aws

开发人员指南

Amazon 应用程序恢复控制器 (ARC)

Amazon 应用程序恢复控制器 (ARC): 开发人员指南

Table of Contents

什么是 ARC？

Amazon 应用程序恢复控制器 (ARC) 可帮助您准备并更快地完成 AWS 在全球云基础设施上运行的应用程序的恢复。

ARC 提供以下功能：

- 多可用区 (AZ) 恢复，包括区域转移和区域自动切换，这使您可以通过将流量从受损的可用区暂时转移到健康的可用区，从单个可用区受损中恢复过来。
- 多区域恢复，包括用于故障转移的路由控制和用于应用程序监控的就绪性检查。

多可用区恢复

可用区转移

您可以使用 ARC 区域切换来快速隔离单个可用区 (AZ) 缺陷并从中恢复。区域转移会暂时将受支持资源的流量从受损的可用区转移到同一 AWS 区域 AZs 的健康可用区。启动区域转移可以帮助您的应用程序快速恢复，例如，从开发人员的错误代码部署或单个可用区的 AWS 损坏中恢复。将流量从受损的可用区转移出去可以减少对在受损可用区中使用您的应用程序的客户端的影响。

您可以为某个区域中账户中任何受支持的资源开始 AWS 区域切换。区域偏移是手动的，也是临时的。开始区域转移时，必须指定最长为三天的（可延长）到期时间。要为支持的资源启用区域偏移，请参阅。支持的资源

区域自动换档

ARC zonal autoshift 授权 AWS 代表您将受支持资源的受损可用区 AZs 中的流量转移到同一区域的健康可用区。 AWS AWS 当内部遥测显示某个区域中的一个可用区存在可能影响客户的损伤时，将启动 AWS 区域自动切换。内部遥测包含来自多个来源的指标，包括 AWS 网络、Amazon EC2 和 Elastic Load Balancing 服务。

区域自动换档是暂时的。 AWS 当内部遥测指示器显示不再存在问题或潜在问题时，结束区域自动移位。

要了解有关这些功能的更多信息，请参阅以下章节：

- ARC 中的区域偏移
- ARC 中的区域自动换档

多区域恢复

路由控制

ARC 极其可靠的路由控制支持多区域恢复，因此您的应用程序可以跨 AWS 区域故障转移域名系统 DNS 流量。

如果您的应用程序设计为在多个 AWS 区域外运行，则可以使用 ARC 路由控制在区域之间进行故障转移。路由控制使您可以将流量从受损 AWS 区域故障转移到健康 AWS 区域，从而确保应用程序保持可用性。路线控制包括安全规则，这些规则通过强加您定义的护栏来帮助保护您免受意外结果的影响。例如，您可以强加一条安全规则，规定只有一个应用程序副本（活动副本或备用副本）处于启用和使用状态。

准备情况检查

ARC 就绪检查持续监控 AWS 资源配额、容量和网络路由策略，并可以通知您有关可能影响您故障转移到副本应用程序和从区域受损中恢复的能力的更改。持续的就绪性检查可确保您可以将多区域应用程序保持在经过扩展和配置以处理故障转移流量的状态。首次配置 ARC 时和应用程序正常运行期间，就绪检查非常有用。准备情况检查不打算用于事件期间故障转移的关键路径。

要了解有关这些功能的更多信息，请参阅以下章节：

- ARC 中的路由控制
- ARC 中的准备情况检查

比较 ARC 中的多可用区和多区域恢复功能

Amazon 应用程序恢复控制器 (ARC) 中的区域切换、区域自动切换和路由控制都可以实现快速恢复，并帮助您确保应用程序的弹性。 AWS 这些功能具有很高的可用性，有助于在应用程序延迟增加或可用性降低的情况下支持恢复。这些功能还可以通过将流量从孤立的损伤中转移出来，从而帮助快速恢复应用程序，从而限制损伤造成的影响和时间损失。

路由控制主要集中在多个 AWS 区域（多区域）中的 AWS 应用程序上，而区域转移和区域自动切换仅支持使用多可用区应用程序为支持的资源转移流量。

下表中的信息包括区域偏移、分区自动移位和路径控制的一些关键功能。这些描述可以帮助您更好地了解特定选项如何成为满足应用程序需求的最佳选择。

路由控制	可用区转移	可用区自动转移
区域性 将流量从一个 AWS 区域重新路由到另一个区域（主要）	可用区 将流量从可用区转移出去 流量流向该区域的其他可用区，而非特定目标	可用区 将流量从可用区转移出去 流量流向该区域的其他可用区，而非特定目标
需要设置 需要配置和设置	可能需要设置 某些支持的资源需要选择加入 有关更多信息，请参阅 支持的资源。	需要设置 必须为支持的资源启用 有关更多信息，请参阅 支持的资源。
客户发起 客户决定何时重新路由流量	客户发起 客户决定何时启动可用区转移	AWS发起 AWS 代表你将应用程序流量从可用区转移出去
收费 需要单独收取路由控制费用	包含在服务中（不收取额外费用） 对于支持的资源，包括创建区域班次以将流量从中 AZs 移开	包含在服务中（不收取额外费用） 支持的资源包括启动自动换档以 AZs 代表您转移流量

路由控制	可用区转移	可用区自动转移
不会过期	暂时	暂时
流量可以无限期地重新路由到副本	所有可用区转移都必须设置为到期	AWS 开始和结束自动换档

要了解上述每个功能的更多信息，请参阅以下章节：

- ARC 中的区域偏移
- ARC 中的区域自动换档
- ARC 中的路由控制

使用分区移位和分区自动移位来恢复 ARC 中的应用程序

本节介绍如何使用 Amazon 应用程序恢复控制器 (ARC) 中的功能可靠地从受损的可用区 (AZ) 中出现的问题中恢复 AWS 资源。区域转移和区域自动切换会暂时将受支持资源的流量从受损的可用区转移出去，从而缩短应用程序的恢复时间。

区域移位和区域自动换档之间的主要区别在于，一种是您可以控制的手动交通换档，另一种是代表您自动将交通从障碍中移开。

- 通过区域切换，您可以手动将受支持资源的流量转移到 AWS 区域 远离可用区域的地方。
- 使用区域自动切换，受支持资源的流量将自动从受损的可用区转移出去，并在同一区域重新路由到运行 AZs 状况良好。 AWS

以下主题描述了分区移位和分区自动移位功能以及如何使用它们。

主题

ARC 中的区域偏移

Amazon App lication Recovery Controller (ARC) 区域转移允许您将受支持资源的流量从同一区域的受损可用区 (AZ) 转移 AWS 区域 到运行状况良好 AZs 。将资源流量从受损的可用区转移出去，可以缩短由停电或可用区中的硬件或软件问题造成的影响持续时间和严重性，并有助于缓解问题并快速恢复应用程序。由于部署不当导致延迟问题或者由于可用区损坏等原因，您可能会选择转移流量。

您必须选择使用资源才能使用区域移动。有关更多信息，请参阅支持的资源。

在开始区域转移之前，必须预先扩展应用程序，并确保有足够的容量将流量从可用区域转移出去。预缩放后，您可以选择要转移的可用区和要转移流量的资源，然后开始区域切换。您可以随时取消班次，让流量开始返回原始可用区。有关更多信息，请参阅 ARC 中的可用区转移最佳实践。

所有区域变化都是暂时的缓解措施。在开始分区班次时，您可以设置初始到期时间，从一分钟到三天（72 小时），如果您需要继续进行流量切换，则可以延长。

在特定情况下，区域转移不会将流量从可用区转移出去。有关更多信息，请参阅 支持的资源。

可用区转移的工作原理

当您开始对支持的资源进行区域转移时，该资源的流量将从您指定的可用区 (AZ) 移开。ARC 支持的资源提供了将指定可用区标记为不健康的集成，这会导致流量从受损的可用区转移出去。

流量开始转移-当您在 ARC 中开始区域转移时，您可能不会看到流量立即从可用区移出。可用区中现有的、正在进行的连接可能需要很短的时间才能完成，具体取决于客户端行为和连接重复使用情况。DNS 设置和其他因素（包括现有连接）可以在短短几分钟内完成，但可能需要更长的时间。有关更多信息，请参阅确保交通转移快速完成。

交通转移结束 ——当区域班次到期或您取消时，ARC会采取措施停止交通转移并撤消开始交通转移的流程。现在，恢复的可用区被识别为可用于该资源，流量将恢复流向该可用区。

您必须将所有区域班次设置为在开始轮班时到期。最初，可用区转移最多可设置为三天（72 小时）后到期。但是您可以随时更新可用区转移，以设置新的到期时间。如果您已准备好将流量恢复到可用区，也可以在可用区转移到期之前取消它。

当流量没有转移时-在特定情况下，区域转移不会将流量从可用区域转移出去。例如，假设当负载均衡器中的目标组没有任何实例，或者所有实例都运行状况 AZs 不佳时，您就开始对负载均衡器进行区域切换。在这种情况下，负载均衡器处于失效打开状态，启动区域转移不会转移流量。

在开始对资源进行区域转移之前，请确保满足成功进行区域偏移的所有条件。 AWS 资源对区域变化的处理方式有所不同。有关可用区转移支持的更多信息，请参阅支持的资源。

AWS 区域 区域转移的可用性

有关亚马逊应用程序恢复控制器 (ARC) 的区域支持和服务终端节点的详细信息，请参阅《亚马逊网络服务通用参考》中的亚马逊应用程序恢复控制器 (ARC) 终端节点和配额。

此处列出的当前提供区域移位和区域自动换档。 AWS 区域 中国区域，即中国（北京）区域和中国（宁夏）区域，也提供区域移位和区域自动切换。使用 Amazon 应用程序恢复控制器 (ARC) 的资源可能还有其他注意事项。有关更多信息，请参阅支持的资源。

区域名称	区域	端点	协议	
美国东部（俄亥俄州）	us-east-2	arc-zonal-shift.us-east-2.amazonaws.com	HTTPS	
		arc-zonal-shift-fips.us-east-2.api.aws	HTTPS	
		arc-zonal-shift.us-east-2.api.aws	HTTPS	

区域名称	区域	端点	协议
美国东部（弗吉尼亚州北部）	us-east-1	arc-zonal-shift.us-east-1.amazonaws.com	HTTPS
		arc-zonal-shift-fips.us-east-1.api.aws	HTTPS
		arc-zonal-shift.us-east-1.api.aws	HTTPS
美国西部（加利福尼亚北部）	us-west-1	arc-zonal-shift.us-west-1.amazonaws.com	HTTPS
		arc-zonal-shift-fips.us-west-1.api.aws	HTTPS
		arc-zonal-shift.us-west-1.api.aws	HTTPS
美国西部（俄勒冈州）	us-west-2	arc-zonal-shift.us-west-2.amazonaws.com	HTTPS
		arc-zonal-shift-fips.us-west-2.api.aws	HTTPS
		arc-zonal-shift.us-west-2.api.aws	HTTPS
非洲（开普敦）	af-south-1	arc-zonal-shift.af-south-1.amazonaws.com	HTTPS
		arc-zonal-shift.af-south-1.api.aws	HTTPS
亚太地区（香港）	ap-east-1	arc-zonal-shift.ap-east-1.amazonaws.com	HTTPS
		arc-zonal-shift.ap-east-1.api.aws	HTTPS
亚太地区（海得拉巴）	ap-south-2	arc-zonal-shift.ap-south-2.amazonaws.com	HTTPS
		arc-zonal-shift.ap-south-2.api.aws	HTTPS
亚太地区（雅加达）	ap-southeast-3	arc-zonal-shift.ap-southeast-3.amazonaws.com	HTTPS
		arc-zonal-shift.ap-southeast-3.api.aws	HTTPS
亚太地区（马来西亚）	ap-southeast-5	arc-zonal-shift.ap-southeast-5.amazonaws.com	HTTPS
		arc-zonal-shift.ap-southeast-5.api.aws	HTTPS

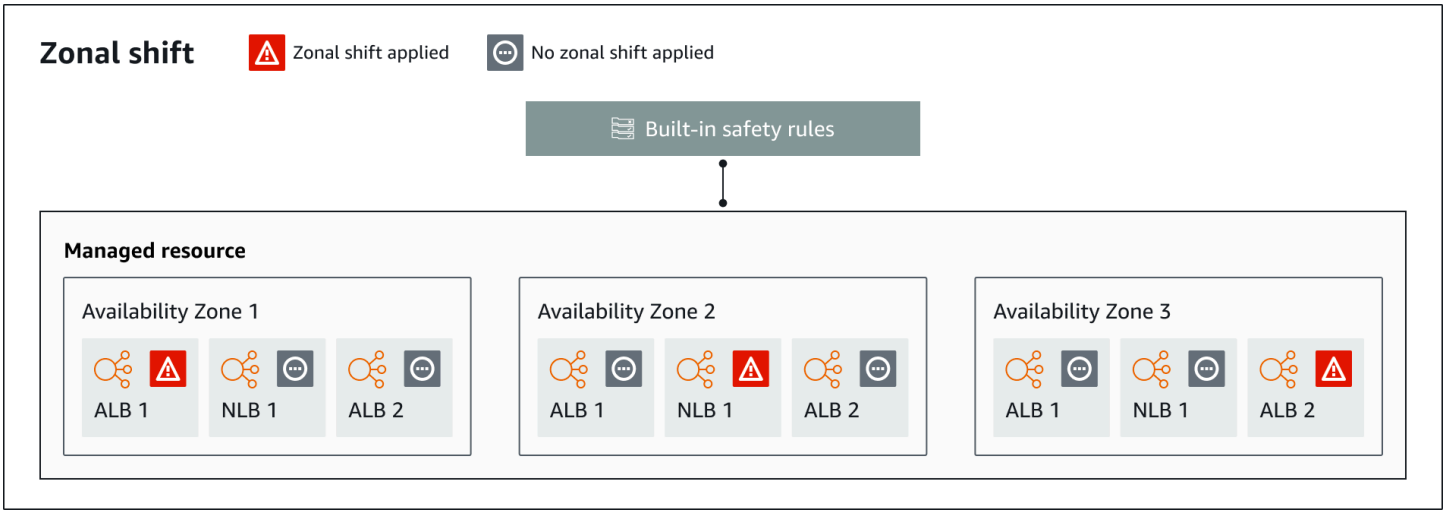
区域名称	区域	端点	协议
亚太地区（墨尔本）	ap-southeast-4	arc-zonal-shift.ap-southeast-4.amazonaws.com	HTTPS
		arc-zonal-shift.ap-southeast-4.api.aws	HTTPS
亚太地区（孟买）	ap-south-1	arc-zonal-shift.ap-south-1.amazonaws.com	HTTPS
		arc-zonal-shift.ap-south-1.api.aws	HTTPS
亚太地区（大阪）	ap-northeast-3	arc-zonal-shift.ap-northeast-3.amazonaws.com	HTTPS
		arc-zonal-shift.ap-northeast-3.api.aws	HTTPS
亚太地区（首尔）	ap-northeast-2	arc-zonal-shift.ap-northeast-2.amazonaws.com	HTTPS
		arc-zonal-shift.ap-northeast-2.api.aws	HTTPS
亚太地区（新加坡）	ap-southeast-1	arc-zonal-shift.ap-southeast-1.amazonaws.com	HTTPS
		arc-zonal-shift.ap-southeast-1.api.aws	HTTPS
亚太地区（悉尼）	ap-southeast-2	arc-zonal-shift.ap-southeast-2.amazonaws.com	HTTPS
		arc-zonal-shift.ap-southeast-2.api.aws	HTTPS
亚太地区（台北）	ap-east-2	arc-zonal-shift.ap-east-2.amazonaws.com	HTTPS
		arc-zonal-shift.ap-east-2.api.aws	HTTPS
亚太地区（泰国）	ap-southeast-7	arc-zonal-shift.ap-southeast-7.amazonaws.com	HTTPS
		arc-zonal-shift.ap-southeast-7.api.aws	HTTPS
亚太地区（东京）	ap-northeast-1	arc-zonal-shift.ap-northeast-1.amazonaws.com	HTTPS
		arc-zonal-shift.ap-northeast-1.api.aws	HTTPS

区域名称	区域	端点	协议
加拿大（中部）	ca-central-1	arc-zonal-shift.ca-central-1.amazonaws.com	HTTPS
		arc-zonal-shift-fips.ca-central-1.api.aws	HTTPS
		arc-zonal-shift.ca-central-1.api.aws	HTTPS
加拿大西部（卡尔加里）	ca-west-1	arc-zonal-shift.ca-west-1.amazonaws.com	HTTPS
		arc-zonal-shift-fips.ca-west-1.api.aws	HTTPS
		arc-zonal-shift.ca-west-1.api.aws	HTTPS
欧洲地区（法兰克福）	eu-central-1	arc-zonal-shift.eu-central-1.amazonaws.com	HTTPS
		arc-zonal-shift.eu-central-1.api.aws	HTTPS
欧洲地区（爱尔兰）	eu-west-1	arc-zonal-shift.eu-west-1.amazonaws.com	HTTPS
		arc-zonal-shift.eu-west-1.api.aws	HTTPS
欧洲地区（伦敦）	eu-west-2	arc-zonal-shift.eu-west-2.amazonaws.com	HTTPS
		arc-zonal-shift.eu-west-2.api.aws	HTTPS
欧洲地区（米兰）	eu-south-1	arc-zonal-shift.eu-south-1.amazonaws.com	HTTPS
		arc-zonal-shift.eu-south-1.api.aws	HTTPS
欧洲地区（巴黎）	eu-west-3	arc-zonal-shift.eu-west-3.amazonaws.com	HTTPS
		arc-zonal-shift.eu-west-3.api.aws	HTTPS
欧洲（西班牙）	eu-south-2	arc-zonal-shift.eu-south-2.amazonaws.com	HTTPS
		arc-zonal-shift.eu-south-2.api.aws	HTTPS
欧洲地区（斯德哥尔摩）	eu-north-1	arc-zonal-shift.eu-north-1.amazonaws.com	HTTPS
		arc-zonal-shift.eu-north-1.api.aws	HTTPS

区域名称	区域	端点	协议
欧洲（苏黎世）	eu-central-2	arc-zonal-shift.eu-central-2.amazonaws.com	HTTPS
		arc-zonal-shift.eu-central-2.api.aws	HTTPS
以色列（特拉维夫）	il-central-1	arc-zonal-shift.il-central-1.amazonaws.com	HTTPS
		arc-zonal-shift.il-central-1.api.aws	HTTPS
墨西哥（中部）	mx-central-1	arc-zonal-shift.mx-central-1.amazonaws.com	HTTPS
		arc-zonal-shift.mx-central-1.api.aws	HTTPS
中东（巴林）	me-south-1	arc-zonal-shift.me-south-1.amazonaws.com	HTTPS
		arc-zonal-shift.me-south-1.api.aws	HTTPS
中东（阿联酋）	me-central-1	arc-zonal-shift.me-central-1.amazonaws.com	HTTPS
		arc-zonal-shift.me-central-1.api.aws	HTTPS
南美洲（圣保罗）	sa-east-1	arc-zonal-shift.sa-east-1.amazonaws.com	HTTPS
		arc-zonal-shift.sa-east-1.api.aws	HTTPS
AWS GovCloud（美国东部）	us-gov-east-1	arc-zonal-shift.us-gov-east-1.amazonaws.com	HTTPS
		arc-zonal-shift-fips.us-gov-east-1.api.aws	HTTPS
		arc-zonal-shift.us-gov-east-1.api.aws	HTTPS
AWS GovCloud（美国西部）	us-gov-west-1	arc-zonal-shift.us-gov-west-1.amazonaws.com	HTTPS
		arc-zonal-shift-fips.us-gov-west-1.api.aws	HTTPS
		arc-zonal-shift.us-gov-west-1.api.aws	HTTPS

可用区转移组件

下图说明了区域转移将流量从可用区转移的示例。 AWS 区域区域偏移中内置的检查可防止您在资源已经处于活动移位状态时开始另一个区域移动。



以下是 ARC 中分区偏移能力的组成部分。

可用区转移

您可以开始对 AWS 账户中的托管资源进行区域切换，以暂时将流量从该区域的可用区转移到正常 AZs 可用区 AWS 区域，从而快速从一个可用区出现的问题中恢复过来。有关支持区域偏移资源的更多信息，请参阅。支持的资源

内置安全检查

ARC 中内置的检查可防止一个资源的多个流量转移同时生效。也就是说，只有一次由客户发起的区域转移、练习运行或资源自动切换才能主动将流量从可用区转移出去。例如，如果您对某个资源启动可用区转移，而该资源当前正在通过自动转移而转移出去，则优先进行可用区转移。有关更多信息，请参阅 ARC 中的区域自动换档和练习运行结果。

资源标识符

要包含在可用区转移中的资源的标识符。资源标识符是资源的 Amazon 资源名称 (ARN)。

对于区域转移，您只能在账户中为 ARC 支持的 AWS 服务选择资源。有关支持区域偏移资源的更多信息，请参阅。支持的资源

托管资源

有些 AWS 资源必须手动选择启用区域移动，而另一些资源则会自动启用。有关支持区域偏移资源的更多信息，请参阅。支持的资源

资源名称

ARC 中可以为区域偏移指定的资源名称。

状态（可用区转移状态）

可用区转移的状态。可用区转移的 Status 可以具有以下值之一：

- ACTIVE：可用区转移已启动并处于活动状态。
- EXPIRED：可用区转移已到期（已超过到期时间）。
- CANCELED：可用区转移已取消。

已应用状态

已应用状态表示资源的转移是否有效。状态为的转移APPLIED决定了资源应用程序流量转移到哪个可用区，以及该转移何时结束。

班次类型

定义分区偏移类型。shiftType可以具有以下值：

- Zonal_shift
- 区域_自动切换
- 练习跑
- FIS_实验

到期时间（过期时间）

可用区转移的到期时间（过期时间）。可用区转移是暂时的。对于区域偏移，您最初可以将区域偏移设置为最长三天（72 小时）处于活动状态。

当你开始区域偏移时，你可以指定你希望它在多长时间内处于活动状态，ARC 会将其转换为到期时间（到期时间）。您可以取消可用区转移，例如您准备好将流量恢复到可用区时。您也可以通过更新客户发起的可用区转移，指定另一个到期时长，从而延长其时间。

你可以取消作为分区自动换档一部分的分区移位练习。

用于区域偏移的数据和控制平面

在规划故障转移和灾难恢复时，请考虑故障转移机制的弹性。我们建议您确保在故障转移期间所依赖的机制具有高可用性，以便在灾难情况下可以根据需要使用它们。通常，应尽可能为机制使用数据平面函数，以获得最大的可靠性和容错性。考虑到这一点，请务必了解服务的功能如何在控制面板和数据面板之间划分，以及何时可以依赖服务的数据面板可预期的极高可靠性。

与大多数 AWS 服务一样，控制平面和数据平面支持区域移位功能的功能。虽然这两者都是为了可靠而构建的，但控制平面针对数据一致性进行了优化，而数据平面则针对可用性进行了优化。数据面板专为弹性而设计，因此即使在中断事件期间，当控制面板可能不可用时，它也能保持可用性。

一般而言，控制面板允许您执行基本的管理功能，例如在服务中创建、更新和删除资源。数据面板提供服务的核心功能。

有关数据平面、控制平面以及如何 AWS 构建服务以满足高可用性目标的更多信息，请参阅 Amazon Builders Library 中的 “使用可用区的静态稳定性” 论文。

ARC 中区域偏移的定价

对于区域转移，您可以开始对支持的资源进行区域切换，以使您的应用程序从可用区出现的问题中恢复。使用可用区转移不收取任何额外费用。

有关 ARC 的详细定价信息和定价示例，请参阅 ARC 定价。

ARC 中的可用区转移最佳实践

对于在 ARC 中使用区域转移进行多可用区恢复，我们推荐以下最佳实践。

主题

- 容量规划和预扩展
- 限制客户端与您的终端保持连接的时间
- 提前测试开始区域偏移
- 确保所有可用区域都运行良好并占用流量
- 使用数据平面 API 操作进行灾难恢复
- 只能暂时通过区域偏移来移动交通

容量规划和预扩展

确保您已计划好并且已经预扩展或可以自动扩展足够的容量，以适应可用区转移启动时给可用区施加的额外负载。对于面向恢复的架构，典型的建议是预扩展计算容量，保留足够的余量，以便在三个副本（典型情况下）之一离线时承担流量高峰。

当您开始对支持的资源进行区域转移并且流量从可用区转移出去时，您的应用程序用于服务请求的容量就会被移除。您必须确保已计划将流量从可用区转移出去，并且可以继续处理其余可用区的请求 AZs。

限制客户端与您的终端保持连接的时间

当 Amazon Application Recovery Controller (ARC) 将流量从受损中转移出去时，例如使用区域转移或区域自动切换，ARC 用来转移应用程序流量的机制是 DNS 更新。DNS 更新会导致所有新连接都被定向到远离受损位置。

但是，在客户端重新连接之前，具有已打开连接的客户端可能会继续向受损位置发出请求。为确保快速恢复，我们建议您限制客户端与您的终端保持连接的时间。

提前测试开始区域偏移

通过启动可用区转移，定期测试从可用区移走应用程序的流量。最好是同时在测试和生产环境中计划和执行可用区转移，并将该过程纳入到定期的失效转移测试中，以测试发生灾难时恢复应用程序的能力。要确保您已准备好并有信心在运营事件发生时缓解问题，定期测试是一个关键环节。

确保所有可用区域都运行良好并占用流量

可用区转移的工作原理是，将某可用区中的一个资源（即应用程序副本）标记为运行状况不佳。这意味着，确保应用程序中的资源总体运行状况良好，并在某个区域的可用区域中积极获取流量至关重要。我们建议您使用仪表板来跟踪该情况，包括针对不正常目标的 Elastic Load Balancing 指标和每个可用区的 bytesProcessed 等。

考虑从第二个相邻区域监控资源的运行状况。这种方法的优势在于，它可以更好地代表您的最终用户的体验，还可以降低您的应用程序和监控同时受到同一灾难影响的风险。

使用数据平面 API 操作进行灾难恢复

要在需要快速恢复几乎没有依赖关系的应用程序时开始区域移动，我们建议使用 AWS Command Line Interface 或 API，并尽可能使用带有预存储凭据的区域移位操作和预先存储的凭据。为了便于使用 AWS Management Console，您也可以在中开始区域移动。但是，当快速、可靠的恢复至关重要时，数据面板操作是更好的选择。有关更多信息，请参阅可用区转移 API 参考指南。

只能暂时通过区域偏移来移动交通

可用区转移会暂时将流量从可用区移走，以减轻损失。在采取措施纠正问题后，应立即将应用程序资源恢复为可用。这样能确保整个应用程序恢复到原始的完全冗余的弹性状态。

可用区转移 API 操作

下表列出了您可以使用区域转移的 ARC API 操作，该操作可将流量从可用区转移出多可用区应用程序的可用区。该表还包括相关文档的链接。

有关如何在 AWS Command Line Interface中使用常见可用区转移 API 操作的示例，请参阅使用 AWS CLI 带区域移位的示例。

操作	使用 ARC 控制台	使用 ARC API
启动可用区转移	请参阅 启动可用区转移。	请参阅 StartZonalShift
更新可用区转移	请参阅 更新或取消可用区转移。	请参阅 UpdateZonalShift
列出可用区转移	请参阅 ARC 中的区域偏移。	请参阅 ListZonalShifts
列出托管资源	请参阅 支持的资源。	请参阅 ListManagedResources
获取托管资源	请参阅 支持的资源。	请参阅 GetManagedResource
取消可用区转移	请参阅 更新或取消可用区转移。	请参阅 CancelZonalShift

使用 AWS CLI 带区域移位的示例

本节提供了使用区域偏移的应用示例，以及使用 API 操作在 Amazon 应用程序恢复控制器 (ARC) 中使用区域偏移功能。 AWS Command Line Interface 这些示例旨在帮助您基本了解如何使用 CLI 处理区域偏移。

ARC 中的区域切换允许您暂时将受支持资源的流量从可用区移开，这样您的应用程序就可以继续在中的其他可用区域正常运行。 AWS 区域

所有可用区转移都是暂时性的，最初必须设置为三天内到期。但是您后期可以更新可用区转移，以设置新的到期时间。

有关使用的更多信息 AWS CLI，请参阅《AWS CLI 命令参考》。有关可用区转移 API 操作的列表和指向更多信息的链接，请参阅可用区转移 API 操作。

启动可用区转移

您可以使用 `start-zonal-shift` 命令在 CLI 中启动可用区转移。

```
aws arc-zonal-shift start-zonal-shift \
```

```
       --resource-identifier arn:aws:elasticloadbalancing:us-
east-1:111122223333:loadbalancer/app/Testing/5a19403ecd42dc05 \
       --away-from use1-az1 \
       --expires-in 10m \
       --comment "Shifting traffic away from use1-az1"
```

```
{
    "awayFrom": "use1-az1",
    "comment": "Shifting traffic away from use1-az1",
    "expiryTime": "2024-12-17T21:37:26-08:00",
    "resourceIdentifier": "arn:aws:elasticloadbalancing:us-
east-1:111122223333:loadbalancer/app/Testing/5a19403ecd42dc05",
    "startTime": "2024-12-17T21:27:26-08:00",
    "status": "ACTIVE",
    "zonalShiftId": "9ac9ec1e-1df1-0755-3dc5-8cf573cd9c38"
}
```

获取托管资源

您可以使用 get-managed-resource 命令在 CLI 中获取有关托管资源的信息。

```
aws arc-zonal-shift get-managed-resource \
       --resource-identifier arn:aws:elasticloadbalancing:us-
east-1:111122223333:loadbalancer/app/Testing/5a19403ecd42dc05
```

```
{
    "appliedWeights": {
        "use1-az1": 0.0,
        "use1-az2": 1.0,
        "use1-az6": 1.0
    },
    "arn": "arn:aws:elasticloadbalancing:us-east-1:111122223333:loadbalancer/app/
Testing/5a19403ecd42dc05",
    "autoshifts": [],
    "name": "Testing",
    "zonalAutoshiftStatus": "DISABLED",
    "zonalShifts": [
        {
            "appliedStatus": "APPLIED",
            "awayFrom": "use1-az1",
            "comment": "Shifting traffic away from use1-az1",
            "expiryTime": "2024-12-17T21:37:26-08:00",
```

```
            "resourceIdentifier": "arn:aws:elasticloadbalancing:us-
east-1:111122223333:loadbalancer/app/Testing/5a19403ecd42dc05",
            "startTime": "2024-12-17T21:27:26-08:00",
            "zonalShiftId": "9ac9ec1e-1df1-0755-3dc5-8cf573cd9c38"
            "shiftType": "MANUAL"
        }
    ]
}
```

列出托管资源

您可以使用 `list-managed-resources` 命令在 CLI 中列出您账户中的托管资源。

```
aws arc-zonal-shift list-managed-resources
```

```
{
    "items": [
        {
            "appliedWeights": {
                "use1-az1": 0.0,
                "use1-az2": 1.0,
                "use1-az6": 1.0
            },
            "arn": "arn:aws:elasticloadbalancing:us-east-1:111122223333:loadbalancer/
app/Testing/5a19403ecd42dc05",
            "autoshifts": [],
            "availabilityZones": [
                "use1-az1",
                "use1-az2",
                "use1-az6"
            ],
            "name": "Testing",
            "practiceRunStatus": "DISABLED",
            "zonalAutoshiftStatus": "DISABLED",
            "zonalShifts": [
                {
                    "appliedStatus": "APPLIED",
                    "awayFrom": "use1-az1",
                    "comment": "Shifting traffic away from use1-az1",
                    "expiryTime": "2024-12-17T21:37:26-08:00",
                    "resourceIdentifier": "arn:aws:elasticloadbalancing:us-
east-1:111122223333:loadbalancer/app/Testing/5a19403ecd42dc05",
                    "startTime": "2024-12-17T21:27:26-08:00",
```

```
                    "zonalShiftId": "9ac9ec1e-1df1-0755-3dc5-8cf573cd9c38"
                }
            ]
        }
    ]
}
```

列出可用区转移

您可以使用 list-zonal-shifts 命令在 CLI 中列出您账户中的可用区转移。

```
aws arc-zonal-shift list-zonal-shifts
```

```
{
    "items": [
        {
            "awayFrom": "use1-az1",
            "comment": "Shifting traffic away from use1-az1",
            "expiryTime": "2024-12-17T21:37:26-08:00",
            "resourceIdentifier": "arn:aws:elasticloadbalancing:us-
east-1:111122223333:loadbalancer/app/Testing/5a19403ecd42dc05",
            "startTime": "2024-12-17T21:27:26-08:00",
            "status": "ACTIVE",
            "zonalShiftId": "9ac9ec1e-1df1-0755-3dc5-8cf573cd9c38"
        }
    ]
}
```

更新可用区转移

您可以使用 update-zonal-shift 命令在 CLI 中更新可用区转移。

```
aws arc-zonal-shift update-zonal-shift \
       --zonal-shift-id 9ac9ec1e-1df1-0755-3dc5-8cf573cd9c38 \
       --expires-in 1h \
       --comment "Still shifting traffic away from use1-az1"
```

```
{
    "awayFrom": "use1-az1",
    "comment": "Still shifting traffic away from use1-az1",
```

```
    "expiryTime": "2024-12-17T22:29:38-08:00",
    "resourceIdentifier": "arn:aws:elasticloadbalancing:us-
east-1:111122223333:loadbalancer/app/Testing/5a19403ecd42dc05",
    "startTime": "2024-12-17T21:27:26-08:00",
    "status": "ACTIVE",
    "zonalShiftId": "9ac9ec1e-1df1-0755-3dc5-8cf573cd9c38"
}
```

取消可用区转移

您可以使用 `cancel-zonal-shift` 命令在 CLI 中取消可用区转移。

```
aws arc-zonal-shift cancel-zonal-shift \
       --zonal-shift-id 9ac9ec1e-1df1-0755-3dc5-8cf573cd9c38
```

```
{
    "awayFrom": "use1-az1",
    "comment": "Still shifting traffic away from use1-az1",
    "expiryTime": "2024-12-17T22:29:38-08:00",
    "resourceIdentifier": "arn:aws:elasticloadbalancing:us-
east-1:111122223333:loadbalancer/app/Testing/5a19403ecd42dc05",
    "startTime": "2024-12-17T21:27:26-08:00",
    "status": "CANCELED",
    "zonalShiftId": "9ac9ec1e-1df1-0755-3dc5-8cf573cd9c38"
}
```

支持的资源

Amazon 应用程序恢复控制器 (ARC) 目前支持为区域转移和区域自动移位启用以下资源：

- Amazon A EC2 uto Scaling 群组
- Amazon Elastic Kubernetes Service
- 应用程序负载均衡器启用或禁用跨区域负载均衡
- 网络负载均衡器启用或禁用跨区域负载均衡

有关网络负载均衡器和应用程序负载均衡器的具体要求，请参阅本节中的其他主题。

在 ARC 中查看以下使用区域偏移、区域自动移位和资源的条件：

- 资源必须处于活动状态并已完全预置，才能为其转移流量。在开始对资源进行区域转移之前，请检查该资源是否是 ARC 中的托管资源。例如，在中查看托管资源的列表 AWS Management Console，或者使用带有资源标识符的`get-managed-resource`操作。
- 要开始使用资源进行区域转移，必须将其部署在可用区以及您开始转移 AWS 区域 的地方。确保在要转移的可用区所在的同一个区域开始区域转移，并且要转移流量的资源也位于同一个可用区和区域。
- 确保您拥有对资源使用区域转移的正确 IAM 权限。有关更多信息，请参阅 IAM 和可用区转移权限。
- 当 Network Load Balancer 或 Application Load Balancer 处于失效打开状态时，区域偏移将无效。这是预期的行为，因为当负载均衡器无法打开时，区域转移不能强制可用区运行状况不佳，然后将流量转移到区域 AZs 中的另一个区域。有关更多信息，请参阅网络负载均衡器用户指南中的对负载均衡器使用 Route 53 DNS 故障转移和应用程序负载均衡器用户指南中的对负载均衡器使用 Route 53 DNS 故障转移。
- 如果多个负载均衡器将流量转发到相同的目标，则在启用跨区域的负载均衡器上进行区域转移将降低所有负载均衡器的目标容量，即使它们没有进行区域移动。

Amazon A EC2 uto Scaling 群组

Amazon A EC2 uto Scaling 组包含一组亚马逊 EC2 实例，出于自动扩展和管理的目的，这些实例被视为逻辑分组。Auto Scaling 组还允许您使用 Amazon A EC2 uto Scaling 功能，例如运行状况检查替换和扩展策略。Amazon Auto Scaling 服务的核心功能是保持 Auto Scaling 组中的实例数量和 EC2自动扩展。

对 Auto Scaling 群组使用区域偏移

要启用区域偏移，请使用以下方法之一。

Console

在新群组上启用区域切换（控制台）

1. 按照使用启动模板创建 Auto Scaling 组中的说明完成过程中的每个步骤，直到步骤 10。
2. 在 “与其他服务集成” 页面上，对于 ARC 区域移动，选中复选框以启用区域移动。
3. 对于运行状况检查行为，请选择忽略不健康状态或替换不健康状态。如果设置为`replace-unhealthy`，则可用区中运行状况不佳的实例将替换为有效的区域切换。如果设置为`ignore-unhealthy`，则可用区中运行状况不佳的实例将不会被活跃的区域切换所取代。
4. 继续执行使用启动模板创建 Auto Scaling 组中的步骤。

AWS CLI

要在新群组上启用区域偏移 ()AWS CLI

向 create-auto-scaling-group 命令添加 `--availability-zone-impairment-policy` 参数。

该`--availability-zone-impairment-policy`参数有两个选项：

- ZonalShiftEnabled— 如果设置为`true`，Auto Scaling 将使用 ARC 区域偏移注册 Auto Scaling 组，您可以在 ARC 控制台上启动、更新或取消区域偏移。如果设置为`false`，则 Auto Scaling 会从 ARC 区域偏移中取消注册 Auto Scaling 组。必须已启用区域偏移才能将其设置为。`false`
- ImpairedZoneHealthCheckBehavior— 如果设置为`replace-unhealthy`，则可用区中运行状况不佳的实例将替换为有效的区域切换。如果设置为`ignore-unhealthy`，则可用区中运行状况不佳的实例将不会被活跃的区域切换所取代。

以下示例在名*my-asg*为的新 Auto Scaling 组上启用区域偏移。

```
aws autoscaling create-auto-scaling-group \
  --launch-template LaunchTemplateName=my-launch-template,Version='1' \
  --auto-scaling-group-name my-asg \
  --min-size 1 \
  --max-size 10 \
  --desired-capacity 5 \
  --availability-zones us-east-1a us-east-1b us-east-1c \
  --availability-zone-impairment-policy '{
      "ZonalShiftEnabled": true,
      "ImpairedZoneHealthCheckBehavior": IgnoreUnhealthy
    }'
```

Console

在现有群组上启用区域切换（控制台）

1. 在上打开亚马逊 EC2 控制台 https://console.aws.amazon.com/ec2/，然后从导航窗格中选择 Auto Scaling Gro ups。
2. 在屏幕顶部的导航栏中，选择您在其中创建了自动扩缩组的 AWS 区域 。
3. 选中 Auto Scaling 组旁边的复选框。

   这时将在页面底部打开一个拆分窗格。

4. 在集成选项卡上的 ARC 区域偏移下，选择编辑。
5. 选中该复选框以启用区域移动。
6. 对于运行状况检查行为，请选择 “忽略不健康状况” 或 “替换不健康”。如果设置为`replace-unhealthy`，则可用区中运行状况不佳的实例将替换为有效的区域切换。如果设置为`ignore-unhealthy`，则可用区中运行状况不佳的实例将不会被活跃的区域切换所取代。
7. 选择更新。

AWS CLI

要对现有群组启用区域移动 ()AWS CLI

向 update-auto-scaling-group 命令添加 `--availability-zone-impairment-policy` 参数。

该`--availability-zone-impairment-policy`参数有两个选项：

- ZonalShiftEnabled— 如果设置为`true`，Auto Scaling 将使用 ARC 区域偏移注册 Auto Scaling 组，您可以在 ARC 控制台上启动、更新或取消区域偏移。如果设置为`false`，则 Auto Scaling 会从 ARC 区域偏移中取消注册 Auto Scaling 组。必须已启用区域偏移才能将其设置为。`false`
- ImpairedZoneHealthCheckBehavior— 如果设置为`replace-unhealthy`，则可用区中运行状况不佳的实例将替换为有效的区域切换。如果设置为`ignore-unhealthy`，则可用区中运行状况不佳的实例将不会被活跃的区域切换所取代。

以下示例在指定的 Auto Scaling 组上启用区域偏移。

```
aws autoscaling update-auto-scaling-group --auto-scaling-group-name my-asg \
  --availability-zone-impairment-policy '{
      "ZonalShiftEnabled": true,
      "ImpairedZoneHealthCheckBehavior": IgnoreUnhealthy
    }'
```

要触发区域偏移，请参阅 启动、更新或取消区域偏移。

Auto Scaling 群组的区域偏移是如何运作的

假设您有一个包含以下可用区域的 Auto Scaling 组：

- `us-east-1a`
- `us-east-1b`

- us-east-1c

您会注意到失败us-east-1a并触发区域偏移。当触发区域偏移时，会发生以下行为。us-east-1a

- 向@@ 外扩展 — Auto Scaling 将在运行状况良好的可用区（us-east-1b和us-east-1c）中启动所有新的容量请求。
- 动态扩展 — Auto Scaling 将阻止扩展策略减少所需容量。Auto Scaling 不会阻止扩展策略增加所需容量。
- 实例刷新 — Auto Scaling 将延长在活动区域转移期间延迟的任何实例刷新过程的超时时间。

可用区运行状况检查行为选择受损	Health Check 行为
替换不健康的	所有可用区（us-east-1a 、us-east-1b 和us-east-1c ）中显示运行状况不佳的实例将被替换。
忽略不健康	显示为运行状况不佳的实例将在us-east-1b 和us-east-1c 中替换。可用区中的实例不会被有效的区域移动 (us-east-1a) 所取代。

使用区域偏移的最佳实践

为了在使用区域转移时保持应用程序的高可用性，我们建议采用以下最佳实践。

- 监控 EventBridge 通知以确定何时出现持续的可用区损坏事件。有关更多信息，请参阅使用事件桥自动执行 Amazon A EC2 uto Scaling。
- 使用具有适当阈值的扩展策略，确保您有足够的容量来容忍可用区的损失。
- 将实例维护策略设置为最低健康百分比为 100。使用此设置，Auto Scaling 会等待新实例准备就绪，然后再终止运行状况不佳的实例。

对于预先缩放的客户，我们还建议采取以下措施：

- 选择 I gnore un healthy 作为受损可用区的运行状况检查行为，因为在受损事件期间，您无需更换运行状况不佳的实例。

- 在 ARC 中为你的 Auto Scaling 组使用分区自动移位。中的区域自动切换功能 Amazon 应用程序恢复控制器 (ARC) 允许在 AWS 检测 AWS 到可用区存在障碍时将资源的流量从可用区转移出去。有关更多信息，请参阅《A mazon 应用程序恢复控制器 (ARC) 开发人员指南》中的 ARC 中的区域自动切换。

对于禁用跨区域负载均衡器的客户，我们还建议：

- 仅在可用区分配中使用平衡。
- 如果您在 Auto Scaling 组和负载均衡器上都使用区域偏移，请务必先取消您的 Auto Scaling 组的区域偏移。然后，等到所有可用区域的容量均衡后再取消负载均衡器上的区域切换。
- 由于启用区域转移并使用禁用跨区域的负载均衡器时，可能会出现容量不平衡的情况，因此 Auto Scaling 需要进行额外的验证。如果您遵循最佳实践，则可以通过选中中的复选框 AWS Management Console 或使用`CreateAutoScalingGroup`、`UpdateAutoScalingGroup`或中的`skip-zonal-shift-validation`标志来确认这种可能性`AttachTrafficSources`。

Amazon Elastic Kubernetes Service

Amazon EKS 提供的功能使您的应用程序能够更灵活地应对运行状况下降或可用区 (AZ) 受损等事件。在 Amazon EKS 集群中运行工作负载时，您可以使用区域转移或区域自动切换进一步改善应用程序环境的容错能力和应用程序恢复能力。

为亚马逊 Elastic Kubernetes Service 使用区域偏移 Amazon Kubernetes Service

要启用区域切换，请使用以下方法之一。有关更多信息，请参阅启用 Amazon EKS 区域转移以避免可用区域受损。

Console

在新的 Amazon EKS 集群上启用区域切换（控制台）

1. 找到您要向 ARC 注册的 Amazon EKS 集群的名称和区域。
2. 在 https://console.aws.amazon.com/eks/home#/ 集群中打开 Amazon EKS 控制台。
3. 选择您的集群。
4. 在集群信息页面上，选择概述选项卡。
5. 在可用区转移标题下，选择管理按钮。
6. 选择 “启用” 或 “禁用 EKS 区域移动”。

AWS CLI

在新的 Amazon EKS 集群上启用区域切换 ()AWS CLI

- 输入以下命令：

```
aws eks create-cluster --name my-eks-cluster --role-
arn my-role-arn-to-create-cluster --resources-vpc-config
 subnetIds=string,string,securityGroupIds=string,string,endpointPublicAccess=boolean,end
 --zonal-shift-config enabled=true
```

要在现有 Amazon EKS 集群上启用区域切换 ()AWS CLI

- 输入以下命令：

```
aws eks update-cluster-config --name my-eks-cluster --zonal-shift-config
 enabled=true
```

您可以为 Amazon EKS 集群触发区域移动，也可以通过启用区域自动移位 AWS 来允许您执行此操作。使用 ARC 启用 Amazon EKS 集群区域偏移后，您可以使用 ARC 控制台、CL AWS I 或区域偏移和区域自动移位触发区域偏移或启用区域自动切换。 APIs

有关触发区域偏移的更多信息，请参阅 启动、更新或取消区域偏移。

有关为亚马逊 EKS 启用区域偏移的更多信息，请参阅亚马逊 Ela stic Kubernetes Servic e 用户指南中的 “了解亚马逊 EKS 中的 ARC 区域偏移” 主题。

亚马逊 Elastic Kubernetes Service 的区域偏移是如何运作的

在 Amazon EKS 区域转移期间，将自动进行以下操作：

- 受影响可用区中的所有节点都将被封锁。这将防止 Kubernetes 调度器将新容器组（pod）调度到运行状况不佳的可用区中的节点上。
- 如果您使用的是托管节点组，则可用区重新平衡将被暂停，并且您的 Auto Scaling 组 (ASG) 也将更新，以确保新的 Amazon EKS 数据平面节点仅在正常运行状态下启动。 AZs
- 运行状况不佳的可用区中的节点不会被终止，容器组（pod）也不会被逐出这些节点。这是为了确保当区域转移到期或被取消时，您的流量可以安全地返回到仍处于满负荷状态的可用区。

- EndpointSlice 控制器将在受损的可用区中找到所有 Pod 端点，并将其从相关可用区中移除 EndpointSlices。这将确保只有运行状况良好 AZs 的 Pod 端点才会成为接收网络流量的目标。当区域转移取消或到期时， EndpointSlice 控制器将更新 EndpointSlices 以包括已恢复的可用区中的端点。

有关更多信息，请参阅AWS 容器博客。

应用程序负载均衡器

对应用程序负载均衡器使用区域偏移

要使用具有区域偏移功能的应用程序负载均衡器，必须在 Application Load Balancer 属性中启用 ARC 区域偏移集成。Application Load Balancer 在启用跨区域或禁用跨区域配置的情况下支持区域移动。

在启用 ARC 集成并开始使用区域偏移之前，请查看以下内容：

- 只能为单个可用区中的特定负载均衡器启动可用区转移。无法为多个可用区启动可用区转移。
- AWS 当多个基础设施问题影响服务时，主动从 DNS 中删除区域负载均衡器 IP 地址。在开始可用区转移之前，请务必检查当前的可用区容量。
- 当应用程序负载均衡器是网络负载均衡器的目标时，请始终从网络负载均衡器启动可用区转移。如果从应用程序负载均衡器启动可用区转移，则网络负载均衡器将不会识别转移，并继续向应用程序负载均衡器发送流量。

您可以在 Elastic Load Balancing 控制台（大多数情况下 AWS 区域）或 ARC 控制台中启动负载均衡器的区域切换。

Console

在负载均衡器上启用区域切换（控制台）

1. 打开亚马逊 EC2 控制台，网址为https://console.aws.amazon.com/ec2/。
2. 在导航页面的负载平衡下，选择负载均衡器。
3. 选择 App lication Load Balancer 名称。
4. 在属性选项卡上，选择编辑。
5. 在可用区路由配置下，将 ARC 可用区转移集成设置为启用。
6. 选择保存。

AWS CLI

在负载均衡器上启用区域切换 ()AWS CLI

- 输入以下命令：

```
aws elbv2 modify-load-balancer-attributes --load-balancer-arn my-alb-arn --
attributes Key=zonal_shift.config.enabled,Value=true
```

有关触发区域偏移的更多信息，请参阅 启动、更新或取消区域偏移。

您可以使用该keepalive选项来配置连接的持续时间。有关更多信息，请参阅《Application Load Balancer 用户指南》中的 HTTP 客户端保持连接时长。默认情况下，应用程序负载均衡器将 HTTP 客户端 keepalive 持续时间值设置为 3600 秒或 1 小时。我们建议您降低该值，使其与应用程序的恢复时间目标保持一致，例如 300 秒。选择 HTTP 客户端 keepalive 持续时间时，请考虑此值是在更频繁地重新连接（这可能会影响延迟）和更快地将所有客户端从受损的可用区或区域移出受损的可用区或区域之间进行权衡。

应用程序负载均衡器的区域偏移是如何工作的

在启用跨区域负载平衡的 Application Load Balancer 上开始区域转移时，所有目标流量都将在受影响的可用区中被阻止，并从 DNS 中删除区域 IP 地址。

有关更多信息，请参阅 App lication Load Balancer 用户指南中的应用程序负载均衡器集成。

网络负载均衡器

对网络负载均衡器使用区域偏移

要使用具有区域偏移功能的网络负载均衡器，必须在 Network Load Balancer 属性中启用 ARC 区域偏移集成。Network Load Balancer 支持启用跨区域或禁用跨区域配置的区域移动。

您可以选择使用哪些资源来使用区域移位和区域自动切换，以及何时要从受损的可用区中进行故障切换。支持面向 Internet 的网络负载均衡器和内部网络负载均衡器。

要为启用跨区域的 Network Load Balancer 启用区域切换，连接到负载均衡器的所有目标组都必须满足以下要求。

- 必须启用跨区域负载均衡，或将其设置为。use_load_balancer_configuration
  - 有关目标组跨区域负载平衡的更多信息，请参阅目标组的跨区域负载均衡。

- 目标组协议必须是 TCP 或 TLS。
  - 有关 Network Load Balancer 目标组协议的更多信息，请参阅路由配置。
- 必须禁用运行状况不佳的目标的连接终止。
  - 有关终止目标组连接的更多信息，请参阅终止运行状况不佳的目标的连接。
- 目标组不得将任何应用程序负载均衡器作为目标。
  - 有关应用程序负载均衡器作为目标的更多信息，请参阅使用应用程序负载均衡器作为 Network Load Balancer 的目标。

您可以使用 AWS CLI、 AWS 控制台或 Elastic Load Balancing 微件开始网络负载均衡器的区域切换。当应用程序负载均衡器成为网络负载均衡器的目标时，您必须从网络负载均衡器开始区域切换。如果您从应用程序负载均衡器开始区域切换，则网络负载均衡器不会停止向应用程序负载均衡器及其目标发送流量。

Console

在负载均衡器上启用区域切换（控制台）

1. 打开亚马逊 EC2 控制台，网址为https://console.aws.amazon.com/ec2/。
2. 在导航页面的负载平衡下，选择负载均衡器。
3. 选择 Network Load Balancer 名称。
4. 在属性选项卡上，选择编辑。
5. 在可用区路由配置下，将 ARC 可用区转移集成设置为启用。
6. 选择保存。

AWS CLI

在负载均衡器上启用区域切换 ()AWS CLI

- 输入以下命令：

```
aws elbv2 modify-load-balancer-attributes --load-balancer-arn my-nlb-arn --
attributes Key=zonal_shift.config.enabled,Value=true
```

有关触发区域偏移的更多信息，请参阅 启动、更新或取消区域偏移。

网络负载均衡器的区域偏移是如何工作的

ARC 会导致注册的网络负载均衡器的运行状况检查失败，因此当您触发区域转移时，受损可用区中的网络负载均衡器节点将从 DNS 中删除。网络负载均衡器将禁用受影响区域中的目标，使其停止接收流量，而 Elastic Load Balancing 会通过区域偏移将这些目标视为已禁用的目标。处于禁用状态的目标将继续接受运行状况检查。当目标处于健康状态并且区域偏移到期（或被取消）时，将恢复到先前受损区域中目标的路由。

在启用跨区域负载均衡的网络负载均衡器上进行可用区转移期间，将从 DNS 中移除可用区负载均衡器 IP 地址。与受损可用区中目标的现有连接会一直持续，直到它们自然关闭，而新的连接将不再路由到受损可用区中的目标。

有关更多信息，请参阅《网络负载均衡器用户指南》中的 “网络负载均衡器的区域偏移” 主题。

启动、更新或取消区域偏移

本节提供了处理区域偏移的程序，包括开始区域偏移和取消区域偏移。

启动可用区转移

本节中的步骤说明了如何在 Amazon 应用程序恢复控制器 (ARC) 控制台上启动客户启动的区域转移。要以编程方式使用可用区转移，请参阅《可用区转移 API 参考指南》。

除了在 ARC 中启动区域切换外，您还可以在 Elastic Load Balancing 控制台（在支持的区域中）中为负载均衡器启动区域切换。有关更多信息，请参阅《Elastic Load Balancing 用户指南》中的区域偏移。

启动可用区转移

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 在多可用区下，选择可用区转移。
3. 在可用区转移页面上，选择启动可用区转移。
4. 选择您要转移流量的可用区。
5. 从 “资源” 表中选择要转移流量的支持的资源。
6. 在设置可用区转移到期时间中，选择或输入可用区转移的到期时间。可用区转移的活动时间最初可设置为 1 分钟到三天（72 小时）。

   所有可用区转移都是暂时的。您必须设置到期时间，但稍后可以更新活动的可用区转移，以设置新的到期时间，最长是三天。

7. 输入注释。如果您愿意，可以稍后更新可用区转移以编辑注释。
8. 选中该复选框以确认启动可用区转移，这会将流量移离该可用区，从而减少应用程序的可用容量。
9. 选择启动。

更新或取消可用区转移

本节中的步骤说明了如何在 Amazon 应用程序恢复控制器 (ARC) 控制台上更新您启动的区域偏移或取消区域偏移。要以编程方式使用可用区转移，请参阅《可用区转移 API 参考指南》。

您可以更新可用区转移，以设置新的到期时间，也可以编辑或替换可用区转移的注释。在可用区转移到期之前，您可以随时取消它。

你可以取消你启动的区域移动，也可以取消为区域自动移位练习跑而 AWS 开始的区域移动。要详细了解区域自动换档中的练习移动，请参阅。可用区自动转移和练习运行的工作原理

更新可用区转移

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 在多可用区下，选择可用区转移。
3. 选择要更新的可用区转移，然后选择更新可用区转移。
4. 对于设置可用区转移到期时间，可以选择或输入到期时间。
5. 对于 Comment（注释），可以选择编辑现有注释或输入新注释。
6. 选择更新。

取消可用区转移

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 在多可用区下，选择可用区转移。
3. 选择要取消的可用区转移，然后选择取消可用区转移。
4. 在确认模态对话框中，选择确认。

在 Amazon 应用程序恢复控制器 (ARC) 中记录和监控区域偏移

您可以使用 AWS CloudTrail 监控 Amazon 应用程序恢复控制器 (ARC) 中的区域偏移，以分析模式并帮助解决问题。

主题

使用记录区域移动 API 调用 AWS CloudTrail

ARC 的区域切换与 AWS CloudTrail一项服务集成，该服务提供用户、角色或 AWS 服务在 ARC 中采取的操作的记录。 CloudTrail 将所有用于区域偏移的 API 调用捕获为事件。捕获的调用包括来自 ARC 控制台的调用和用于区域移位的 ARC API 操作的代码调用。

如果您创建跟踪，则可以将 CloudTrail 事件持续传输到 Amazon S3 存储桶，包括用于区域转移的事件。如果您未配置跟踪，您仍然可以在 CloudTrail 控制台的 “事件历史记录” 中查看最新的事件。

使用收集的信息 CloudTrail，您可以确定向 ARC 发出的区域转移请求、发出请求的 IP 地址、谁提出了请求、何时提出请求以及其他详细信息。

要了解更多信息 CloudTrail，请参阅《AWS CloudTrail 用户指南》。

区域偏移信息在 CloudTrail

CloudTrail 在您创建账户 AWS 账户 时已在您的账户上启用。当活动在 ARC 中发生区域偏移时，该活动将与其他 AWS 服务 CloudTrail 事件一起记录在事件历史记录中。您可以在中查看、搜索和下载最近发生的事件 AWS 账户。有关更多信息，请参阅使用 CloudTrail 事件历史记录。

要持续记录您的事件 AWS 账户，包括 ARC 中区域偏移的事件，请创建一条跟踪。跟踪允许 CloudTrail 将日志文件传输到 Amazon S3 存储桶。预设情况下，在控制台中创建跟踪记录时，此跟踪记录应用于所有 AWS 区域。跟踪记录 AWS 分区中所有区域的事件，并将日志文件传送到您指定的 Amazon S3 存储桶。此外，您还可以配置其他 AWS 服务，以进一步分析和处理 CloudTrail 日志中收集的事件数据。有关更多信息，请参阅下列内容：

- 创建跟踪记录概述
- CloudTrail 支持的服务和集成
- 配置 Amazon SNS 通知 CloudTrail
- 接收来自多个区域的 CloudTrail 日志文件和接收来自多个账户的 CloudTrail 日志文件

所有 ARC 操作均由 Amazon 应用程序恢复控制器的路由控制 API 参考指南记录 CloudTrail 并记录在案。例如，调用`StartZonalShift`和`ListManagedResources`操作会在 CloudTrail 日志文件中生成条目。

每个事件或日志条目都包含有关生成请求的人员信息。身份信息有助于您确定以下内容：

- 请求是使用根证书还是 AWS Identity and Access Management (IAM) 用户凭证发出。
- 请求是使用角色还是联合用户的临时安全凭证发出的。
- 请求是否由其他 AWS 服务发出。

有关更多信息，请参阅 CloudTrail userIdentity 元素。

在事件历史记录中查看 ARC 事件

CloudTrail 允许您在事件历史记录中查看最近的事件。有关更多信息，请参阅《AWS CloudTrail 用户指南》中的 “使用 CloudTrail 事件历史记录”。

了解分区移位日志文件条目

跟踪是一种配置，允许将事件作为日志文件传输到您指定的 Amazon S3 存储桶。 CloudTrail 日志文件包含一个或多个日志条目。事件代表来自任何来源的单个请求，包括有关请求的操作、操作的日期和时间、请求参数等的信息。 CloudTrail 日志文件不是公共 API 调用的有序堆栈跟踪，因此它们不会按任何特定的顺序出现。

以下示例显示了一个演示区域偏移`ListManagedResources`操作的 CloudTrail 日志条目。

```
{
      "eventVersion": "1.08",
      "userIdentity": {
        "type": "AssumedRole",
        "principalId": "A1B2C3D4E5F6G7EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/admin",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
          "sessionIssuer": {
            "type": "Role",
            "principalId": "AROA33L3W36EXAMPLE",
            "arn": "arn:aws:iam::111122223333:role/admin",
            "accountId": "111122223333",
            "userName": "EXAMPLENAME"
          },
          "webIdFederationData": {},
          "attributes": {
            "creationDate": "2022-11-14T16:01:51Z",
            "mfaAuthenticated": "false"
          }
```

```
        }
      },
      "eventTime": "2022-11-14T16:14:41Z",
      "eventSource": "arc-zonal-shift.amazonaws.com",
      "eventName": "ListManagedResources",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.50",
      "userAgent": "Boto3/1.17.101 Python/3.8.10 Linux/4.14.231-180.360.amzn2.x86_64
 exec-env/AWS_Lambda_python3.8 Botocore/1.20.102",
      "requestParameters": null,
      "responseElements": null,
      "requestID": "VGXG4ZUE7UZTVCMTJGIAF_EXAMPLE",
      "eventID": "4b5c42df-1174-46c8-be99-d67_EXAMPLE",
      "readOnly": true,
      "eventType": "AwsApiCall",
      "managementEvent": true,
      "recipientAccountId": "111122223333"
      "eventCategory": "Management"
      }
    }
```

