检查和 Auto Scaling \(p. 45\)](#)。

运行状况检查宽限期

通常，刚刚投入使用的 Auto Scaling 实例需要预热才能通过 Auto Scaling 运行状况检查。Auto Scaling 等运行状况检查宽限期结束才检查实例的运行状况。EC2 状态检查和 ELB 运行状况检查可以在运行状况检查宽限期过期前完成，但 Auto Scaling 直到运行状况检查宽限期过期后才执行这些检查。为了给实例提供足够的预热时间，请确保运行状况检查宽限期包含应用程序的预期启动时间。请注意，如果您通过添加生命周期挂钩在实例启动时执行操作，在完成生命周期挂钩并且实例进入 `InService` 状态后，运行状况检查宽限期才会启动。

自定义运行状况检查

如果您有自定义运行状况检查，则可以将运行状况检查的信息发送给 Auto Scaling，以便 Auto Scaling 能够利用此信息。例如，如果您确定实例未按预期运行，则可将实例的运行状况设置为 `Unhealthy`。Auto Scaling 下次对实例执行运行状况检查时，它将确定实例运行状况不佳，然后启动替换实例。

使用以下 `set-instance-health` 命令，将指定实例的运行状态设置为 `Unhealthy`：

```
aws autoscaling set-instance-health --instance-id i-123abc45d --health-status Unhealthy
```

使用以下 `describe-auto-scaling-groups` 命令验证实例状态是否为 `Unhealthy`：

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-names my-asg
```

下面是一个示例响应，说明实例的运行状况为 `Unhealthy` 并且实例正在终止：

```
{
  "AutoScalingGroups": [
    {
      ...
      "Instances": [
        {
          "InstanceId": "i-123abc45d",
          "AvailabilityZone": "us-west-2a",
          "HealthStatus": "Unhealthy",
          "LifecycleState": "Terminating",
          "LaunchConfigurationName": "my-lc"
        }
      ]
    }
  ]
}
```

```
    },  
    ...  
  ]  
}  
]
```

使用 Amazon CloudWatch 监控 Auto Scaling 组和实例

利用 Amazon CloudWatch，您可以按一组有序的时间序列数据（称为指标）来检索统计数据。您可使用这些指标来验证您的系统是否按预期运行。

Amazon EC2 将向 CloudWatch 发送描述 Auto Scaling 实例的指标。这些指标对任何 EC2 实例（而不仅仅是 Auto Scaling 组中的实例）可用。有关更多信息，请参阅 Amazon EC2 用户指南（适用于 Linux 实例）中的[实例指标](#)。

Auto Scaling 组可以向 CloudWatch 发送描述该组本身的指标。您必须启用这些指标。

内容

- [Auto Scaling 组指标 \(p. 97\)](#)
- [Auto Scaling 组指标的维度 \(p. 98\)](#)
- [启用 Auto Scaling 组指标 \(p. 98\)](#)
- [为 Auto Scaling 实例配置监控 \(p. 99\)](#)
- [查看 CloudWatch 指标 \(p. 99\)](#)
- [创建 Amazon CloudWatch 警报 \(p. 100\)](#)

Auto Scaling 组指标

AWS/AutoScaling 命名空间包括以下指标。

指标	说明
GroupMinSize	Auto Scaling 组的最小大小。
GroupMaxSize	Auto Scaling 组的最大大小。
GroupDesiredCapacity	Auto Scaling 组试图维护的实例数量。
GroupInServiceInstances	作为 Auto Scaling 组的一部分运行的实例数量。该指标不包括处于挂起或终止状态的实例。
GroupPendingInstances	处于挂起状态的实例数量。挂起的实例尚不可用。该指标不包括处于可用状态或终止状态的实例。
GroupStandbyInstances	处于 Standby 状态的实例数。处于此状态的实例仍在运行，但不能有效使用。
GroupTerminatingInstances	正处于终止过程中的实例的数量。该指标不包括处于可用状态或挂起状态的实例。
GroupTotalInstances	Auto Scaling 组中的实例总数。该指标用于标识处于可用状态、挂起状态和终止状态的实例的数量。

Auto Scaling 组指标的维度

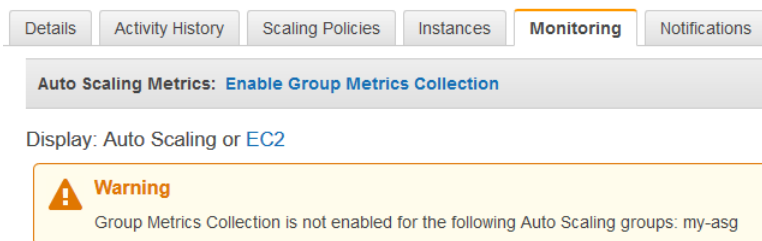
要按组名称筛选 Auto Scaling 组的指标，请使用 `AutoScalingGroupName` 维度。

启用 Auto Scaling 组指标

启用 Auto Scaling 组指标后，Auto Scaling 每分钟向 CloudWatch 发送采样数据。

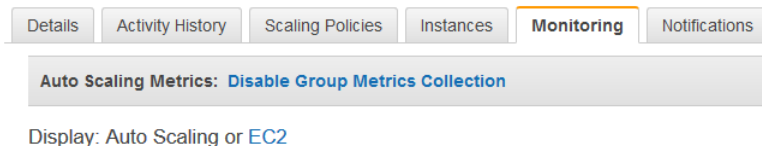
使用控制台启用组指标

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格中，选择 Auto Scaling Groups。
3. 选择您的 Auto Scaling 组。
4. 在 Monitoring 选项卡上，为 Auto Scaling Metrics 选择 Enable Group Metrics Collection。如果您看不到此选项，请为 Display 选择 Auto Scaling。



使用控制台禁用组指标

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格中，选择 Auto Scaling Groups。
3. 选择您的 Auto Scaling 组。
4. 在 Monitoring 选项卡上，为 Auto Scaling Metrics 选择 Disable Group Metrics Collection。如果您看不到此选项，请为 Display 选择 Auto Scaling。



使用 AWS CLI 启用组指标

使用 `enable-metrics-collection` 命令启用一个或多个组指标。例如，以下命令可启用 `GroupDesiredCapacity` 指标。

```
aws autoscaling enable-metrics-collection --auto-scaling-group-name my-asg --metrics GroupDesiredCapacity --granularity "1Minute"
```

如果您忽略 `--metrics` 选项，则所有指标都会启用。

```
aws autoscaling enable-metrics-collection --auto-scaling-group-name my-asg --granularity "1Minute"
```

使用 AWS CLI 禁用组指标

请使用 `disable-metrics-collection` 命令。例如，以下命令可禁用所有 Auto Scaling 组指标：

```
aws autoscaling disable-metrics-collection --auto-scaling-group-name my-asg
```

为 Auto Scaling 实例配置监控

默认情况下，在使用 AWS 管理控制台创建启动配置时启用基本监控，在使用 AWS CLI 或 API 创建启动配置时启用详细监控。

如果您拥有 Auto Scaling 组并需要更改为 Auto Scaling 实例启用的监控类型，则必须创建新的启动配置并更新 Auto Scaling 组以使用该启动配置。之后，Auto Scaling 组启动的实例将使用更新后的监控类型。请注意，Auto Scaling 组中的现有实例将继续使用上一个监控类型。您可以终止这些实例以便 Auto Scaling 更换它们，也可以使用 `monitor-instances` 和 `unmonitor-instances` 逐个更新每个实例。

如果您具有与您的 Auto Scaling 组关联的 CloudWatch 警报，请使用 `put-metric-alarm` 命令更新每个警报，以便其期间与监控类型匹配（对于基本监控，为 300 秒；对于详细监控，为 60 秒）。如果您从详细监控更改为基本监控，但不将警报更新为匹配 5 分钟时间段，警报将继续每分钟检查统计数据并可能每 5 分钟有 4 分钟找不到任何可用的数据。

使用控制台配置 CloudWatch 监控

在使用 AWS 管理控制台创建启动配置时，在 Configure Details 页上，选择 Enable CloudWatch detailed monitoring。否则，将启用基本监控。有关更多信息，请参阅 [创建启动配置 \(p. 19\)](#)。

使用 AWS CLI 配置 CloudWatch 监控

将 `create-launch-configuration` 命令与 `--instance-monitoring` 选项结合使用。将此选项设置为 `true` 可启用详细监控，将此选项设置为 `false` 可启用基本监控。

```
--instance-monitoring Enabled=true
```

查看 CloudWatch 指标

您可以使用 Amazon EC2 控制台查看 Auto Scaling 组和实例的 CloudWatch 指标。这些指标显示为监控图表。

或者，您可以使用 CloudWatch 控制台查看这些指标。

使用 Amazon EC2 控制台查看指标

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格中，选择 Auto Scaling Groups。
3. 选择您的 Auto Scaling 组。
4. 选择 Monitoring 选项卡。
5. (可选) 要按时间筛选结果，请从 Showing data for 中选择时间范围。
6. 要查看组的指标，请为 Display 选择 Auto Scaling。要获得单个指标的一个较大视图，请选择其图形。以下指标可用于组：

- 最小组大小 — `GroupMinSize`
- 最大组大小 — `GroupMaxSize`
- 所需容量 — `GroupDesiredCapacity`
- 在服务实例中 — `GroupInServiceInstances`
- 待处理实例 — `GroupPendingInstances`

- 备用实例 — GroupStandbyInstances
 - 终止实例 — GroupTerminatingInstances
 - 实例总数 — GroupTotalInstances
7. 要查看实例的指标，请为 Display 选择 EC2。要获得单个指标的一个较大视图，请选择其图形。以下指标可用于实例：
- CPU 利用率 — CPUUtilization
 - 磁盘读取 — DiskReadBytes
 - 读磁盘操作 — DiskReadOps
 - 磁盘写入 — DiskWriteBytes
 - 写磁盘操作 — DiskWriteOps
 - 网络输入 — NetworkIn
 - 网络输出 — NetworkOut
 - 状态检查失败(任意) — StatusCheckFailed
 - 状态检查失败(实例) — StatusCheckFailed_Instance
 - 状态检查失败(系统) — StatusCheckFailed_System

使用 CloudWatch 控制台查看指标

有关更多信息，请参阅[按 Auto Scaling 组聚合统计数据](#)。

使用 AWS CLI 查看 CloudWatch 指标

要查看所有 Auto Scaling 组的所有指标，请使用以下 `list-metrics` 命令：

```
aws cloudwatch list-metrics --namespace "AWS/AutoScaling"
```

要查看单个 Auto Scaling 组的指标，请指定 AutoScalingGroupName 维度，如下所示：

```
aws cloudwatch list-metrics --namespace "AWS/AutoScaling" --dimensions  
Name=AutoScalingGroupName,Value=my-asg
```

要查看适用于所有 Auto Scaling 组的单个指标，请指定该指标的名称，如下所示：

