实施指南

虚拟等候室已开启 AWS

虚拟等候室已开启 AWS: 实施指南

Table of Contents

开启虚拟等候室后，为您的网站吸收大量流量 AWS

发布日期：二零二一年十一月（最后更新时间：二零二四年九月）

Virtual Waiting Room on AWS 解决方案可帮助控制在流量激增期间向您的网站传入的用户请求。它创建了一个云基础架构，旨在临时将传入流量转移到您的网站，并提供自定义和集成虚拟等候室的选项。该解决方案可以与新的或现有的网站集成，以实现无缝扩展，以应对突然激增的流量。

可能导致网站流量激增的大型活动示例包括：

- 开始销售音乐会或体育赛事门票
- 大甩卖或其他大型零售销售，例如黑色星期五
- 发布新产品并发布广泛的营销公告
- 在线考试和课程的考试准入和课堂出勤率
- 发放医疗预约时段
- 推出一项需要创建账户和付款的新 direct-to-customer 服务

该解决方案充当访问您网站的访客的屏蔽区，并在有足够的容量时允许流量通过。访问者使用的客户端软件可以配置为透明地允许流量通过等候室，直到网站达到最大容量；此时等候室会阻碍访问者。当您的网站能够容纳更多流量时，该解决方案会生成允许用户访问该网站的JSON网络令牌 (JWT)。例如，如果您的活动持续两个小时，并且您的网站每秒可以处理 50 个用户，但您预计流量为每秒 250 个，那么您可以使用此解决方案来调节流量，同时允许用户保持在队列中的位置。

该解决方案提供以下主要功能：

- 让用户有条理地排队进入你的网站
- 可扩展性，可控制大型事件的流量
- JSON生成网络令牌以允许进入目标站点
- 所有功能均通过以下方式控制 REST APIs
- 客户端解决方案的交钥匙API网关授权器
- 独立集成或与 OpenID 一起使用

本实施指南描述了在 Amazon Web Services (AWS) 云 AWS 中部署虚拟等候室的架构注意事项和配置步骤。它包括指向AWS CloudFormation模板的链接，这些模板使用安全性和可用性 AWS 最佳实践启动和配置部署此解决方案所需的 AWS 服务。

该指南适用于具有 AWS 云架构实践经验的 IT 架构师、开发 DevOps 人员、员工、数据分析师和营销技术专业人员。

费用

运行此解决方案时使用的 AWS 服务费用由您承担。从本次修订开始，在美国东部（弗吉尼亚北部）地区使用默认设置运行此解决方案的成本约为每个堆栈每天10.00美元，外加相对于事件规模的 API 请求和数据流量的费用。

在没有任何事件的情况下维护解决方案的每日成本

AWS service	请求/时间	成本 [美元]
Amazon API Gateway	0	0.00
Amazon CloudFront	0	0.00
Amazon CloudWatch	0	0.00
Amazon DynamoDB	0	0.00
Amazon ElastiCache	计算节点小时数 (Redis)	大约 6.00 美元
AWS Lambda	免费等级*	0.00
AWS Secrets Manager	免费等级*	0.00
Amazon Simple Storage Service (Amazon S3)	免费等级*	0.00
Amazon Virtual Private Cloud (Amazon VPC)	VPC 终端节点小时数 NAT 网关小时数	大约 5.00 美元
总计：		大约 11.00 美元

*费用估算基于清洁的环境。如果您在本解决方案之外使用此 AWS 服务，则可能会超过免费套餐配额。

下表显示了 50,000 个用户和 100,000 个用户的等候室的估计成本，活动持续时间为 2-4 小时，每秒传入 500 个用户，每分钟传出 1,000 个用户。价格可能会发生变化。有关完整详情，请参阅此解决方案中使用的每项 AWS 服务的定价网页。

在 2 小时活动期间，50,000 名等候室用户的估计费用

AWS service	尺寸	成本 [美元]
Amazon API Gateway	请求	2.00
CloudFront	请求、带宽	75.00 美元
CloudWatch	指标、警报、存储	1.00 美元
亚马逊 CloudWatch 活动	事件	1.00 美元
DynamoDB	读/写单元，存储	1.00 美元
ElastiCache	节点小时数	8.00 美元
Lambda	请求、计算时间	1.00 美元
AWS Secrets Manager	秘密，请求	1.00 美元
Amazon S3	请求、存储	1.00 美元
Amazon VPC	数据传输，端点时间	2.00
总计		94.00 美元

