

---

# Elastic Load Balancing

用户指南



## Elastic Load Balancing: 用户指南

## Table of Contents

什么是 Elastic Load Balancing ? .....	1
负载均衡器优势 .....	1
Elastic Load Balancing 的功能 .....	1
访问 Elastic Load Balancing .....	1
相关服务 .....	1
定价 .....	2
Elastic Load Balancing 的工作原理 .....	3
可用区与负载均衡器节点 .....	3
跨区域负载均衡 .....	3
请求路由选择 .....	4
路由算法 .....	5
HTTP 连接 .....	5
HTTP 标头 .....	5
HTTP 标头限制 .....	6
负载均衡器模式 .....	6
入门 .....	7
创建 应用程序负载均衡器 .....	7
创建 网络负载均衡器 .....	7
创建 传统负载均衡器 .....	7
身份验证和访问控制 .....	8
使用 IAM 策略授予权限 .....	8
Elastic Load Balancing 的 API 操作 .....	8
Elastic Load Balancing 资源 .....	9
Elastic Load Balancing 的资源级权限 .....	10
Elastic Load Balancing 的条件密钥 .....	12
预定义的 AWS 托管策略 .....	13
API 权限 .....	14
2015-12-01 API 所需的权限 .....	14
2012-06-01 API 所需的权限 .....	14
服务相关角色 .....	15
服务相关角色授予的权限 .....	16
创建服务相关角色 .....	16
编辑服务相关角色 .....	17
删除服务相关角色 .....	17
迁移 .....	18
步骤 1：创建新负载均衡器 .....	18
选项 1：使用迁移向导进行迁移 .....	18
选项 2：使用负载均衡器复制实用程序进行迁移 .....	19
选项 3：手动迁移 .....	19
步骤 2：逐步将流量重定向到您的新负载均衡器 .....	19
步骤 3：更新对您的 传统负载均衡器 的引用 .....	20
步骤 4：删除 传统负载均衡器 .....	20

# 什么是 Elastic Load Balancing ?

Elastic Load Balancing 跨多个可用区中的多个目标（如 Amazon EC2 实例、容器和 IP 地址）分发传入应用程序或网络流量。Elastic Load Balancing 会在应用程序的传入流量随时间的推移发生更改时扩展负载均衡器，并可自动扩展以处理大部分工作负载。

## 负载均衡器优势

负载均衡器跨多个计算资源（如虚拟服务器）分布工作负载。使用负载均衡器可提高您的应用程序的可用性和容错性。

可以根据需求变化在负载均衡器中添加和删除计算资源，而不会中断应用程序的整体请求流。

您可以配置运行状况检查，这些检查可用于监控计算资源的运行状况，以便负载均衡器只能将请求发送到正常运行的目标。此外，您可以将加密和解密的工作交给负载均衡器完成，以使您的计算资源能够专注于完成主要工作。

## Elastic Load Balancing 的功能

Elastic Load Balancing 支持三种类型的负载均衡器：Application Load Balancer、Network Load Balancer 和 Classic Load Balancer。可以根据应用程序需求选择负载均衡器。有关更多信息，请参阅 [Elastic Load Balancing 产品比较](#)。

有关使用每种负载均衡器的更多信息，请参阅 [Application Load Balancer 用户指南](#)、[Network Load Balancer 用户指南](#) 和 [Classic Load Balancer 用户指南](#)。

## 访问 Elastic Load Balancing

可以使用以下任意接口创建、访问和管理负载均衡器：

- AWS 管理控制台— 提供您用来访问 Elastic Load Balancing 的 Web 界面。
- AWS 命令行界面 (AWS CLI) — 提供了用于众多 AWS 服务（包括 Elastic Load Balancing）的命令，且在 Windows、Mac 和 Linux 上受支持。有关更多信息，请参阅 [AWS Command Line Interface](#)。
- AWS 软件开发工具包 — 提供了特定于语言的 API，并关注许多连接详细信息，例如计算签名、处理请求重试和错误处理。有关更多信息，请参阅 [AWS 开发工具包](#)。
- 查询 API — 提供了您使用 HTTPS 请求调用的低级别 API 操作。使用查询 API 是用于访问 Elastic Load Balancing 的最直接的方式，但需要您的应用程序处理低级别的详细信息，例如生成哈希值以签署请求以及进行错误处理。有关更多信息，请参阅下列内容：
  - Application Load Balancer 和 Network Load Balancer — [API 版本 2015-12-01](#)
  - Classic Load Balancer — [API 版本 2012-06-01](#)

## 相关服务

Elastic Load Balancing 可与以下服务一起使用来提高应用程序的可用性和可扩展性。

- Amazon EC2 — 在云中运行应用程序的虚拟服务器。您可以将负载均衡器配置为将流量路由到您的 EC2 实例。有关更多信息，请参阅 [Amazon EC2 用户指南（适用于 Linux 实例）](#) 或 [Amazon EC2 用户指南（适用于 Windows 实例）](#)。
- Amazon EC2 Auto Scaling — 确保运行所需数量的实例（即使实例失败也是如此），并可让您根据实例需求的变化自动增加或减少实例数。如果您使用 Elastic Load Balancing 启用 Auto Scaling，则 Auto Scaling 启动的实例将自动向负载均衡器注册，并且 Auto Scaling 终止的实例将自动从负载均衡器注销。有关更多信息，请参阅 [Amazon EC2 Auto Scaling 用户指南](#)。
- AWS Certificate Manager — 在创建 HTTPS 侦听器时，您必须指定由 ACM 提供的证书。负载均衡器使用证书终止连接并解密来自客户端的请求。
- Amazon CloudWatch — 使您能够监控负载均衡器并执行所需操作。有关更多信息，请参阅 [Amazon CloudWatch 用户指南](#)。
- Amazon ECS — 使您能够在 EC2 实例集群上运行、停止和管理 Docker 容器。您可以将负载均衡器配置为将流量路由到您的容器。有关更多信息，请参阅 [Amazon Elastic Container Service Developer Guide](#)。
- Route 53 — 通过将域名（如 `www.example.com`）转换为计算机相互连接所用的数字 IP 地址（如 `192.0.2.1`），以一种可靠且经济的方式将访问者路由至网站。AWS 将向您的资源（如负载均衡器）分配 URL。不过，您可能希望使用方便用户记忆的 URL。例如，您可以将域名映射到负载均衡器。有关更多信息，请参阅 [Amazon Route 53 开发人员指南](#)。
- AWS WAF — 您可以结合使用 AWS WAF 和 应用程序负载均衡器 以根据 Web 访问控制列表 (Web ACL) 中的规则允许或阻止请求。有关更多信息，请参阅 [AWS WAF 开发人员指南](#)。