```
aws cloudwatch list-metrics --namespace "AWS/AutoScaling" --metric-name  
GroupDesiredCapacity
```

创建 Amazon CloudWatch 警报

CloudWatch 警报 是指在特定时间段监控单个指标的对象。指标是指要监控的变量，如 EC2 实例的平均 CPU 利用率，或者从许多不同的 EC2 实例传入的网络流量。如果指标值违反定义的范围，并在指定周期数内保持该变动，警报会自动更改状态。

警报中包含三个状态：

- OK – 指标值保持在您指定的范围内。
- ALARM – 指标值在指定时间段内超出您指定的范围。
- INSUFFICIENT_DATA – 指标不可用或指标数据不足以判断警报状态。

如果警报更改为 ALARM 状态，并在该状态保持较长时间，它将调用一项或多项操作。这些操作可能是发送到 Auto Scaling 组以更改该组的所需容量的消息。

可以通过标识要监控的指标来配置警报。例如，您可以配置警报来监控 Auto Scaling 组中 EC2 实例的平均 CPU 使用率。

创建 CloudWatch 警报

1. 通过以下网址打开 CloudWatch 控制台：<https://console.amazonaws.cn/cloudwatch/>。
2. 在导航窗格上，选择 Alarms。
3. 选择 Create Alarm。
4. 选择 EC2 Metrics 类别。
5. （可选）您可以筛选结果。要查看实例指标，请选择 Per-Instance Metrics。要查看 Auto Scaling 组指标，请选择 By Auto Scaling Group。
6. 选择一个指标，然后选择 Next。
7. 指定警报的阈值和要执行的操作。

有关更多信息，请参阅 Amazon CloudWatch 用户指南 中的 [创建 CloudWatch 警报](#)。

8. 选择 Create Alarm。

在 Auto Scaling 组扩展时获取 CloudWatch Events

使用 Auto Scaling 自动扩展应用程序时，知道 Auto Scaling 何时启动或终止 Auto Scaling 组中的 EC2 实例会很有用。可以将 Auto Scaling 配置为在 Auto Scaling 组扩展时将事件发送到 Amazon CloudWatch Events。

有关更多信息，请参阅 [Amazon CloudWatch Events 用户指南](#)。

内容

- [Auto Scaling 事件](#) (p. 101)
- [创建 Lambda 函数](#) (p. 105)
- [将事件路由到 Lambda 函数](#) (p. 105)

Auto Scaling 事件

Auto Scaling 支持在发生以下事件时向 CloudWatch Events 发送通知：

- [EC2 实例启动生命周期操作](#) (p. 101)
- [EC2 实例启动成功](#) (p. 102)
- [EC2 实例启动失败](#) (p. 102)
- [EC2 实例终止生命周期操作](#) (p. 103)
- [EC2 实例终止成功](#) (p. 103)
- [EC2 实例终止失败](#) (p. 104)

EC2 实例启动生命周期操作

由于生命周期挂钩，Auto Scaling 已将实例移至 Pending:Wait 状态。

事件数据

以下是此事件的示例数据。

```
{
```

```
"version": "0",
"id": "12345678-1234-1234-1234-123456789012",
"detail-type": "EC2 Instance-launch Lifecycle Action",
"source": "aws.autoscaling",
"account": "123456789012",
"time": "yyyy-mm-ddThh:mm:ssZ",
"region": "us-west-2",
"resources": [
  "auto-scaling-group-arn"
],
"detail": {
  "LifecycleActionToken": "87654321-4321-4321-4321-210987654321",
  "AutoScalingGroupName": "my-asg",
  "LifecycleHookName": "my-lifecycle-hook",
  "EC2InstanceId": "i-12345678",
  "LifecycleTransition": "autoscaling:EC2_INSTANCE_LAUNCHING"
}
}
```

EC2 实例启动成功

Auto Scaling 已成功启动实例。

事件数据

以下是此事件的示例数据。

```
{
  "version": "0",
  "id": "12345678-1234-1234-1234-123456789012",
  "detail-type": "EC2 Instance Launch Successful",
  "source": "aws.autoscaling",
  "account": "123456789012",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-west-2",
  "resources": [
    "auto-scaling-group-arn",
    "instance-arn"
  ],
  "detail": {
    "StatusCode": "InProgress",
    "Description": "Launching a new EC2 instance: i-12345678",
    "AutoScalingGroupName": "my-auto-scaling-group",
    "ActivityId": "87654321-4321-4321-4321-210987654321",
    "Details": {
      "Availability Zone": "us-west-2b",
      "Subnet ID": "subnet-12345678"
    },
    "RequestId": "12345678-1234-1234-1234-123456789012",
    "StatusMessage": "",
    "EndTime": "yyyy-mm-ddThh:mm:ssZ",
    "EC2InstanceId": "i-12345678",
    "StartTime": "yyyy-mm-ddThh:mm:ssZ",
    "Cause": "description-text",
  }
}
```

EC2 实例启动失败

Auto Scaling 未能启动实例。

事件数据

以下是此事件的示例数据。

```
{
  "version": "0",
  "id": "12345678-1234-1234-1234-123456789012",
  "detail-type": "EC2 Instance Launch Unsuccessful",
  "source": "aws.autoscaling",
  "account": "123456789012",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-west-2",
  "resources": [
    "auto-scaling-group-arn",
    "instance-arn"
  ],
  "detail": {
    "StatusCode": "Failed",
    "AutoScalingGroupName": "my-auto-scaling-group",
    "ActivityId": "87654321-4321-4321-4321-210987654321",
    "Details": {
      "Availability Zone": "us-west-2b",
      "Subnet ID": "subnet-12345678"
    },
    "RequestId": "12345678-1234-1234-1234-123456789012",
    "StatusMessage": "message-text",
    "EndTime": "yyyy-mm-ddThh:mm:ssZ",
    "EC2InstanceId": "i-12345678",
    "StartTime": "yyyy-mm-ddThh:mm:ssZ",
    "Cause": "description-text",
  }
}
```

EC2 实例终止生命周期操作

由于生命周期挂钩，Auto Scaling 已将实例移至 Terminating:Wait 状态。

事件数据

以下是此事件的示例数据。

```
{
  "version": "0",
  "id": "12345678-1234-1234-1234-123456789012",
  "detail-type": "EC2 Instance-terminate Lifecycle Action",
  "source": "aws.autoscaling",
  "account": "123456789012",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-west-2",
  "resources": [
    "auto-scaling-group-arn"
  ],
  "detail": {
    "LifecycleActionToken": "87654321-4321-4321-4321-210987654321",
    "AutoScalingGroupName": "my-asg",
    "LifecycleHookName": "my-lifecycle-hook",
    "EC2InstanceId": "i-12345678",
    "LifecycleTransition": "autoscaling:EC2_INSTANCE_TERMINATING"
  }
}
```

EC2 实例终止成功

Auto Scaling 已成功终止实例。

事件数据

以下是此事件的示例数据。

```
{
  "version": "0",
  "id": "12345678-1234-1234-1234-123456789012",
  "detail-type": "EC2 Instance Terminate Successful",
  "source": "aws.autoscaling",
  "account": "123456789012",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-west-2",
  "resources": [
    "auto-scaling-group-arn",
    "instance-arn"
  ],
  "detail": {
    "StatusCode": "InProgress",
    "Description": "Terminating EC2 instance: i-12345678",
    "AutoScalingGroupName": "my-auto-scaling-group",
    "ActivityId": "87654321-4321-4321-4321-210987654321",
    "Details": {
      "Availability Zone": "us-west-2b",
      "Subnet ID": "subnet-12345678"
    },
    "RequestId": "12345678-1234-1234-1234-123456789012",
    "StatusMessage": "",
    "EndTime": "yyyy-mm-ddThh:mm:ssZ",
    "EC2InstanceId": "i-12345678",
    "StartTime": "yyyy-mm-ddThh:mm:ssZ",
    "Cause": "description-text",
  }
}
```

EC2 实例终止失败

Auto Scaling 未能终止实例。

事件数据

以下是此事件的示例数据。

```
{
  "version": "0",
  "id": "12345678-1234-1234-1234-123456789012",
  "detail-type": "EC2 Instance Terminate Unsuccessful",
  "source": "aws.autoscaling",
  "account": "123456789012",
  "time": "yyyy-mm-ddThh:mm:ssZ",
  "region": "us-west-2",
  "resources": [
    "auto-scaling-group-arn",
    "instance-arn"
  ],
  "detail": {
    "StatusCode": "Failed",
    "AutoScalingGroupName": "my-auto-scaling-group",
    "ActivityId": "87654321-4321-4321-4321-210987654321",
    "Details": {
      "Availability Zone": "us-west-2b",
      "Subnet ID": "subnet-12345678"
    },
    "RequestId": "12345678-1234-1234-1234-123456789012",
  }
}
```

```
"StatusMessage": "message-text",  
"EndTime": "yyyy-mm-ddThh:mm:ssZ",  
"EC2InstanceId": "i-12345678",  
"StartTime": "yyyy-mm-ddThh:mm:ssZ",  
"Cause": "description-text",  
}  
}
```

创建 Lambda 函数

使用以下过程可创建 Lambda 函数来处理 Auto Scaling 事件。

创建 Lambda 函数

1. 通过以下网址打开 AWS Lambda 控制台：<https://console.amazonaws.cn/lambda/>。
2. 如果您是首次使用 Lambda，则会看到一个欢迎页面；请选择 Get Started Now；否则，请选择 Create a Lambda function。
3. 在 Select blueprint 页面上，为 Filter 键入 hello-world，然后选择 hello-world 蓝图。
4. 在 Configure triggers 页面上，选择 Next。
5. 在 Configure function 页面上，执行以下操作：
 - a. 键入 Lambda 函数的名称和说明。
 - b. 编辑 Lambda 函数的代码。例如，以下代码仅记录事件：