以下示例显示了一个 CloudTrail 日志条目，该条目演示了带有区域偏移冲突异常的`StartZonalShift`操作。

```
{
      "eventVersion": "1.08",
      "userIdentity": {
        "type": "AssumedRole",
        "principalId": "A1B2C3D4E5F6G7EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/admin",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
          "sessionIssuer": {
            "type": "Role",
            "principalId": "AROA33L3W36EXAMPLE",
            "arn": "arn:aws:iam::111122223333:role/admin",
            "accountId": "111122223333",
            "userName": "EXAMPLENAME"
          },
          "webIdFederationData": {},
          "attributes": {
            "creationDate": "2022-11-14T16:01:51Z",
```

```
            "mfaAuthenticated": "false"
          }
        }
      },
      "eventTime": "2022-11-14T16:10:38Z",
      "eventSource": "arc-zonal-shift.amazonaws.com",
      "eventName": "StartZonalShift",
     "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.50",
      "userAgent": "Boto3/1.17.101 Python/3.8.10 Linux/4.14.231-180.360.amzn2.x86_64
 exec-env/AWS_Lambda_python3.8 Botocore/1.20.102",
      "errorCode": "ConflictException",
      "errorMessage": "There's already an active zonal shift for that resource
 identifier: 'arn:aws:testservice:us-west-2:077059137270:testResource/456apples'.
 Active zonal shift: 'bac23b74-176e-c073-de8f-484ca508910f'",
      "requestParameters": {
        "resourceIdentifier": "arn:aws:testservice:us-
west-2:077059137270:testResource/456apples",
        "awayFrom": "usw2-az1",
        "expiresIn": "2m",
        "comment": "HIDDEN_FOR_SECURITY_REASONS"
      },
      "responseElements": null,
      "requestID": "OP4OYXZ54HUPMIPGWH_EXAMPLE",
      "eventID": "0bca6660-e999-43a5-9008-EXAMPLE",
      "readOnly": false,
      "eventType": "AwsApiCall",
      "managementEvent": true,
      "recipientAccountId": "111122223333"
      "eventCategory": "Management"
      }
    }
```


用于亚马逊应用程序恢复控制器 (ARC) 区域转移的 Identity and Access Management

AWS Identity and Access Management (IAM) AWS 服务 可帮助管理员安全地控制对 AWS 资源的访问权限。IAM 管理员控制谁可以进行身份验证（登录）和授权（有权限）使用 ARC 资源。您可以使用 IAM AWS 服务 ，无需支付额外费用。

内容

- IAM 和可用区转移权限
- ARC 中基于身份的区域转移策略示例

区域偏移如何与 IAM 配合使用

在使用 IAM 管理对 Amazon 应用程序恢复控制器 (ARC) 中区域转移的访问权限之前，请先了解哪些可用于区域转移的 IAM 功能。

可在区域偏移中使用的 IAM 功能

IAM 特征	支持区域移动
基于身份的策略	是
基于资源的策略	否
策略操作	是
策略资源	是
策略条件键	是
ACLs	否
ABAC（策略中的标签）	部分
临时凭证	是
主体权限	是
服务角色	否
服务相关角色	是

要全面了解 AWS 服务如何与大多数 IAM 功能配合使用，请参阅 IAM 用户指南中的与 IAM 配合使用的 AWS 服务。

ARC 基于身份的策略

支持基于身份的策略：是

基于身份的策略是可附加到身份（如 IAM 用户、用户组或角色）的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的使用客户管理型策略定义自定义 IAM 权限。

通过使用 IAM 基于身份的策略，您可以指定允许或拒绝的操作和资源以及允许或拒绝操作的条件。您无法在基于身份的策略中指定主体，因为它适用于其附加的用户或角色。要了解可在 JSON 策略中使用的所有元素，请参阅《IAM 用户指南》中的 IAM JSON 策略元素引用。

要查看 ARC 基于身份的策略的示例，请参阅。Amazon 应用程序恢复控制器 (ARC) 中基于身份的策略示例

ARC 内部基于资源的政策

支持基于资源的策略：否

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资源的访问。

区域转移的政策行动

支持策略操作：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

JSON 策略的 `Action` 元素描述可用于在策略中允许或拒绝访问的操作。策略操作通常与关联的 AWS API 操作同名。有一些例外情况，例如没有匹配 API 操作的仅限权限 操作。还有一些操作需要在策略中执行多个操作。这些附加操作称为相关操作。

在策略中包含操作以授予执行关联操作的权限。

要查看用于区域转移的 ARC 操作列表，请参阅《服务授权参考》中的 Amazon Route 53 区域偏移定义的操作。

ARC 中用于区域转移的策略操作在操作前使用以下前缀：

```
arc-zonal-shift
```

要在单个语句中指定多项操作，请使用逗号将它们隔开。例如，以下内容：

```
"Action": [
```

```
      "arc-zonal-shift:action1",
      "arc-zonal-shift:action2"
         ]
```

您也可以使用通配符（*）指定多个操作。例如，要指定以单词 Describe 开头的所有操作，包括以下操作：

```
"Action": "arc-zonal-shift:Describe*"
```

要查看 ARC 基于身份的区域转移策略示例，请参阅。ARC 中基于身份的区域转移策略示例

区域转移的政策资源

支持策略资源：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

Resource JSON 策略元素指定要向其应用操作的一个或多个对象。语句必须包含 Resource 或 NotResource 元素。作为最佳实践，请使用其 Amazon 资源名称（ARN）指定资源。对于支持特定资源类型（称为资源级权限）的操作，您可以执行此操作。

对于不支持资源级权限的操作（如列出操作），请使用通配符（*）指示语句应用于所有资源。

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

要查看资源类型及其列表 ARNs，以及您可以使用每种资源的 ARN 指定的操作，请参阅《服务授权参考》中的以下主题：

- Amazon Route 53 定义的操作——区域移动

要查看可与条件键配合使用的操作和资源，请参阅《服务授权参考》中的以下主题：

- 由 Amazon Route 53 定义的条件键——区域移动

要查看 ARC 基于身份的区域转移策略示例，请参阅。ARC 中基于身份的区域转移策略示例

区域转移的策略条件密钥

支持特定于服务的策略条件键：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

在 Condition 元素（或 Condition 块）中，可以指定语句生效的条件。Condition 元素是可选的。您可以创建使用条件运算符（例如，等于或小于）的条件表达式，以使策略中的条件与请求中的值相匹配。

如果您在一个语句中指定多个 Condition 元素，或在单个 Condition 元素中指定多个键，则 AWS 使用逻辑 AND 运算评估它们。如果您为单个条件键指定多个值，则使用逻辑OR运算来 AWS 评估条件。在授予语句的权限之前必须满足所有的条件。

在指定条件时，您也可以使用占位符变量。例如，只有在使用 IAM 用户名标记 IAM 用户时，您才能为其授予访问资源的权限。有关更多信息，请参阅《IAM 用户指南》中的 IAM 策略元素：变量和标签。

AWS 支持全局条件密钥和特定于服务的条件密钥。要查看所有 AWS 全局条件键，请参阅 IAM 用户指南中的AWS 全局条件上下文密钥。

要查看区域移位条件键列表，请参阅《服务授权参考》中的以下主题：

- 由 Amazon Route 53 定义的条件键——区域移动

要查看可与条件键配合使用的操作和资源，请参阅《服务授权参考》中的以下主题：

- Amazon Route 53 定义的操作——区域移动
- 由 Amazon Route 53 定义的资源类型——区域偏移

要查看 ARC 基于身份的区域转移策略示例，请参阅。ARC 中基于身份的区域转移策略示例

ARC 中的访问控制列表 (ACLs)

支持 ACLs：否

访问控制列表 (ACLs) 控制哪些委托人（账户成员、用户或角色）有权访问资源。 ACLs 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

使用 ARC 实现基于属性的访问控制 (ABAC)

支持 ABAC（策略中的标签）：部分支持

基于属性的访问控制（ABAC）是一种授权策略，该策略基于属性来定义权限。在中 AWS，这些属性称为标签。您可以向 IAM 实体（用户或角色）和许多 AWS 资源附加标签。标记实体和资源是 ABAC 的第一步。然后设计 ABAC 策略，以在主体的标签与他们尝试访问的资源标签匹配时允许操作。

ABAC 在快速增长的环境中非常有用，并在策略管理变得繁琐的情况下可以提供帮助。

要基于标签控制访问，您需要使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 条件键在策略的条件元素中提供标签信息。

如果某个服务对于每种资源类型都支持所有这三个条件键，则对于该服务，该值为是。如果某个服务仅对于部分资源类型支持所有这三个条件键，则该值为部分。

有关 ABAC 的更多信息，请参阅《IAM 用户指南》中的使用 ABAC 授权定义权限。要查看设置 ABAC 步骤的教程，请参阅《IAM 用户指南》中的使用基于属性的访问权限控制（ABAC）。

ARC 包括对 ABAC 的以下部分支持：

- 对于在 ARC 中注册的用于区域偏移的托管资源，区域偏移支持 ABAC。有关网络负载均衡器和应用程序负载均衡器托管资源的 ABAC 的更多信息，请参阅《Elastic Load Balancing 用户指南》之 Elastic Load Balancing 中的 ABAC。

在 ARC 中使用临时证书

支持临时凭证：是

当你使用临时证书登录时，有些 AWS 服务 不起作用。有关更多信息，包括哪些 AWS 服务 适用于临时证书，请参阅 IAM 用户指南中的AWS 服务 与 IAM 配合使用的信息。

如果您使用除用户名和密码之外的任何方法登录，则 AWS Management Console 使用的是临时证书。例如，当您 AWS 使用公司的单点登录 (SSO) 链接进行访问时，该过程会自动创建临时证书。当您以用户身份登录控制台，然后切换角色时，您还会自动创建临时凭证。有关切换角色的更多信息，请参阅《IAM 用户指南》中的从用户切换到 IAM 角色（控制台）。

您可以使用 AWS CLI 或 AWS API 手动创建临时证书。然后，您可以使用这些临时证书进行访问 AWS。 AWS 建议您动态生成临时证书，而不是使用长期访问密钥。有关更多信息，请参阅 IAM 中的临时安全凭证。

ARC 的跨服务主体权限

支持转发访问会话（FAS）：是

当您使用 IAM 实体（用户或角色）在中执行操作时 AWS，您被视为委托人。策略向主体授予权限。使用某些服务时，您可能会执行一个操作，此操作然后在不同服务中触发另一个操作。在这种情况下，您必须具有执行这两个操作的权限。

要查看某项操作是否需要策略中的其他相关操作，请参阅《服务授权参考》中的以下主题：

- Amazon Route 53 可用区转移

ARC 的服务角色

支持服务角色：否

服务角色是由一项服务担任、代表您执行操作的 IAM 角色。IAM 管理员可以在 IAM 中创建、修改和删除服务角色。有关更多信息，请参阅《IAM 用户指南》中的创建向 AWS 服务委派权限的角色。

ARC 的服务相关角色

支持服务相关角色：是

服务相关角色是一种链接到的服务角色。 AWS 服务服务可以代入代表您执行操作的角色。服务相关角色出现在您的中 AWS 账户 ，并且归服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。

区域转移不使用与服务相关的角色。

IAM 和可用区转移权限

本节提供了有关权限如何使用亚马逊应用程序恢复控制器 (ARC) 中的区域转移功能的更多信息，特别是如果您使用其他 AWS 服务（例如 Elastic Load Balancing）中的该功能。要了解 ARC 功能如何与 IAM 和一般权限配合使用，请查看概述主题中的信息用于亚马逊应用程序恢复控制器 (ARC) 区域转移的 Identity and Access Management。

区域转移支持应用程序负载均衡器、网络负载均衡器、Amazon A EC2 uto Scaling 组和 Amazon EKS。您可以使用 IAM 条件键将 IAM 权限策略的范围限定为这些资源。以下是一个使用条件密钥的策略示例，其中包含多个不同类型的资源：

```
{
    "Condition": {
        "StringLike": {
            "arc-zonal-shift:ResourceIdentifier": [
                "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/net/
*",
```

```
                "arn:aws:elasticloadbalancing:us-east-1:123456789012:loadbalancer/app/
*",
                "arn:aws:eks:us-east-1:123456789012:cluster/*"
            ]
        }
    },
    "Action": [
        "arc-zonal-shift:StartZonalShift"
    ],
    "Resource": "*",
    "Effect": "Allow"
}
```

有关更多信息，请参阅 支持的资源。

除了 IAM 概述主题中概述的权限外，以下内容还适用于 IAM 和权限的区域转移：

- 确保您拥有在 ARC 中处理区域偏移所需的权限。有关更多信息，请参阅分区移位控制台访问权限和区域移位操作访问权限。
- 您无需通过 IAM 添加额外的 Elastic Load Balancing 权限，即可在 ARC 中对账户中的托管负载均衡器资源进行区域切换。
- 为 Elastic Load Balancing 提供完全访问权限的 AWS 托管策略包括使用区域转移的权限。如果您使用 AWS 托管策略获取 Elastic Load Balancing 访问权限，则无需在 IAM 中获得额外的区域转移权限即可启动负载均衡器的区域转移或在 Elastic Load Balancing 控制台中使用。有关更多信息，请参阅 Elastic Load Balancing 的AWS 托管策略。

ARC 中基于身份的区域转移策略示例

默认情况下，用户和角色无权创建或修改 ARC 资源。他们也无法使用 AWS Management Console、AWS Command Line Interface (AWS CLI) 或 AWS API 执行任务。要授予用户对所需资源执行操作的权限，IAM 管理员可以创建 IAM 策略。管理员随后可以向角色添加 IAM 策略，用户可以代入角色。

要了解如何使用这些示例 JSON 策略文档创建基于 IAM 身份的策略，请参阅《IAM 用户指南》中的创建 IAM 策略（控制台）。

有关 ARC 定义的操作和资源类型（包括每种资源类型的格式）的详细信息，请参阅《 ARNs 服务授权参考》中的 Amazon Application Recovery Controller (ARC) 的操作、资源和条件密钥。

主题

- 策略最佳实践

- 示例：区域移位控制台访问权限
- 示例：区域移动 API 操作

策略最佳实践

基于身份的策略决定了某人是否可以在您的账户中创建、访问或删除 ARC 资源。这些操作可能会使 AWS 账户产生成本。创建或编辑基于身份的策略时，请遵循以下指南和建议：

- 开始使用 AWS 托管策略并转向最低权限权限 — 要开始向用户和工作负载授予权限，请使用为许多常见用例授予权限的AWS 托管策略。它们在你的版本中可用 AWS 账户。我们建议您通过定义针对您的用例的 AWS 客户托管策略来进一步减少权限。有关更多信息，请参阅《IAM 用户指南》中的AWS 托管式策略或工作职能的AWS 托管式策略。
- 应用最低权限：在使用 IAM 策略设置权限时，请仅授予执行任务所需的权限。为此，您可以定义在特定条件下可以对特定资源执行的操作，也称为最低权限许可。有关使用 IAM 应用权限的更多信息，请参阅《IAM 用户指南》中的 IAM 中的策略和权限。
- 使用 IAM 策略中的条件进一步限制访问权限：您可以向策略添加条件来限制对操作和资源的访问。例如，您可以编写策略条件来指定必须使用 SSL 发送所有请求。如果服务操作是通过特定 AWS 服务的（例如）使用的，则也可以使用条件来授予对服务操作的访问权限 AWS CloudFormation。有关更多信息，请参阅《IAM 用户指南》中的 IAM JSON 策略元素：条件。
- 使用 IAM Access Analyzer 验证您的 IAM 策略，以确保权限的安全性和功能性 – IAM Access Analyzer 会验证新策略和现有策略，以确保策略符合 IAM 策略语言（JSON）和 IAM 最佳实践。IAM Access Analyzer 提供 100 多项策略检查和可操作的建议，以帮助您制定安全且功能性强的策略。有关更多信息，请参阅《IAM 用户指南》中的使用 IAM Access Analyzer 验证策略。
- 需要多重身份验证 (MFA)-如果 AWS 账户您的场景需要 IAM 用户或根用户，请启用 MFA 以提高安全性。若要在调用 API 操作时需要 MFA，请将 MFA 条件添加到您的策略中。有关更多信息，请参阅《IAM 用户指南》中的使用 MFA 保护 API 访问。

有关 IAM 中的最佳实操的更多信息，请参阅《IAM 用户指南》中的 IAM 中的安全最佳实践。

示例：区域移位控制台访问权限

要访问 Amazon 应用程序恢复控制器 (ARC) 控制台，您必须拥有一组最低权限。这些权限必须允许您列出和查看有关您的 ARC 资源的详细信息 AWS 账户。如果创建比必需的最低权限更为严格的基于身份的策略，对于附加了该策略的实体（用户或角色），控制台将无法按预期正常运行。

对于仅调用 AWS CLI 或 AWS API 的用户，您无需为其设置最低控制台权限。相反，只允许访问与其尝试执行的 API 操作相匹配的操作。

要向用户提供在中使用区域转移的完全访问权限 AWS Management Console，请向用户附加如下所示的策略：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                   "arc-zonal-shift:ListManagedResources",
                   "arc-zonal-shift:GetManagedResource",
                   "arc-zonal-shift:ListZonalShifts",
                   "arc-zonal-shift:StartZonalShift",
                   "arc-zonal-shift:UpdateZonalShift",
                   "arc-zonal-shift:CancelZonalShift"
             ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DescribeAvailabilityZones",
            "Resource": "*"
        }
    ]
}
```

示例：区域移动 API 操作

区域移动 API 会暂时将流量从可用区域移开，以恢复应用程序。

为确保用户可以使用区域移动 API 操作，请附加与用户需要使用的 API 操作相对应的策略，例如：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                   "arc-zonal-shift:ListManagedResources",
                   "arc-zonal-shift:GetManagedResource",
                   "arc-zonal-shift:ListZonalShifts",
                   "arc-zonal-shift:StartZonalShift",
                   "arc-zonal-shift:UpdateZonalShift",
```

```
                "arc-zonal-shift:CancelZonalShift"
            ],
            "Resource": "*"
        }
    ]
}
```

ARC 中的区域自动换档

使用区域自动切换，您可以授权 AWS 在活动期间代表您转移应用程序的资源流量从可用区 (AZ)，以帮助缩短恢复时间。 AWS 当内部遥测数据显示存在可能影响客户的可用区域受损时，会启动自动换档。 AWS 启动自动切换时，您为区域自动切换配置的资源的应用程序流量开始从可用区转移出去。

请注意，ARC 不会检查单个资源的运行状况。 AWS 当 AWS 遥测检测到存在可能影响客户的可用区域受损时，就会启动自动换档。在某些情况下，流量可能会转移到没有受到影响的资源上。

通过区域自动切换，您还可以代表您授权 AWS 将应用程序的资源流量从可用区转移出来，用于常规练习。可用区自动转移需要练习运行。ARC 在练习运行中开始的区域转移可帮助您确保在自动换档期间将流量从可用区转移出来对您的应用程序是安全的。通过启动可用区转移，将资源的流量从可用区转移出去，练习运行能够定期测试您的应用程序能否在没有一个可用区的情况下正常运行。每周进行一次练习，并提供结果（例如`SUCCEEDED`或`FAILED`），以帮助您了解应用程序是否按预期运行。

> **Important**
>
> 在配置练习运行或启用区域自动切换之前，我们强烈建议您在部署应用程序资源的区域的所有可用区中预先扩展应用程序资源容量。当自动转移或练习运行启动时，您不应依赖于按需扩展。可用区自动转移（包括练习运行）独立工作，且不会等待自动扩缩操作完成。依赖 auto Scaling 而不是预缩放可能会导致应用程序需要更长的时间才能恢复。
> 如果您使用自动扩缩来处理定期的流量周期，则强烈建议您配置自动扩缩的最小容量，以便在可用区丢失的情况下能够继续正常运行。

如果您计划启用区域自动移位或配置练习运行，请在预先扩展应用程序资源容量之后，测试您的应用程序是否可以在没有一个可用区的情况下正常运行。要进行此测试，请启动可用区转移，将资源的流量从可用区转移出去。

启用区域自动切换后，我们建议您通过启动和评估按需运行区域移位的实践来验证您的应用程序是否可以在流量从可用区转移出去的情况下继续正常运行。然后，ARC进行的常规练习可以帮助你持续确认自己有足够的容量进行自动换档。

为了确保您的区域偏移测试有效，请务必验证您离开的可用区的流量是否如预期的那样流失。例如，应用程序负载均衡器和网络负载均衡器都在 Amazon 中提供了每个可用区的指标 CloudWatch ，您可以使用这些指标来监控这一点。根据服务和客户端重复使用连接的时间长度，流量继续流向您已离开的可用区的持续时间可能会比您预期的要长。要了解更多信息，请参阅限制客户端与您的终端保持连接的时间。

您可以在 ARC 控制台中为支持的资源启用区域自动切换。或者，在 Amazon EC2 控制台中，您可以选择为特定的负载均衡器资源启用区域自动切换。要详细了解如何使用 Elastic Load Balancing 启用区域自动切换，请参阅 Elastic Lo ad Balancing 用户指南中的区域偏移。

自动转移和练习运行可用区转移是暂时的。通过自动切换，当受影响的可用区恢复时， AWS 会停止将资源流量从可用区转移出去。客户的应用程序流量会返回到区域中的所有可用区。在练习运行中，流量会从单个资源的可用区中转移出去约 30 分钟，然后再转移回区域中的所有可用区。

您可以将 Amazon EventBridge 通知配置为提醒您有关自动换档和练习跑的信息。有关更多信息，请参阅 在 Amazon 上使用区域自动换档 EventBridge。

可用区自动转移和练习运行的工作原理

Amazon Application Recovery Controller (ARC) 中的区域自动切换功能允许 AWS 您在 AWS 确定存在可能影响可用区客户的损害时，代表您将资源的流量从可用区转移出去。Zonal autoshift 专为在中的所有可用区中预先扩展的资源而设计 AWS 区域，这样应用程序就可以在失去一个可用区域的情况下正常运行。

使用区域自动切换，您需要配置练习跑，ARC 会定期将资源流量从一个可用区转移出去。ARC 为每个与之关联的练习运行配置的资源安排大约每周进行一次练习。每个资源的练习运行都是独立安排的。

对于每一次练习，ARC 都会记录结果。如果练习运行因阻止条件而中断，则练习运行结果不会标记为成功。有关练习运行结果的更多信息，请参阅练习运行结果。

您可以将 Amazon EventBridge 通知配置为向您发送有关自动换档和练习跑的信息。有关更多信息，请参阅 在 Amazon 上使用区域自动换档 EventBridge。

内容

- 区域偏移的优先级
- 停止资源的活动自动转移或练习运行
- 流量是如何转移出去的
- 练习运行警报
- 阻止日期和阻止时段 (UTC)

关于分区自动换档

区域自动切换是一种代表您 AWS 将应用程序资源流量从可用区转移出去的功能。 AWS 当内部遥测数据显示存在可能影响客户的可用区域受损时，会启动自动换档。内部遥测包含来自多个来源的指标，包括 AWS 网络、Amazon EC2 和 Elastic Load Balancing 服务。

您必须为支持的 AWS 资源手动启用区域自动切换。

当您在一个区域的多个（通常是三个） AZs 的负载均衡器上部署和运行 AWS 应用程序，并预先扩展以支持静态稳定性时， AWS 可以通过使用自动移位功能转移流量，从而快速恢复可用区中的客户应用程序。通过将资源流量转移到该 AZs 地区的其他地方， AWS 可以缩短由停电、可用区硬件或软件问题或其他损伤造成的潜在影响的持续时间和严重程度。

ARC 支持的资源提供了将指定可用区标记为不健康的集成，这会导致流量从受损的可用区转移出去。

为资源启用区域自动移位时，还必须为该资源配置练习运行。 AWS 大约每周进行一次练习，持续 30 分钟，以帮助您确保有足够的容量来运行您的应用程序，而无需该区域的可用区。

与可用区转移一样，在某些特定情况下，可用区自动转移不会将流量从可用区转移出去。例如，如果中的负载均衡器目标组 AZs 没有任何实例，或者所有实例都运行状况不佳，则负载均衡器处于失效打开状态，您无法移开其中一个实例。 AZs

要了解有关可用区自动转移的更多信息，请参阅 ARC 中的区域自动换档。

何时 AWS 启动和停止自动换档

当您为资源启用区域自动切换时，即表示您授权在事件发生期间 AWS 将应用程序的资源流量从可用区转移出去，以帮助缩短恢复时间。

为实现这一目标，zonal autoshift 使用 AWS 遥测技术尽早检测到存在可能影响客户的可用区损害。当 AWS 启动自动转移时，传输到已配置资源的流量会立即开始从可能会影响客户的受损可用区转移。

Zonal autoshift 是一项专为已为中所有可用区预先扩展其应用程序资源的客户设计的功能。 AWS 区域当自动转移或练习运行启动时，您不应依赖于按需扩展。

AWS 当它确定可用区已恢复时，将结束自动切换。

当 ARC 安排、开始和结束练习时

ARC 每周安排一次资源练习，持续大约 30 分钟。ARC 独立安排、开始和管理每种资源的练习跑。ARC 不会批量使用同一个账户中的资源进行练习。您也可以自己开始按需练习，以帮助验证您的设置对于区域自动换档赛事是否安全。

当练习运行在预期的持续时间内不间断进行时，它的结果会标记为 SUCCESSFUL。还有其它几种可能的结果：FAILED、INTERRUPTED 和 PENDING。结果值和描述包含在练习运行结果部分。

在某些情况下，ARC 会中断练习跑并结束练习。例如，如果在练习跑期间开始自动换档，ARC 会中断练习跑并结束练习。再举一个例子，假设资源对练习运行有不良影响，并导致您指定的用于监控练习运行的警报进入 ALARM 状态。在这种情况下，ARC 还会中断练习并结束练习。

此外，在某些情况下，ARC 不会为资源启动计划练习。

为了应对资源中断和阻塞的练习运行，ARC 会执行以下操作：

- 如果某项资源的练习在进行过程中中断，ARC 会认为每周的练习已经结束，并安排在下周为该资源进行一次新的练习。在这种情况下，每周练习的结果为 INTERRUPTED，而不是 FAILED。只有当监控练习运行的结果警报在练习运行期间进入 ALARM 状态时，练习运行结果才会设置为 FAILED。
- 如果在计划开始某项资源的练习时存在阻塞限制，则 ARC 不会开始练习跑。ARC 继续进行定期监控，以确定是否还有一个或多个阻塞限制。当没有任何阻塞限制时，ARC 会开始对资源进行练习。

以下是阻止 ARC 开始或继续资源练习运行的屏蔽约束示例：

- 当有 AWS Fault Injection Service 实验进行时，ARC 不会开始或继续练习。如果在 ARC 安排练习跑开始时某个 AWS FIS 赛事处于活动状态，则 ARC 不会开始练习跑。ARC 在整个练习跑中监视阻挡限制，包括 AWS FIS 赛事。如果 AWS FIS 活动在练习跑处于活动状态时开始，ARC 将结束练习跑，并且在资源下一次定期安排的练习跑之前不会尝试开始另一场练习。
- 如果某个地区有当前 AWS 赛事，ARC 不会开始为资源而开始练习，而是结束该区域的活跃练习。

当练习跑没有中断的情况下结束时，ARC会像往常一样在一周内安排下一次练习。如果由于阻塞限制（例如您指定的 AWS FIS 实验或被封锁的时间窗口）而没有开始练习，ARC 会继续尝试开始练习，直到练习跑可以开始。

练习跑的容量检查

当练习开始时，为了暂时将流量从可用区移开，ARC 会进行检查，以验证您在其他可用区域中是否有足够的容量来安全地将流量从可用区转移出去。如果没有足够的可用容量，则练习跑的交通转移不会开始，练习跑将结束。

此外，在 ARC 结束自动换档启动的流量转移之前，当区域自动换档完成时，ARC 会对负载均衡器资源进行容量检查。如果自动切换结束时容量检查失败，则流量不会转移回原来的可用区。

仅对负载均衡器和 Auto Scaling 组完成容量平衡检查。

对于负载均衡器资源，容量检查可验证与负载均衡器关联的健康主机是否分布在各个可用区中。具体而言，容量检查可确保注册资源的所有可用区中运行良好的主机数量保持平衡。对于容量检查，平衡意味着每个可用区的健康容量与其他区域相当，差异很小。

请注意，容量检查不适用于目标组为 Lambda 的负载均衡器，也不适用于应用程序负载均衡器，因为这些目标不是按区域配置的。

还完成了 Auto Scaling 群组的容量检查。对于 Auto Scaling 组，容量检查会验证 Auto Scaling 组的总健康区域容量（即所有可用区域中运行状况良好的主机总数）是否符合为该组设置的所需容量。

当容量检查失败时

当容量检查发现资源的可用容量不平衡时，练习运行的结果是`CAPACITY_CHECK_FAILED`。要详细了解容量检查失败的原因，请参阅的评论字段`ZonalShiftSummary`。要查找练习跑步区域偏移的评论栏，请执行以下操作：

1. 使用 AWS CLI，列出您在使用 ListZonalShiftsAPI 操作的练习运行中指定的资源的区域偏移。

   FOr 例如，要返回区域偏移，可以运行类似于以下内容的命令：

   ```
   aws arc-zonal-shift start-practice-run
       --resource-
   identifier="arn:aws:elasticloadbalancing:Region:111122223333:ExampleALB123456890"
   ```

2. 查看返回的`ZonalShiftSummary`对象数组，找出由于容量检查而失败的练习跑的区域偏移。
3. 要了解适用的区域偏移，请查看`Comment`字段中的信息。

练习跑和自动换档通知

通过设置 Amazon 通知，您可以选择收到有关练习跑和资源自动轮班的 EventBridge 通知。即使您尚未为任何资源启用区域自动切换（称为自动移位观察者 EventBridge 通知），也可以设置通知。通过 autoshift 观察者通知，当可用区可能受损时，ARC 启动的所有自动换档都会通知您。请注意，您必须在每个要接收通知 AWS 区域 的内容中配置此选项。

要查看启用自动移位旁听者通知的步骤，请参阅启用或禁用自动移位观察者通知。要了解有关通知选项以及如何在中配置通知选项的更多信息 EventBridge，请参阅在 Amazon 上使用区域自动换档 EventBridge。

区域偏移的优先级

在给定时间，应用的区域偏移不能超过一个。也就是说，只有一家诊所对资源进行区域移动、客户启动的区域移动、自动移位或 AWS FIS 实验。当第二次区域偏移开始时，ARC 会遵循优先顺序来确定哪种区域偏移类型对资源有效。

优先顺序的一般原则是，您作为客户开始的区域班次优先于其他班次类型。但是，请注意，当前 AWS 启动的练习会阻止您开始按需练习。

为了说明 ARC 中的优先级，以下是示例场景中优先级的工作原理：

应用了分区移位类型	区域偏移类型已启动	结果
AWS FIS 实验	跑步练习	练习跑将无法开始，因为 AWS FIS 实验优先。
AWS FIS 实验	手动分区切换	AWS FIS 实验将被取消，并将应用手动分区偏移。
AWS FIS 实验	可用区自动转移	AWS FIS 实验将被取消，并将应用区域自动移位。
AWS FIS 实验	AWS FIS 实验	启动的 AWS FIS 实验将无法启动，因为现有实验正在运行，触发了 AWS FIS 自动移位动作。

应用了分区移位类型	区域偏移类型已启动	结果
跑步练习	手动分区切换	练习赛将被取消，结果将设置为INTERRUPTED ，并将应用区域偏移。
跑步练习	AWS FIS 实验	练习跑将被取消，结果将设置为INTERRUPTED ， AWS FIS 实验将被应用。
跑步练习	可用区自动转移	练习赛将被取消，结果设置为INTERRUPTED ，区域自动移位将被应用。
手动分区切换	跑步练习	练习跑将无法开始。
手动分区切换	AWS FIS 实验	AWS FIS 实验将无法启动，如果实验已经在进行中，则实验将失败。
手动分区切换	可用区自动转移	区域自动移位将位于资源上，ACTIVE但不APPLIED在资源上。手动分区移位优先。
可用区自动转移	AWS FIS 实验	AWS FIS 实验将无法启动，或者如果正在进行则会失败。
可用区自动转移	手动分区切换	区域自动移位将位于资源上，ACTIVE但不APPLIED在资源上。手动分区移位优先。
可用区自动转移	跑步练习	练习跑将无法开始，因为区域自动换档优先。

当前对资源有效的流量转移已将应用的可用区转移状态设置为 APPLIED。任何时候只有一个转移设置为 APPLIED。其他正在进行的转变已准备就绪NOT_APPLIED，但仍保持不ACTIVE变。

停止资源的活动自动转移或练习运行

要停止正在进行的资源自动移位，必须取消区域移动。

该资源仍会按相同的计划进行定期练习运行。如果除了禁用自动转移之外您还想停止练习运行，则必须删除与该资源关联的练习运行配置。

删除练习运行配置后，将 AWS 停止执行每周将资源流量从可用区转移的练习运行。此外，由于区域自动移位需要练习跑，因此当您使用 ARC 控制台删除练习跑配置时，此操作还会禁用资源的区域自动移位。但是，请注意，如果您使用可用区自动转移 API 来删除练习运行，则必须先禁用针对资源的可用区自动转移。

有关更多信息，请参阅取消区域自动换档和 启用和使用分区自动换档。

流量是如何转移出去的

对于自动换档和练习跑区域换档，使用与ARC用于客户启动的区域换档相同的机制将流量从可用区转移出去。不健康的运行状况检查会导致 Amazon Route 53 从 DNS 中撤回该资源的相应的 IP 地址，从而将流量从可用区域重定向。现在，新连接将 AWS 区域 改为路由到中的其他可用区。

使用自动换档时，当可用区恢复并 AWS 决定结束自动换档时，ARC 会撤消运行状况检查流程，请求恢复 Route 53 的运行状况检查。然后，恢复原来的区域 IP 地址，如果运行状况检查继续良好，则可用区将再次包含在应用程序的路由中。

务必注意，自动转移并非基于监控负载均衡器或应用程序底层运行状况的运行状况检查。ARC 使用运行状况检查将流量从可用区域移开，方法是请求将运行状况检查设置为不健康，然后在结束自动移位或区域转移时将运行状况检查恢复为正常。

练习运行警报

在分区自动换档中，您可以为练习跑指定两个 CloudWatch 警报。第一个警报结果警报是必需的。您应该配置结果警报，以便在每次为期 30 分钟的练习运行期间，在将流量从可用区转移出去时监控应用程序的运行状况。

为了使练习生效，请指定一个警报作为结果警报，该 CloudWatch 警报用于监控资源或应用程序的指标，当您的应用程序因失去一个可用区而受到不利影响时，这些指标会以ALARM状态做出响应。有关更多信息，请参阅 配置区域自动移位时的最佳实践中的为练习运行指定的警报部分。

结果警报还提供了 ARC 为每次练习跑报告的练习跑结果的信息。如果警报进入 ALARM 状态，则练习运行结束，练习运行结果返回为 FAILED。如果练习运行完成了 30 分钟的计划测试期，并且结果警报

未进入 ALARM 状态，则结果将返回为 SUCCEEDED。练习运行结果部分提供了所有结果值的列表及其描述。

您也可以选择指定第二个警报，即阻止警报。阻止警报将在处于 ALARM 状态时阻止练习运行启动或继续。当此警报处于 ALARM 状态时，它会阻止练习运行流量转移启动，并停止任何正在进行的练习运行。

例如，在具有多个微服务的大型架构中，当一个微服务遇到问题时，您通常希望停止应用程序环境中的所有其它更改，其中包括阻止练习运行。

阻止日期和阻止时段 (UTC)

你可以选择在特定的日历日期或特定的时间窗口（即天和时间），以 UTC 为单位阻止练习。

例如，如果您计划于 2024 年 5 月 1 日进行应用程序更新，并且您不希望练习运行在此时转移流量，则可以将阻止日期设置为 2024-05-01。

或者，假设您每周三天运行业务报告摘要。对于这种情况，您可以将采用 UTC 时间的以下重复日期和时间设置为阻止时段，例如：MON-20:30-21:30 WED-20:30-21:30 FRI-20:30-21:30。

AWS 区域 区域自动换档的可用性

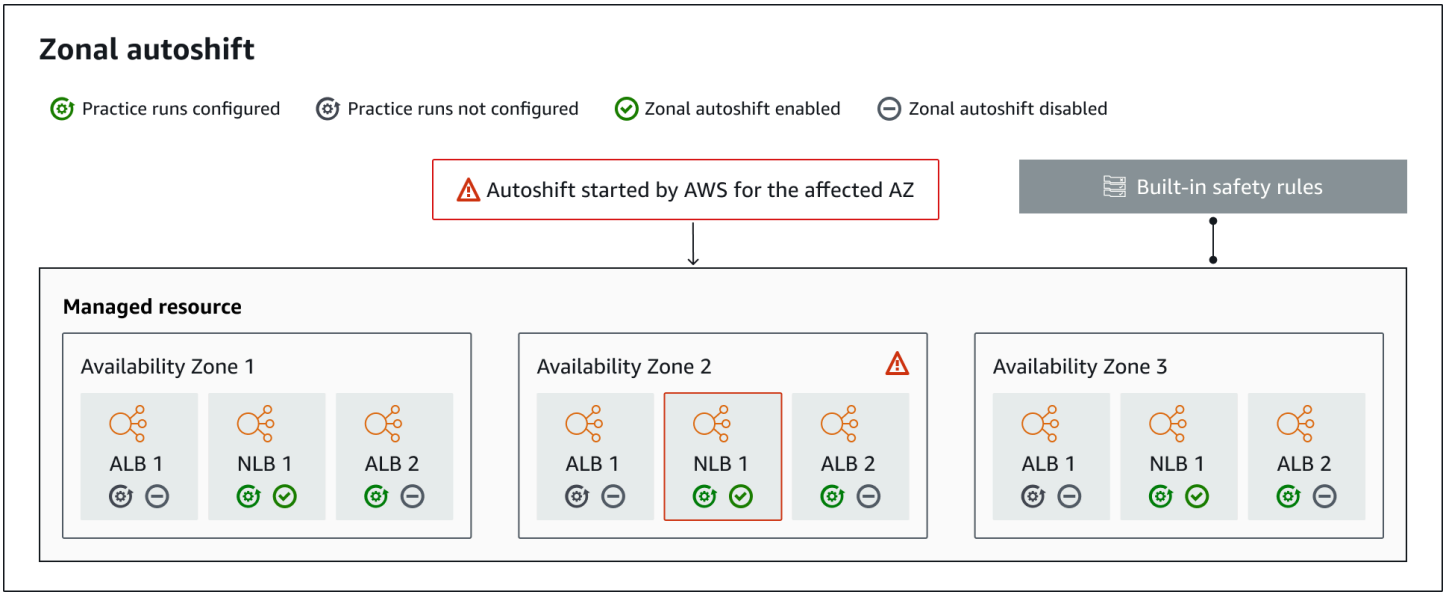
目前，商业 AWS 区域版以及中国区域（即中国（北京）区域和中国（宁夏）区域均提供区域移位和区域自动换档。

使用 Amazon 应用程序恢复控制器 (ARC) 的资源可能包括其他注意事项。有关更多信息，请参阅 支持的资源。

有关区域列表以及有关 ARC 区域支持和服务终端节点的详细信息，请参阅 Amazon Web Services 通用参考中的亚马逊应用程序恢复控制器 (ARC) 终端节点和配额。

可用区自动转移组件

下图说明了自动切换将流量从可用区移开的示例。 AWS 当内部遥测数据显示存在可能影响客户的可用区域受损时，会启动自动换档。



以下是 ARC 中分区自动移位功能的组成部分。

可用区自动转移

可用区自动转移无需您执行任何操作即可将资源的流量转移出去。区域自动换档是 ARC 中的一项功能，当内部遥测显示存在可能影响客户的可用区受损时，它会 AWS 启动自动换档。请注意，在某些情况下，没有受到影响的资源可能会被转移出去。

练习运行

为资源启用区域自动移位时，还必须为该资源配置区域自动移位练习运行。 AWS 大约每周进行一次练习，持续大约 30 分钟。您也可以按需安排练习。

练习运行可确保您的应用程序可以在丢失一个可用区的情况下正常运行。在练习运行中，通过区域 AWS 转移将资源的流量从一个可用区转移出去，然后在练习运行结束时将流量转移回去。

练习运行配置

练习运行配置定义了封锁的日期和窗口（如果有），以及您在区域自动移位中为资源的 AWS 练习运行指定的 CloudWatch警报。您可以随时编辑练习跑配置，添加或更改封锁日期或窗口，或者更新练习跑步的警报。

要启用区域自动移位，必须为资源准备好练习运行配置。您也可以删除练习跑。要删除资源的练习运行配置，必须禁用可用区自动转移。

练习运行警报

配置练习跑时，您可以根据资源和应用程序要求指定在中 CloudWatch创建的 CloudWatch 警报。如果您的应用程序受到练习运行的不利影响，则您指定的警报可以阻止练习运行启动，或可以停止正在进行的练习运行。

如果您指定的警报进入ALARM状态，ARC 将结束练习运行的区域移动，这样资源的流量就不会再从可用区域转移出去。

您可以为练习运行指定两种类型的警报：一种是结果警报，用于在练习运行期间监控资源和应用程序的运行状况；另一种是阻止警报，您可以将其配置为防止练习运行启动或停止正在进行的练习运行。结果警报是必需的；阻止警报是可选的。

练习运行结果

ARC 报告每次练习的结果。以下是可能的练习运行结果：

- 待处理：练习运行的可用区转移处于活动状态（正在进行中）。目前还没有结果可以返回。
- 已成功：在练习运行期间，结果警报未进入 ALARM 状态，练习运行完成了整整 30 分钟的测试周期。
- 已中断：练习运行结束的原因并非结果警报进入 ALARM 状态。练习运行中断的原因可能有多种。例如，由于为练习运行指定的阻止警报进入 ALARM 状态而结束的练习运行的结果为 INTERRUPTED。有关出现 INTERRUPTED 结果的原因的详细信息，请参阅练习运行结果。
- 已失败：在练习运行期间，结果警报进入了 ALARM 状态。
- C@@ APACITY_CHECK_FAILED：检查负载平衡和 Auto Scaling 组资源的可用区容量平衡失败。

内置安全规则

ARC 中内置的安全规则可防止资源同时发生多个流量变动。也就是说，只有一次由客户发起的区域转移、实践运行的区域转移（由客户 AWS 或由客户发起）或资源的自动切换可以主动将流量从可用区转移出去。例如，如果您对某个资源启动可用区转移，而该资源当前正在通过自动转移而转移出去，则优先进行可用区转移。有关更多信息，请参阅区域偏移的优先级。

资源标识符

要为其启用区域自动移位的资源的标识符，即该资源的亚马逊资源名称 (ARN)。您只能为账户中位于 ARC 支持的 AWS 服务中的资源启用区域自动切换。

托管资源

应用程序负载均衡器会自动向 ARC 注册资源，以实现区域自动切换。您必须手动选择其他资源才能进行区域自动切换。

资源名称

ARC 中托管资源的名称。

已应用状态

已应用状态指示资源的流量转移是否有效。配置可用区自动转移时，一个资源可以有多个活动的流量转移，即练习运行可用区转移、客户发起的可用区转移或自动转移。但是，只应用一个资源，也就是说，一次只能对资源生效。状态为 APPLIED 的转移可确定资源的应用程序流量已转移出去的可用区，以及该流量转移何时结束。

班次类型

定义分区偏移类型。区域偏移可以有以下类型之一：

- Zonal_shift
- ZONAL_AutoShift
- 练习跑
- FIS_实验

分区自动换档的数据和控制平面

在规划故障转移和灾难恢复时，请考虑故障转移机制的弹性。我们建议您确保在故障转移期间所依赖的机制具有高可用性，以便在灾难情况下可以根据需要使用它们。通常，应尽可能为机制使用数据平面函数，以获得最大的可靠性和容错性。考虑到这一点，请务必了解服务的功能如何在控制面板和数据面板之间划分，以及何时可以依赖服务的数据面板可预期的极高可靠性。

一般而言，控制面板允许您执行基本的管理功能，例如在服务中创建、更新和删除资源。数据面板提供服务的核心功能。

有关数据平面、控制平面以及如何 AWS 构建服务以满足高可用性目标的更多信息，请参阅 Amazon Builders Library 中的 “使用可用区的静态稳定性” 论文。

ARC 中区域自动换档的定价

对于区域自动切换，在 AWS 确定存在可能对客户应用程序产生不利影响的潜在问题时，代表您将流量从可用区域 AWS 转移出受支持资源。启用可用区自动转移不收取任何额外费用。

有关 ARC 的详细定价信息和定价示例，请参阅 ARC 定价。

配置区域自动移位时的最佳实践

在 Amazon 应用程序恢复控制器 (ARC) 中启用区域自动切换时，请注意以下最佳做法和注意事项。

区域自动换档包括两种类型的交通换档：自动换档和练习跑分区换档。

- 借 AWS 助自动切换，可在活动期间代表您转移可用区的应用程序资源流量，从而缩短恢复时间。
- 在练习跑中，ARC 代表你开始区域移动，或者你开始区域移位练习。 AWS 练习运行区域转移将资源从可用区域转移出可用区，然后按每周的节奏再次转移回来。练习运行可帮助您确保已为区域中的可用区纵向扩展了足够的容量，以便您的应用程序能够容忍丢失一个可用区。

对于自动换档和练习跑，需要记住几个最佳做法和注意事项。在启用可用区自动转移或为资源配置练习运行之前，请先阅读以下主题。

主题

限制客户端与您的终端保持连接的时间

当 Amazon Application Recovery Controller (ARC) 将流量从受损中转移出去时，例如使用区域转移或区域自动切换，ARC 用来转移应用程序流量的机制是 DNS 更新。DNS 更新会导致所有新连接被定向到远离受损位置。但是，在客户端重新连接之前，具有已打开连接的客户端可能会继续向受损位置发出请求。为确保快速恢复，我们建议您限制客户端与您的终端保持连接的时间。

如果您使用 Application Load Balancer，则可以使用该`keepalive`选项来配置连接的持续时间。我们建议您降低该`keepalive`值，使其与应用程序的恢复时间目标保持一致，例如 300 秒。在选择`keepalive`时间时，请考虑此值是在更频繁地重新连接（这可能会影响延迟）和更快地将所有客户端从受损的可用区或区域移出受损的可用区或区域之间进行权衡。

有关为 Application Load Balancer 设置`keepalive`选项的更多信息，请参阅《Application Load Balancer 用户指南》中的 HTTP 客户端保持连接时长。

预先扩展您的资源容量并测试流量的转移

当 AWS 将流量从一个可用区转移出来进行区域转移或自动切换时，重要的是剩余的可用区能够满足更高的资源请求速率。这种模式称为静态稳定性。有关更多信息，请参阅 Amazon Builders' Library 中的“使用可用区的静态稳定性”白皮书。

例如，如果您的应用程序需要 30 个实例来为其客户端提供服务，则应跨三个可用区预置 15 个实例，总共预置 45 个实例。通过这样做，当流量从一个可用区域 AWS 转移出去时（使用自动换档或在练习运行期间），仍然AWS 可以跨两个可用区为应用程序的客户端提供剩余的 30 个实例。

ARC 中的区域自动切换功能可帮助您快速从可用区 AWS 的事件中恢复，因为您的应用程序的资源已预先扩展，可以在失去一个可用区的情况下正常运行。在为资源启用可用区自动转移之前，请在 AWS 区域的所有已配置可用区中扩展您的资源容量。然后，对资源启动可用区转移，以测试当流量从可用区转移出去时，您的应用程序是否仍能正常运行。

使用可用区转移进行测试后，启用可用区自动转移并为应用程序资源配置练习运行。运行您自己的按需练习，以帮助确保您的配置得到正确扩展。使用区域自动转移进行定期练习运行可帮助您持续确保容量仍得到适当扩展。由于跨可用区有足够的容量，您的应用程序可以在自动转移期间不间断地继续为客户端提供服务。

有关为资源启动可用区转移的更多信息，请参阅ARC 中的区域偏移。

注意资源类型和限制

可用区自动转移支持将由可用区转移支持的所有资源的流量移出可用区。在一些特定的资源场景中，可用区自动转移不会将流量从可用区转移出来进行自动转移。

例如，如果可用区中的负载均衡器目标组没有任何实例，或者所有实例都运行状况不佳，则负载均衡器进入打开失败状态。如果在这种情况下为负载均衡器 AWS 启动自动切换，则自动切换不会更改负载均衡器使用的可用区，因为负载均衡器已经处于失效打开状态。这是预料之中的行为。AWS 区域 如果所有可用区都无法打开（不正常），Autoshift 不会导致一个可用区运行状况不佳，也不会将流量转移到其他可用区。

要查看有关受支持资源的详细信息（包括所有需要注意的要求和例外情况），请参阅支持的资源。

为练习跑指定警报

您至少要为使用区域自动移位的练习跑配置一个警报（结果警报）。或者，您也可以配置第二个警报（阻塞警报）。

在考虑为资源练习跑配置的 CloudWatch 警报时，请记住以下几点：

- 对于必需的结果警报，我们建议您将 CloudWatch 警报配置为在资源或应用程序的指标表明将流量从可用区转移出去会对性能产生不利影响时进入ALARM状态。例如，您可以确定资源的请求速率阈值，然后将警报配置为在超出该阈值时进入 ALARM 状态。您负责配置适当的警报，从而使 AWS 结束练习运行并返回 FAILED 结果。
- 我们建议您遵循架构AWS 完善的框架，该框架建议您将关键性能指标 (KPIs) 作为 CloudWatch 警报来实现。如果您这样做，则可以使用这些警报来创建复合警报以用作安全触发器，以防止在练习运行可能导致应用程序未达到 KPI 要求的情况下启动练习运行。当警报不再处于ALARM状态时，ARC 会在下次为资源安排练习运行时开始练习。
- 对于练习跑阻塞警报，如果您选择对其进行配置，则可以选择跟踪用于表示您不想开始 AWS 练习跑的特定指标。
- 对于练习运行警报，您可以为每个警报指定亚马逊资源名称 (ARN)，您必须先在 Amazon 中配置该名称。 CloudWatch您指定的 CloudWatch 警报可以是复合警报，这样您就可以为应用程序和资源添加多个指标和检查，从而触发警报进入ALARM状态。有关更多信息，请参阅 Amazon CloudWatch 用户指南中的合并警报。
- 确保您为练习跑指定的 CloudWatch 警报与您为其配置练习跑的资源位于同一区域。

评估练习跑的结果

ARC 报告每次练习的结果。练习跑后，评估结果，并确定是否需要采取行动。例如，您可能需要扩展容量或调整警报的配置。

以下是可能的练习运行结果：

- 已成功：在练习运行期间，结果警报未进入 ALARM 状态，练习运行完成了整整 30 分钟的测试周期。
- 已失败：在练习运行期间，结果警报进入了 ALARM 状态。
- 已中断：练习运行结束的原因并非结果警报进入 ALARM 状态。练习运行可能因各种原因而中断，包括以下几个原因：
  - 练习跑之所以结束，是因为在该地区 AWS 开始了自动换档 AWS 区域 或者该地区出现了警报情况。
  - 练习运行之所以结束，是因为已删除资源的练习运行配置。
  - 练习运行之所以结束，是因为已为可用区中的资源启动客户发起的可用区转移，而练习运行可用区转移已将流量从可用区中转移出去。
  - 练习跑已结束，因为无法再访问为练习跑配置指定的 CloudWatch 警报。
  - 练习运行之所以结束，是由于为练习运行指定的阻止警报进入 ALARM 状态。
  - 练习运行因未知原因而结束。

  - 练习赛结束了，因为启动了具有优先级的区域自动移位。有关区域偏移，请参见优先级。
- C@@ APACITY_CHECK_FAILED：检查负载平衡和 Auto Scaling 组资源的可用区容量平衡失败。
- 待处理：练习运行处于活动状态（正在进行中）。目前还没有结果可以返回。

可用区自动转移 API 操作

下表列出了可用于区域自动移位的 ARC API 操作。有关使用区域自动移位 API 操作的示例 AWS CLI，请参阅。

有关如何在 AWS Command Line Interface中使用常见可用区自动转移 API 操作的示例，请参阅使用 AWS CLI 带区域自动换档的示例。

操作	使用 ARC 控制台	使用 ARC API
创建练习运行配置	请参阅 启用或禁用可用区自动转移。	请参阅 CreatePracticeRunConfiguration
删除练习运行配置	请参阅 配置、编辑或删除练习运行配置。	请参阅 DeletePracticeRunConfiguration
列出自动转移	请参阅 ARC 中的区域自动换档。	请参阅 ListAutoshifts
列出要进行可用区自动转移的资源	请参阅 支持的资源。	请参阅 ListManagedResources
获取要进行可用区自动转移的资源	请参阅 支持的资源。	请参阅 GetManagedResource
编辑练习运行配置	请参阅 配置、编辑或删除练习运行配置。	请参阅 UpdatePracticeRunConfiguration
启用或禁用可用区自动转移	请参阅 启用或禁用可用区自动转移。	请参阅 UpdateZonalAutoshiftConfiguration
启用或禁用自动移位观察者通知	请参阅 启用和使用分区自动换档。	请参阅 UpdateAutoshiftObserverNotificationStatus

操作	使用 ARC 控制台	使用 ARC API
开始练习跑	请参阅 开始练习跑分区移动。	请参阅 StartPracticeRun
取消练习	请参阅 取消练习运行可用区转移。	请参阅 CancelPracticeRun

使用 AWS CLI 带区域自动换档的示例

本节介绍使用区域自动移位的简单应用示例，使用使用 API 操作在 AWS Command Line Interface Amazon 应用程序恢复控制器 (ARC) 中使用区域自动移位功能。这些示例旨在帮助您基本了解如何使用 CLI 使用区域自动移位。

区域自动换档是 ARC 中的一项功能。使用 zonal autoshift，您可以授权 AWS 在活动期间代表您转移可用区域中支持的应用程序资源流量，以帮助缩短恢复时间。有关可用于 zonal autoShift 的资源的更多信息，请参阅。支持的资源

区域自动换档包括练习跑，它还可以将流量从可用区域转移出去，以帮助验证自动换档对您的应用程序是否安全。

有关可用区自动转移 API 操作的列表和指向更多信息的链接，请参阅可用区自动转移 API 操作。有关使用的更多信息 AWS CLI，请参阅AWS CLI 命令参考。

内容

- 创建练习运行配置
- 启用或禁用自动转移
- 开始按需练习
- 取消正在进行的练习运行
- 取消正在进行的自动转移
- 编辑练习运行配置
- 删除练习运行配置

创建练习运行配置

在能够为资源启用可用区自动转移之前，必须为该资源创建练习运行配置，以便为所需的练习运行选择选项。您可以通过使用 `create-practice-run-configuration` 命令，借助 CLI 为资源创建练习运行配置。

在为资源创建练习运行配置时，请注意以下几点：

- 此时，唯一受支持的警报类型为 CLOUDWATCH。
- 您必须使用与资源部署相同的 AWS 区域 警报。
- 必须指定结果警报。可以选择指定阻止警报。
- 可选择指定阻止日期或阻止时段。

您可以通过使用 `create-practice-run-configuration` 命令，借助 CLI 创建练习运行配置。

例如，要为资源创建练习运行配置，可使用如下命令：

```
aws arc-zonal-shift create-practice-run-configuration \
      --resource-
identifier="arn:aws:elasticloadbalancing:Region:111122223333:ExampleALB123456890" \
      --outcome-alarms
 type=CLOUDWATCH,alarmIdentifier=arn:aws:cloudwatch:Region:111122223333:alarm:Region-
MyAppHealthAlarm \
      --blocking-alarms
 type=CLOUDWATCH,alarmIdentifier=arn:aws:cloudwatch:Region:111122223333:alarm:Region-
BlockWhenALARM \
      --blocked-dates 2023-12-01 --blocked-windows Mon:10:00-Mon:10:30
```

```
{
   "arn": "arn:aws:elasticloadbalancing:us-west-2:111122223333:ExampleALB123456890",
   "name": "zonal-shift-elb"
   "zonalAutoshiftStatus": "DISABLED",
   "practiceRunConfiguration": {
       "blockingAlarms": [
                  {
              "type": "CLOUDWATCH",
              "alarmIdentifier": "arn:aws:cloudwatch:us-west-2:111122223333:alarm:us-
west-2-BlockWhenALARM"
           }
       ]
```

```
        "outcomeAlarms": [
            {
                "type": "CLOUDWATCH",
                "alarmIdentifier": "arn:aws:cloudwatch:us-west-2:111122223333:alarm:us-
west-2-MyAppHealthAlarm"
            }
        ],
        "blockedWindows": [
            "Mon:10:00-Mon:10:30"
        ],
        "blockedDates": [
            "2023-12-01"
        ]
}
```

启用或禁用自动转移

您可以通过使用 CLI 更新可用区自动转移状态来对资源启用或禁用自动转移。要更改可用区自动转移状态，请使用 update-zonal-autoshift-configuration 命令。

例如，要对资源启用自动转移，请使用如下命令：

```
aws arc-zonal-shift update-zonal-autoshift-configuration \
      --resource-
identifier="arn:aws:elasticloadbalancing:Region:111122223333:ExampleALB123456890" \
      --zonal-autoshift-status="ENABLED"
```

```
{
   "resourceIdentifier": "arn:aws:elasticloadbalancing:us-
west-2:111122223333:ExampleALB123456890",
   "zonalAutoshiftStatus": "ENABLED"
}
```

开始按需练习

您可以使用start-practice-run命令开始按需练习，使用 CLI 进行区域移动。

例如，要开始对资源进行练习，请使用如下命令：

```
aws arc-zonal-shift start-practice-run
    --resource-
identifier="arn:aws:elasticloadbalancing:Region:111122223333:ExampleALB123456890" \
```

```
    "awayFrom": "usw2-az1",
```

```
{
    "awayFrom": "usw2-az1",
    "comment": "Practice run started. Shifting traffic away from Availability Zone
 usw2-az1.",
}
```

取消正在进行的练习运行

您可以使用cancel-practice-run命令取消正在进行的使用 CLI 的练习。

例如，要对资源取消练习运行，请使用如下命令：

```
aws arc-zonal-shift cancel-practice-run \
   --zonal-shift-id="="arn:aws:testservice::111122223333:ExampleALB123456890"
```

```
{
    "zonalShiftId": "2222222-3333-444-1111",
    "resourceIdentifier": "arn:aws:testservice::111122223333:ExampleALB123456890",
    "awayFrom": "usw2-az1",
    "expiryTime": 2024-11-15T10:35:42+00:00,
    "startTime": 2024-11-15T09:35:42+00:00,
    "status": "CANCELED",
    "comment": "Practice run canceled"
}
```

取消正在进行的自动转移

您可以使用 CLI 取消正在进行的自动移位，方法是取消资源的区域自动移位。要取消区域自动切换，请使用。cancel-zonal-shift command

```
aws arc-zonal-shift cancel-zonal-shift --zonal-shift-id
 9ac9ec1e-1df1-0755-3dc5-8cf573cd9c38
```

```
{
    "awayFrom": "usw2-az1",
    "comment": "Zonal autoshift started. Shifting traffic away from Availability Zone
 usw2-az1.",
```

```
    "expiryTime": "2024-12-17T22:29:38-08:00",
    "resourceIdentifier": "arn:aws:elasticloadbalancing:us-
east-1:111122223333:loadbalancer/app/Testing/5a19403ecd42dc05",
    "startTime": "2024-12-17T21:27:26-08:00",
    "status": "CANCELED",
    "zonalShiftId": "9ac9ec1e-1df1-0755-3dc5-8cf573cd9c38"
}
```

编辑练习运行配置

您可以使用 CLI 编辑资源的练习跑配置，以更新不同的配置选项，例如更改练习跑的警报，或者在 ARC 无法开始练习跑步时更新被屏蔽的日期或封锁的窗口。要编辑练习运行配置，请使用 update-practice-run-configuration 命令。