## 定价

利用负载均衡器，您可以按实际用量付费。有关更多信息，请参阅 [Elastic Load Balancing 定价](#)。

# Elastic Load Balancing 的工作原理

负载均衡器接受来自客户端的传入流量并将请求路由到一个或多个可用区中的已注册目标 (例如 EC2 实例)。负载均衡器还会监控已注册目标的运行状况, 并确保它只将流量路由到正常运行的目标。当负载均衡器检测到运行不正常的目标时, 它会停止向该目标路由流量, 然后在检测到目标恢复正常运行后继续向其路由流量。

您可通过指定一个或多个侦听器 将您的负载均衡器配置为接受传入流量。侦听器是用于检查连接请求的进程。它配置了用于从客户端连接到负载均衡器的协议和端口号以及用于从负载均衡器连接到目标的协议和端口号。

Elastic Load Balancing 支持三种类型的负载均衡器: Application Load Balancer、Network Load Balancer 和 Classic Load Balancer。以下是这些负载均衡器的配置方式之间的主要区别。对于 Application Load Balancer 和 Network Load Balancer, 可以在目标组中注册目标, 并将流量路由到目标组。对于 Classic Load Balancer, 可以向负载均衡器注册实例。

## 可用区与负载均衡器节点

如果为负载均衡器启用可用区, Elastic Load Balancing 会在该可用区中创建一个负载均衡器节点。如果您在可用区中注册目标但不启用可用区, 这些已注册目标将无法接收流量。请注意, 如果您确保每个启用的可用区均具有至少一个已注册目标, 则负载均衡器将具有最高效率。

我们建议您启用多个可用区。(请注意, 使用应用程序负载均衡器时, 我们需要您启用多个可用区。) 借助此配置, 如果一个可用区变得不可用或没有正常运行的目标, 负载均衡器会继续将流量路由到其他可用区中正常运行的目标。

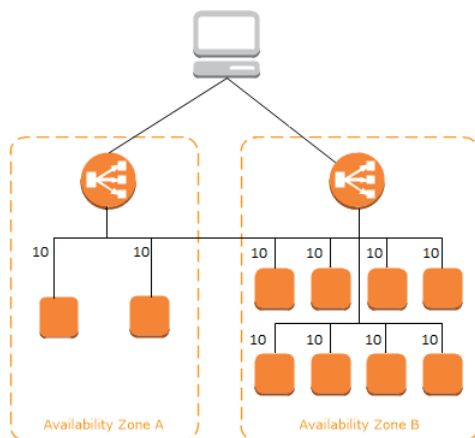
在禁用一个可用区后, 该可用区中的目标仍将注册负载均衡器, 但负载均衡器不会将流量路由到这些目标。

## 跨区域负载均衡

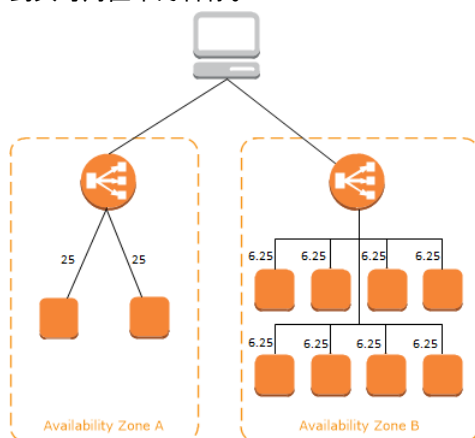
负载均衡器的节点将来自客户端的请求分配给已注册目标。启用了跨区域负载均衡后, 每个负载均衡器节点会在所有启用的可用区中的已注册目标之间分配流量。禁用了跨区域负载均衡后, 每个负载均衡器节点会仅在其可用区中的已注册目标之间分配流量。

下图演示了跨区域负载均衡的效果。有 2 个已启用的可用区, 其中可用区 A 中有 2 个目标, 可用区 B 中有 8 个目标。客户端发送请求, Amazon Route 53 使用负载均衡器节点之一的 IP 地址响应每个请求。这会分配流量, 以便每个负载均衡器节点接收来自客户端的 50% 的流量。每个负载均衡器节点会在其范围中的已注册目标之间分配其流量份额。

如果启用了跨区域负载均衡, 则 10 个目标中的每个目标接收 10% 的流量。这是因为每个负载均衡器节点可将其 50% 的客户端流量路由到所有 10 个目标。



如果禁用了跨区域负载均衡，则可用区 A 中的 2 个目标中的每个目标会接收 25% 的流量，可用区 B 中的 8 个目标中的每个目标会接收 6.25% 的流量。这是因为每个负载均衡器节点只能将其 50% 的客户端流量路由到其可用区中的目标。



对于 Application Load Balancer，始终启用跨区域负载均衡。

对于 Network Load Balancer，默认情况下禁用跨区域负载均衡。创建网络负载均衡器后，您随时可以启用或禁用跨区域负载均衡。有关更多信息，请参阅 Network Load Balancer 用户指南中的[跨区域负载均衡](#)。

