
AWS 入门

部署 Web 应用程序



AWS 入门: 部署 Web 应用程序

Copyright © 2017 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Deploying a Web App	1
AWS Elastic Beanstalk	1
注册应用程序	1
Amazon DynamoDB	2
Amazon Simple Notification Service	2
设置	3
Sign Up for AWS	3
下载应用程序代码	3
创建 IAM 角色	4
步骤 1：创建 DynamoDB 表	5
步骤 2：创建 SNS 主题	6
步骤 3：部署应用程序	8
准备源数据包	8
创建应用程序	8
测试应用程序	10
解决部署问题	10
步骤 4：更改应用程序配置（可选）	12
步骤 5：清除	14
删除 AWS Elastic Beanstalk 资源	14
删除 Amazon DynamoDB 表	14
删除 Amazon SNS 资源	15
更多 AWS 部署选项	16
相关资源	17

使用 Elastic Beanstalk 部署 Web 应用程序

使用 AWS，您可快速部署 Web 应用程序，然后将其部署到按需扩展的云环境中。由于具有多种 AWS 部署服务可供选择，您可创建为您提供正确的自动化和控制组合的部署解决方案。

在本教程中，我们假定您在使用一个新的、尚未做好生产准备的 Web 应用程序，但同时您计划部署一个小的占位符应用程序来收集为了了解更多信息而注册的网站访客的联系信息。此注册应用程序将帮助您接触潜在客户 - 可能成为早期采用者或参与封闭的测试版测试的人。

以下是对 AWS Elastic Beanstalk 和我们将使用的其他技术的快速简介。（要直接进入本教程的动手部分，请跳至下一节。）

AWS Elastic Beanstalk

Elastic Beanstalk 是一个高级部署工具，可帮助您在几分钟内将应用程序从桌面部署到网站上。Elastic Beanstalk 可处理您的托管环境的各种细节（容量预置、负载均衡、扩展和应用程序运行状况监控），因此您不必费心。

Elastic Beanstalk 支持使用 Java、PHP、.NET、Node.js、Python 和 Ruby 开发的应用程序以及每种语言的不同容器类型。容器定义了将在指定环境中使用的基础设施和软件堆栈。在部署您的应用程序时，Elastic Beanstalk 将配置一个或多个 AWS 资源，例如 EC2 实例。在 EC2 实例上运行的软件堆栈取决于容器类型。例如，Elastic Beanstalk 支持两种容器类型的 Node.js：32 位的 Amazon Linux 映像和 64 位的 Amazon Linux 映像。每种 Node.js 运行一个为托管 Node.js 应用程序定制的软件堆栈。

您可通过使用 AWS 管理控制台、AWS Command Line Interface (AWS CLI) 或 eb（专为 Elastic Beanstalk 设计的高级 CLI）与 Elastic Beanstalk 交互。在本教程中，我们将使用 AWS 管理控制台。

注册应用程序

在本教程中，我们将部署一个示例应用程序，让客户提交联系信息并表达对预览您正开发的假设的 Web 应用程序的兴趣。

此应用程序是基于 [Node.js](#)（一个使用服务器端 JavaScript 构建网络应用程序的平台）构建的。Node.js 包含一个库和一个运行时。此运行时是由 [V8 JavaScript Engine](#) 提供的。

Node.js 是围绕非阻塞、事件驱动型 I/O 模型（可用于创建可高度扩展的 Web 服务器）设计的。我们的应用程序使用两个外部节点模块：[Express](#)（一个 Web 应用程序框架）和 [Jade](#)（一个可用于创建 HTML 文档的 Node.js 模板引擎）。

AWS 提供了一个 Node.js SDK，它通过提供适用于 AWS 的 JavaScript 对象来帮助降低代码编写的难度。我们已使用 Node.js SDK 构建来我们的示例应用程序。要了解有关 Node.js SDK 的更多信息，请参阅 [Node.js 中的适用于 JavaScript 的 AWS SDK](#)。