在为资源编辑练习运行配置时，请注意以下几点：

- 此时，唯一受支持的警报类型为 CLOUDWATCH。
- 您必须使用与资源部署相同的 AWS 区域 警报。
- 必须指定结果警报。可以选择指定阻止警报。
- 可选择指定阻止日期或阻止时段。
- 您指定的阻止日期或阻止时段将替换任何现有值。

例如，要编辑资源的练习运行配置以指定新的阻止日期，请使用如下命令：

```
aws arc-zonal-shift update-practice-run-configuration \
      --resource-
identifier="arn:aws:elasticloadbalancing:Region:111122223333:ExampleALB123456890" \
      --blocked-dates 2024-03-01
```

```
{
   "arn": "arn:aws:elasticloadbalancing:us-west-2:111122223333:ExampleALB123456890",
   "name": "zonal-shift-elb"
   "zonalAutoshiftStatus": "DISABLED",
   "practiceRunConfiguration": {
       "blockingAlarms": [
                   {
              "type": "CLOUDWATCH",
              "alarmIdentifier": "arn:aws:cloudwatch:us-west-2:111122223333:alarm:us-
west-2-BlockWhenALARM"
           }
```

```
        ]
        "outcomeAlarms": [
            {
                "type": "CLOUDWATCH",
                "alarmIdentifier": "arn:aws:cloudwatch:us-west-2:111122223333:alarm:us-
west-2-MyAppHealthAlarm"
            }
        ],
        "blockedWindows": [
            "Mon:10:00-Mon:10:30"
        ],
        "blockedDates": [
            "2024-03-01"
        ]
}
```

删除练习运行配置

您可以删除资源的练习运行配置，但必须先对该资源禁用可用区自动转移。资源需要具有练习运行配置，才能启用可用区自动转移。定期练习运行有助于您确保应用程序可以在没有一个可用区的情况下正常运行。

要使用 CLI 删除练习运行配置，请先使用 update-zonal-autoshift 命令禁用可用区自动转移（如果需要）。然后，可使用 delete-practice-run-configuration 命令删除练习运行配置。

首先，使用如下命令对资源禁用可用区自动转移：

```
aws arc-zonal-shift update-zonal-autoshift-configuration \
      --resource-
identifier="arn:aws:elasticloadbalancing:Region:111122223333:ExampleALB123456890" \
      --zonal-autoshift-status="DISABLED"
```

```
{
   "resourceIdentifier": "arn:aws:elasticloadbalancing:us-
west-2:111122223333:ExampleALB123456890",
   "zonalAutoshiftStatus": "DISABLED"
}
```

然后，使用如下命令删除练习运行配置：

```
aws arc-zonal-shift delete-practice-run-configuration \
```

```
      --resource-
identifier="arn:aws:elasticloadbalancing:Region:111122223333:ExampleALB123456890"
```

```
{
   "arn": "arn:aws:elasticloadbalancing:us-west-2:111122223333:ExampleALB123456890",
   "name": "TestResource",
   "zonalAutoshiftStatus": "DISABLED"
}
```

启用和使用分区自动换档

本节提供在 Amazon 应用程序恢复控制器 (ARC) 中使用区域自动移位的程序。启用区域自动换档后，您可以更改练习跑配置、开始按需练习、取消正在进行的轮班（包括练习跑）或启用自动换档观察者通知。

启用或禁用可用区自动转移

此处的步骤说明了如何在 Amazon 应用程序恢复控制器 (ARC) 控制台上启用或禁用区域自动切换。要以编程方式使用可用区转移，请参阅可用区转移和可用区自动转移 API 参考指南。

启用区域自动切换后，您授权 AWS 在活动期间代表您转移可用区的应用程序资源流量，以帮助缩短恢复时间。

启用或禁用可用区自动转移

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 在多可用区下，选择可用区自动转移。
3. 在资源可用区自动转移配置下，选择资源。
4. 在“操作”菜单中，选择“启用区域自动切换”，然后按照步骤完成更新。

如果资源没有练习运行配置，则启用可用区自动转移不可用。要配置练习运行配置并启用可用区自动转移，请选择配置可用区自动转移。

内容

- 配置、编辑或删除练习运行配置
- 取消区域自动换档
- 开始练习跑分区移动

- 取消练习运行可用区转移
- 启用或禁用自动移位观察者通知

配置、编辑或删除练习运行配置

本节中的步骤说明了如何在 Amazon 应用程序恢复控制器 (ARC) 控制台上编辑或删除练习运行配置。要以编程方式使用可用区转移（包括对练习运行配置进行更改），请参阅可用区转移和可用区自动转移 API 参考指南。

如果您在控制台中删除练习运行配置，则会禁用可用区自动转移。在可以通过 API 操作删除练习运行配置之前，必须禁用可用区自动转移。您可以在不启用可用区自动转移的情况下配置练习运行。但是，要为资源启用可用区自动转移，您需要为该资源配置练习运行。

配置练习运行

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 在多可用区下，选择可用区自动转移。
3. 选择配置可用区自动转移。
4. 选择要为可用区自动转移配置的资源。
5. 如果您不想 AWS 在发生事件时为资源启动自动移位，请选择禁用区域自动移位。 AWS 您可以选择继续使用向导来配置练习运行配置，而不启用自动转移。
6. 为资源选择练习运行选项。对于警报，您可以执行以下操作：

   - （必需）指定结果警报以监控该资源的练习运行。
   - （可选）为该资源的练习运行指定阻止警报。

   有关更多信息，请参阅 配置区域自动移位时的最佳实践中的为练习运行指定的警报部分。
7. 可选择指定阻止日期和阻止时段。选择日期或窗口（日期和时间）以阻止 ARC 开始为此资源进行练习。所有日期和时间均采用 UTC 时间。
8. 选中相应复选框，确认您已阅读确认说明。
9. 选择创建。

编辑练习运行配置

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。

2. 在多可用区下，选择可用区自动转移。
3. 在资源可用区自动转移配置下，选择资源。
4. 在操作菜单中，选择编辑练习运行配置。
5. 对练习运行配置进行更改，以执行以下一项或多项操作：
   - 对于警报，您可以执行以下操作：
     - 对于阻止警报，您可以添加警报、删除警报或指定其它阻止警报。
     - 对于监控练习跑的结果警报，您可以指定不同的 CloudWatch 警报来使用。结果警报是必需的，因此您无法删除结果警报。
   - 对于阻止日期和阻止时段，您可以添加新的日期或时间，也可以删除或更新现有的日期或时间。所有日期和时间均采用 UTC 时间。
6. 选择保存。

删除练习运行配置

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 在多可用区下，选择可用区自动转移。
3. 在资源可用区自动转移配置下，选择资源。
4. 在操作菜单中，选择删除练习运行配置。
5. 在确认模态对话框中，键入 `Delete`，然后选择删除。

   请注意，在控制台中删除练习运行配置也会禁用资源的可用区自动转移。可用区自动转移需要为资源配置练习运行。

取消区域自动换档

要停止资源正在进行的区域自动移位，必须取消区域自动移位。

停止正在进行的分区自动换档

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 在多可用区下，选择可用区转移。
3. 选择要取消的分区自动切换，然后选择 “取消分区偏移”。
4. 在确认模态对话框中，选择确认。

开始练习跑分区移动

本节中的步骤说明了如何在 ARC 控制台上开始按需练习区域切换。要以编程方式使用可用区转移和可用区自动转移，请参阅可用区转移和可用区自动转移 API 参考指南。

在配置区域自动移位并创建练习跑配置后，您可以开始练习跑区域移动。

要开始练习，请进行区域移动

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 在多可用区下，选择可用区自动转移。
3. 在 “区域自动移位资源” 下，浏览到配置了区域自动移位的单个资源。
4. 在资源概述页面上，选择开始练习运行。
5. 选择一个可用区，然后输入练习跑步的评论。练习运行会将流量从您选择的可用区转移出去。
6. 选择启动。

取消练习运行可用区转移

本节中的步骤说明了如何在 ARC 控制台上取消区域偏移。要以编程方式使用可用区转移和可用区自动转移，请参阅可用区转移和可用区自动转移 API 参考指南。

你可以取消分区移动，也可以取消自己发起的练习。您也可以取消为区域自动移位练习跑的资源而 AWS 开始的区域移动。

取消练习运行可用区转移

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 在多可用区下，选择可用区转移。
3. 选择要取消的练习跑区域移动，然后选择 “取消分区移动” 或 “取消练习”。
4. 在确认模态对话框中，选择确认。

启用或禁用自动移位观察者通知

您可以配置区域自动切换，以便在 AWS 启动自动换档时通过 Amazon 通知您 EventBridge，将流量从可能受损的可用区域转移出去。您必须在要接收通知 AWS 区域 的每个选项中配置此选项。您无需使用区域自动移位配置任何特定资源即可启用这些单独的通知。有关更多信息，请参阅 在 Amazon 上使用区域自动换档 EventBridge。

本节中的步骤说明了如何使用 Amazon 应用程序恢复控制器 (ARC) 控制台启用自动移位观察者通知。要以编程方式使用可用区转移，请参阅可用区转移和可用区自动转移 API 参考指南。

启用或禁用自动移位旁听者通知

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 在 “入门” 下，选择 “启用自动移位旁听者通知”。
3. 在确认对话框中，选择启用旁听者通知。

使用以下方法测试区域自动换档 AWS FIS

您可以使用 AWS Fault Injection Service 来设置和运行实验，以帮助您模拟现实世界中的条件，例如 “可用区可用性：电源中断” 场景，该场景将演示在可能存在广泛的 AZ 损伤期间，在启用自动换档的资源上 AWS 启动区域自动换档时会发生什么。

启动`aws:arc:start-zonal-autoshift`恢复操作允许您演示在执行可用区可用性场景 AWS 区域期间，如何自动 AWS 将启用区域自动移位的资源的流量从可能受损的可用区转移出去，并将其重新路由到正常 AZs 运行状态。

例如，您可以使用 AWS FIS 场景库来模拟由电源中断引起的可用区损害。在本实验中，在可用区电源中断开始五分钟后，恢复操作`aws:arc:start-zonal-autoshift`会自动将资源流量从指定可用区转移出去。在电力中断的剩余25分钟内，流量会被移动，以演示在可能存在广泛的可用区损伤时如何触发自动换档。实验完成后，交通转移结束，交通 AZs 再次开始流向所有人。此过程演示如何从影响可用区的电源事件中完全恢复。

实验与分区自动换档练习有何不同

AWS FIS 实验与区域自动移位练习的不同之处在于，在练习运行期间，ARC 会将您的资源流量从一个可用区转移出去，这是正常流程的一部分，以确保您的应用程序能够承受可用区的损失。但是，在 AWS FIS 实验中， AWS FIS 演示如何代表您为启用自动换档的资源触发 AZ 损伤和自动换档，然后在损伤得到解决后取消自动换档。

在 AWS FIS 启动的区域偏移运行期间，您无法对其进行更新。此外，如果您取消外面的区域偏移 AWS FIS，则 AWS FIS 实验结束。

AWS FIS 基于到期的安全机制

AWS FIS 使用StartZonalShift、UpdateZonalShift和 CancelZonalShiftAPI 操作管理区域偏移，作为安全机制，将这些请求的`expiresIn`字段设置为 1 分钟。这使得 AWS FIS 在出现意外事件（例如网

络中断或系统问题）时可以快速回滚区域偏移。在 ARC 控制台中，到期时间字段将显示 AWS FIS-managed，实际的预期到期时间由区域移位操作中指定的持续时间决定。有关练习跑的更多信息，请参阅分区自动换档和练习跑的工作原理

在给定时间，应用的区域偏移不能超过一个。也就是说，只有一家诊所对资源进行区域移动、客户启动的区域移动、自动移位或 AWS FIS 实验。当第二次区域偏移开始时，ARC 会遵循优先顺序来确定哪种区域偏移类型对资源有效。有关区域偏移优先级的更多信息，请参阅区域偏移的优先级。

有关 AWS FIS 恢复操作的更多信息，请参阅《AWS Fault Injection Service 用户指南》中的AWS FIS 恢复操作。

在 Amazon 应用程序恢复控制器 (ARC) 中记录和监控区域自动切换

您可以使用 AWS CloudTrail 和 Amazon 监控亚马逊 EventBridge 应用程序恢复控制器 (ARC) 中的区域自动切换，以分析模式并帮助解决问题。

主题

使用记录区域自动移位 API 调用 AWS CloudTrail

ARC 的 Zonal autoshift 与 AWS CloudTrail一项服务集成，该服务提供用户、角色或 AWS 服务在 ARC 中采取的操作的记录。 CloudTrail 将所有用于区域偏移的 API 调用捕获为事件。捕获的调用包括来自 ARC 控制台的调用和用于区域移位的 ARC API 操作的代码调用。

如果您创建跟踪，则可以将 CloudTrail 事件持续传输到 Amazon S3 存储桶，包括用于区域转移的事件。如果您未配置跟踪，您仍然可以在 CloudTrail 控制台的 “事件历史记录” 中查看最新的事件。

使用收集的信息 CloudTrail，您可以确定向 ARC 发出的区域转移请求、发出请求的 IP 地址、谁提出了请求、何时提出请求以及其他详细信息。

要了解更多信息 CloudTrail，请参阅《AWS CloudTrail 用户指南》。

区域自动换档信息在 CloudTrail

CloudTrail 在您创建账户 AWS 账户 时已在您的账户上启用。当 ARC 中发生区域自动移位的活动时，该活动将与其他 AWS 服务 CloudTrail 事件一起记录在事件历史记录中。您可以在中查看、搜索和下载最近发生的事件 AWS 账户。有关更多信息，请参阅使用 CloudTrail 事件历史记录。

要持续记录您的事件 AWS 账户，包括 ARC 中区域自动移位的事件，请创建一条跟踪。跟踪允许 CloudTrail 将日志文件传输到 Amazon S3 存储桶。预设情况下，在控制台中创建跟踪记录时，此跟踪记录应用于所有 AWS 区域。跟踪记录 AWS 分区中所有区域的事件，并将日志文件传送到您指定的 Amazon S3 存储桶。此外，您可以配置其他 AWS 服务，以进一步分析和处理 CloudTrail 日志中收集的事件数据。有关更多信息，请参阅下列内容：

- 创建跟踪记录概述
- CloudTrail 支持的服务和集成
- 配置 Amazon SNS 通知 CloudTrail
- 接收来自多个地区的 CloudTrail 日志文件和接收来自多个账户的 CloudTrail 日志文件

所有 ARC 操作均由 Amazon 应用程序恢复控制器的路由控制 API 参考指南记录 CloudTrail 并记录在案。例如，调用`StartZonalShift`和`ListManagedResources`操作会在 CloudTrail 日志文件中生成条目。

每个事件或日志条目都包含有关生成请求的人员信息。身份信息有助于您确定以下内容：

- 请求是使用根证书还是 AWS Identity and Access Management (IAM) 用户凭证发出。
- 请求是使用角色还是联合用户的临时安全凭证发出的。
- 请求是否由其他 AWS 服务发出。

有关更多信息，请参阅 CloudTrail userIdentity 元素。

在事件历史记录中查看 ARC 事件

CloudTrail 允许您在事件历史记录中查看最近的事件。有关更多信息，请参阅《AWS CloudTrail 用户指南》中的 “使用 CloudTrail 事件历史记录”。

了解区域自动移位日志文件条目

跟踪是一种配置，允许将事件作为日志文件传输到您指定的 Amazon S3 存储桶。 CloudTrail 日志文件包含一个或多个日志条目。事件代表来自任何来源的单个请求，包括有关请求的操作、操作的日期和时间、请求参数等的信息。 CloudTrail 日志文件不是公共 API 调用的有序堆栈跟踪，因此它们不会按任何特定的顺序出现。

以下示例显示了一个 CloudTrail 日志条目，该条目演示了区域自动移位的`ListManagedResources`操作。

```
{
```

```
      "eventVersion": "1.08",
      "userIdentity": {
        "type": "AssumedRole",
        "principalId": "A1B2C3D4E5F6G7EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/admin",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
          "sessionIssuer": {
            "type": "Role",
            "principalId": "AROA33L3W36EXAMPLE",
            "arn": "arn:aws:iam::111122223333:role/admin",
            "accountId": "111122223333",
            "userName": "EXAMPLENAME"
          },
          "webIdFederationData": {},
          "attributes": {
            "creationDate": "2022-11-14T16:01:51Z",
            "mfaAuthenticated": "false"
          }
        }
      },
      "eventTime": "2022-11-14T16:14:41Z",
      "eventSource": "arc-zonal-shift.amazonaws.com",
      "eventName": "ListManagedResources",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.50",
      "userAgent": "Boto3/1.17.101 Python/3.8.10 Linux/4.14.231-180.360.amzn2.x86_64
exec-env/AWS_Lambda_python3.8 Botocore/1.20.102",
      "requestParameters": null,
      "responseElements": null,
      "requestID": "VGXG4ZUE7UZTVCMTJGIAF_EXAMPLE",
      "eventID": "4b5c42df-1174-46c8-be99-d67_EXAMPLE",
      "readOnly": true,
      "eventType": "AwsApiCall",
      "managementEvent": true,
      "recipientAccountId": "111122223333"
      "eventCategory": "Management"
      }
    }
```

在 Amazon 上使用区域自动换档 EventBridge

使用 Amazon EventBridge，您可以设置事件驱动的规则，以监控您的区域自动转移资源并启动使用其他服务的目标操作。 AWS 例如，您可以设置发送电子邮件通知的规则，方法是在区域自动移位练习开始时向 Amazon SNS 主题发送信号。

您可以在 Amazon 中创建规则 EventBridge 以对区域自动移位进行操作。区域自动移位事件指定有关练习跑或自动换档的状态信息，例如，练习跑何时开始。您可以配置区域自动移动，以通知您为服务启用的资源的区域自动移位事件。

除了或取代其他通知之外，您还可以选择启用自动移位观察者通知，每当为可能受损的可用区 AWS 启动自动换档时，它都会提供通知事件。Autoshift 观察者通知与您为区域自动移位启用的资源流量从可用区转移出去时收到的通知是分开的。您无需使用区域自动移位配置任何资源即可启用自动移位观察者通知。有关更多信息，请参阅 启用和使用分区自动换档。

要捕获您感兴趣的特定区域自动移位事件，请定义 EventBridge 可用于检测事件的特定事件模式。事件模式与它们匹配的事件具有相同的结构。模式引用了您要匹配的字段，并提供您所查找的值。

尽最大努力发出事件。在正常运行 EventBridge 情况下，它们几乎实时地从 ARC 交付到。但是，可能会出现延迟或阻止事件交付的情况。

有关 EventBridge 规则如何与事件模式配合使用的信息，请参阅中的事件和事件模式 EventBridge。

使用以下命令监控区域自动移位资源 EventBridge

借 EventBridge助，您可以创建规则，以定义 ARC 为其资源发出事件时要采取的操作。例如，您可以创建一条规则，在区域自动移位练习开始时发送电子邮件。

要在控制台中键入或复制并粘贴事件模式，请选择该选项以在 EventBridge 控制台中使用 Enter my own 选项。为了帮助您确定可能对您有用的事件模式，本主题包括区域自动移位事件匹配模式和区域自动移位事件的示例，供您使用。

要为资源事件创建规则

1. 打开 Amazon EventBridge 控制台，网址为https://console.aws.amazon.com/events/。
2. 选择您 AWS 区域 要在其中创建规则的区域，即您有兴趣观看其赛事的区域。
3. 选择 Create rule (创建规则)。
4. 输入规则的 Name (名称) 和“Description (描述)”（可选）。
5. 对于事件总线，保留默认值，即默认。

6. 选择下一步。
7. 对于构建事件模式步骤，对于事件源，保留默认值，即 AWS 事件。
8. 在示例事件下，选择输入我自己的。
9. 对于示例事件，键入或复制并粘贴事件模式。

区域自动移位事件模式示例

事件模式与它们匹配的事件具有相同的结构。模式引用了您要匹配的字段，并提供您所查找的值。

您可以将此部分中的事件模式复制并粘贴 EventBridge 到中，以创建可用于监控区域自动移位操作和资源的规则。

在为可用区自动转移事件创建事件模式时，可以为 `detail-type` 指定以下任一选项：

- `Autoshift In Progress`
- `Autoshift Completed`
- `Practice Run Started`
- `Practice Run Succeeded`
- `Practice Run Interrupted`
- `Practice Run Failed`
- `FIS Experiment Autoshift In Progress`
- `FIS Experiment Autoshift Completed`
- `FIS Experiment Autoshift Canceled`

当练习运行中断时，可参阅 `additionalFailureInfo` 字段，以详细了解导致中断的原因。

您可以通过启用自动 AWS 换档观察者通知来选择监控所有自动换档。启用自动移位旁听者通知后，要接收通知，请选择接收区域自动移位详细信息类型的通知。`Autoshift In Progress`要查看启用自动移位旁听者通知的步骤，请参阅 启用和使用分区自动换档。

有关示例，请参阅 “区域自动移位事件示例” 部分。

- 从已开始自动换档的区域自动切换中选择所有事件。

  请注意以下几点：

- 如果您启用了自动移位观察者通知，ARC 会返回所有自动移位事件。
- 如果您未启用自动移位观察者通知，则只有在自动移位中包含您为区域自动移位配置的资源时，ARC 才会返回自动移位事件。

```
{
    "source": [
        "aws.arc-zonal-shift"
    ],
    "detail-type": [
        "Autoshift In Progress"
    ]
}
```

- 从已开始练习跑的区域自动切换中选择所有赛事。

```
{
    "source": [
        "aws.arc-zonal-shift"
    ],
    "detail-type": [
        "Practice Run Started"
    ]
}
```

- 从区域自动移位中选择练习跑失败的所有赛事。

```
{
    "source": [
        "aws.arc-zonal-shift"
    ],
    "detail-type": [
        "Practice Run Failed"
    ]
}
```

区域自动移位事件示例

本节包括区域自动移位操作的示例事件。

以下是该`Autoshift In Progress`操作的示例事件，当 1) 启用自动移位观察者通知且 2) 您尚未将资源配置为包含在自动移位中的区域自动移位时：

```
{
    "version": "0",
    "id": "05d4d2d5-9c76-bfea-72d2-d4614802adb4",
    "detail-type": "Autoshift In Progress",
    "source": "aws.arc-zonal-shift",
    "account": "111122223333",
    "time": "2023-11-16T23:38:14Z",
    "region": "us-east-1",
    "resources": [],
    "detail": {
        "version": "0.0.1",
        "data": "",
        "metadata": {
            "awayFrom": "use1-az2",
            "notes":"AWS has started an autoshift for an impaired Availability Zone.
 This notification
            is separate from autoshift notifications for resources, if any, that you
 have configured for
            zonal autoshift. For details, see the Developer Guide."
        }
    }
}
```

以下是该Autoshift In Progress操作的示例事件，当 1) 禁用自动移位观察者通知以及 2) 您已将资源配置为包含在自动移位中的区域自动移位时：

```
{
    "version": "0",
    "id": "05d4d2d5-9c76-bfea-72d2-d4614802adb4",
    "detail-type": "Autoshift In Progress",
    "source": "aws.arc-zonal-shift",
    "account": "111122223333",
    "time": "2023-11-16T23:38:14Z",
    "region": "us-east-1",
    "resources": [
        "TEST-EXAMPLE-2023-11-16-23-28-11-5"
    ],
    "detail": {
        "version": "0.0.1",
        "data": "",
        "metadata": {
            "awayFrom": "use1-az2",
            "notes":""
```

```
        }
    }
}
```

以下是该Practice Run Interrupted操作的示例事件：

```
{
    "version": "0",
    "id": "05d4d2d5-9c76-bfea-72d2-d4614802adb4",
    "detail-type": "Practice Run Interrupted",
    "source": "aws.arc-zonal-shift",
    "account": "111122223333",
    "time": "2023-11-16T23:38:14Z",
    "region": "us-east-1",
    "resources": [
        "TEST-EXAMPLE-2023-11-16-23-28-11-5"
    ],
    "detail": {
        "version": "0.0.1",
        "data": {
            "additionalFailureInfo": "Practice run interrupted. The blocking alarm
 entered ALARM state."
        },
        "metadata": {
            "awayFrom": "use1-az2"
        }
    }
}
```

以下是该FIS Experiment Autoshift In Progress操作的示例事件：

```
{
    "version": "0",
    "id": "05d4d2d5-9c76-bfea-72d2-d4614802adb4",
    "detail-type": "FIS Experiment Autoshift In Progress",
    "source": "aws.arc-zonal-shift",
    "account": "111122223333",
    "time": "2023-11-16T23:38:14Z",
    "region": "us-east-1",
    "resources": [
        "TEST-EXAMPLE-2023-11-16-23-28-11-5"
    ],
    "detail": {
```

```
        "version": "0.0.1",
        "data": "",
        "metadata": {
            "awayFrom": "use1-az2",
            "notes":""
        }
    }
}
```

指定要用作目标的 CloudWatch 日志组

创建 EventBridge 规则时，必须指定将与该规则匹配的事件发送到哪个目标。有关可用目标的列表 EventBridge，请参阅 EventBridge 控制台中的可用目标。您可以添加到 EventBridge 规则的目标之一是 Amazon CloudWatch 日志组。本节介绍将 CloudWatch 日志组添加为目标的要求，并提供了在创建规则时添加日志组的过程。

要将 CloudWatch 日志组添加为目标，可以执行以下操作之一：

- 创建新的日志组
- 选择现有的日志组

如果您在创建规则时使用控制台指定了新的日志组，则 EventBridge 会自动为您创建该日志组。确保用作 EventBridge 规则目标的日志组以开头/aws/events。如果要选择现有的日志组，请注意，只有以开头的日志组才/aws/events会作为选项出现在下拉菜单中。有关更多信息，请参阅 Amazon CloudWatch 用户指南中的创建新日志组。

如果您使用控制台之外的 CloudWatch 操作创建或使用 CloudWatch 日志组作为目标，请确保正确设置权限。如果您使用控制台向 EventBridge 规则添加日志组，则该日志组的基于资源的策略会自动更新。但是，如果您使用 AWS Command Line Interface 或 S AWS DK 来指定日志组，则必须更新该日志组的基于资源的策略。以下示例策略说明了您必须在基于资源的策略中为日志组定义的权限：

```
{
  "Statement": [
    {
      "Action": [
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ],
      "Effect": "Allow",
      "Principal": {
```

```
        "Service": [
          "events.amazonaws.com",
          "delivery.logs.amazonaws.com"
        ]
      },
      "Resource": "arn:aws:logs:region:account:log-group:/aws/events/*:*",
      "Sid": "TrustEventsToStoreLogEvent"
    }
  ],
  "Version": "2012-10-17"
}
```

您无法使用控制台为日志组配置基于资源的策略。要向基于资源的策略添加所需的权限，请使用 CloudWatch PutResourcePolicyAPI 操作。然后，您可以使用 describe-resource-policiesCLI 命令来检查您的策略是否已正确应用。

为资源事件创建规则并指定 CloudWatch 日志组目标

1. 打开 Amazon EventBridge 控制台，网址为https://console.aws.amazon.com/events/。
2. 选择 AWS 区域 要在其中创建规则的。
3. 选择 “创建规则”，然后输入有关该规则的任何信息，例如事件模式或计划详细信息。

   有关为 ARC 创建 EventBridge 规则的更多信息，请参阅本主题前面的部分。
4. 在 “选择目标” 页面上，选择CloudWatch作为您的目标。
5. 从下拉菜单中选择一个 CloudWatch 日志组。

用于区域自动换档的 Identity and Access Management

AWS Identity and Access Management (IAM) AWS 服务 可帮助管理员安全地控制对 AWS 资源的访问权限。IAM 管理员控制谁可以进行身份验证（登录）和授权（有权限）使用 ARC 资源。您可以使用 IAM AWS 服务 ，无需支付额外费用。

内容

- ARC 中的区域自动换档如何与 IAM 配合使用
- 基于身份的区域自动换档策略示例
- 在 ARC 中使用服务关联角色进行区域自动切换
- AWS Amazon 应用程序恢复控制器 (ARC) 中区域自动切换的托管策略

ARC 中的区域自动换档如何与 IAM 配合使用

在使用 IAM 在 Amazon 应用程序恢复控制器 (ARC) 中管理对区域自动移位的访问权限之前，请先了解有哪些 IAM 功能可用于区域自动切换。

您可以在 ARC 中使用区域自动移位的 IAM 功能

IAM 特征	区域自动换档支持
基于身份的策略	是
基于资源的策略	否
策略操作	是
策略资源	是
策略条件键	是
ACLs	否
ABAC（策略中的标签）	部分
临时凭证	是
主体权限	是
服务角色	否
服务相关角色	是

要全面了解 AWS 服务如何与大多数 IAM 功能配合使用，请参阅 IAM 用户指南中的与 IAM 配合使用的 AWS 服务。

ARC 基于身份的策略

支持基于身份的策略：是

基于身份的策略是可附加到身份（如 IAM 用户、用户组或角色）的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的使用客户管理型策略定义自定义 IAM 权限。

通过使用 IAM 基于身份的策略，您可以指定允许或拒绝的操作和资源以及允许或拒绝操作的条件。您无法在基于身份的策略中指定主体，因为它适用于其附加的用户或角色。要了解可在 JSON 策略中使用的所有元素，请参阅《IAM 用户指南》中的 IAM JSON 策略元素引用。

要查看 ARC 基于身份的策略的示例，请参阅。Amazon 应用程序恢复控制器 (ARC) 中基于身份的策略示例

ARC 内部基于资源的政策

支持基于资源的策略：否

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资源的访问。

ARC 的政策行动

支持策略操作：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

JSON 策略的 `Action` 元素描述可用于在策略中允许或拒绝访问的操作。策略操作通常与关联的 AWS API 操作同名。有一些例外情况，例如没有匹配 API 操作的仅限权限 操作。还有一些操作需要在策略中执行多个操作。这些附加操作称为相关操作。

在策略中包含操作以授予执行关联操作的权限。

要查看区域自动移位的 ARC 操作列表，请参阅《服务授权参考》中的 Amazon Route 53 区域移位定义的操作。

ARC 中用于区域自动移位的策略操作在操作前使用以下前缀：

```
arc-zonal-shift
```

要在单个语句中指定多项操作，请使用逗号将它们隔开。例如，以下内容：

```
"Action": [
      "arc-zonal-shift:action1",
      "arc-zonal-shift:action2"
         ]
```

您也可以使用通配符（*）指定多个操作。例如，要指定以单词 Describe 开头的所有操作，包括以下操作：

```
"Action": "arc-zonal-shift:Describe*"
```

要查看 ARC 基于身份的区域自动移位策略示例，请参阅。基于身份的区域自动换档策略示例

ARC 中区域自动换档的策略资源

支持策略资源：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

Resource JSON 策略元素指定要向其应用操作的一个或多个对象。语句必须包含 Resource 或 NotResource 元素。作为最佳实践，请使用其 Amazon 资源名称（ARN）指定资源。对于支持特定资源类型（称为资源级权限）的操作，您可以执行此操作。

对于不支持资源级权限的操作（如列出操作），请使用通配符（*）指示语句应用于所有资源。

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

要查看资源类型及其列表 ARNs，以及您可以使用每种资源的 ARN 指定的操作，请参阅《服务授权参考》中的以下主题：

- Amazon Route 53 定义的操作——区域移动

要查看可与条件键配合使用的操作和资源，请参阅《服务授权参考》中的以下主题：

- 由 Amazon Route 53 定义的条件键——区域移动

要查看 ARC 基于身份的区域自动移位策略示例，请参阅。基于身份的区域自动换档策略示例

ARC 中分区自动切换的策略条件密钥

支持特定于服务的策略条件键：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

在 Condition 元素（或 Condition 块）中，可以指定语句生效的条件。Condition 元素是可选的。您可以创建使用条件运算符（例如，等于或小于）的条件表达式，以使策略中的条件与请求中的值相匹配。

如果您在一个语句中指定多个 Condition 元素，或在单个 Condition 元素中指定多个键，则 AWS 使用逻辑 AND 运算评估它们。如果您为单个条件键指定多个值，则使用逻辑OR运算来 AWS 评估条件。在授予语句的权限之前必须满足所有的条件。

在指定条件时，您也可以使用占位符变量。例如，只有在使用 IAM 用户名标记 IAM 用户时，您才能为其授予访问资源的权限。有关更多信息，请参阅《IAM 用户指南》中的 IAM 策略元素：变量和标签。

AWS 支持全局条件密钥和特定于服务的条件密钥。要查看所有 AWS 全局条件键，请参阅 IAM 用户指南中的AWS 全局条件上下文密钥。

要查看区域自动移位的 ARC 条件键列表，请参阅《服务授权参考》中的以下主题：

- Amazon Route 53 可用区转移的条件键

要查看可与条件键配合使用的操作和资源，请参阅《服务授权参考》中的以下主题：

- Amazon Route 53 可用区转移定义的操作

要查看 ARC 基于身份的区域自动移位策略示例，请参阅。基于身份的区域自动换档策略示例

ARC 中的访问控制列表 (ACLs)

支持 ACLs：否

访问控制列表 (ACLs) 控制哪些委托人（账户成员、用户或角色）有权访问资源。 ACLs 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

使用 ARC 实现基于属性的访问控制 (ABAC)

支持 ABAC（策略中的标签）：部分支持

基于属性的访问控制（ABAC）是一种授权策略，该策略基于属性来定义权限。在中 AWS，这些属性称为标签。您可以向 IAM 实体（用户或角色）和许多 AWS 资源附加标签。标记实体和资源是 ABAC 的第一步。然后设计 ABAC 策略，以在主体的标签与他们尝试访问的资源标签匹配时允许操作。

ABAC 在快速增长的环境中非常有用，并在策略管理变得繁琐的情况下可以提供帮助。

要基于标签控制访问，您需要使用 aws:ResourceTag/*key-name*、aws:RequestTag/*key-name* 或 aws:TagKeys 条件键在策略的条件元素中提供标签信息。

如果某个服务对于每种资源类型都支持所有这三个条件键，则对于该服务，该值为是。如果某个服务仅对于部分资源类型支持所有这三个条件键，则该值为部分。

有关 ABAC 的更多信息，请参阅《IAM 用户指南》中的使用 ABAC 授权定义权限。要查看设置 ABAC 步骤的教程，请参阅《IAM 用户指南》中的使用基于属性的访问权限控制（ABAC）。

ARC 中的区域自动换档包括对 ABAC 的以下部分支持：

- 区域自动移位支持 ABAC，这些资源在 ARC 中注册以进行区域移动。有关网络负载均衡器和应用程序负载均衡器托管资源的 ABAC 的更多信息，请参阅《Elastic Load Balancing 用户指南》之 Elastic Load Balancing 中的 ABAC。

在 ARC 中使用临时证书

支持临时凭证：是

当你使用临时证书登录时，有些 AWS 服务 不起作用。有关更多信息，包括哪些 AWS 服务 适用于临时证书，请参阅 IAM 用户指南中的AWS 服务 与 IAM 配合使用的信息。

如果您使用除用户名和密码之外的任何方法登录，则 AWS Management Console 使用的是临时证书。例如，当您 AWS 使用公司的单点登录 (SSO) 链接进行访问时，该过程会自动创建临时证书。当您以用户身份登录控制台，然后切换角色时，您还会自动创建临时凭证。有关切换角色的更多信息，请参阅《IAM 用户指南》中的从用户切换到 IAM 角色（控制台）。

您可以使用 AWS CLI 或 AWS API 手动创建临时证书。然后，您可以使用这些临时证书进行访问 AWS。 AWS 建议您动态生成临时证书，而不是使用长期访问密钥。有关更多信息，请参阅 IAM 中的临时安全凭证。

ARC 的跨服务主体权限

支持转发访问会话（FAS）：是

当您使用 IAM 实体（用户或角色）在中执行操作时 AWS，您被视为委托人。策略向主体授予权限。使用某些服务时，您可能会执行一个操作，此操作然后在不同服务中触发另一个操作。在这种情况下，您必须具有执行这两个操作的权限。

要查看某项操作是否需要策略中的其他相关操作，请参阅《服务授权参考》中的以下主题：

- Amazon Route 53 可用区转移

ARC 的服务角色

支持服务角色：否

服务角色是由一项服务担任、代表您执行操作的 IAM 角色。IAM 管理员可以在 IAM 中创建、修改和删除服务角色。有关更多信息，请参阅《IAM 用户指南》中的创建向 AWS 服务委派权限的角色。

ARC 的服务相关角色

支持服务相关角色：是

服务相关角色是一种与服务相关联的 AWS 服务服务角色。服务可以代入代表您执行操作的角色。服务相关角色出现在您的中 AWS 账户 ，并且归服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。

有关创建或管理 ARC 服务相关角色的详细信息，请参阅在 ARC 中使用服务关联角色进行区域自动切换。

有关创建或管理服务相关角色的详细信息，请参阅能够与 IAM 搭配使用的AWS 服务。在表中查找服务相关角色列中包含 Yes 的表。选择是链接以查看该服务的服务相关角色文档。

基于身份的区域自动换档策略示例

默认情况下，用户和角色无权创建或修改 ARC 资源。他们也无法使用 AWS Management Console、AWS Command Line Interface (AWS CLI) 或 AWS API 执行任务。要授予用户对所需资源执行操作的权限，IAM 管理员可以创建 IAM 策略。管理员随后可以向角色添加 IAM 策略，用户可以代入角色。

要了解如何使用这些示例 JSON 策略文档创建基于 IAM 身份的策略，请参阅《IAM 用户指南》中的创建 IAM 策略（控制台）。

有关 ARC 定义的操作和资源类型（包括每种资源类型的格式）的详细信息，请参阅《服务授权参考》中的 Amazon 应用程序恢复控制器 (ARC) 的操作、资源和条件密钥。 ARNs

主题

- 策略最佳实践
- 示例：区域自动切换控制台访问权限
- 示例：ARC API 操作

策略最佳实践

基于身份的策略决定了某人是否可以在您的账户中创建、访问或删除 ARC 资源。这些操作可能会使 AWS 账户产生成本。创建或编辑基于身份的策略时，请遵循以下指南和建议：

- 开始使用 AWS 托管策略并转向最低权限权限 — 要开始向用户和工作负载授予权限，请使用为许多常见用例授予权限的AWS 托管策略。它们在你的版本中可用 AWS 账户。我们建议您通过定义针对您的用例的 AWS 客户托管策略来进一步减少权限。有关更多信息，请参阅《IAM 用户指南》中的AWS 托管式策略或工作职能的AWS 托管式策略。
- 应用最低权限：在使用 IAM 策略设置权限时，请仅授予执行任务所需的权限。为此，您可以定义在特定条件下可以对特定资源执行的操作，也称为最低权限许可。有关使用 IAM 应用权限的更多信息，请参阅《IAM 用户指南》中的 IAM 中的策略和权限。
- 使用 IAM 策略中的条件进一步限制访问权限：您可以向策略添加条件来限制对操作和资源的访问。例如，您可以编写策略条件来指定必须使用 SSL 发送所有请求。如果服务操作是通过特定的方式使用的，则也可以使用条件来授予对服务操作的访问权限 AWS 服务，例如 AWS CloudFormation。有关更多信息，请参阅《IAM 用户指南》中的 IAM JSON 策略元素：条件。
- 使用 IAM Access Analyzer 验证您的 IAM 策略，以确保权限的安全性和功能性 – IAM Access Analyzer 会验证新策略和现有策略，以确保策略符合 IAM 策略语言（JSON）和 IAM 最佳实践。IAM Access Analyzer 提供 100 多项策略检查和可操作的建议，以帮助您制定安全且功能性强的策略。有关更多信息，请参阅《IAM 用户指南》中的使用 IAM Access Analyzer 验证策略。
- 需要多重身份验证 (MFA)-如果 AWS 账户您的场景需要 IAM 用户或根用户，请启用 MFA 以提高安全性。若要在调用 API 操作时需要 MFA，请将 MFA 条件添加到您的策略中。有关更多信息，请参阅《IAM 用户指南》中的使用 MFA 保护 API 访问。

有关 IAM 中的最佳实操的更多信息，请参阅《IAM 用户指南》中的 IAM 中的安全最佳实践。

示例：区域自动切换控制台访问权限

要访问 Amazon 应用程序恢复控制器 (ARC) 控制台，您必须拥有一组最低权限。这些权限必须允许您列出和查看有关您的 ARC 资源的详细信息 AWS 账户。如果创建比必需的最低权限更为严格的基于身份的策略，对于附加了该策略的实体（用户或角色），控制台将无法按预期正常运行。

对于仅调用 AWS CLI 或 AWS API 的用户，您无需为其设置最低控制台权限。相反，只允许访问与其尝试执行的 API 操作相匹配的操作。

要执行某些任务，用户必须有权在 ARC 中创建与 zonal auto-shift 关联的服务相关角色。要了解更多信息，请参阅 在 ARC 中使用服务关联角色进行区域自动切换。

要向用户提供在中使用区域自动移位的完全访问权限 AWS Management Console，请向用户附加类似以下内容的策略：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                   "arc-zonal-shift:ListManagedResources",
                   "arc-zonal-shift:GetManagedResource",
                   "arc-zonal-shift:ListZonalShifts",
                   "arc-zonal-shift:StartZonalShift",
                   "arc-zonal-shift:UpdateZonalShift",
                   "arc-zonal-shift:CancelZonalShift",
                   "arc-zonal-shift:CreatePracticeRunConfiguration",
                   "arc-zonal-shift:DeletePracticeRunConfiguration",
                   "arc-zonal-shift:ListAutoshifts",
                   "arc-zonal-shift:UpdatePracticeRunConfiguration",
                   "arc-zonal-shift:UpdateZonalAutoshiftConfiguration"
             ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DescribeAvailabilityZones",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "cloudwatch:DescribeAlarms",
            "Resource": "*"
        }
    ]
}
```

示例：ARC API 操作

您可以使用策略来确保用户可以使用区域自动切换的 ARC API 操作来配置区域自动切换，从而代表您将应用程序资源流量从可用区 AWS 转移到健康 AZs 可用区，从而帮助缩短事件期间恢复的时间。AWS 区域要提供这些权限，请附加与用户需要使用的 API 操作相对应的策略，如下所述。

要执行某些任务，用户必须拥有与 ARC 关联的服务相关角色的权限。创建服务相关角色所需的权限包含在以下示例策略中。要了解更多信息，请参阅 在 ARC 中使用服务关联角色进行区域自动切换。

要使用区域自动移位的 API 操作，请向用户附加如下策略：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                   "arc-zonal-shift:ListManagedResources",
                   "arc-zonal-shift:GetManagedResource",
                   "arc-zonal-shift:ListZonalShifts",
                   "arc-zonal-shift:StartZonalShift",
                   "arc-zonal-shift:UpdateZonalShift",
                   "arc-zonal-shift:CancelZonalShift",
                   "arc-zonal-shift:CreatePracticeRunConfiguration",
                   "arc-zonal-shift:DeletePracticeRunConfiguration",
                   "arc-zonal-shift:ListAutoshifts",
                   "arc-zonal-shift:UpdatePracticeRunConfiguration",
                   "arc-zonal-shift:UpdateZonalAutoshiftConfiguration"
             ],
            "Resource": "*"
        },
        {
            "Effect" : "Allow",
            "Action" : [
                   "cloudwatch:DescribeAlarms",
                   "health:DescribeEvents"
            ],
            "Resource" : "*"
        },
        {
            "Effect" : "Allow",
            "Action" : [
                   "arc-zonal-shift:CancelZonalShift",
                   "arc-zonal-shift:GetManagedResource",
                   "arc-zonal-shift:StartZonalShift",
                   "arc-zonal-shift:UpdateZonalShift"
            ],
            "Resource" : "*"
        }
    ]
```

```
}
```

在 ARC 中使用服务关联角色进行区域自动切换

Amazon 应用程序恢复控制器中的区域自动切换使用 AWS Identity and Access Management (IAM) 服务相关角色。服务相关角色是一种独特的 IAM 角色，直接链接到服务（在本例中为 ARC）。服务相关角色由 ARC 预定义，包括该服务出于特定目的代表您调用其他 AWS 服务所需的所有权限。

服务相关角色可以更轻松地设置 ARC，因为您不必手动添加必要的权限。ARC 定义服务相关角色的权限，除非另有定义，否则只有 ARC 可以担任其角色。定义的权限包括信任策略和权限策略，而且权限策略不能附加到任何其它 IAM 实体。

只有在首先删除服务相关角色的相关资源后，才能删除该角色。这可以保护您的 ARC 区域自动移位资源，因为您不会无意中移除对资源的访问权限。

有关支持服务相关角色的其他服务的信息，请参阅与 IAM 配合使用的AWS 服务，并在服务相关角色列中查找标有 “是” 的服务。选择是和链接，查看该服务的服务相关角色文档。

的服务相关角色权限 AWSService RoleForZonalAutoshiftPracticeRun

ARC 使用名为的服务相关角色AWSServiceRoleForZonalAutoshiftPracticeRun执行以下操作：

- 监控客户提供的 Amazon CloudWatch 警报和客户 AWS Health Dashboard 事件以进行练习
- 管理练习运行（练习可用区转移）

本节介绍适用于该服务相关角色的权限，以及有关创建、编辑和删除该角色的信息。

的服务相关角色权限 AWSService RoleForZonalAutoshiftPracticeRun

此服务相关角色使用托管策略 `AWSZonalAutoshiftPracticeRunSLRPolicy`。

AWSServiceRoleForZonalAutoshiftPracticeRun 服务相关角色仅信任以下服务来担任该角色：

- `practice-run.arc-zonal-shift.amazonaws.com`

要查看此策略的权限，请参阅《AWS 托管式策略参考》中的 AWSZonalAutoshiftPracticeRunSLRPolicy。

您必须配置权限，允许 IAM 实体（如用户、组或角色）创建、编辑或删除服务相关角色。有关更多信息，请参阅《IAM 用户指南》中的服务相关角色权限。

为 ARC 创建AWSServiceRoleForZonalAutoshiftPracticeRun服务相关角色

无需手动创建 AWSServiceRoleForZonalAutoshiftPracticeRun 服务相关角色。当你在 AWS Management Console、或 AWS SDK 中创建第一个练习运行配置时，ARC 会为你创建服务相关角色。 AWS CLI

如果您删除该服务相关角色，然后需要再次创建，您可以使用相同流程在账户中重新创建此角色。当您创建第一个练习运行配置时，ARC 会再次为您创建服务相关角色。

编辑 AR AWSServiceRoleForZonalAutoshiftPracticeRunC 的服务相关角色

ARC 不允许您编辑AWSServiceRoleForZonalAutoshiftPracticeRun服务相关角色。创建该服务相关角色后，将无法更改角色名称，因为可能有其它实体引用该角色。不过，您可以使用 IAM 编辑角色的说明。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 AR AWSServiceRoleForZonalAutoshiftPracticeRunC 的服务相关角色

如果不再需要使用某个需要服务相关角色的功能或服务，我们建议您删除该角色。这样就没有未被主动监控或维护的未使用实体。但是，您必须先清除服务相关角色的资源，然后才能手动删除它。

禁用自动切换后，您可以删除AWSServiceRoleForZonalAutoshiftPracticeRun与服务相关的角色。有关自动转移功能的更多信息，请参阅 ARC 中的区域偏移。

> **Note**
>
> 如果您尝试删除资源时 ARC 服务正在使用该角色，则删除服务角色可能会失败。如果发生这种情况，请等待几分钟，然后重新尝试删除该角色。

使用 IAM 手动删除服务相关角色

使用 IAM 控制台 AWS CLI、或 AWS API 删除 AWSServiceRoleForZonalAutoshiftPracticeRun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。

更新了用于区域自动换档的 ARC 服务相关角色

有关 ARC 服务相关角色 AWS 托管策略的更新，请参阅 ARC 的AWS 托管策略更新表。您也可以在 ARC 文档历史记录页面上订阅自动 RSS 提醒。

AWS Amazon 应用程序恢复控制器 (ARC) 中区域自动切换的托管策略

AWS 托管策略是由创建和管理的独立策略 AWS。 AWS 托管策略旨在为许多常见用例提供权限，以便您可以开始为用户、组和角色分配权限。

请记住，AWS 托管策略可能不会为您的特定用例授予最低权限权限，因为它们可供所有 AWS 客户使用。我们建议通过定义特定于您的使用场景的客户管理型策略来进一步减少权限。

您无法更改 AWS 托管策略中定义的权限。如果 AWS 更新 AWS 托管策略中定义的权限，则更新会影响该策略所关联的所有委托人身份（用户、组和角色）。AWS 最有可能在启动新的 API 或现有服务可以使用新 AWS 服务 的 API 操作时更新 AWS 托管策略。

有关更多信息，请参阅《IAM 用户指南》中的 AWS 托管策略。

AWS 托管策略：AWSZonalAutoshiftPracticeRunSLRPolicy

您不能将 `AWSZonalAutoshiftPracticeRunSLRPolicy` 附加到自己的 IAM 实体。此策略附加到服务相关角色，该角色允许 Amazon 应用程序恢复控制器 (ARC) 对区域自动切换执行以下操作：

- 监控客户提供的 Amazon CloudWatch 警报和客户 AWS Health Dashboard 事件以进行练习
- 管理练习运行（练习可用区转移）
- 管理练习跑和自动换档的平衡容量检查

有关更多信息，请参阅 在 ARC 中使用服务关联角色进行区域自动切换。

区域自动换档 AWS 托管策略更新

有关自该服务开始跟踪这些更改以来，ARC 中区域自动换档的 AWS 托管策略更新的详细信息，请参阅。Amazon 应用程序恢复控制器 (ARC) AWS 托管策略的更新要获得有关此页面变更的自动提醒，请订阅 ARC 文档历史记录页面上的 RSS 提要。

使用路由控制恢复 ARC 中的多区域应用程序

本节介绍如何使用 Amazon Application Recovery Controller (ARC) 中的路由控制功能来最大限度地减少中断，并帮助您在多个 AWS 应用程序中部署时为用户提供连续性 AWS 区域。

您还可以了解准备情况检查，这是ARC中的一项功能，可用于深入了解您的应用程序和资源是否为恢复做好了准备。

本节中的主题描述了路由控制和就绪检查功能、如何设置它们以及如何使用它们。

主题

ARC 中的路由控制

要将流量故障转移到多个应用程序副本 AWS 区域，您可以使用 Amazon 应用程序恢复控制器 (ARC) 中的路由控制，这些控制与 Amazon Route 53 中的特定类型的运行状况检查集成。路由控制是简单的开/关开关，使您可以将客户端流量从一个区域副本切换到另一个区域副本。流量重新路由通过使用 Amazon Route 53 DNS 记录设置的路由控制运行状况检查来完成。例如，DNS 故障转移记录，与每个区域中应用程序副本前面的域名相关联。

本节介绍路由控制的工作原理、如何设置路由控制组件以及如何使用它们重新路由流量以进行故障转移。

ARC 中的路由控制组件包括：集群、控制面板、路由控制和路由控制运行状况检查。所有路由控制都组合到控制面板上。您可以在 ARC 为您的集群创建的默认控制面板上对它们进行分组，也可以创建自己的自定义控制面板。您必须先创建集群，然后才能创建控制面板或路由控制。ARC 中的每个集群都是一个由五个端点组成的数据平面 AWS 区域。

创建路由控制和路由控制运行状况检查后，您可以为路由控制创建安全规则，以帮助防止意外的恢复自动化副作用。您可以使用或 API 操作（推荐）或使用，更新路由控制状态以单独 AWS CLI 或批量重新路由流量。 AWS Management Console

本节介绍路由控制的工作原理，以及如何创建和使用它们为应用程序重新路由流量。

> **Important**
>
> 要了解如何准备使用 ARC 重新路由流量，以此作为应用程序在灾难情况下的故障转移计划的一部分，请参阅ARC 中路由控制的最佳实践。

关于路由控制

路由控制通过使用 Amazon Route 53 中的运行状况检查来重定向流量，这些检查配置了与恢复组中单元格的顶级资源（例如 Elastic Load Balancing 负载均衡器）关联的 DNS 记录。例如，您可以将流量从一个单元格重定向到另一个单元格，方法是将一个路由控制状态更新为 Off（以停止流向一个单元格的流量），并将另一个路由控制状态更新为 On（以启动流向另一个单元格的流量）。更改流量流的过程是与路由控制关联的 Route 53 运行状况检查，此后 ARC 会根据相应的路由控制状态对其进行更新以将其设置为健康或不健康。

路由控制支持在任何具有 DNS 端点的 AWS 服务之间进行故障转移。您可以更新路由控制状态，以便在灾难恢复情况下、检测到应用程序延迟衰退或其它问题时对流量进行灾难恢复。

您还可以为路由控制配置安全规则，以确保使用路由控制重新路由流量不会影响可用性。有关更多信息，请参阅 为路径控制创建安全规则 。

请务必注意，路由控制本身并不是监控端点底层运行状况的运行状况检查。例如，与 Route 53 运行状况检查不同，路由控制不会监控响应时间或 TCP 连接时间。路由控制是一个控制运行状况检查的简单开关机构。通常，您会更改其状态以重定向流量，而这种状态更改会将流量转移到整个应用程序堆栈的特定端点，或者阻止路由到整个应用程序堆栈。例如，在一个简单的场景中，当您将路由控制状态从 On 更改为 Off 时，它会更新 Route 53 运行状况检查，该检查已与 DNS 故障转移记录相关联，以将流量移出端点。

如何使用路由控制

要更新路由控制状态以便重新路由流量，必须连接到 ARC 中的一个集群终端节点。如果您尝试连接的端点不可用，请尝试使用其他集群端点更改状态。在更改路由控制状态的过程中，应准备好轮流尝试每个端点，因为集群端点会在可用和不可用状态之间循环，以便定期维护和更新。

创建路由控制时，您可以配置 DNS 记录，将路由控制运行状况检查与每个应用程序副本前面的 Route 53 DNS 名称相关联。例如，要控制两个负载均衡器（两个区域中各有一个）之间的流量失效转移，您可以创建两个路由控制运行状况检查，并将它们与两个 DNS 记录相关联，例如失效转移路由策略中的别名记录，其中包含各自负载均衡器的域名。

您还可以使用 ARC 路由控制以及 Route 53 运行状况检查和 DNS 记录集，使用带有加权路由策略的 DNS 记录，来设置更复杂的流量故障转移方案。要查看详细示例，请参阅以下 AWS 博客文章中有关用户流量故障转移的部分：使用 Amazon 应用程序恢复控制器 (ARC) 构建高弹性应用程序，第 2 部分：多区域堆栈

当您为 AWS 区域 正在使用的路由控制启动故障转移时，由于流量涉及的步骤，您可能不会看到流量立即流出该区域。该地区现有的、正在进行的连接也可能需要很短的时间才能完成，具体取决于客户端行为和连接重复使用情况。根据您的 DNS 设置和其他因素，现有连接可能只需几分钟即可完成，也可能需要更长时间。有关更多信息，请参阅确保交通转移快速完成。

路由控制的好处

与使用传统运行状况检查重新路由流量相比，ARC 中的路由控制有几个好处。例如：

- 路由控制为您提供了一种对整个应用程序堆栈进行失效转移的方法。这与基于资源级运行状况检查的 Amazon EC2 实例对堆栈的各个组件进行故障切换形成鲜明对比。
- 路由控制为您提供了一个安全、简单的手动覆盖机制，您可以用来转移流量以进行维护工作，或者在内部监控器未检测到问题时从故障中恢复。
- 您可以将路由控制与安全规则结合使用，以防止基于运行状况检查的全自动化机制可能产生的常见副作用，例如失效转移到尚未做好失效转移准备的备用基础设施。

以下是将路由控制纳入故障转移策略的示例，以提高中应用程序的弹性和可用性 AWS。

您可以 AWS 通过跨区域运行多个（通常是三个）冗余副本来支持高可用性 AWS 应用程序。之后，您可以使用 Amazon Route 53 路由控制将流量路由到适当的副本。

例如，您可以将一个应用程序副本设置为活动状态并提供应用程序流量，而另一个则设置为备用副本。当活动副本出现故障时，您可以将用户流量重新路由到备用副本，以恢复应用程序的可用性。您应该根据来自监控和运行状况检查系统的信息来决定是从副本转移还是故障转移到副本。

如果您想更快地恢复，可以选择另一个架构选项，即主动-主动实现。使用这种方法，您的副本可以同时处于活动状态。这意味着，只要将流量重新路由到另一个活动副本，就可以将用户从受损的应用程序副本中移开，从而从故障中恢复。

AWS 用于路由控制的区域可用性

有关亚马逊应用程序恢复控制器 (ARC) 的区域支持和服务终端节点的详细信息，请参阅《亚马逊网络服务通用参考》中的亚马逊应用程序恢复控制器 (ARC) 终端节点和配额。

> **Note**
>
> Amazon 应用程序恢复控制器 (ARC) 中的路由控制是一项全球功能。但是，您必须在区域 ARC AWS CLI 命令中指定美国西部（俄勒冈`--region us-west-2`）区域（指定参数）。也就是说，当您创建诸如群集、控制面板或路由控件之类的资源时。

ARC 路由控制是一种 on/off 交换机，用于更改 ARC 运行状况检查的状态，然后可以将其与重定向流量的 DNS 记录相关联，例如，将流量从主部署副本重定向到备用部署副本。

如果应用程序出现故障或延迟问题，您可以更新路由控制状态以转移流量，例如将流量从主副本转移到备用副本。通过使用高度可靠的 ARC 数据平面 API 操作进行路由控制查询和路由控制状态更新，您可以依靠 ARC 在灾难恢复场景中进行故障转移。有关更多信息，请参阅 使用 ARC API 获取和更新路由控制状态（推荐）。

ARC 在集群中维护路由控制状态，集群由五个冗余区域端点组成。ARC 将路由控制状态更改传播到位于 Amazon EC2 队列中的集群，以获得跨五个区域的法定人数。 AWS 传播后，当您使用 API 和高度可靠的数据平面查询 ARC 以获取路由控制状态时，它会返回共识视图。

您可以与五个集群端点中的任何一个进行交互，以更新路由控制状态，例如从 `Off` 更新为 `On`。然后，ARC 将更新传播到集群的五个区域。

所有五个集群端点平均在 5 秒内实现数据一致性，最多不超过 15 秒。

ARC 的数据平面提供了极高的可靠性，可让您跨单元手动对应用程序进行故障切换。ARC 确保您始终可以访问五个集群终端节点中至少有三个来执行路由控制状态更改。请注意，每个 ARC 集群都是单租户的，以确保您不会受到 “嘈杂的邻居” 的影响，这可能会降低您的访问模式。

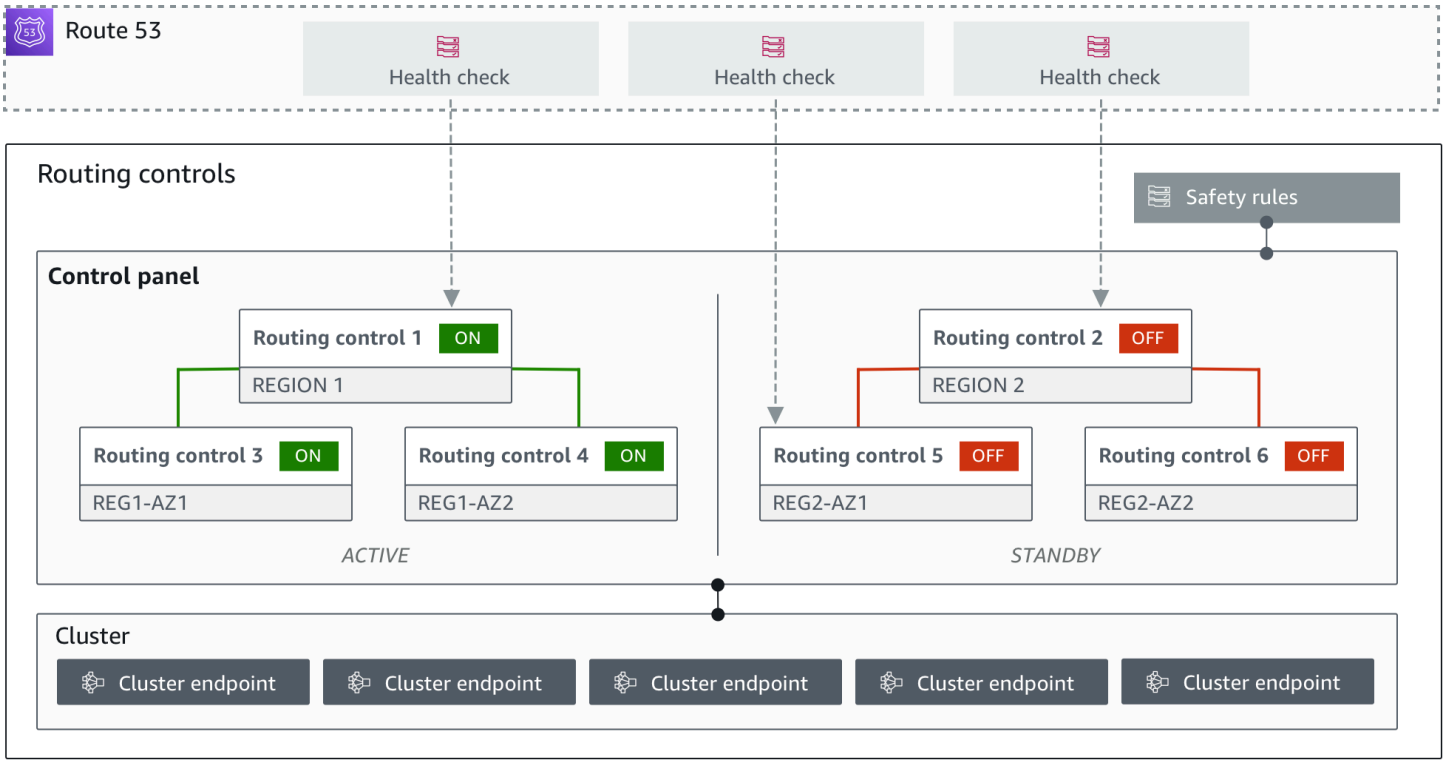
更改路由控制状态时，需要遵循以下三个极不可能失效的标准：

- 五个端点中至少有三个可用并参与仲裁。
- 您具备有效的 IAM 凭证，并且可以在工作的区域集群端点上进行身份验证。
- Route 53 数据面板运行正常（此数据面板旨在满足 100% 可用性 SLA）。