```
console.log('Loading function');  
  
exports.handler = function(event, context) {  
    console.log("AutoScalingEvent()");  
    console.log("Event data:\n" + JSON.stringify(event, null, 4));  
    context.succeed("...");  
};
```

- c. 对于 Role，选择 Choose an existing role（如果您现在具有要使用的角色），然后从 Existing role 中选择您的角色。或者，要创建新角色，请为 Role 选择其他选项之一，然后按照说明执行操作。
 - d. （可选）对于 Advanced settings，进行所需的任何更改。
 - e. 选择 Next。
6. 在 Review 页面上，选择 Create function。

将事件路由到 Lambda 函数

使用以下过程将 Auto Scaling 事件路由到 Lambda 函数。

将事件路由到 Lambda 函数

1. 通过以下网址打开 CloudWatch 控制台：<https://console.amazonaws.cn/cloudwatch/>。
2. 在导航窗格上，选择 Events。
3. 选择 Create rule。
4. 对于 Event selector，选择 Auto Scaling 作为事件源。默认情况下，规则将应用于您的所有 Auto Scaling 组的所有 Auto Scaling 事件。或者，您可以选择特定的事件或特定的 Auto Scaling 组。
5. 对于 Targets，选择 Add target。选择 Lambda function 作为目标类型，然后选择 Lambda 函数。
6. 选择 Configure details。
7. 对于 Rule definition，键入规则的名称和描述。
8. 选择 Create rule。

要测试您的规则，请更改 Auto Scaling 组的大小。如果您已对 Lambda 函数使用示例代码，则它会将事件记录到 CloudWatch Logs。

测试您的规则

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上，选择 Auto Scaling Groups，然后选择您的 Auto Scaling 组。
3. 在 Details 选项卡上，选择 Edit。
4. 更改 Desired 的值，然后选择 Save。
5. 通过以下网址打开 CloudWatch 控制台：<https://console.amazonaws.cn/cloudwatch/>。
6. 在导航窗格上，选择 Logs。
7. 选择 Lambda 函数的日志组（例如，`/aws/lambda/my-function`）。
8. 选择日志流以查看事件数据。此时将显示数据，类似以下内容：

```
Event Data
▼ 2016-02-22T17:48:20.778Z ealfjqinxq6pwo9d Loading function
▼ START RequestId: 7560439b-d98c-11e5-932d-f52757e7aee0 Version: $LATEST
▼ 2016-02-22T17:48:20.813Z 7560439b-d98c-11e5-932d-f52757e7aee0 AutoScalingEvent()
▼ 2016-02-22T17:48:20.814Z 7560439b-d98c-11e5-932d-f52757e7aee0 Event data:
{
  "version": "0",
  "id": "df9b0c8c-89c6-4748-92cb-ac68a9029ada",
  "detail-type": "EC2 Instance Launch Successful",
  "source": "aws.autoscaling",
```

在 Auto Scaling 组扩展时获取 SNS 通知

使用 Auto Scaling 自动扩展应用程序时，知道 Auto Scaling 何时启动或终止 Auto Scaling 组中的 EC2 实例会很有用。Amazon SNS 协调并管理传输，或将通知发送到订阅客户端或终端节点。您可以将 Auto Scaling 配置为在 Auto Scaling 组扩展时发送 SNS 通知。

Amazon SNS 能够以 HTTP 或 HTTPS POST 以及电子邮件（SMTP，纯文本或 JSON 格式）的形式传输通知，或将通知作为消息发布到 Amazon SQS 队列。有关更多信息，请参阅 Amazon Simple Notification Service 开发人员指南 中的[什么是 Amazon SNS](#)。

例如，如果将 Auto Scaling 组配置为使用 `autoscaling: EC2_INSTANCE_TERMINATE` 通知类型，并且您的 Auto Scaling 组终止了某个实例，则它会发送电子邮件通知。该电子邮件包含已终止实例的详细信息，如实例 ID 以及终止该实例的原因。

Tip

如果您愿意，可以使用 Amazon CloudWatch Events 将目标配置为在 Auto Scaling 组扩展时或在执行生命周期操作时调用 Lambda 函数。有关更多信息，请参阅 [在 Auto Scaling 组扩展时获取 CloudWatch Events](#) (p. 101)。

内容

- [SNS 通知](#) (p. 106)
- [配置 Amazon SNS](#) (p. 107)
- [配置 Auto Scaling 组以发送通知](#) (p. 108)
- [测试通知配置](#) (p. 108)
- [验证是否收到扩展事件的通知](#) (p. 109)
- [删除通知配置](#) (p. 110)

SNS 通知

Auto Scaling 支持在发生以下事件时发送 Amazon SNS 通知。

事件	说明
autoscaling:EC2_INSTANCE_LAUNCH	实例启动成功
autoscaling:EC2_INSTANCE_LAUNCH_ERROR	实例启动失败
autoscaling:EC2_INSTANCE_TERMINATE	实例终止成功
autoscaling:EC2_INSTANCE_TERMINATE_ERROR	实例终止失败

消息包含以下信息：

- Event — 活动。
- AccountId — AWS 账户 ID。
- AutoScalingGroupName — Auto Scaling 组的名称。
- AutoScalingGroupARN — Auto Scaling 组的 ARN。
- EC2InstanceId — EC2 实例的 ID。

例如：

```
Service: AWS Auto Scaling
Time: 2016-09-30T19:00:36.414Z
RequestId: 4e6156f4-a9e2-4bda-a7fd-33f2ae528958
Event: autoscaling:EC2_INSTANCE_LAUNCH
AccountId: 123456789012
AutoScalingGroupName: my-asg
AutoScalingGroupARN: arn:aws:autoscaling:us-west-2:123456789012:autoScalingGroup...
ActivityId: 4e6156f4-a9e2-4bda-a7fd-33f2ae528958
Description: Launching a new EC2 instance: i-0598c7d356eba48d7
Cause: At 2016-09-30T18:59:38Z a user request update of AutoScalingGroup constraints to ...
StartTime: 2016-09-30T19:00:04.445Z
EndTime: 2016-09-30T19:00:36.414Z
StatusCode: InProgress
StatusMessage:
Progress: 50
EC2InstanceId: i-0598c7d356eba48d7
Details: {"Subnet ID":"subnet-c9663da0","Availability Zone":"us-west-2b"}
```

配置 Amazon SNS

要使用 Amazon SNS 发送电子邮件通知，必须先创建一个主题，然后用您的电子邮件地址订阅该主题。

创建一个 Amazon SNS 主题

SNS 主题是一个逻辑接入点，即 Auto Scaling 组用来发送通知的通信通道。您可通过为主题指定名称来创建主题。

有关更多信息，请参阅 Amazon Simple Notification Service 开发人员指南 中的 [创建主题](#)。

订阅 Amazon SNS 主题

要接收您的 Auto Scaling 组发送到该主题的通知，必须让一个终端节点订阅该主题。在此过程中，为 Endpoint 指定要用来自 Auto Scaling 接收通知的电子邮件地址。

有关更多信息，请参阅 Amazon Simple Notification Service 开发人员指南 中的 [订阅主题](#)。

确认 Amazon SNS 订阅

Amazon SNS 向在上一步骤中指定的电子邮件地址发送确认电子邮件。

确保打开来自 AWS 通知的电子邮件，选择链接以确认订阅，然后继续执行下一步骤。

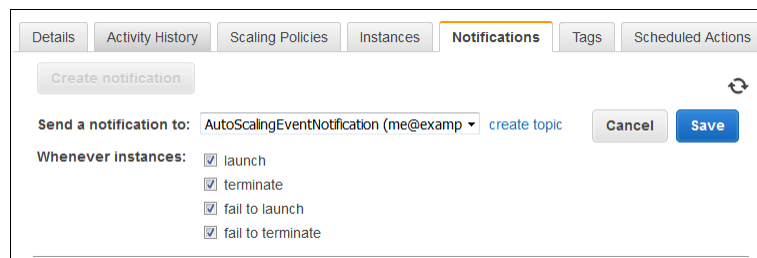
您将从 AWS 收到确认消息。Amazon SNS 现已配置为接收通知并将通知以电子邮件形式发送到您指定的电子邮件地址。

配置 Auto Scaling 组以发送通知

您可以配置 Auto Scaling 组，以便在发生扩展事件（例如，启动或终止实例）时向 Amazon SNS 发送通知。Amazon SNS 向您指定的电子邮件地址发送通知，通知中包含有关实例的信息。

使用控制台为 Auto Scaling 组配置 Amazon SNS 通知

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择您的 Auto Scaling 组。
4. 在 Notifications 选项卡上，选择 Create notification。
5. 在 Create notifications 窗格上，执行以下操作：
 - a. 对于 Send a notification to:，选择您的 SNS 主题。
 - b. 对于 Whenever instances:，选择要为其发送通知的事件。
 - c. 选择 Save。



使用 AWS CLI 为 Auto Scaling 组配置 Amazon SNS 通知

使用以下 `put-notification-configuration` 命令：

```
aws autoscaling put-notification-configuration --auto-scaling-group-name my-  
asg --topic-arn arn --notification-types "autoscaling:EC2_INSTANCE_LAUNCH"  
"autoscaling:EC2_INSTANCE_TERMINATE"
```

测试通知配置

要为启动事件生成通知，请通过将 Auto Scaling 组的所需容量增加 1 来更新 Auto Scaling 组。Auto Scaling 会启动 EC2 实例，您将在数分钟内收到电子邮件通知。

使用控制台更改所需容量

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。

2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择您的 Auto Scaling 组。
4. 在 Details 选项卡上，选择 Edit。
5. 对于 Desired，将当前值增加 1。请注意，如果此值超过 Max，您还必须将 Max 的值增加 1。
6. 选择 Save。
7. 在数分钟后，您将收到启动事件的通知电子邮件。如果您不需要您为此测试启动的其他实例，则可以将 Desired 减少 1。在数分钟后，您将收到终止事件的通知电子邮件。

使用 AWS CLI 更改所需容量

使用以下 `set-desired-capacity` 命令：

```
aws autoscaling set-desired-capacity --auto-scaling-group-name my-asg --desired-capacity 2
```

验证是否收到扩展事件的通知

检查来自 Amazon SNS 的电子邮件消息并打开电子邮件。在收到 Auto Scaling 组的扩展事件通知后，您可以通过查看 Auto Scaling 组的描述来确认该扩展事件。您将需要通知电子邮件中的信息，例如已启动或终止的实例的 ID。

使用控制台验证 Auto Scaling 组已启动新实例

1. 选择您的 Auto Scaling 组。
2. 在 Activity History 选项卡上，Status 列显示实例的当前状态。例如，如果通知指示实例已启动，则使用刷新按钮来验证启动活动的状态是否为 Successful。
3. 在 Instances 选项卡上，可以查看通知电子邮件中收到的 ID 的实例的当前 Lifecycle 状态。新实例启动后，其生命周期状态更改为 InService。

使用 AWS CLI 验证 Auto Scaling 组已启动新实例

使用以下 `describe-auto-scaling-groups` 命令确认已更改 Auto Scaling 组的大小：

```
aws autoscaling describe-auto-scaling-groups --auto-scaling-group-name my-asg
```

下面的示例输出表明该组有两个实例。检查通知电子邮件中收到的 ID 的实例。

```
{
  "AutoScalingGroups": [
    {
      "AutoScalingGroupARN": "arn",
      "HealthCheckGracePeriod": 0,
      "SuspendedProcesses": [],
      "DesiredCapacity": 2,
      "Tags": [],
      "EnabledMetrics": [],
      "LoadBalancerNames": [],
      "AutoScalingGroupName": "my-asg",
      "DefaultCooldown": 300,
      "MinSize": 1,
      "Instances": [
        {
          "InstanceId": "i-d95eb0d4",
          "AvailabilityZone": "us-west-2b",
          "HealthStatus": "Healthy",

```