在创建传统负载均衡器时，跨区域负载均衡的默认值取决于创建负载均衡器的方式。默认情况下，使用 API 或 CLI 时将禁用跨区域负载均衡。默认情况下，使用 AWS 管理控制台时启用跨区域负载均衡的选项处于选中状态。创建传统负载均衡器后，您随时可以启用或禁用跨区域负载均衡。有关更多信息，请参阅 Classic Load Balancer 用户指南中的[启用跨区域负载均衡](#)。

## 请求路由选择

在客户端将请求发送到负载均衡器之前，它会利用域名系统 (DNS) 服务器解析负载均衡器的域名。DNS 条目由 Amazon 控制，因为您的负载均衡器位于 `amazonaws.com` 域中。Amazon DNS 服务器会返回一个或多个 IP 地址到客户端，这些 IP 地址属于负载均衡器的负载均衡器节点。对于 Network Load Balancer，Elastic Load Balancing 将为启用的每个可用区创建一个网络接口。可用区内的每个负载均衡器节点使用该网络接口来获取一个静态 IP 地址。在您创建负载均衡器时，可以选择将一个弹性 IP 地址与每个网络接口关联。

当流向应用程序的流量随时间变化时，Elastic Load Balancing 会扩展负载均衡器并更新 DNS 条目。请注意，DNS 条目也将生存时间 (TTL) 指定为 60 秒，这可确保 IP 地址可以迅速重新映射以响应不断变化的流量。

客户端可以确定使用哪个 IP 地址将请求发送到负载均衡器。用于接收请求的负载均衡器节点会选择一个正常运行的已注册目标，并使用其私有 IP 地址将请求发送到该目标。

## 路由算法

对于 Application Load Balancer，用于接收请求的负载均衡器节点会按照优先级顺序评估侦听器规则以确定应用哪个规则，然后使用轮询路由算法从目标组中选择一个目标以实施规则操作。每个目标组的路由都是单独进行的，即使某个目标已在多个目标组中注册。

对于 Network Load Balancer，用于接收连接的负载均衡器节点会根据协议、源 IP 地址、源端口、目标 IP 地址、目标端口和 TCP 序列号，使用流式哈希算法从目标组中为默认规则选择一个目标。来自客户端的 TCP 连接具有不同的源端口和序列号，可以路由到不同的目标。每个单独的 TCP 连接在连接的有效期内路由到单个目标。

对于 Classic Load Balancer，接受请求的负载均衡器节点将利用不同算法来选择已注册的实例。它对 TCP 侦听器使用轮询路由算法，对 HTTP 和 HTTPS 侦听器使用最少未完成请求路由算法。

## HTTP 连接

Classic Load Balancer 使用提前打开的连接，但 Application Load Balancer 不使用。Classic Load Balancer 和 Application Load Balancer 都使用多路复用连接。也就是说，来自多个前端连接上的多个客户端的请求可通过单一的后端连接路由到指定目标。多路复用连接可缩短延迟并减少您的应用程序上的负载。要禁止多路复用连接，请在您的 HTTP 响应中设置 `Connection: close` 标头，禁用 HTTP 保持活动状态。

对于前端连接（客户端到负载均衡器），Classic Load Balancer 支持以下协议：HTTP/0.9、HTTP/1.0 和 HTTP/1.1。

对于前端连接，Application Load Balancer 支持以下协议：HTTP/0.9、HTTP/1.0、HTTP/1.1 和 HTTP/2。HTTP/2 仅适用于 HTTPS 侦听器，使用一个 HTTP/2 连接可并行发送多达 128 个请求。Application Load Balancer 还支持将连接从 HTTP 升级到 Websockets。

对于后端连接（负载均衡器到已注册目标），Application Load Balancer 和 Classic Load Balancer 均使用 HTTP/1.1。对于后端连接，默认支持保持连接。如果 HTTP/1.0 请求来自没有主机标头的客户端，负载均衡器会对后端连接发送的 HTTP/1.1 请求生成一个主机标头。对于应用程序负载均衡器，主机标头包含负载均衡器的 DNS 名称。对于传统负载均衡器，主机标头包含负载均衡器节点的 IP 地址。

对于 Application Load Balancer 和 Classic Load Balancer，均可设置空闲超时值。默认值为 60 秒。对于应用程序负载均衡器，空闲超时值只适用于前端连接。对于传统负载均衡器，如果前端连接或后端连接的空闲时间超过了空闲超时值，连接会断开，客户端会收到一条错误响应。已注册的目标可以使用保持连接超时保持后端连接处于开启状态，直到它准备就绪再断开。

对于前端连接，Application Load Balancer 和 Classic Load Balancer 均支持管道化 HTTP。对于后端连接它们均不支持管道化 HTTP。

## HTTP 标头

Application Load Balancer 和 Classic Load Balancer 支持 X-Forwarded-For、X-Forwarded-Proto 和 X-Forwarded-Port 标头。

对于使用 HTTP/2 的前端连接，标头名称是小写的。使用 HTTP/1.1 将请求发送到目标之前，以下标头名称将转换为混合大小写：X-Forwarded-For、X-Forwarded-Proto、X-Forwarded-Port、Host、X-Amzn-Trace-Id、Upgrade 和 Connection。所有其他标头名称是小写的。

Application Load Balancer 和 Classic Load Balancer 将响应代理返回客户端后，遵守来自传入客户端请求的连接标头。



## HTTP 标头限制

Application Load Balancer 的 HTTP/1.x 标头具有以下大小限制：

- 请求行：16K
- 单个标头：16K
- 完整标头：64K

Application Load Balancer 的 HTTP/2 标头具有以下大小限制：

- 请求行：8K
- 单个标头：8K
- 完整标头：64K

## 负载均衡器模式

在创建负载均衡器时，您必须选择使其成为内部负载均衡器还是面向 Internet 的负载均衡器。请注意，当您在 EC2-Classical 中创建传统负载均衡器时，它必须是面向 Internet 的负载均衡器。