路由控制组件

下图说明了支持 ARC 中路由控制功能的组件示例。此处显示的路由控制（组合到一个控制面板）允许您管理两个区域中每个区域的两个可用区的流量。当您更新路由控制状态时，ARC 会更改 Amazon

Route 53 中的运行状况检查，从而将 DNS 流量重定向到不同的信元。您为路由控制配置的安全规则有助于避免打开失败的情况和其他意外的后果。



以下是 ARC 中路由控制功能的组件。

集群

集群是一组由五个冗余区域端点组成的集合，您可以在这些端点上执行 API 调用，以更新或获取路由控制状态。集群包括默认控制面板，您可以在一个集群上托管多个控制面板和路由控制。

路由控制

路由控制是托管在集群上的简单 on/off 交换机，用于控制进出小区的客户端流量的路由。在创建路由控制时，您可以在 Route 53 中添加 ARC 运行状况检查。这使您能够在更新 ARC 中的路由控制状态时重新路由流量（使用运行状况检查，为您的应用程序配置 DNS 记录）。

路由控制运行状况检查

路由控制与 Route 53 中的运行状况检查相集成。运行状况检查与每个应用程序副本前面的 DNS 记录相关联，例如失效转移记录。当您更改路由控制状态时，ARC 会更新相应的运行状况检查，这些检查将流量重定向到备用副本，例如故障转移到备用副本。

控制面板

控制面板将一组相关的路由控制聚合在一起。您可以将多个路由控制与一个控制面板相关联，然后为控制面板创建安全规则，以确保进行的流量重定向更新是安全的。例如，您可以为每个可用区中的每个负载均衡器配置一个路由控制，然后将它们组合到同一个控制面板中。然后，您可以添加安全规则（“断言规则”），确保在任何时候至少有一个可用区（由路由控制表示）处于活动状态，以避免出现意外的“打开失败”情况。

默认控制面板

创建集群时，ARC 会创建默认控制面板。默认情况下，您在集群上创建的所有路由控制都将添加到默认控制面板中。您也可以创建自己的控制面板来组合相关的路由控制。

安全规则

安全规则是您在路由控制中添加的规则，用于确保恢复操作不会意外影响应用程序的可用性。例如，您可以创建一个安全规则，该规则创建一个路由控制作为整体的“开关”，以便您可以启用或禁用一组其他路由控制。

端点（集群端点）

ARC 中的每个集群都有五个区域终端节点，可用于设置和检索路由控制状态。访问端点的过程应假设 ARC 定期启动和关闭端点以进行维护，因此您应该连续尝试每个端点，直到连接到一个端点。您可以访问端点以获取路由控制的当前状态（开或关），并通过更改路由控制状态来触发应用程序的失效转移。

用于路由控制的数据和控制平面

在规划故障转移和灾难恢复时，请考虑故障转移机制的弹性。我们建议您确保在故障转移期间所依赖的机制具有高可用性，以便在灾难情况下可以根据需要使用它们。通常，应尽可能为机制使用数据平面函数，以获得最大的可靠性和容错性。考虑到这一点，请务必了解服务的功能如何在控制面板和数据面板之间划分，以及何时可以依赖服务的数据面板可预期的极高可靠性。

与大多数 AWS 服务一样，控制平面和数据平面支持路由控制功能。虽然这两者都是为了可靠而构建的，但控制平面针对数据一致性进行了优化，而数据平面则针对可用性进行了优化。数据面板专为弹性而设计，因此即使在中断事件期间，当控制面板可能不可用时，它也能保持可用性。

一般而言，控制面板允许您执行基本的管理功能，例如在服务中创建、更新和删除资源。数据面板提供服务的核心功能。因此，我们建议您在可用性很重要的情况下使用数据面板操作，例如，在中断期间需要将流量重新路由到备用副本时。

对于路由控制，控制平面和数据平面按以下方式划分：

- 用于路由控制的控制平面 API 是恢复控制配置 API，在美国西部（俄勒冈）区域 (us-west-2) 中支持。您可以使用这些 API 操作或创建或删除集群、控制面板和路由控件，以帮助为可能需要为应用程序重新路由流量时发生的灾难恢复事件做好准备。 AWS Management Console 路由控制配置控制面板不是高度可用的。
- 路由控制数据平面是一个横跨五个地理 AWS 隔离区域的专用集群。每个客户都使用路由控制控制面板创建一个或多个集群。该集群托管控制面板和路由控制。然后，当您想要为应用程序重新路由流量时，可使用路由控制（恢复集群）API 获取、列出和更新路由控制状态。路由控制数据面板是高度可用的。

由于路由控制数据平面高度可用，因此我们建议您计划在 AWS Command Line Interface 要进行故障切换以从事件中恢复时，使用进行 API 调用以处理路由控制状态。有关使用路由控制准备和完成恢复操作时的关键注意事项的更多信息，请参阅ARC 中路由控制的最佳实践。

有关数据平面、控制平面以及如何 AWS 构建服务以满足高可用性目标的更多信息，请参阅 Amazon Builders Library 中的 “使用可用区的静态稳定性” 论文。

在 Amazon 应用程序恢复控制器 (ARC) 中为路由控制添加标签

标签是您用来识别和组织 AWS 资源的单词或短语（元数据）。您可以向每个资源添加多个标签，并且每个标签都包含您定义的一个键和一个值。例如，键可能是环境，值可能是生产。您可以根据添加的标签搜索和筛选您的资源。

您可以在 ARC 的路由控制中标记以下资源：

- 集群
- 控制面板
- 安全规则

ARC 中的标签只能通过 API 使用，例如，使用。 AWS CLI

以下是使用在路由控制中进行标记的示例。 AWS CLI

```
aws route53-recovery-control-config --region us-west-2 create-cluster --
cluster-name example1-cluster --tags Region=PDX,Stage=Prod
```

```
aws route53-recovery-control-config --region us-west-2 create-control-panel
--control-panel-name example1-control-panel --cluster-arn arn:aws:route53-
```

```
recovery-control::111122223333:cluster/5678abcd-abcd-5678-abcd-5678abcdefgh
--tags Region=PDX,Stage=Prod
```

有关更多信息，请参阅 TagResourceAmazon 应用程序恢复控制器 (ARC) 的恢复控制配置 API 参考指南。

ARC 中路由控制的定价

对于 ARC 中的路由控制，您需要为创建的每个集群支付每小时费用。每个集群可以托管多个路由控制，您可以使用这些控制来触发应用程序失效转移。

为了帮助管理成本和提高效率，您可以为集群设置跨账户共享，将一个集群与多个 AWS 账户共享。有关更多信息，请参阅 Support 在 ARC 中支持跨账户集群。

有关 ARC 的详细定价信息和定价示例，请参阅 ARC 定价。

开始使用 Amazon 应用程序恢复控制器 (ARC) 中的多区域恢复

要使用 Amazon 应用程序恢复控制器 (ARC) 中的路由控制对应用程序进行故障切换，您的 AWS 应用程序必须是多个应用程序 AWS 区域。首先，请确保您的应用程序设置在每个区域的孤立副本中，这样您就可以在活动期间从一个区域故障转移到另一个区域。然后，您可以创建路由控件来重新路由应用程序流量，使其从主应用程序故障转移到辅助应用程序，从而保持用户的连续性。

> **Note**
>
> 如果您的应用程序被可用区隔开，请考虑使用区域转移或区域自动切换进行故障转移恢复。无需进行任何设置即可使用区域切换或区域自动切换来可靠地从可用区损坏中恢复应用程序。有关更多信息，请参阅 使用分区移位和分区自动移位来恢复 ARC 中的应用程序。

为了使您可以使用 ARC 路由控制在事件期间恢复应用程序，我们建议您至少设置两个相互复制的应用程序。每个副本或单元格代表一个 AWS 区域。将应用程序资源设置为与区域保持一致后，请执行以下步骤，确保您的应用程序已设置为成功恢复。

提示：为了帮助简化设置，我们提供 AWS CloudFormation 了 HashiCorp Terraform 模板，用于创建具有相互独立失败的冗余副本的应用程序。要了解更多信息并下载模板，请参阅设置示例应用程序。

要准备使用路由控制，请执行以下操作，确保您的应用程序设置为具有弹性：

1. 构建应用程序堆栈（网络和计算层）的独立副本，这些副本是每个区域中彼此的副本，以便在发生事件时可以将流量从一个区域故障转移到另一个区域。确保您的应用程序代码中没有任何会导致一

个副本失败影响另一个副本的跨区域依赖关系。要在两者之间成功进行故障转移 AWS 区域，您的堆栈边界应位于一个区域内。

2. 在副本中复制应用程序所需的所有状态数据。您可以使用 AWS 数据库服务来帮助复制数据。

开始使用流量故障转移的路由控制

Amazon Application Recovery Controller (ARC) 中的路由控制使您可以触发故障转移，让流量在单独 AWS 区域运行的冗余应用程序副本或副本之间进行故障转移。故障转移是使用 Amazon Route 53 数据平面通过 DNS 执行的。

在每个区域设置副本后（如下一节所述），您可以将每个副本与路由控制相关联。首先，将路由控制与每个区域中副本的顶级域名相关联。然后，向路由控制添加路由控制运行状况检查，使其可以开启和关闭流量。这使您能够控制应用程序副本之间的流量路由。

您可以更新中的路由控制状态 AWS Management Console 以故障转移流量，但我们建议您改用 ARC 操作、API 或 AWS CLI，来更改它们。API 操作不依赖于控制台，因此它们更具弹性。

例如，要在区域之间进行故障转移，从 us-west-1 到 us-east-1，您可以`update-routing-control-state`使用 API 操作将状态设置为和到。`us-west-1` `Off` `us-east-1` `On`

在创建路由控制组件来为应用程序设置故障转移之前，请确保您的应用程序孤立到区域副本中，以便您可以从一个副本故障转移到另一个副本。要了解更多信息并开始孤立新应用程序或创建示例堆栈，请参阅下一节。

设置示例应用程序

为了帮助您了解路由控制的工作原理，我们提供了一个名为的示例应用程序`TicTacToe`。该示例使用 AWS CloudFormation 模板来简化流程，并使用可下载的 AWS CloudFormation 模板让您可以自己快速探索如何设置和使用 ARC。

部署示例应用程序后，您可以使用模板创建 ARC 组件，然后使用路由控制来管理通往该应用程序的流量。您可以根据自己的场景和应用程序调整模板和流程。

要开始使用示例应用程序和 AWS CloudFormation 模板，请参阅 ARC GitHub 存储库中的自述文件说明。您可以通过阅读 AWS CloudFormation 用户指南中的AWS CloudFormation 概念来了解有关使用 AWS CloudFormation 模板的更多信息。

ARC 中路由控制的最佳实践

对于在 ARC 中进行路由控制的恢复和故障切换准备，我们推荐以下最佳实践。

主题

确保专门构建、使用寿命长的 AWS 凭证安全且始终可访问

在灾难恢复 (DR) 场景中，通过使用一种简单的方法来访问 AWS 和执行恢复任务，将系统依赖性降至最低。专为 DR 任务创建 IAM 长效凭证，并将凭证安全地保存在本地物理保险箱或虚拟保管库中，以便在需要时进行访问。借助 IAM，您可以集中管理安全证书，例如访问密钥和 AWS 资源访问权限。对于非 DR 任务，我们建议您继续使用 AWS 单点登录等 AWS 服务进行联合访问。

要使用恢复集群数据平面 API 在 ARC 中执行故障转移任务，您可以将 ARC IAM 策略附加到您的用户。要了解更多信息，请参阅Amazon 应用程序恢复控制器 (ARC) 中基于身份的策略示例。

为故障转移中涉及的 DNS 记录选择较低的 TTL 值

对于在失效转移机制中可能需要更改的 DNS 记录，尤其是经过运行状况检查的记录，使用较低的 TTL 值是合适的做法。在这种情况下，通常选择将 TTL 设置为 60 秒或 120 秒。

DNS TTL（生存时间）设置会告诉 DNS 解析器在一条记录缓存多长时间后再请求新记录。选择 TTL 时，要在延迟和可靠性与应变能力之间进行权衡。如果记录的 TTL 较短，DNS 解析器将更快地注意到记录的更新，因为 TTL 指定了它们必须更频繁地查询。

有关更多信息，请参阅 Amazon Route 53 DNS 最佳实践中的为 DNS 记录选择 TTL 值。

限制客户端与您的终端保持连接的时间

当您使用路由控制从一个路由控制切换 AWS 区域 到另一个时，Amazon 应用程序恢复控制器 (ARC) 用来移动应用程序流量的机制是 DNS 更新。此更新使所有新连接都被定向到远离受损位置。

但是，在客户端重新连接之前，具有已打开连接的客户端可能会继续向受损位置发出请求。为确保快速恢复，我们建议您限制客户端与您的终端保持连接的时间。

如果您使用 Application Load Balancer，则可以使用该`keepalive`选项来配置连接的持续时间。有关更多信息，请参阅《Application Load Balancer 用户指南》中的 HTTP 客户端保持连接时长。

默认情况下，应用程序负载均衡器将 HTTP 客户端 keepalive 持续时间值设置为 3600 秒或 1 小时。我们建议您降低该值，使其与应用程序的恢复时间目标保持一致，例如 300 秒。选择 HTTP 客户端 keepalive 持续时间时，请考虑此值是在更频繁地重新连接（这可能会影响延迟）和更快地将所有客户端从受损的可用区或区域移出受损的可用区或区域之间进行权衡。

为您的五个区域集群终端节点和路由控制添加书签或硬编码 ARNs

我们建议您将 ARC 区域集群终端节点的本地副本保存在书签中，或者保存在用于重试终端节点的自动化代码中。在发生故障事件期间，您可能无法访问某些 API 操作，包括未托管在极其可靠的数据平面集群上的 ARC API 操作。您可以使用 DescribeClusterAPI 操作列出 ARC 集群的终端节点。

随机选择一个终端节点来更新您的路由控制状态

路由控制提供五个区域端点，即使在处理故障时也能确保高可用性。为了实现其完全的弹性，重要的是要有可以根据需要使用所有五个端点的重试逻辑。有关在 AWS SDK 中使用代码示例的信息，包括试用集群终端节点的示例，请参阅应用程序恢复控制器的代码示例 AWS SDKs。

使用极其可靠的数据平面 API 来列出和更新路由控制状态，而不是使用控制台

使用 ARC 数据平面 API，查看ListRoutingControls操作中的路由控制和状态，并更新路由控制状态以重定向流量，以便在UpdateRoutingControlState操作中进行故障转移。您可以使用 AWS CLI（如这些示例所示）或使用其中一个编写的代码 AWS SDKs。ARC 通过数据平面中的 API 提供极高的可靠性，可对流量进行故障切换。我们建议使用 API，而不是在 AWS Management Console中更改路由控制状态。

连接您的一个区域集群终端节点，让 ARC 使用数据平面 API。如果端点不可用，请尝试连接到另一个集群端点。

如果安全规则阻止路由控制状态更新，则可以绕过该规则进行更新并对流量进行失效转移。有关更多信息，请参阅 覆盖安全规则以重新路由流量。

使用 ARC 测试故障转移

使用 ARC 路由控制定期测试故障转移，以便从主应用程序堆栈故障转移到辅助应用程序堆栈。重要的是要确保您添加的 ARC 结构与堆栈中的正确资源保持一致，并且一切都按预期运行。您应该在为环境设置 ARC 之后对此进行测试，并继续定期进行测试，以便您的故障转移环境准备就绪，以免出现故障情况，因为您需要辅助系统快速启动并运行，以避免用户停机。

路由控制 API 操作

本节包括列出可用于在 Amazon 应用程序恢复控制器 (ARC) 中设置和使用路由控制的 API 操作的表格，以及相关文档的链接。

有关如何使用常见路由控制配置 API 操作的示例 AWS Command Line Interface，请参阅 使用 ARC 路由控制 API 操作的示例 AWS CLI。

下表列出了可用于路由控制配置的 ARC API 操作以及相关文档的链接。

操作	使用 ARC 控制台	使用 ARC API
创建集群	请参阅 在 ARC 中创建路由控制组件。	请参阅 CreateCluster
描述集群	请参阅 在 ARC 中创建路由控制组件。	请参阅 DescribeCluster
删除集群	请参阅 在 ARC 中创建路由控制组件。	请参阅 DeleteCluster
列出账户的集群	请参阅 在 ARC 中创建路由控制组件。	请参阅 ListClusters
创建路由控制	请参阅 在 ARC 中创建路由控制组件。	请参阅 CreateRoutingControl
描述路由控制	请参阅 在 ARC 中创建路由控制组件。	请参阅 DescribeRoutingControl
更新路由控制	请参阅 在 ARC 中创建路由控制组件。	请参阅 UpdateRoutingControl
删除路由控制	请参阅 在 ARC 中创建路由控制组件。	请参阅 DeleteRoutingControl
列出路由控制	请参阅 在 ARC 中创建路由控制组件。	请参阅 ListRoutingControls
创建控制面板	请参阅 在 ARC 中创建路由控制组件。	请参阅 CreateControlPanel
描述控制面板	请参阅 在 ARC 中创建路由控制组件。	请参阅 DescribeControlPanel

操作	使用 ARC 控制台	使用 ARC API
更新控制面板	请参阅 在 ARC 中创建路由控制组件 。	请参阅 UpdateControlPanel
删除控制面板	请参阅 在 ARC 中创建路由控制组件 。	请参阅 DeleteControlPanel
列出控制面板	请参阅 在 ARC 中创建路由控制组件 。	请参阅 ListControlPanels
创建安全规则	请参阅 为路径控制创建安全规则 。	请参阅 CreateSafetyRule
描述安全规则	请参阅 为路径控制创建安全规则 。	请参阅 DescribeSafetyRule
更新安全规则	请参阅 为路径控制创建安全规则 。	请参阅 UpdateSafetyRule
删除安全规则	请参阅 为路径控制创建安全规则 。	请参阅 DeleteSafetyRule
列出安全规则	请参阅 为路径控制创建安全规则 。	请参阅 ListSafetyRules
列出关联的 Route 53 运行状况检查	请参阅 在 ARC 中创建路由控制运行状况检查 。	见 ListAssociatedRoute53 HealthChecks
列出用于群集共享的 AWS RAM 资源策略	请参阅 Support 在 ARC 中支持跨账户集群。	见 GetResourcePolicy

下表列出了可用于通过路由控制数据平面管理流量故障转移的常见 ARC API 操作，以及相关文档的链接。

操作	使用 ARC 控制台	使用 ARC API
获取路由控制状态	请参阅 获取和更新中的路由控制状态 AWS Management Console。	请参阅 GetRoutingControlState
列出路由控制	不适用	请参阅 ListRoutingControls。
更新路由控制状态	请参阅 获取和更新中的路由控制状态 AWS Management Console。	请参阅 UpdateRoutingControlState
更新多个路由控制状态	请参阅 获取和更新中的路由控制状态 AWS Management Console。	请参阅 UpdateRoutingControlStates

将此服务与 AWS SDK 配合使用

AWS 软件开发套件 (SDKs) 可用于许多流行的编程语言。每个软件开发工具包都提供 API、代码示例和文档，使开发人员能够更轻松地以其首选语言构建应用程序。

SDK 文档	代码示例
适用于 C++ 的 AWS SDK	适用于 C++ 的 AWS SDK 代码示例
AWS CLI	AWS CLI 代码示例
适用于 Go 的 AWS SDK	适用于 Go 的 AWS SDK 代码示例
适用于 Java 的 AWS SDK	适用于 Java 的 AWS SDK 代码示例
适用于 JavaScript 的 AWS SDK	适用于 JavaScript 的 AWS SDK 代码示例
适用于 Kotlin 的 AWS SDK	适用于 Kotlin 的 AWS SDK 代码示例
适用于 .NET 的 AWS SDK	适用于 .NET 的 AWS SDK 代码示例
适用于 PHP 的 AWS SDK	适用于 PHP 的 AWS SDK 代码示例

SDK 文档	代码示例
AWS Tools for PowerShell	AWS Tools for PowerShell 代码示例
适用于 Python (Boto3) 的 AWS SDK	适用于 Python (Boto3) 的 AWS SDK 代码示例
适用于 Ruby 的 AWS SDK	适用于 Ruby 的 AWS SDK 代码示例
AWS SDK for Rust	AWS SDK for Rust 代码示例
适用于 SAP ABAP 的 AWS SDK	适用于 SAP ABAP 的 AWS SDK 代码示例
AWS SDK for Swift	AWS SDK for Swift 代码示例

有关特定于此服务的示例，请参阅应用程序恢复控制器的代码示例 AWS SDKs。

> **示例可用性**
>
> 找不到所需的内容？ 通过使用此页面底部的提供反馈链接请求代码示例。

使用 ARC 路由控制 API 操作的示例 AWS CLI

本节介绍使用路由控制的简单应用示例，使用使用 API 操作 AWS Command Line Interface 与 Amazon 应用程序恢复控制器 (ARC) 中的路由控制功能配合使用。这些示例旨在帮助您基本了解如何使用 CLI 进行路由控制。

借助 Amazon Application Recovery Controller (ARC) 中的路由控制，您可以在独立 AWS 区域 或可用区域中运行的冗余应用程序副本或副本之间触发流量故障转移。

您可以将路由控制组织成在集群上配置的名为控制面板的组。ARC 集群是一组在全球部署的区域终端节点。集群端点提供了一个高度可用的 API，可用于设置和检索路由控制状态。有关路由控制功能组件的更多信息，请参阅路由控制组件 。

> **Note**
>
> ARC 是一项支持多个端点的全球服务 AWS 区域。但是，您必须在大多数 ARC CLI 命令中指定美国西部（俄勒冈）区域，即指定参数`--region us-west-2`。例如，在创建恢复组、控制面板和群集时使用`region`参数。

创建集群时，ARC 会为您提供一组区域终端节点。要获取或更新路由控制状态，必须在 CLI 命令中指定区域终端节点（ AWS 区域 和终端节点 URL ）。

有关使用的更多信息 AWS CLI，请参阅 AWS CLI 命令参考。有关路由控制 API 操作的列表，请参阅路由控制 API 操作和 路由控制 API 操作。

首先，我们将使用路由控制创建管理故障转移所需的组件，首先是创建集群。

设置路由控制组件

第一步是创建集群。ARC 集群由五个终端节点组成，五个不同的终端节点各一个 AWS 区域。ARC 基础设施支持这些端点协同工作，从而保证故障转移操作的高可用性和顺序一致性。

1. 创建集群

1a. 创建集群。network-type是可选的，可以是IPV4或DUALSTACK。默认值为 IPV4。

```
aws route53-recovery-control-config create-cluster --cluster-name test --network-type
 DUALSTACK
```

```
"Cluster": {
    "ClusterArn": "arn:aws:route53-recovery-
control::123456789123:cluster/12341234-1234-1234-1234-123412341234",
    "Name": "test",
    "Status": "PENDING",
    "Owner": "123456789123",
    "NetworkType": "DUALSTACK"
}
```

首次创建 ARC 资源时，在创建集群PENDING时，其状态为。您可以通过调用 describe-cluster 查看其进度。

1b. 描述集群。

```
aws route53-recovery-control-config --region us-west-2 \
    describe-cluster --cluster-arn arn:aws:route53-recovery-
control::111122223333:cluster/5678abcd-abcd-5678-abcd-5678abcdefgh
```

```
"Cluster": {
```

```
    "ClusterArn": "arn:aws:route53-recovery-
control::123456789123:cluster/12341234-1234-1234-1234-123412341234",
    "Name": "test",
    "Status": "DEPLOYED",
    "Owner": "123456789123",
    "NetworkType": "DUALSTACK"
}
```

当状态为 DEPLOYED 时，ARC 已成功创建集群，其中包含一组供您与之交互的终端节点。您可以通过调用 `list-clusters` 列出所有集群。

1c. 列出您的集群。

```
aws route53-recovery-control-config --region us-west-2 list-clusters
```

```
"Cluster": {
    "ClusterArn": "arn:aws:route53-recovery-
control::123456789123:cluster/12341234-1234-1234-1234-123412341234",
    "Name": "test",
    "Status": "DEPLOYED",
    "Owner": "123456789123",
    "NetworkType": "DUALSTACK"
}
```

1d。更新集群的网络类型。选项有 IPV4 或 DUALSTACK。

```
aws route53-recovery-control-config update-cluster \
--cluster-arn arn:aws:route53-recovery-
control::123456789123:cluster/12341234-1234-1234-1234-123412341234 \
--network-type DUALSTACK
```

```
"Cluster": {
    "ClusterArn": "arn:aws:route53-recovery-
control::123456789123:cluster/12341234-1234-1234-1234-123412341234",
    "Name": "test",
    "Status": "PENDING",
    "Owner": "123456789123",
   "NetworkType": "DUALSTACK"
}
```

2. 创建控制面板

控制面板是用于组织 ARC 路由控件的逻辑分组。当您创建集群时，ARC 会自动为您调用提供一个控制面板DefaultControlPanel。您可以立即使用该控制面板。

一个控制面板只能存在于一个集群中。如果要将控制面板移到另一个集群，则必须将其删除，然后在第二个集群中创建它。您可以通过调用 list-control-panels 查看账户中的所有控制面板。要仅查看特定集群中的控制面板，请添加 --cluster-arn 字段。

2a. 列出控制面板。

```
aws route53-recovery-control-config --region us-west-2 \
    list-control-panels --cluster-arn arn:aws:route53-recovery-
control::111122223333:cluster/eba23304-1a51-4674-ae32-b4cf06070bdd
```

```
{
    "ControlPanels": [
        {
            "ControlPanelArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/1234567dddddd1234567dddddd1234567",
            "ClusterArn": "arn:aws:route53-recovery-
control::111122223333:cluster/5678abcd-abcd-5678-abcd-5678abcdefgh",
            "DefaultControlPanel": true,
            "Name": "DefaultControlPanel",
            "RoutingControlCount": 0,
            "Status": "DEPLOYED"
        }
    ]
}
```

也可以选择通过调用 create-control-panel 创建自己的控制面板。

2b. 创建控制面板。

```
aws route53-recovery-control-config --region us-west-2 create-control-panel \
        --control-panel-name NewControlPanel2 \
        --cluster-arn arn:aws:route53-recovery-control::111122223333:cluster/5678abcd-
abcd-5678-abcd-5678abcdefgh
```

```
{
```

```
    "ControlPanel": {
        "ControlPanelArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456",
        "ClusterArn": "arn:aws:route53-recovery-control::111122223333:cluster/5678abcd-
abcd-5678-abcd-5678abcdefgh",
        "DefaultControlPanel": false,
        "Name": "NewControlPanel2",
        "RoutingControlCount": 0,
        "Status": "PENDING"
    }
}
```

首次创建 ARC 资源时，其状态为创建PENDING时。您可以通过调用 `describe-control-panel` 查看进度。

2c. 描述控制面板。

```
aws route53-recovery-control-config --region us-west-2 describe-control-panel \
    --control-panel-arn arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456
```

```
{
    "ControlPanel": {
        "ControlPanelArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456",
        "ClusterArn": "arn:aws:route53-recovery-control::111122223333:cluster/5678abcd-
abcd-5678-abcd-5678abcdefgh",
        "DefaultControlPanel": true,
        "Name": "DefaultControlPanel",
        "RoutingControlCount": 0,
        "Status": "DEPLOYED"
    }
}
```

3. 创建路由控制

现在您已设置集群并查看控制面板，接着可以开始创建路由控制。创建路由控制时，您必须至少指定路由控制所在集群的 Amazon 资源名称 (ARN)。您也可以为路由控制指定控制面板的 ARN。您还需要指定控制面板所在的集群。

如果您未指定控制面板，路由控制将添加到自动创建的控制面板 `DefaultControlPanel`。

通过调用 create-routing-control 创建路由控制。

3a. 创建路由控制。

```
aws route53-recovery-control-config --region us-west-2 create-routing-control \
    --routing-control-name NewRc1 \
    --cluster-arn arn:aws:route53-recovery-control::111122223333:cluster/5678abcd-
abcd-5678-abcd-5678abcdefgh
```

```
{
    "RoutingControl": {
        "ControlPanelArn": " arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456",
        "Name": "NewRc1",
        "RoutingControlArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
abcdefg1234567",
        "Status": "PENDING"
    }
}
```

路由控件遵循与其他 ARC 资源相同的创建模式，因此您可以通过调用描述操作来跟踪它们的进度。

3b. 描述路由控制。

```
aws route53-recovery-control-config --region us-west-2 describe-routing-control \
       --routing-control-arn arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
abcdefg1234567
```

```
{
    "RoutingControl": {
        "ControlPanelArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456",
        "Name": "NewRc1",
        "RoutingControlArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
abcdefg1234567",
        "Status": "DEPLOYED"
    }
}
```

您可以通过调用 `list-routing-controls` 列出控制面板中的路由控制。控制面板 ARN 为必填项。

3c. 列出路由控制。

```
aws route53-recovery-control-config --region us-west-2 list-routing-controls \
        --control-panel-arn arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456
```

```
{
    "RoutingControls": [
        {
            "ControlPanelArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456",
            "Name": "Rc1",
            "RoutingControlArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
abcdefg1234567",
            "Status": "DEPLOYED"
        },
        {
            "ControlPanelArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456",
            "Name": "Rc2",
            "RoutingControlArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
hijklmnop987654321",
            "Status": "DEPLOYED"
        }
    ]
}
```

在使用路由控制状态的以下示例中，我们假设您有本节中列出的两个路由控制（Rc1 和 Rc2）。在本例中，每个路由控制代表部署了应用程序的一个可用区。

4. 创建安全规则

同时使用多个路由控制时，您可能会决定在启用和禁用它们时采取一些保障措施，以避免意想不到的后果，例如关闭两个路由控制和停止所有流量。要创建这些安全措施，您需要创建路由控制安全规则。

安全规则有两种类型：断言规则和门控规则。如需了解有关安全规则的详情，请参阅[为路径控制创建安全规则](#)。

以下调用提供了创建断言规则的示例，该规则可确保在任何给定时间至少将两个路由控制之一设置为 On。要创建规则，请运行使用 `assertion-rule` 参数的 `create-safety-rule`。

有关断言规则 API 操作的详细信息，请参阅 AssertionRuleAmazon 应用程序恢复控制器的路由控制 API 参考指南。

4a. 创建断言规则。

```
aws route53-recovery-control-config --region us-west-2 create-safety-rule \
        --assertion-rule '{"Name": "TestAssertionRule",
        "ControlPanelArn": "arn:aws:route53-recovery-
control::888888888888:controlpanel/zzz123yyy456xxx789zzz123yyy456xxx",
        "WaitPeriodMs": 5000,
        "AssertedControls":
        ["arn:aws:route53-recovery-control::888888888888:controlpanel/
zzz123yyy456xxx789zzz123yyy456xxx/routingcontrol/def123def123def"
        "arn:aws:route53-recovery-control::888888888888:controlpanel/
zzz123yyy456xxx789zzz123yyy456xxx/routingcontrol/ghi456ghi456ghi"],
        "RuleConfig": {"Threshold": 1, "Type": "ATLEAST", "Inverted": false}}'
```

```
{
    "Rule": {
        "ASSERTION": {
            "Arn": "arn:aws:route53-recovery-control::888888888888:controlpanel/
zzz123yyy456xxx789zzz123yyy456xxx/safetyrule/333333444444",
            "AssertedControls": [
                "arn:aws:route53-recovery-control::888888888888:controlpanel/
zzz123yyy456xxx789zzz123yyy456xxx/routingcontrol/def123def123def"
                "arn:aws:route53-recovery-control::888888888888:controlpanel/
zzz123yyy456xxx789zzz123yyy456xxx/routingcontrol/ghi456ghi456ghi"],
            "ControlPanelArn": "arn:aws:route53-recovery-
control::888888888888:controlpanel/zzz123yyy456xxx789zzz123yyy456xxx",
            "Name": "TestAssertionRule",
            "RuleConfig": {
                "Inverted": false,
                "Threshold": 1,
                "Type": "ATLEAST"
            },
            "Status": "PENDING",
            "WaitPeriodMs": 5000
        }
    }
```

```
}
```

以下调用提供了创建门控规则的示例，该规则为控制面板中的一组目标路由控制提供了整体的“开启/关闭”或“门控”开关。这样便可以禁止更新目标路由控制，例如，自动化机制无法进行未授权更新。在本例中，门控开关是通过 GatingControls 参数指定的路由控制，受到控制或“门控”的两个路由控制通过 TargetControls 参数指定。

> **Note**
>
> 在创建门控规则之前，必须创建门控路由控制（不包括 DNS 故障转移记录）和目标路由控制（需配置有 DNS 故障转移记录）。

要创建规则，请运行使用 gating-rule 参数的 create-safety-rule。

有关断言规则 API 操作的详细信息，请参阅 GatingRuleAmazon 应用程序恢复控制器的路由控制 API 参考指南。

4b. 创建门控规则。

```
aws route53-recovery-control-config --region us-west-2 create-safety-rule \
        --gating-rule '{"Name": "TestGatingRule",
        "ControlPanelArn": "arn:aws:route53-recovery-
control::888888888888:controlpanel/zzz123yyy456xxx789zzz123yyy456xxx",
        "WaitPeriodMs": 5000,
        "GatingControls": ["arn:aws:route53-recovery-
control::888888888888:controlpanel/zzz123yyy456xxx789zzz123yyy456xxx/routingcontrol/
def123def123def"]
        "TargetControls": ["arn:aws:route53-recovery-
control::888888888888:controlpanel/zzz123yyy456xxx789zzz123yyy456xxx/routingcontrol/
ghi456ghi456ghi",
        "arn:aws:route53-recovery-control::888888888888:controlpanel/
zzz123yyy456xxx789zzz123yyy456xxx/routingcontrol/lmn789lmn789lmn"],
        "RuleConfig": {"Threshold": 0, "Type": "OR", "Inverted": false}}'
```

```
{
    "Rule": {
        "GATING": {
            "Arn": "arn:aws:route53-recovery-control::888888888888:controlpanel/
zzz123yyy456xxx789zzz123yyy456xxx/safetyrule/444444444444",
```

```
            "GatingControls": [
                "arn:aws:route53-recovery-control::888888888888:controlpanel/
zzz123yyy456xxx789zzz123yyy456xxx/routingcontrol/def123def123def"
            ],
            "TargetControls": [
                "arn:aws:route53-recovery-control::888888888888:controlpanel/
zzz123yyy456xxx789zzz123yyy456xxx/routingcontrol/ghi456ghi456ghi"
                "arn:aws:route53-recovery-control::888888888888:controlpanel/
zzz123yyy456xxx789zzz123yyy456xxx/routingcontrol/lmn789lmn789lmn"
            ],
            "ControlPanelArn": "arn:aws:route53-recovery-
control::888888888888:controlpanel/zzz123yyy456xxx789zzz123yyy456xxx",
            "Name": "TestGatingRule",
            "RuleConfig": {
                "Inverted": false,
                "Threshold": 0,
                "Type": "OR"
            },
            "Status": "PENDING",
            "WaitPeriodMs": 5000
        }
    }
}
```

与其他路由控制资源一样，您可以在安全规则传播到数据平面后对其进行描述、列出或删除。

设置一个或多个安全规则后，您可以继续与集群交互，以设置或检索路由控制的状态。如果某项 `set-routing-control-state` 操作违反了您创建的规则，您将收到类似下方的异常：

```
Cannot modify control state for [0123456bbbbbbb0123456bbbbbb01234560123
abcdefg1234567] due to failed rule evaluation
0123456bbbbbbb0123456bbbbbb01234563333333444444
```

第一个标识符是控制面板 ARN 与路由控制 ARN 的连接体。第二个标识符是控制面板 ARN 与安全规则 ARN 的连接体。

5. 创建运行状况检查

要使用路由控制对流量进行故障转移，您可以在 Amazon Route 53 中创建运行状况检查，然后将运行状况检查与您的 DNS 记录关联起来。为了故障转移流量，ARC 路由控制会将运行状况检查设置为失败，这样 Route 53 就会重新路由流量。（运行状况检查对应用程序的运行状况无效；它只是用作重新路由流量的方法。）

例如，假设您有两个单元（区域或可用区）。您可以将一个单元配置为应用程序的主单元，将另一个配置为辅助单元，以便进行故障切换。

要为失效转移设置运行状况检查，您可以执行以下操作，例如：

1. 使用 ARC CLI 为每个信元创建路由控制。
2. 使用 Route 53 CLI 在 Route 53 中为每个路由控制创建 ARC 运行状况检查。
3. 使用 Route 53 CLI 在 Route 53 中创建两个失效转移 DNS 记录，并将运行状况检查与每个记录关联起来。

5a. 为每个单元格创建路由控制。

```
aws route53-recovery-control-config --region us-west-2 create-routing-control \
        --routing-control-name RoutingControlCell1 \
        --cluster-arn arn:aws:route53-recovery-control::111122223333:cluster/5678abcd-
abcd-5678-abcd-5678abcdefgh
```

```
aws route53-recovery-control-config --region us-west-2 create-routing-control \
        --routing-control-name RoutingControlCell2 \
        --cluster-arn arn:aws:route53-recovery-control::111122223333:cluster/5678abcd-
abcd-5678-abcd-5678abcdefgh
```

5b. 为每个路由控制创建运行状况检查。

> **Note**
>
> 您可以使用 Amazon Route 53 CLI 创建 ARC 运行状况检查。

```
aws route53 create-health-check --caller-reference RoutingControlCell1 \
        --health-check-config \
        Type=RECOVERY_CONTROL,RoutingControlArn=arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
abcdefg1234567
```

```
{
    "Location": "https://route53.amazonaws.com/2015-01-01/healthcheck/11111aaaa-bbbb-
cccc-dddd-ffffff22222",
```

```
    "HealthCheck": {
        "Id": "xxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
        "CallerReference": "RoutingControlCell1",
        "HealthCheckConfig": {
            "Type": "RECOVERY_CONTROL",
            "Inverted": false,
            "Disabled": false,
            "RoutingControlArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
abcdefg1234567"
        },
        "HealthCheckVersion": 1
    }
}
```

```
aws route53 create-health-check --caller-reference RoutingControlCell2 \
    --health-check-config \
    Type=RECOVERY_CONTROL,RoutingControlArn=arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
abcdefg1234567
```

```
{
    "Location": "https://route53.amazonaws.com/2015-01-01/healthcheck/11111aaaa-bbbb-
cccc-dddd-ffffff22222",
    "HealthCheck": {
        "Id": "xxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
        "CallerReference": "RoutingControlCell2",
        "HealthCheckConfig": {
            "Type": "RECOVERY_CONTROL",
            "Inverted": false,
            "Disabled": false,
            "RoutingControlArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
abcdefg1234567"
        },
        "HealthCheckVersion": 1
    }
}
```

5c. 创建两个失效转移 DNS 记录，并将运行状况检查与每个记录关联起来。

使用 Route 53 CLI 在 Route 53 中创建失效转移 DNS 记录。要创建记录，请按照《Amazon Route 53 AWS CLI 命令参考》中该change-resource-record-sets命令的说明进行操作。在记录中，指定每个单元格的 DNS 值以及 Route 53 为运行状况检查创建的相应 HealthCheckID 值（请参阅 6b）。

对于主单元格：

```
{
    "Name": "myapp.yourdomain.com",
    "Type": "CNAME",
    "SetIdentifier": "primary",
    "Failover": "PRIMARY",
    "TTL": 0,
    "ResourceRecords": [
        {
            "Value": "cell1.yourdomain.com"
        }
    ],
    "HealthCheckId": "xxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
}
```

对于辅助单元格：

```
{
    "Name": "myapp.yourdomain.com",
    "Type": "CNAME",
    "SetIdentifier": "secondary",
    "Failover": "SECONDARY",
    "TTL": 0,
    "ResourceRecords": [
        {
            "Value": "cell2.yourdomain.com"
        }
    ],
    "HealthCheckId": "yyyyyy-yyyy-yyyy-yyyy-yyyyyyyyyyyy"
}
```

现在，要从主单元格失效转移到辅助单元格，可以按照步骤 4b 中的 CLI 示例将 RoutingControlCell1 的状态更新为 OFF，将 RoutingControlCell2 的状态更新为 ON。

使用列出并更新路由控制和状态 AWS CLI

创建 Amazon Application Recovery Controller (ARC) 资源（例如集群、路由控制和控制面板）后，您可以与集群交互以列出和更新故障转移的路由控制状态。

对于您创建的每个集群，ARC 都会为您提供一组集群终端节点，每五个集群终端节点中各一个 AWS 区域。在调用集群以检索或将路由控制状态设置为或时，必须指定其中一个区域终端节点（ AWS 区域和终端节点 URL）`Off`。`On`当您使用 AWS CLI、获取或更新路由控制状态时，除了区域终端节点外，还必须指定区域终端节点`--region`的终端节点，如本节中的示例所示。

您可以使用任何区域集群端点。我们建议您的系统在区域终端节点之间轮换，并做好使用每个可用终端节点重试的准备。有关说明按顺序尝试集群端点的代码示例，请参阅应用程序恢复控制器使用的操作 AWS SDKs 。

有关使用的更多信息 AWS CLI，请参阅 AWS CLI 命令参考。有关路由控制 API 操作的列表以及指向更多信息的链接，请参阅 路由控制 API 操作。

> **Important**
>
> 尽管您可以在 Amazon Route 53 控制台上更新路由控制状态，但我们建议您使用 AWS CLI 或 AWS SDK 更新路由控制状态。ARC 通过 ARC 路由控制数据平面提供极高的可靠性，用于重新路由流量和跨单元进行故障转移。有关使用 ARC 进行故障转移的更多建议，请参阅ARC 中路由控制的最佳实践。

创建路由控制时，状态设置为 `Off`。这意味着流量不会路由到该路由控制的目标单元格。您可以通过运行 `get-routing-control-state` 命令验证路由控制的状态。

要确定将要指定的区域和端点，请运行 `describe-clusters` 命令以查看 `ClusterEndpoints`。每个都`ClusterEndpoint`包括一个区域和相应的终端节点，您可以使用它们来获取或更新路由控制状态。 DescribeCluster是一项恢复控制配置 API 操作。我们建议您将 ARC 区域集群终端节点的本地副本保存在书签中，或者在用于重试终端节点的自动化代码中进行硬编码。

1. 列出路由控制

您可以使用高度可靠的 ARC 数据平面端点查看您的路由控制和路由控制状态。

1. 列出特定控制面板的路由控制。如果不指定控制面板，`list-routing-controls` 会返回集群中的所有路由控制。

```
aws route53-recovery-cluster list-routing-controls --control-panel-arn \
```

```
        arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456 \
        --region us-west-2 \
        --endpoint-url https://host-dddddd.us-west-2.example.com/v1
```

```
{
    "RoutingControls": [{
        "ControlPanelArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456",
        "ControlPanelName": "ExampleControlPanel",
        "RoutingControlArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
abcdefg1234567",
        "RoutingControlName": "RCOne",
        "RoutingControlState": "On"
    },
    {
        "ControlPanelArn": "arn:aws:route53-recovery-
control::023759465626:controlpanel/0123456bbbbbbb0123456bbbbbb0123456",
        "ControlPanelName": "ExampleControlPanel",
        "RoutingControlArn": "arn:aws:route53-recovery-
control::023759465626:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
zzzzxxxxyyyy123456",
        "RoutingControlName": "RCTwo",
        "RoutingControlState": "Off"
    }
]
```

2. 获取路由控制

2. 获取路由控制状态。

```
aws route53-recovery-cluster get-routing-control-state --routing-control-arn \
        arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
abcdefg1234567 \
        --region us-west-2 \
        --endpoint-url https://host-dddddd.us-west-2.example.com/v1
```

```
{"RoutingControlArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
abcdefg1234567",
```

```
    "RoutingControlName": "RCOne",
    "RoutingControlState": "On"
}
```

2. 更新路由控制

要将流量路由到路由控制所控制的目标端点，请将路由控制状态更新为 On。通过运行 update-routing-control-state 命令更新路由控制状态。（请求成功时，响应为空。）

2a. 更新路由控制状态。

```
aws route53-recovery-cluster update-routing-control-state \
        --routing-control-arn \
        arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
abcdefg1234567 \
        --routing-control-state On \
        --region us-west-2 \
        --endpoint-url https://host-dddddd.us-west-2.example.com/v1
```

```
{}
```

您可以通过一个 API 调用同时更新多个路由控制：update-routing-control-states。（请求成功时，响应为空。）

2b. 一次更新多个路由控制状态（批量更新）。

```
aws route53-recovery-cluster update-routing-control-states \
        --update-routing-control-state-entries \
        '[{"RoutingControlArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
abcdefg1234567",
        "RoutingControlState": "Off"}, \
        {"RoutingControlArn": "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
hijklmnop987654321",
        "RoutingControlState": "On"}]' \
        --region us-west-2 \
        --endpoint-url https://host-dddddd.us-west-2.example.com/v1
```

```
{}
```

在 ARC 中使用路由控制组件

主题

在 ARC 中创建路由控制组件

本节介绍如何创建集群、路由控制、运行状况检查和控制面板，以便在 Amazon 应用程序恢复控制器 (ARC) 中使用路由控制。

首先创建一个集群，以托管您的路由控制和用于给路由控制分组的控制面板。然后创建路由控制和运行状况检查，这样就可以重新路由流量，从一个单元格失效转移到另一个单元格，使流量流向别处，例如您的备份副本。

请注意，您创建的每个集群均按小时收费。通常，您只需要一个集群来托管路由控制和控制面板，以管理应用程序的恢复控制。此外，您可以使用设置资源共享 AWS Resource Access Manager，以便一个集群可以托管路由控制和多个集群拥有的其他 ARC 资源 AWS 账户。要了解 ARC 中的资源共享，Support 在 ARC 中支持跨账户集群. 有关定价信息，请参阅亚马逊应用程序恢复控制器 (ARC) 定价并向下滚动至 Amazon Route 53。

要使用路由控制对流量进行失效转移，您需要创建路由控制运行状况检查，并将其与应用程序中资源的 Amazon Route 53 DNS 记录相关联。举个例子，假设您有两个单元格，一个配置为应用程序的主单元格，另一个配置为辅助单元格（失效转移的目的地）。

要为失效转移设置运行状况检查，请执行以下操作：

1. 为每个单元格创建路由控制。
2. 为每个路由控制创建运行状况检查。
3. 创建两个 DNS 记录（例如两个 DNS 故障转移记录），并将运行状况检查与每个记录关联起来。

当您创建的安全规则是门控规则时，也可能需要创建路由控制。在这种情况下，不要将运行状况检查和 DNS 记录与路由控制相关联，因为您要把它用作门控路由控制。有关更多信息，请参阅 为路径控制创建安全规则 。

这些部分包含在 ARC 控制台上创建路由控制组件的步骤。要了解如何在 ARC 中使用恢复控制配置 API 操作，请参阅 路由控制 API 操作。

在 ARC 中创建集群

必须在 ARC 中创建集群来托管路由控件和控制面板。

集群是一组冗余的区域端点，您可以在这些端点上执行 API 调用，以更新或获取一个或多个路由控制的状态。一个集群可以托管许多路由控制。

> **Important**
>
> 请注意，您创建的每个集群均按小时收费。一个集群可以托管许多路由控制和控制面板，通常足够用来管理应用程序的恢复控制。

创建集群

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 选择 Clusters (集群)。
3. 选择创建，然后输入集群的名称。
4. 选择创建集群。

在 ARC 中创建路由控件

为流量路由的每个目标单元格创建路由控制。例如，当您的应用程序的资源孤立以实现可恢复性时，每个应用程序可能都有一个单元，每个区域的每个 AWS 区域可用区都有嵌套单元格。在这种情况下，要为每个单元格和每个嵌套单元格创建一个路由控制。

创建路由控制时，请记住路由控制名称在每个控制面板中必须是唯一的。

创建用于重新路由流量的路由控制后，您可以将每个路由控制与运行状况检查相关联，这样您就可以根据与每个检查关联的 DNS 记录将流量路由到单元格。如果您要设置门控规则作为安全规则并创建门控路由控制，则不要向路由控制中添加运行状况检查。

创建路由控制的步骤

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 选择路由控制。

3. 在路由控制页面上，选择创建，然后选择路由控制。
4. 输入路由控制的名称，选择要添加控制的目标集群，然后选择把它添加到现有的控制面板，包括使用默认控制面板。或者创建新的控制面板。
5. 如果您选择创建新的控制面板，请选择要在上面创建控制面板的集群，然后输入面板的名称。
6. 选择创建路由控制。
7. 按照步骤命名和创建路由控制。

在 ARC 中创建路由控制运行状况检查

请将路由控制运行状况检查与要用于重新路由流量的每个路由控制关联起来。然后，为每个运行状况检查配置 Amazon Route 53 DNS 记录（例如失效转移 DNS 记录）。然后，您只需更新关联路由控制的状态，将其设置为On或Off，即可在 Amazon 应用程序恢复控制器 (ARC) 中重新路由流量。

> **Note**
>
> 您无法编辑现有的路由控制运行状况检查，将其与其他路由控制相关联。

创建路由控制运行状况检查的步骤

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 选择路由控制。
3. 在路由控制页面上，选择一个路由控制。
4. 在路由控制详细信息页面上，选择创建运行状况检查。
5. 输入运行状况检查的名称，然后选择创建。

接下来，创建 Route 53 DNS 记录，并将您的路由控制运行状况检查与每个记录相关联。例如，假设您希望使用两个 DNS 失效转移记录来与路由控制运行状况检查相关联。要让 ARC 使用路由控制正确地对流量进行故障切换，请先在 Route 53 中创建两个故障转移记录：主记录和辅助记录。有关配置 DNS 故障转移记录的更多信息，请参阅运行状况检查概念。

创建主失效转移记录时，其值应如下所示：

```
Name: myapp.yourdomain.com
Type: CNAME
Set Identifier: Primary
```

```
Failover: Primary
TTL: 0
Resource Records:
Value: cell1.yourdomain.com
Health Check ID: xxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
```

辅助失效转移记录值应如下所示：

```
Name: myapp.yourdomain.com
Type: CNAME
Set Identifier: Secondary
Failover: Secondary
TTL: 0
Resource Records:
Value: cell2.yourdomain.com
Health Check ID: xxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
```

现在，假设您要重新路由流量，因为出现了故障。为此，您需要更新关联的路由控制状态，将主路由控制状态更改为 OFF，将辅助路由控制状态更改为 ON。执行此操作时，关联的运行状况检查会阻止流量流向主副本，而是将其路由到辅助副本。有关使用路由控制对流量进行失效转移的更多信息，请参阅使用 ARC API 获取和更新路由控制状态（推荐）。

要查看使用 ARC API 操作创建路由控制和相关运行状况检查的 AWS CLI 命令示例，请参阅使用 ARC 路由控制 API 操作的示例 AWS CLI。

在 ARC 中创建控制面板

Amazon 应用程序恢复控制器 (ARC) 中的控制面板允许您将相关的路由控制组合在一起。控制面板可以包含代表应用程序中的微服务、整个应用程序或一组应用程序的路由控制，具体取决于失效转移的范围。将路径控制组合到控制面板中的一个好处是，您可以配合使用安全规则和控制面板，帮助保护流量路由的变化。

创建集群时，ARC 会创建默认控制面板。您可以使用默认控制面板放置路由控制，也可以创建一个或多个控制面板，对路由控制进行分组。请注意，控制面板名称仅支持 ASCII 字符。

本节包含在 ARC 控制台上创建控制面板的步骤。有关在 ARC 中使用恢复控制配置 API 操作的信息，请参阅路由控制 API 操作。

创建控制面板的步骤

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。

2. 选择路由控制。
3. 在路由控制页面上，选择创建，然后选择控制面板。
4. 请选择要在上面创建控制面板的集群，然后输入面板的名称。
5. 选择创建控制面板。

在 ARC 中查看和更新路由控制状态

本节介绍如何在 Amazon 应用程序恢复控制器 (ARC) 中查看和更新路由控制状态。路由控制是简单的开关机构，管理流向恢复组中的单元格的流量。单元通常是可用区 AWS 区域，有时是包含您的资源的可用区。当路由控制状态为 On 时，流量会流向由受该路由控制所控的单元格。

您可以将路由控制组合到控制面板中，后者是失效转移逻辑分组。例如，当您在控制台上打开控制面板时，您可以同时查看一个分组的所有路由控制，从而看清流量流向何处。

您可以在 ARC 控制台上或使用 ARC API 更新路由控制状态。我们建议您使用 API 更新路由控制状态。首先，ARC 通过数据平面中的 API 来执行这些操作，从而提供了极高的可靠性。当您更改路由控制状态时，这一点很重要，因为路由状态的更改会通过重新路由应用程序流量进行跨单元格的失效转移。此外，使用 API 时，如果您尝试连接的集群端点不可用，可以根据需要尝试轮流连接到不同的集群端点。

您可以更新一个路由控制状态，也可以同时更新多个路由控制状态。例如，您可能想要将一个路由控制状态设置为 Off，以阻止流量流向一个单元格，比如应用程序延迟增加的可用区。同时，您可能想要将另一个路由控制状态设置为 On，以启动流向另一个单元格或可用区的流量。在这种情况下，您可以同时更新两个路由控制状态，以使流量可以持续流动。

主题

使用 ARC API 获取和更新路由控制状态（推荐）

我们建议您使用 Amazon Application Recovery Controller (ARC) API 操作来获取或更新路由控制状态，方法是使用 AWS CLI 命令或使用您开发的用于将 ARC API 操作与其中一个操作一起使用的代码 AWS SDKs。建议使用 CLI 或代码中的 API 操作（而不是使用 AWS Management Console）来处理路由控制状态。

ARC 通过使用 API 更新路由控制状态，为跨单元 (AWS 区域) 进行故障转移提供了极高的可靠性，因为路由控制存储在高度可用的集群中。ARC 确保您始终可以访问五个区域集群终端节点中的至少三个

以进行路由控制状态更改。要使用 API 获取或更改路由控制状态，您需要连接到其中一个区域集群端点。如果该端点不可用，可尝试连接到另一个集群端点。

您可以在 Route 53 控制台或使用 API 操作查看集群的区域集群终端节点列表DescribeCluster。在获取和更改路由控制状态的过程中，应根据需要轮流尝试每个端点，因为集群端点会在可用和不可用状态之间循环，以便定期维护和更新。

我们提供了有关使用 ARC API 操作获取和更新路由控制状态以及使用区域集群终端节点的详细信息和代码示例。有关更多信息，请参阅下列内容：

- 有关说明如何轮换区域集群端点以获取和设置路由控制状态的代码示例，请参阅应用程序恢复控制器使用的操作 AWS SDKs 。
- 有关使用获取和更新路由控制状态的信息，请参阅 使用列出并更新路由控制和状态 AWS CLI。AWS CLI

获取和更新中的路由控制状态 AWS Management Console

您可以在 AWS Management Console中获取和更新路由控制状态。但请注意，您不能在控制台中选择不同的区域集群端点。也就是说，没有像使用 Amazon 应用程序恢复控制器 (ARC) API 那样在控制台中选择和轮换集群终端节点的过程。此外，控制台的可用性不高，而ARC数据平面提供了极高的可靠性。出于这些原因，我们建议您使用 ARC API 来获取和更新生产操作的路由控制状态。

有关使用 ARC 进行故障转移的更多建议，请参阅ARC 中路由控制的最佳实践。

要在控制台中查看和更新路由控制，请按照以下过程中的步骤操作。

获取路由控制状态的步骤

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 选择路由控制。
3. 从列表中选择控制面板并查看路由控制。

更新一个或多个路由控制状态的步骤

1. 在https://console.aws.amazon.com/route53/家中打开亚马逊 Route 53 控制台。
2. 在应用程序恢复控制器下，选择路由控制。
3. 选择操作，然后选择更改流量路由。

4. 根据您想让应用程序的流量流向或停止流向何处，将一个或多个路由控制的状态更新为 Off 或 On。
5. 在文本框中输入 confirm。
6. 选择更新流量路由。

为路径控制创建安全规则

当你同时使用多个路由控制时，你可能会决定要采取安全措施来避免意想不到的后果。例如，您可能希望防止无意中关闭应用程序的所有路由控制，因为这样会导致出现打开失败情况。也许为了防止自动化机制重新路由流量，您也可能想实现一个主开关机构来禁用一组路由控制。要在 ARC 中为路由控制建立类似的保障措施，您需要创建安全规则。

您可以使用路由控制、规则和您指定的其他选项的组合来配置路由控制的安全规则。每个安全规则都与一个控制面板相关联，但一个控制面板可以有多个安全规则。创建安全规则时，请记住在每个控制面板中安全规则名称必须是唯一的。

主题

- 安全规则的类型
- 在控制台上创建安全规则
- 在控制台上编辑或删除安全规则
- 覆盖安全规则以重新路由流量

安全规则的类型

安全规则有两种类型，即断言规则和门控规则，您可以使用它们以不同的方式保护失效转移。

断言规则

使用断言规则，当您更改一个或一组路由控制状态时，ARC 会强制您满足您在配置规则时设置的标准，否则路由控制状态不会更改。

预防打开失败就是适合使用这种规则的一个例子。在打开失败的情况中，您会阻止流量流向一个单元格，但没有让流量开始流向另一个单元格。为了避免这种情况，断言规则确保在任何给定时间控制面板的一组路由控制中至少有一个路由控制是 On 状态。这样可以确保流量流向应用程序的至少一个区域或可用区。

要查看创建断言规则以强制执行此标准的 AWS CLI 命令示例，请参阅中的 使用 ARC 路由控制 API 操作的示例 AWS CLI创建安全规则。

有关断言规则 API 操作属性的详细信息，请参阅 AssertionRuleAmazon 应用程序恢复控制器的路由控制 API 参考指南。

门控规则

使用门控规则时，您可以对一组路由控制实施整体的开关，以根据您在规则中指定的一组标准来判断这些路由控制状态是否可以更改。最简单的标准是，指定为开关的单个路由控制设置为 ON 还是 OFF。

要实现这一点，您需要创建门控路由控制作为整体开关，并创建目标路由控制，以控制流量流向不同的区域或可用区。然后，要防止手动或自动更新您为门控规则配置的目标路由控制的状态，您需要将门控路由控制状态设置为 Off。要允许更新，请将其设置为 On。