```
        "LifecycleState": "InService",
        "LaunchConfigurationName": "my-lc"
    },
    {
        "InstanceId": "i-13d7dc1f",
        "AvailabilityZone": "us-west-2a",
        "HealthStatus": "Healthy",
        "LifecycleState": "InService",
        "LaunchConfigurationName": "my-lc"
    }
],
"MaxSize": 5,
"VPCZoneIdentifier": null,
"TerminationPolicies": [
    "Default"
],
"LaunchConfigurationName": "my-lc",
"CreatedTime": "2015-03-01T16:12:35.608Z",
"AvailabilityZones": [
    "us-west-2b",
    "us-west-2a"
],
"HealthCheckType": "EC2"
}
]
```

删除通知配置

您可以随时删除 Auto Scaling 通知配置。

使用控制台删除 Auto Scaling 通知配置

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 Auto Scaling 下，选择 Auto Scaling Groups。
3. 选择您的 Auto Scaling 组。
4. 在 Notifications 选项卡上，选择通知旁边的 Delete。

使用 AWS CLI 删除 Auto Scaling 通知配置

使用以下 delete-notification-configuration 命令：

```
aws autoscaling delete-notification-configuration --auto-scaling-group-name my-asg --topic-arn arn:aws:sns:us-west-2:123456789012:my-sns-topic
```

有关删除与 Auto Scaling 组关联的 Amazon SNS 主题以及删除对该主题的所有订阅的信息，请参阅 Amazon Simple Notification Service 开发人员指南 中的 [清除](#)。

使用 AWS CloudTrail 记录 Auto Scaling API 调用

Auto Scaling 与 CloudTrail 集成在一起，后者是一种服务，它在 AWS 账户中捕获由 Auto Scaling 或代表其发出的 API 请求，并将日志文件提交到您指定的 Amazon S3 存储桶。CloudTrail 从 Auto Scaling 控制台或从 Auto Scaling API 捕获 API 调用。通过使用由 CloudTrail 收集的信息，您可以确定向 Auto Scaling 发出了什么请求、发出请求的源 IP 地址、何人发出的请求以及发出请求的时间等。有关 CloudTrail 的更多信息，包括如何对其进行配置和启用，请参阅 [AWS CloudTrail User Guide](#)。

CloudTrail 中的 Auto Scaling 信息

在您的 AWS 账户中启用 CloudTrail 日志记录后，将在日志文件中跟踪对 Auto Scaling 操作执行的 API 调用。Auto Scaling 记录与其他 AWS 服务记录一起写入日志文件。CloudTrail 基于时间段和文件大小来确定何时创建新文件并向其写入内容。

所有 Auto Scaling 操作都会记录，[Auto Scaling API 参考](#)对这些操作进行了介绍。例如，对 CreateLaunchConfiguration、DescribeAutoScalingGroup 和 UpdateAutoScalingGroup 操作进行的调用会在 CloudTrail 日志文件中生成条目。

每个日志条目都包含有关生成请求的人员的信息。日志中的用户身份信息有助于确定请求是由账户或 IAM 用户凭证发出，通过某个角色或联合身份用户的临时安全凭证发出，还是由其他 AWS 服务发出。有关更多信息，请参阅 AWS CloudTrail User Guide 的 [CloudTrail 事件参考](#)部分中的 `userIdentity`。

日志文件可以在存储桶中存储任意长时间，不过您也可以定义 Amazon S3 生命周期规则以自动存档或删除日志文件。默认情况下，将使用 Amazon S3 服务器端加密 (SSE) 对日志文件进行加密。

如果需要针对日志文件传输快速采取措施，可选择让 CloudTrail 在传输新日志文件时发布 Amazon SNS 通知。有关更多信息，请参阅 AWS CloudTrail User Guide 中的 [配置 Amazon SNS 通知](#)。

您也可以将多个 AWS 区域和多个 AWS 账户中的 Auto Scaling 日志文件聚合到单个 Amazon S3 存储桶中。有关更多信息，请参阅 AWS CloudTrail User Guide 中的 [将 CloudTrail 日志文件聚合到单个 Amazon S3 存储桶中](#)。

了解 Auto Scaling 日志文件条目

CloudTrail 日志文件可包含一个或多个日志条目，每个条目由多个 JSON 格式的事件组成。一个日志条目表示来自任何源的一个请求，包括有关所请求的操作、所有参数以及操作的日期和时间等信息。日志条目不一定具有任何特定顺序。也即，它们不是公用 API 调用的有序堆栈跟踪。

下面的示例显示了一个 CloudTrail 日志条目，该条目演示了 CreateLaunchConfiguration 操作。

```
{
  "Records": [
    {
      "eventVersion": "1.01",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:user/iamUser1",
        "accountId": "123456789012",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "iamUser1"
      },
      "eventTime": "2014-06-24T16:53:14Z",
      "eventSource": "autoscaling.amazonaws.com",
      "eventName": "CreateLaunchConfiguration",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "Amazon CLI/AutoScaling 1.0.61.3 API 2011-01-01",
      "requestParameters": {
        "imageId": "ami-2f726546",
        "instanceType": "m1.small",
        "launchConfigurationName": "launch_configuration_1"
      },
      "responseElements": null,
      "requestID": "07a1becf-fbc0-11e3-bfd8-a5209058e7bb",
      "eventID": "ad30abf7-57db-4a6d-93fa-13deb1fd4cff"
    },
    ...additional entries
  ]
}
```

```
]
}
```

下面的示例显示了一个 CloudTrail 日志条目，该条目演示了 DescribeAutoScalingGroups 操作。

```
{
  "Records": [
    {
      "eventVersion": "1.01",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:user/iamUser1",
        "accountId": "123456789012",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "iamUser1"
      },
      "eventTime": "2014-06-23T23:20:56Z",
      "eventSource": "autoscaling.amazonaws.com",
      "eventName": "DescribeAutoScalingGroups",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "Amazon CLI/AutoScaling 1.0.61.3 API 2011-01-01",
      "requestParameters": {
        "maxRecords": 20
      },
      "responseElements": null,
      "requestID": "0737e2ea-fb2d-11e3-bfd8-a5209058e7bb",
      "eventID": "0353fb04-281e-47d9-93bb-588bf2256538"
    },
    ...additional entries
  ]
}
```

下面的示例显示了一个 CloudTrail 日志条目，该条目演示了 UpdateAutoScalingGroups 操作。

```
{
  "Records": [
    {
      "eventVersion": "1.01",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:user/iamUser1",
        "accountId": "123456789012",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "iamUser1"
      },
      "eventTime": "2014-06-24T16:54:46Z",
      "eventSource": "autoscaling.amazonaws.com",
      "eventName": "UpdateAutoScalingGroup",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "Amazon CLI/AutoScaling 1.0.61.3 API 2011-01-01",
      "requestParameters": {
        "maxSize": 8,
        "minSize": 1,
        "autoScalingGroupName": "asg1"
      },
      "responseElements": null,
      "requestID": "3ed07c03-fbc0-11e3-bfd8-a5209058e7bb",
      "eventID": "b52ca0aa-5199-4873-a546-55f7c896a4ce"
    },
    ...additional entries
  ]
}
```



```
    ...additional entries  
  ]  
}
```

控制对您 Auto Scaling 资源的访问权

Auto Scaling 与 AWS Identity and Access Management (IAM) 集成，后者是一种服务，允许您执行以下操作：

- Create users and groups under your organization's AWS account
- 为您的 AWS 账户下的每个用户分配唯一的安全证书
- 控制每个用户使用 AWS 资源执行任务的权限
- 允许另一 AWS 账户的用户共享 AWS 资源
- 创建 AWS 账户角色并定义可以担任这些角色的用户或服务
- 借助企业的现有身份验证，授予使用 AWS 资源执行任务的权限

例如，您可创建 IAM 策略向 Managers 组授予仅使用

DescribeAutoScalingGroups、DescribeLaunchConfigurations、DescribeScalingActivities 和 DescribePolicies API 操作的权限。随后，Managers 组中的用户可以对任何 Auto Scaling 组和启动配置执行这些操作。

您还可创建 IAM 策略来将访问权限限制到一个特定的 Auto Scaling 组或启动配置。

有关更多信息，请参阅 [Identity and Access Management \(IAM\)](#) 或 [IAM 用户指南](#)。

内容

- [Auto Scaling 操作](#) (p. 114)
- [Auto Scaling 资源](#) (p. 115)
- [Auto Scaling 条件键](#) (p. 116)
- [支持的资源级权限](#) (p. 116)
- [预定义的 AWS 托管策略](#) (p. 118)
- [客户管理的策略](#) (p. 118)
- [使用 IAM 角色启动 Auto Scaling 实例](#) (p. 122)

Auto Scaling 操作

您可在 IAM 策略中指定任何及所有 Auto Scaling 操作。请使用以下前缀为操作命名：autoscaling:。例如：

```
"Action": "autoscaling:CreateAutoScalingGroup"
```

要在单个语句中指定多项操作，请使用方括号将操作括起来并以逗号分隔，如下所示：

```
"Action": [  
  "autoscaling:CreateAutoScalingGroup",  
  "autoscaling:UpdateAutoScalingGroup"  
]
```

还可使用通配符。例如，使用 `autoscaling:*` 指定所有 Auto Scaling 操作。

```
"Action": "autoscaling:*"
```

使用 `Describe:*` 指定名称以 `Describe` 开头的所有操作。

```
"Action": "autoscaling:Describe*"
```

有关更多信息，请参阅 Auto Scaling API 参考 中的 [Auto Scaling 操作](#)。

Auto Scaling 资源

对于支持资源级权限的操作，您可控制允许用户访问的 Auto Scaling 组或启动配置。

要指定 Auto Scaling 组，您必须指定其 Amazon 资源名称 (ARN)，如下所示：

```
"Resource": "arn:aws-  
cn:autoscaling:region:123456789012:autoScalingGroup:uuid:autoScalingGroupName/asg-name"
```

要使用 `CreateAutoScalingGroup` 指定 Auto Scaling 组，您必须将 UUID 替换为 `*`，如下所示：

```
"Resource": "arn:aws-  
cn:autoscaling:region:123456789012:autoScalingGroup:*:autoScalingGroupName/asg-name"
```

要指定启动配置，您必须指定其 ARN，如下所示：

```
"Resource": "arn:aws-  
cn:autoscaling:region:123456789012:launchConfiguration:uuid:launchConfigurationName/lc-  
name"
```

要使用 `CreateLaunchConfiguration` 指定启动配置，您必须将 UUID 替换为 `*`，如下所示：