面向 Internet 的负载均衡器的节点具有公共 IP 地址。面向 Internet 的负载均衡器的 DNS 名称可公开解析为节点的公共 IP 地址。因此，面向 Internet 的负载均衡器可以通过 Internet 从客户端路由请求。

内部负载均衡器的节点只有私有 IP 地址。内部负载均衡器的 DNS 名称可公开解析为节点的私有 IP 地址。因此，内部负载均衡器可路由的请求只能来自对负载均衡器的 VPC 具有访问权限的客户端。

请注意，面向 Internet 的负载均衡器和内部负载均衡器均使用私有 IP 地址将请求路由到您的目标。因此，您的目标无需使用公有 IP 地址从内部负载均衡器或面向 Internet 的负载均衡器接收请求。

如果您的应用程序具有多个层 (例如，必须连接到 Internet 的 Web 服务器和只连接到 Web 服务器的数据库服务器)，则您可以设计一个同时使用内部负载均衡器和面向 Internet 的负载均衡器的架构。创建一个面向 Internet 的负载均衡器并向其注册 Web 服务器。创建一个内部负载均衡器并向它注册数据库服务器。Web 服务器接收来自面向 Internet 的负载均衡器的请求，并将数据库服务器的请求发送到内部负载均衡器。数据库服务器接收来自内部负载均衡器的请求。

# Elastic Load Balancing 入门

有三种类型的负载均衡器：Application Load Balancer、Network Load Balancer 和 Classic Load Balancer。可以根据应用程序需求选择负载均衡器。有关更多信息，请参阅 [Elastic Load Balancing 产品比较](#)。

如果您现在有 传统负载均衡器，则可以迁移到 应用程序负载均衡器 或 网络负载均衡器。有关更多信息，请参阅 [迁移您的 Classic Load Balancer \(p. 18\)](#)。

## 创建 应用程序负载均衡器

要使用 AWS 管理控制台 创建 应用程序负载均衡器，请使用 Application Load Balancer 用户指南 中的 [Application Load Balancer 入门](#)。

要使用 AWS CLI 创建 应用程序负载均衡器，请参阅 Application Load Balancer 用户指南 中的 [使用 AWS CLI 创建 应用程序负载均衡器](#)

## 创建 网络负载均衡器

要使用 AWS 管理控制台 创建 网络负载均衡器，请参阅 Network Load Balancer 用户指南 中的 [Network Load Balancer 入门](#)。

要使用 AWS CLI 创建 网络负载均衡器，请参阅 Network Load Balancer 用户指南 中的 [使用 AWS CLI 创建 网络负载均衡器](#)

## 创建 传统负载均衡器

要使用 AWS 管理控制台 创建 传统负载均衡器，请参阅 Classic Load Balancer 用户指南 中的 [创建 传统负载均衡器](#)。

# 负载均衡器的身份验证和访问控制

AWS 使用安全凭证来识别您的身份并向您授予对 AWS 资源的访问权。利用 AWS Identity and Access Management (IAM) 的功能，可在不共享您的安全凭证的情况下允许其他用户、服务和应用程序完全使用或受限使用您的 AWS 资源。

默认情况下，IAM 用户没有创建、查看或修改 AWS 资源的权限。要允许某个 IAM 用户访问资源（如负载均衡器）和执行任务，您必须创建一个 IAM 策略（此策略向该 IAM 用户授予使用其所需的特定资源和 API 操作的权限），然后将此策略附加到该 IAM 用户或该 IAM 所属的组。在将策略附加到一个用户或一组用户时，它会授权或拒绝用户使用指定资源执行指定任务。

例如，您可以使用 IAM 在您的 AWS 账户下创建用户和组（IAM 用户可以是人员、系统或应用程序）。然后，使用 IAM 策略向用户和组授予对指定资源执行特定操作的权限。

## 使用 IAM 策略授予权限

在将策略附加到一个用户或一组用户时，它会授权或拒绝用户使用指定资源执行指定任务。

IAM 策略是包含一个或多个语句的 JSON 文档。每个语句的结构如下：

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "effect",
    "Action": "action",
    "Resource": "resource-arn",
    "Condition": {
      "condition": {
        "key": "value"
      }
    }
  }]
}
```

- Effect — effect 可以为 Allow 或 Deny。默认情况下 IAM 用户没有使用资源和 API 操作的权限，因此，所有请求均会被拒绝。显式允许将覆盖默认规则。显式拒绝将覆盖任何允许。
- Action — action 是对其授予或拒绝权限的特定 API 操作。有关指定 action 的更多信息，请参阅 [Elastic Load Balancing 的 API 操作 \(p. 8\)](#)。
- Resource — 受操作影响的资源。对于许多 Elastic Load Balancing API 操作，通过在此语句中指定 Amazon 资源名称 (ARN)，可以限制对特定负载均衡器授予或拒绝的权限。否则，您可以使用 \* 通配符指定所有负载均衡器。有关更多信息，请参阅 [Elastic Load Balancing 资源 \(p. 9\)](#)。
- Condition — 您可以选择性地使用条件来控制策略的生效时间。有关更多信息，请参阅 [Elastic Load Balancing 的条件密钥 \(p. 12\)](#)。

有关更多信息，请参见 [IAM 用户指南](#)。

## Elastic Load Balancing 的 API 操作

在 IAM 策略语句的 Action 元素中，您可以指定 Elastic Load Balancing 所提供的任意 API 操作。如以下示例所示，您必须使用小写形式的字符串 elasticloadbalancing: 作为操作名称的前缀：

```
"Action": "elasticloadbalancing:DescribeLoadBalancers"
```

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": [  
  "elasticloadbalancing:DescribeLoadBalancers",  
  "elasticloadbalancing>DeleteLoadBalancer"  
]
```

您也可以使用 \* 通配符指定多项操作。以下示例指定以 Describe 开头的 **所有** Elastic Load Balancing API 操作名称：

```
"Action": "elasticloadbalancing:Describe*"
```