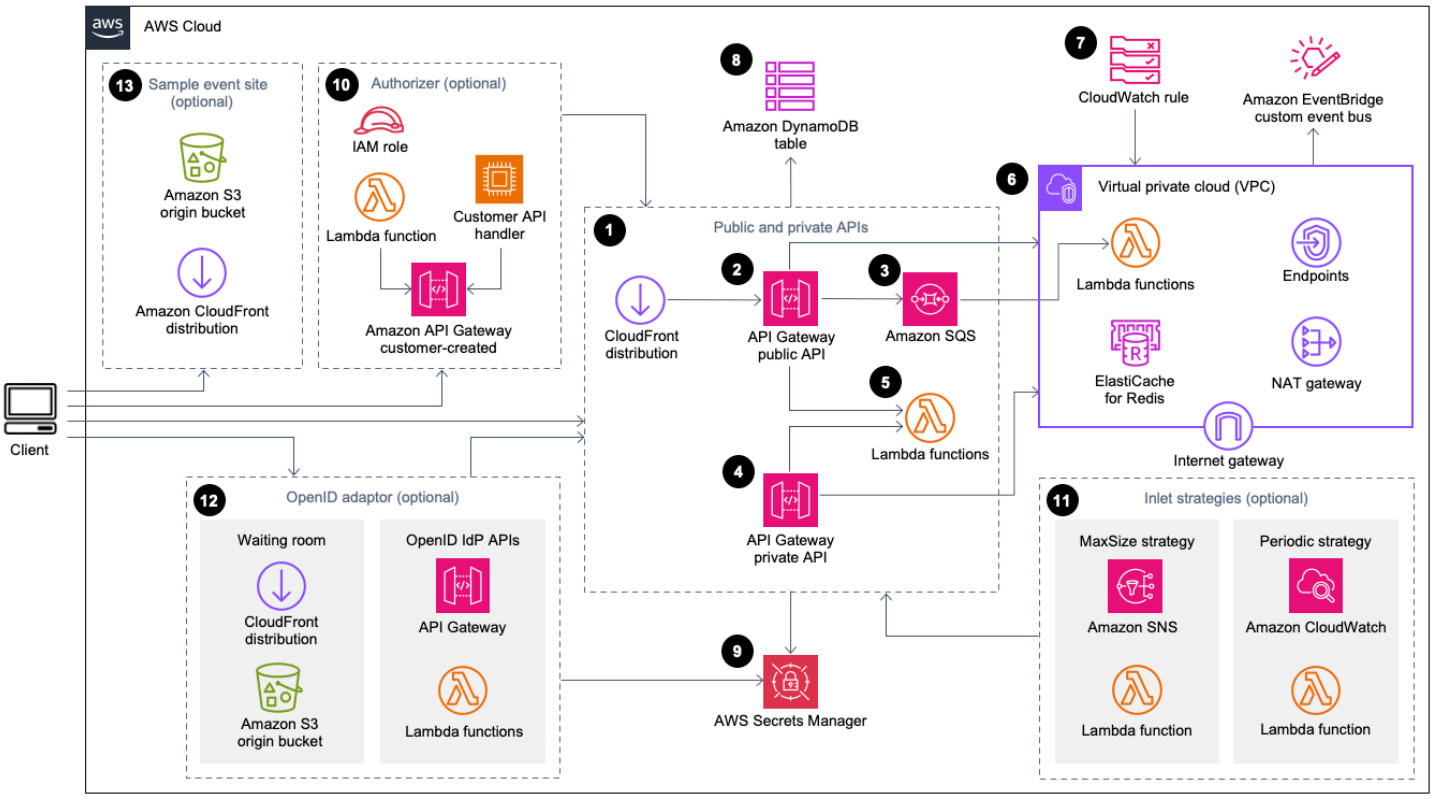
在 2 小时活动期间，100,000 名等候室用户的估计费用

AWS service	尺寸	成本 [美元]
Amazon API Gateway	请求	4.00 美元
CloudFront	请求、带宽	296.00 美元
CloudWatch	指标、警报、存储	1.00 美元
CloudWatch 大事记	事件	1.00 美元
DynamoDB	读/写单元，存储	4.00 美元