为了使我们的应用程序外观好看，我们使用 [Bootstrap](#)（一个作为 Twitter 项目开始的移动优先的前端框架）。

Amazon DynamoDB

我们将使用 Amazon DynamoDB（一种 NoSQL 数据库服务）存储用户提交的联系信息。DynamoDB 是一个无架构数据库，因此您需要仅指定主键属性。我们将使用 `email` 字段作为我们的应用程序的键。

Amazon Simple Notification Service

我们希望在客户提交表格时得到通知，因此我们将使用 Amazon Simple Notification Service (Amazon SNS)（一种可通过多种协议发送通知的消息推送服务）。对于我们的应用程序，我们会将通知推送到一个电子邮件地址。

设置

在开始本教程之前，请完成下列任务。

任务

- [Sign Up for AWS](#) (p. 3)
- [下载应用程序代码](#) (p. 3)
- [创建 IAM 角色](#) (p. 4)

Sign Up for AWS

When you sign up for Amazon Web Services (AWS), your AWS account is automatically signed up for all services in AWS and you can start using them immediately. You are charged only for the services that you use.

If you created your AWS account less than 12 months ago, you can get started with AWS for free. For more information, see [AWS Free Tier](#).

If you have an AWS account already, skip to the next step. If you don't have an AWS account, use the following procedure to create one.

To create an AWS account

1. 打开 <https://aws.amazon.com/>，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。

下载应用程序代码

您可从 AWS Labs GitHub 存储库下载应用程序代码。打开 [eb-node-express-signup repo](#)，然后选择 Download ZIP。

您在完成教程的过程中将会对代码进行一些更改，因此需要将 `eb-node-express-signup-master.zip` 解压缩。应用程序代码存储在 `eb-node-express-signup-master` 目录中。

创建 IAM 角色

接下来，您需要创建一个 IAM 角色来为您的应用程序授予权限，允许其发布 Amazon SNS 通知并将项目放入您的 DynamoDB 表中。在创建 AWS Elastic Beanstalk 环境时，您可以将角色应用到运行您的应用程序的 EC2 实例。

创建带有所需策略的 IAM 角色

1. 打开 AWS Identity and Access Management (IAM) <https://console.aws.amazon.com/iam> 中的控制台。
2. 在导航窗格中，选择 Policies。
3. 选择 Create Policy 以启动“Create Policy Wizard”。
4. 在 Create Policy 页面上的 Create Your Own Policy 旁，选择 Select。
5. 在 Review Policy 页上，输入有助于在日后识别策略的策略名称。例如，**ElasticBeanstalkStartupPolicy**。
6. 从 `eb-node-express-signup-master` 目录中打开 `iam_policy.json` 文件并复制其内容。将内容粘贴至策略文档框中。
7. 准备好之后，选择 Create Policy。
8. 在导航窗格中，选择 Roles。
9. 选择 Create New Role 以启动“Create Role Wizard”。
10. 在 Set Role Name 页上的 Role Name 框中，输入角色名称。选择 Next Step。
11. 在 Select Role Type 页上的 Amazon EC2 旁，选择 Select 以允许 EC2 实例代表您调用 AWS 服务。
12. 在 Attach Policy 页上的 Filter 旁，选择 Policy Type，然后选择 Customer Managed Policies。
13. 在您之前创建的策略的名称旁边，选中复选框。选择 Next Step。
14. 在 Review 页面上，验证输入的信息，然后选择 Create Role。

有关权限的更多信息，请参阅 AWS Elastic Beanstalk 开发人员指南中的 <http://docs.aws.amazon.com/elasticbeanstalk/latest/dg/concepts-roles.html>。