要指定 Elastic Load Balancing 的所有 API 操作，请按下列所示使用通配符 \*：

```
"Action": "elasticloadbalancing:*"
```

有关 Elastic Load Balancing API 操作的完整列表，请参阅以下文档：

- Application Load Balancer 和 Network Load Balancer — [API 参考版本 2015-12-01](#)
- Classic Load Balancer — [API 参考版本 2012-06-01](#)

## Elastic Load Balancing 资源

资源级权限是指指定允许用户对哪些资源执行操作的能力。Elastic Load Balancing 对资源级权限提供部分支持。对于支持资源级权限的 API 操作，您可以控制用户可与操作结合使用的资源。要在策略中指定资源，您必须使用其 Amazon 资源名称 (ARN)。指定 ARN 时，您可以在路径中使用 \* 通配符；例如，在不希望指定确切负载均衡器名称时可以这样做。

应用程序负载均衡器的 ARN 采用以下格式：

```
arn:aws-cn:elasticloadbalancing:region-code:account-id:loadbalancer/app/load-balancer-name/load-balancer-id
```

网络负载均衡器的 ARN 采用以下格式：

```
arn:aws-cn:elasticloadbalancing:region-code:account-id:loadbalancer/net/load-balancer-name/load-balancer-id
```

传统负载均衡器的 ARN 采用以下格式：

```
arn:aws-cn:elasticloadbalancing:region-code:account-id:loadbalancer/load-balancer-name
```

侦听器的 ARN 和应用程序负载均衡器的侦听器规则采用以下格式：

```
arn:aws-cn:elasticloadbalancing:region-code:account-id:listener/app/load-balancer-name/load-balancer-id/listener-id  
arn:aws-cn:elasticloadbalancing:region-code:account-id:listener-rule/app/load-balancer-name/load-balancer-id/listener-id/rule-id
```

网络负载均衡器的侦听器的 ARN 采用以下格式：

```
arn:aws-cn:elasticloadbalancing:region-code:account-id:listener/net/load-balancer-name/load-balancer-id/listener-id
```

目标组的 ARN 采用以下格式：

```
arn:aws-cn:elasticloadbalancing:region-code:account-id:targetgroup/target-group-name/target-group-id
```

不支持资源级权限的 API 操作

以下 Elastic Load Balancing 操作不支持资源级权限：

- API 版本 2015-12-01：
  - DescribeAccountLimits
  - DescribeListenerCertificates
  - DescribeListeners
  - DescribeLoadBalancerAttributes
  - DescribeLoadBalancers
  - DescribeRules
  - DescribeSSLPolicies
  - DescribeTags
  - DescribeTargetGroupAttributes
  - DescribeTargetGroups
  - DescribeTargetHealth
- API 版本 2012-06-01：
  - DescribeInstanceHealth
  - DescribeLoadBalancerAttributes
  - DescribeLoadBalancerPolicyTypes
  - DescribeLoadBalancers
  - DescribeLoadBalancerPolicies
  - DescribeTags

对于不支持资源级权限的 API 操作，必须指定以下资源语句：

```
"Resource": "*"
```

## Elastic Load Balancing 的资源级权限

下面的各表介绍支持资源级权限的 Elastic Load Balancing 操作，以及每个操作支持的资源。

API 版本 2015-12-01

API 操作	资源 ARN
AddListenerCertificates	侦听器
AddTags	负载均衡器，目标组
CreateListener	负载均衡器

API 操作	资源 ARN
CreateLoadBalancer	负载均衡器
CreateRule	侦听器
CreateTargetGroup	目标组
DeleteListener	侦听器
DeleteLoadBalancer	负载均衡器
DeleteRule	侦听器规则
DeleteTargetGroup	目标组
DeregisterTargets	目标组
ModifyListener	侦听器
ModifyLoadBalancerAttributes	负载均衡器
ModifyRule	侦听器规则
ModifyTargetGroup	目标组
ModifyTargetGroupAttributes	目标组
RegisterTargets	目标组
RemoveListenerCertificates	侦听器
RemoveTags	负载均衡器, 目标组
SetIpAddressType	负载均衡器
SetRulePriorities	侦听器规则
SetSecurityGroups	负载均衡器
SetSubnets	负载均衡器

API 版本 2012-06-01

API 操作	资源 ARN
AddTags	负载均衡器
ApplySecurityGroupsToLoadBalancer	负载均衡器
AttachLoadBalancerToSubnets	负载均衡器
ConfigureHealthCheck	负载均衡器
CreateAppCookieStickinessPolicy	负载均衡器
CreateLBCookieStickinessPolicy	负载均衡器
CreateLoadBalancer	负载均衡器
CreateLoadBalancerListeners	负载均衡器

API 操作	资源 ARN
CreateLoadBalancerPolicy	负载均衡器
DeleteLoadBalancer	负载均衡器
DeleteLoadBalancerListeners	负载均衡器
DeleteLoadBalancerPolicy	负载均衡器
DeregisterInstancesFromLoadBalancer	负载均衡器
DetachLoadBalancerFromSubnets	负载均衡器
DisableAvailabilityZonesForLoadBalancer	负载均衡器
EnableAvailabilityZonesForLoadBalancer	负载均衡器
ModifyLoadBalancerAttributes	负载均衡器
RegisterInstancesWithLoadBalancer	负载均衡器
RemoveTags	负载均衡器
SetLoadBalancerListenerSSLCertificate	负载均衡器
SetLoadBalancerPoliciesForBackendServer	负载均衡器
SetLoadBalancerPoliciesOfListener	负载均衡器