ElastiCache	节点小时数	32.00 美元
Lambda	请求、计算时间	1.00 美元
AWS Secrets Manager	秘密，请求	1.00 美元
Amazon Simple Queue Service(Amazon SQS)	请求	1.00 美元
Amazon S3	请求、存储	1.00 美元
Amazon VPC	数据传输，端点时间	6.00 美元
总计		348.00 美元

架构概述

使用必需模板和可选模板部署此解决方案，使用默认参数在 AWS 云中构建以下环境。



AWS 建筑虚拟等候室

这些 AWS CloudFormation 模板部署了以下基础架构：

1. A mazon CloudFront 发行版，用于为客户提供公开API呼叫。
2. Amazon API Gateway 公共API资源用于处理来自虚拟等候室的队列请求、跟踪队列位置以及支持验证允许访问目标网站的令牌。
3. A mazon 简单队列服务 (AmazonSQS) 队列，用于调节流向处理队列消息的AWS Lambda函数的流量。队列不是为每个请求调用 Lambda 函数，而是对传入SQS的突发请求进行批处理。
4. API网关私有API资源以支持管理职能。
5. Lambda 函数用于验证和处理公共和私有API请求，并返回相应的响应。
6. 亚马逊 Virtual Private Cloud (VPC) 用于托管直接与 Ela sticache (Redis) 集群交互的 Lambda 函数。OSSVPC终端节点允许中的 Lambda 函数VPC与解决方案中的服务进行通信。此外，NAT网关允许中的 Lambda 函数根据VPC需要连接 CloudFront 终端节点并使缓存失效。

7. 一项亚马逊 CloudWatch规则，用于调用 Lambda 函数，该函数与自定义 A ma zon EventBridge 总线配合使用，定期广播状态更新。
8. 用于存储令牌、队列位置和提供计数器数据的 Amazon DynamoDB 表。
9. AWS S@@ ecrets Manager 用于存储令牌操作的密钥和其他敏感数据。
10. （可选）授权方组件，由 AWS Identity and Access Management(IAM) 角色和用于网关的 Lambda 授权方函数组成。API
11. （可选）亚马逊简单通知服务 (AmazonSNS) 和 Lambda 函数支持两种入口策略。 CloudWatch
12. （可选）带有网关API和 Lambda 函数的 OpenID 适配器组件，允许 OpenID 提供商对访问您网站的用户进行身份验证。 CloudFront 使用亚马逊简单存储服务 (Amazon S3) 存储桶进行分发，用于此组件的等候室页面。
13. （可选）使用 Amazon S3 原始存储桶进行 CloudFront 分发，用于等候室 Web 应用程序示例。

解决方案的工作原理

本节简要介绍 AWS 虚拟等候室工作流程中的步骤。有关 GitHub为网站构建、自定义和集成等候室的详细信息，请参阅上的《开发者指南》。

等候室的公众API可以位于您的场地外围安全措施后面，也可以在未经任何授权的情况下进入等候室。根据你使用哪种方法将等候室与网站集成，用户可能需要先向网站进行身份验证，然后才能导航到等候室并在队列中获得一席之地。

客户端软件必须具有事件 ID 才能进入等候室并提出其他请求。事件 ID 是大多数针对公共和私人的请求所必需的唯一 ID APIs。事件 ID 是在安装核心API堆栈期间设置的。在操作过程中，事件 ID 可以通过等候室页面作为URL参数或 Cookie 提供；它可以作为身份验证令牌声明的一部分提供，也可以通过不同的数据路径分发给客户端。