步骤 1：创建 DynamoDB 表

我们的注册应用程序使用 DynamoDB 表存储用户提交的联系信息。

创建 DynamoDB 表

1. 通过以下网址打开 DynamoDB 控制台：<https://console.aws.amazon.com/dynamodb/home>。
2. 在菜单栏中，选择要在其中部署应用程序的区域。在本教程中，选择 US West (Oregon)（美国西部（俄勒冈））。
3. 选择 Create Table。
4. 对于 Table Name，键入 **gsg-signup-app**。
5. 对于 Primary Key，键入 **email**。选择 Create。
6. 将表名和区域添加到应用程序的配置文件，如下所示：
 - a. 在文本编辑器中打开 `app_config.json` 文件。
 - b. 将 `STARTUP_SIGNUP_TABLE` 的值更改为 **gsg-signup-app**：

```
"STARTUP_SIGNUP_TABLE": "gsg-signup-app",
```

- c. 对于美国西部（俄勒冈）区域，将 `AWS_REGION` 的值更改为 `us-west-2`：

```
"AWS_REGION": "us-west-2",
```

- d. 保存您对文件的编辑。您可使文件保持打开状态，因为我们将下一步中对其再次进行更改。

步骤 2：创建 SNS 主题

我们的注册应用程序将在每次有用户注册时通知您。当注册表的数据写入 DynamoDB 表后，该应用程序将向您发送一条 SNS 通知。首先，您需要创建一个 SNS 主题（这是一个通知流），然后需要创建一个告知 SNS 通知的发送位置和发送方式的订阅。

设置 Amazon SNS 通知

1. 通过以下网址打开 Amazon SNS 控制台：<https://console.aws.amazon.com/sns/v2/home>。
2. 选择 Create Topic。
3. 对于 Topic name，键入 **signup-notifications**。选择 Create topic。
4. 将 SNS 主题的唯一标识符添加到应用程序的配置文件，如下所示：
 - a. 在主题详细信息页上，复制 Topic ARN (主题 ARN) 中的字符串。请注意，每个 Amazon 资源名称 (ARN) 都采用以下语法：

```
arn:aws:[service]:[region]:[account]:resourceType/resourcePath
```

- b. 在文本编辑器中打开 `app_config.json` 文件（如果尚未打开）。
- c. “NEW_SIGNUP_TOPIC”的值以一个空字符串开始。

```
"NEW_SIGNUP_TOPIC": "",
```

插入新的 SNS 主题的名称，如下所示。

```
"NEW_SIGNUP_TOPIC": "arn:aws:sns:us-west-2:123456789012:signup-notifications",
```

- d. 您最终的配置文件看上去应与下面类似。保存对文件的编辑，然后关闭文件。

```
{
  "AWS_REGION": "us-west-2",
  "STARTUP_SIGNUP_TABLE": "my-signup-table",
  "NEW_SIGNUP_TOPIC": "arn:aws:sns:us-west-2:123456789012:signup-notifications"
}
```

5. 选择 Create Subscription。
6. 在 Create Subscription (创建订阅) 对话框中，选择 Email (电子邮件) 作为协议。在 Endpoint (终端节点) 框中，输入用于接收通知的电子邮件地址。（在本教程中，输入您自己的电子邮件地址，因为您必须回复确认邮件。）选择 Create Subscription。在确认对话框中，选择 Close。

7. 转到您的电子邮件应用程序。您将收到一封标题为“AWS Notification — Subscription Confirmation”(AWS 通知 - 订阅确认) 的邮件。打开此邮件，然后选择链接以确认您的订阅。要验证您的订阅是否已确认，请在 SNS 控制台刷新此页。

步骤 3：使用 AWS Elastic Beanstalk 部署应用程序

您可使用 Elastic Beanstalk 轻松部署注册应用程序。将一个应用程序版本（例如，一个 .zip 文件）上传到 Elastic Beanstalk，然后提供有关该应用程序的一些信息。Elastic Beanstalk 将启动一个环境并配置运行您的代码所需的 AWS 资源（例如 Amazon EC2 实例）。在您的环境启动之后，您可管理您的环境并部署新的应用程序版本。

任务

- [准备源数据包 \(p. 8\)](#)
- [创建应用程序 \(p. 8\)](#)
- [测试应用程序 \(p. 10\)](#)
- [解决部署问题 \(p. 10\)](#)