要查看用于创建实现此类整体开关的门控规则的示例 AWS CLI 命令，请参阅中的创建安全规则 使用 ARC 路由控制 API 操作的示例 AWS CLI。

有关门控规则 API 操作属性的详细信息，请参阅 GatingRuleAmazon 应用程序恢复控制器的路由控制 API 参考指南。

在控制台上创建安全规则

本节中的步骤说明了如何在 ARC 控制台上创建安全规则。无论您创建断言规则还是门控规则，步骤都是相似的。差异已在程序中注明。

要了解如何在 Amazon 应用程序恢复控制器 (ARC) 中使用恢复和路由控制 API 操作，请参阅 路由控制 API 操作。

创建安全规则的步骤

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 选择路由控制。
3. 在路由控制页面上，选择一个控制面板。
4. 在控制面板详细信息页面上，选择操作，然后选择添加安全规则。
5. 选择要添加的规则类型：断言规则或门控规则。
6. 选择一个名称，然后（可选）更改等待时间。
7. 指定安全规则的配置选项。

   - 对于断言规则，请指定断言的路由控制。

- 对于门控规则，请指定门控路由控制和目标路由控制。

对于两种规则，通过选择类型和阈值以及规则是否反转来指定规则配置。

> **Note**
>
> 要了解有关指定断言规则的更多信息，请参阅 Amazon 应用程序恢复控制器的路由控制 API 参考指南中提供的AssertionRule操作信息。要了解有关指定门控规则的更多信息，请参阅 Amazon 应用程序恢复控制器的路由控制 API 参考指南中为该GatingRule操作提供的信息。

8. 选择创建。

在控制台上编辑或删除安全规则

本节中的步骤说明了如何在 ARC 控制台上编辑或删除安全规则。您只能对安全规则进行有限的编辑，以更改名称或更新等待时间。要进行其他更改，请删除并重新创建安全规则。

要了解如何将 API 操作与 Amazon 应用程序恢复控制器 (ARC) 配合使用，请参阅 路由控制 API 操作。

删除安全规则的步骤

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 选择路由控制。
3. 在路由控制页面上，选择一个控制面板。
4. 在控制面板详细信息页面上，选择安全规则，然后选择删除或编辑。

覆盖安全规则以重新路由流量

在某些情况下，您可能想绕过通过配置的安全规则强制执行的路由控制保护措施。例如，您可能想要快速失效转移以进行灾难恢复，而一个或多个安全规则可能会意外阻止您更新路由控制状态以重新路由流量。在这种“打碎玻璃”的情况下，您可以覆盖一个或多个安全规则来更改路由控制状态并对应用程序进行失效转移。

使用带`safety-rules-to-override`参数的或`update-routing-control-states` AWS CLI 命令更新路由控制状态（或多个路由控制状态）时，可以绕过安全规则。`update-routing-control-`

state使用您要覆盖的安全规则的 Amazon 资源名称 (ARN) 来指定参数，或者指定以逗号分隔的列表 ARNs 来覆盖两个或多个安全规则。

当安全规则阻止路由控制状态更新时，错误消息将包含阻止更新的规则的 ARN。您可以记下 ARN，然后在路由控制状态 CLI 命令的安全规则覆盖参数中指定它。

> **Note**
>
> 由于您正在更新的路由控制可能设有多个安全规则，因此您可能在运行 CLI 命令更新路由控制状态时只覆盖一个安全规则，而收到另一个安全规则阻止更新的错误。继续将安全规则 ARNs 添加到更新命令中要覆盖的规则列表中，以逗号分隔，直到更新命令成功完成。

要详细了解如何将该SafetyRulesToOverride属性与 API 和一起使用 SDKs，请参阅UpdateRoutingControlState。

以下是覆盖安全规则以更新路由控制状态的两个 CLI 命令示例。

覆盖一个安全规则

```
aws route53-recovery-cluster --region us-west-2 update-routing-control-state \
    --routing-control-arn \
    arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/
routingcontrol/abcdefg1234567 \
    --routing-control-state On \
    --safety-rules-to-override arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/safetyrule/
yyyyyyy8888888 \
    --endpoint-url https://host-dddddd.us-west-2.example.com/v1
```

覆盖两个安全规则

```
aws route53-recovery-cluster --region us-west-2 update-routing-control-state \
    --routing-control-arn \
    arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/
routingcontrol/abcdefg1234567 \
    --routing-control-state On \
    --safety-rules-to-override "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/safetyrule/
yyyyyyy8888888" \
```

```
    "arn:aws:route53-recovery-
control::111122223333:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/safetyrule/
qqqqqqq7777777"
    --endpoint-url https://host-dddddd.us-west-2.example.com/v1
```

Support 在 ARC 中支持跨账户集群

Amazon 应用程序恢复控制器 (ARC) 与集成 AWS Resource Access Manager 以实现资源共享。 AWS RAM 是一项使您能够与他人共享资源 AWS 账户 或通过共享资源的服务 AWS Organizations。对于 ARC，您可以共享群集资源。

使用 AWS RAM，您可以通过创建资源共享来共享您拥有的资源。资源共享指定要共享的资源以及共享资源的参与者。参与者可以包括：

- 特定于所有者组织 AWS 账户 内部或外部 AWS Organizations
- 其组织内部的组织单位 AWS Organizations
- 它的整个组织都在 AWS Organizations

有关的更多信息 AWS RAM，请参阅《AWS RAM 用户指南》。

通过使用 AWS Resource Access Manager 在 ARC 中跨账户共享集群资源，您可以使用一个集群来托管由多个不同账户拥有的控制面板和路由控件 AWS 账户。当您选择共享集群时，您指定的其他 AWS 账户 人可以使用该集群来托管自己的控制面板和路由控件，从而可以对不同团队之间的路由功能进行更多的控制和灵活性。

AWS RAM 是一项可帮助 AWS 客户安全地共享资源的服务 AWS 账户。借 AWS RAM助，您可以使用 IAM 角色和用户在 AWS Organizations组织或组织单位 (OUs) 内共享资源。 AWS RAM 是一种集中和受控的群集共享方式。

通过共享集群，可以减少组织所需的集群总数。使用共享集群，您可以将运行集群的总成本分配给不同的团队，从而以更低的成本最大限度地发挥 ARC 的优势。（创建托管在集群中的资源不会给拥有者或参与者增加成本。）跨账户共享集群还可以简化将多个应用程序加载到 ARC 的过程，尤其是在您有大量应用程序分布在多个客户和运营团队的情况下。

要开始在 ARC 中进行跨账户共享，请在中创建资源共享。 AWS RAM资源共享指定有权共享您的账户所拥有的集群的参与者。然后，参与者可以在集群中创建资源，例如控制面板和路由控件，方法是使用 AWS Management Console 或运行 ARC API 操作 AWS SDKs。 AWS Command Line Interface

本主题说明如何共享您拥有的资源以及如何使用共享给您的资源。

内容

共享集群的先决条件

- 要共享集群，您必须在自己的集群中拥有该集群 AWS 账户。这意味着资源必须分配或预调配到您的账户。您无法共享已共享给您的集群。
- 要与您的组织或 AWS Organizations内的组织单位共享集群，您必须允许与 AWS Organizations共享。有关更多信息，请参阅《AWS RAM 用户指南》中的允许与 AWS Organizations共享。

共享集群

当您共享自己拥有的集群时，您指定共享该集群的参与者可以在集群中创建和托管他们自己的 ARC 资源。

要共享集群，您必须将它添加到资源共享中。资源共享是一项 AWS RAM 资源，可让您跨 AWS 账户共享资源。资源共享将指定要共享的资源以及共享资源的参与者。要共享集群，您可以创建新的资源共享或将资源添加到现有资源共享。要创建新的资源共享，您可以使用AWS RAM 控制台，也可以将 AWS RAM API 操作与 AWS Command Line Interface 或一起使用 AWS SDKs。

如果您是组织中组织的一员， AWS Organizations 并且启用了组织内部共享，则系统会自动授予组织中的参与者访问共享群集的权限。否则，参与者会收到加入资源共享的邀请，并在接受邀请后获得对共享集群的访问权限。

您可以使用 AWS RAM 控制台共享您拥有的集群，也可以通过使用或的 AWS RAM API 操作来共享您拥有的 AWS CLI 集群 SDKs。

使用 AWS RAM 控制台共享您拥有的集群

请参阅《AWS RAM 用户指南》中的创建资源共享。

要共享您拥有的集群，请使用 AWS CLI

使用 create-resource-share 命令。

授予共享集群的权限

跨账户共享集群需要通过共享集群的 IAM 委托人的权限 AWS RAM。

我们建议使用AmazonRoute53RecoveryControlConfigFullAccess托管 IAM 策略来确保您的 IAM 委托人拥有共享和使用共享集群所需的权限。

使用自定义 IAM 策略共享集群需要route53-recovery-control-config:PutResourcePolicyroute53-recovery-control-config:GetResourcePolicy、和该集群的route53-recovery-control-config:DeleteResourcePolicy权限。PutResourcePolicy并且DeleteResourcePolicy是仅限权限的 IAM 操作。在没有这些权限 AWS RAM 的情况下尝试通过共享集群将导致错误。

有关 AWS Resource Access Manager 使用 IAM 的方式的更多信息，请参阅AWS RAM 用户指南中的如何 AWS Resource Access Manager 使用 IAM。

取消共享集群

取消共享集群时，以下规则适用于参与者和拥有者：

- 现有参与者资源将继续留存在已取消共享的集群中。
- 参与者可以继续在已取消共享的集群中更新路由控制状态，以管理应用程序失效转移的路由。
- 参与者不能再在已取消共享的集群中创建新资源。
- 如果参与者在已取消共享的集群中仍有资源，则拥有者无法删除共享集群。

要取消共享您拥有的共享集群，必须从资源共享中将其删除。为此，您可以使用 AWS RAM 控制台或将 AWS RAM API 操作与 AWS CLI 或一起使用 SDKs。

使用控制台取消共享您拥有的共享集群 AWS RAM

请参阅《AWS RAM 用户指南》中的更新资源共享。

要取消共享您拥有的共享集群，请使用 AWS CLI

使用 disassociate-resource-share 命令。

识别共享集群

拥有者和参与者可以通过查看 AWS RAM中的信息来识别共享集群。他们还可以使用 ARC 控制台获取有关共享资源的信息，以及 AWS CLI。

一般而言，要详细了解您已共享或已与您共享的资源，请参阅《 AWS Resource Access Manager 用户指南》中的信息：

- 作为拥有者，您可以使用 AWS RAM查看与他人共享的所有资源。有关更多信息，请参阅中的查看共享资源 AWS RAM。
- 作为参与者，您可以使用查看与您共享的所有资源 AWS RAM。有关更多信息，请参阅中的查看共享资源 AWS RAM。

作为所有者，您可以通过查看中的信息 AWS Management Console 或使用 with ARC API 操作来确定是否共享集群。 AWS Command Line Interface

使用控制台确定您拥有的集群是否已共享的步骤

在 AWS Management Console集群的详细信息页面上，查看集群共享状态。

要确定您拥有的集群是否已共享，请使用 AWS CLI

使用 get-resource-policy 命令。如果集群有资源策略，则该命令将返回有关该策略的信息。

作为参与者，当集群共享给您时，您通常必须接受共享。此外，集群的拥有者字段包含集群拥有者的帐户。

共享集群的责任和权限

拥有者的权限

当您与其他人共享您拥有的集群时 AWS 账户，允许使用该集群的参与者可以在集群中创建控制面板、路由控件和其他资源。

作为集群的拥有者，您负责创建、管理和删除集群。您不能修改或删除参与者创建的资源，例如路由控制和安全规则。例如，您不能更新参与者创建的路由控制，以更改路由控制状态。

但是，您可以查看参与者在您拥有的集群中创建的路由控制的详细信息。例如，您可以使用 AWS Command Line Interface 或调用 ARC 路由控制 API 操作来查看路由控制状态 AWS SDKs。

如果您需要修改参与者创建的资源，他们可以在 IAM 中设置一个拥有资源访问权限的角色，并将您的账户添加到该角色中。

参与者的权限

一般而言，参与者可以创建和使用他们在共享的集群中创建的控制面板、路由控制、安全规则和运行状况检查。只有他们在共享集群中拥有资源，才能查看、修改或删除这些集群资源。例如，参与者可以为自己创建的控制面板创建和删除安全规则。

以下限制适用于参与者：

- 参与者无法查看、修改或删除由使用共享集群的其他账户创建的控制面板。
- 参与者无法查看、创建或修改其他账户在共享集群中创建的资源的路由控制，包括路由控制状态。
- 参与者无法创建、修改或查看其他账户在共享集群中创建的安全规则。
- 参与者无法在共享集群的默认控制面板中添加资源，因为它属于集群拥有者。

如前所述，参与者无法在共享集群的默认控制面板中创建路由控制，因为集群拥有者拥有默认控制面板。但是，集群拥有者可以创建跨账户 IAM 角色，以提供访问集群默认控制面板的权限。然后，拥有者可以向参与者授予权限以担任该角色，这样参与者就可以访问默认控制面板，按照拥有者通过角色权限指定的方式使用该面板。

成本计费

ARC 中集群的所有者需要支付与该集群相关的费用。对于集群拥有者或参与者来说，创建托管在集群中的资源不会产生任何额外成本。

有关详细的定价信息和示例，请参阅亚马逊应用程序恢复控制器 (ARC) 定价，然后向下滚动至亚马逊应用程序恢复控制器 (ARC)。

限额

在共享集群中创建的所有资源（包括有权访问共享集群的所有参与者创建的资源）都计入该群集和其他资源（例如路由控制）的有效限额。如果共享群集资源的账户的配额高于群集所有者的配额，则群集所有者的配额优先于共享的账户的配额。

要更好地理解其工作原理，请参阅以下示例。为了说明配额如何与资源共享一起使用，在这些示例中，假设集群所有者是所有者，而与之共享集群的账户是参与者。

控制面板配额

所有者对每个集群的控制面板总数实行配额。

例如，假设 Owner 每个集群的控制面板数量配额为 50，并且集群中有 13 个控制面板。现在，假设参与者的配额设置为 150。在这种情况下，参与者只能在共享集群中创建最多 37 个控制面板（即 50-13）。

此外，如果共享集群的其他账户也创建了控制面板，则这些控制面板也都计入集群总配额（50 个控制面板）。

路由控制配额

路由控制有多个配额：每个控制面板的配额、每个集群的配额以及每个安全规则的配额。所有这些配额均优先考虑所有者的配额。

例如，假设 Owner 每个集群的路由控制数量配额为 300，并且集群中已经有 300 个路由控件。现在，假设参与者已将此配额设置为 500。在这种情况下，参与者无法在共享集群中创建任何新的路由控件。

安全规则配额

根据控制面板配额，所有者的安全规则将强制执行配额。

例如，假设所有者每个控制面板的安全规则数量配额为 20，参与者将此配额设置为 80。在这种情况下，由于所有者的下限优先，因此参与者只能在共享集群的控制面板中创建最多 20 条安全规则。

有关路由控制配额的列表，请参阅路由控制配额。

在 Amazon 应用程序恢复控制器 (ARC) 中记录和监控路由控制

您可以使用 AWS CloudTrail 监控 Amazon 应用程序恢复控制器 (ARC) 中的路由控制，以分析模式并帮助解决问题。

主题

- 使用记录 ARC API 调用 AWS CloudTrail

使用记录 ARC API 调用 AWS CloudTrail

与 AWS CloudTrail一项服务集成，该服务提供用户、角色或 AWS 服务在 ARC 中执行的操作的记录。CloudTrail 将 ARC 的所有 API 调用捕获为事件。捕获的调用包括来自 ARC 控制台的调用和对 ARC API 操作的代码调用。

如果您创建跟踪，则可以允许将 CloudTrail 事件持续传输到 Amazon S3 存储桶，包括 ARC 的事件。如果您未配置跟踪，您仍然可以在 CloudTrail 控制台的 “事件历史记录” 中查看最新的事件。

使用收集的信息 CloudTrail，您可以确定向 ARC 发出的请求、发出请求的 IP 地址、谁发出了请求、何时发出请求以及其他详细信息。

要了解更多信息 CloudTrail，请参阅《AWS CloudTrail 用户指南》。

ARC 信息在 CloudTrail

CloudTrail 在您创建账户 AWS 账户 时已在您的账户上启用。当活动在 ARC 中发生时，该活动会与其他 AWS 服务 CloudTrail 事件一起记录在事件历史记录中。您可以在中查看、搜索和下载最近发生的事件 AWS 账户。有关更多信息，请参阅使用 CloudTrail 事件历史记录。

要持续记录您的 AWS 账户事件（包括 ARC 的事件），请创建跟踪。跟踪允许 CloudTrail 将日志文件传输到 Amazon S3 存储桶。预设情况下，在控制台中创建跟踪记录时，此跟踪记录应用于所有 AWS 区域。跟踪记录 AWS 分区中所有区域的事件，并将日志文件传送到您指定的 Amazon S3 存储桶。此外，您可以配置其他 AWS 服务，以进一步分析和处理 CloudTrail 日志中收集的事件数据。有关更多信息，请参阅下列内容：

- 创建跟踪记录概述
- CloudTrail 支持的服务和集成
- 配置 Amazon SNS 通知 CloudTrail
- 接收来自多个区域的 CloudTrail 日志文件和接收来自多个账户的 CloudTrail 日志文件

所有 ARC 操作均由 CloudTrail 《亚马逊应用程序恢复控制器恢复准备 API 参考指南》、《亚马逊应用程序恢复控制器恢复控制器恢复控制配置 API 参考指南》和《亚马逊应用程序恢复控制器路由控制 API 参考指南》记录并记录在《亚马逊应用程序恢复控制器 API 参考指南》中。例如，调用`UpdateRoutingControlState`和`CreateRecoveryGroup`操作会在 CloudTrail 日志文件中生成条目。`CreateCluster`

每个事件或日志条目都包含有关生成请求的人员信息。身份信息有助于您确定以下内容：

- 请求是使用根证书还是 AWS Identity and Access Management (IAM) 用户凭证发出。
- 请求是使用角色还是联合用户的临时安全凭证发出的。
- 请求是否由其他 AWS 服务发出。

有关更多信息，请参阅 CloudTrail userIdentity 元素。

在事件历史记录中查看 ARC 事件

CloudTrail 允许您在事件历史记录中查看最近的事件。要查看 ARC API 请求的事件，您必须在控制台顶部的区域选择器中选择美国西部（俄勒冈）。有关更多信息，请参阅《AWS CloudTrail 用户指南》中的 “使用 CloudTrail 事件历史记录”。

了解 ARC 日志文件条目

跟踪是一种配置，允许将事件作为日志文件传输到您指定的 Amazon S3 存储桶。 CloudTrail 日志文件包含一个或多个日志条目。事件代表来自任何来源的单个请求，包括有关请求的操作、操作的日期和时间、请求参数等的信息。 CloudTrail 日志文件不是公共 API 调用的有序堆栈跟踪，因此它们不会按任何特定的顺序出现。

以下示例显示了一个 CloudTrail 日志条目，该条目演示了配置路由控制的`CreateCluster`操作。

```
{
  "eventVersion": "1.08",
   "userIdentity": {
     "type": "IAMUser",
     "principalId": "A1B2C3D4E5F6G7EXAMPLE",
     "arn": "arn:aws:iam::111122223333:user/smithj",
     "accountId": "111122223333",
     "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
     "sessionContext": {
          "sessionIssuer": {
              "type": "Role",
              "principalId": "A1B2C3D4E5F6G7EXAMPLE",
              "arn": "arn:aws:iam::111122223333:role/smithj",
              "accountId": "111122223333",
              "userName": "smithj"
          },
          "webIdFederationData": {},
          "attributes": {
              "mfaAuthenticated": "false",
              "creationDate": "2021-06-30T04:44:41Z"
          }
      }
  },
  "eventTime": "2021-06-30T04:45:46Z",
  "eventSource": "route53-recovery-control-config.amazonaws.com",
  "eventName": "CreateCluster",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "192.0.2.50",
```

```
  "userAgent": "aws-cli/2.0.0 Python/3.8.2 Darwin/19.6.0 botocore/2.0.0dev7",
  "requestParameters": {
      "ClientToken": "12345abcdef-1234-5678-abcd-12345abcdef",
      "ClusterName": "XYZCluster"
  },
  "responseElements": {
      "Cluster": {
          "Arn": "arn:aws:route53-recovery-control::012345678901:cluster/abc123456-
aa11-bb22-cc33-abc123456",
          "ClusterArn": "arn:aws:route53-recovery-control::012345678901:cluster/
abc123456-aa11-bb22-cc33-abc123456",
          "Name": "XYZCluster",
          "Status": "PENDING"
      }
  },
  "requestID": "6090509a-5a97-4be6-8e6a-7d73example",
  "eventID": "9cab44ef-0777-41e6-838f-f249example",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "eventCategory": "Management",
  "recipientAccountId": "111122223333"
}
```

以下示例显示了一个演示路由控制UpdateRoutingControlState操作的 CloudTrail 日志条目。

```
{
  "eventVersion": "1.08",
   "userIdentity": {
     "type": "AssumedRole",
     "principalId": "A1B2C3D4E5F6G7EXAMPLE",
     "arn": "arn:aws:sts::111122223333:assumed-role/admin/smithj",
     "accountId": "111122223333",
     "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
     "sessionContext": {
          "sessionIssuer": {
              "type": "Role",
              "principalId": "A1B2C3D4E5F6G7EXAMPLE",
              "arn": "arn:aws:iam::111122223333:role/admin",
              "accountId": "111122223333",
              "userName": "admin"
          },
          "webIdFederationData": {},
```

```
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2021-06-30T04:44:41Z"
            }
        }
    },
    "eventTime": "2021-06-30T04:45:46Z",
    "eventSource": "route53-recovery-control-config.amazonaws.com",
    "eventName": "UpdateRoutingControl",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.50",
    "userAgent": "aws-cli/2.0.0 Python/3.8.2 Darwin/19.6.0 botocore/2.0.0dev7",
    "requestParameters": {
        "RoutingControlName": "XYZRoutingControl3",
        "RoutingControlArn": "arn:aws:route53-recovery-
control::012345678:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
abcdefg1234567"
    },
    "responseElements": {
        "RoutingControl": {
            "ControlPanelArn": "arn:aws:route53-recovery-
control::012345678:controlpanel/0123456bbbbbbb0123456bbbbbb0123456",
            "Name": "XYZRoutingControl3",
            "Status": "DEPLOYED",
            "RoutingControlArn": "arn:aws:route53-recovery-
control::012345678:controlpanel/0123456bbbbbbb0123456bbbbbb0123456/routingcontrol/
abcdefg1234567"
        }
    },
    "requestID": "6090509a-5a97-4be6-8e6a-7d73example",
    "eventID": "9cab44ef-0777-41e6-838f-f249example",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "111122223333"
}
```


用于路由控制的 Identity and Access Management

AWS Identity and Access Management (IAM) AWS 服务 可帮助管理员安全地控制对 AWS 资源的访问权限。IAM 管理员控制谁可以进行身份验证（登录）和授权（有权限）使用 ARC 资源。您可以使用 IAM AWS 服务 ，无需支付额外费用。

内容

Amazon 应用程序恢复控制器 (ARC) 中的路由控制如何与 IAM 配合使用

在使用 IAM 管理对 Amazon 应用程序恢复控制器 (ARC) 中的路由控制的访问权限之前，请先了解有哪些 IAM 功能可用于路由控制。

您可以在 Amazon 应用程序恢复控制器 (ARC) 中使用路由控制的 IAM 功能

IAM 特征	路由控制支持
基于身份的策略	是
基于资源的策略	否
策略操作	是
策略资源	是
策略条件键	是
ACLs	否
ABAC（策略中的标签）	部分
临时凭证	是
主体权限	是
服务角色	否
服务相关角色	否

要全面了解 AWS 服务如何与大多数 IAM 功能配合使用，请参阅 IAM 用户指南中的与 IAM 配合使用的 AWS 服务。

ARC 基于身份的策略

支持基于身份的策略：是

基于身份的策略是可附加到身份（如 IAM 用户、用户组或角色）的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的使用客户管理型策略定义自定义 IAM 权限。

通过使用 IAM 基于身份的策略，您可以指定允许或拒绝的操作和资源以及允许或拒绝操作的条件。您无法在基于身份的策略中指定主体，因为它适用于其附加的用户或角色。要了解可在 JSON 策略中使用的所有元素，请参阅《IAM 用户指南》中的 IAM JSON 策略元素引用。

要查看用于路由控制的 ARC 基于身份的策略示例，请参阅。Amazon 应用程序恢复控制器 (ARC) 中基于身份的路由控制策略示例

路由控制中基于资源的策略

支持基于资源的策略：否

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资源的访问。

路由控制的策略操作

支持策略操作：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

JSON 策略的 `Action` 元素描述可用于在策略中允许或拒绝访问的操作。策略操作通常与关联的 AWS API 操作同名。有一些例外情况，例如没有匹配 API 操作的仅限权限 操作。还有一些操作需要在策略中执行多个操作。这些附加操作称为相关操作。

在策略中包含操作以授予执行关联操作的权限。

要查看用于路由控制的 ARC 操作列表，请参阅《服务授权参考》中的 Amazon Route 53 恢复控制定义的操作和 Amazon Route 53 恢复集群定义的操作。

ARC 中用于路由控制的策略操作在操作前使用以下前缀，具体取决于您正在使用的 API：

```
route53-recovery-control-config
```

```
route53-recovery-cluster
```

要在单个语句中指定多项操作，请使用逗号将它们隔开。例如，可以：

```
"Action": [
      "route53-recovery-control-config:action1",
      "route53-recovery-control-config:action2"
         ]
```

您也可以使用通配符（*）指定多个操作。例如，要指定以单词 Describe 开头的所有操作，包括以下操作：

```
"Action": "route53-recovery-control-config:Describe*"
```

要查看用于路由控制的 ARC 基于身份的策略示例，请参阅。Amazon 应用程序恢复控制器 (ARC) 中基于身份的路由控制策略示例

ARC 的政策资源

支持策略资源：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

Resource JSON 策略元素指定要向其应用操作的一个或多个对象。语句必须包含 Resource 或 NotResource 元素。作为最佳实践，请使用其 Amazon 资源名称（ARN）指定资源。对于支持特定资源类型（称为资源级权限）的操作，您可以执行此操作。

对于不支持资源级权限的操作（如列出操作），请使用通配符（*）指示语句应用于所有资源。

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

在《服务授权参考》中，您可以看到以下与 ARC 相关的信息：

要查看资源类型及其列表 ARNs，以及您可以使用每种资源的 ARN 指定的操作，请参阅《服务授权参考》中的以下主题：

- Amazon Route 53 恢复控制定义的操作
- 由 Amazon Route 53 恢复集群定义的操作。

要查看用于路由控制的 ARC 基于身份的策略示例，请参阅。Amazon 应用程序恢复控制器 (ARC) 中基于身份的路由控制策略示例

ARC 的策略条件密钥

支持特定于服务的策略条件键：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

在 Condition 元素（或 Condition 块）中，可以指定语句生效的条件。Condition 元素是可选的。您可以创建使用条件运算符（例如，等于或小于）的条件表达式，以使策略中的条件与请求中的值相匹配。

如果您在一个语句中指定多个 Condition 元素，或在单个 Condition 元素中指定多个键，则 AWS 使用逻辑 AND 运算评估它们。如果您为单个条件键指定多个值，则使用逻辑OR运算来 AWS 评估条件。在授予语句的权限之前必须满足所有的条件。

在指定条件时，您也可以使用占位符变量。例如，只有在使用 IAM 用户名标记 IAM 用户时，您才能为其授予访问资源的权限。有关更多信息，请参阅《IAM 用户指南》中的 IAM 策略元素：变量和标签。

AWS 支持全局条件密钥和特定于服务的条件密钥。要查看所有 AWS 全局条件键，请参阅 IAM 用户指南中的AWS 全局条件上下文密钥。

要查看路由控制的 ARC 条件键列表，请参阅《服务授权参考》中的以下主题：

- Amazon Route 53 恢复控制的条件键
- Amazon Route 53 恢复集群的条件键

要查看可与条件键配合使用的操作和资源，请参阅《服务授权参考》中的以下主题：

- 要查看资源类型及其列表 ARNs，请参阅 Amazon Route 53 恢复控制定义的操作和 Amaz on Route 53 恢复集群定义的操作。
- 要查看您可以使用每种资源的 ARN 指定的操作列表，请参阅 Amazon Route 53 恢复控制定义的资源和由 Amazon Route 53 恢复集群定义的资源。

要查看 ARC 基于身份的路由控制策略示例，请参阅 Amazon 应用程序恢复控制器 (ARC) 中基于身份的路由控制策略示例

ARC 中的访问控制列表 (ACLs)

支持 ACLs：否

访问控制列表 (ACLs) 控制哪些委托人（账户成员、用户或角色）有权访问资源。 ACLs 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

使用 ARC 实现基于属性的访问控制 (ABAC)

支持 ABAC（策略中的标签）：部分支持

基于属性的访问控制（ABAC）是一种授权策略，该策略基于属性来定义权限。在中 AWS，这些属性称为标签。您可以向 IAM 实体（用户或角色）和许多 AWS 资源附加标签。标记实体和资源是 ABAC 的第一步。然后设计 ABAC 策略，以在主体的标签与他们尝试访问的资源标签匹配时允许操作。

ABAC 在快速增长的环境中非常有用，并在策略管理变得繁琐的情况下可以提供帮助。

要基于标签控制访问，您需要使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 条件键在策略的条件元素中提供标签信息。

如果某个服务对于每种资源类型都支持所有这三个条件键，则对于该服务，该值为是。如果某个服务仅对于部分资源类型支持所有这三个条件键，则该值为部分。

有关 ABAC 的更多信息，请参阅《IAM 用户指南》中的使用 ABAC 授权定义权限。要查看设置 ABAC 步骤的教程，请参阅《IAM 用户指南》中的使用基于属性的访问权限控制（ABAC）。

ARC 路由控制包括对 ABAC 的以下支持：

- 恢复控制 Config 支持 ABAC。
- 恢复集群不支持 ABAC。

在 ARC 中使用临时证书

支持临时凭证：是

当你使用临时证书登录时，有些 AWS 服务 不起作用。有关更多信息，包括哪些 AWS 服务 适用于临时证书，请参阅 IAM 用户指南中的AWS 服务 与 IAM 配合使用的信息。

如果您使用除用户名和密码之外的任何方法登录，则 AWS Management Console 使用的是临时证书。例如，当您 AWS 使用公司的单点登录 (SSO) 链接进行访问时，该过程会自动创建临时证书。当您以

用户身份登录控制台，然后切换角色时，您还会自动创建临时凭证。有关切换角色的更多信息，请参阅《IAM 用户指南》中的从用户切换到 IAM 角色（控制台）。

您可以使用 AWS CLI 或 AWS API 手动创建临时证书。然后，您可以使用这些临时证书进行访问 AWS。 AWS 建议您动态生成临时证书，而不是使用长期访问密钥。有关更多信息，请参阅 IAM 中的临时安全凭证。

ARC 的跨服务主体权限

支持转发访问会话（FAS）：是

当您使用 IAM 实体（用户或角色）在中执行操作时 AWS，您被视为委托人。策略向主体授予权限。使用某些服务时，您可能会执行一个操作，此操作然后在不同服务中触发另一个操作。在这种情况下，您必须具有执行这两个操作的权限。

要查看某个操作是否需要策略中的其他相关操作，请参阅《服务授权参考》中的以下主题：

- Amazon Route 53 恢复集群
- Amazon Route 53 恢复控制

ARC 的服务角色

支持服务角色：否

服务角色是由一项服务担任、代表您执行操作的 IAM 角色。IAM 管理员可以在 IAM 中创建、修改和删除服务角色。有关更多信息，请参阅《IAM 用户指南》中的创建向 AWS 服务委派权限的角色。

ARC 的服务相关角色

支持服务相关角色：是

服务相关角色是一种与服务关联的 AWS 服务角色。服务可以代入代表您执行操作的角色。服务相关角色显示在您的 AWS 账户中，并归服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。

路由控制不使用服务相关角色。

Amazon 应用程序恢复控制器 (ARC) 中基于身份的路由控制策略示例

默认情况下，用户和角色无权创建或修改 ARC 资源。他们也无法使用 AWS Management Console、AWS Command Line Interface (AWS CLI) 或 AWS API 执行任务。要授予用户对所需资源执行操作的权限，IAM 管理员可以创建 IAM 策略。管理员随后可以向角色添加 IAM 策略，用户可以代入角色。

要了解如何使用这些示例 JSON 策略文档创建基于 IAM 身份的策略，请参阅《IAM 用户指南》中的创建 IAM 策略（控制台）。

有关 ARC 定义的操作和资源类型（包括每种资源类型的格式）的详细信息，请参阅《 ARNs 服务授权参考》中的 Amazon Application Recovery Controller (ARC) 的操作、资源和条件密钥。

主题

- 策略最佳实践
- 示例：用于路由控制的 ARC 控制台访问权限
- 示例：用于路由控制配置的 ARC API 操作

策略最佳实践

基于身份的策略决定了某人是否可以在您的账户中创建、访问或删除 ARC 资源。这些操作可能会使 AWS 账户产生成本。创建或编辑基于身份的策略时，请遵循以下指南和建议：

- 开始使用 AWS 托管策略并转向最低权限权限 — 要开始向用户和工作负载授予权限，请使用为许多常见用例授予权限的AWS 托管策略。它们在你的版本中可用 AWS 账户。我们建议您通过定义针对您的用例的 AWS 客户托管策略来进一步减少权限。有关更多信息，请参阅《IAM 用户指南》中的AWS 托管式策略或工作职能的AWS 托管式策略。
- 应用最低权限：在使用 IAM 策略设置权限时，请仅授予执行任务所需的权限。为此，您可以定义在特定条件下可以对特定资源执行的操作，也称为最低权限许可。有关使用 IAM 应用权限的更多信息，请参阅《IAM 用户指南》中的 IAM 中的策略和权限。
- 使用 IAM 策略中的条件进一步限制访问权限：您可以向策略添加条件来限制对操作和资源的访问。例如，您可以编写策略条件来指定必须使用 SSL 发送所有请求。如果服务操作是通过特定 AWS 服务的（例如）使用的，则也可以使用条件来授予对服务操作的访问权限 AWS CloudFormation。有关更多信息，请参阅《IAM 用户指南》中的 IAM JSON 策略元素：条件。
- 使用 IAM Access Analyzer 验证您的 IAM 策略，以确保权限的安全性和功能性 – IAM Access Analyzer 会验证新策略和现有策略，以确保策略符合 IAM 策略语言（JSON）和 IAM 最佳实践。IAM Access Analyzer 提供 100 多项策略检查和可操作的建议，以帮助您制定安全且功能性强的策略。有关更多信息，请参阅《IAM 用户指南》中的使用 IAM Access Analyzer 验证策略。
- 需要多重身份验证 (MFA)-如果 AWS 账户您的场景需要 IAM 用户或根用户，请启用 MFA 以提高安全性。若要在调用 API 操作时需要 MFA，请将 MFA 条件添加到您的策略中。有关更多信息，请参阅《IAM 用户指南》中的使用 MFA 保护 API 访问。

有关 IAM 中的最佳实操的更多信息，请参阅《IAM 用户指南》中的 IAM 中的安全最佳实践。

示例：用于路由控制的 ARC 控制台访问权限

要访问 Amazon 应用程序恢复控制器 (ARC) 控制台，您必须拥有一组最低权限。这些权限必须允许您列出和查看有关您的 ARC 资源的详细信息 AWS 账户。如果创建比必需的最低权限更为严格的基于身份的策略，对于附加了该策略的实体（用户或角色），控制台将无法按预期正常运行。

对于仅调用 AWS CLI 或 AWS API 的用户，您无需为其设置最低控制台权限。相反，只允许访问与其尝试执行的 API 操作相匹配的操作。

为确保在您仅允许访问特定 API 操作时用户和角色仍然可以使用 ARC 控制台，还要为实体附加 ARC `ReadOnly` AWS 托管策略。有关更多信息，请参阅 ARC 托管策略页面或 IAM 用户指南中的向用户添加权限。

要向用户提供通过控制台使用 ARC 路由控制功能的完全访问权限，请向用户附加如下策略，以授予用户配置 ARC 路由控制资源和操作的完全权限：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                   "route53-recovery-cluster:GetRoutingControlState",
                   "route53-recovery-cluster:UpdateRoutingControlState",
                   "route53-recovery-cluster:UpdateRoutingControlStates",
                   "route53-recovery-control-config:CreateCluster",
                   "route53-recovery-control-config:CreateControlPanel",
                   "route53-recovery-control-config:CreateRoutingControl",
                   "route53-recovery-control-config:CreateSafetyRule",
                   "route53-recovery-control-config:DeleteCluster",
                   "route53-recovery-control-config:DeleteControlPanel",
                   "route53-recovery-control-config:DeleteRoutingControl",
                   "route53-recovery-control-config:DeleteSafetyRule",
                   "route53-recovery-control-config:DescribeCluster",
                   "route53-recovery-control-config:DescribeControlPanel",
                   "route53-recovery-control-config:DescribeSafetyRule",
                   "route53-recovery-control-config:DescribeRoutingControl",
                   "route53-recovery-control-config:ListAssociatedRoute53HealthChecks",
                   "route53-recovery-control-config:ListClusters",
                   "route53-recovery-control-config:ListControlPanels",
                   "route53-recovery-control-config:ListRoutingControls",
                   "route53-recovery-control-config:ListSafetyRules",
                   "route53-recovery-control-config:UpdateControlPanel",
```

```
                "route53-recovery-control-config:UpdateRoutingControl",
                "route53-recovery-control-config:UpdateSafetyRule"
            ],
            "Resource": "*"
        },
            {
            "Effect": "Allow",
            "Action": [
                "route53:GetHealthCheck",
                "route53:CreateHealthCheck",
                "route53:DeleteHealthCheck",
                "route53:ChangeTagsForResource"
            ],
            "Resource": "*"
        }
    ]
}
```

示例：用于路由控制配置的 ARC API 操作

为确保用户可以使用 ARC API 操作来处理 ARC 路由控制配置，请附加与用户需要使用的 API 操作相对应的策略，如下所述。

要使用 API 操作进行恢复控制配置，请向用户附加类似以下的策略：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "route53-recovery-control-config:CreateCluster",
                "route53-recovery-control-config:CreateControlPanel",
                "route53-recovery-control-config:CreateRoutingControl",
                "route53-recovery-control-config:CreateSafetyRule",
                "route53-recovery-control-config:DeleteCluster",
                "route53-recovery-control-config:DeleteControlPanel",
                "route53-recovery-control-config:DeleteRoutingControl",
                "route53-recovery-control-config:DeleteSafetyRule",
                "route53-recovery-control-config:DescribeCluster",
                "route53-recovery-control-config:DescribeControlPanel",
                "route53-recovery-control-config:DescribeSafetyRule",
                "route53-recovery-control-config:DescribeRoutingControl",
                "route53-recovery-control-config:GetResourcePolicy",
```

```
                    "route53-recovery-control-config:ListAssociatedRoute53HealthChecks",
                    "route53-recovery-control-config:ListClusters",
                    "route53-recovery-control-config:ListControlPanels",
                    "route53-recovery-control-config:ListRoutingControls",
                    "route53-recovery-control-config:ListSafetyRules",
                    "route53-recovery-control-config:ListTagsForResource",
                    "route53-recovery-control-config:UpdateControlPanel",
                    "route53-recovery-control-config:UpdateRoutingControl",
                    "route53-recovery-control-config:UpdateSafetyRule",
                    "route53-recovery-control-config:TagResource",
                    "route53-recovery-control-config:UntagResource"
            ],
            "Resource": "*"
        }
    ]
}
```

要使用恢复集群数据平面 API 在 ARC 路由控制中执行任务，例如，更新路由控制状态以在灾难事件期间进行故障转移，您可以将如下所示的 ARC IAM 策略附加到您的 IAM 用户。

AllowSafetyRuleOverride 布尔值提供权限来覆盖您为保障路由控制安全而配置的安全规则。在“打碎玻璃”的情况下，可能需要此权限才能在灾难或其他紧急失效转移情况下绕过这些安全措施。例如，操作员可能需要快速失效转移以进行灾难恢复，而一个或多个安全规则可能会意外阻止流量重新路由所需的路由控制状态更新操作。此权限允许操作员在调用 API 更新路由控制状态时指定要覆盖的安全规则。有关更多信息，请参阅 覆盖安全规则以重新路由流量。

如果要允许操作员使用恢复集群数据平面 API，但又不想覆盖安全规则，则可以附加如下所示的策略，并将*AllowSafetyRuleOverrides*布尔值设置为*false*。要允许操作员忽略安全规则，请将AllowSafetyRuleOverrides布尔值设置为true。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "route53-recovery-cluster:GetRoutingControlState",
                "route53-recovery-cluster:ListRoutingControls"
            ],
            "Resource": "*"
        },
        {
```

```
            "Effect": "Allow",
            "Action": [
                "route53-recovery-cluster:UpdateRoutingControlStates",
                "route53-recovery-cluster:UpdateRoutingControlState"
            ],
            "Resource": "*",
            "Condition": {
                "Bool": {
                    "route53-recovery-cluster:AllowSafetyRulesOverrides": "false"
                }
            }
        }
    ]
}
```

AWS Amazon 应用程序恢复控制器 (ARC) 中用于路由控制的托管策略

AWS 托管策略是由创建和管理的独立策略 AWS。 AWS 托管策略旨在为许多常见用例提供权限，以便您可以开始为用户、组和角色分配权限。

请记住， AWS 托管策略可能不会为您的特定用例授予最低权限权限，因为它们可供所有 AWS 客户使用。我们建议通过定义特定于您的使用场景的客户管理型策略来进一步减少权限。

您无法更改 AWS 托管策略中定义的权限。如果 AWS 更新 AWS 托管策略中定义的权限，则更新会影响该策略所关联的所有委托人身份（用户、组和角色）。 AWS 最有可能在启动新的 API 或现有服务可以使用新 AWS 服务 的 API 操作时更新 AWS 托管策略。

有关更多信息，请参阅《IAM 用户指南》中的 AWS 托管策略。

AWS 托管策略： AmazonRoute53 RecoveryControlConfigFullAccess

您可以将 AmazonRoute53RecoveryControlConfigFullAccess 附加到 IAM 实体。此策略授予对 ARC 中使用恢复控制配置的操作的完全访问权限。将此策略附加到需要恢复控制配置操作的完全访问权限的 IAM 用户和其他主体。

您可以自行决定添加对其他 Amazon Route 53 操作的访问权限，以使用户能够为路由控制创建运行状况检查。例如，您可以提供对以下一项或多项操作的访问权限：route53:GetHealthCheck、route53:CreateHealthCheck、route53:DeleteHealthCheck 和 route53:ChangeTagsForResource。

要查看此策略的权限，请参阅《AWS 托管策略参考》RecoveryControlConfigFullAccess中的 AmazonRoute53。

AWS 托管策略： AmazonRoute53 RecoveryControlConfigReadOnlyAccess

您可以将 AmazonRoute53RecoveryControlConfigReadOnlyAccess 附加到 IAM 实体。它适用于需要查看路由控制和安全规则配置的用户。此策略授予对 ARC 中使用恢复控制配置的操作的只读访问权限。这些用户无法创建、更新或删除恢复控制资源。

要查看此策略的权限，请参阅《AWS 托管策略参考》RecoveryControlConfigReadOnlyAccess中的AmazonRoute53。

AWS 托管策略： AmazonRoute53 RecoveryClusterFullAccess

您可以将 AmazonRoute53RecoveryClusterFullAccess 附加到 IAM 实体。此策略授予对 ARC 中使用集群数据平面操作的完全访问权限。将此策略附加到需要更新和检索路由控制状态的完全访问权限的 IAM 用户和其他主体。

要查看此策略的权限，请参阅《AWS 托管策略参考》RecoveryClusterFullAccess中的AmazonRoute53。

AWS 托管策略： AmazonRoute53 RecoveryClusterReadOnlyAccess

您可以将 AmazonRoute53RecoveryClusterReadOnlyAccess 附加到 IAM 实体。此策略授予对 ARC 中集群数据平面的只读访问权限。这些用户可以检索路由控制状态，但无法更新这些状态。

要查看此策略的权限，请参阅《AWS 托管策略参考》RecoveryClusterReadOnlyAccess中的AmazonRoute53。

路由控制 AWS 托管策略更新

有关自该服务开始跟踪这些更改以来，ARC 中路由控制 AWS 托管策略更新的详细信息，请参阅Amazon 应用程序恢复控制器 (ARC) AWS 托管策略的更新。要获得有关此页面变更的自动提醒，请订阅 ARC 文档历史记录页面上的 RSS 提要。

路由控制配额

Amazon 应用程序恢复控制器 (ARC) 中的路由控制受以下配额（以前称为限制）的约束。

实体	配额
每个账户的集群数	2

实体	配额
每个集群的控制面板数	50
每个控制面板的路由控制数	100
每个集群的路由控制总数（在所有控制面板中）	300
每个控制面板的安全规则数	20
每次工UpdateRoutingControlStates序调用的路由控制数量	10
每秒对集群端点的可变 API 调用次数	3

ARC 中的准备情况检查

通过 Amazon 应用程序恢复控制器 (ARC) 中的准备情况检查，您可以深入了解您的应用程序和资源是否为恢复做好了准备。在 ARC 中对 AWS 应用程序进行建模并创建就绪检查后，这些检查会持续监控有关您的应用程序的信息，例如 AWS 资源配额、容量和网络路由策略。然后，您可以选择收到有关更改的通知，这些更改会影响您故障转移到应用程序副本以及从事件中恢复的能力。就绪性检查有助于确保您可以持续地将多区域应用程序保持在经过扩展和配置以处理故障转移流量的状态。

本章介绍如何在 ARC 中对应用程序进行建模，通过创建恢复组和描述应用程序的单元来设置使准备就绪检查起作用的结构。然后，您可以按照步骤添加就绪检查和就绪范围，以便ARC可以审计您的应用程序的准备情况。

创建就绪检查后，您可以监控资源的就绪状态。就绪性检查可帮助您确保备用应用程序副本及其资源持续与您的生产副本相匹配，从而反映生产应用程序的容量、路由策略和其他配置细节。如果副本不匹配，则可以增加容量或更改配置，以便应用程序副本重新对齐。

> **Important**
>
> 就绪检查非常有助于持续验证应用程序副本配置和运行时状态是否一致。您不该使用就绪检查来指示生产副本是否正常，也不该依赖就绪检查作为灾难事件期间失效转移的主要触发条件。

Amazon 应用程序恢复控制器 (ARC) 中的准备情况检查是什么？

ARC 中的准备情况检查持续（每隔一分钟），审核检查中包含的资源的 AWS 预配置容量、服务配额、油门限制以及配置和版本差异方面的不匹配情况。就绪检查可以将这些差异通知给您，这样您就可以确保每个副本具有相同的配置设置和相同的运行时状态。尽管就绪检查可确保您在副本之间配置的容量一致，但不应期待就绪检查能代表您决定副本的容量应该是多少。例如，您应该了解应用程序需求，在设定自动扩缩组的大小时在每个副本中留足缓冲容量，以应对另一个单元格不可用的情况。

对于配额，当 ARC 通过就绪检查检测到不匹配时，它可以采取措施通过增加较低的配额以匹配较高的配额来调整副本的配额。当限额匹配时，就绪检查状态显示 READY。（请注意，这个过程不是立即更新的，总时间取决于特定的资源类型和其他因素。）

第一步是设置就绪检查，以创建代表应用程序的恢复组。每个恢复组都包括应用程序的每个故障控制单位或副本的对应单元格。接下来，为应用程序中的每种资源类型创建资源集，并将就绪检查与资源集关联起来。最后，将资源与就绪范围相关联，这样您就可以获得恢复组（您的应用程序）或单个单元（副本，即区域或可用区（AZs））中资源的就绪状态。

就绪状态（即 READY 或 NOT READY）基于就绪检查范围内的资源和某一资源类型的规则集。每种资源类型都有一组就绪规则，ARC 检查使用这些规则来审计资源的准备情况。资源是否 READY 取决于每条就绪规则的定义方式。所有就绪规则都会评估资源，但有些规则会对资源进行比较，有些则会查看有关资源集中每种资源的具体信息。

通过添加就绪检查，您可以通过以下几种方式之一监控就绪状态：使用 EventBridge、在 ARC API 操作中或使用 ARC API 操作。 AWS Management Console您还可以在不同的上下文中监控资源的就绪状态，包括单元格的就绪情况和应用程序的就绪情况。使用 ARC 中的跨账户授权功能，可以更轻松地设置和监控来自单个 AWS 账户的分布式资源。

通过就绪性检查监控应用程序副本

ARC 通过使用就绪检查来审计您的应用程序副本，以确保每个副本具有相同的配置设置和相同的运行时状态。就绪性检查会持续审核应用程序的 AWS 资源容量、配置、 AWS 配额和路由策略，这些信息可用于帮助确保副本已准备好进行故障转移。就绪性检查可帮助您确保恢复环境已扩展并配置为在需要时进行故障切换。

以下各节提供了有关准备情况检查工作原理的更多详细信息。

就绪性检查和您的应用程序副本

为了做好恢复准备，您必须始终在副本中保持足够的备用容量，以吸收来自其他可用区或区域的故障转移流量。ARC 会持续（每分钟一次）检查您的应用程序，以确保您的预配置容量在所有可用区或区域之间相匹配。

例如，ARC 检查的容量包括亚马逊 EC2 实例数量、Aurora 读取和写入容量单位以及 Amazon EBS 卷大小。如果您在主副本中扩大资源值的容量，但忘记同时增加备用副本中的相应值，ARC 会检测到不匹配情况，以便您可以增加备用副本中的值。

> **Important**
>
> 就绪检查非常有助于持续验证应用程序副本配置和运行时状态是否一致。您不该使用就绪检查来指示生产副本是否正常，也不该依赖就绪检查作为灾难事件期间失效转移的主要触发条件。

在主动-备用配置中，您应该根据监控和运行状况检查系统来确定是否从某单元格或向某单元格进行失效转移，并考虑将就绪检查作为这些系统的补充服务。ARC 就绪检查的可用性不高，因此您不应依赖中断期间可访问的检查。此外，在灾难事件发生期间，所检查的资源也可能不可用。

您可以监控特定单元（AWS 区域或可用区）中应用程序资源的就绪状态，也可以监控整个应用程序的就绪状态。例如，通过在中创建规则，当准备情况检查状态更改为（变为）时`Not ready`，您会收到通知 EventBridge。有关更多信息，请参阅 在 Amazon 上使用 ARC 中的准备情况检查 EventBridge。您还可以在中查看就绪状态 AWS Management Console，或者使用 API 操作（例如）来查看就绪状态`get-recovery-readiness`。有关更多信息，请参阅 准备情况检查 API 操作。

准备情况检查的工作原理

ARC 通过使用就绪检查来审计您的应用程序副本，以确保每个副本具有相同的配置设置和相同的运行时状态。

例如，为了做好恢复准备，您必须始终保持足够的备用容量，以吸收来自其他可用区或区域的失效转移流量。ARC 会持续（每分钟一次）检查您的应用程序，以确保您的预配置容量在所有可用区或区域之间相匹配。例如，ARC 检查的容量包括亚马逊 EC2 实例数量、Aurora 读取和写入容量单位以及 Amazon EBS 卷大小。如果您在主副本中扩大资源值的容量，但忘记同时增加备用副本中的相应值，ARC 会检测到不匹配情况，以便您可以增加备用副本中的值。

> **Important**
>
> 就绪检查非常有助于持续验证应用程序副本配置和运行时状态是否一致。您不该使用就绪检查来指示生产副本是否正常，也不该依赖就绪检查作为灾难事件期间失效转移的主要触发条件。

在主动-备用配置中，您应该根据监控和运行状况检查系统来确定是否从某单元格或向某单元格进行失效转移，并考虑将就绪检查作为这些系统的补充服务。ARC 就绪检查的可用性不高，因此您不应依赖中断期间可访问的检查。此外，在灾难事件发生期间，所检查的资源也可能不可用。

您可以监控特定单元（AWS 区域或可用区）中应用程序资源的就绪状态，也可以监控整个应用程序的就绪状态。例如，通过在中创建规则，当准备情况检查状态更改为（变为）时Not ready，您会收到通知 EventBridge。有关更多信息，请参阅 在 Amazon 上使用 ARC 中的准备情况检查 EventBridge。您还可以在中查看就绪状态 AWS Management Console，或者使用 API 操作（例如）来查看就绪状态get-recovery-readiness。有关更多信息，请参阅 准备情况检查 API 操作。

就绪规则如何确定就绪状态

ARC 就绪性检查根据每种资源类型的预定义规则以及这些规则的定义方式来确定就绪状态。ARC 为其支持的每种资源类型都包含一组规则。例如，ARC 有针对 Amazon Aurora 集群、Auto Scaling 群组等的准备规则组。有些就绪规则会对一个资源集里的资源进行比较，有些则会查看有关资源集中每种资源的具体信息。

您无法添加、编辑或删除就绪规则或规则组。但是，您可以创建 Amazon CloudWatch 警报并创建准备情况检查以监控警报的状态。例如，您可以创建自定义 CloudWatch 警报来监控 Amazon EKS 容器服务，并创建就绪检查以审计警报的就绪状态。

您可以在创建资源集 AWS Management Console 时查看每种资源类型的所有就绪规则，也可以稍后通过导航到资源集的详细信息页面来查看就绪规则。您还可以在以下部分中查看就绪规则：ARC 中的准备规则。

当就绪检查使用一组规则审计一组资源时，每条规则的定义方式将决定所有资源的检查结果都是 READY 或 NOT READY，还是结果因资源而异。此外，您还可以通过多种方式查看就绪状态。例如，您可以查看资源集中一组资源的就绪状态，也可以查看恢复组或单元（即 AWS 区域或可用区，具体取决于您设置恢复组的方式）的就绪状态摘要。

每条规则的描述语言将说明在应用该规则时，它如何评估资源以确定就绪状态。规则定义为检查资源集中的每个资源或所有资源以确定就绪情况。具体而言，规则的工作原理如下：

- 规则检查资源集中的每个资源，以确保符合条件。
  - 如果所有资源都符合条件，则所有资源都设置为 READY。
  - 如果一个资源不符合，则该资源设置为 NOT READY，其他单元格仍然是 READY。

  例如：MskClusterState:检查每个 Amazon MSK 集群以确保其处于状态。ACTIVE
- 该规则检查资源集中的所有资源，以确保符合条件。
  - 如果符合条件，则所有资源都设置为 READY。
  - 如果有任何资源不符合条件，所有资源都设置为 NOT READY。

  例如：VpcSubnetCount:检查全部 VPC 子网，以确保它们拥有相同数量的子网。

- 非关键条件：该规则检查资源集中的所有资源，以确保符合条件。
  - 如果有任何资源不符合，就绪状态保持不变。有此行为的规则会在描述中包含一个注释。

  例如：ElbV2CheckAzCount:检查每个 Network Load Balancer，确保其仅连接到一个可用区。注意：该规则不影响就绪状态。

此外，ARC在配额方面采取了额外措施。如果就绪检查检测到任何受支持资源的服务配额（资源创建和操作的最大值）各单元之间存在不匹配的情况，ARC 会自动提高配额较低的资源的配额。这仅适用于限额（限制）。对于容量，您应该根据应用程序需求添加额外的容量。

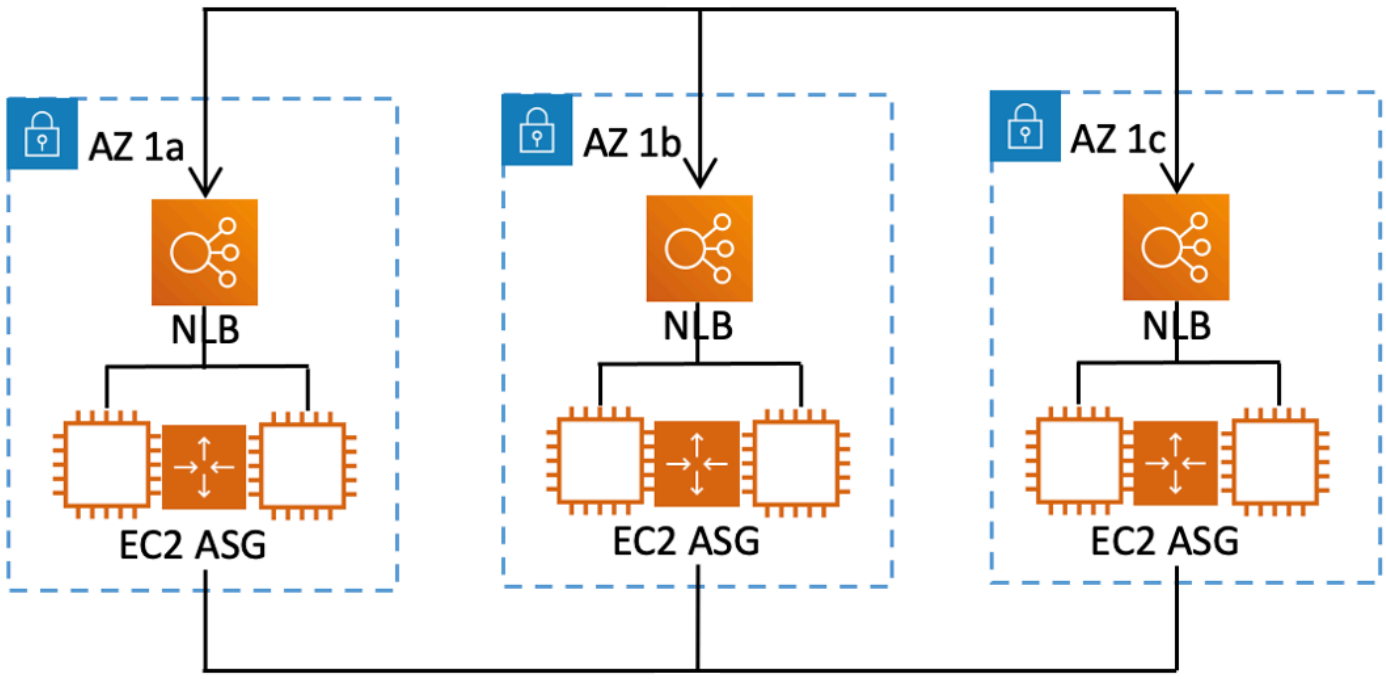
您还可以为准备情况检查设置 Amazon EventBridge 通知，例如，当任何准备情况检查状态更改为时NOT READY。然后，当检测到配置不匹配时， EventBridge 会向您发送通知，您可以采取更正措施来确保您的应用程序副本已对齐并做好恢复准备。有关更多信息，请参阅 在 Amazon 上使用 ARC 中的准备情况检查 EventBridge。

准备情况检查、资源集和就绪范围如何协同工作

就绪性检查始终会审计资源集中的资源组。您可以创建资源集（单独或在创建就绪检查时），以对 ARC 恢复组中单元（可用区或 AWS 区域）中的资源进行分组，以便可以定义就绪检查。资源集通常是一组相同类型的资源（如网络负载均衡器），但也可以是 DNS 目标资源（用于架构就绪检查）。

一般为应用程序中的每种资源创建一个资源集和就绪检查。对于架构就绪检查，您可以为其创建顶级 DNS 目标资源和全局（恢复组级别）资源集，然后为单独的资源集创建单元格级 DNS 目标资源。

下图显示了一个包含三个单元格（可用区）的恢复组示例，每个单元格都有一个网络负载均衡器 (NLB) 和自动扩缩组 (ASG)。



在这种情况下，您将为三个网络负载均衡器创建资源集和就绪检查，并为三个自动扩缩组创建资源集和就绪检查。现在，您可以按资源类型对恢复组的每个资源集进行就绪检查了。

通过为资源创建就绪范围，您可以为单元格或恢复组添加就绪检查摘要。要为资源指定就绪范围，请将单元格或恢复组的 ARN 与资源集中的每个资源关联起来。您可以在为资源集创建就绪检查时执行此操作。

例如，当您为该恢复组的网络负载均衡器资源集添加就绪检查时，可以同时向每个 NLB 添加就绪范围。在这种情况下，您可以将 AZ 1a 的 ARN 关联到 AZ 1a 中的 NLB，将 AZ 1b 的 ARN 关联到 AZ 1b 中的 NLB，将 AZ 1c 的 ARN 关联到 AZ 1c 中的 NLB。为自动扩缩组创建就绪检查时，您也要这样做，在为自动扩缩组资源集创建就绪检查时，为每个组分配就绪范围。

创建就绪检查时，关联就绪范围是可选操作，但是我们强烈建议您设置范围。就绪范围允许 ARC 显示恢复组摘要NOT READY就绪检查和单元级摘要就绪检查的正确状态READY或就绪状态。除非您设置就绪范围，否则 ARC 无法提供这些摘要。

请注意，在添加应用程序级资源或全局资源（例如 DNS 路由策略）时，不能为就绪范围选择恢复组或单元格，而是选择全局资源(不含单元格)。

DNS 目标资源就绪检查：审计弹性就绪

借助 ARC 中的 DNS 目标资源就绪性检查，您可以审核应用程序的架构和弹性准备情况。这种就绪检查会持续扫描应用程序架构和 Amazon Route 53 路由策略，以审计跨可用区和跨区域的依赖关系。

以恢复为导向的应用程序有多个副本，这些副本孤立在可用区或 AWS 区域中，因此副本可以相互独立地发生故障。如果您的应用程序需要调整以正确实现孤立，ARC 将建议您在需要时进行更改以更新您的架构，以帮助确保其具有弹性并可以进行故障转移。

ARC 会自动检测应用程序中单元的数量和范围（代表副本或故障控制单元），以及这些单元是按可用区还是按区域孤立。然后，ARC 会识别并向您提供有关单元中应用程序资源的信息，以确定它们是否正确地孤立到区域或区域。例如，如果单元格范围限定在特定可用区中，则就绪检查可以监控负载均衡器及其后面的目标是否也隔离到这些可用区。

利用这些信息，您可以确定是否需要进行更改，以使单元格中的资源对应到正确的可用区或区域。

首先，您需要为应用程序创建 DNS 目标资源及其资源集和就绪检查。有关更多信息，请参阅 在 ARC 中获取架构建议。

就绪检查和灾难恢复场景

ARC 就绪性检查可帮助您确保应用程序已扩展以处理故障转移流量，从而深入了解您的应用程序和资源是否已准备就绪，可以进行恢复。不应使用就绪检查状态作为指示生产副本是否正常的信号。但是，您可以使用就绪检查作为应用程序和基础架构监控或运行状况检查系统的补充，以确定是否从某副本或向某副本进行失效转移。

在紧急情况下或发生中断时，结合使用运行状况检查和其他信息来确定备用单元格是否已扩展、运行状况良好，并且准备好进行生产流量的失效转移。例如，除了验证备用单元格的就绪检查状态为 READY 之外，还要检查备用单元格上运行的金丝雀是否符合您的成功标准。

请注意，ARC 准备情况检查托管在美国西部（俄勒冈州）的单一 AWS 区域，在停电或灾难期间，准备情况检查信息可能会过时或支票可能不可用。有关更多信息，请参阅 用于路由控制的数据和控制平面。

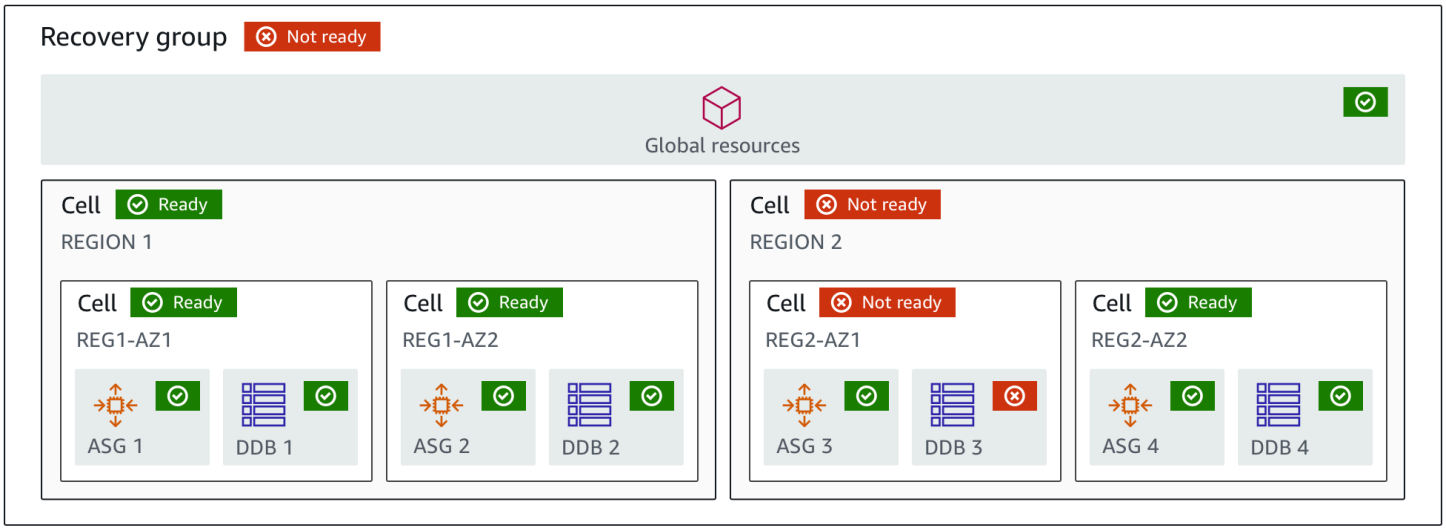
AWS 区域可用性以进行准备情况检查

有关亚马逊应用程序恢复控制器 (ARC) 的区域支持和服务终端节点的详细信息，请参阅《亚马逊网络服务通用参考》中的亚马逊应用程序恢复控制器 (ARC) 终端节点和配额。

> **Note**
>
> Amazon 应用程序恢复控制器 (ARC) 中的准备情况检查是一项全球功能。但是，就绪检查资源位于美国西部（俄勒冈）区域，因此在创建资源集和就绪检查等资源时，您必须在区域 ARC AWS CLI 命令中指定美国西部（俄勒冈`--region us-west-2`）区域（指定参数）。

就绪检查组件

下图展示了配置为支持就绪检查功能的恢复组示例。此示例中的资源在恢复组中按单元格（按 AWS 区域）和嵌套单元格（按可用区）分组。这里有恢复组（应用程序）的总体就绪状态，以及每个单元格（区域）和嵌套单元格（可用区）的个体就绪状态。



以下是 ARC 中准备就绪检查功能的组件。

单元格

单元格定义了应用程序的副本或独立的失效转移单位。它将应用程序在副本中独立运行所需的所有 AWS 资源进行分组。例如，您的主单元格中可能有一组资源，备用单元格中可能有另一组资源。您可以确定单元格所含内容的边界，但单元格通常代表可用区或区域。一个单元格中可以有多个单元格（嵌套单元格），例如 AZs 在一个区域内。每个嵌套单元格代表一个孤立的失效转移单位。

恢复组

单元格组合成一个恢复组。恢复组代表您要检查失效转移就绪情况的一个或一组应用程序。它由功能上彼此匹配的两个或多个单元格或副本组成。例如，如果您有一个在 us-east-1a 和 us-east-1b 之间复制的 Web 应用程序，其中 us-east-1b 是您的故障转移环境，则可以在 ARC 中将此应用

程序表示为包含两个单元的恢复组：一个在 us-east-1a 中，一个在 us-east-1a 中，另一个在 us-east-1b 中。恢复组还可以包括全局资源，例如 Route 53 运行状况检查。

资源和资源标识符

在 ARC 中创建用于就绪检查的组件时，您可以使用资源标识符指定资源，例如 Amazon DynamoDB 表、网络负载均衡器或 DNS 目标资源。资源标识符可以是资源的 Amazon 资源名称 (ARN)，对于 DNS 目标资源，则是 ARC 在创建资源时生成的标识符。

DNS 目标资源

DNS 目标资源是应用程序的域名和其他 DNS 信息（例如该域所指向的 AWS 资源）的组合。您可以选择是否包含 AWS 资源，但如果提供该资源，它必须是 Route 53 资源记录或网络负载均衡器。当您提供 AWS 资源时，您可以获得更详细的架构建议，这些建议可以帮助您提高应用程序的恢复弹性。您可以在 ARC 中为 DNS 目标资源创建资源集，然后为资源集创建就绪性检查，以便获得应用程序的架构建议。就绪检查还会根据 DNS 目标资源的就绪规则监控应用程序的 DNS 路由策略。

资源集

资源集是一组跨越多个单元的 AWS 资源，包括资源或 DNS 目标资源。例如，您可能有一个负载均衡器在 us-east-1a 中，还有一个在 us-east-1b 中。要监控负载均衡器的恢复就绪情况，您可以创建一个包含两个负载均衡器的资源集，然后为该资源集创建就绪检查。ARC 将持续检查集合中资源的准备情况。您还可以添加就绪范围，将资源集中的资源与您为应用程序创建的恢复组相关联。

就绪规则

就绪规则是 ARC 对资源集中的一组资源执行的审计。ARC 针对其支持准备情况检查的每种资源都有一套就绪规则。每条规则都包含一个 ID 和一个描述，用于说明 ARC 检查资源的目的。

就绪检查

准备情况检查会监控您的应用程序中的资源集，例如一组 Amazon Aurora 实例，ARC 正在审核其恢复准备情况。准备情况检查可以包括审计，例如容量配置、 AWS 配额或路由策略。例如，如果您想审核跨两个可用区的 Amazon A EC2 uto Scaling 组的准备情况，则可以为包含两个资源的资源集创建准备情况检查ARNs，每个 Auto Scaling 组对应一个资源。然后，为了确保每个组的比例相等，ARC 会持续监控两个组中的实例类型和计数。

就绪范围

就绪范围标识特定就绪检查所包含的资源分组。就绪检查的范围可以是恢复组（即整个应用程序全局）或单元格（即区域或可用区）。对于作为 ARC 全局资源的资源，请将就绪范围设置为恢复组或全局资源级别。例如，Route 53 运行状况检查是 ARC 中的一项全球资源，因为它不是特定于区域或可用区域的。

用于准备情况检查的数据和控制平面

在规划故障转移和灾难恢复时，请考虑故障转移机制的弹性。我们建议您确保在故障转移期间所依赖的机制具有高可用性，以便在灾难情况下可以根据需要使用它们。通常，应尽可能为机制使用数据平面函数，以获得最大的可靠性和容错性。考虑到这一点，请务必了解服务的功能如何在控制面板和数据面板之间划分，以及何时可以依赖服务的数据面板可预期的极高可靠性。

与大多数 AWS 服务一样，控制平面和数据平面支持就绪检查功能。虽然这两者都是为了可靠而构建的，但控制平面针对数据一致性进行了优化，而数据平面则针对可用性进行了优化。数据面板专为弹性而设计，因此即使在中断事件期间，当控制面板可能不可用时，它也能保持可用性。

一般而言，控制面板允许您执行基本的管理功能，例如在服务中创建、更新和删除资源。数据面板提供服务的核心功能。

对于准备情况检查，控制平面和数据平面只有一个 API，即恢复就绪 API。就绪检查和就绪资源仅位于美国西部（俄勒冈州）区域 (us-west-2)。就绪检查控制平面和数据平面可靠，但可用性不高。

有关数据平面、控制平面以及如何 AWS 构建服务以满足高可用性目标的更多信息，请参阅 Amazon Builders Library 中的 “使用可用区的静态稳定性” 论文。

在 Amazon 应用程序恢复控制器 (ARC) 中标记准备就绪性检查

标签是您用来识别和组织 AWS 资源的单词或短语（元数据）。您可以向每个资源添加多个标签，并且每个标签都包含您定义的一个键和一个值。例如，键可能是环境，值可能是生产。您可以根据添加的标签搜索和筛选您的资源。

您可以在 ARC 中将以下资源标记为准备情况检查：

- 资源集
- 就绪检查

ARC 中的标签只能通过 API 使用，例如，使用。 AWS CLI

以下是使用在准备情况检查中进行标记的示例。 AWS CLI