在某些情况下，客户端需要事件 ID 和请求 ID 才能拨打某些API电话。请求编号是等候室签发的唯一 ID，代表排队的特定客户。

以下步骤描述了进入队列、等待队列进度以及使用网站访问令牌退出等候室的API请求流程。

用户进入等候室：

1. 用户将看到一个代表等候室入口点的屏幕或页面。他们选择进入队列，客户端软件（浏览器、移动设备、设备）打电话API给`assign_queue_num`公众请求队列位置。
2. Gate API way 会立即将API请求传送到 Amazon SQS 队列。
3. 当请求进入队列时，该`assign_queue_num`API呼叫将返回。客户端会收到一个唯一的请求 ID，以后可用于检索队列位置、请求时间和访问令牌。
4. `AssignQueueNumLambda` 函数从队列中接收多达 10 个请求的批次。SQSLambda 服务分散调用以处理多批请求。
5. `AssignQueueNumLambda` 函数会验证其批处理中的每条消息，在 Elasticache (Redis) 中增加队列计数器，并将每个请求及其关联的队列位置存储在 Elasticache (RedisOSS) 中。OSS
6. 每封邮件在成功处理后即被删除。错误条件中涉及的消息将在以后的批次中重新处理一次。第二次失败后，它们会被发送到 dead-letter-queue 已连接到CloudWatch警报的。
7. 客户端可以在收到来自呼叫的请求 ID `queue_num` API 后开始轮`assign_queue_num`询。客户端将事件 ID 和请求 ID 发送到，`queue_num`API并收到一个数字队列位置或表示请求尚未处理的响应。在大型活动期间，客户可能需要多次拨打此电话。`GetQueueNumLambda` 函数由 API Gateway 调用，并从 DynamoDB 返回客户端在队列中的数字位置。

用户在等候室等候：

8. 当客户端在队列中占据一席之地后，它就可以开始定期轮询了。serving_num API使用事件 ID 调serving_numAPI用，并返回队列的当前服务位置。的响应serving_numAPI告诉客户他们何时可以从等候室转移到可能发生最终交易的实际目标站点。L GetServingNum ambda 函数返回等候室的当前服务位置。
9. 当服务位置等于或大于客户端的队列（请求）位置时，客户端可以向公众请求 JSON Web Token (JWT) API。该代币可以与目标网站一起使用以完成交易。使用事件 ID 和请求 ID 调用。generate_token APIAPI网关使用参数调用 Lamb GenerateToken da 函数。
10. GenerateTokenLambda 函数验证请求并检查此令牌之前是否已生成。Lambda 函数在 DynamoDB 表中查询匹配的令牌。如果找到，则该令牌将返回给调用者，并且不会重新生成。此过程可防止使用单个请求 ID 生成具有新到期时间的多个不同令牌。
11. 如果在 DynamoDB 中找不到令牌，则 Lambda 函数会检索密钥以创建令牌，并将该令牌与事件 ID 和客户端的请求 ID 一起保存在 DynamoDB 中。Lambda 函数 EventBridge 向写入一个事件，表示已生成新令牌。Lambda 函数会增加一个 Elasticache (RedisOSS) 计数器，该计数器用于跟踪为该事件生成的代币数量。
12. 如果queue_pos_expiry已启用，则客户端可以通过调用 Lambda 函数GetQueuePositionExpiryTime来查询其到期前的剩余时间。queue_pos_expiry API

用户离开等候室：

13. 当客户端收到其令牌时，它会进入目标站点开始交易。根据您的基础设施支持与集成的方式JWT，客户端可能需要以请求标头、Cookie 或其他方式呈现令牌。APIGateway 的授权者可用于验证客户端请求中包含的令牌。任何用于验证和管理的商业或开源库都JWTs可以在 AWS 代币上与虚拟等候室一起使用。如果令牌有效，则允许客户继续交易。
14. 客户端完成交易后，会调用私API有账户来更新客户端令牌的状态，并在 DynamoDB 中完成。