## Elastic Load Balancing 的条件密钥

在创建策略时，您可指定控制策略生效时间的条件。每个条件都包含一个或多个键值对。有全局条件键和特定于服务的条件键。

`elasticloadbalancing:ResourceTag/key` 条件键特定于 Elastic Load Balancing。以下操作支持此条件键。

API 版本 2015-12-01

- AddTags
- CreateListener
- CreateLoadBalancer
- DeleteLoadBalancer
- DeleteTargetGroup
- DeregisterTargets
- ModifyLoadBalancerAttributes
- ModifyTargetGroup
- ModifyTargetGroupAttributes
- RegisterTargets
- RemoveTags
- SetIpAddressType
- SetSecurityGroups

- SetSubnets

API 版本 2012-06-01

- AddTags
- ApplySecurityGroupsToLoadBalancer
- AttachLoadBalancersToSubnets
- ConfigureHealthCheck
- CreateAppCookieStickinessPolicy
- CreateLBCookieStickinessPolicy
- CreateLoadBalancer
- CreateLoadBalancerListeners
- CreateLoadBalancerPolicy
- DeleteLoadBalancer
- DeleteLoadBalancerListeners
- DeleteLoadBalancerPolicy
- DeregisterInstancesFromLoadBalancer
- DetachLoadBalancersFromSubnets
- DisableAvailabilityZonesForLoadBalancer
- EnableAvailabilityZonesForLoadBalancer
- ModifyLoadBalancerAttributes
- RegisterInstancesWithLoadBalancer
- RemoveTags
- SetLoadBalancerListenerSSLCertificate
- SetLoadBalancerPoliciesForBackendServer
- SetLoadBalancerPoliciesOfListener

有关全局条件键的更多信息，请参阅 IAM 用户指南 中的 [AWS 全局条件上下文键](#)。

以下操作支持 `aws:RequestTag/key` 和 `aws:TagKeys` 条件键：

- AddTags
- CreateLoadBalancer
- RemoveTags

## 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。您可以根据您的 IAM 用户对 Elastic Load Balancing 所需的访问权限将这些策略附加到这些用户：

- ElasticLoadBalancingFullAccess — 授予使用 Elastic Load Balancing 功能所需的完整访问权限。
- ElasticLoadBalancingReadOnly — 授予对 Elastic Load Balancing 功能的只读访问权限。

有关每个 Elastic Load Balancing 操作所需的权限的更多信息，请参阅 [Elastic Load Balancing API 权限 \(p. 14\)](#)。



## Elastic Load Balancing API 权限

您必须为 IAM 用户授予调用所需 Elastic Load Balancing API 操作的权限，如[Elastic Load Balancing 的 API 操作 \(p. 8\)](#)中所述。此外，对于某些 Elastic Load Balancing 操作，您必须授予 IAM 用户从 Amazon EC2 API 调用特定操作的权限。

### 2015-12-01 API 所需的权限

从 2015-12-01 API 调用以下操作时，您必须授予 IAM 用户调用指定操作的权限。

#### CreateLoadBalancer

- `elasticloadbalancing:CreateLoadBalancer`
- `ec2:DescribeAccountAttributes`
- `ec2:DescribeAddresses`
- `ec2:DescribeInternetGateways`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeSubnets`
- `ec2:DescribeVpcs`
- `iam:CreateServiceLinkedRole`

#### CreateTargetGroup

- `elasticloadbalancing:CreateTargetGroup`
- `ec2:DescribeInternetGateways`
- `ec2:DescribeVpcs`

#### RegisterTargets

- `elasticloadbalancing:RegisterTargets`
- `ec2:DescribeInstances`
- `ec2:DescribeInternetGateways`
- `ec2:DescribeSubnets`
- `ec2:DescribeVpcs`

#### SetIpAddressType

- `elasticloadbalancing:SetIpAddressType`
- `ec2:DescribeSubnets`

#### SetSubnets

- `elasticloadbalancing:SetSubnets`
- `ec2:DescribeSubnets`

### 2012-06-01 API 所需的权限

从 2012-06-01 API 调用以下操作时，您必须授予 IAM 用户调用指定操作的权限。

#### ApplySecurityGroupsToLoadBalancer

- `elasticloadbalancing:ApplySecurityGroupsToLoadBalancer`
- `ec2:DescribeAccountAttributes`
- `ec2:DescribeSecurityGroups`

#### AttachLoadBalancerToSubnets

- `elasticloadbalancing:AttachLoadBalancerToSubnets`

- ec2:DescribeSubnets

CreateLoadBalancer

- elasticloadbalancing:CreateLoadBalancer
- ec2:CreateSecurityGroup
- ec2:DescribeAccountAttributes
- ec2:DescribeInternetGateways
- ec2:DescribeSecurityGroups
- ec2:DescribeSubnets
- ec2:DescribeVpcs
- iam:CreateServiceLinkedRole

DeregisterInstancesFromLoadBalancer

- elasticloadbalancing:DeregisterInstancesFromLoadBalancer
- ec2:DescribeClassicLinkInstances
- ec2:DescribeInstances

DescribeInstanceHealth

- elasticloadbalancing:DescribeInstanceHealth
- ec2:DescribeClassicLinkInstances
- ec2:DescribeInstances

DescribeLoadBalancers

- elasticloadbalancing:DescribeLoadBalancers
- ec2:DescribeSecurityGroups

DisableAvailabilityZonesForLoadBalancer

- elasticloadbalancing:DisableAvailabilityZonesForLoadBalancer
- ec2:DescribeAccountAttributes
- ec2:DescribeInternetGateways
- ec2:DescribeVpcs

EnableAvailabilityZonesForLoadBalancer

- elasticloadbalancing:EnableAvailabilityZonesForLoadBalancer
- ec2:DescribeAccountAttributes
- ec2:DescribeInternetGateways
- ec2:DescribeSubnets
- ec2:DescribeVpcs

RegisterInstancesWithLoadBalancer

- elasticloadbalancing:RegisterInstancesWithLoadBalancer
- ec2:DescribeAccountAttributes
- ec2:DescribeClassicLinkInstances
- ec2:DescribeInstances
- ec2:DescribeVpcClassicLink