准备源数据包

Elastic Beanstalk 要求您的应用程序捆绑为一个 .zip 或 .war 文件。捆绑不得包含顶级文件夹，因此您必须压缩单独的应用程序文件，而不是压缩包含这些文件的目录。

准备源数据包

1. 在类似 Windows 资源管理器或 Mac OS X Finder 的工具中打开应用程序文件夹 (eb-node-express-signup-master)。
2. 选择文件夹中的所有项目，包括子文件夹。请勿选择顶级文件夹。
3. 右键单击以选择选定项目并选择用于压缩它们的选项，例如 Send to > Compressed (zipped) Folder (Windows) 或 Compress Items (Finder)。

有关使用不同工具压缩文件的更多信息，请参阅 AWS Elastic Beanstalk 开发人员指南 中的 [创建应用程序源数据包](#)。

创建应用程序

创建 Elastic Beanstalk 应用程序并将应用程序版本部署到新环境。

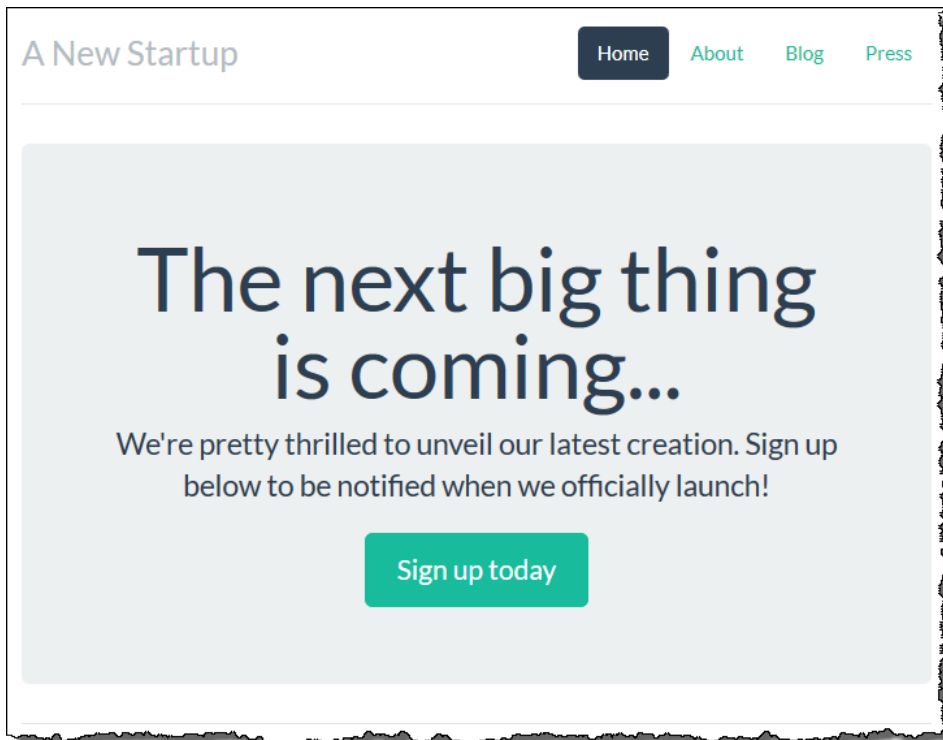
删除应用程序和环境

1. 通过以下网址打开 Elastic Beanstalk 控制台：<https://console.aws.amazon.com/elasticbeanstalk/>。
2. 选择 Create New Application。
3. 在 Application Information (应用程序信息) 页上，输入应用程序的名称。在本教程中，让我们将该应用程序命名为 my-startup-app。选择 Next。
4. 在 New Environment 页上选择 Create web server。
5. 在环境类型页，将预定义配置设为 Node.js，然后选择下一个。
6. 在 Application Version (应用程序版本) 页上，选择 Upload your own (上传您自己的应用程序版本)，选择 Choose File (选择文件)，然后选择您创建的源包。选择 Next。
7. 在 Environment Information (环境信息) 页上，输入环境名称和唯一的环境 URL。选择 Check availability (检查可用性) 以确保您的 URL 可用。选择 Next。
8. 在其他资源页面上，选择下一个。
9. 在配置详细信息页面，通过选择下一个来对所有设置使用默认值。
10. 在 Environment Tags (环境标签) 页上，选择 Next (下一步)。
11. 在许可 页面，将实例配置文件设置为您在 [创建 IAM 角色 \(p. 4\)](#) 中创建的角色。请勿更改服务角色的 值。
12. 选择 Next。如果系统提示，请接受提示以创建默认服务角色并将其分配至您的环境。