队列位置到期：

15. 激活此功能后，与特定队列位置对应的请求 ID 只能在指定的时间间隔内生成令牌。

队列位置到期时增量发球计数器：

16. 激活此功能后，服务计数器将根据无法生成代币的过期队列位置自动递增。

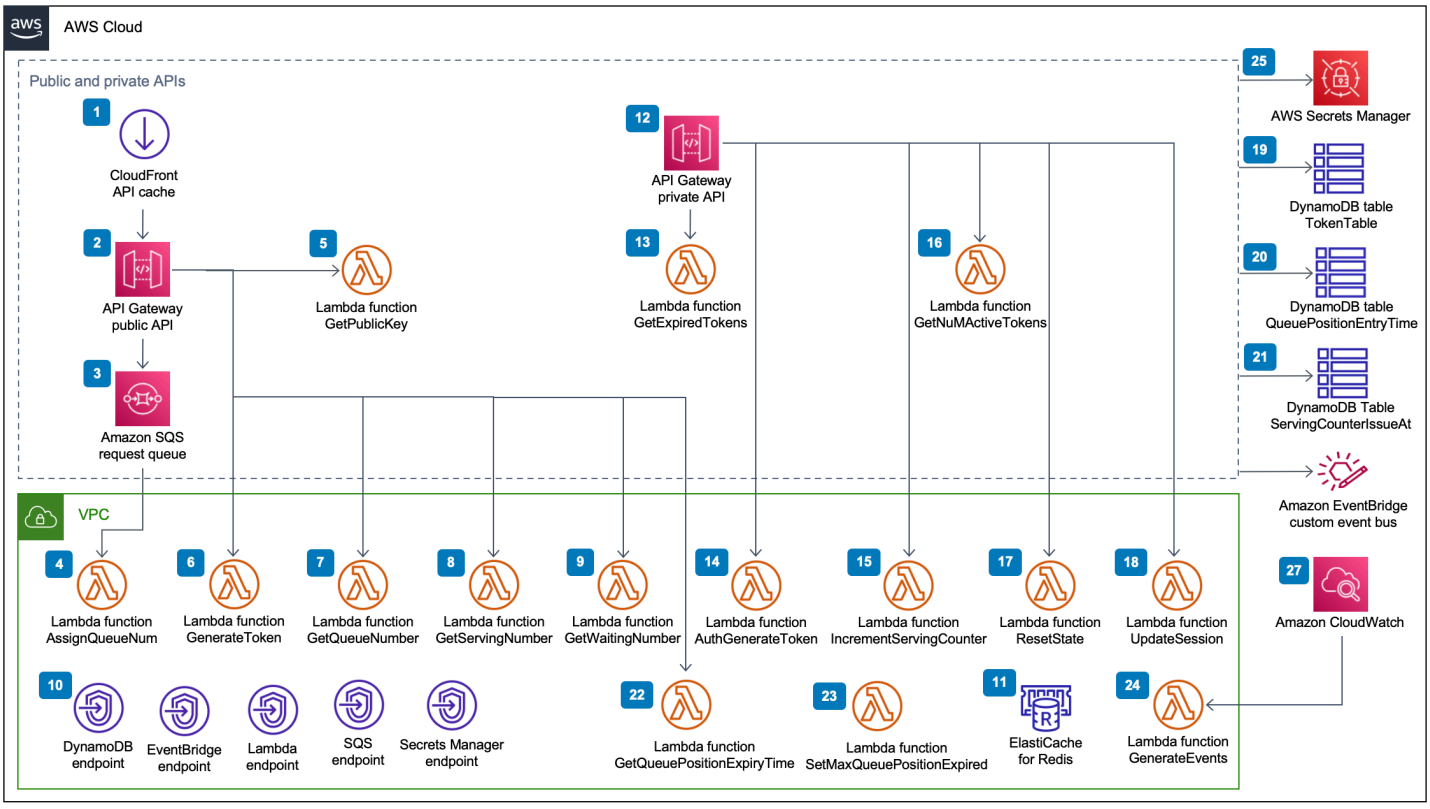
解决方案组件

公共和私人等候室 APIs

AWS 解决方案上的虚拟等候室的主要目的是以可控的方式控制客户JSON网络令牌 (JWT) 的生成，以避免大量新用户涌入可能使目标网站不堪重负。JWTs可用于站点保护，在获得等候室令牌之前阻止访问网页，也可以用于API访问授权。

核心模板安装了公共和API私人（IAM已授权），API用于大多数虚拟等候室的 AWS 操作。根据API路径API为公众配置了具有多个缓存策略的 CloudFront分发。创建了 DynamoDB 表 EventBridge和事件总线。该模板添加了一个VPC包含两个可用区 (AZs) 的新可用区，两个可用区中都有一个 Elasticache (RedisOSS) 集群AZs，还有几个 Lambda 函数。与 Elasticache (RedisOSS) 交互的 Lambda 函数在内部具有网络接口，而VPC所有其他 Lambda 函数都具有默认的网络连接。核心APIs是与解决方案交互的最低层。其他 Lambda 函数、Amazon Elastic Compute Cloud EC2 (Amazon) 实例和容器可以充当扩展，调用核心APIs来构建等候室、控制入口流量并对解决方案生成的事件做出反应。