```
"Resource": "arn:aws-  
cn:autoscaling:region:123456789012:launchConfiguration:*:launchConfigurationName/lc-name"
```

以下 Auto Scaling 操作不支持资源级权限：

- `DescribeAccountLimits`
- `DescribeAdjustmentTypes`
- `DescribeAutoScalingGroups`
- `DescribeAutoScalingInstances`

- DescribeAutoScalingNotificationTypes
- DescribeLaunchConfigurations
- DescribeLifecycleHooks
- DescribeLifecycleHookTypes
- DescribeLoadBalancers
- DescribeLoadBalancerTargetGroups
- DescribeMetricCollectionTypes
- DescribeNotificationConfigurations
- DescribePolicies
- DescribeScalingActivities
- DescribeScalingProcessTypes
- DescribeScheduledActions
- DescribeTags
- DescribeTerminationPolicyTypes

对于不支持资源级权限的操作，您必须将“*”用作资源。

```
"Resource": "*"

```

Auto Scaling 条件键

在创建策略时，您可指定策略应何时生效的条件。要表示条件，请使用预定义的条件键。有特定于 Auto Scaling 的条件键以及 AWS 范围内的条件键。

以下条件键是特定于 Auto Scaling 的：

- autoscaling:ImageId
- autoscaling:InstanceType
- autoscaling:LaunchConfigurationName
- autoscaling:LoadBalancerNames
- autoscaling:MaxSize
- autoscaling:MinSize
- autoscaling:ResourceTag/*key*
- autoscaling:SpotPrice
- autoscaling:TargetGroupARNs
- autoscaling:VPCZoneIdentifiers

有关每个 AWS 服务支持的上下文密钥以及 AWS 范围的策略密钥的列表，请参阅 IAM 用户指南 中的 [AWS 服务操作和条件上下文密钥](#) 和 [条件的可用密钥](#)。

支持的资源级权限

下表介绍了支持资源级权限的 Auto Scaling API 操作，以及每个操作支持的条件键和资源。

API 操作	条件密钥	资源 ARN
AttachInstances	autoscaling:ResourceTag/ key	Auto Scaling 组
AttachLoadBalancers	autoscaling:LoadBalancerNames、autoscaling:ResourceTag/ key	Auto Scaling 组
AttachLoadBalancerTargetGroups	autoscaling:ResourceTag/ key 、autoscaling:TargetGroupARNs	Auto Scaling 组
CompleteLifecycleAction	autoscaling:ResourceTag/ key	Auto Scaling 组
CreateAutoScalingGroup	autoscaling:LaunchConfigurationName、 key 、autoscaling:TargetGroupARNs、autoscaling:LoadBalancerNames、 key 、aws:TagKeys	Auto Scaling 组 (将 key 替换为 *)
CreateLaunchConfiguration	autoscaling:ImageId、autoscaling:InstanceProfile、autoscaling:SpotPrice	启动配置 (将 key 替换为 *)
CreateOrUpdateTags	autoscaling:ResourceTag/ key 、aws:RequestTag/ key 、aws:TagKeys	Auto Scaling 组
DeleteAutoScalingGroup	autoscaling:ResourceTag/ key	Auto Scaling 组
DeleteLaunchConfiguration		启动配置
DeleteLifecycleHook	autoscaling:ResourceTag/ key	Auto Scaling 组
DeleteNotificationConfiguration	autoscaling:ResourceTag/ key	Auto Scaling 组
DeletePolicy	autoscaling:ResourceTag/ key	Auto Scaling 组
DeleteScheduledAction	autoscaling:ResourceTag/ key	Auto Scaling 组
DeleteTags	autoscaling:ResourceTag/ key 、aws:RequestTag/ key 、aws:TagKeys	Auto Scaling 组
DetachInstances	autoscaling:ResourceTag/ key	Auto Scaling 组
DetachLoadBalancers	autoscaling:LoadBalancerNames、autoscaling:ResourceTag/ key	Auto Scaling 组
DetachLoadBalancerTargetGroups	autoscaling:ResourceTag/ key 、autoscaling:TargetGroupARNs	Auto Scaling 组
DisableMetricsCollection	autoscaling:ResourceTag/ key	Auto Scaling 组
EnableMetricsCollection	autoscaling:ResourceTag/ key	Auto Scaling 组
EnterStandby	autoscaling:ResourceTag/ key	Auto Scaling 组
ExecutePolicy	autoscaling:ResourceTag/ key	Auto Scaling 组
ExitStandby	autoscaling:ResourceTag/ key	Auto Scaling 组
PutLifecycleHook	autoscaling:ResourceTag/ key	Auto Scaling 组
PutNotificationConfiguration	autoscaling:ResourceTag/ key	Auto Scaling 组
PutScalingPolicy	autoscaling:ResourceTag/ key	Auto Scaling 组

API 操作	条件密钥	资源 ARN
PutScheduledUpdateGroupAction	autoscaling:MaxSize、autoscaling:MinSize、autoscaling:ResourceTag/ key	Auto Scaling 组
RecordLifecycleActionHeartbeat	autoscaling:ResourceTag/ key	Auto Scaling 组
ResumeProcesses	autoscaling:ResourceTag/ key	Auto Scaling 组
SetDesiredCapacity	autoscaling:ResourceTag/ key	Auto Scaling 组
SetInstanceHealth	autoscaling:ResourceTag/ key	Auto Scaling 组
SetInstanceProtection	autoscaling:ResourceTag/ key	Auto Scaling 组
SuspendProcesses	autoscaling:ResourceTag/ key	Auto Scaling 组
TerminateInstanceInAutoScalingGroup	autoscaling:ResourceTag/ key	Auto Scaling 组
UpdateAutoScalingGroup	autoscaling:LaunchConfigurationName、autoscaling:MaxSize、autoscaling:MinSize、autoscaling:ResourceTag/ key 、autoscaling:VPCZoneIdentifiers	Auto Scaling 组

预定义的 AWS 托管策略

AWS 创建的托管策略将授予针对常用案例的必要权限。您可以将这些策略附加到 IAM 用户。下面是适用于 Auto Scaling 的 AWS 托管策略。

- [AutoScalingConsoleFullAccess](#) - 授予对控制台使用的针对 Auto Scaling 资源的所有 API 操作的访问权限。这包括针对 Auto Scaling 的所有 API 操作以及针对 Amazon EC2、CloudWatch、Elastic Load Balancing 和 Amazon SNS 的选定 API 操作。
- [AutoScalingConsoleReadOnlyAccess](#) - 授予对控制台使用的针对 Auto Scaling 资源的只读 API 操作的访问权限。这包括针对 Auto Scaling 的所有只读 API 操作以及针对 Amazon EC2、CloudWatch、Elastic Load Balancing 和 Amazon SNS 的选定只读 API 操作。
- [AutoScalingFullAccess](#) - 授予对所有 Auto Scaling API 操作的访问权限。
- [AutoScalingReadOnlyAccess](#) - 授予对只读 Auto Scaling API 操作的访问权限。

客户管理的策略

您可以创建自定义 IAM 策略来向 IAM 用户授予对特定资源执行特定操作的权限。以下是 Auto Scaling 的示例策略。

示例：创建并管理启动配置

以下策略允许用户使用在名称中包含 `LaunchConfiguration` 字符串的所有 Auto Scaling 操作。或者，您还可以明确地列出每项操作，而不是使用通配符。但是，此策略不会自动应用于其名称中包含 `LaunchConfiguration` 的任何新 Auto Scaling 操作。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": "autoscaling:*LaunchConfiguration*",
    "Resource": "*"
  }]
}
```

```
}
```

以下策略授权用户在实例类型为 `t2.micro` 且启动配置的名称以 `t2micro-` 开头时创建启动配置，并为 Auto Scaling 组（仅当其名称以 `t2micro-` 开头时）指定启动配置。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "autoscaling:CreateLaunchConfiguration",
      "Resource": [
        "arn:aws-cn:autoscaling:us-
west-2:123456789012:launchConfiguration:*:launchConfigurationName/t2micro-*"
      ],
      "Condition": {
        "StringEquals": { "autoscaling:InstanceType": "t2.micro" }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "autoscaling:CreateAutoScalingGroup",
        "autoscaling:UpdateAutoScalingGroup"
      ],
      "Resource": "*",
      "Condition": {
        "StringLikeIfExists": { "autoscaling:LaunchConfigurationName": "t2micro-*" }
      }
    }
  ]
}
```

示例：创建并管理 Auto Scaling 组和扩展策略

以下策略允许用户使用在名称中包含 `Scaling` 字符串的所有 Auto Scaling 操作。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": ["autoscaling:*Scaling*"],
    "Resource": "*"
  }]
}
```

以下策略授权用户使用其名称中包含 `Scaling` 字符串的所有 Auto Scaling 操作，前提是 Auto Scaling 组具有标签 `purpose=webserver`。由于 `Describe` 操作不支持资源级权限，因此，您必须在不带条件的单独语句中必须指定它们。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": ["autoscaling:*Scaling*"],
      "Resource": "*",
      "Condition": {
        "StringEquals": { "autoscaling:ResourceTag/purpose": "webserver" }
      }
    },
    {

```



```
    "Effect": "Allow",
    "Action": "autoscaling:Describe*Scaling*",
    "Resource": "*"
  }]
}
```

以下策略授权用户使用其名称中包含字符串 `scaling` 的所有 Auto Scaling 操作，前提是它们未指定小于 1 的最小大小或大于 10 的最大大小。由于 `Describe` 操作不支持资源级权限，因此您必须在不带条件的单独语句中必须指定它们。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": ["autoscaling:*Scaling*"],
      "Resource": "*",
      "Condition": {
        "NumericGreaterThanEqualsIfExists": { "autoscaling:MinSize": 1 },
        "NumericLessThanEqualsIfExists": { "autoscaling:MaxSize": 10 }
      }
    },
    {
      "Effect": "Allow",
      "Action": "autoscaling:Describe*Scaling*",
      "Resource": "*"
    }
  ]
}
```

示例：使用标签控制访问

要授予用户创建 Auto Scaling 组或为该组添加标签 (仅当用户指定特定标签时) 的权限，请使用 `aws:RequestTag` 条件键。要仅允许特定的标签键，请使用带 `ForAnyValue` 修饰符的 `aws:TagKeys` 条件键。

以下策略需要用户使用标签 `purpose=webserver` 和 `cost-center=cc123` 标记任何 Auto Scaling 组，并且仅允许 `purpose` 和 `cost-center` 标签 (不能指定其他标签)。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "autoscaling:CreateAutoScalingGroup",
      "autoscaling:CreateOrUpdateTags"
    ],
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "aws:RequestTag/purpose": "webserver",
        "aws:RequestTag/cost-center": "cc123"
      },
      "ForAllValues:StringEquals": { "aws:TagKeys": ["purpose", "cost-center"] }
    }
  }]
}
```

以下策略需要用户在请求中指定包含键 `environment` 的标签。