Note

创建角色需要额外的权限。参阅<http://docs.aws.amazon.com/elasticbeanstalk/latest/dg/concepts-roles.html#concepts-roles-user> Elastic Beanstalk 开发人员指南中的 Elastic Beanstalk User 策略以了解详细信息。

13. 在 Review (审核) 页上，验证您的设置，然后选择 Launch (启动)。在 AWS Elastic Beanstalk 仪表板上，您可在 Elastic Beanstalk 创建环境和部署应用程序时进行实时观察。此过程需要几分钟时间。

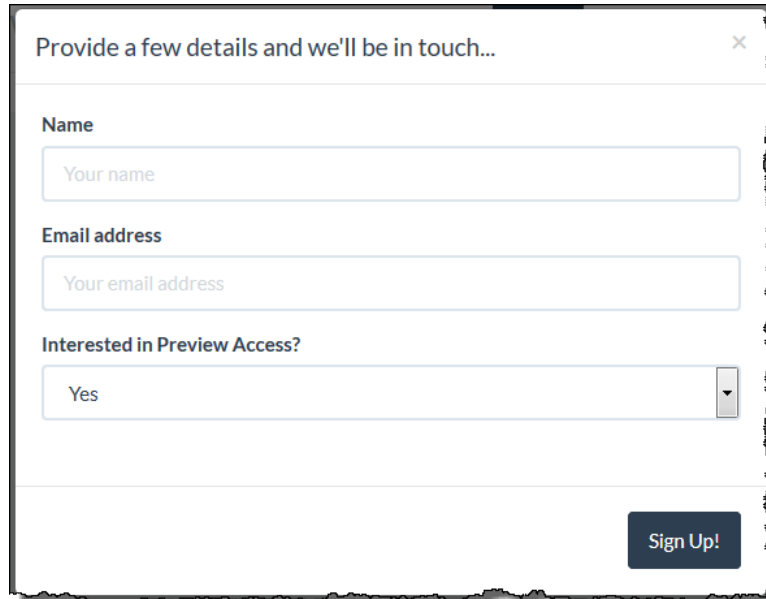


测试应用程序

在部署完成并且环境运行状况显示为绿色时，打开应用程序的 URL。

gsg-deployment (Environment ID: e-xau5iymnmu URL: gsg-deployment-us-west-2.elasticbeanstalk.com)

您可通过填写表格并确认收到通知来测试注册应用程序。



解决部署问题

如果您执行了所有步骤，打开了 URL，但未获取到任何应用程序，则存在部署问题。对于我们的在 nginx 服务器上运行的示例应用程序，部署问题可能导致“502 错误的网关”消息。“502”消息的信息不是很丰富。要解决部署问题，您可能需要使用 Elastic Beanstalk 提供的日志。

例如，让我们假定，在更新 `app_config.json` 的过程中，您意外地漏掉了引号。然后，在您完成部署后，出现了 502 错误。如何查找并解决此问题？

当然，您在开发过程中会尽力捕获此类错误。但如果将某个错误带到了生产阶段，或者您只是想要更新应用程序，Elastic Beanstalk 可以让重新部署变得快速而轻松。