此外，核心堆栈会针对其所有 Lambda 函数错误和限制条件创建警报，并为每个 API Gateway 部署的 4XX 和 5XX 状态代码创建警报。



AWS公共和私有APIs组件上的虚拟等候室

1. CloudFront 分发会为客户端提供公共API调用，并在适当时缓存结果。
2. Amazon API Gateway 公开API处理来自虚拟等候室的队列请求，跟踪队列位置，并支持验证允许访问目标网站的令牌。
3. SQSqueue 控制流向处理队列消息的 AWS Lambda 函数的流量。
4. AssignQueueNumLambda 函数验证其批次收到的每条消息，在 Elasticache (Redis) 中增加队列计数器，并将每个请求及其关联的队列位置存储在 Elasticache (RedisOSS) 中。OSS
5. GetPublicKeyLambda 函数从 Secrets Manager 中检索公钥值。
6. JWT对于允许在目标站点完成交易的有效请求，GenerateTokenLambda 函数会生成一个。它向等候室的自定义事件总线写入一个已生成令牌的事件。如果之前已经为此请求生成了令牌，则不会生成任何新令牌。
7. GetQueueNumberLambda 函数从 Elasticache (Redis) 检索并返回客户端在队列中的数字位置。OSS
8. GetServingNumberLambda 函数从 Elasticache (Redis) 检索并返回等候室当前正在提供的号码。OSS
9. GetWaitingNumLambda 函数返回当前在等候室排队但尚未发放令牌的号码。
10. VPC终端节点允许中的 Lambda 函数VPC与解决方案中的服务进行通信。
11. Elasticache (RedisOSS) 集群使用有效的事件 ID 存储所有进入等候室的请求。它还存储多个计数器，例如已排队的请求数、当前正在处理的请求数、生成的令牌数量、已完成的会话数和放弃的会话数。
12. API网关私有API资源以支持管理职能。私APIs有 AWS IAM经过身份验证。
13. GetExpiredTokensLambda 函数返回包含过期令牌IDs的请求列表。
14. AuthGenerateTokenLambda 函数为允许在目标站点完成交易的有效请求生成令牌。在核心堆栈部署期间最初设置的令牌的发行者和有效期可以被覆盖。它向等候室的自定义事件总线写入一个已生成令牌的事件。如果之前已经为此请求生成了令牌，则不会生成任何新令牌。
15. 如果按值递增，IncrementServingCounterLambda 函数会递增存储在 Elasticache (RedisOSS) 中的等候室服务计数器。
16. GetNumActiveTokensLambda 函数向 DynamoDB 查询尚未过期、尚未用于完成交易以及尚未标记为已放弃的代币数量。
17. ResetStateLambda 函数会重置存储在 Elasticache (Redis) 中的所有计数器。OSS它还会删除和重新创建TokenTableQueuePositionEntryTime、和 ServingCounterIssuedAt DynamoDB 表。此外，它还会执行 CloudFront 缓存失效。

18.UpdateSessionLambda 函数更新存储在 DynamoD TokenTable B 表中的会话（令牌）的状态。会话状态用整数表示。设置为状态的会话1表示已完成，-1表示已放弃。它向等候室的自定义事件总线写入会话已更新的事件。

19.TokenTableDynamoDB 表存储令牌数据。

20.QueuePositionEntryTimeDynamoDB 表存储队列位置和进入时间数据。

21.ServingCounterIssuedAtDynamoDB 表存储服务计数器的更新。

22.当客户端请求剩余队列位置到期时间时，就会调用 GetQueuePositionExpireTime Lambda 函数。

23.SetMaxQueuePositionExpiredLambda 函数根据ServingCounterIssuedAt表值设置已过期的最大队列位置。如果在核心堆栈部署true期间将IncrSvcOnQueuePositionExpiry参数设置为，则它每分钟运行一次。

24.GenerateEventsLambda 函数将各种等候室指标写入等候室的自定义事件总线。如果在核心堆栈部署true期间将 “启用事件生成” 参数设置为，则每分钟运行一次。

25.AWS Secrets Manager 存储令牌操作的密钥和其他敏感数据。

26.每当生成令牌并更新 TokenTable DynamoDB 表中的会话时，Amazon EventBridge 自定义事件总线都会收到一个事件。当发球计数器在 SetMaxQueuePositionExpired Lambda 中移动时，它还会接收事件。如果在核心堆栈部署期间激活，则会写入各种等候室指标。

27.如果在核心堆栈部署期间将 “启用事件生成” 参数设置为 true，则会创建 Amazon CloudWatch 事件规则。此事件规则每分钟启动一次 Lamb GenerateEvents da 函数。