```
aws route53-recovery-readiness --region us-west-2 create-resource-
set --resource-set-name dynamodb_resource_set --resource-set-type
AWS::DynamoDB::Table --resources ReadinessScopes=arn:aws:aws-recovery-
readiness::111122223333:cell/PDXCell,ResourceArn=arn:aws:dynamodb:us-
```

```
west-2:111122223333:table/PDX_Table ReadinessScopes=arn:aws:aws-recovery-
readiness::111122223333:cell/IADCell,ResourceArn=arn:aws:dynamodb:us-
east-1:111122223333:table/IAD_Table --tags Stage=Prod

aws route53-recovery-readiness --region us-west-2 create-readiness-
check --readiness-check-name dynamodb_readiness_check --resource-set-name
dynamodb_resource_set --tags Stage=Prod
```

有关更多信息，请参阅 TagResourceAmazon 应用程序恢复控制器 (ARC) 的恢复就绪 API 参考指南。

ARC 中准备就绪检查的定价

您为配置的每项准备情况检查支付每小时费用。

有关 ARC 的详细定价信息和定价示例，请参阅 ARC 定价。

为您的应用程序设置弹性恢复流程

要将 Amazon Application Recovery Controller (ARC) 用于多个 AWS 区域的应用程序，需要遵循一些指导方针来设置应用程序的弹性，以便您可以有效地支持恢复就绪。 AWS 然后，您可以为应用程序创建就绪性检查，并设置路由控制以重新路由流量以进行故障转移。您还可以查看 ARC 提供的有关您的应用程序架构的建议，这些建议可以提高弹性。

> **Note**
>
> 如果您的应用程序被可用区隔开，请考虑使用区域转移或区域自动切换进行故障转移恢复。无需进行任何设置即可使用区域切换或区域自动切换来可靠地从可用区损坏中恢复应用程序。要将流量从可用区域移出负载均衡器资源，请在 ARC 控制台或 Elastic Load Balancing 控制台中开始区域切换。或者，您可以将 AWS Command Line Interface 或 AWS SDK 与区域移动 API 操作配合使用。有关更多信息，请参阅 ARC 中的区域偏移。

要了解有关弹性故障转移配置入门的更多信息，请参阅开始使用 Amazon 应用程序恢复控制器 (ARC) 中的多区域恢复。

ARC 中准备就绪检查的最佳实践

我们建议使用以下最佳实践来检查 Amazon 应用程序恢复控制器 (ARC) 的准备情况。

添加就绪状态变更通知

在 Amazon 中设置规则， EventBridge 以便在准备情况检查状态发生变化时发送通知，例如从变READY为NOT READY。收到通知后，您可以调查并解决问题，以确保您的应用程序和资源按照预期准备好进行失效转移。

您可以设置 EventBridge 规则以发送多项就绪检查状态更改的通知，包括恢复组（针对您的应用程序）、单元（例如 AWS 区域）或资源集的就绪性检查的通知。

有关更多信息，请参阅 在 Amazon 上使用 ARC 中的准备情况检查 EventBridge。

准备情况检查 API 操作

下表列出了可用于恢复就绪（就绪检查）的 ARC 操作以及相关文档的链接。

有关如何在 AWS Command Line Interface中使用常见恢复就绪 API 操作的示例，请参阅 使用 ARC 就绪检查 API 操作的示例 AWS CLI。

操作	使用 ARC 控制台	使用 ARC API
创建单元格	请参阅 在 ARC 中创建、更新和删除恢复组。	请参阅 CreateCell
获取单元格	请参阅 在 ARC 中创建、更新和删除恢复组。	请参阅 GetCell
删除单元格	请参阅 在 ARC 中创建、更新和删除恢复组。	请参阅 DeleteCell
更新单元格	不适用	请参阅 UpdateCell。
列出账户的单元格	请参阅 在 ARC 中创建、更新和删除恢复组。	请参阅 ListCells
创建恢复组	请参阅 在 ARC 中创建、更新和删除恢复组。	请参阅 CreateRecoveryGroup
获取恢复组	请参阅 在 ARC 中创建、更新和删除恢复组。	请参阅 GetRecoveryGroup
更新恢复组	请参阅 在 ARC 中创建、更新和删除恢复组。	请参阅 UpdateRecoveryGroup

操作	使用 ARC 控制台	使用 ARC API
删除恢复组	请参阅 在 ARC 中创建、更新和删除恢复组。	请参阅 DeleteRecoveryGroup
列出恢复组	请参阅 在 ARC 中创建、更新和删除恢复组。	请参阅 ListRecoveryGroups
创建资源集	请参阅 在 ARC 中创建和更新准备情况检查。	请参阅 CreateResourceSet
获取资源集	请参阅 在 ARC 中创建和更新准备情况检查。	请参阅 GetResourceSet
更新资源集	请参阅 在 ARC 中创建和更新准备情况检查。	请参阅 UpdateResourceSet
删除资源集	请参阅 在 ARC 中创建和更新准备情况检查。	请参阅 DeleteResourceSet
列出资源集	请参阅 在 ARC 中创建和更新准备情况检查。	请参阅 ListResourceSets
创建就绪检查	请参阅 在 ARC 中创建和更新准备情况检查。	请参阅 CreateReadinessCheck
获取就绪检查	请参阅 在 ARC 中创建和更新准备情况检查。	请参阅 GetReadinessCheck
更新就绪检查	请参阅 在 ARC 中创建和更新准备情况检查。	请参阅 UpdateReadinessCheck
删除就绪检查	请参阅 在 ARC 中创建和更新准备情况检查。	请参阅 DeleteReadinessCheck
列出就绪检查	请参阅 在 ARC 中创建和更新准备情况检查。	请参阅 ListReadinessChecks
列出就绪规则	请参阅 ARC 中的就绪规则描述。	请参阅 ListRules

操作	使用 ARC 控制台	使用 ARC API
检查整个就绪检查的状态	请参阅 在 ARC 中监控就绪状态。	请参阅 GetReadinessCheckStatus
检查资源的状态	请参阅 在 ARC 中监控就绪状态。	请参阅 GetReadinessCheckResourceStatus
检查单元格的状态	请参阅 在 ARC 中监控就绪状态。	请参阅 GetCellReadinessSummary
检查恢复组的状态	请参阅 在 ARC 中监控就绪状态。	请参阅 GetRecoveryGroupReadinessSummary

使用 ARC 就绪检查 API 操作的示例 AWS CLI

本节介绍简单的应用程序示例，使用使用 Amazon 应用程序恢复控制器 (ARC) 中的准备情况检查功能，使用 API 操作。 AWS Command Line Interface 这些示例旨在帮助您基本了解如何使用 CLI 使用就绪检查功能。

在 ARC 审计中检查应用程序副本中是否存在资源不匹配的情况。要为应用程序设置就绪性检查，您必须在 ARC 单元中设置或建模您的应用程序资源，使其与您为应用程序创建的副本保持一致。然后，您可以设置就绪性检查来审计这些副本，以帮助您确保备用应用程序副本及其资源持续与生产副本相匹配

让我们来看一个简单的案例，其中有一个名为的应用程序 Simple-Service 目前在美国东部（弗吉尼亚北部）区域（us-east-1）运行。您还在美国西部（俄勒冈州）区域 (us-west-2) 有一个应用程序备用副本。在本例中，我们将配置就绪检查，以比较应用程序的这两个版本。这样，我们就可以确保美国西部（俄勒冈州）区域的备用副本在失效转移场景中能够准备好接收流量。

有关使用的更多信息 AWS CLI，请参阅《AWS CLI 命令参考》。有关就绪 API 操作的列表和指向更多信息的链接，请参阅 准备情况检查 API 操作。

ARC 中的信@@ 元代表故障边界（如可用区或区域），并被收集到恢复组中。恢复组代表您要检查失效转移就绪情况的应用程序。有关就绪检查组成部分的更多信息，请参阅就绪检查组件 。

> **Note**
>
> ARC 是一项支持多个终端节点的全球服务，AWS 区域 但您必须在大多数 ARC CLI 命令中指定美国西部（俄勒冈--region us-west-2）区域（即指定参数）。例如，创建诸如恢复组或准备情况检查之类的资源。

在我们的应用程序示例中，首先要为拥有资源的每个区域创建一个单元格。然后，创建一个恢复组，接着完成就绪检查的设置。

1. 创建单元格

1a. 创建 us-east-1 单元格。

```
aws route53-recovery-readiness --region us-west-2 create-cell \
    --cell-name east-cell
```

```
{
    "CellArn": "arn:aws:route53-recovery-readiness::111122223333:cell/east-cell",
    "CellName": "east-cell",
    "Cells": [],
    "ParentReadinessScopes": [],
    "Tags": {}
}
```

1b. 创建 us-west-1 单元格。

```
aws route53-recovery-readiness --region us-west-2 create-cell \
    --cell-name west-cell
```

```
{
    "CellArn": "arn:aws:route53-recovery-readiness::111122223333:cell/west-cell",
    "CellName": "west-cell",
    "Cells": [],
    "ParentReadinessScopes": [],
    "Tags": {}
}
```

1c. 现在，我们有了两个单元格。您可以通过调用 `list-cells` API 来验证它们是否存在。

```
aws route53-recovery-readiness --region us-west-2 list-cells
```

```
{
    "Cells": [
        {
            "CellArn": "arn:aws:route53-recovery-readiness::111122223333:cell/east-
cell",
            "CellName": "east-cell",
            "Cells": [],
            "ParentReadinessScopes": [],
            "Tags": {}
        },
        {
            "CellArn": "arn:aws:route53-recovery-readiness::111122223333:cell/west-
cell",
            "CellName": "west-cell"
            "Cells": [],
            "ParentReadinessScopes": [],
            "Tags": {}
        }
    ]
}
```

2. 创建恢复组

恢复组是 ARC 中恢复准备的顶级资源。恢复组代表整个应用程序。在该步骤中，我们将创建一个恢复组，对整个应用程序进行建模，然后添加我们创建的两个单元格。

2a. 创建恢复组。

```
aws route53-recovery-readiness --region us-west-2 create-recovery-group \
    --recovery-group-name simple-service-recovery-group \
       --cells "arn:aws:route53-recovery-readiness::111122223333:cell/east-cell"\
       "arn:aws:route53-recovery-readiness::111122223333:cell/west-cell"
```

```
{
    "Cells": [],
    "RecoveryGroupArn": "arn:aws:route53-recovery-readiness::111122223333:recovery-
group/simple-service-recovery-group",
    "RecoveryGroupName": "simple-service-recovery-group",
    "Tags": {}
```

```
}
```

2b. （可选）您可以通过调用 `list-recovery-groups` API 来验证恢复组是否已正确创建。

```
aws route53-recovery-readiness --region us-west-2 list-recovery-groups
```

```
{
    "RecoveryGroups": [
        {
            "Cells": [
                "arn:aws:route53-recovery-readiness::111122223333:cell/east-cell",
                "arn:aws:route53-recovery-readiness::111122223333:cell/west-cell"
            ],
            "RecoveryGroupArn": "arn:aws:route53-recovery-
readiness::111122223333:recovery-group/simple-service-recovery-group",
            "RecoveryGroupName": "simple-service-recovery-group",
            "Tags": {}
        }
    ]
}
```

现在我们已经有了应用程序模型，接着添加要监控的资源。在 ARC 中，您要监控的一组资源称为资源集。资源集包含全部属于相同类型的资源。我们对资源集中的资源进行相互比较，以帮助确定单元格是否准备好进行失效转移。

3. 创建资源集

让我们假设我们的 Simple-Service 应用程序确实非常简单，只使用 DynamoDB 表。它在 us-east-1 中有一张 DynamoDB 表，在 us-west-2 中也有一张。资源集还包含就绪范围，用于标识每个资源包含在哪个单元格中。

3a. 创建反映我们的资源集 Simple-Service 应用程序的资源。

```
aws route53-recovery-readiness --region us-west-2 create-resource-set \
    --resource-set-name ImportantInformationTables \
    --resource-set-type AWS::DynamoDB::Table \
    --resources
    ResourceArn="arn:aws:dynamodb:us-west-2:111122223333:table/
TableInUsWest2",ReadinessScopes="arn:aws:route53-recovery-readiness::111122223333:cell/
west-cell"
```

```
    ResourceArn="arn:aws:dynamodb:us-west-2:111122223333:table/
TableInUsEast1",ReadinessScopes="arn:aws:route53-recovery-readiness::111122223333:cell/
east-cell"
```

```
{
    "ResourceSetArn": "arn:aws:route53-recovery-readiness::111122223333:resource-set/
sample-resource-set",
    "ResourceSetName": "ImportantInformationTables",
    "Resources": [
        {
            "ReadinessScopes": [
                "arn:aws:route53-recovery-readiness::111122223333:cell/west-cell"
            ],
            "ResourceArn": "arn:aws:dynamodb:us-west-2:111122223333:table/
TableInUsWest2"
        },
        {
            "ReadinessScopes": [
                "arn:aws:route53-recovery-readiness::111122223333:cell/east-cell"
            ],
            "ResourceArn": "arn:aws:dynamodb:us-west-2:111122223333:table/
TableInUsEast1"
        }
    ],
    "Tags": {}
}
```

3b.（可选）您可以通过调用 `list-resource-sets` API 来验证资源集中包含的资源。这列出了 AWS 账户的所有资源集。在这里，您可以看到我们只有上面创建的一个资源集。

```
aws route53-recovery-readiness --region us-west-2 list-resource-sets
```

```
{
    "ResourceSets": [
        {
            "ResourceSetArn": "arn:aws:route53-recovery-
readiness::111122223333:resource-set/ImportantInformationTables",
            "ResourceSetName": "ImportantInformationTables",
            "Resources": [
                {
                    "ReadinessScopes": [
```

```
                        "arn:aws:route53-recovery-readiness::111122223333:cell/west-
cell"
                    ],
                    "ResourceArn": "arn:aws:dynamodb:us-west-2:111122223333:table/
TableInUsWest2"
                },
                {
                    "ReadinessScopes": [
                        "arn:aws:route53-recovery-readiness::111122223333:cell/east-
cell"
                    ],
                    "ResourceArn": "arn:aws:dynamodb:us-west-2:111122223333:table/
TableInUsEast1"
                }
            ],
            "Tags": {}
        }
    ]
}{
    "ResourceSets": [
        {
            "ResourceSetArn": "arn:aws:route53-recovery-
readiness::111122223333:resource-set/ImportantInformationTables",
            "ResourceSetName": "ImportantInformationTables",
            "Resources": [
                {
                    "ReadinessScopes": [
                        "arn:aws:route53-recovery-readiness::111122223333:cell/west-
cell"
                    ],
                    "ResourceArn": "arn:aws:dynamodb:us-west-2:111122223333:table/
TableInUsWest2"
                },
                {
                    "ReadinessScopes": [
                        "arn:aws:route53-recovery-
readiness::&ExampleAWSAccountNo1;:cell/east-cell"
                    ],
                    "ResourceArn": "arn:aws:dynamodb:us-west-2:111122223333:table/
TableInUsEast1"
                }
            ],
            "Tags": {}
        }
```

```
    ]
}
```

现在，我们已经创建了单元、恢复组和资源集来建模 Simple-Service 在 ARC 中的应用。接下来，我们将设置就绪检查，以监控资源是否准备好进行失效转移。

4. 创建就绪检查

就绪检查将一组规则应用于附加到检查的资源集中的每个资源。规则特定于每种资源类型。也就是说，AWS::DynamoDB::Table、AWS::EC2::Instance 等有不同的规则。规则会从各个维度检查资源，包括配置、容量（如果可用而且适用）、限制（如果可用而且适用）和路由配置。

> **Note**
>
> 要查看就绪检查中应用于资源的规则，可以使用 get-readiness-check-resource-status API，如步骤 5 中所述。要查看 ARC 中所有就绪规则的列表，请使用list-rules或查看ARC 中的就绪规则描述。ARC 有一套针对每种资源类型运行的特定规则；这些规则目前不可自定义。

4a. 为资源集创建准备情况检查，ImportantInformationTables.

```
aws route53-recovery-readiness --region us-west-2 create-readiness-check \
    --readiness-check-name ImportantInformationTableCheck --resource-set-name
 ImportantInformationTables
```

```
{
    "ReadinessCheckArn": "arn:aws:route53-recovery-readiness::111122223333:readiness-
check/ImportantInformationTableCheck",
    "ReadinessCheckName": "ImportantInformationTableCheck",
    "ResourceSet": "ImportantInformationTables",
    "Tags": {}
}
```

4b.（可选）要验证是否已成功创建就绪检查，请运行 list-readiness-checks API。该 API 显示账户中的所有就绪检查。

```
aws route53-recovery-readiness --region us-west-2 list-readiness-checks
```

```
{
```

```
    "ReadinessChecks": [
        {
            "ReadinessCheckArn": "arn:aws:route53-recovery-
readiness::111122223333:readiness-check/ImportantInformationTableCheck",
            "ReadinessCheckName": "ImportantInformationTableCheck",
            "ResourceSet": "ImportantInformationTables",
            "Tags": {}
        }
    ]
}
```

5. 监控就绪检查

现在，我们已经对应用程序进行了建模并添加了就绪检查，接下来可以监控资源了。您可以在四个级别上对应用程序的就绪情况进行建模：就绪检查级别（一组资源）、单个资源级别、单元格级别（可用区或区域中的所有资源）和恢复组级别（整个应用程序）。下面提供了获取上述每种类型的就绪状态的命令。

5a. 查看就绪检查的状态。

```
aws route53-recovery-readiness --region us-west-2 get-readiness-check-status\
    --readiness-check-name ImportantInformationTableCheck
```

```
{
    "Readiness": "READY",
    "Resources": [
        {
            "LastCheckedTimestamp": "2021-01-07T00:53:39Z",
            "Readiness": "READY",
            "ResourceArn": "arn:aws:dynamodb:us-west-2:111122223333:table/
TableInUsWest2"
        },
        {
            "LastCheckedTimestamp": "2021-01-07T00:53:39Z",
            "Readiness": "READY",
            "ResourceArn": "arn:aws:dynamodb:us-west-2:111122223333:table/
TableInUsEast2"
    ]
}
```

5b. 查看就绪检查中单个资源的详细就绪状态，包括检查的每条规则的状态。

```
aws route53-recovery-readiness --region us-west-2 get-readiness-check-resource-status \
    --readiness-check-name ImportantInformationTableCheck \
    --resource-identifier "arn:aws:dynamodb:us-west-2:111122223333:table/
TableInUsWest2"
```


```
{"Readiness": "READY",
    "Rules": [
        {
            "LastCheckedTimestamp": "2021-01-07T00:55:41Z",
            "Messages": [],
            "Readiness": "READY",
            "RuleId": "DynamoTableStatus"
        },
        {
            "LastCheckedTimestamp": "2021-01-07T00:55:41Z",
            "Messages": [],
            "Readiness": "READY",
            "RuleId": "DynamoCapacity"
        },
        {
            "LastCheckedTimestamp": "2021-01-07T00:55:41Z",
            "Messages": [],
            "Readiness": "READY",
            "RuleId": "DynamoPeakRcuWcu"
        },
        {
            "LastCheckedTimestamp": "2021-01-07T00:55:41Z",
            "Messages": [],
            "Readiness": "READY",
            "RuleId": "DynamoGSIsPeakRcuWcu"
        },
        {
            "LastCheckedTimestamp": "2021-01-07T00:55:41Z",
            "Messages": [],
            "Readiness": "READY",
            "RuleId": "DynamoGSIsConfig"
        },
        {
            "LastCheckedTimestamp": "2021-01-07T00:55:41Z",
            "Messages": [],
            "Readiness": "READY",
            "RuleId": "DynamoGSIsStatus"
        },
```

```
        {
            "LastCheckedTimestamp": "2021-01-07T00:55:41Z",
            "Messages": [],
            "Readiness": "READY",
            "RuleId": "DynamoGSIsCapacity"
        },
        {
            "LastCheckedTimestamp": "2021-01-07T00:55:41Z",
            "Messages": [],
            "Readiness": "READY",
            "RuleId": "DynamoReplicationLatency"
        },
        {
            "LastCheckedTimestamp": "2021-01-07T00:55:41Z",
            "Messages": [],
            "Readiness": "READY",
            "RuleId": "DynamoAutoScalingConfiguration"
        },
        {
            "LastCheckedTimestamp": "2021-01-07T00:55:41Z",
            "Messages": [],
            "Readiness": "READY",
            "RuleId": "DynamoLimits"
        }
    ]
}
```

5c. 查看单元格的总体就绪情况。

```
aws route53-recovery-readiness --region us-west-2 get-cell-readiness-summary \
    --cell-name west-cell
```

```
{
    "Readiness": "READY",
    "ReadinessChecks": [
        {
            "Readiness": "READY",
            "ReadinessCheckName": "ImportantTableCheck"
        }
    ]
}
```

5d. 最后，查看恢复组级别上应用程序的顶级就绪情况。

```
aws route53-recovery-readiness --region us-west-2 get-recovery-group-readiness-summary
 \
    --recovery-group-name simple-service-recovery-group
```

```
{
    "Readiness": "READY",
    "ReadinessChecks": [
        {
            "Readiness": "READY",
            "ReadinessCheckName": "ImportantTableCheck"
        }
    ]
}
```

与恢复小组合作并进行准备情况检查

本节介绍并提供了恢复组和准备情况检查的过程，包括创建、更新和删除这些资源。

在 ARC 中创建、更新和删除恢复组

恢复组代表您在 Amazon 应用程序恢复控制器 (ARC) 中的应用程序。它通常由两个或多个单元格组成，这些单元格在资源和功能上是彼此的副本，因此您可以从一个单元格向另一个单元格进行失效转移。每个单元格都包含一个 AWS 区域或可用区域的活跃资源的 Amazon 资源名称 (ARNs)。资源可能是 Elastic Load Balancing 负载均衡器、自动扩缩组或其他资源。代表另一个可用区或区域的相应单元格包含与活动单元格类型相同的备用资源（负载均衡器、自动扩缩组等）。

单元格代表应用程序的副本。ARC 中的就绪检查可帮助您确定您的应用程序是否已准备好从一个副本故障转移到另一个副本。但是，您应该根据监控和运行状况检查系统来确定是否从某副本或向某副本进行失效转移，并考虑将就绪检查作为这些系统的补充服务。

就绪检查会审计资源，根据该类型资源的一组预定义规则来确定其就绪情况。使用副本创建恢复组后，您可以为应用程序中的资源添加 ARC 就绪性检查，这样 ARC 可以帮助确保副本在一段时间内具有相同的设置和配置。

主题

- 创建恢复组
- 更新和删除恢复组和单元

创建恢复组

本节中的步骤说明了如何在 ARC 控制台上创建恢复组。要了解如何将恢复就绪 API 操作与 Amazon 应用程序恢复控制器 (ARC) 配合使用，请参阅 准备情况检查 API 操作。

创建恢复组的步骤

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 选择准备情况检查。
3. 在恢复就绪页面上，选择创建，然后选择恢复组。
4. 输入恢复组的名称，然后选择下一步。
5. 选择创建单元格，然后选择添加单元格。
6. 输入单元格名称。例如，如果您在美国西部（北加利福尼亚）中有一个应用程序副本，则可以添加一个名为 `MyApp-us-west-1` 的单元格。
7. 选择添加单元格，然后为第二个单元格添加名称。例如，如果您在美国东部（俄亥俄州）中有一个副本，则可以添加一个名为 `MyApp-us-east-2` 的单元格。
8. 如果要添加嵌套单元格（区域内可用区中的副本），请选择操作，再选择添加嵌套单元格，然后输入名称。
9. 为应用程序副本添加了所有单元格和嵌套单元格后，请选择下一步。
10. 查看您的恢复组，然后选择创建恢复组。

更新和删除恢复组和单元

本节中的步骤说明如何在 ARC 控制台上更新和删除恢复组以及删除单元。要了解如何将恢复就绪 API 操作与 Amazon 应用程序恢复控制器 (ARC) 配合使用，请参阅 准备情况检查 API 操作。

更新或删除恢复组或删除单元格的步骤

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 选择准备情况检查。
3. 在恢复就绪页面上，选择一个恢复组。
4. 要操作恢复组，请选择操作，然后选择编辑恢复组或删除恢复组。
5. 编辑恢复组时，可以添加或删除单元格或嵌套单元格。
   - 要添加单元格，请选择添加单元格。
   - 要删除单元格，请在单元格旁边的操作标签下，选择删除单元格。

在 ARC 中创建和更新准备情况检查

本节提供准备情况检查和资源集的过程，包括创建、更新和删除这些资源。

创建和更新就绪检查

本节中的步骤说明了如何在 ARC 控制台上创建准备情况检查。要了解如何将恢复就绪 API 操作与 Amazon 应用程序恢复控制器 (ARC) 配合使用，请参阅 准备情况检查 API 操作。

要更新就绪检查，您可以编辑就绪检查的资源集，以添加或删除资源或者更改资源的就绪范围。

创建就绪检查的步骤

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 选择准备情况检查。
3. 在就绪页面上，选择创建，然后选择就绪检查。
4. 输入就绪检查的名称，选择要检查的资源类型，然后选择下一步。
5. 为就绪检查添加资源集。资源集是不同副本中相同类型的一组资源。选择下列选项之一：

   - 使用已创建资源集中的资源创建就绪检查。
   - 创建新的资源集。

   如果您选择创建新的资源集，请输入资源集的名称并选择添加。
6. 为要包含在集合中的每种资源逐一复制并粘贴 Amazon 资源名称 (ARNs)，然后选择 “下一步”。

   > **Tip**
   >
   > 有关 ARC 期望的每种资源类型的 ARN 格式的示例和更多信息，请参阅。ARC 中的资源类型和 ARN 格式
7. 如果你愿意，可以查看 ARC 检查你在此准备情况检查中包含的资源类型时将使用的就绪规则。然后选择下一步。
8. （可选）在恢复组名称下，选择要与就绪检查关联的恢复组，然后从资源所在的下拉菜单中为每个资源 ARN 选择一个单元格（区域或可用区）。如果它是应用程序级资源，比如 DNS 路由策略，请选择全局资源(不含单元格)。

   这一步为就绪检查中的资源指定了就绪范围。

> **Important**
>
> 尽管该步骤是可选的，但必须添加就绪范围才能获取恢复组和单元格的就绪摘要信息。如果您跳过此步骤，并且没有通过在此处选择就绪范围将就绪检查与恢复组的资源相关联，则 ARC 无法返回恢复组或单元的准备情况摘要信息。

9. 选择下一步。
10. 检查确认页面上的信息，然后选择创建就绪检查。

删除就绪检查的步骤

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 选择准备情况检查。
3. 选择就绪检查，然后在操作下选择删除。

创建和编辑资源集

通常情况下，在创建就绪检查的过程中创建资源集，但也可以单独创建资源集。您也可以编辑资源集，以添加或删除资源。本节中的步骤说明了如何在 ARC 控制台上创建或编辑资源集。要了解如何将恢复就绪 API 操作与 Amazon 应用程序恢复控制器 (ARC) 配合使用，请参阅 准备情况检查 API 操作。

创建资源集的步骤

1. 在https://console.aws.amazon.com/route53/家中打开 Route 53 控制台。
2. 在应用程序恢复控制器下，选择资源集。
3. 选择创建。
4. 输入资源集的名称，然后选择要包含在资源集中的资源类型。
5. 选择添加，然后输入要添加到资源集的资源的 Amazon 资源名称 (ARN)。
6. 添加完资源后，选择创建资源集。

编辑资源集的步骤

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 选择准备情况检查。
3. 在 “资源集” 下，选择 “操作”，然后选择 “编辑”。

4. 请执行以下操作之一：
   - 要从资源集中删除资源，请选择删除。
   - 要向资源集中添加资源，请选择添加，然后输入该资源的 Amazon 资源名称 (ARN)。
5. 您还可以编辑资源的就绪范围，以便将资源与其他单元格关联起来，进行就绪检查。
6. 选择保存。

在 ARC 中监控就绪状态

您可以在 Amazon 应用程序恢复控制器 (ARC) 中按以下级别查看应用程序的准备情况：

- 资源集中资源的就绪检查级别
- 单个资源级别
- 可用区或 AWS 区域中所有资源的单元（应用程序副本）级别
- 整个应用程序的恢复组级别

您可以收到有关就绪状态变化的通知，也可以在 Route 53 控制台中或使用 ARC CLI 命令监控就绪状态的变化。

就绪状态通知

您可以使用 Amazon EventBridge 设置事件驱动的规则，以监控 ARC 资源并通知您有关就绪状态的变化。有关更多信息，请参阅 在 Amazon 上使用 ARC 中的准备情况检查 EventBridge。

在 ARC 控制台中监控就绪状态

以下过程介绍如何在中监控恢复准备情况 AWS Management Console。

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 选择准备情况检查。
3. 在就绪页面的恢复组下，查看每个恢复组（应用程序）的恢复组就绪状态。

   您还可以查看特定单元格或单个资源的就绪情况。

使用 CLI 命令监控就绪状态

本节提供了用于查看不同级别应用程序和资源的就绪状态的 AWS CLI 命令示例。

资源集的就绪情况

您为资源集（一组资源）创建的就绪检查的状态。

```
aws route53-recovery-readiness --region us-west-2 get-readiness-check-status --readiness-check-name ReadinessCheckName
```

单个资源的就绪情况

要在就绪检查中获取单个资源的状态，包括检查的每条就绪规则的状态，请指定就绪检查名称和资源 ARN。例如：

```
aws route53-recovery-readiness --region us-west-2 get-readiness-check-status --readiness-check-name ReadinessCheckName --resource-arn "arn:aws:dynamodb:us-west-2:111122223333:table/TableName"
```

单元格的就绪情况

一个单元格（即区域或可用区）的状态。

```
aws route53-recovery-readiness --region us-west-2 get-cell-readiness-summary --cell-name CellName
```

应用程序的就绪情况

整个应用程序（恢复组级别）的状态。

```
aws route53-recovery-readiness --region us-west-2 get-recovery-group-readiness-summary --recovery-group-name RecoveryGroupName
```

在 ARC 中获取架构建议

如果您已有应用程序，Amazon Application Recovery Controller (ARC) 可以评估您的应用程序架构和路由策略，为修改设计提供建议，以提高应用程序的恢复弹性。在 ARC 中创建代表您的应用程序的恢复组后，请按照本节中的步骤获取有关应用程序架构的建议。

如果您尚未为恢复组 DNS 目标资源指定目标资源，我们建议您指定一个，以便我们提供更详细的建议。当您提供其他信息时，ARC 可以为您提供更好的建议。例如，如果您输入 Amazon Route 53 资源记录或网络负载均衡器作为目标资源，ARC 可以提供有关您是否为恢复组创建了最佳单元数量的信息。

对于 DNS 目标资源，请注意以下几点：

- 仅为目标资源指定 Route 53 资源记录或网络负载均衡器。
- 仅为每个恢复组创建一个 DNS 目标资源。
- 建议：为每个单元格创建一个 DNS 目标资源。
- 在就绪检查中，将 DNS 目标资源组成一个资源集。

以下步骤说明了如何创建 DNS 目标资源以及如何获取适用于应用程序的架构建议。

获取架构更新建议的步骤

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 选择准备情况检查。
3. 在恢复组名称下，选择代表应用程序的恢复组。
4. 在恢复组的详细信息页面的操作菜单上，选择获取该恢复组的架构建议。
5. 如果您尚未创建 DNS 目标资源准备情况检查，请创建一个，以便 ARC 可以提供架构建议。选择创建 DNS 目标资源。

   有关 DNS 目标资源的更多信息，请参阅就绪检查组件 。
6. 要为 DNS 目标资源创建资源集，请创建就绪检查。输入就绪检查的名称，然后对于就绪检查类型，选择 DNS 目标资源。
7. 输入资源集的名称。
8. 输入应用程序的属性，包括 DNS 名称、托管区 ARN 和记录集 ID。

   > **Tip**
   >
   > 要查看托管区 ARN 的格式，请参阅ARC 中的资源类型和 ARN 格式中托管区的 ARN 格式。

   可选但强烈推荐的步骤是，选择添加可选属性，然后提供网络负载均衡器 ARN 或域的 Route 53 资源记录。
9. （可选）在恢复组配置中，为您的 DNS 目标资源选择一个单元格，以设置就绪范围。
10. 选择创建资源集。
11. 在恢复组的详细信息页面上，选择获取架构建议。ARC 在页面上显示一组建议。

查看建议列表。然后，您可以决定是否以及如何进行更改，以提高应用程序的恢复弹性。

在 ARC 中创建跨账户授权

您的资源可能分布在多个 AWS 账户中，这使得全面了解应用程序的运行状况变得困难。它还可能使获取快速决策所需的信息变得困难。为了帮助简化在 Amazon 应用程序恢复控制器 (ARC) 中检查准备情况，您可以使用跨账户授权。

ARC 中的跨账户授权与准备情况检查功能配合使用。通过跨账户授权，您可以使用一个中央 AWS 账户来监控位于多个 AWS 账户中的资源。在包含要监控的资源的每个账户中，您可以授权中央账户访问这些资源。然后，该中央账户可以为所有账户中的资源创建就绪检查，并且您可以从该中央账户监控失效转移就绪情况。

> **Note**
>
> 在控制台中不能设置跨账户授权。取而代之的是使用 ARC API 操作来设置和使用跨账户授权。为了帮助您入门，本节提供了 AWS CLI 命令示例。

假设某个应用程序有一个账户在美国西部（俄勒冈州）区域 (us-west-2) 拥有资源，还有一个账户在美国东部（弗吉尼亚州北部）区域 (us-east-1) 拥有您想要监控的资源。ARC 允许您使用跨账户授权从一个账户 us-west-2 监控两组资源。

例如，假设您有以下 AWS 账户：

- 美国西部账户：999999999999
- 美国东部账户：111111111111

在 us-east-1 账户 (111111111111) 中，我们可以启用跨账户授权，并为 us-west-2 IAM 账户中的（根）用户指定 Amazon 资源名称 (ARN)：`arn:aws:iam::999999999999:root`，从而允许 us-west-2 账户 (999999999999) 访问。创建授权后，us-west-2 账户便可将 us-east-1 拥有的资源添加到资源集中，并创建要对该资源集运行的就绪检查。

以下示例说明了如何为一个账户设置跨账户授权。您必须在每个拥有要在 ARC 中添加和监控的 AWS 资源的额外账户中启用跨账户授权。

> **Note**
>
> ARC 是一项全球服务，支持多个 AWS 区域的终端节点，但您必须在大多数 ARC CLI 命令中指定美国西部（俄勒冈`--region us-west-2`）区域（即指定参数）。

以下 AWS CLI 命令显示如何为此示例设置跨账户授权：

```
aws route53-recovery-readiness --region us-west-2 --profile profile-in-us-east-1-
account \
    create-cross-account-authorization --cross-account-authorization
 arn:aws:iam::999999999999:root
```

要禁用该授权，请执行以下操作：

```
aws route53-recovery-readiness --region us-west-2 --profile profile-in-us-east-1-
account \
    delete-cross-account-authorization --cross-account-authorization
 arn:aws:iam::999999999999:root
```

要在一个特定账户中查看所有您已提供跨账户授权的账户，请使用 `list-cross-account-authorizations` 命令。请注意，目前无法反向检查。也就是说，您无法在某账户资料中使用 API 操作来列出它已获得跨账户授权（以添加和监控资源）的所有账户。

```
aws route53-recovery-readiness --region us-west-2 --profile profile-in-us-east-1-
account \
    list-cross-account-authorizations
```