```
{
  "Version": "2012-10-17",
  "Statement": [{
```

```
    "Effect": "Allow",
    "Action": [
        "autoscaling:CreateAutoScalingGroup",
        "autoscaling:CreateOrUpdateTags"
    ],
    "Resource": "*",
    "Condition": {
        "StringLike": { "aws:RequestTag/environment": "*" }
    }
  }
}
```

以下策略需要用户在请求中指定至少一个标签，并且仅允许 **cost-center** 和 **owner** 键。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
        "autoscaling:CreateAutoScalingGroup",
        "autoscaling:CreateOrUpdateTags"
    ],
    "Resource": "*",
    "Condition": {
        "ForAnyValue:StringEquals": { "aws:TagKeys": ["cost-center", "owner"] }
    }
  }]
}
```

以下策略授权用户访问具有标签 **allowed=true** 的 Auto Scaling 组，并允许它们仅应用标签 **environment=test**。由于启动配置不支持标签并且 Describe 操作不支持资源级权限，因此您必须在不带条件的单独语句中指定它们。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": "autoscaling:*Scaling*",
    "Resource": "*",
    "Condition": {
        "StringEquals": { "autoscaling:ResourceTag/allowed": "true" },
        "StringEqualsIfExists": { "aws:RequestTag/environment": "test" },
        "ForAllValues:StringEquals": { "aws:TagKeys": "environment" }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
        "autoscaling:*LaunchConfiguration*",
        "autoscaling:Describe*"
    ],
    "Resource": "*"
  }]
}
```

示例：更改 Auto Scaling 组的容量

以下策略授权用户使用 **SetDesiredCapacity** 操作更改任何 Auto Scaling 组的容量。

```
{
  "Version": "2012-10-17",
```

```
"Statement": [{
  "Effect": "Allow",
  "Action": "autoscaling:SetDesiredCapacity",
  "Resource": "*"
}]
}
```

以下策略授权用户使用 `SetDesiredCapacity` 操作更改指定的 Auto Scaling 组的容量。请注意，包括 UUID 可确保将访问权授予特定的 Auto Scaling 组。如果您要删除 Auto Scaling 组并创建一个新的同名组，新组的 UUID 将与原始组的 UUID 不同。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": "autoscaling:SetDesiredCapacity",
    "Resource": [
      "arn:aws-cn:autoscaling:us-west-2:123456789012:autoScalingGroup:7fe02b8e-7442-4c9e-8c8e-85fa99e9b5d9:autoScalingGroupName/group-1",
      "arn:aws-cn:autoscaling:us-west-2:123456789012:autoScalingGroup:9d8e8ea4-22e1-44c7-a14d-520f8518c2b9:autoScalingGroupName/group-2",
      "arn:aws-cn:autoscaling:us-west-2:123456789012:autoScalingGroup:60d6b363-ae8b-467c-947f-f1d308935521:autoScalingGroupName/group-3"
    ]
  }]
}
```

以下策略授权用户使用 `SetDesiredCapacity` 操作更改其名称以 **group-** 开头的任何 Auto Scaling 组的容量。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": "autoscaling:SetDesiredCapacity",
    "Resource": [
      "arn:aws-cn:autoscaling:us-west-2:123456789012:autoScalingGroup:*:autoScalingGroupName/group-*"
    ]
  }]
}
```

使用 IAM 角色启动 Auto Scaling 实例

将 AWS Identity and Access Management (IAM) 角色用于 EC2 实例可以使用户更加轻松安全地访问 EC2 实例中的其他 AWS 服务。使用 IAM 角色启动的 EC2 实例将自动具有可用的 AWS 安全证书。

您可使用 IAM 角色启动 Auto Scaling 实例以自动允许实例上运行的应用程序安全地访问其他 AWS 资源。通过使用 EC2 实例配置文件创建启动配置来做到这一点。实例配置文件就是 IAM 角色的容器。首先，创建具有访问 AWS 资源所需的所有权限的 IAM 角色，然后将您的角色添加到实例配置文件。

有关 IAM 角色和实例配置文件的更多信息，请参阅 IAM 用户指南 中的 [IAM 角色](#)。

先决条件

为您的 EC2 实例创建 IAM 角色。控制台创建与 IAM 角色同名的实例配置文件。

创建 IAM 角色

1. 通过以下网址打开 IAM 控制台：<https://console.amazonaws.cn/iam/>。
2. 在导航窗格中，选择 Roles 和 Create new role。
3. 在 Select role type 页上，选择 Amazon EC2 旁的 Select。
4. 在 Attach Policy 页上，选择向实例授予对所需资源的访问权的 AWS 托管策略。
5. 在 Set role name and review 页上，键入角色的名称，然后选择 Create role。

创建启动配置

在使用 AWS 管理控制台创建启动配置时，在 Configure Details 页上，从 IAM role 中选择角色。有关更多信息，请参阅 [创建启动配置 \(p. 19\)](#)。

在从 AWS CLI 使用 `create-launch-configuration` 命令创建启动配置时，请指定实例配置文件的名称，如下所示：

```
aws autoscaling create-launch-configuration --launch-configuration-name my-lc-with-  
instance-profile \  
--image-id ami-baba68d3 --instance-type m1.small \  
--iam-instance-profile my-instance-profile
```

Auto Scaling 问题排查

Amazon Web Services 提供特定的描述性错误消息来帮助您排查关于 Auto Scaling 的问题。可以从 Auto Scaling 活动的描述中发现错误消息。

内容

- [检索错误消息 \(p. 124\)](#)
- [Auto Scaling 问题排查：EC2 实例启动失败 \(p. 126\)](#)
- [Auto Scaling 问题排查：AMI 问题 \(p. 128\)](#)
- [Auto Scaling 问题排查：负载均衡器问题 \(p. 130\)](#)
- [Auto Scaling 问题排查：容量限制 \(p. 131\)](#)

检索错误消息

要从 Auto Scaling 活动的描述中检索错误消息，请使用 `describe-scaling-activities` 命令，如下所示：

```
aws autoscaling describe-scaling-activities --auto-scaling-group-name my-asg
```

在下面的示例响应中，`StatusCode` 包含活动的当前状态，`StatusMessage` 包含错误消息：

```
{
  "Activities": [
    {
      "Description": "Launching a new EC2 instance: i-4ba0837f",
      "AutoScalingGroupName": "my-asg",
      "ActivityId": "f9f2d65b-flf2-43e7-b46d-d86756459699",
      "Details": "{\"Availability Zone\":\"us-west-2c\"}",
      "StartTime": "2013-08-19T20:53:29.930Z",
      "Progress": 100,
      "EndTime": "2013-08-19T20:54:02Z",
      "Cause": "At 2013-08-19T20:53:25Z a user request created an
AutoScalingGroup...",
      "StatusCode": "Failed",
      "StatusMessage": "The image id 'ami-4edb0327' does not exist. Launching EC2
instance failed."
    }
  ]
}
```