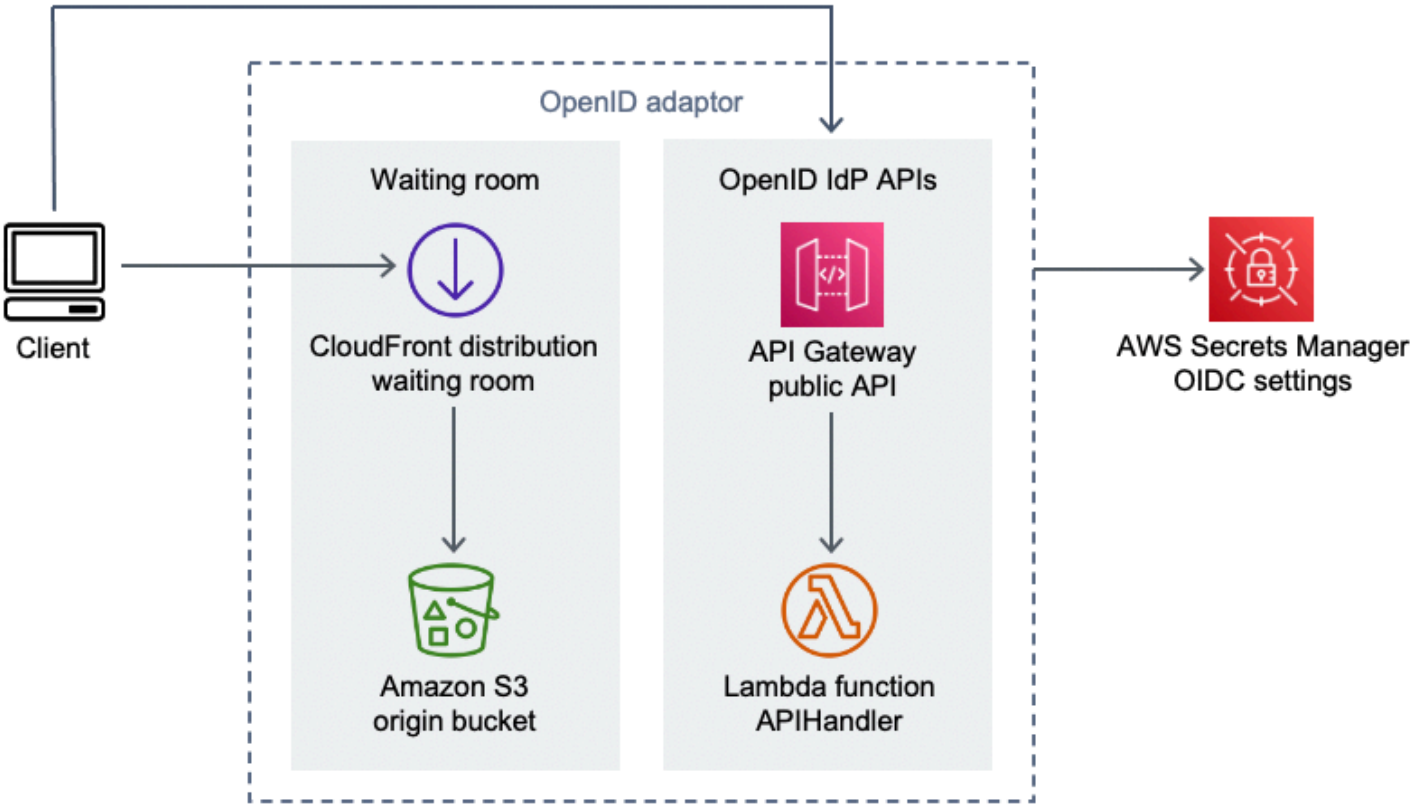
授权方

该解决方案包括API网关 Lambda 授权方堆栈。堆栈由一个IAM角色和一个 Lambda 函数组成。APIGatewayAuthorizerLambda 函数是 API Gateway 的授权者，可以验证虚拟等候室在上发布的令牌的签名和声明。 AWS API堆栈附带的 Lambda 函数可用于保护云，APIs直到用户通过等候室并收到访问令牌。授权方会自动从核心中检索和缓存公钥和配置，API以进行令牌验证。它无需修改即可使用，并且可以安装在任何支持的 AWS 区域 AWS Lambda。