解决部署问题

1. 在 Elastic Beanstalk 控制台中，在您的环境的导航窗格中，选择 Logs。
2. 在“Logs”页上，选择 Snapshot Logs。等待环境更新，然后选择 View log file。在日志文件中，您看到在部署期间和运行时服务器端发生了什么情况。有大量要梳理的材料，在本教程中我们不会涉及日志的各个部分。但如果您确实在配置文件中漏掉了引号，则当您日志滚动至 `/var/log/nodejs/nodejs.log` 时，将会发现与下面类似的错误：

```
SyntaxError: Unexpected token u
    at Object.parse (native)
    at Object.<anonymous> (/var/app/current/server.js:23:15)
    at Module._compile (module.js:449:26)
    at Object.Module._extensions..js (module.js:467:10)
```

```
at Module.load (module.js:356:32)
at Function.Module._load (module.js:312:12)
at Module.runMain (module.js:492:10)
at process.startup.processNextTick.process._tickCallback (node.js:245:9)

undefined:2
  "AWS_REGION": us-west-2",^
```

在此情况下，将显示“Unexpected token u”(意外标记 u) 消息，因为分析器在字符串 `us-west-2` 中需要一个引号而不是“u”。既然我们找到了问题，我们就可解决它。

3. 如果您实际上正在解决此问题，则返回到您的本地环境中的应用程序代码，修复缺少的引号。然后创建一个新的 `.zip` 文件以进行上传。
4. 要重新部署应用程序，请转至 Elastic Beanstalk 仪表板，选择 Upload and Deploy，然后选择更新的 `.zip` 文件。
5. 将版本标签更新为新的内容。例如，如果您的第一个部署标记为“Archive”，则可将第二个部署标记为“Archive-2”。
6. 选择 Deploy。Elastic Beanstalk 将更新环境。
7. 当环境进行了更新并显示为绿色后，使用应用程序 URL 测试您的应用程序。

步骤 4：更改应用程序配置（可选）

我们的 Web 应用程序使用环境变量 `theme` 来控制所应用的 CSS。`server.js` 文件包含以下语句以使此环境变量对应用程序可用：

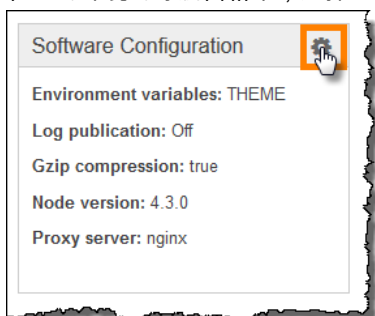
```
app.locals.theme = process.env.THEME;
```

更改此环境变量的设置将更改应用程序的外观。有关更多信息，请参阅应用程序代码中的 `public/static/bootstrap/css/theme` 和 `views/layout.jade`。

更新此环境变量最简单的方式是使用 Elastic Beanstalk 更新环境的配置。

使用 Elastic Beanstalk 更改应用程序主题

1. 打开 Elastic Beanstalk 控制台。
2. 在您的环境的导航窗格中，选择 Configuration，然后打开 Software Configuration。



3. 在 Environment Properties (环境属性) 下，找到 **THEME** 环境变量。将值从默认值 (`flatly`) 更改为下列值之一，然后选择 Save。
 - `amelia`
 - `default`
 - `slate`
 - `united`

Static Files

To improve performance, you can configure Apache or Nginx to serve static files from a set of directories inside your web application. [Learn more.](#)

Virtual Path (Example: /assets)	Directory (Example: /static/assets)
/static/	/public/static/ ✕
<input type="text"/>	<input type="text"/> +