```
}
]
```

下表列出了错误消息的类型，并提供问题排查资源的链接，您在排查 Auto Scaling 问题时可以使用这些资源。

EC2 实例启动失败

问题	错误消息
Auto Scaling 组	AutoScalingGroup <Auto Scaling 组名称> 未找到。 (p. 127)
可用区域	不再支持请求的可用区域。请重新尝试请求..... (p. 127)
AWS 账户	您没有订阅此项服务。请参阅 http://www.amazonaws.cn 。 (p. 127)
块储存设备映射	设备名称上载无效。启动 EC2 实例失败。 (p. 128)
块储存设备映射	用于参数 virtualName 的值 (<与实例存储设备相关联的名称>) 无效... (p. 128)
块储存设备映射	实例存储 AMI 不支持 EBS 块储存设备映射。 (p. 128)
实例类型和可用区域	请求的实例类型 (<实例类型>) 在请求的可用区域 (<实例可用区域>) 中不受支持.... (p. 127)
密钥对	密钥对 <与您的 EC2 实例相关联的密钥对> 不存在。启动 EC2 实例失败。 (p. 127)
启动配置	当前不支持请求的配置。 (p. 127)
置放群组	置放群组可能无法与类型为“m1.large”的实例一起使用。启动 EC2 实例失败。 (p. 128)
安全组	安全组 <该安全组的名称> 不存在。启动 EC2 实例失败。 (p. 126)

AMI 问题

问题	错误消息
AMI ID	AMI ID <您的 AMI 的 ID> 不存在。启动 EC2 实例失败。 (p. 129)
AMI ID	AMI <AMI ID> 正在等待，无法运行。启动 EC2 实例失败。 (p. 129)
AMI ID	用于参数 virtualName 的值 (<AMI ID>) 无效。 (p. 129)
架构不匹配	请求的实例类型的架构 (i386) 与用于 ami-6622f00f (x86_64) 的清单中的架构不匹配。启动 EC2 实例失败。 (p. 130)
虚拟化类型	虚拟化类型为“hvm”的非 Windows AMI 当前可能仅适用于群集计算实例类型。启动 EC2 实例失败。 (p. 129)

负载均衡器问题

问题	错误消息
找不到负载均衡器	找不到负载均衡器 <您的启动环境>。验证负载均衡器配置失败。 (p. 130)
VPC 中的实例	EC2 实例 <实例 ID> 不在 VPC 内。更新负载均衡器配置失败。 (p. 131)

问题	错误消息
没有活动的负载均衡器	名为 <负载均衡器名称> 的活动负载均衡器不存在。更新负载均衡器配置失败。(p. 130)
安全令牌	请求中包含的安全令牌无效。验证负载均衡器配置失败。(p. 131)

容量限制

问题	错误消息
容量限制	已运行 <实例数量> 个实例。启动 EC2 实例失败。(p. 132)
可用区域中容量不足	您请求的可用区域 (<请求的可用区域>) 中当前没有足够的 <实例类型> 容量.... (p. 131)

Auto Scaling 问题排查：EC2 实例启动失败

此页提供有关未能通过 Auto Scaling 启动的 EC2 实例的信息、可能原因，以及可用来解决问题的步骤。

要检索错误消息，请参阅[检索错误消息](#) (p. 124)。

EC2 实例启动失败时，您可能会收到一条或多条以下错误消息：

错误消息

- 安全组 <该安全组的名称> 不存在。启动 EC2 实例失败。(p. 126)
- 密钥对 <与您的 EC2 实例相关联的密钥对> 不存在。启动 EC2 实例失败。(p. 127)
- 当前不支持请求的配置。(p. 127)
- AutoScalingGroup <Auto Scaling 组名称> 未找到。(p. 127)
- 不再支持请求的可用区域。请重新尝试请求..... (p. 127)
- 请求的实例类型 (<实例类型>) 在请求的可用区域 (<实例可用区域>) 中不受支持.... (p. 127)
- 您没有订阅此项服务。请参阅 <http://www.amazonaws.cn>。(p. 127)
- 设备名称上载无效。启动 EC2 实例失败。(p. 128)
- 用于参数 virtualName 的值 (<与实例存储设备相关联的名称>) 无效... (p. 128)
- 实例存储 AMI 不支持 EBS 块储存设备映射。(p. 128)
- 置放群组可能无法与类型为“m1.large”的实例一起使用。启动 EC2 实例失败。(p. 128)

安全组 <该安全组的名称> 不存在。启动 EC2 实例失败。

- 原因：可能已删除启动配置中指定的安全组。
- 解决方案：
 1. 使用 [describe-security-groups](#) 命令获取与您的账户相关联的安全组列表。
 2. 从该列表中选择要使用的安全组。要改为创建安全组，请使用 [create-security-group](#) 命令。
 3. 创建新的启动配置。
 4. 通过 [update-auto-scaling-group](#) 命令，使用新的启动配置更新 Auto Scaling 组。

密钥对 <与您的 EC2 实例相关联的密钥对> 不存在。启动 EC2 实例失败。

- 原因：可能已删除启动实例时使用的密钥对。
- 解决方案：
 1. 使用 `describe-key-pairs` 命令获取可用密钥对的列表。
 2. 从该列表中选择要使用的密钥对。要改为创建密钥对，请使用 `create-key-pair` 命令。
 3. 创建新的启动配置。
 4. 通过 `update-auto-scaling-group` 命令，使用新的启动配置更新 Auto Scaling 组。

当前不支持请求的配置。

- 原因：启动配置中的某些选项当前可能不受支持。
- 解决方案：
 1. 创建新的启动配置。
 2. 通过 `update-auto-scaling-group` 命令，使用新的启动配置更新 Auto Scaling 组。

AutoScalingGroup <Auto Scaling 组名称> 未找到。

- 原因：该 Auto Scaling 组可能已删除。
- 解决方案：创建新 Auto Scaling 组。

不再支持请求的可用区域。请重新尝试请求.....

- Error Message：不再支持请求的可用区域。请通过不指定可用区域或选择 <提供的可用区域的列表> 来重新尝试请求。启动 EC2 实例失败。
- 原因：当前可能无法使用与 Auto Scaling 组相关联的可用区。
- 解决方案：采用错误消息中的建议更新 Auto Scaling 组。

请求的实例类型 (<实例类型>) 在请求的可用区域 (<实例可用区域>) 中不受支持....

- Error Message：请求的实例类型 (<实例类型>) 在请求的可用区域 (<实例可用区域>) 中不受支持。请通过不指定可用区域或选择 <支持该实例类型的可用区域列表> 来重新尝试请求。启动 EC2 实例失败。
- 原因：当前可能无法在 Auto Scaling 组中指定的可用区内使用与启动配置相关联的实例类型。
- 解决方案：采用错误消息中的建议更新 Auto Scaling 组。

您没有订阅此项服务。请参阅 <http://www.amazonaws.cn>。

- 原因：您的 AWS 账户可能已过期。
- 解决方案：转至 <http://www.amazonaws.cn>，然后选择 Sign Up Now 以开立新账户。

设备名称上载无效。启动 EC2 实例失败。

- 原因：启动配置中的块储存设备映射所包含的块储存设备名称可能无法使用或目前不受支持。
- 解决方案：
 1. 使用 [describe-volumes](#) 命令查看卷如何向实例公开。
 2. 使用卷描述中列出的设备名称创建新的启动配置。
 3. 通过 [update-auto-scaling-group](#) 命令，使用新的启动配置更新 Auto Scaling 组。

用于参数 virtualName 的值 (<与实例存储设备相关联的名称>) 无效...

- Error Message：用于参数 virtualName 的值 (<与实例存储设备相关联的名称>) 无效。预期格式：“ephemeralNUMBER”。启动 EC2 实例失败。
- 原因：与块储存设备相关联的虚拟名称的指定格式不正确。
- 解决方案：
 1. 通过在 virtualName 参数中指定设备名称创建新的启动配置。有关设备名称格式的信息，请参阅 Amazon EC2 用户指南 (适用于 Linux 实例) 中的 [实例存储设备名称](#)。
 2. 通过 [update-auto-scaling-group](#) 命令，使用新的启动配置更新 Auto Scaling 组。

实例存储 AMI 不支持 EBS 块储存设备映射。

- 原因：启动配置中指定的块储存设备映射在您的实例上不受支持。
- 解决方案：
 1. 使用实例类型支持的块储存设备映射来创建新的启动配置。有关更多信息，请参阅 Amazon EC2 用户指南 (适用于 Linux 实例) 中的 [块储存设备映射](#)。
 2. 通过 [update-auto-scaling-group](#) 命令，使用新的启动配置更新 Auto Scaling 组。

置放群组可能无法与类型为“m1.large”的实例一起使用。启动 EC2 实例失败。

- 原因：您的群集置放群组包含无效实例类型。
- 解决方案：
 1. 有关置放群组所支持的有效实例类型的信息，请参阅 Amazon EC2 用户指南 (适用于 Linux 实例) 中的 [置放群组](#)。
 2. 按照 [置放群组](#) 中的详细说明创建新的置放群组。
 3. 或者，也可以使用受支持的实例类型创建新的启动配置。
 4. 通过 [update-auto-scaling-group](#) 命令，使用新的置放群组或启动配置更新 Auto Scaling 组。

Auto Scaling 问题排查：AMI 问题

本页提供与 AMI 相关联的问题的有关信息、可能原因，以及可用来解决这些问题的步骤。

要检索错误消息，请参阅[检索错误消息](#) (p. 124)。

当 AMI 的问题导致 EC2 实例启动失败时，您可能会收到一条或多条以下错误消息：

错误消息

- AMI ID <您的 AMI 的 ID> 不存在。启动 EC2 实例失败。 (p. 129)
- AMI <AMI ID> 正在等待，无法运行。启动 EC2 实例失败。 (p. 129)
- 虚拟化类型为“hvm”的非 Windows AMI 当前可能仅适用于群集计算实例类型。启动 EC2 实例失败。 (p. 129)
- 用于参数 `virtualName` 的值 (<AMI ID>) 无效。 (p. 129)
- 请求的实例类型的架构 (i386) 与用于 `ami-6622f00f` (x86_64) 的清单中的架构不匹配。启动 EC2 实例失败。 (p. 130)

AMI ID <您的 AMI 的 ID> 不存在。启动 EC2 实例失败。

- 原因：创建启动配置后，可能已删除 AMI。
- 解决方案：
 1. 使用有效 AMI 创建新的启动配置。
 2. 通过 `update-auto-scaling-group` 命令，使用新的启动配置更新 Auto Scaling 组。

AMI <AMI ID> 正在等待，无法运行。启动 EC2 实例失败。

- 原因：您可能刚创建 AMI (通过获取运行实例的快照或任何其他方式) ，它可能还无法使用。
- 解决方案：必须等待您的 AMI 可用后，才能创建启动配置。

虚拟化类型为“hvm”的非 Windows AMI 当前可能仅适用于群集计算实例类型。启动 EC2 实例失败。

- 原因：具有 hvm 虚拟化的 Linux AMI 不能用于启动非群集计算实例。
- 解决方案：
 1. 使用半虚拟化的 AMI 创建新的启动配置以启动非群集计算实例。
 2. 通过 `update-auto-scaling-group` 命令，使用新的启动配置更新 Auto Scaling 组。

用于参数 `virtualName` 的值 (<AMI ID>) 无效。

- 原因：不正确的值。`virtualName` 参数表示与设备相关联的虚拟名称。
- 解决方案：
 1. 通过指定使用 `virtualName` 参数的实例的虚拟设备名称来创建新的启动配置。
 2. 通过 `update-auto-scaling-group` 命令，使用新的启动配置更新 Auto Scaling 组。

请求的实例类型的架构 (i386) 与用于 ami-6622f00f (x86_64) 的清单中的架构不匹配。启动 EC2 实例失败。

- 原因：启动配置中提到的 `InstanceType` 的架构与映像架构不匹配。
- 解决方案：
 1. 使用与请求的实例类型的架构相匹配的 AMI 架构来创建新的启动配置。
 2. 通过 `update-auto-scaling-group` 命令，使用新的启动配置更新 Auto Scaling 组。

Auto Scaling 问题排查：负载均衡器问题

本页提供与 Auto Scaling 组相关联的负载均衡器所导致的问题有关的信息、可能原因，以及可用来解决这些问题的步骤。

要检索错误消息，请参阅[检索错误消息 \(p. 124\)](#)。

如果与 Auto Scaling 组关联的负载均衡器导致 EC2 实例启动失败，您可能会收到一条或多条以下错误消息：

错误消息

- 找不到负载均衡器 <您的启动环境>。验证负载均衡器配置失败。(p. 130)
- 名为 <负载均衡器名称> 的活动负载均衡器不存在。更新负载均衡器配置失败。(p. 130)
- EC2 实例 <实例 ID> 不在 VPC 内。更新负载均衡器配置失败。(p. 131)