```
{
    "CrossAccountAuthorizations": [
        "arn:aws:iam::999999999999:root"
    ]
}
```

就绪规则、资源类型和 ARNS

本节包含有关准备规则描述、支持的资源类型以及您用于资源集的 Amazon 资源名称 (ARNs) 格式的参考信息。

ARC 中的就绪规则描述

本节列出了 Amazon 应用程序恢复控制器 (ARC) 支持的所有资源类型的就绪规则描述。要查看 ARC 支持的资源类型列表，请参阅ARC 中的资源类型和 ARN 格式。

您还可以在 ARC 控制台上或使用 API 操作查看就绪规则描述，方法是执行以下操作：

- 要在控制台中查看就绪规则，请按照以下过程中的步骤操作：在控制台上查看就绪规则。
- 要使用 API 查看就绪规则，请参阅ListRules操作。

主题

ARC 中的准备规则

本节列出了 ARC 支持的每种资源类型的一组就绪规则。

在浏览规则描述时，您可以看到大部分都包含词语检查所有或检查每个。要了解这些术语如何解释规则在准备情况检查背景下的工作原理，以及有关 ARC 如何设置就绪状态的其他详细信息，请参阅就绪规则如何确定就绪状态。

就绪规则

ARC 使用以下就绪规则对资源进行审计。

Amazon API Gateway 版本 1 阶段

- ApiGwV1ApiKeyCount：检查所有 API Gateway 阶段，确保它们关联的 API 密钥数量相同。
- ApiGwV1ApiKeySource：检查所有 API Gateway 阶段，确保它们的 `API Key Source` 值相同。
- ApiGwV1BasePath：检查所有 API Gateway 阶段，确保它们链接到相同的基本路径。
- ApiGwV1BinaryMediaTypes：检查所有 API Gateway 阶段，确保它们支持相同的二进制媒体类型。
- ApiGwV1CacheClusterEnabled：检查所有 API Gateway 阶段，确保全部启用或都没启用 `Cache Cluster`。
- ApiGwV1CacheClusterSize：检查所有 API Gateway 阶段，确保它们的 `Cache Cluster Size` 相同。如果有一个的值比较大，则其他标记为 NOT READY。
- ApiGwV1CacheClusterStatus：检查所有 API Gateway 阶段，确保 `Cache Cluster` 处于 AVAILABLE 状态。
- ApiGwV1DisableExecuteApiEndpoint：检查所有 API Gateway 阶段，确保全部禁用或都没禁用 `Execute API Endpoint`。

- ApiGwV1DomainName：检查所有 API Gateway 阶段，确保它们链接到相同的域名。
- ApiGwV1EndpointConfiguration：检查所有 API Gateway 阶段，确保它们链接到具有相同端点配置的域。
- ApiGwV1EndpointDomainNameStatus：检查所有 API Gateway 阶段，确保它们关联的域名处于 AVAILABLE 状态。
- ApiGwV1MethodSettings：检查所有 API Gateway 阶段，确保它们的 `Method Settings` 值相同。
- ApiGwV1MutualTlsAuthentication：检查所有 API Gateway 阶段，确保它们的 `Mutual TLS Authentication` 值相同。
- ApiGwV1Policy：检查所有 API Gateway 阶段，确保全部使用或都不使用 API 级别策略。
- ApiGwV1RegionalDomainName：检查所有 API Gateway 阶段，确保它们链接到相同的区域域名。注意：该规则不影响就绪状态。
- ApiGwV1ResourceMethodConfigs：检查所有 API Gateway 阶段，确保它们具有相似的资源层次结构，包括相关配置。
- ApiGwV1SecurityPolicy：检查所有 API Gateway 阶段，确保它们的 `Security Policy` 值相同。
- ApiGwV1Quotas：检查所有 API Gateway 组，确保它们符合由 Service Quotas 管理的限额（限制）。
- ApiGwV1UsagePlans：检查所有 API Gateway 阶段，确保它们链接到具有相同配置的 `Usage Plans`。

Amazon API Gateway 版本 2 阶段

- ApiGwV2ApiKeySelectionExpression：检查所有 API Gateway 阶段，确保它们的 `API Key Selection Expression` 值相同。
- ApiGwV2ApiMappingSelectionExpression：检查所有 API Gateway 阶段，确保它们的 `API Mapping Selection Expression` 值相同。
- ApiGwV2CorsConfiguration：检查所有 API Gateway 阶段，确保它们具有相同的 CORS 相关配置。
- ApiGwV2DomainName：检查所有 API Gateway 阶段，确保它们链接到相同的域名。
- ApiGwV2DomainNameStatus：检查所有 API Gateway 阶段，确保域名处于 AVAILABLE 状态。
- ApiGwV2EndpointType：检查所有 API Gateway 阶段，确保它们的 `Endpoint Type` 值相同。
- ApiGwV2Quotas：检查所有 API Gateway 组，确保它们符合由 Service Quotas 管理的限额（限制）。

- ApiGwV2MutualTlsAuthentication：检查所有 API Gateway 阶段，确保它们的 `Mutual TLS Authentication` 值相同。
- ApiGwV2ProtocolType：检查所有 API Gateway 阶段，确保它们的 `Protocol Type` 值相同。
- ApiGwV2RouteConfigs：检查所有 API Gateway 阶段，确保它们具有相同的路由层次结构和相同的配置。
- ApiGwV2RouteSelectionExpression：检查所有 API Gateway 阶段，确保它们的 `Route Selection Expression` 值相同。
- ApiGwV2RouteSettings：检查所有 API Gateway 阶段，确保它们的 `Default Route Settings` 值相同。
- ApiGwV2SecurityPolicy：检查所有 API Gateway 阶段，确保它们的 `Security Policy` 值相同。
- ApiGwV2StageVariables: 检查所有 API Gateway 阶段，确保它们的 `Stage Variables` 都与其他阶段相同。
- ApiGwV2ThrottlingBurstLimit：检查所有 API Gateway 阶段，确保它们的 `Throttling Burst Limit` 值相同。
- ApiGwV2ThrottlingRateLimit：检查所有 API Gateway 阶段，确保它们的 `Throttling Rate Limit` 值相同。

Amazon Aurora 集群

- RdsClusterStatus：检查每个 Aurora 集群，以确保其状态为 `AVAILABLE` 或 `BACKING-UP`。
- RdsEngineMode：检查所有 Aurora 集群，确保它们的 `Engine Mode` 值相同。
- RdsEngineVersion：检查所有 Aurora 集群，确保它们的 `Major Version` 值相同。
- RdsGlobalReplicaLag：检查每个 Aurora 集群，确保其 `Global Replica Lag` 小于 30 秒。
- RdsNormalizedCapacity：检查所有 Aurora 集群，确保其标准化容量占资源集中最大容量的 15% 以内。
- RdsInstanceType：检查所有 Aurora 集群，确保它们有相同的实例类型。
- RdsQuotas：检查所有 Aurora 集群，确保它们符合由 Service Quotas 管理的限额（限制）。

自动扩缩组

- AsgMinSizeAndMaxSize：检查所有自动扩缩组，确保它们的组大小下限和上限相同。
- AsgAZCount：检查所有自动扩缩组，确保其可用区数量相同。
- AsgInstanceTypes：检查所有自动扩缩组，确保它们有相同的实例类型。注意：该规则不影响就绪状态。
- AsgInstanceSizes：检查所有自动扩缩组，确保它们的实例大小相同。

- AsgNormalizedCapacity：检查所有自动扩缩组，确保其标准化容量占资源集中最大容量的 15% 以内。
- AsgQuotas：检查所有自动扩缩组，确保它们符合由 Service Quotas 管理的限额（限制）。

CloudWatch 警报

- CloudWatchAlarmState：检查 CloudWatch 警报以确保每个警报都未处于`ALARM`或`INSUFFICIENT_DATA`状态。

客户网关

- CustomerGatewayIpAddress：检查所有客户网关，确保它们的 IP 地址相同。
- CustomerGatewayState: 检查客户网关，确保每个网关都处于 `AVAILABLE` 状态。
- CustomerGatewayVPNType：检查所有客户网关，确保它们的 VPN 类型相同。

DNS target resources

- DnsTargetResourceHostedZoneConfigurationRule：检查所有 DNS 目标资源，确保它们具有相同的 Amazon Route 53 托管区 ID，并且每个托管区都不是私有的。注意：该规则不影响就绪状态。
- DnsTargetResourceRecordSetConfigurationRule：检查所有 DNS 目标资源，确保它们的资源记录缓存存活时间 (TTL) 相同， TTLs 并且小于或等于 300。
- DnsTargetResourceRoutingRule：检查每个与别名资源记录集关联的 DNS 目标资源，确保其将流量路由到目标资源上配置的 DNS 名称。注意：该规则不影响就绪状态。
- DnsTargetResourceHealthCheckRule：检查所有 DNS 目标资源，确保运行状况检查在适当时与其资源记录集相关联，而非相反情况。注意：该规则不影响就绪状态。

Amazon DynamoDB 表

- DynamoConfiguration：检查所有 DynamoDB 表，确保它们具有相同的密钥、属性、服务器端加密和流配置。
- DynamoTableStatus：检查每个 DynamoDB 表，确保其状态为 ACTIVE。
- DynamoCapacity：检查所有 DynamoDB 表，确保其预配置的读取容量和写入容量占资源集中最大容量的 20% 以内。
- DynamoPeakRcuWcu：检查每个 DynamoDB 表，确保其峰值流量与其他表类似，以保证预配置的容量。
- DynamoGsiPeakRcuWcu：检查每个 DynamoDB 表，确保其最大读取和写入容量与其他表类似，以保证预配置的容量。
- DynamoGsiConfig：检查所有具有全局二级索引的 DynamoDB 表，确保这些表使用相同的索引、键架构和投影。

- DynamoGsiStatus：检查所有具有全局二级索引的 DynamoDB 表，确保全局二级索引处于 ACTIVE 状态。
- DynamoGsiCapacity：检查所有具有全局二级索引的 DynamoDB 表，确保这些表的预配置 GSI 读取容量和 GSI 写入容量占资源集中最大容量的 20% 以内。
- DynamoReplicationLatency：检查所有作为全局表的 DynamoDB 表，确保它们的复制延迟相同。
- DynamoAutoScalingConfiguration：检查所有启用了自动扩缩的 DynamoDB 表，确保它们具有相同的最小容量、最大容量及目标读取和写入容量。
- DynamoQuotas：检查所有 DynamoDB 表，确保它们符合由 Service Quotas 管理的限额（限制）。

Elastic Load Balancing（经典负载均衡器）

- ElbV1CheckAzCount：检查每个经典负载均衡器，确保其仅连接到一个可用区。注意：该规则不影响就绪状态。
- ElbV1AnyInstances: 检查所有经典负载均衡器，确保它们至少有一个 EC2 实例。
- ElbV1AnyInstancesHealthy: 检查所有经典负载均衡器，确保它们至少有一个运行良好的 EC2 实例。
- ElbV1Scheme：检查所有经典负载均衡器，确保它们采用相同的负载均衡器方案。
- ElbV1HealthCheckThreshold：检查所有经典负载均衡器，确保它们的运行状况检查阈值相同。
- ElbV1HealthCheckInterval：检查所有经典负载均衡器，确保它们的运行状况检查间隔值相同。
- ElbV1CrossZoneRoutingEnabled：检查所有经典负载均衡器，确保它们具有相同的跨区域负载均衡值（ENABLED 或 DISABLED）。
- ElbV1AccessLogsEnabledAttribute：检查所有经典负载均衡器，确保它们的访问日志值相同（ENABLED 或 DISABLED）。
- ElbV1ConnectionDrainingEnabledAttribute：检查所有经典负载均衡器，确保它们的连接耗尽值相同（ENABLED 或 DISABLED）。
- ElbV1ConnectionDrainingTimeoutAttribute：检查所有经典负载均衡器，确保它们的连接耗尽超时值相同。
- ElbV1IdleTimeoutAttribute：检查所有经典负载均衡器，确保它们的空闲超时值相同。
- ElbV1ProvisionedCapacityLcuCount：检查所有预配置的 LCU 大于 10 的经典负载均衡器，确保它们占资源集中最高预配置 LCU 的 20% 以内。
- ElbV1ProvisionedCapacityStatus：检查每个经典负载均衡器的预配置容量状态，确保其值不是 DISABLED 或 PENDING。

Amazon EBS 卷

- EbsVolumeEncryption: 检查全部 EBS 卷以确保它们的加密值相同（启用或禁用）。
- EbsVolumeEncryptionDefault: 检查全部 EBS 卷以确保它们在默认情况下具有相同的加密值（启用或禁用）。
- EbsVolumeIops: 检查全部 EBS 卷，以确保它们的每秒输入/输出操作数 (IOPS) 相同。
- EbsVolumeKmsKeyId: 检查全部 EBS 卷以确保它们具有相同的默认 AWS KMS 密钥 ID。
- EbsVolumeMultiAttach: 检查全部 EBS 卷以确保它们在多重连接（启用或禁用）中具有相同的值。
- EbsVolumeQuotas: 检查全部 EBS 卷以确保它们符合 Service Quotas 设置的配额（限制）。
- EbsVolumeSize: 检查全部 EBS 卷以确保它们具有相同的可读大小。
- EbsVolumeState: 检查全部 EBS 卷以确保它们具有相同的卷状态。
- EbsVolumeType: 检查全部 EBS 卷以确保它们具有相同的卷类型。

AWS Lambda 函数

- LambdaMemorySize：检查所有 Lambda 函数，确保它们的内存大小相同。如果有一个内存比较多，则其他标记为 `NOT READY`。
- LambdaFunctionTimeout：检查所有 Lambda 函数，确保它们的超时值相同。如果有一个的值比较大，则其他标记为 `NOT READY`。
- LambdaFunctionRuntime：检查所有 Lambda 函数，确保它们都具有相同的运行时间。
- LambdaFunctionReservedConcurrentExecutions：检查所有 Lambda 函数，确保它们都具有相同的 `Reserved Concurrent Executions` 值。如果有一个的值比较大，则其他标记为 `NOT READY`。
- LambdaFunctionDeadLetterConfig：检查所有 Lambda 函数，确保它们全都定义或都没定义 `Dead Letter Config`。
- LambdaFunctionProvisionedConcurrencyConfig：检查所有 Lambda 函数，确保它们的 `Provisioned Concurrency` 值相同。
- LambdaFunctionSecurityGroupCount：检查所有 Lambda 函数，确保它们的 `Security Groups` 值相同。
- LambdaFunctionSubnetIdCount：检查所有 Lambda 函数，确保它们的 `Subnet Ids` 值相同。
- LambdaFunctionEventSourceMappingMatch：检查所有 Lambda 函数，确保它们之间所有选定的 `Event Source Mapping` 属性都匹配。
- LambdaFunctionLimitsRule：检查所有 Lambda 函数，确保它们符合由 Service Quotas 管理的限额（限制）。

网络负载均衡器和应用程序负载均衡器

- ElbV2CheckAzCount：检查每个网络负载均衡器，确保其仅连接到一个可用区。注意：该规则不影响就绪状态。
- ElbV2TargetGroupsCanServeTraffic：检查每个 Network Load Balancer 和 Application Load Balancer，确保其至少有一个运行良好的亚马逊 EC2 实例。
- ElbV2State：检查每个网络负载均衡器和应用程序负载均衡器，确保其处于 ACTIVE 状态。
- ElbV2IpAddressType：检查所有网络负载均衡器和应用程序负载均衡器，确保它们的 IP 地址类型相同。
- ElbV2Scheme：检查所有网络负载均衡器和应用程序负载均衡器，确保它们的方案相同。
- ElbV2Type：检查所有网络负载均衡器和应用程序负载均衡器，确保它们的类型相同。
- ElbV2S3LogsEnabled：检查所有网络负载均衡器和应用程序负载均衡器，确保它们的 Amazon S3 服务器访问日志值相同（ENABLED 或 DISABLED）。
- ElbV2DeletionProtection：检查所有网络负载均衡器和应用程序负载均衡器，确保它们的删除保护值相同（ENABLED 或 DISABLED）。
- ElbV2IdleTimeoutSeconds：检查所有网络负载均衡器和应用程序负载均衡器，确保它们的空闲时间秒数值相同。
- ElbV2HttpDropInvalidHeaders：检查所有网络负载均衡器和应用程序负载均衡器，确保它们的 HTTP 丢弃无效标题值相同。
- ElbV2Http2Enabled：检查所有网络负载均衡器和应用程序负载均衡器，确保它们的值相同 HTTP2 （启用或禁用）。
- ElbV2CrossZoneEnabled：检查所有网络负载均衡器和应用程序负载均衡器，确保它们具有相同的跨区域负载均衡值（ENABLED 或 DISABLED）。
- ElbV2ProvisionedCapacityLcuCount：检查所有预配置的 LCU 大于 10 的网络负载均衡器和应用程序负载均衡器，确保它们占资源集中最高预配置 LCU 的 20% 以内。
- ElbV2ProvisionedCapacityEnabled：检查所有网络负载均衡器和应用程序负载均衡器的预配置容量状态，确保其值不是 DISABLED 或 PENDING。

Amazon MSK 集群

- MskClusterClientSubnet：检查每个 MSK 集群，确保它只有两个或只有三个客户端子网。
- MskClusterInstanceType：检查所有 MSK 集群，确保它们具有相同的 Amazon EC2 实例类型。
- MskClusterSecurityGroups：检查所有 MSK 集群，确保它们具有相同的安全组。
- MskClusterStorageInfo：检查所有 MSK 集群，确保它们的 EBS 存储卷大小相同。如果有一个的值比较大，则其他标记为 NOT READY。

- MskClusterACMCertificate：检查所有 MSK 集群，确保它们具有相同的客户端授权证书列表。ARNs
- MskClusterServerProperties：检查所有 MSK 集群，确保它们的 `Current Broker Software Info` 值相同。
- MskClusterKafkaVersion：检查所有 MSK 集群，确保它们的 Kafka 版本相同。
- MskClusterEncryptionInTransitInCluster：检查所有 MSK 集群，确保它们的 `Encryption In Transit In Cluster` 值相同。
- MskClusterEncryptionInClientBroker：检查所有 MSK 集群，确保它们的 `Encryption In Transit Client Broker` 值相同。
- MskClusterEnhancedMonitoring：检查所有 MSK 集群，确保它们的 `Enhanced Monitoring` 值相同。
- MskClusterOpenMonitoringInJmx：检查所有 MSK 集群，确保它们的 `Open Monitoring JMX Exporter` 值相同。
- MskClusterOpenMonitoringInNode: 检查所有 MSK 集群，确保它们的 `Open Monitoring Not Exporter.` 值相同。
- MskClusterLoggingInS3：检查所有 MSK 集群，确保它们的 `Is Logging in S3` 值相同。
- MskClusterLoggingInFirehose：检查所有 MSK 集群，确保它们的 `Is Logging In Firehose` 值相同。
- MskClusterLoggingInCloudWatch：检查所有 MSK 集群，确保它们的 `Is Logging Available In CloudWatch Logs` 值相同。
- MskClusterNumberOfBrokerNodes：检查所有 MSK 集群，确保它们的 `Number of Broker Nodes` 值相同。如果有一个的值比较大，则其他标记为 NOT READY。
- MskClusterState：检查每个 MSK 集群，确保其处于 ACTIVE 状态。
- MskClusterLimitsRule：检查所有 Lambda 函数，确保它们符合由 Service Quotas 管理的限额（限制）。

Amazon Route 53 运行状况检查

- R53HealthCheckType：检查每个 Route 53 运行状况检查，确保其类型不是 CALCULATED，并且所有检查的类型都相同。
- R53HealthCheckDisabled：检查每个 Route 53 运行状况检查，确保其不处于 DISABLED 状态。
- R53HealthCheckStatus：检查每个 Route 53 运行状况检查，确保其处于 SUCCESS 状态。
- R53HealthCheckRequestInterval：检查所有 Route 53 运行状况检查，确保它们都具有相同的 `Request Interval` 值。

- R53HealthCheckFailureThreshold：检查所有 Route 53 运行状况检查，确保它们都具有相同的 `Failure Threshold.` 值。
- R53HealthCheckEnableSNI：检查所有 Route 53 运行状况检查，确保它们都具有相同的 `Enable SNI.` 值。
- R53HealthCheckSearchString：检查所有 Route 53 运行状况检查，确保它们都具有相同的 `Search String.` 值。
- R53HealthCheckRegions：检查所有 Route 53 运行状况检查，确保它们都有相同的 AWS 区域列表。
- R53HealthCheckMeasureLatency：检查所有 Route 53 运行状况检查，确保它们都具有相同的 `Measure Latency` 值。
- R53HealthCheckInsufficientDataHealthStatus：检查所有 Route 53 运行状况检查，确保它们都具有相同的 `Insufficient Data Health Status` 值。
- R53HealthCheckInverted：检查所有 Route 53 运行状况检查，确保它们全都倒置或全未倒置。
- R53HealthCheckResourcePath：检查所有 Route 53 运行状况检查，确保它们都具有相同的 `Resource Path` 值。
- R53HealthCheckCloudWatchAlarm：检查所有 Route 53 运行状况检查，确保与之关联的 CloudWatch 警报具有相同的设置和配置。

Amazon SNS 订阅

- SnsSubscriptionProtocol：检查所有 SNS 订阅，确保它们的协议相同。
- SnsSubscriptionSqsLambdaEndpoint：检查所有包含 Lambda 或 SQS 端点的 SNS 订阅，确保它们包含不同的端点。
- SnsSubscriptionNonAwsEndpoint：检查所有具有非AWS 服务端点类型（例如电子邮件）的 SNS 订阅，以确保订阅具有相同的终端节点。
- SnsSubscriptionPendingConfirmation：检查所有 SNS 订阅，确保它们的“待确认”值相同。
- SnsSubscriptionDeliveryPolicy：检查所有使用 HTTP/S 的 SNS 订阅，确保它们的“有效传输期”值相同。
- SnsSubscriptionRawMessageDelivery：检查所有 SNS 订阅，确保它们的“原始消息传输”值相同。
- SnsSubscriptionFilter：检查所有 SNS 订阅，确保它们的“筛选策略”值相同。
- SnsSubscriptionRedrivePolicy：检查所有 SNS 订阅，确保它们的“重新驱动政策”值相同。
- SnsSubscriptionEndpointEnabled：检查所有 SNS 订阅，确保它们的“端点已启用”的值相同。
- SnsSubscriptionLambdaEndpointValid：检查所有包含 Lambda 端点的 SNS 订阅，确保它们包含有效的 Lambda 端点。

- SnsSubscriptionSqsEndpointValidRule：检查所有使用 SQS 端点的 SNS 订阅，确保它们包含有效的 SQS 端点。
- SnsSubscriptionQuotas：检查所有 SNS 订阅，确保它们符合由 Service Quotas 管理的限额（限制）。

Amazon SNS 主题

- SnsTopicDisplayName：检查所有 SNS 主题，确保它们的 `Display Name` 值相同。
- SnsTopicDeliveryPolicy：检查所有拥有 HTTPS 订阅用户的 SNS 主题，确保它们的 `EffectiveDeliveryPolicy` 相同。
- SnsTopicSubscription：检查所有 SNS 主题，确保每个协议的订阅用户数相同。
- SnsTopicAwsKmsKey：检查所有 SNS 主题，确保所有主题都有或都没有 AWS KMS 密钥。
- SnsTopicQuotas：检查所有 SNS 主题，确保它们符合由 Service Quotas 管理的限额（限制）。

Amazon SQS 队列

- SqsQueueType：检查所有 SQS 队列，确保它们都具有相同的 `Type` 值。
- SqsQueueDelaySeconds：检查所有 SQS 队列，确保它们都具有相同的 `Delay Seconds` 值。
- SqsQueueMaximumMessageSize：检查所有 SQS 队列，确保它们都具有相同的 `Maximum Message Size` 值。
- SqsQueueMessageRetentionPeriod：检查所有 SQS 队列，确保它们都具有相同的 `Message Retention Period` 值。
- SqsQueueReceiveMessageWaitTimeSeconds：检查所有 SQS 队列，确保它们都具有相同的 `Receive Message Wait Time Seconds` 值。
- SqsQueueRedrivePolicyMaxReceiveCount：检查所有 SQS 队列，确保它们都具有相同的 `Redrive Policy Max Receive Count` 值。
- SqsQueueVisibilityTimeout：检查所有 SQS 队列，确保它们都具有相同的 `Visibility Timeout` 值。
- SqsQueueContentBasedDeduplication：检查所有 SQS 队列，确保它们都具有相同的 `Content-Based Deduplication` 值。
- SqsQueueQuotas：检查所有 SQS 队列，确保它们符合由 Service Quotas 管理的限额（限制）。

Amazon VPCs

- VpcCidrBlock：检查所有内容 VPCs ，确保它们的 CIDR 块网络大小值相同。
- VpcCidrBlocksSameProtocolVersion：检查所有 VPCs 具有相同 CIDR 块的内容，确保它们的 Internet 流协议版本号值相同。

- VpcCidrBlocksStateInAssociationSets：检查所有的 CIDR 区块关联集， VPCs 确保它们都有处于状态的 CIDR 块。`ASSOCIATED`
- VpcIpv6CidrBlocksStateInAssociationSets：检查所有地址的所有 CIDR 块关联集， VPCs 确保它们的 CIDR 块都具有相同数量的地址。
- VpcCidrBlocksInAssociationSets：检查所有的 CIDR 区块关联集 VPCs ，确保它们的大小相同。
- VpcIpv6CidrBlocksInAssociationSets：检查所有的 IPv6 CIDR 区块关联集， VPCs 确保它们的大小相同。
- VpcState：检查每个 VPC，确保其处于 `AVAILABLE` 状态。
- VpcInstanceTenancy: 检查所有 VPCs 内容以确保它们都具有相同的值。`Instance Tenancy`
- VpcIsDefault: 检查所有 VPCs 内容以确保它们具有相同的值 `Is Default.`
- VpcSubnetState：检查每个 VPC 子网，确保其处于 AVAILABLE 状态。
- VpcSubnetAvailableIpAddressCount：检查每个 VPC 子网，确保其可用的 IP 地址数大于零。
- VpcSubnetCount：检查所有 VPC 子网，确保它们的子网数量相同。
- VpcQuotas：检查所有 VPC 子网，确保它们符合由 Service Quotas 管理的限额（限制）。

AWS VPN 连接

- VpnConnectionsRouteCount：检查所有 VPN 连接，确保它们至少有一条路由，而且路由数量相同。
- VpnConnectionsEnableAcceleration：检查所有 VPN 连接，确保它们的 `Enable Accelerations` 值相同。
- VpnConnectionsStaticRoutesOnly：检查所有 VPN 连接，确保它们的 `Static Routes Only.` 值相同。
- VpnConnectionsCategory：检查所有 VPN 连接，确保它们的类别为 `VPN`。
- VpnConnectionsCustomerConfiguration：检查所有 VPN 连接，确保它们的 `Customer Gateway Configuration` 值相同。
- VpnConnectionsCustomerGatewayId：检查每个 VPN 连接，确保它连接了客户网关。
- VpnConnectionsRoutesState：检查所有 VPN 连接，确保它们处于 `AVAILABLE` 状态。
- VpnConnectionsVgwTelemetryStatus：检查每个 VPN 连接，确保其 VGW 状态为 `UP`。
- VpnConnectionsVgwTelemetryIpAddress：检查每个 VPN 连接，确保其每个 VGW 遥测都有不同的外部 IP 地址。
- VpnConnectionsTunnelOptions：检查所有 VPN 连接，确保它们的隧道选项相同。

- VpnConnectionsRoutesCidr：检查所有 VPN 连接，确保它们的目标 CIDR 块相同。
- VpnConnectionsInstanceType：检查所有 VPN 连接，确保它们的 `Instance Type` 相同。

AWS VPN 网关

- VpnGatewayState：检查所有 VPN 网关，确保它们处于 AVAILABLE 状态。
- VpnGatewayAsn：检查所有 VPN 网关，确保它们的 ASN 相同。
- VpnGatewayType：检查所有 VPN 网关，确保它们的类型相同。
- VpnGatewayAttachment：检查所有 VPN 网关，确保它们的连接配置相同。

在控制台上查看就绪规则

您可以在上查看按每种资源类型列出的就绪规则。 AWS Management Console

在控制台上查看就绪规则的步骤

1. 打开 ARC 控制台，网址为https://console.aws.amazon.com/route53recovery/home#/dashboard。
2. 选择准备情况检查。
3. 在资源类型下，选择需要的资源类型以查看其规则。

ARC 中的资源类型和 ARN 格式

在 Amazon 应用程序恢复控制器 (ARC) 中创建资源集时，您可以指定要包含在集合中的资源类型以及要包含的每个资源的亚马逊资源名称 (ARNs)。ARC 期望每种资源类型都有特定的 ARN 格式。本节列出了 ARC 支持的资源类型以及每种资源的关联 ARN 格式。

具体格式取决于资源。当您提供 ARN 时，请用您的资源特定信息替换*italicized*文本。

> **Note**
>
> 请注意，ARC 要求的资源的 ARN 格式可能不同于服务本身为其资源所要求的 ARN 格式。例如，《服务授权参考》中每项服务的资源类型部分中描述的 ARN 格式可能不包括 ARC 支持 ARC 服务中的功能所需的 AWS 账户 ID 或其他信息。

AWS::ApiGateway::Stage

Amazon API Gateway 版本 1 阶段。

- ARN 格式：arn:*partition*:apigateway:*region*:*account*:/restapis/*api-id*/stages/*stage-name*

  示例：arn:aws:apigateway:us-east-1:111122223333:/restapis/123456789/stages/ExampleStage

  有关更多信息，请参阅 API Gateway Amazon 资源名称 (ARN) 参考。

AWS::ApiGatewayV2::Stage

Amazon API Gateway 版本 2 阶段。

- ARN 格式：arn:*partition*:apigateway:*region*:*account*:/apis/*api-id*/stages/*stage-name*

  示例：arn:aws:apigateway:us-east-1:111122223333:/apis/123456789/stages/ExampleStage

  有关更多信息，请参阅 API Gateway Amazon 资源名称 (ARN) 参考。

AWS::CloudWatch::Alarm

亚马逊 CloudWatch 警报。

- ARN 格式：arn:*partition*:cloudwatch:*region*:*account*:alarm:*alarm-name*

  示例：arn:aws:cloudwatch:us-west-2:111122223333:alarm:test-alarm-1

  有关更多信息，请参阅 Amazon 定义的资源类型 CloudWatch。

AWS::DynamoDB::Table

Amazon DynamoDB 表。

- ARN 格式：arn:*partition*:dynamodb:*region*:*account*:table/*table-name*

  示例：arn:aws:dynamodb:us-west-2:111122223333:table/BigTable

  有关更多信息，请参阅 DynamoDB 资源和操作。

AWS::EC2::CustomerGateway

客户网关设备。

- ARN 格式：arn:*partition*:ec2:*region*:*account*:customer-gateway/*CustomerGatewayId*

  示例：arn:aws:ec2:us-west-2:111122223333:customer-gateway/vcg-123456789

有关更多信息，请参阅 Amazon 定义的资源类型 EC2。

AWS::EC2::Volume

Amazon EBS 卷。

- ARN 格式：arn:*partition*:ec2:*region*:*account*:volume/*VolumeId*

  示例：arn:aws:ec2:us-west-2:111122223333:volume/volume-of-cylinder-is-pi

  有关更多信息，请参阅 API Gateway Amazon 资源名称 (ARN) 参考。

AWS::ElasticLoadBalancing::LoadBalancer

经典负载均衡器。

- ARN 格式：arn:*partition*:elasticloadbalancing:*region*:*account*:loadbalancer/*LoadBalanc*

  示例：arn:aws:elasticloadbalancing:us-west-2:111122223333:loadbalancer/123456789abcbdeCLB

  有关更多信息，请参阅 Elastic Load Balancing 资源。

AWS::ElasticLoadBalancingV2::LoadBalancer

网络负载均衡器或应用程序负载均衡器。

- 网络负载均衡器的 ARN 格式：arn:*partition*:elasticloadbalancing:*region*:*account*:loadbalancer/net/*LoadBalancerName*

  网络负载均衡器示例：arn:aws:elasticloadbalancing:us-west-2:111122223333:loadbalancer/net/sandbox-net/123456789acbdeNLB

- 应用程序负载均衡器的 ARN 格式：arn:*partition*:elasticloadbalancing:*region*:*account*:loadbalancer/app/*LoadBalancerName*

  应用程序负载均衡器示例：arn:aws:elasticloadbalancing:us-west-2:111122223333:loadbalancer/app/sandbox-alb/123456789acbdeALB

  有关更多信息，请参阅 Elastic Load Balancing 资源。

AWS::Lambda::Function

一个 AWS Lambda 函数。

- ARN 格式：arn:*partition*:lambda:*region*:*account*:function:*FunctionName*

  示例：arn:aws:lambda:us-west-2:111122223333:function:my-function

  有关更多信息，请参阅 Lambda 操作的资源和条件。

AWS::MSK::Cluster

Amazon MSK 集群。

- ARN 格式：arn:*partition*:kafka:*region*:*account*:cluster/*ClusterName*/*UUID*

  示例：arn:aws:kafka:us-east-1:111122223333:cluster/demo-cluster-1/123456-1111-2222-3333

  有关更多信息，请参阅 Amazon Managed Streaming for Apache Kafka 定义的资源类型。

AWS::RDS::DBCluster

Aurora 数据库集群。

- ARN 格式：arn:*partition*:rds:*region*:*account*:cluster:*DbClusterInstanceName*

  示例：arn:aws:rds:us-west-2:111122223333:cluster:database-1

  有关更多信息，请参阅在 Amazon RDS 中使用亚马逊资源名称 (ARNs)。

AWS::Route53::HealthCheck

Amazon Route 53 运行状况检查。

- ARN 格式：arn:*partition*:route53:::healthcheck/*Id*

  示例：arn:aws:route53:::healthcheck/123456-1111-2222-3333

AWS::SQS::Queue

Amazon SQS 队列。

- ARN 格式：arn:*partition*:sqs:*region*:*account*:*QueueName*

  示例：arn:aws:sqs:us-west-2:111122223333:StandardQueue

  有关更多信息，请参阅 Amazon Simple Queue Service 资源和操作。

AWS::SNS::Topic

Amazon SNS 主题。

- ARN 格式：arn:*partition*:sns:*region*:*account*:*TopicName*

  示例：arn:aws:sns:us-west-2:111122223333:TopicName

  有关更多信息，请参阅 Amazon SNS 资源 ARN 格式。

AWS::SNS::Subscription

Amazon SNS 订阅。

- ARN 格式：arn:*partition*:sns:*region*:*account*:*TopicName*:*SubscriptionId*

  示例：arn:aws:sns:us-west-2:111122223333:TopicName:12345678901234567890

AWS::EC2::VPC

虚拟私有云 (VPC)。

- ARN 格式：arn:*partition*:ec2:*region*:*account*:vpc/*VpcId*

  示例：arn:aws:ec2:us-west-2:111122223333:vpc/vpc-123456789

  有关更多信息，请参阅 VPC 资源。

AWS::EC2::VPNConnection

虚拟专用网络 (VPN) 连接。

- ARN 格式：arn:*partition*:ec2:*region*:*account*:vpn-connection/*VpnConnectionId*

  示例：arn:aws:ec2:us-west-2:111122223333:vpn-connection/vpn-123456789

  有关更多信息，请参阅 Amazon 定义的资源类型 EC2。

AWS::EC2::VPNGateway

虚拟专用网络 (VPN) 网关。

- ARN 格式：arn:*partition*:ec2:*region*:*account*:vpn-gateway/*VpnGatewayId*

  示例：arn:aws:ec2:us-west-2:111122223333:vpn-gateway/vgw-123456789acbdefgh

有关更多信息，请参阅 Amazon 定义的资源类型 EC2。

AWS::Route53RecoveryReadiness::DNSTargetResource

用于就绪检查的 DNS 目标资源包括 DNS 记录类型、域名、Route 53 托管区 ARN 以及网络负载均衡器 ARN 或 Route 53 记录集 ID。

- 托管区的 ARN 格式：`arn:partition:route53::account:hostedzone/Id`

  托管区示例：`arn:aws:route53::111122223333:hostedzone/abcHostedZone`

  注意：您必须按照此处指定的方式在托管区域 ARNs中包含账户 ID。必须提供账户 ID，这样 ARC 才能轮询资源。该格式故意不同于 Amazon Route 53 要求的 ARN 格式（在《服务授权参考》的 Route 53 服务资源类型中有描述）。
- 网络负载均衡器的 ARN 格式：`arn:partition:elasticloadbalancing:region:account:loadbalancer/net/LoadBalancerName`

  网络负载均衡器示例：`arn:aws:elasticloadbalancing:us-west-2:111122223333:loadbalancer/net/sandbox-net/123456789acbdefgh`

  有关更多信息，请参阅 Elastic Load Balancing 资源。

在 Amazon 应用程序恢复控制器 (ARC) 中记录和监控准备情况检查

您可以使用 Amazon CloudWatch 和 Amazon EventBridge 在 Amazon 应用程序恢复控制器 (ARC) 中监控准备情况检查，以分析模式并帮助解决问题。 AWS CloudTrail

> **Note**
>
> 无论是在控制台还是在使用时，您都必须在控制台中查看美国西部（俄勒冈）区域的 ARC CloudWatch 指标和日志 AWS CLI。使用时 AWS CLI，请通过包括以下参数为您的命令指定美国西部（俄勒冈）区域：`--region us-west-2`。

主题

- 在 ARC 中 CloudWatch 使用亚马逊进行准备情况检查
- 使用记录准备情况检查 API 调用 AWS CloudTrail
- 在 Amazon 上使用 ARC 中的准备情况检查 EventBridge

在 ARC 中 CloudWatch 使用亚马逊进行准备情况检查

亚马逊应用程序恢复控制器 (ARC) 将数据点发布到亚马逊， CloudWatch 供您检查准备情况。CloudWatch 允许您以一组有序的时间序列数据（称为指标）的形式检索有关这些数据点的统计信息。可将指标视为要监控的变量，而将数据点视为该变量随时间变化的值。例如，您可以监控指定时间段内通过某个 AWS 区域的流量。每个数据点都有相关联的时间戳和可选测量单位。

您可使用指标来验证系统是否正常运行。例如，您可以创建 CloudWatch 警报来监控指定的指标，并在该指标超出您认为可接受的范围时启动操作（例如向电子邮件地址发送通知）。

有关更多信息，请参阅 Amazon CloudWatch 用户指南。

主题

ARC 指标

AWS/Route53RecoveryReadiness 命名空间包括以下指标。

指标	描述
ReadinessChecks	表示 ARC 处理的准备情况检查的数量。该指标可以按状态确定维度，如下所示。 单位：Count。 报告标准：有非零值。 统计数据：唯一有用的统计数据是 Sum。 Dimensions • READY • NOT_READY • NOT_AUTHORIZED • UNKNOWN

指标	描述
Resources	表示 ARC 处理的资源数量，可以根据 API 定义的资源标识符来确定其尺寸。 单位：Count。 报告标准：有非零值。 统计数据：唯一有用的统计数据是 Sum。 Dimensions • ResourceSetType ：这些是资源类型，按 ARC 评估的每种给定类型的资源数量进行筛选 例如：AWS::CloudWatch::Alarm

ARC 指标的统计信息

CloudWatch 根据 ARC 发布的指标数据点提供统计信息。统计数据是在指定的时间段内汇总的指标数据。当请求统计数据时，返回的数据流按指标名称和维度进行识别。维度是用于唯一标识指标的名称/值对。

以下是您可能认为有用的指标/维度组合示例：

- 查看 ARC 评估的准备情况检查数量。
- 查看 ARC 评估的给定资源集类型的资源总数。

在 ARC 中查看 CloudWatch 指标

您可以使用 CloudWatch 控制台或查看 ARC 的 CloudWatch 指标 AWS CLI。在控制台中，这些指标显示为监控图表。

您必须在控制台中或使用时查看美国西部（俄勒冈）地区的 ARC CloudWatch 指标 AWS CLI。使用时 AWS CLI，请通过包括以下参数为您的命令指定美国西部（俄勒冈）区域：--region us-west-2。

使用 CloudWatch 控制台查看指标

1. 打开 CloudWatch 控制台，网址为https://console.aws.amazon.com/cloudwatch/。

2. 在导航窗格中，选择指标。
3. 选择 Route53 命RecoveryReadiness名空间。
4. (可选) 要跨所有维度查看某个指标，请在搜索字段中键入其名称。

要查看指标，请使用 AWS CLI

使用以下 list-metrics 命令列出可用指标：

```
aws cloudwatch list-metrics --namespace AWS/Route53RecoveryReadiness --region us-west-2
```

要获取指标的统计数据，请使用 AWS CLI

使用以下get-metric-statistics命令获取指定指标和维度的统计信息。请注意， CloudWatch 将每个唯一的维度组合视为一个单独的指标。您无法使用未专门发布的维度组合检索统计数据。您必须指定创建指标时使用的同一维度。

以下示例列出了 ARC 中某个账户每分钟评估的总准备情况检查。

```
aws cloudwatch get-metric-statistics --namespace AWS/Route53RecoveryReadiness \
--metric-name ReadinessChecks \
--region us-west-2 \
--statistics Sum --period 60 \
--dimensions Name=State,Value=READY \
--start-time 2021-07-03T01:00:00Z --end-time 2021-07-03T01:20:00Z
```

下面是该命令的示例输出：

```
{
    "Label": "ReadinessChecks",
    "Datapoints": [
        {
            "Timestamp": "2021-07-08T18:00:00Z",
            "Sum": 1.0,
            "Unit": "Count"
        },
        {
            "Timestamp": "2021-07-08T18:04:00Z",
            "Sum": 1.0,
            "Unit": "Count"
        },
        {
```

```
            "Timestamp": "2021-07-08T18:01:00Z",
            "Sum": 1.0,
            "Unit": "Count"
        },
        {
            "Timestamp": "2021-07-08T18:02:00Z",
            "Sum": 1.0,
            "Unit": "Count"
        },
        {
            "Timestamp": "2021-07-08T18:03:00Z",
            "Sum": 1.0,
            "Unit": "Count"
        }
    ]
}
```

使用记录准备情况检查 API 调用 AWS CloudTrail

与 AWS CloudTrail一项服务集成，该服务提供用户、角色或 AWS 服务在 ARC 中执行的操作的记录。CloudTrail 将 ARC 的所有 API 调用捕获为事件。捕获的调用包括来自 ARC 控制台的调用和对 ARC API 操作的代码调用。

如果您创建跟踪，则可以允许将 CloudTrail 事件持续传输到 Amazon S3 存储桶，包括 ARC 的事件。如果您未配置跟踪，您仍然可以在 CloudTrail 控制台的 “事件历史记录” 中查看最新的事件。

使用收集的信息 CloudTrail，您可以确定向 ARC 发出的请求、发出请求的 IP 地址、谁发出了请求、何时发出请求以及其他详细信息。

要了解更多信息 CloudTrail，请参阅《AWS CloudTrail 用户指南》。

ARC 信息在 CloudTrail

CloudTrail 在您创建账户 AWS 账户 时已在您的账户上启用。当活动在 ARC 中发生时，该活动会与其他 AWS 服务 CloudTrail 事件一起记录在事件历史记录中。您可以在中查看、搜索和下载最近发生的事件 AWS 账户。有关更多信息，请参阅使用 CloudTrail 事件历史记录。

要持续记录您的 AWS 账户事件（包括 ARC 的事件），请创建跟踪。跟踪允许 CloudTrail 将日志文件传输到 Amazon S3 存储桶。预设情况下，在控制台中创建跟踪记录时，此跟踪记录应用于所有 AWS 区域。跟踪记录 AWS 分区中所有区域的事件，并将日志文件传送到您指定的 Amazon S3 存储桶。此外，您可以配置其他 AWS 服务，以进一步分析和处理 CloudTrail 日志中收集的事件数据。有关更多信息，请参阅下列内容：

- 创建跟踪记录概述
- CloudTrail 支持的服务和集成
- 配置 Amazon SNS 通知 CloudTrail
- 接收来自多个区域的 CloudTrail 日志文件和接收来自多个账户的 CloudTrail 日志文件

所有 ARC 操作均由 CloudTrail 《亚马逊应用程序恢复控制器恢复准备 API 参考指南》、《亚马逊应用程序恢复控制器恢复控制器恢复控制配置 API 参考指南》和《亚马逊应用程序恢复控制器路由控制 API 参考指南》记录并记录在《亚马逊应用程序恢复控制器 API 参考指南》中。例如，调用`UpdateRoutingControlState`和`CreateRecoveryGroup`操作会在 CloudTrail 日志文件中生成条目。`CreateCluster`

每个事件或日志条目都包含有关生成请求的人员信息。身份信息有助于您确定以下内容：

- 请求是使用根证书还是 AWS Identity and Access Management (IAM) 用户凭证发出。
- 请求是使用角色还是联合用户的临时安全凭证发出的。
- 请求是否由其他 AWS 服务发出。

有关更多信息，请参阅 CloudTrail userIdentity 元素。

在事件历史记录中查看 ARC 事件

CloudTrail 允许您在事件历史记录中查看最近的事件。要查看 ARC API 请求的事件，您必须在控制台顶部的区域选择器中选择美国西部（俄勒冈）。有关更多信息，请参阅《AWS CloudTrail 用户指南》中的 “使用 CloudTrail 事件历史记录”。

了解 ARC 日志文件条目

跟踪是一种配置，允许将事件作为日志文件传输到您指定的 Amazon S3 存储桶。 CloudTrail 日志文件包含一个或多个日志条目。事件代表来自任何来源的单个请求，包括有关请求的操作、操作的日期和时间、请求参数等的信息。 CloudTrail 日志文件不是公共 API 调用的有序堆栈跟踪，因此它们不会按任何特定的顺序出现。

以下示例显示了一个 CloudTrail 日志条目，该条目演示了准备情况检查的`CreateRecoveryGroup`操作。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
```

```
        "principalId": "A1B2C3D4E5F6G7EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/admin",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROA33L3W36EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/admin",
                "accountId": "111122223333",
                "userName": "EXAMPLENAME"
            },
            "webIdFederationData": {},
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2021-07-06T17:38:05Z"
            }
        }
    },
    "eventTime": "2021-07-06T18:08:03Z",
    "eventSource": "route53-recovery-readiness.amazonaws.com",
    "eventName": "CreateRecoveryGroup",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.50",
    "userAgent": "Boto3/1.17.101 Python/3.8.10 Linux/4.14.231-180.360.amzn2.x86_64
 exec-env/AWS_Lambda_python3.8 Botocore/1.20.102",
    "requestParameters": {
        "recoveryGroupName": "MyRecoveryGroup"
    },
    "responseElements": {
        "Access-Control-Expose-Headers": "x-amzn-errortype,x-amzn-requestid,x-amzn-
errormessage,x-amzn-trace-id,x-amzn-requestid,x-amz-apigw-id,date",
        "cells": [],
        "recoveryGroupName": "MyRecoveryGroup",
        "recoveryGroupArn": "arn:aws:route53-recovery-readiness::111122223333:recovery-
group/MyRecoveryGroup",
        "tags": "***"
    },
    "requestID": "fd42dcf7-6446-41e9-b408-d096example",
    "eventID": "4b5c42df-1174-46c8-be99-d67aexample",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
```

```
    "recipientAccountId": "111122223333"
}
```

在 Amazon 上使用 ARC 中的准备情况检查 EventBridge

使用 Amazon EventBridge，您可以设置事件驱动的规则，监控您在亚马逊应用程序恢复控制器 (ARC) 中的准备情况检查资源，然后启动使用其他 AWS 服务的目标操作。例如，当准备情况检查状态从 “就绪” 变为 “未就绪” 时，您可以向 Amazon SNS 主题发送信号，从而设置发送电子邮件通知的规则。

> **Note**
>
> ARC 仅在美国西部（俄勒冈）（us-west- AWS 2）地区发布准备情况检查 EventBridge 活动。要接收 EventBridge 事件以进行准备情况检查，请在美国西部（俄勒冈）区域创建 EventBridge 规则。

您可以在 Amazon 中创建规则 EventBridge ，以便对以下 ARC 准备情况检查事件采取行动：

- 就绪检查就绪。该事件指定就绪检查状态是否发生变化，例如，从 READY 变为 NOT READY。

要捕获您感兴趣的特定 ARC 事件，请定义 EventBridge 可用于检测事件的特定事件模式。事件模式与它们匹配的事件具有相同的结构。模式引用了您要匹配的字段，并提供您所查找的值。

尽最大努力发出事件。在正常运行情况下，它们几乎实时 EventBridge 地从ARC交付。但是，可能会出现延迟或阻止事件交付的情况。

有关 EventBridge 规则如何处理事件模式的信息，请参阅中的事件和事件模式 EventBridge。

使用监控准备情况检查资源 EventBridge

借 EventBridge助，您可以创建规则，以定义 ARC 为准备情况检查资源发出事件时要采取的操作。

要在 EventBridge 控制台中键入或复制并粘贴事件模式，请在控制台中选择 “Enter my ow n” 选项。为了帮助您确定可能对您有用的事件模式，本主题包括准备事件模式示例。

要为资源事件创建规则

1. 打开 Amazon EventBridge 控制台，网址为https://console.aws.amazon.com/events/。
2. AWS 区域 要在中创建规则，请选择美国西部（俄勒冈）。这是就绪事件的必填区域。
3. 选择 Create rule (创建规则)。

4. 输入规则的 Name (名称) 和“Description (描述)”（可选）。
5. 对于事件总线，保留默认值，即默认。
6. 选择下一步。
7. 对于构建事件模式步骤，对于事件源，保留默认值，即 AWS 事件。
8. 在示例事件下，选择输入我自己的。
9. 对于示例事件，键入或复制并粘贴事件模式。有关示例，请参阅下一节。

就绪事件模式示例

事件模式与它们匹配的事件具有相同的结构。模式引用了您要匹配的字段，并提供您所查找的值。

您可以将本节中的事件模式复制并粘贴 EventBridge 到中，以创建可用于监控 ARC 操作和资源的规则。

以下事件模式提供了一些示例，您可以在 ARC EventBridge 的准备情况检查功能中使用这些示例。

- 从 ARC 准备情况检查中选择所有事件。

```
{
    "source": [
        "aws.route53-recovery-readiness"
    ]
}
```

- 仅选择与单元相关的事件。

```
{
    "source": [
        "aws.route53-recovery-readiness"
    ],
    "detail-type": [
        "Route 53 Application Recovery Controller cell readiness status change"
    ]
}
```

- 仅选择与名为 *MyExampleCell* 的特定单元相关的事件。

```
{
    "source": [
        "aws.route53-recovery-readiness"
```

```
    ],
    "detail-type": [
        "Route 53 Application Recovery Controller cell readiness status change"
    ],
    "resources": [
        "arn:aws:route53-recovery-readiness::111122223333:cell/MyExampleCell"
    ]
}
```

- 仅选择任何恢复组、单元或就绪检查状态变为 *NOT READY* 时的事件。

```
{
   "source":[
      "aws.route53-recovery-readiness"
   ],
   "detail-type":{
      "new-state":{
         "readiness-status":[
            "NOT_READY"
         ]
      }
   }
}
```

- 仅选择任何恢复组、单元或就绪检查变为 *READY* 以外状态时的事件。

```
{
   "source":[
      "aws.route53-recovery-readiness"
   ],
   "detail":{
      "new-state":{
         "readiness-status":[
            {
               "anything-but":"READY"
            }
         ]
      }
   }
}
```

以下是恢复组就绪状态更改的 ARC 事件示例：

```
{
    "version": "0",
    "account":"111122223333",
    "detail-type":"Route 53 Application Recovery Controller recovery group readiness
 status change",
    "source":"route53-recovery-readiness.amazonaws.com",
    "time":"2020-11-03T00:31:54Z",
    "id": "1234a678-1b23-c123-12fd3f456e78",
    "region": "us-west-2",
    "resources":[
        "arn:aws:route53-recovery-readiness::111122223333:recovery-group/BillingApp"
    ],
    "detail": {
        "recovery-group-name": "BillingApp",
        "previous-state": {
            "readiness-status": "READY|NOT_READY|UNKNOWN|NOT_AUTHORIZED"
        },
        "new-state": {
            "readiness-status": "READY|NOT_READY|UNKNOWN|NOT_AUTHORIZED"
        }
    }
}
```

以下是小区就绪状态更改的 ARC 事件示例：

```
{
    "version": "0",
    "account":"111122223333",
    "detail-type":"Route 53 Application Recovery Controller cell readiness status
 change",
    "source":"route53-recovery-readiness.amazonaws.com",
    "time":"2020-11-03T00:31:54Z",
    "id": "1234a678-1b23-c123-12fd3f456e78",
    "region": "us-west-2",
    "resources":[
        "arn:aws:route53-recovery-readiness::111122223333:cell/PDXCell"
    ],
    "detail": {
        "cell-name": "PDXCell",
        "previous-state": {
            "readiness-status": "READY|NOT_READY|UNKNOWN|NOT_AUTHORIZED"
        },
        "new-state": {
```

```
            "readiness-status": "READY|NOT_READY|UNKNOWN|NOT_AUTHORIZED"
        }
    }
}
```

以下是准备情况检查状态更改的 ARC 事件示例：

```
{
    "version": "0",
    "account":"111122223333",
    "detail-type":"Route 53 Application Recovery Controller readiness check status
 change",
    "source":"route53-recovery-readiness.amazonaws.com",
    "time":"2020-11-03T00:31:54Z",
    "id": "1234a678-1b23-c123-12fd3f456e78",
    "region": "us-west-2",
    "resources":[
        "arn:aws:route53-recovery-readiness::111122223333:readiness-check/
UserTableReadinessCheck"
    ],
    "detail": {
    "readiness-check-name": "UserTableReadinessCheck",
        "previous-state": {
            "readiness-status": "READY|NOT_READY|UNKNOWN|NOT_AUTHORIZED"
        },
        "new-state": {
            "readiness-status": "READY|NOT_READY|UNKNOWN|NOT_AUTHORIZED"
        }
    }
}
```

指定要用作目标的 CloudWatch 日志组

创建 EventBridge 规则时，必须指定将与该规则匹配的事件发送到哪个目标。有关可用目标的列表 EventBridge，请参阅 EventBridge 控制台中的可用目标。您可以添加到 EventBridge 规则的目标之一是 Amazon CloudWatch 日志组。本节介绍将 CloudWatch 日志组添加为目标的要求，并提供了在创建规则时添加日志组的过程。

要将 CloudWatch 日志组添加为目标，可以执行以下操作之一：

- 创建新的日志组
- 选择现有的日志组

如果您在创建规则时使用控制台指定了新的日志组，则 EventBridge 会自动为您创建该日志组。确保用作 EventBridge 规则目标的日志组以开头/aws/events。如果要选择现有的日志组，请注意，只有以开头的日志组才/aws/events会作为选项出现在下拉菜单中。有关更多信息，请参阅 Amazon CloudWatch 用户指南中的创建新日志组。

如果您使用控制台之外的 CloudWatch 操作创建或使用 CloudWatch 日志组作为目标，请确保正确设置权限。如果您使用控制台向 EventBridge 规则添加日志组，则该日志组的基于资源的策略会自动更新。但是，如果您使用 AWS Command Line Interface 或 S AWS DK 来指定日志组，则必须更新该日志组的基于资源的策略。以下示例策略说明了您必须在基于资源的策略中为日志组定义的权限：

```
{
  "Statement": [
    {
      "Action": [
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ],
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "events.amazonaws.com",
          "delivery.logs.amazonaws.com"
        ]
      },
      "Resource": "arn:aws:logs:region:account:log-group:/aws/events/*:*",
      "Sid": "TrustEventsToStoreLogEvent"
    }
  ],
  "Version": "2012-10-17"
}
```

您无法使用控制台为日志组配置基于资源的策略。要向基于资源的策略添加所需的权限，请使用 CloudWatch PutResourcePolicyAPI 操作。然后，您可以使用 describe-resource-policiesCLI 命令来检查您的策略是否已正确应用。

为资源事件创建规则并指定 CloudWatch 日志组目标

1. 打开 Amazon EventBridge 控制台，网址为https://console.aws.amazon.com/events/。
2. 选择 AWS 区域 要在其中创建规则的。
3. 选择 “创建规则”，然后输入有关该规则的任何信息，例如事件模式或计划详细信息。

有关创建就绪性 EventBridge 规则的更多信息，请参阅使用监控准备情况检查资源 EventBridge。

4. 在 “选择目标” 页面上，选择CloudWatch作为您的目标。
5. 从下拉菜单中选择一个 CloudWatch 日志组。

Identity and Access Management 用于准备情况检查

AWS Identity and Access Management (IAM) AWS 服务 可帮助管理员安全地控制对 AWS 资源的访问权限。IAM 管理员控制谁可以进行身份验证（登录）和授权（有权限）使用 ARC 资源。您可以使用 IAM AWS 服务 ，无需支付额外费用。

内容

- 准备情况如何登记 SERVICElong；与 IAM 配合使用
- 在 Amazon 应用程序恢复控制器 (ARC) 中进行准备情况检查的基于身份的策略示例
- 在 ARC 中使用服务关联角色进行准备情况检查
- AWS 在 Amazon 应用程序恢复控制器 (ARC) 中进行准备情况检查的托管策略

准备情况如何登记 SERVICElong；与 IAM 配合使用

在使用 IAM 管理对 ARC 的访问权限之前，请先了解有哪些 IAM 功能可用于 ARC。

在使用 IAM 管理对 Amazon 应用程序恢复控制器 (ARC) 中准备情况检查的访问权限之前，请先了解有哪些 IAM 功能可用于准备情况检查。

您可以在 Amazon 应用程序恢复控制器 (ARC) 中进行准备情况检查时使用的 IAM 功能

IAM 特征	准备情况检查支持
基于身份的策略	是
基于资源的策略	否
策略操作	是
策略资源	是
策略条件键	是
ACLs	否

IAM 特征	准备情况检查支持
ABAC（策略中的标签）	是
临时凭证	是
主体权限	是
服务角色	否
服务相关角色	是

要全面了解 AWS 服务如何与大多数 IAM 功能配合使用，请参阅 IAM 用户指南中的与 IAM 配合使用的 AWS 服务。

基于身份的准备情况检查策略

支持基于身份的策略：是

基于身份的策略是可附加到身份（如 IAM 用户、用户组或角色）的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的使用客户管理型策略定义自定义 IAM 权限。

通过使用 IAM 基于身份的策略，您可以指定允许或拒绝的操作和资源以及允许或拒绝操作的条件。您无法在基于身份的策略中指定主体，因为它适用于其附加的用户或角色。要了解可在 JSON 策略中使用的所有元素，请参阅《IAM 用户指南》中的 IAM JSON 策略元素引用。

要查看 ARC 基于身份的策略的示例，请参阅。Amazon 应用程序恢复控制器 (ARC) 中基于身份的策略示例

准备情况检查中基于资源的策略

支持基于资源的策略：否

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资源的访问。

准备情况检查的政策行动

支持策略操作：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

JSON 策略的 Action 元素描述可用于在策略中允许或拒绝访问的操作。策略操作通常与关联的 AWS API 操作同名。有一些例外情况，例如没有匹配 API 操作的仅限权限 操作。还有一些操作需要在策略中执行多个操作。这些附加操作称为相关操作。

在策略中包含操作以授予执行关联操作的权限。

要查看准备情况检查的 ARC 操作列表，请参阅《服务授权参考》中的 Amazon Route 53 恢复准备情况定义的操作。

ARC 中的准备情况检查策略操作在操作前使用以下前缀：

```
route53-recovery-readiness
```

要在单个语句中指定多项操作，请使用逗号将它们隔开。例如，以下内容：

```
"Action": [
      "route53-recovery-readiness:action1",
      "route53-recovery-readiness:action2"
         ]
```

您也可以使用通配符（*）指定多个操作。例如，要指定以单词 Describe 开头的所有操作，包括以下操作：

```
"Action": "route53-recovery-readiness:Describe*"
```

要查看 ARC 基于身份的准备情况检查策略示例，请参阅。在 Amazon 应用程序恢复控制器 (ARC) 中进行准备情况检查的基于身份的策略示例

准备情况检查的政策资源

支持策略资源：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

Resource JSON 策略元素指定要向其应用操作的一个或多个对象。语句必须包含 Resource 或 NotResource 元素。作为最佳实践，请使用其 Amazon 资源名称（ARN）指定资源。对于支持特定资源类型（称为资源级权限）的操作，您可以执行此操作。

对于不支持资源级权限的操作（如列出操作），请使用通配符（*）指示语句应用于所有资源。

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

要查看用于区域转移的 ARC 操作列表，请参阅 Amazon Route 53 恢复准备状态定义的操作。

要查看 ARC 基于身份的准备情况检查策略示例，请参阅。在 Amazon 应用程序恢复控制器 (ARC) 中进行准备情况检查的基于身份的策略示例

准备情况检查的策略条件密钥

支持特定于服务的策略条件键：是

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

在 Condition 元素（或 Condition 块）中，可以指定语句生效的条件。Condition 元素是可选的。您可以创建使用条件运算符（例如，等于或小于）的条件表达式，以使策略中的条件与请求中的值相匹配。

如果您在一个语句中指定多个 Condition 元素，或在单个 Condition 元素中指定多个键，则 AWS 使用逻辑 AND 运算评估它们。如果您为单个条件键指定多个值，则使用逻辑OR运算来 AWS 评估条件。在授予语句的权限之前必须满足所有的条件。

在指定条件时，您也可以使用占位符变量。例如，只有在使用 IAM 用户名标记 IAM 用户时，您才能为其授予访问资源的权限。有关更多信息，请参阅《IAM 用户指南》中的 IAM 策略元素：变量和标签。

AWS 支持全局条件密钥和特定于服务的条件键。要查看所有 AWS 全局条件键，请参阅 IAM 用户指南中的AWS 全局条件上下文密钥。

要查看准备情况检查的 ARC 操作列表，请参阅 Amazon Route 53 恢复准备情况的条件密钥

要查看可用于带就绪检查的条件密钥的操作和资源，请参阅 Amazon Route 53 恢复准备情况定义的操作

要查看 ARC 基于身份的准备情况检查策略示例，请参阅。在 Amazon 应用程序恢复控制器 (ARC) 中进行准备情况检查的基于身份的策略示例

准备情况检查中的访问控制列表 (ACLs)

支持 ACLs：否

访问控制列表 (ACLs) 控制哪些委托人（账户成员、用户或角色）有权访问资源。 ACLs 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

带就绪性检查的基于属性的访问控制 (ABAC)

支持 ABAC（策略中的标签）：部分支持

基于属性的访问控制（ABAC）是一种授权策略，该策略基于属性来定义权限。在中 AWS，这些属性称为标签。您可以向 IAM 实体（用户或角色）和许多 AWS 资源附加标签。标记实体和资源是 ABAC 的第一步。然后设计 ABAC 策略，以在主体的标签与他们尝试访问的资源标签匹配时允许操作。

ABAC 在快速增长的环境中非常有用，并在策略管理变得繁琐的情况下可以提供帮助。

要基于标签控制访问，您需要使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 条件键在策略的条件元素中提供标签信息。

如果某个服务对于每种资源类型都支持所有这三个条件键，则对于该服务，该值为是。如果某个服务仅对于部分资源类型支持所有这三个条件键，则该值为部分。

有关 ABAC 的更多信息，请参阅《IAM 用户指南》中的使用 ABAC 授权定义权限。要查看设置 ABAC 步骤的教程，请参阅《IAM 用户指南》中的使用基于属性的访问权限控制（ABAC）。

恢复就绪（就绪检查）支持 ABAC。

使用带有准备情况检查的临时证书

支持临时凭证：是

当你使用临时证书登录时，有些 AWS 服务 不起作用。有关更多信息，包括哪些 AWS 服务 适用于临时证书，请参阅 IAM 用户指南中的AWS 服务 与 IAM 配合使用的信息。

如果您使用除用户名和密码之外的任何方法登录，则 AWS Management Console 使用的是临时证书。例如，当您 AWS 使用公司的单点登录 (SSO) 链接进行访问时，该过程会自动创建临时证书。当您以用户身份登录控制台，然后切换角色时，您还会自动创建临时凭证。有关切换角色的更多信息，请参阅《IAM 用户指南》中的从用户切换到 IAM 角色（控制台）。

您可以使用 AWS CLI 或 AWS API 手动创建临时证书。然后，您可以使用这些临时证书进行访问 AWS。 AWS 建议您动态生成临时证书，而不是使用长期访问密钥。有关更多信息，请参阅 IAM 中的临时安全凭证。

准备情况检查的跨服务主体权限

支持转发访问会话（FAS）：是

当您使用 IAM 实体（用户或角色）在中执行操作时 AWS，您被视为委托人。策略向主体授予权限。使用某些服务时，您可能会执行一个操作，此操作然后在不同服务中触发另一个操作。在这种情况下，您必须具有执行这两个操作的权限。

要查看准备情况检查中的操作是否需要策略中的其他相关操作，请参阅 Amazon Route 53 恢复准备情况

准备情况检查的服务角色

支持服务角色：否

服务角色是由一项服务担任、代表您执行操作的 IAM 角色。IAM 管理员可以在 IAM 中创建、修改和删除服务角色。有关更多信息，请参阅《IAM 用户指南》中的创建向 AWS 服务委派权限的角色。

用于就绪性检查的服务关联角色

支持服务相关角色：是

服务相关角色是一种链接到的服务角色。 AWS 服务服务可以代入代表您执行操作的角色。服务相关角色出现在您的中 AWS 账户 ，并且归服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。

有关创建或管理 ARC 服务相关角色的详细信息，请参阅在 ARC 中使用服务关联角色进行准备情况检查。

有关创建或管理服务相关角色的详细信息，请参阅能够与 IAM 搭配使用的AWS 服务。在表中查找服务相关角色列中包含 Yes 的表。选择是链接以查看该服务的服务相关角色文档。

在 Amazon 应用程序恢复控制器 (ARC) 中进行准备情况检查的基于身份的策略示例

默认情况下，用户和角色无权创建或修改 ARC 资源。他们也无法使用 AWS Management Console、AWS Command Line Interface (AWS CLI) 或 AWS API 执行任务。要授予用户对所需资源执行操作的权限，IAM 管理员可以创建 IAM 策略。管理员随后可以向角色添加 IAM 策略，用户可以代入角色。

要了解如何使用这些示例 JSON 策略文档创建基于 IAM 身份的策略，请参阅《IAM 用户指南》中的创建 IAM 策略（控制台）。

有关 ARC 定义的操作和资源类型（包括每种资源类型的格式）的详细信息，请参阅《 ARNs 服务授权参考》中的 Amazon Application Recovery Controller (ARC) 的操作、资源和条件密钥。

主题

- 策略最佳实践

- 示例：准备情况检查控制台访问权限
- 示例：就绪性检查 API 操作以进行就绪性检查

策略最佳实践

基于身份的策略决定了某人是否可以在您的账户中创建、访问或删除 ARC 资源。这些操作可能会使 AWS 账户产生成本。创建或编辑基于身份的策略时，请遵循以下指南和建议：

- 开始使用 AWS 托管策略并转向最低权限权限 — 要开始向用户和工作负载授予权限，请使用为许多常见用例授予权限的AWS 托管策略。它们在你的版本中可用 AWS 账户。我们建议您通过定义针对您的用例的 AWS 客户托管策略来进一步减少权限。有关更多信息，请参阅《IAM 用户指南》中的AWS 托管式策略或工作职能的AWS 托管式策略。
- 应用最低权限：在使用 IAM 策略设置权限时，请仅授予执行任务所需的权限。为此，您可以定义在特定条件下可以对特定资源执行的操作，也称为最低权限许可。有关使用 IAM 应用权限的更多信息，请参阅《IAM 用户指南》中的 IAM 中的策略和权限。
- 使用 IAM 策略中的条件进一步限制访问权限：您可以向策略添加条件来限制对操作和资源的访问。例如，您可以编写策略条件来指定必须使用 SSL 发送所有请求。如果服务操作是通过特定 AWS 服务的（例如）使用的，则也可以使用条件来授予对服务操作的访问权限 AWS CloudFormation。有关更多信息，请参阅《IAM 用户指南》中的 IAM JSON 策略元素：条件。
- 使用 IAM Access Analyzer 验证您的 IAM 策略，以确保权限的安全性和功能性 – IAM Access Analyzer 会验证新策略和现有策略，以确保策略符合 IAM 策略语言（JSON）和 IAM 最佳实践。IAM Access Analyzer 提供 100 多项策略检查和可操作的建议，以帮助您制定安全且功能性强的策略。有关更多信息，请参阅《IAM 用户指南》中的使用 IAM Access Analyzer 验证策略。
- 需要多重身份验证 (MFA)-如果 AWS 账户您的场景需要 IAM 用户或根用户，请启用 MFA 以提高安全性。若要在调用 API 操作时需要 MFA，请将 MFA 条件添加到您的策略中。有关更多信息，请参阅《IAM 用户指南》中的使用 MFA 保护 API 访问。

有关 IAM 中的最佳实操的更多信息，请参阅《IAM 用户指南》中的 IAM 中的安全最佳实践。

示例：准备情况检查控制台访问权限

要访问 Amazon 应用程序恢复控制器 (ARC) 控制台，您必须拥有一组最低权限。这些权限必须允许您列出和查看有关您的 ARC 资源的详细信息 AWS 账户。如果创建比必需的最低权限更为严格的基于身份的策略，对于附加了该策略的实体（用户或角色），控制台将无法按预期正常运行。

对于仅调用 AWS CLI 或 AWS API 的用户，您无需为其设置最低控制台权限。相反，只允许访问与其尝试执行的 API 操作相匹配的操作。

为确保在您仅允许访问特定 API 操作时用户和角色仍然可以使用就绪检查控制台，还要向实体附加准备情况检查的ReadOnly AWS 托管策略。有关更多信息，请参阅 IAM 用户指南中的准备情况检查准备情况检查托管策略页面或向用户添加权限。

要执行某些任务，用户必须有权在 ARC 中创建与 “就绪检查” 关联的服务相关角色。要了解更多信息，请参阅 在 ARC 中使用服务关联角色进行准备情况检查。

要向用户提供通过控制台使用准备情况检查功能的完全访问权限，请向用户附加类似以下的策略：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                   "route53-recovery-readiness:CreateCell",
                   "route53-recovery-readiness:CreateCrossAccountAuthorization",
                   "route53-recovery-readiness:CreateReadinessCheck",
                   "route53-recovery-readiness:CreateRecoveryGroup",
                   "route53-recovery-readiness:CreateResourceSet",
                   "route53-recovery-readiness:DeleteCell",
                   "route53-recovery-readiness:DeleteCrossAccountAuthorization",
                   "route53-recovery-readiness:DeleteReadinessCheck",
                   "route53-recovery-readiness:DeleteRecoveryGroup",
                   "route53-recovery-readiness:DeleteResourceSet",
                   "route53-recovery-readiness:GetArchitectureRecommendations",
                   "route53-recovery-readiness:GetCell",
                   "route53-recovery-readiness:GetCellReadinessSummary",
                   "route53-recovery-readiness:GetReadinessCheck",
                   "route53-recovery-readiness:GetReadinessCheckResourceStatus",
                   "route53-recovery-readiness:GetReadinessCheckStatus",
                   "route53-recovery-readiness:GetRecoveryGroup",
                   "route53-recovery-readiness:GetRecoveryGroupReadinessSummary",
                   "route53-recovery-readiness:GetResourceSet",
                   "route53-recovery-readiness:ListCells",
                   "route53-recovery-readiness:ListCrossAccountAuthorizations",
                   "route53-recovery-readiness:ListReadinessChecks",
                   "route53-recovery-readiness:ListRecoveryGroups",
                   "route53-recovery-readiness:ListResourceSets",
                   "route53-recovery-readiness:ListRules",
                   "route53-recovery-readiness:UpdateCell",
                   "route53-recovery-readiness:UpdateReadinessCheck",
                   "route53-recovery-readiness:UpdateRecoveryGroup",
```

```
                    "route53-recovery-readiness:UpdateResourceSet"
                ],
                "Resource": "*"
            }
        ]
}
```

示例：就绪性检查 API 操作以进行就绪性检查

为确保用户可以使用 ARC API 操作来处理 ARC 就绪检查控制平面（例如，创建恢复组、资源集和就绪检查），请附上与用户需要使用的 API 操作相对应的策略，如下所述。

要执行某些任务，用户必须有权在 ARC 中创建与 “就绪检查” 关联的服务相关角色。要了解更多信息，请参阅 在 ARC 中使用服务关联角色进行准备情况检查。

要使用 API 操作进行准备情况检查，请向用户附加如下策略：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                    "route53-recovery-readiness:CreateCell",
                    "route53-recovery-readiness:CreateCrossAccountAuthorization",
                    "route53-recovery-readiness:CreateReadinessCheck",
                    "route53-recovery-readiness:CreateRecoveryGroup",
                    "route53-recovery-readiness:CreateResourceSet",
                    "route53-recovery-readiness:DeleteCell",
                    "route53-recovery-readiness:DeleteCrossAccountAuthorization",
                    "route53-recovery-readiness:DeleteReadinessCheck",
                    "route53-recovery-readiness:DeleteRecoveryGroup",
                    "route53-recovery-readiness:DeleteResourceSet",
                    "route53-recovery-readiness:GetArchitectureRecommendations",
                    "route53-recovery-readiness:GetCell",
                    "route53-recovery-readiness:GetCellReadinessSummary",
                    "route53-recovery-readiness:GetReadinessCheck",
                    "route53-recovery-readiness:GetReadinessCheckResourceStatus",
                    "route53-recovery-readiness:GetReadinessCheckStatus",
                    "route53-recovery-readiness:GetRecoveryGroup",
                    "route53-recovery-readiness:GetRecoveryGroupReadinessSummary",
                    "route53-recovery-readiness:GetResourceSet",
                    "route53-recovery-readiness:ListCells",
```

```
                "route53-recovery-readiness:ListCrossAccountAuthorizations",
                "route53-recovery-readiness:ListReadinessChecks",
                "route53-recovery-readiness:ListRecoveryGroups",
                "route53-recovery-readiness:ListResourceSets",
                "route53-recovery-readiness:ListRules",
                "route53-recovery-readiness:ListTagsForResources",
                "route53-recovery-readiness:UpdateCell",
                "route53-recovery-readiness:UpdateReadinessCheck",
                "route53-recovery-readiness:UpdateRecoveryGroup",
                "route53-recovery-readiness:UpdateResourceSet",
                "route53-recovery-readiness:TagResource",
                "route53-recovery-readiness:UntagResource"
            ],
            "Resource": "*"
        }
    ]
}
```