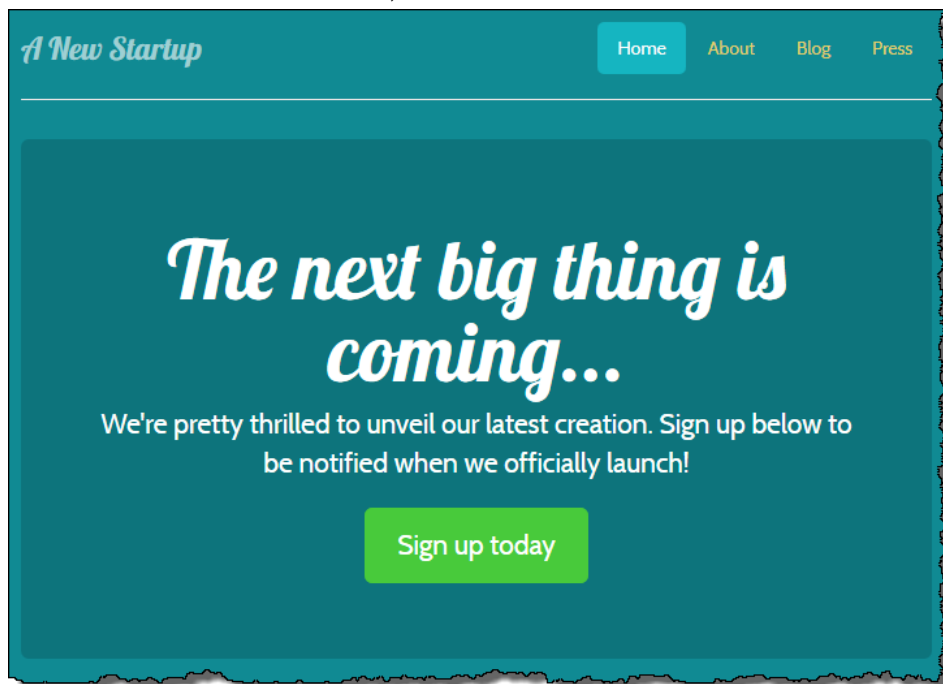
Environment Properties

The following properties are passed into the application as environment variables. [Learn more.](#)

Property Name	Property Value
THEME	amelia ✕
<input type="text"/>	<input type="text"/> +

Cancel Apply

4. 在 Elastic Beanstalk 更新完环境后，选择应用程序的 URL。应用程序将具有新的外观。



步骤 5：清除

为了防止您的账户产生额外费用，您应清除您在本教程中使用 Elastic Beanstalk 创建的应用程序和环境。

主题

- [删除 AWS Elastic Beanstalk 资源 \(p. 14\)](#)
- [删除 Amazon DynamoDB 表 \(p. 14\)](#)
- [删除 Amazon SNS 资源 \(p. 15\)](#)

删除 AWS Elastic Beanstalk 资源

要删除 AWS Elastic Beanstalk 资源，请执行以下步骤：

要删除 AWS Elastic Beanstalk 资源

1. 通过以下网址打开 Elastic Beanstalk 控制台：<https://console.aws.amazon.com/elasticbeanstalk>。
2. 删除所有应用程序版本，如下所示：
 - a. 选择您的应用程序名称，然后选择导航区域的应用程序版本。
 - b. 选择您在本教程中创建的所有应用程序版本，然后选择 Delete (删除)。
 - c. 当系统提示进行确认时，选择 Delete。
 - d. 选择完成。
3. 按如下所示方式终止环境：
 - a. 选择 Environments (环境)，然后选择您的环境。
 - b. 选择 Actions (操作)，Terminate Environment (终止环境)。
 - c. 当系统提示您确认时，选择 Terminate。
4. 选择 Actions，然后选择 Delete Application。当系统提示进行确认时，选择 Delete。

删除 Amazon DynamoDB 表

要删除 Amazon DynamoDB 表，请执行以下步骤：

要删除 Amazon DynamoDB 表。

1. 通过以下网址打开 DynamoDB 控制台：<https://console.aws.amazon.com/dynamodb/>。
2. 选择您在第 1 步中创建的表。
3. 选择 Delete Table。
4. 选择删除该表，然后选择删除。

删除 Amazon SNS 资源

要删除 Amazon SNS 资源，请执行以下步骤：

要删除 Amazon SNS 订阅

1. 通过以下网址打开 Amazon SNS 控制台：<https://console.aws.amazon.com/sns/v2/home>。
2. 在左侧导航窗格中，选择 Subscriptions。
3. 选择在您订阅列表中的主题，选择操作，然后选择 删除订阅。
4. 选择 Delete。

要删除 Amazon DynamoDB 表。

1. 通过以下网址打开 DynamoDB 控制台：<https://console.aws.amazon.com/dynamodb/>。
2. 选择您在第 1 步中创建的表。
3. 选择 Delete Table。
4. 选择删除该表，然后选择删除。