## Elastic Load Balancing 服务相关角色

Elastic Load Balancing 使用服务相关角色来获取它代表您调用其他 AWS 服务所需的权限。有关更多信息，请参阅 IAM 用户指南 中的 [使用服务相关角色](#)。

## 服务相关角色授予的权限

Elastic Load Balancing 使用名为 `AWSServiceRoleForElasticLoadBalancing` 的服务相关角色代表您调用以下操作：

- `ec2:DescribeAddresses`
- `ec2:DescribeInstances`
- `ec2:DescribeSubnets`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeVpcs`
- `ec2:DescribeInternetGateways`
- `ec2:DescribeAccountAttributes`
- `ec2:DescribeClassicLinkInstances`
- `ec2:DescribeVpcClassicLink`
- `ec2:CreateSecurityGroup`
- `ec2:CreateNetworkInterface`
- `ec2>DeleteNetworkInterface`
- `ec2:ModifyNetworkInterface`
- `ec2:ModifyNetworkInterfaceAttribute`
- `ec2:AuthorizeSecurityGroupIngress`
- `ec2:AssociateAddress`
- `ec2:DisassociateAddress`
- `ec2:AttachNetworkInterface`
- `ec2:DetachNetworkInterface`
- `ec2:AssignPrivateIpAddresses`
- `ec2:AssignIpv6Addresses`
- `ec2:UnassignIpv6Addresses`
- `logs:CreateLogDelivery`
- `logs:GetLogDelivery`
- `logs:UpdateLogDelivery`
- `logs>DeleteLogDelivery`
- `logs:ListLogDeliveries`

`AWSServiceRoleForElasticLoadBalancing` 信任 `elasticloadbalancing.amazonaws.com` 服务来代入该角色。

## 创建服务相关角色

您无需手动创建 `AWSServiceRoleForElasticLoadBalancing` 角色。Elastic Load Balancing 将在您创建负载均衡器时为您创建此角色。

要让 Elastic Load Balancing 代表您创建服务相关角色，您必须具有所需权限。有关更多信息，请参阅 IAM 用户指南 中的 [服务相关角色权限](#)。

如果您在 2018 年 1 月 11 日之前创建了负载均衡器，则 Elastic Load Balancing 已在您的 AWS 账户中创建了 `AWSServiceRoleForElasticLoadBalancing`。有关更多信息，请参阅 IAM 用户指南 中的 [我的 AWS 账户中出现新角色](#)。

## 编辑服务相关角色

您可以使用 IAM 编辑 `AWSServiceRoleForElasticLoadBalancing` 的描述。有关更多信息，请参阅 IAM 用户指南 中的 [编辑服务相关角色](#)。

## 删除服务相关角色

如果您不再需要使用 Elastic Load Balancing，我们建议您删除 `AWSServiceRoleForElasticLoadBalancing`。

只有在删除 AWS 账户中的所有负载均衡器后，才能删除此服务相关角色。这可确保您不会无意中删除访问您的负载均衡器的权限。有关更多信息，请参阅 [删除 应用程序负载均衡器](#)、[删除网络负载均衡器](#) 和 [删除 传统负载均衡器](#)。

您可以使用 IAM 控制台、IAM CLI 或 IAM API 删除服务相关角色。有关更多信息，请参阅 IAM 用户指南 中的 [删除服务相关角色](#)。

在您删除 `AWSServiceRoleForElasticLoadBalancing` 之后，Elastic Load Balancing 将在您创建负载均衡器时再次为您创建该角色。

# 迁移您的 Classic Load Balancer

如果您在 VPC 中已经有传统负载均衡器，并且确信应用程序负载均衡器或网络负载均衡器满足您的需求，那么您就可以迁移传统负载均衡器。在迁移过程完成后，您就可以利用新负载均衡器的功能了。有关更多信息，请参阅 [Elastic Load Balancing 产品比较](#)。

## 迁移过程

- [步骤 1：创建新负载均衡器 \(p. 18\)](#)
- [步骤 2：逐步将流量重定向到您的新负载均衡器 \(p. 19\)](#)
- [步骤 3：更新对您的传统负载均衡器的引用 \(p. 20\)](#)
- [步骤 4：删除传统负载均衡器 \(p. 20\)](#)

## 步骤 1：创建新负载均衡器

创建配置等同于您的传统负载均衡器的应用程序负载均衡器或网络负载均衡器。

您可以使用以下方法之一来创建负载均衡器和目标组：

- [控制台中的迁移向导 \(p. 18\)](#)
- [负载均衡器复制实用工具 \(p. 19\)](#)
- [手动方式 \(p. 19\)](#)

## 选项 1：使用迁移向导进行迁移

迁移向导根据传统负载均衡器的配置创建应用程序负载均衡器或网络负载均衡器。所创建负载均衡器的类型取决于传统负载均衡器的配置。

### 迁移向导发行说明

- 传统负载均衡器必须位于 VPC 中。
- 如果传统负载均衡器具有 HTTP 或 HTTPS 侦听器，则该向导可以创建应用程序负载均衡器。如果传统负载均衡器具有 TCP 侦听器，则该向导可以创建网络负载均衡器。
- 如果传统负载均衡器的名称与现有应用程序负载均衡器或网络负载均衡器的名称匹配，则该向导将要求您为新负载均衡器指定不同的名称。
- 如果传统负载均衡器具有一个子网，则该向导将要求您在创建应用程序负载均衡器时指定另一个子网。
- 如果传统负载均衡器已在 EC2-Classic 中注册实例，这些实例不会注册到新负载均衡器的目标组。
- 如果传统负载均衡器具有以下类型的已注册实例，不会向网络负载均衡器的目标组注册它们：C1、CC1、CC2、CG1、CG2、CR1、CS1、G1、G2、HI1、HS1、M1、M2、M3 和 T1。
- 如果传统负载均衡器具有 HTTP/HTTPS 侦听器，但使用 TCP 运行状况检查，则在创建应用程序负载均衡器时，默认情况下，向导将更改为 HTTP 运行状况检查并将路径设置为“/”。
- 如果将传统负载均衡器迁移到网络负载均衡器，则将更改运行状况检查设置以满足 Network Load Balancer 的要求。
- 如果传统负载均衡器有多个 HTTPS 侦听器，则向导将选择一个侦听器并使用其证书和策略。如果端口 443 上有一个 HTTPS 侦听器，向导将选择此侦听器。如果所选侦听器使用自定义策略或 Application Load Balancer 不支持的策略，则向导将更改为默认安全策略。
- 如果传统负载均衡器具有安全 TCP 侦听器，则网络负载均衡器将使用 TCP 侦听器，但不使用证书或安全策略。