在 ARC 中使用服务关联角色进行准备情况检查

Amazon 应用程序恢复控制器使用 AWS Identity and Access Management (IAM) 服务相关角色。服务相关角色是一种独特的 IAM 角色，直接链接到服务（在本例中为 ARC）。服务相关角色由 ARC 预定义，包括该服务出于特定目的代表您调用其他 AWS 服务所需的所有权限。

服务相关角色使设置 ARC 变得更加容易，因为您不必手动添加必要的权限。ARC 定义其服务相关角色的权限，除非另有定义，否则只有 ARC 可以担任其角色。定义的权限包括信任策略和权限策略，而且权限策略不能附加到任何其它 IAM 实体。

只有在首先删除服务相关角色的相关资源后，才能删除该角色。这可以保护您的 ARC 资源，因为您不能无意中移除访问这些资源的权限。

有关支持服务相关角色的其他服务的信息，请参阅与 IAM 配合使用的AWS 服务，并在服务相关角色列中查找标有 “是” 的服务。选择是和链接，查看该服务的服务相关角色文档。

ARC 具有以下服务相关角色，本章将对此进行介绍：

- ARC 使用名为 Route53 的服务相关角色访问资源和配置RecoveryReadinessServiceRolePolicy以检查准备情况。
- ARC 使用为自动换档练习命名的服务相关角色来监控客户提供的 A CloudWatch mazon 警报和 AWS Health Dashboard 客户事件，并开始练习。

Route53 的服务相关角色权限 RecoveryReadinessServiceRolePolicy

ARC 使用名为 Route53 的服务相关角色访问资源和配置RecoveryReadinessServiceRolePolicy以检查准备情况。本节介绍适用于该服务相关角色的权限，以及有关创建、编辑和删除该角色的信息。

Route53 的服务相关角色权限 RecoveryReadinessServiceRolePolicy

此服务相关角色使用托管策略 `Route53RecoveryReadinessServiceRolePolicy`。

Route53 RecoveryReadinessServiceRolePolicy 服务相关角色信任以下服务来代入该角色：

- `route53-recovery-readiness.amazonaws.com`

要查看此策略的权限，请参阅《AWS 托管策略参考》RecoveryReadinessServiceRolePolicy中的Route53。

您必须配置权限，允许 IAM 实体（如用户、组或角色）创建、编辑或删除服务相关角色。有关更多信息，请参阅《IAM 用户指南》中的服务相关角色权限。

为 ARC 创建 Route53 RecoveryReadinessServiceRolePolicy 服务相关角色

您无需手动创建 Route53 RecoveryReadinessServiceRolePolicy 服务相关角色。当您在 AWS Management Console、或 AWS API 中创建首次准备情况检查或跨账户授权时，ARC 会为您创建服务相关角色。 AWS CLI

如果您删除该服务相关角色，然后需要再次创建，您可以使用相同流程在账户中重新创建此角色。当您创建第一次准备情况检查或跨账户授权时，ARC 会再次为您创建服务相关角色。

编辑 ARC 的 Route53 RecoveryReadinessServiceRolePolicy 服务相关角色

ARC 不允许您编辑 Route53 RecoveryReadinessServiceRolePolicy 服务相关角色。创建该服务相关角色后，将无法更改角色名称，因为可能有其它实体引用该角色。不过，您可以使用 IAM 编辑角色的说明。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 ARC 的 Route53 RecoveryReadinessServiceRolePolicy 服务相关角色

如果不再需要使用某个需要服务相关角色的功能或服务，我们建议您删除该角色。这样就没有未被主动监控或维护的未使用实体。但是，必须先清除服务相关角色的资源，然后才能手动删除它。

删除准备情况检查和跨账户授权后，您可以删除 Route RecoveryReadinessServiceRolePolicy 53 服务相关角色。有关就绪检查的更多信息，请参阅ARC 中的准备情况检查。有关跨账户授权的更多信息，请参阅在 ARC 中创建跨账户授权。

> **Note**
>
> 如果您尝试删除资源时 ARC 服务正在使用该角色，则删除服务角色可能会失败。如果发生这种情况，请等待几分钟，然后重新尝试删除该角色。

使用 IAM 手动删除服务相关角色

使用 IAM 控制台 AWS CLI、或 AWS API 删除 Route53 RecoveryReadinessServiceRolePolicy 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。

更新了 ARC 服务相关角色以进行准备情况检查

有关 ARC 服务相关角色 AWS 托管策略的更新，请参阅 ARC 的AWS 托管策略更新表。您也可以在 ARC 文档历史记录页面上订阅自动 RSS 提醒。

AWS 在 Amazon 应用程序恢复控制器 (ARC) 中进行准备情况检查的托管策略

AWS 托管策略是由创建和管理的独立策略 AWS。 AWS 托管策略旨在为许多常见用例提供权限，以便您可以开始为用户、组和角色分配权限。

请记住， AWS 托管策略可能不会为您的特定用例授予最低权限权限，因为它们可供所有 AWS 客户使用。我们建议通过定义特定于您的使用场景的客户管理型策略来进一步减少权限。

您无法更改 AWS 托管策略中定义的权限。如果 AWS 更新 AWS 托管策略中定义的权限，则更新会影响该策略所关联的所有委托人身份（用户、组和角色）。 AWS 最有可能在启动新的 API 或现有服务可以使用新 AWS 服务 的 API 操作时更新 AWS 托管策略。

有关更多信息，请参阅《IAM 用户指南》中的 AWS 托管策略。

AWS 托管策略：Route53 RecoveryReadinessServiceRolePolicy

您不能将 `Route53RecoveryReadinessServiceRolePolicy` 附加到自己的 IAM 实体。此策略附加到服务相关角色，该角色允许 Amazon 应用程序恢复控制器 (ARC) 访问由 ARC 使用或管理的 AWS 服务和资源。有关更多信息，请参阅 在 ARC 中使用服务关联角色进行准备情况检查。

AWS 托管策略： AmazonRoute53 RecoveryReadinessFullAccess

您可以将 `AmazonRoute53RecoveryReadinessFullAccess` 附加到 IAM 实体。此策略允许用户完全访问在 ARC 中处理恢复准备情况（准备情况检查）的操作。将此策略附加到需要恢复就绪操作的完全访问权限的 IAM 用户和其他主体。

要查看此策略的权限，请参阅《AWS 托管策略参考》RecoveryReadinessFullAccess中的AmazonRoute53。

AWS 托管策略： AmazonRoute53 RecoveryReadinessReadOnlyAccess

您可以将 AmazonRoute53RecoveryReadinessReadOnlyAccess 附加到 IAM 实体。此策略授予对 ARC 中恢复准备状态的操作的只读访问权限。这种权限适用于需要查看就绪状态和恢复组配置的用户。这些用户无法创建、更新或删除恢复就绪资源。

要查看此策略的权限，请参阅《AWS 托管策略参考》RecoveryReadinessReadOnlyAccess中的AmazonRoute53。

更新 AWS 托管策略以备不时之需

有关自该服务开始跟踪这些更改以来在 ARC 中进行就绪检查的 AWS 托管策略更新的详细信息，请参阅Amazon 应用程序恢复控制器 (ARC) AWS 托管策略的更新。要获得有关此页面变更的自动提醒，请订阅 ARC 文档历史记录页面上的 RSS 提要。

准备情况检查配额

Amazon 应用程序恢复控制器 (ARC) 中的准备情况检查受以下配额（以前称为限制）的约束。

实体	限额
每个账户的恢复组数	5
每个账户的规则数	15
每个单元格的嵌套单元格数	3
每个恢复组的单元格数	3
每个单元格的资源数	10
每个恢复组的资源数	10
每个资源集的资源数	6
每个账户的资源集数	200
每个账户的就绪检查数	200

实体	限额
跨账户授权数	100

应用程序恢复控制器的代码示例 AWS SDKs

以下代码示例展示了如何将应用程序恢复控制器与 AWS 软件开发套件 (SDK) 配合使用。

操作是大型程序的代码摘录，必须在上下文中运行。您可以通过操作了解如何调用单个服务函数，还可以通过函数相关场景的上下文查看操作。

有关 S AWS DK 开发者指南和代码示例的完整列表，请参阅将此服务与 AWS SDK 配合使用。本主题还包括有关入门的信息以及有关先前的 SDK 版本的详细信息。

代码示例

- 应用程序恢复控制器的基本示例 AWS SDKs
  - 应用程序恢复控制器使用的操作 AWS SDKs
    - 与 AWS SDK GetRoutingControlState 配合使用
    - 与 AWS SDK UpdateRoutingControlState 配合使用

应用程序恢复控制器的基本示例 AWS SDKs

以下代码示例展示了如何将 Amazon Route 53 应用程序恢复控制器的基础知识与配合使用 AWS SDKs。

示例

- 应用程序恢复控制器使用的操作 AWS SDKs
  - 与 AWS SDK GetRoutingControlState 配合使用
  - 与 AWS SDK UpdateRoutingControlState 配合使用

应用程序恢复控制器使用的操作 AWS SDKs

以下代码示例演示了如何使用执行单个应用程序恢复控制器操作 AWS SDKs。每个示例都包含一个指向的链接 GitHub，您可以在其中找到有关设置和运行代码的说明。

以下示例仅包括最常用的操作。有关完整列表，请参阅 Amazon Route 53 应用程序恢复控制器 API 参考。

示例

- 与 AWS SDK GetRoutingControlState 配合使用

- 与 AWS SDK UpdateRoutingControlState 配合使用

与 AWS SDK **GetRoutingControlState** 配合使用

以下代码示例演示如何使用 GetRoutingControlState。

Java

适用于 Java 的 SDK 2.x

> **Note**
>
> 还有更多相关信息 GitHub。查找完整示例，学习如何在 AWS 代码示例存储库中进行设置和运行。

```
    public static GetRoutingControlStateResponse
 getRoutingControlState(List<ClusterEndpoint> clusterEndpoints,
            String routingControlArn) {
        // As a best practice, we recommend choosing a random cluster endpoint to
 get or
        // set routing control states.
        // For more information, see
        // https://docs.aws.amazon.com/r53recovery/latest/dg/route53-arc-best-
practices.html#route53-arc-best-practices.regional
        Collections.shuffle(clusterEndpoints);
        for (ClusterEndpoint clusterEndpoint : clusterEndpoints) {
            try {
                System.out.println(clusterEndpoint);
                Route53RecoveryClusterClient client =
 Route53RecoveryClusterClient.builder()
                        .endpointOverride(URI.create(clusterEndpoint.endpoint()))
                        .region(Region.of(clusterEndpoint.region())).build();
                return client.getRoutingControlState(
                        GetRoutingControlStateRequest.builder()
                                .routingControlArn(routingControlArn).build());
            } catch (Exception exception) {
                System.out.println(exception);
            }
        }
        return null;
    }
```

- 有关 API 的详细信息，请参阅 AWS SDK for Java 2.x API 参考GetRoutingControlState中的。

Python

适用于 Python 的 SDK（Boto3）

> **Note**
>
> 还有更多相关信息 GitHub。查找完整示例，学习如何在 AWS 代码示例存储库中进行设置和运行。

```
import boto3


def create_recovery_client(cluster_endpoint):
    """
    Creates a Boto3 Route 53 Application Recovery Controller client for the
 specified
    cluster endpoint URL and AWS Region.

    :param cluster_endpoint: The cluster endpoint URL and Region.
    :return: The Boto3 client.
    """
    return boto3.client(
        "route53-recovery-cluster",
        endpoint_url=cluster_endpoint["Endpoint"],
        region_name=cluster_endpoint["Region"],
    )



def get_routing_control_state(routing_control_arn, cluster_endpoints):
    """
    Gets the state of a routing control. Cluster endpoints are tried in
    sequence until the first successful response is received.

    :param routing_control_arn: The ARN of the routing control to look up.
```

```
    :param cluster_endpoints: The list of cluster endpoints to query.
    :return: The routing control state response.
    """

    # As a best practice, we recommend choosing a random cluster endpoint to get
 or set routing control states.
    # For more information, see https://docs.aws.amazon.com/r53recovery/latest/
dg/route53-arc-best-practices.html#route53-arc-best-practices.regional
    random.shuffle(cluster_endpoints)
    for cluster_endpoint in cluster_endpoints:
        try:
            recovery_client = create_recovery_client(cluster_endpoint)
            response = recovery_client.get_routing_control_state(
                RoutingControlArn=routing_control_arn
            )
            return response
        except Exception as error:
            print(error)
            raise error
```

- 有关 API 的详细信息，请参阅适用GetRoutingControlState于 Python 的AWS SDK (Boto3) API 参考。

有关 S AWS DK 开发者指南和代码示例的完整列表，请参阅将此服务与 AWS SDK 配合使用。本主题还包括有关入门的信息以及有关先前的 SDK 版本的详细信息。

与 AWS SDK **UpdateRoutingControlState** 配合使用

以下代码示例演示如何使用 UpdateRoutingControlState。

Java

适用于 Java 的 SDK 2.x

> **Note**
>
> 还有更多相关信息 GitHub。查找完整示例，学习如何在 AWS 代码示例存储库中进行设置和运行。

```
    public static UpdateRoutingControlStateResponse
 updateRoutingControlState(List<ClusterEndpoint> clusterEndpoints,
            String routingControlArn,
            String routingControlState) {
        // As a best practice, we recommend choosing a random cluster endpoint to
 get or
        // set routing control states.
        // For more information, see
        // https://docs.aws.amazon.com/r53recovery/latest/dg/route53-arc-best-
practices.html#route53-arc-best-practices.regional
        Collections.shuffle(clusterEndpoints);
        for (ClusterEndpoint clusterEndpoint : clusterEndpoints) {
            try {
                System.out.println(clusterEndpoint);
                Route53RecoveryClusterClient client =
 Route53RecoveryClusterClient.builder()
                        .endpointOverride(URI.create(clusterEndpoint.endpoint()))
                        .region(Region.of(clusterEndpoint.region()))
                        .build();
                return client.updateRoutingControlState(
                        UpdateRoutingControlStateRequest.builder()

 .routingControlArn(routingControlArn).routingControlState(routingControlState).build());
            } catch (Exception exception) {
                System.out.println(exception);
            }
        }
        return null;
    }
```

- 有关 API 的详细信息，请参阅 AWS SDK for Java 2.x API 参考[UpdateRoutingControlState](#)中的。

Python

适用于 Python 的 SDK（Boto3）

> **Note**
>
> 还有更多相关信息 GitHub。查找完整示例，学习如何在 AWS 代码示例存储库中进行设置和运行。

```
import boto3


def create_recovery_client(cluster_endpoint):
    """
    Creates a Boto3 Route 53 Application Recovery Controller client for the
 specified
    cluster endpoint URL and AWS Region.

    :param cluster_endpoint: The cluster endpoint URL and Region.
    :return: The Boto3 client.
    """
    return boto3.client(
        "route53-recovery-cluster",
        endpoint_url=cluster_endpoint["Endpoint"],
        region_name=cluster_endpoint["Region"],
    )



def update_routing_control_state(
    routing_control_arn, cluster_endpoints, routing_control_state
):
    """
    Updates the state of a routing control. Cluster endpoints are tried in
    sequence until the first successful response is received.

    :param routing_control_arn: The ARN of the routing control to update the
 state for.
    :param cluster_endpoints: The list of cluster endpoints to try.
    :param routing_control_state: The new routing control state.
    :return: The routing control update response.
```

```
    """

    # As a best practice, we recommend choosing a random cluster endpoint to get
 or set routing control states.
    # For more information, see https://docs.aws.amazon.com/r53recovery/latest/
dg/route53-arc-best-practices.html#route53-arc-best-practices.regional
    random.shuffle(cluster_endpoints)
    for cluster_endpoint in cluster_endpoints:
        try:
            recovery_client = create_recovery_client(cluster_endpoint)
            response = recovery_client.update_routing_control_state(
                RoutingControlArn=routing_control_arn,
                RoutingControlState=routing_control_state,
            )
            return response
        except Exception as error:
            print(error)
```

- 有关 API 的详细信息，请参阅适用UpdateRoutingControlState于 Python 的AWS SDK (Boto3) API 参考。

有关 S AWS DK 开发者指南和代码示例的完整列表，请参阅将此服务与 AWS SDK 配合使用。本主题还包括有关入门的信息以及有关先前的 SDK 版本的详细信息。

Amazon 应用程序恢复控制器中的安全性

云安全 AWS 是重中之重。作为 AWS 客户，您可以受益于专为满足大多数安全敏感型组织的要求而构建的数据中心和网络架构。

安全是双方共同承担 AWS 的责任。责任共担模式将其描述为云的安全性和云中的安全性：

- 云安全 — AWS 负责保护在云中运行 AWS 服务的基础架构 AWS Cloud。 AWS 还为您提供可以安全使用的服务。作为AWS 合规计划合规计划合规计划合的一部分，第三方审计师定期测试和验证我们安全的有效性。要了解适用于 Amazon Application Recovery Controller 的合规计划，请参阅合规计划范围内的AWSAWS 服务服务。
- 云端安全-您的责任由您使用的 AWS 服务决定。您还需要对其他因素负责，包括您的数据的敏感性、您公司的要求以及适用的法律法规。

本文档可帮助您了解在使用 ARC 时如何应用分担责任模型。以下主题向您展示如何配置 ARC 以满足您的安全和合规性目标。您还将学习如何使用其他 AWS 服务来帮助您监控和保护您的 ARC 资源。

主题

- Amazon 应用程序恢复控制器中的数据保护
- 适用于亚马逊应用程序恢复控制器 (ARC) 的 Identity and Access Management
- 在 ARC 中进行日志记录和监控
- Amazon 应用程序恢复控制器的合规性验证
- Amazon 应用程序恢复控制器中的弹性
- Amazon 应用程序恢复控制器中的基础设施安全

Amazon 应用程序恢复控制器中的数据保护

AWS 分担责任模式适用于 Amazon 应用程序恢复控制器中的数据保护。如本模型所述 AWS ，负责保护运行所有内容的全球基础架构 AWS Cloud。您负责维护对托管在此基础结构上的内容的控制。您还负责您所使用的 AWS 服务 的安全配置和管理任务。有关数据隐私的更多信息，请参阅数据隐私常见问题。有关欧洲数据保护的信息，请参阅 AWS Security Blog 上的 AWS Shared Responsibility Model and GDPR 博客文章。

出于数据保护目的，我们建议您保护 AWS 账户 凭证并使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 设置个人用户。这样，每个用户只获得履行其工作职责所需的权限。还建议您通过以下方式保护数据：

- 对每个账户使用多重身份验证（MFA）。
- 用于 SSL/TLS 与 AWS 资源通信。我们要求使用 TLS 1.2，建议使用 TLS 1.3。
- 使用设置 API 和用户活动日志 AWS CloudTrail。有关使用 CloudTrail 跟踪捕获 AWS 活动的信息，请参阅《AWS CloudTrail 用户指南》中的使用跟 CloudTrail 踪。
- 使用 AWS 加密解决方案以及其中的所有默认安全控件 AWS 服务。
- 使用高级托管安全服务（例如 Amazon Macie），它有助于发现和保护存储在 Amazon S3 中的敏感数据。
- 如果您在 AWS 通过命令行界面或 API 进行访问时需要经过 FIPS 140-3 验证的加密模块，请使用 FIPS 端点。有关可用的 FIPS 端点的更多信息，请参阅《美国联邦信息处理标准（FIPS）第 140-3 版》。

强烈建议您切勿将机密信息或敏感信息（如您客户的电子邮件地址）放入标签或自由格式文本字段（如名称字段）。这包括您 AWS 服务 使用控制台、API 或与 ARC 或其他人合作时 AWS SDKs。AWS CLI在用于名称的标签或自由格式文本字段中输入的任何数据都可能会用于计费或诊断日志。如果您向外部服务器提供网址，强烈建议您不要在网址中包含凭证信息来验证对该服务器的请求。

静态加密

客户配置信息存储在服务拥有的 Amazon DynamoDB 全局表中，并进行静态加密。

包含 ARC 集群中单元格状态的数据集将写入 Amazon EBS 卷进行备份。ARC 在数据处于静态状态时使用默认的 Amazon EBS 加密。

传输中加密

在整个服务传输过程中，使用 TLS 对客户请求和响应（ARC 配置、就绪状态查询、蜂窝状态更新等）进行加密。

适用于亚马逊应用程序恢复控制器 (ARC) 的 Identity and Access Management

AWS Identity and Access Management (IAM) AWS 服务 可帮助管理员安全地控制对 AWS 资源的访问权限。IAM 管理员控制谁可以进行身份验证（登录）和授权（有权限）使用 ARC 资源。您可以使用 IAM AWS 服务 ，无需支付额外费用。

受众

您的使用方式 AWS Identity and Access Management (IAM) 会有所不同，具体取决于您在 ARC 中所做的工作。

服务用户-如果您使用 ARC 服务完成工作，则您的管理员会为您提供所需的凭证和权限。当你使用更多 ARC 功能来完成工作时，你可能需要额外的权限。了解如何管理访问权限有助于您向管理员请求适合的权限。如果您无法访问 ARC 中的功能，请参阅排查 身份和访问问题。

服务管理员 — 如果您负责公司的 ARC 资源，则可能拥有对 ARC 的完全访问权限。您的工作是确定您的服务用户应访问哪些 ARC 功能和资源。然后，您必须向 IAM 管理员提交请求以更改服务用户的权限。请查看该页面上的信息以了解 IAM 的基本概念。要详细了解贵公司如何在 ARC 中使用 IAM，请参阅Amazon 应用程序恢复控制器 (ARC) 功能如何与 IAM 配合使用。

IAM 管理员 — 如果您是 IAM 管理员，则可能需要详细了解如何编写策略来管理 ARC 的访问权限。要查看您可以在 IAM 中使用的基于身份的 ARC 策略示例，请参阅。Amazon 应用程序恢复控制器 (ARC) 中基于身份的策略示例

使用身份进行身份验证

身份验证是您 AWS 使用身份凭证登录的方式。您必须以 IAM 用户身份或通过担 AWS 账户根用户任 IAM 角色进行身份验证（登录 AWS）。

您可以使用通过身份源提供的凭据以 AWS 联合身份登录。 AWS IAM Identity Center （IAM Identity Center）用户、贵公司的单点登录身份验证以及您的 Google 或 Facebook 凭据就是联合身份的示例。当您以联合身份登录时，您的管理员以前使用 IAM 角色设置了身份联合验证。当你使用联合访问 AWS 时，你就是在间接扮演一个角色。

根据您的用户类型，您可以登录 AWS Management Console 或 AWS 访问门户。有关登录的更多信息 AWS，请参阅《AWS 登录 用户指南》中的如何登录到您 AWS 账户的。

如果您 AWS 以编程方式访问，则会 AWS 提供软件开发套件 (SDK) 和命令行接口 (CLI)，以便使用您的凭据对请求进行加密签名。如果您不使用 AWS 工具，则必须自己签署请求。有关使用推荐的方法自行签署请求的更多信息，请参阅《IAM 用户指南》中的用于签署 API 请求的AWS 签名版本 4。

无论使用何种身份验证方法，您都可能需要提供其他安全信息。例如， AWS 建议您使用多重身份验证 (MFA) 来提高账户的安全性。要了解更多信息，请参阅《AWS IAM Identity Center 用户指南》中的多重身份验证和《IAM 用户指南》中的 IAM 中的AWS 多重身份验证。

AWS 账户 root 用户

创建时 AWS 账户，首先要有一个登录身份，该身份可以完全访问账户中的所有资源 AWS 服务 和资源。此身份被称为 AWS 账户 root 用户，使用您创建账户时使用的电子邮件地址和密码登录即可访问该身份。强烈建议您不要使用根用户执行日常任务。保护好根用户凭证，并使用这些凭证来执行仅根用户可以执行的任务。有关要求您以根用户身份登录的任务的完整列表，请参阅 IAM 用户指南中的需要根用户凭证的任务。

联合身份

作为最佳实践，要求人类用户（包括需要管理员访问权限的用户）使用与身份提供商的联合身份验证 AWS 服务 通过临时证书进行访问。

联合身份是指您的企业用户目录、Web 身份提供商、Identity C enter 目录中的用户，或者任何使用 AWS 服务 通过身份源提供的凭据进行访问的用户。 AWS Directory Service当联合身份访问时 AWS 账户，他们将扮演角色，角色提供临时证书。

要集中管理访问权限，建议您使用 AWS IAM Identity Center。您可以在 IAM Identity Center 中创建用户和群组，也可以连接并同步到您自己的身份源中的一组用户和群组，以便在您的所有 AWS 账户 和应用程序中使用。有关 IAM Identity Center 的信息，请参阅 AWS IAM Identity Center 用户指南中的什么是 IAM Identity Center？。

IAM 用户和群组

I AM 用户是您 AWS 账户 内部对个人或应用程序具有特定权限的身份。在可能的情况下，我们建议使用临时凭证，而不是创建具有长期凭证（如密码和访问密钥）的 IAM 用户。但是，如果您有一些特定的使用场景需要长期凭证以及 IAM 用户，建议您轮换访问密钥。有关更多信息，请参阅《IAM 用户指南》中的对于需要长期凭证的用例，应在需要时更新访问密钥。

IAM 组是一个指定一组 IAM 用户的身份。您不能使用组的身份登录。您可以使用组来一次性为多个用户指定权限。如果有大量用户，使用组可以更轻松地管理用户权限。例如，您可以拥有一个名为的群组，IAMAdmins并向该群组授予管理 IAM 资源的权限。

用户与角色不同。用户唯一地与某个人员或应用程序关联，而角色旨在让需要它的任何人代入。用户具有永久的长期凭证，而角色提供临时凭证。要了解更多信息，请参阅《IAM 用户指南》中的 IAM 用户的使用案例。

IAM 角色

I AM 角色是您内部具有特定权限 AWS 账户 的身份。它类似于 IAM 用户，但与特定人员不关联。要在中临时担任 IAM 角色 AWS Management Console，您可以从用户切换到 IAM 角色（控制台）。您可以通过调用 AWS CLI 或 AWS API 操作或使用自定义 URL 来代入角色。有关使用角色的方法的更多信息，请参阅《IAM 用户指南》中的代入角色的方法。

具有临时凭证的 IAM 角色在以下情况下很有用：

- 联合用户访问：要向联合身份分配权限，请创建角色并为角色定义权限。当联合身份进行身份验证时，该身份将与角色相关联并被授予由此角色定义的权限。有关用于联合身份验证的角色的信息，请参阅《IAM 用户指南》中的针对第三方身份提供商创建角色（联合身份验证）。如果您使用 IAM Identity Center，则需要配置权限集。为控制您的身份在进行身份验证后可以访问的内容，IAM Identity Center 将权限集与 IAM 中的角色相关联。有关权限集的信息，请参阅《AWS IAM Identity Center 用户指南》中的权限集。
- 临时 IAM 用户权限：IAM 用户可代入 IAM 用户或角色，以暂时获得针对特定任务的不同权限。
- 跨账户存取：您可以使用 IAM 角色以允许不同账户中的某个人（可信主体）访问您的账户中的资源。角色是授予跨账户访问权限的主要方式。但是，对于某些资源 AWS 服务，您可以将策略直接附加到资源（而不是使用角色作为代理）。要了解用于跨账户访问的角色和基于资源的策略之间的差别，请参阅 IAM 用户指南中的 IAM 中的跨账户资源访问。
- 跨服务访问 — 有些 AWS 服务 使用其他 AWS 服务服务中的功能。例如，当您在服务中拨打电话时，该服务通常会在 Amazon 中运行应用程序 EC2 或在 Amazon S3 中存储对象。服务可能会使用发出调用的主体的权限、使用服务角色或使用服务相关角色来执行此操作。
  - 转发访问会话 (FAS) — 当您使用 IAM 用户或角色在中执行操作时 AWS，您被视为委托人。使用某些服务时，您可能会执行一个操作，然后此操作在其他服务中启动另一个操作。FAS 使用调用委托人的权限以及 AWS 服务 向下游服务发出请求的请求。 AWS 服务只有当服务收到需要与其他 AWS 服务 或资源交互才能完成的请求时，才会发出 FAS 请求。在这种情况下，您必须具有执行这两项操作的权限。有关发出 FAS 请求时的策略详情，请参阅转发访问会话。
  - 服务角色 - 服务角色是服务代表您在您的账户中执行操作而分派的 IAM 角色。IAM 管理员可以在 IAM 中创建、修改和删除服务角色。有关更多信息，请参阅《IAM 用户指南》中的创建向 AWS 服务委派权限的角色。

- 服务相关角色-服务相关角色是一种链接到的服务角色。 AWS 服务服务可以代入代表您执行操作的角色。服务相关角色出现在您的中 AWS 账户 ，并且归服务所有。IAM 管理员可以查看但不能编辑服务相关角色的权限。

- 在 A@@ mazon 上运行的应用程序 EC2 — 您可以使用 IAM 角色管理在 EC2 实例上运行并发出 AWS CLI 或 AWS API 请求的应用程序的临时证书。这比在 EC2 实例中存储访问密钥更可取。要为 EC2 实例分配 AWS 角色并使其可供其所有应用程序使用，您需要创建一个附加到该实例的实例配置文件。实例配置文件包含角色并允许在 EC2 实例上运行的程序获得临时证书。有关更多信息，请参阅 IAM 用户指南中的使用 IAM 角色向在 A mazon EC2 实例上运行的应用程序授予权限。

使用策略管理访问

您可以 AWS 通过创建策略并将其附加到 AWS 身份或资源来控制中的访问权限。策略是其中的一个对象 AWS ，当与身份或资源关联时，它会定义其权限。 AWS 在委托人（用户、root 用户或角色会话）发出请求时评估这些策略。策略中的权限确定是允许还是拒绝请求。大多数策略都以 JSON 文档的 AWS 形式存储在中。有关 JSON 策略文档的结构和内容的更多信息，请参阅 IAM 用户指南中的 JSON 策略概览。

管理员可以使用 AWS JSON 策略来指定谁有权访问什么。也就是说，哪个主体可以对什么资源执行操作，以及在什么条件下执行。

默认情况下，用户和角色没有权限。要授予用户对所需资源执行操作的权限，IAM 管理员可以创建 IAM 策略。管理员随后可以向角色添加 IAM 策略，用户可以代入角色。

IAM 策略定义操作的权限，无关乎您使用哪种方法执行操作。例如，假设您有一个允许 `iam:GetRole` 操作的策略。拥有该策略的用户可以从 AWS Management Console AWS CLI、或 AWS API 获取角色信息。

基于身份的策略

基于身份的策略是可附加到身份（如 IAM 用户、用户组或角色）的 JSON 权限策略文档。这些策略控制用户和角色可在何种条件下对哪些资源执行哪些操作。要了解如何创建基于身份的策略，请参阅《IAM 用户指南》中的使用客户托管策略定义自定义 IAM 权限。

基于身份的策略可以进一步归类为内联策略或托管式策略。内联策略直接嵌入单个用户、组或角色中。托管策略是独立的策略，您可以将其附加到中的多个用户、群组和角色 AWS 账户。托管策略包括 AWS 托管策略和客户托管策略。要了解如何在托管策略和内联策略之间进行选择，请参阅《IAM 用户指南》中的在托管策略与内联策略之间进行选择。

基于资源的策略

基于资源的策略是附加到资源的 JSON 策略文档。基于资源的策略的示例包括 IAM 角色信任策略和 Amazon S3 存储桶策略。在支持基于资源的策略的服务中，服务管理员可以使用它们来控制对特定资源的访问。对于在其中附加策略的资源，策略定义指定主体可以对该资源执行哪些操作以及在什么条件下执行。您必须在基于资源的策略中指定主体。委托人可以包括账户、用户、角色、联合用户或 AWS 服务。

基于资源的策略是位于该服务中的内联策略。您不能在基于资源的策略中使用 IAM 中的 AWS 托管策略。

访问控制列表 (ACLs)

访问控制列表 (ACLs) 控制哪些委托人（账户成员、用户或角色）有权访问资源。 ACLs 与基于资源的策略类似，尽管它们不使用 JSON 策略文档格式。

Amazon S3 和 Amazon VPC 就是支持的服务示例 ACLs。 AWS WAF要了解更多信息 ACLs，请参阅《亚马逊简单存储服务开发者指南》中的访问控制列表 (ACL) 概述。

其他策略类型

AWS 支持其他不太常见的策略类型。这些策略类型可以设置更常用的策略类型向您授予的最大权限。

- 权限边界：权限边界是一个高级特征，用于设置基于身份的策略可以为 IAM 实体（IAM 用户或角色）授予的最大权限。您可为实体设置权限边界。这些结果权限是实体基于身份的策略及其权限边界的交集。在 `Principal` 中指定用户或角色的基于资源的策略不受权限边界限制。任一项策略中的显式拒绝将覆盖允许。有关权限边界的更多信息，请参阅IAM 用户指南中的 IAM 实体的权限边界。
- 服务控制策略 (SCPs)- SCPs 是指定组织或组织单位 (OU) 的最大权限的 JSON 策略 AWS Organizations。 AWS Organizations 是一项用于对您的企业拥有的多 AWS 账户 项进行分组和集中管理的服务。如果您启用组织中的所有功能，则可以将服务控制策略 (SCPs) 应用于您的任何或所有帐户。SCP 限制成员账户中的实体（包括每个 AWS 账户根用户实体）的权限。有关 Organization SCPs s 和的更多信息，请参阅《AWS Organizations 用户指南》中的服务控制策略。
- 资源控制策略 (RCPs) — RCPs 是 JSON 策略，您可以使用它来设置账户中资源的最大可用权限，而无需更新附加到您拥有的每个资源的 IAM 策略。RCP 限制成员账户中资源的权限，并可能影响身份（包括身份）的有效权限 AWS 账户根用户，无论这些身份是否属于您的组织。有关 Organizations 的更多信息 RCPs，包括 AWS 服务 该支持的列表 RCPs，请参阅《AWS Organizations 用户指南》中的资源控制策略 (RCPs)。

- 会话策略：会话策略是当您以编程方式为角色或联合用户创建临时会话时作为参数传递的高级策略。结果会话的权限是用户或角色的基于身份的策略和会话策略的交集。权限也可以来自基于资源的策略。任一项策略中的显式拒绝将覆盖允许。有关更多信息，请参阅 IAM 用户指南中的会话策略。

多个策略类型

当多个类型的策略应用于一个请求时，生成的权限更加复杂和难以理解。要了解在涉及多种策略类型时如何 AWS 确定是否允许请求，请参阅 IAM 用户指南中的策略评估逻辑。

Amazon 应用程序恢复控制器 (ARC) 功能如何与 IAM 配合使用

有关每个 Amazon 应用程序恢复控制器 (ARC) 功能如何与 IAM 配合使用的信息，请参阅以下主题：

- 用于区域转移的 IAM
- 区域自动换档的 IAM
- 用于路由控制的 IAM
- IAM 用于准备情况检查

Amazon 应用程序恢复控制器 (ARC) 中基于身份的策略示例

要查看 Amazon 应用程序恢复控制器 (ARC) 中每项功能的基于身份的策略示例，请参阅每种功能 AWS Identity and Access Management 章节中的以下主题：

- 基于身份的区域自动换档策略示例
- ARC 中基于身份的区域转移策略示例
- Amazon 应用程序恢复控制器 (ARC) 中基于身份的路由控制策略示例
- 在 Amazon 应用程序恢复控制器 (ARC) 中进行准备情况检查的基于身份的策略示例

AWS Amazon 应用程序恢复控制器 (ARC) 的托管策略

有关带有 AWS 托管策略的 ARC 功能的托管策略（包括服务相关角色的托管策略）的信息，请参阅以下主题：

- 区域自动换档的托管策略
- 用于路由控制的托管策略
- 用于准备情况检查的托管策略

Amazon 应用程序恢复控制器 (ARC) AWS 托管策略的更新

查看有关自该服务开始跟踪这些更改以来对 ARC 中功能的 AWS 托管策略更新的详细信息。要获得有关此页面变更的自动提醒，请订阅 ARC 文档历史记录页面上的 RSS 提要。

更改	描述	日期
AWSZonalAutoshiftPracticeRunSLRPolicy 托管策略-更新的策略	添加包含权限`autoscaling:DescribeAutoScalingGroups` 、、和`AutoshiftPracticeCheckPermissions` 的策略声明 `ec2:DescribeInstances` `elasticloadbalancing:DescribeTargetHealth` ，`elasticloadbalancing:DescribeTargetHealth` 以支持容量平衡检查。 要了解更多信息，请参阅可用区自动转移和练习运行的工作原理。	2025 年 6 月 30 日
AWSServiceRoleForPercPracticePolicy— 新政策	ARC 为自动换档和练习跑新增了一个与服务相关的角色。 ARC 使用服务相关角色启用的权限来监控客户提供的 Amazon CloudWatch 警报和客户 AWS Health Dashboard 活动以进行练习，并开始练习。 要了解有关新服务相关角色的更多信息，请参阅 的服务相关角色权限 AWSService	2023 年 11 月 30 日

更改	描述	日期
	RoleForZonalAutoshiftPracticeRun。	
AmazonRoute53 RecoveryControlConfigReadOnlyAccess — 更新了政策	为添加权限`GetResourcePolicy` ，以支持返回有关共享 AWS Resource Access Manager 资源的资源策略的详细信息。	2023 年 10 月 18 日

53 号公路 RecoveryReadinessServiceRolePolicy — 更新了政策	ARC 增加了查询亚马逊 EC2 实例相关信息的新权限。
	
	ARC 使用以下权限来支持轮询 Amazon EC2 实例、运行就绪检查和确定实例的就绪状态。
	
	`ec2:DescribeVpnGateways`
	
	`ec2:DescribeCustomerGateways`
		2023 年 2 月 17 日

53 号公路 RecoveryReadinessServiceRolePolicy — 更新了政策	ARC 增加了查询有关 Lambda 函数信息的新权限。 ARC 使用以下权限查询有关 Lambda 函数的信息，以运行就绪检查并确定函数的就绪状态。 `lambda:ListProvisionedConcurrencyConfigs`	2022 年 8 月 31 日
AmazonRoute53 RecoveryControlConfigFullAccess — 更新了政策	从策略中删除了 Amazon Route 53 权限，并增加了列出可选权限的注释。	2022 年 5 月 26 日

更改	描述	日期
AmazonRoute53 RecoveryControlConfigFullAccess — 更新了政策	在策略中增加了缺少的 Amazon Route 53 必要权限。	2022 年 4 月 15 日
AmazonRoute53 RecoveryClusterReadOnlyAccess — 更新了政策	ARC 添加了一项新权限`route53-recovery-cluster:ListRoutingControls` ，以允许 ARNs 具有高可用性的列表路由控制。	2022 年 3 月 15 日
AmazonRoute53 RecoveryControlConfigReadOnlyAccess — 更新了政策	ARC 添加了一项新权限`route53-recovery-control-config:ListTagsForResources` ，允许列出资源的标签。	2021 年 12 月 20 日
53 号公路 RecoveryReadinessServiceRolePolicy — 更新了政策	ARC 增加了查询有关亚马逊 API Gateway 信息的新权限。 ARC 使用权限查询有关 API Gateway 的信息，以运行就绪检查并确定就绪状态。`apigateway:GET`	2021 年 10 月 28 日

更改	描述	日期
AmazonRoute53 RecoveryReadinessReadOnlyAccess-添加了新权限	ARC 在 AmazonRoute53 中添加了两个新权限RecoveryReadinessReadOnlyAccess： ARC 使用`route53-recovery-readiness:GetArchitectureRecommendations` 和`route53-recovery-readiness:GetCellReadinessSummary` 来允许对这些操作进行只读访问以准备恢复。	2021 年 10 月 15 日

更改	描述	日期

53 号公路 RecoveryReadinessServiceRolePolicy — 更新了政策	ARC 增加了查询有关 Lambda 函数信息的新权限。
	
	ARC 使用以下权限查询有关 Lambda 函数的信息，以运行就绪检查并确定这些函数的就绪状态。
	
	`lambda:GetFunctionConcurrency`
	
	`lambda:GetFunctionConfiguration`
	
	`lambda:GetProvisionedConcurrencyConfig`
	
	`lambda:ListAliases`
	
	`lambda:ListVersionsByFunction`
	
	`lambda:ListEventSourceMappings`
	
	`lambda:ListFunctions`
		2021 年 10 月 8 日

更改	描述	日期
Route53 RecoveryReadinessServiceRolePolicy-添加了新的托管策略	ARC 添加了以下新的托管策略： AmazonRoute53 RecoveryReadinessFullAccess AmazonRoute53 RecoveryReadinessReadOnlyAccess AmazonRoute53 RecoveryClusterFullAccess AmazonRoute53 RecoveryClusterReadOnlyAccess AmazonRoute53 RecoveryControlConfigFullAccess AmazonRoute53 RecoveryControlConfigReadOnlyAccess	2021 年 8 月 18 日
ARC 开始追踪变更	ARC 开始跟踪其 AWS 托管策略的更改。	2021 年 7 月 27 日

排查 身份和访问问题

使用以下信息来帮助您诊断和修复在使用 Amazon 应用程序恢复控制器 (ARC) 和 IAM 时可能遇到的常见问题。

主题

- 我无权在 ARC 中执行操作
- 我无权执行 iam：PassRole
- 我想允许我以外的人 AWS 账户 访问我的 ARC 资源

我无权在 ARC 中执行操作

如果 AWS Management Console 告诉您您无权执行某项操作，则必须联系管理员寻求帮助。管理员是向您提供凭证的人员。

当 mateojackson IAM 用户尝试使用控制台查看有关虚构 *my-example-widget* 资源的详细信息，但不拥有虚构 route53-recovery-readiness:*GetWidget* 权限时，会发生以下示例错误。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 route53-recovery-readiness:GetWidget on resource: my-example-widget
```

在这种情况下，Mateo 请求他的管理员更新其策略，以允许他使用 route53-recovery-readiness:*GetWidget* 操作访问 *my-example-widget* 资源。

我无权执行 iam：PassRole

如果您收到错误消息，说您无权执行iam:PassRole操作，则必须更新您的策略以允许您将角色传递给 ARC。

有些 AWS 服务 允许您将现有角色传递给该服务，而不是创建新的服务角色或服务相关角色。为此，您必须具有将角色传递到服务的权限。

当名为的 IAM 用户marymajor尝试使用控制台在 ARC 中执行操作时，会出现以下示例错误。但是，服务必须具有服务角色所授予的权限才可执行此操作。Mary 不具有将角色传递到服务的权限。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

在这种情况下，必须更新 Mary 的策略以允许她执行 iam:PassRole 操作。

如果您需要帮助，请联系您的 AWS 管理员。您的管理员是提供登录凭证的人。

我想允许我以外的人 AWS 账户 访问我的 ARC 资源

您可以创建一个角色，以便其他账户中的用户或您组织外的人员可以使用该角色来访问您的资源。您可以指定谁值得信赖，可以代入角色。对于支持基于资源的策略或访问控制列表 (ACLs) 的服务，您可以使用这些策略向人们授予访问您的资源的权限。

要了解更多信息，请参阅以下内容：

- 要了解 ARC 是否支持这些功能，请参阅Amazon 应用程序恢复控制器 (ARC) 功能如何与 IAM 配合使用。
- 要了解如何提供对您拥有的资源的访问权限 AWS 账户 ，请参阅 IAM 用户指南中的向您拥有 AWS 账户 的另一个 IAM 用户提供访问权限。
- 要了解如何向第三方提供对您的资源的访问权限 AWS 账户，请参阅 IAM 用户指南中的向第三方提供访问权限。 AWS 账户
- 要了解如何通过身份联合验证提供访问权限，请参阅《IAM 用户指南》中的为经过外部身份验证的用户（身份联合验证）提供访问权限。
- 要了解使用角色和基于资源的策略进行跨账户访问之间的差别，请参阅《IAM 用户指南》中的 IAM 中的跨账户资源访问。

在 ARC 中进行日志记录和监控

监控是维护 ARC 和您的 AWS 解决方案可用性和性能的重要组成部分。您应该从 AWS 解决方案的所有部分收集监控数据，以便在出现多点故障时可以更轻松地进行调试。 AWS 提供了多种工具，用于监控您的 ARC 资源和活动，以及响应潜在事件，例如 AWS CloudTrail 和 Amazon CloudWatch。

有关在 ARC 中监控每项功能的信息，请参阅以下主题：

- 记录和监控区域偏移
- 记录和监控区域自动换档
- 路由控制的日志和监控
- 记录和监控准备情况检查

Amazon 应用程序恢复控制器的合规性验证

作为多个合规计划的一部分，第三方审计师会评估 Amazon 应用程序恢复控制器的安全 AWS 性和合规性。其中包括 SOC、PCI、HIPAA 等。

要了解是否属于特定合规计划的范围，请参阅AWS 服务 “按合规计划划分的范围” ”，然后选择您感兴趣的合规计划。 AWS 服务 有关一般信息，请参阅AWS 合规计划AWS。

您可以使用下载第三方审计报告 AWS Artifact。有关更多信息，请参阅中的 “下载报告” 中的 “ AWS Artifact。

您在使用 AWS 服务 时的合规责任取决于您的数据的敏感性、贵公司的合规目标以及适用的法律和法规。 AWS 提供了以下资源来帮助实现合规性：

- Security Compliance & Governance：这些解决方案实施指南讨论了架构考虑因素，并提供了部署安全性和合规性功能的步骤。
- 符合 HIPAA 要求的服务参考：列出符合 HIPAA 要求的服务。并非所有 AWS 服务 人都符合 HIPAA 资格。
- AWS 合AWS 规资源 — 此工作簿和指南集合可能适用于您的行业和所在地区。
- AWS 客户合规指南 — 从合规角度了解责任共担模式。这些指南总结了保护的最佳实践， AWS 服务并将指南映射到跨多个框架（包括美国国家标准与技术研究院 (NIST)、支付卡行业安全标准委员会 (PCI) 和国际标准化组织 (ISO)）的安全控制。
- 使用AWS Config 开发人员指南中的规则评估资源 — 该 AWS Config 服务评估您的资源配置在多大程度上符合内部实践、行业准则和法规。
- AWS Security Hub— 这 AWS 服务 可以全面了解您的安全状态 AWS。Security Hub 通过安全控制措施评估您的 AWS 资源并检查其是否符合安全行业标准和最佳实践。有关受支持服务及控制措施的列表，请参阅 Security Hub 控制措施参考。
- Amazon GuardDuty — 它通过监控您的 AWS 账户环境中是否存在可疑和恶意活动，来 AWS 服务检测您的工作负载、容器和数据面临的潜在威胁。 GuardDuty 通过满足某些合规性框架规定的入侵检测要求，可以帮助您满足各种合规性要求，例如 PCI DSS。
- AWS Audit Manager— 这 AWS 服务 可以帮助您持续审计 AWS 使用情况，从而简化风险管理以及对法规和行业标准的合规性。

Amazon 应用程序恢复控制器中的弹性

AWS 全球基础设施是围绕 AWS 区域 可用区构建的。 AWS 区域 提供多个物理隔离和隔离的可用区，这些可用区通过低延迟、高吞吐量和高度冗余的网络连接。利用可用区，您可以设计和操作在可用区之间无中断地自动实现失效转移的应用程序和数据库。与传统的单个或多个数据中心基础结构相比，可用区具有更高的可用性、容错性和可扩展性。

有关 AWS 区域 和可用区的更多信息，请参阅AWS 全球基础设施。

除了 AWS 全球基础设施外，ARC 还提供多项功能来帮助支持您的数据弹性和备份需求。

Amazon 应用程序恢复控制器中的基础设施安全

作为一项托管服务，受 AWS 全球网络安全的保护。有关 AWS 安全服务以及如何 AWS 保护基础设施的信息，请参阅AWS 云安全。要使用基础设施安全的最佳实践来设计您的 AWS 环境，请参阅 S AWS ecurity Pillar Well-Architected Fram ework 中的基础设施保护。

您可以使用 AWS 已发布的 API 调用通过网络访问 ARC。客户端必须支持以下内容：

- 传输层安全性协议（TLS）。我们要求使用 TLS 1.2，建议使用 TLS 1.3。
- 具有完全向前保密（PFS）的密码套件，例如 DHE（临时 Diffie-Hellman）或 ECDHE（临时椭圆曲线 Diffie-Hellman）。大多数现代系统（如 Java 7 及更高版本）都支持这些模式。

此外，必须使用访问密钥 ID 和与 IAM 主体关联的秘密访问密钥来对请求进行签名。或者，您可以使用 AWS Security Token Service（AWS STS）生成临时安全凭证来对请求进行签名。

Amazon 应用程序恢复控制器 (ARC) 开发者指南的文档历史记录

以下条目描述了对 Amazon 应用程序恢复控制器 (ARC) 文档所做的重要更改。

- 版本：最新
- 最新文档更新：2025 年 6 月 30 日

更改	描述	日期
增强了练习跑的功能	现在，您可以在 ARC 中开始按需练习。此外，现在的练习包括检查该地区其他 AZs 地区是否有足够的容量。 有关更多信息，请参阅其工作原理。	2025 年 6 月 30 日
更新了托管策略	通过添加包含权限`autoscaling:DescribeAutoScalingGroups` 、和的策略声明`Autoshift PracticeCheckPermissions` 来更新`AWSZonalAutoshiftPracticeRunSLRPolicy` 托管策略`ec2:DescribeInstances elasticloadbalancing:DescribeTargetHealth` ，`elasticloadbalancing:DescribeTargetHealth` 以支持容量平衡检查。	2025 年 6 月 30 日

更改	描述	日期
	有关更多信息，请参阅AWSZonalAutoshiftPracticeRunSLRPolicy 托管策略。	
更新了区域自动移位的异常类型	现在，您可以根据每个资源与区域自动移位进行交互。 有关更多信息，请参阅其工作原理。	2025年4月21日
使用以下方法测试 ARC 区域自动换档 AWS FIS	你可以 AWS FIS 用来测试 ARC 区域自动换档在 AZ 电源中断期间如何自动恢复应用程序 有关更多信息，请参阅使用测试区域自动移位。 AWS FIS	2025 年 3 月 26 日
ARC 现在支持路由控制和区域偏移的 IPv6 端点。	ARC 现在支持路由控制和区域偏移的 IPv6 端点。 有关更多信息，请参阅设置路由控制组件。	2024 年 11 月 21 日
Amazon A EC2 uto Scaling 群组的区域偏移功能	ARC 现在支持 Amazon Aut EC2 o Scaling 群组的区域移动。 有关更多信息，请参阅对 Amazon A EC2 uto Scaling 群组的支持。	2024 年 11 月 18 日

更改	描述	日期
亚马逊 EKS 的区域偏移功能	您可以为 Amazon EKS 集群启动区域切换，也可以通过启用区域自动切换 AWS 来允许您进行区域切换。这种转变会更新集群中的 east-to-west网络流量，只考虑运行在工作节点上运行的 Pod 的网络终端节点 AZs。 有关更多信息，请参阅对亚马逊 Elastic Kubernetes 服务的支持 Amazon Kubernetes Servic e。	2024 年 10 月 22 日
网络负载均衡器的区域移位功能	ARC 现在支持启用跨区域或禁用跨区域配置的网络负载均衡器的区域移动。 有关更多信息，请参阅 Suppor t 对网络负载均衡器的支持。	2024 年 10 月 11 日
自动移位观察者通知	借助 autoshift 观察者通知，您可以配置区域自动切换，以便在 AWS 启动自动换档时通过 Amazon 通知您 EventBridge，将流量从可能受损的可用区转移出去。您无需使用区域自动移位配置任何特定资源即可启用这些单独的通知。 有关更多信息，请参阅在 A mazon 上使用区域自动切换。EventBridge	2024 年 7 月 12 日

更改	描述	日期
按每种能力对文档进行重组	重新组织开发者指南内容，将其分成子开发指南。也就是说，现在有单独的章节包含 ARC 中每项功能的全面信息：用于多可用区恢复的区域转移和区域自动移动，以及用于多区域恢复的路由控制和准备情况检查。 有关更多信息，请参阅什么是 Amazon 应用程序恢复控制器 (ARC)。	2024 年 4 月 30 日
增加了可用区自动转移功能	在 ARC 中添加了一项新功能，您可以代表您授权 AWS 将应用程序的资源流量从可用区转移出去，以帮助缩短事件期间的恢复时间。 有关更多信息，请参阅 Amazon 应用程序恢复控制器 (ARC) 中的区域自动切换。	2023 年 11 月 30 日
添加了新服务相关角色	为分区自动换档练习跑AWSServiceRoleForZonalAutoshiftPracticeRun添加新的服务相关角色。 有关更多信息，请参阅 AWSService 的服务相关角色权限RoleForZonalAutoshiftPracticeRun。	2023 年 11 月 30 日

更改	描述	日期
增加了对集群的跨账户支持	添加对 ARC 中集群的跨账户支持 AWS Resource Access Manager，这样您就可以轻松安全地使用一个集群来托管由多个不同 AWS 账户拥有的控制面板和路由控件。 有关更多信息，请参阅 Support 在 ARC 中支持跨账户集群。	2023 年 10 月 18 日
更新了托管策略	更新`AmazonRoute53RecoveryControlConfigReadOnly` 托管策略以添加权限`GetResourcePolicy` ，以支持返回有关共享 AWS Resource Access Manager 资源的资源策略的详细信息。 有关更多信息，请参阅 AWS 托管策略。	2023 年 9 月 19 日
更新了服务相关角色	在 ARC 的服务相关角色中添加了新的权限，以支持轮询 Amazon EC2 实例。`ec2:DescribeVpnGateways` `ec2:DescribeCustomerGateways` 有关更多信息，请参阅为 ARC 使用服务相关角色。	2023 年 2 月 17 日

更改	描述	日期
可用区转移 GA 版本	支持 GA 版本的 ARC 区域移动，其中包括针对在 ARC 中注册的用于区域转移的托管资源的基于属性的访问控制 (ABAC)。 有关更多信息，请参阅使用 ARC 实现基于属性的访问控制 (ABAC)。	2023 年 1 月 10 日
增加了新的多可用区的可用区转移	添加了描述 ARC 中针对多可用区应用程序的新服务（区域移动）的内容。您可以启动可用区转移，将负载均衡器资源的流量暂时从一个可用区移走。 有关更多信息，请参阅 ARC 中的区域偏移。	2022 年 11 月 28 日
更新了服务相关角色	为服务相关角色添加了新的权限`lambda:ListProvisionedConcurrencyConfigs` ，以便 ARC 查询有关 Lambda 函数的信息。 有关更多信息，请参阅为 ARC 使用服务相关角色。	2022 年 8 月 31 日

更改	描述	日期
更新了托管策略	更新了 AmazonRoute53RecoveryControlConfigFullAccess 托管策略，删除了 Amazon Route 53 权限并将其列为可选权限。 有关更多信息，请参阅 Amazon 应用程序恢复控制器 (ARC) 的AWS 托管策略。	2022 年 5 月 26 日
更新了托管策略	更新了 AmazonRoute53RecoveryControlConfigFullAccess 托管策略，使其包含所需的 Amazon Route 53 权限。 有关更多信息，请参阅 Amazon 应用程序恢复控制器 (ARC) 的AWS 托管策略。	2022 年 4 月 15 日
增加了新列出路由控制 API 的 CLI 示例	为极其可靠的 ARC 数据平面 API 中包含的新列表路由控制 API 操作添加了 CLI 命令示例和最佳实践建议。 有关更多信息，请参阅列出和更新路由控制和状态。	2022 年 3 月 31 日

更改	描述	日期
增加了对覆盖安全规则的支持	增加了对覆盖安全规则的支持，允许您绕过通过配置的安全规则强制执行的路由控制保护措施。例如，在“打碎玻璃”的情况下，为灾难恢复而进行失效转移期间可能需要覆盖安全规则。 有关更多信息，请参阅覆盖安全规则以重新路由流量。	2022 年 3 月 2 日
增加了额外的标记支持	增加了对在 ARC 中标记其他资源的支持，包括集群、控制面板、路由控制和安全规则。 有关更多信息，请参阅在 Amazon 应用程序恢复控制器 (ARC) 中添加标签。	2021 年 12 月 20 日
更新了托管策略	更新了 `AmazonRoute53RecoveryControlConfigReadOnly` 托管策略，增加了列出资源标签的权限。 有关更多信息，请参阅 Amazon 应用程序恢复控制器 (ARC) 的AWS 托管策略	2021 年 12 月 20 日

更改	描述	日期
增加了对实时警报的支持 EventBridge	增加了对的支持 EventBridge，这意味着现在您可以添加规则以获取警报并对 ARC 就绪检查状态的变化采取行动，例如，当状态从 “就绪” 变为 “未就绪” 时。 有关更多信息，请参阅在 Amazon 上使用 ARC EventBridge。	2021 年 12 月 20 日
增加了路由控制状态代码示例	增加了代码示例，说明在使用 API 操作获取或更新路由控制状态时按顺序尝试集群端点。 有关更多信息，请参阅 Amazon 应用程序恢复控制器 (ARC) 的 API 示例。	2021 年 11 月 16 日
在只读策略中增加了新权限。	在 `AmazonRoute53RecoveryReadinessReadOnlyAccess` 策略中增加了两个新权限：`route53-recovery-readiness:GetArchitectureRecommendations` 和 `route53-recovery-readiness:GetCellReadinessSummary` 。 有关更多信息，请参阅 Amazon 应用程序恢复控制器 (ARC) 的AWS 托管策略。	2021 年 11 月 9 日

更改	描述	日期
增加了对 Amazon API Gateway 资源类型的支持	添加了新的资源类型 Amazon API Gateway，并更新了 ARC 服务相关角色权限，这样 ARC 就可以通过准备情况检查来审计 API Gateway。 有关更多信息，请参阅就绪规则和支持的资源类型和为 ARC 使用服务相关角色。	2021 年 10 月 28 日
增加了对 Lambda 函数资源类型的支持	添加了新的资源类型 Lambda 函数，并更新了 ARC 服务相关角色权限，以便 ARC 可以通过就绪检查来审计 Lambda 函数。 有关更多信息，请参阅就绪规则和支持的资源类型和为 ARC 使用服务相关角色。	2021 年 10 月 8 日
添加了指向 CloudFormation 和 Terraform 模板的链接	添加了指向可下载模板 AWS CloudFormation 和 Hashicorp Terraform 模板的链接，以帮助您快速开始使用 Arc。有关更多信息，请参阅使用新应用程序做好恢复准备。	2021 年 9 月 13 日

更改	描述	日期
增加了新的托管策略	为 ARC 添加了以下 AWS 托管策略：AmazonRoute53RecoveryReadinessFullAccess AmazonRoute53RecoveryReadinessReadOnlyAccess 、AmazonRoute53RecoveryClusterFullAccess 、AmazonRoute53RecoveryClusterReadOnlyAccess 、AmazonRoute53RecoveryControlConfigFullAccess 、和AmazonRoute53RecoveryControlConfigReadOnlyAccess 。 有关更多信息，请参阅 Amazon 应用程序恢复控制器 (ARC) 的AWS 托管策略。	2021 年 8 月 18 日
已开始追踪 Amazon 应用程序恢复控制器 (ARC) 的 AWS 托管策略	自首次发布日期开始，跟踪托管策略的更新。 有关更多信息，请参阅 Amazon 应用程序恢复控制器 (ARC) 的AWS 托管策略。	2021 年 7 月 27 日

更改	描述	日期
Amazon 应用程序恢复控制器 (ARC) 的初始版本	ARC 通过集中协调一个 AWS 区域内或多个区域之间的故障转移来提高应用程序可用性。ARC 提供就绪检查，以确保您的应用程序经过扩展以处理故障转移流量，并配置为绕过故障。它还提供了极其可靠的路由控制，因此您可以通过重新路由流量（例如跨可用区或区域）来恢复应用程序。有关更多信息，请参阅什么是 ARC？。	2021 年 7 月 27 日

本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。