更多 AWS 部署选项

本教程主要介绍 AWS Elastic Beanstalk 的基本功能。但是，Elastic Beanstalk 具有许多其他功能，它只是随 AWS 提供的部署解决方案之一。此外，您可使用 AWS OpsWorks 部署和管理应用程序，如果您想要自助体验，可使用 Amazon Elastic Compute Cloud、AWS CloudFormation、Amazon CloudWatch 和 Auto Scaling 构建您自己的部署框架。要了解有关选择正确的部署解决方案的更多信息，请参阅 [AWS 的应用程序管理](#)。

要了解有关 Elastic Beanstalk 和其他 AWS 部署选项的更多信息，请查看下列资源。

有关 Elastic Beanstalk 的更多信息

- 有关使用 Elastic Beanstalk 管理应用程序的完整文档，请参阅 [AWS Elastic Beanstalk 开发人员指南](#)。
- 要了解有关 Elastic Beanstalk CLI 的信息，请参阅 [Eb 命令行接口](#)。
- 要了解如何使用 eb 部署 Node.js 应用程序，请参阅[使用 Eb 和 Git 在 Node.js 中部署 AWS Elastic Beanstalk 应用程序](#)。
- 要了解有关 Elastic Beanstalk 应用程序的组件和架构的更多信息，请参阅 [AWS Elastic Beanstalk 的工作原理](#)。
- 通过在源数据包中包含一个配置文件，您可在部署应用程序的同时自定义您的环境。要了解更多信息，请参阅[自定义和配置 AWS Elastic Beanstalk 环境](#)。

更多 AWS 解决方案

- 要了解有关 AWS OpsWorks 的更多信息，请参阅 [AWS OpsWorks 用户指南](#)。
- 要了解有关您一般用于自助部署的独立服务的更多信息，请参阅 [Amazon EC2 入门指南](#)、[AWS CloudFormation 用户指南](#)、[Amazon CloudWatch Getting Started Guide](#)和[Auto Scaling 入门指南](#)。
- AWS Toolkit for Visual Studio 包含一个部署工具。要了解更多信息，请参阅[独立部署工具](#)。
- AWS Toolkit for Eclipse 还提供了部署选项。要了解更多信息，请参阅 [AWS Toolkit for Eclipse 入门](#)。

相关资源

下表列出了在您使用 AWS 时可为您提供帮助的一些 AWS 资源。

资源	说明
AWS 产品与服务	有关 AWS 提供的产品与服务的信息。
AWS 文档	每种 AWS 产品的官方文档，其中包括服务简介、服务功能和 API 参考。
AWS 开发论坛	社区形式的论坛，用于讨论与 Amazon Web Services 有关的技术问题。
联系我们	账单、事件和滥用等账户问题的一个中心联络点。关于技术问题，请使用论坛。
AWS 支持中心	用于创建和管理 AWS Support 案例的中心。还提供指向其他有用资源的链接，如论坛、技术常见问题、服务运行状况以及 AWS Trusted Advisor。
AWS Support	AWS Support 主页，是一对一的快速响应支持通道，可帮助您在云中构建和运行应用程序。
AWS 架构中心	提供必要的指导和最佳实践，用于在 AWS 云中构建高度可扩展且高度可靠的应用程序。这些资源有助于您了解 AWS 平台及其服务和功能。它们还提供架构指南，用于设计和实施在 AWS 基础设施中运行的系统。
AWS 安全中心	提供有关安全功能和资源的信息。
AWS 成本中心	用于访问各种信息、工具和资源，以便比较 Amazon Web Services 和其他备选 IT 基础设施产品的成本。
AWS 技术白皮书	提供技术白皮书，涵盖架构、安全性和经济性等主题。这些白皮书由 Amazon 团队、客户和解决方案提供商编写。
AWS 博客	提供涵盖新服务以及现有服务更新的博客文章。
AWS 播客	提供涵盖新服务、现有服务和提示的播客。