- EC2 实例 <实例 ID> 在 VPC 中。更新负载均衡器配置失败。(p. 131)
- 请求中包含的安全令牌无效。验证负载均衡器配置失败。(p. 131)

找不到负载均衡器 <您的启动环境>。验证负载均衡器配置失败。

- 原因 1：已删除负载均衡器。
- 解决方案 1：
 1. 检查负载均衡器是否仍然存在。可以使用 `describe-load-balancers` 命令。
 2. 如果看到响应中列出了负载均衡器，请参阅原因 2。
 3. 如果没有看到响应中列出负载均衡器，可以创建新的负载均衡器，然后创建新的 Auto Scaling 组，或者可以创建无负载均衡器的新的 Auto Scaling 组。
- 原因 2：创建 Auto Scaling 组时，未采用正确顺序指定负载均衡器名称。
- 解决方案 2：创建新的 Auto Scaling 组，并在最后指定负载均衡器名称。

名为 <负载均衡器名称> 的活动负载均衡器不存在。更新负载均衡器配置失败。

- 原因：可能已删除指定的负载均衡器。
- 解决方案：可以创建新的负载均衡器，然后创建新的 Auto Scaling 组，也可以创建无负载均衡器的新 Auto Scaling 组。

EC2 实例 <实例 ID> 不在 VPC 内。更新负载均衡器配置失败。

- 原因：VPC 中不存在指定的实例。
- 解决方案：可以删除与实例相关联的负载均衡器，或者创建新的 Auto Scaling 组。

EC2 实例 <实例 ID> 在 VPC 中。更新负载均衡器配置失败。

- 原因：负载均衡器在 EC2-Classic 中，但 Auto Scaling 组在 VPC 中。
- 解决方案：确保负载均衡器和 Auto Scaling 组在同一网络（EC2-Classic 或 VPC）中。

请求中包含的安全令牌无效。验证负载均衡器配置失败。

- 原因：您的 AWS 账户可能已过期。
- 解决方案：检查 AWS 账户是否有效。转至 <http://www.amazonaws.cn>，然后选择 Sign Up Now 来开立新账户。

Auto Scaling 问题排查：容量限制

本页提供有关 Auto Scaling 组的容量限制问题的信息、可能原因，以及可用来解决这些问题的步骤。

要检索错误消息，请参阅[检索错误消息](#) (p. 124)。

如果 Auto Scaling 组的容量限制问题导致 EC2 实例启动失败，您可能会收到一条或多条以下错误消息。

错误消息

- 您请求的可用区域（<请求的可用区域>）中当前没有足够的 <实例类型> 容量.... (p. 131)
- 已运行 <实例数量> 个实例。启动 EC2 实例失败。(p. 132)

您请求的可用区域（<请求的可用区域>）中当前没有足够的 <实例类型> 容量....

- Error Message：您请求的可用区域（<请求的可用区域>）中当前没有足够的 <实例类型> 容量。我们的系统将调配额外的容量。您当前可以通过不在请求中指定可用区域，或者选择 <当前支持该实例类型的可用区域列表> 来获取 <实例类型> 容量。启动 EC2 实例失败。
- 原因：现在，Auto Scaling 不能在请求的可用区中支持您的实例类型。
- 解决方案：
 1. 按照错误消息中的建议创建新的启动配置。
 2. 通过 `update-auto-scaling-group` 命令，使用新的启动配置更新 Auto Scaling 组。

已运行 <实例数量> 个实例。启动 EC2 实例失败。

- 原因：Auto Scaling 组已达到 `DesiredCapacity` 参数设置的限制。
- 解决方案：
 - 使用 `update-auto-scaling-group` 命令，通过为 `--desired-capacity` 参数提供新值来更新 Auto Scaling 组。
 - 如果您已达到 EC2 实例的数量限制，可以请求增加。有关更多信息，请参阅 [AWS 服务限制](#)。

Auto Scaling 资源

下列相关资源在您使用此服务的过程中会有所帮助。

- [Auto Scaling](#) – 提供 Auto Scaling 相关信息的主要 Web 页面。
- [Auto Scaling 技术常见问题](#) – 客户提出的 Auto Scaling 问题的答案。
- [Amazon EC2 开发论坛](#) – 从社区获取帮助。

- [课程和研讨会](#) – 指向基于角色的专业课程和自主进度动手实验室的链接，这些课程和实验室旨在帮助您增强 AWS 技能并获得实践经验。
- [AWS 开发人员工具](#) – 指向开发人员工具、软件开发工具包、IDE 工具包和命令行工具的链接，这些资源用于开发和管理 AWS 应用程序。
- [AWS 白皮书](#) – 指向 AWS 技术白皮书的完整列表的链接，这些资料涵盖了架构、安全性、经济性等主题，由 AWS 解决方案架构师或其他技术专家编写。
- [AWS Support 中心](#) - 用于创建和管理 AWS Support 案例的中心。还包括指向其他有用资源的链接，如论坛、技术常见问题、服务运行状况和 AWS Trusted Advisor。
- [AWS Support](#) - 提供有关 AWS Support 信息的主要网页，是一种一对一的快速响应支持渠道，可帮助您在云中构建和运行应用程序。
- [联系我们](#) - 查询有关 AWS 账单、账户、事件、滥用和其他问题的中央联系点。
- [AWS 网站条款](#) - 有关我们的版权和商标、您的账户、许可、网站访问和其他主题的详细信息。

文档历史记录

下表介绍 Auto Scaling 文档的重要补充部分。

功能	描述	发行日期
支持资源级权限	创建 IAM 策略以控制资源级访问。有关更多信息，请参阅 控制对您 Auto Scaling 资源的访问权 (p. 114) 。	2017 年 5 月 15 日
监控改进	Auto Scaling 组指标不再需要您启用详细监控。您现在可以启用组指标集合和从控制台中的 Monitoring 选项卡上查看指标图形。有关更多信息，请参阅 使用 Amazon CloudWatch 监控 Auto Scaling 组和实例 (p. 97) 。	2016 年 8 月 18 日
支持 应用程序负载均衡器	将一个或多个目标组附加到新的或现有的 Auto Scaling 组。有关更多信息，请参阅 将负载均衡器附加到 Auto Scaling 组 (p. 43) 。	2016 年 8 月 11 日
生命周期挂钩的事件	Auto Scaling 在执行生命周期挂钩时会向 CloudWatch Events 发送事件。有关更多信息，请参阅 在 Auto Scaling 组扩展时获取 CloudWatch Events (p. 101) 。	2016 年 2 月 24 日
实例保护	阻止 Auto Scaling 在缩小时选择终止特定实例。有关更多信息，请参阅 实例保护 (p. 81) 。	2015 年 12 月 07 日
分步扩展策略	创建扩展策略，使您能根据警报违例的规模来扩展。有关更多信息，请参阅 扩展策略类型 (p. 63) 。	2015 年 7 月 6 日
更新负载均衡器	将负载均衡器挂载到现有 Auto Scaling 组或从中分离。有关更多信息，请参阅 将负载均衡器附加到 Auto Scaling 组 (p. 43) 。	2015 年 6 月 11 日
支持 ClassicLink	将 Auto Scaling 组中的 EC2-Classic 实例连接到 VPC，让这些链接的 EC2-Classic 实例与 VPC 中的实例能够利用私有 IP 地址进行通信。有关更多信息，请参阅 将 EC2-Classic 实例链接到 VPC (p. 26) 。	2015 年 1 月 19 日
生命周期挂钩	在您对新启动的实例或正在终止的实例执行操作时，请将这些实例保持在等待状态。有关更多信息，请参阅 Auto Scaling 生命周期挂钩 (p. 83) 。	2014 年 7 月 30 日
分离实例	从 Auto Scaling 组分离实例。有关更多信息，请参阅 从 Auto Scaling 组中分离 EC2 实例 (p. 58) 。	2014 年 7 月 30 日

功能	描述	发行日期
将实例置于 Standby 状态	将处于 InService 状态的实例置于 Standby 状态。有关更多信息，请参阅 临时从 Auto Scaling 组中删除实例 (p. 88) 。	2014 年 7 月 30 日
管理标签	使用 AWS 管理控制台管理您的 Auto Scaling 组。有关更多信息，请参阅 标记 Auto Scaling 组和实例 (p. 39) 。	2014 年 5 月 01 日
支持专用实例	通过在创建启动配置时指定部署租期属性启动专用实例。有关更多信息，请参阅 实例部署租期 (p. 25) 。	2014 年 4 月 23 日
从 EC2 实例创建组或启动配置	使用 EC2 实例创建 Auto Scaling 组或启动配置。有关使用 EC2 实例创建启动配置的信息，请参阅 使用 EC2 实例创建启动配置 (p. 20) 有关使用 EC2 实例创建 Auto Scaling 组的信息，请参阅 使用 EC2 实例创建 Auto Scaling 组 (p. 37) 。	2014 年 1 月 02 日
挂载实例	您可以通过将 EC2 实例挂载到现有 Auto Scaling 组，为该实例启用 Auto Scaling。有关更多信息，请参阅 将 EC2 实例附加到 Auto Scaling 组 (p. 55) 。	2014 年 1 月 02 日
查看账户限制	查看对您的账户的 Auto Scaling 资源限制。有关更多信息，请参阅 Auto Scaling 限制 (p. 9) 。	2014 年 1 月 02 日
Auto Scaling 的控制台支持	通过 AWS 管理控制台访问 Auto Scaling。有关更多信息，请参阅 Auto Scaling 入门 (p. 11) 。	2013 年 12 月 10 日
分配公有 IP 地址	将公有 IP 地址分配给启动至 VPC 的实例。有关更多信息，请参阅 在 VPC 中启动 Auto Scaling 实例 (p. 24) 。	2013 年 9 月 19 日
实例终止策略	为 Auto Scaling 指定实例终止策略，用于终止 EC2 实例。有关更多信息，请参阅 控制 Auto Scaling 在缩小过程中终止哪些实例 (p. 78) 。	2012 年 9 月 17 日
对 IAM 角色的支持	用 IAM 实例配置文件启动 EC2 实例。您可以使用此功能为实例指定 IAM 角色，允许您的其他应用程序安全地访问其他 AWS 服务。有关更多信息，请参阅 使用 IAM 角色启动 Auto Scaling 实例 (p. 122) 。	2012 年 6 月 11 日
支持竞价型实例	可以通过在启动配置中指定竞价型实例出价，在 Auto Scaling 组中请求竞价型实例。有关更多信息，请参阅 在 Auto Scaling 组中启动竞价型实例 (p. 28) 。	2012 年 6 月 7 日
对组和实例进行标记	标记 Auto Scaling 组并指定将标签也应用于在创建标签之后启动的 EC2 实例。有关更多信息，请参阅 标记 Auto Scaling 组和实例 (p. 39) 。	2012 年 1 月 26 日

功能	描述	发行日期
支持 Amazon SNS	<p>每当 Auto Scaling 启动或终止 EC2 实例时，都使用 Amazon SNS 接收通知。有关更多信息，请参阅 在 Auto Scaling 组扩展时获取 SNS 通知 (p. 106)。</p> <p>Auto Scaling 还增加了以下新功能：</p> <ul style="list-style-type: none"> 使用 cron 语法设置重复扩展活动的的能力。关于详细信息，请参见 PutScheduledUpdateGroupAction API 命令。 增加了新的配置设置，使您无需将已启动的实例添加到负载均衡器 (LoadBalancer) 即可进行扩展。有关详细信息，请参阅 ProcessType API 数据类型。 ForceDelete 命令中的 DeleteAutoScalingGroup 标志可通知 Auto Scaling 删除 Auto Scaling 组以及与其相关的实例，无需先等待实例终止。关于详细信息，请参见 DeleteAutoScalingGroup API 命令。 	2011 年 7 月 20 日
计划扩展操作	您现在可以创建计划的扩展操作。有关更多信息，请参阅 计划的扩展 (p. 60) 。	2010 年 12 月 2 日
支持 Amazon VPC	增加了对 Amazon VPC 的支持。有关更多信息，请参阅 在 VPC 中启动 Auto Scaling 实例 (p. 24) 。	2010 年 12 月 2 日
支持 HPC 群集	增加了对于高性能计算 (HPC) 群集的支持。	2010 年 12 月 2 日
支持健康检查	增加了对 Auto Scaling 所管理的 EC2 实例进行 Elastic Load Balancing 健康检查的支持。有关更多信息，请参阅 结合使用 ELB 运行状况检查和 Auto Scaling (p. 45) 。	2010 年 12 月 2 日
支持 CloudWatch 警报	删除了较旧的触发机制，并重新设计了 Auto Scaling 以便使用 CloudWatch 警报功能。有关更多信息，请参阅 动态扩展 (p. 63) 。	2010 年 12 月 2 日
暂停和恢复扩展	现在您可以暂停和恢复扩展过程。	2010 年 12 月 2 日
支持 IAM	Auto Scaling 现在支持 IAM。有关更多信息，请参阅 控制对您 Auto Scaling 资源的访问权 (p. 114) 。	2010 年 12 月 2 日