- 如果传统负载均衡器有多个侦听器，该向导将使用端口值最低的侦听器端口作为目标组端口。注册到这些侦听器的每个实例都会在所有侦听器的侦听器端口上注册到目标组。
- 如果传统负载均衡器的一些标签在标签名称中具有 aws 前缀，则这些标签不会添加到新的负载均衡器。

#### 使用迁移向导迁移传统负载均衡器

1. 打开 Amazon EC2 控制台 <https://console.amazonaws.cn/ec2/>。
2. 在导航窗格上的 LOAD BALANCING 下，选择 Load Balancers。
3. 选择您的传统负载均衡器。
4. 在 Migration 选项卡上，选择 Launch ALB Migration Wizard 或 Launch NLB Migration Wizard。显示的按钮取决于检查传统负载均衡器后由向导选择的负载均衡器类型。
5. 在 Review 页面上，验证向导选择的配置选项。要更改某个选项，请选择 Edit。
6. 当您完成配置新的负载均衡器时，选择 Create。

## 选项 2：使用负载均衡器复制实用程序进行迁移

此实用程序在 GitHub 上提供。有关更多信息，请参阅[负载均衡器复制实用程序](#)。

## 选项 3：手动迁移

下面是手动创建基于传统负载均衡器的新负载均衡器的常规过程。您可以使用 AWS 管理控制台、AWS CLI 或 AWS 软件开发工具包进行迁移。有关更多信息，请参阅[Elastic Load Balancing 入门 \(p. 7\)](#)。

- 创建具有与传统负载均衡器相同的模式 (面向 Internet 或内部)、子网和安全组的新负载均衡器。
- 使用传统负载均衡器的运行状况检查设置为负载均衡器创建一个目标组。
- 执行以下任一操作：
  - 如果您的传统负载均衡器附加到 Auto Scaling 组，请将您的目标组附加到 Auto Scaling 组。这样还可以向目标组注册 Auto Scaling 实例。
  - 向目标组注册您的 EC2 实例。
- 创建一个或多个侦听器，每个都具有将请求转发到目标组的默认规则。如果创建 HTTPS 侦听器，则可指定您为传统负载均衡器所指定的同一证书。建议您使用默认安全策略。
- 如果您的传统负载均衡器有标签，请进行检查并将相关标签添加到新负载均衡器。

## 步骤 2：逐步将流量重定向到您的新负载均衡器

向新负载均衡器注册您的实例后，就可以开始测试新负载均衡器，同时逐步重定向流量。

#### 逐步将流量重定向到您的新负载均衡器

1. 将新负载均衡器的 DNS 名称粘贴到已连接 Internet 的 Web 浏览器的地址栏中。如果一切正常，浏览器会显示您服务器的默认页面。
2. 创建一个用于将域名与您的新负载均衡器关联的新 DNS 记录。如果您的 DNS 服务支持权重，则在新 DNS 记录中指定权重为 1；对于您传统负载均衡器的现有 DNS 记录，指定权重为 9。这样可以使 10% 的流量定向到新负载均衡器，而将 90% 的流量定向到传统负载均衡器。
3. 监控您的新负载均衡器，验证它能否接收流量并将请求路由到您的实例。

#### Important

DNS 记录中的生存时间 (TTL) 为 60 秒，这意味着，解析域名的任何 DNS 服务器在缓存中保留记录信息的时间为 60 秒。因此，当您完成上一步骤并且开始向全球 DNS 服务器传播更改之

后，上面这些 DNS 服务器仍然可以在 60 秒内将流量路由到您的传统负载均衡器。在传输过程中，流量可以定向到任一负载均衡器。

4. 继续更新您的 DNS 记录的权重，直到所有流量都定向到您的新负载均衡器。完成后，您可以删除传统负载均衡器的 DNS 记录。

## 步骤 3：更新对您的传统负载均衡器的引用

现在您已迁移到传统负载均衡器，请务必更新对它的任何引用，如下所示：

- 使用 AWS CLI `aws elb` 命令（而不是 `aws elbv2` 命令）的脚本
- 使用 Elastic Load Balancing 版本 2012-06-01（而不是 API 版本 2015-12-01）的代码
- 使用 API 版本 2012-06-01（而不是 2015-12-01 版本）的 IAM 策略
- 使用 CloudWatch 指标的过程
- AWS CloudFormation 模板

### 资源

- AWS CLI Command Reference 中的 [elbv2](#)
- [Elastic Load Balancing API 参考第 2015-12-01 版](#)
- [负载均衡器的身份验证和访问控制 \(p. 8\)](#)
- Application Load Balancer 用户指南 中的 [应用程序负载均衡器 指标](#)
- Network Load Balancer 用户指南 中的 [网络负载均衡器 指标](#)
- AWS CloudFormation 用户指南 中的 [AWS::ElasticLoadBalancingV2::LoadBalancer](#)

## 步骤 4：删除传统负载均衡器

将所有流量重新定向到新负载均衡器且路由到传统负载均衡器的所有现有请求都完成之后，可以删除传统负载均衡器。