OpenID 适配器

OpenID 适配器堆栈部署了充当 OpenID 身份提供商的网关API和 Lambda 函数。OpenID 适配器提供了一组OIDC兼容的APIs，可以与支持OIDC身份提供商（例如 AWS 弹性负载均衡器）的现有虚拟主机软件一起使用，也可以用作 Amazon Cognito 或类似服务的联合身份提供商。 WordPress该适配器允许客户在使用集成选项有限的虚拟 off-the-shelf 主机软件时使用 Authn/Authz 流程中的等候室。堆栈

还安装了一个 CloudFront 分配，其中一个 Amazon S3 存储桶作为源，另一个 S3 存储桶用于记录请求。OpenID 适配器提供一个示例等候室页面，与示例等候室堆栈中提供的页面类似，但专为 OpenID 身份验证流程而设计。通过身份验证的过程包括在等候室队列中获得一个位置，然后等到服务位置等于或大于客户的队列位置。OpenID 等候室页面重定向回目标站点，目标站点使用 OpenID API 完成客户端的令牌获取和会话配置。该解决方案的API端点直接映射到官方的 OpenID Connect 1.0 流量规范。name-for-name有关详细信息，请参阅 OpenID Connect Core 1.0 身份验证。



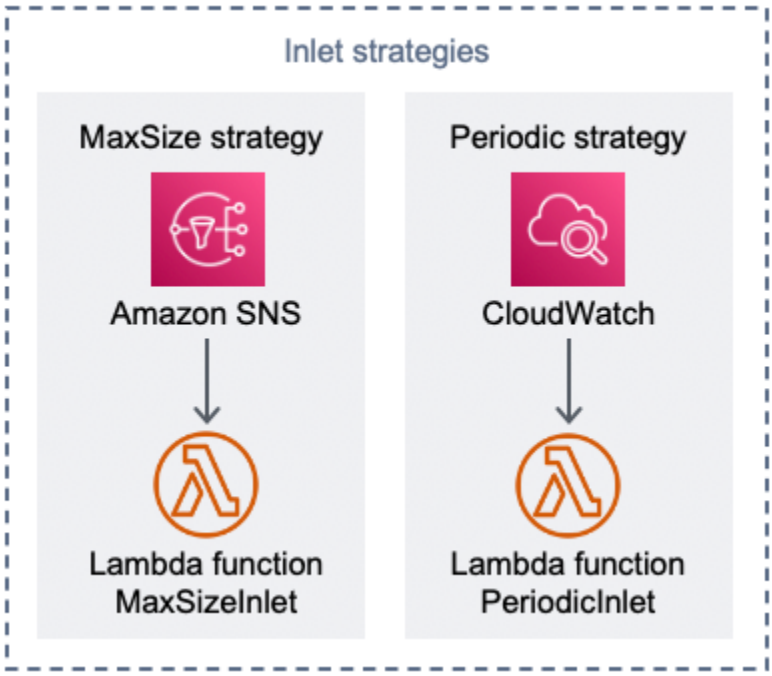
AWS OpenID 适配器组件上的虚拟等候室

1. CloudFront 分发将 S3 存储桶的内容提供给用户。
2. S3 存储桶主机等候室页面示例。
3. Amazon API Gateway API 提供了一组OIDC兼容的APIs，可与支持OIDC身份提供商的 Lambda 授权功能的现有虚拟主机软件一起使用。
4. `APIHandler`Lambda 函数处理所有API网关资源路径的请求。同一模块中的不同 Python 函数映射到每个API路径。例如，API网关中的`/authorize`资源路径在 Lambda 函数`authorize()`中调用。
5. OIDC设置存储在 Secrets Manager 中。

样本入口策略

入口策略决定解决方案的服务柜台何时应向前移动，以容纳目标站点中的更多用户。有关等候室入口策略的更多概念性信息，请参阅设计注意事项。

该解决方案提供了两种样本入口策略：MaxSize和定期。



AWS 入口策略组件上的虚拟等候室

最大尺寸入口策略选项：

1. 客户发出 Amazon SNS 通知，该通知调用 Lamb `MaxSizeInlet` da 函数以根据消息有效负载增加服务计数器。
2. `MaxSizeInletLambda` 函数期望收到一条消息，它使用它来确定要增加多少服务计数器。

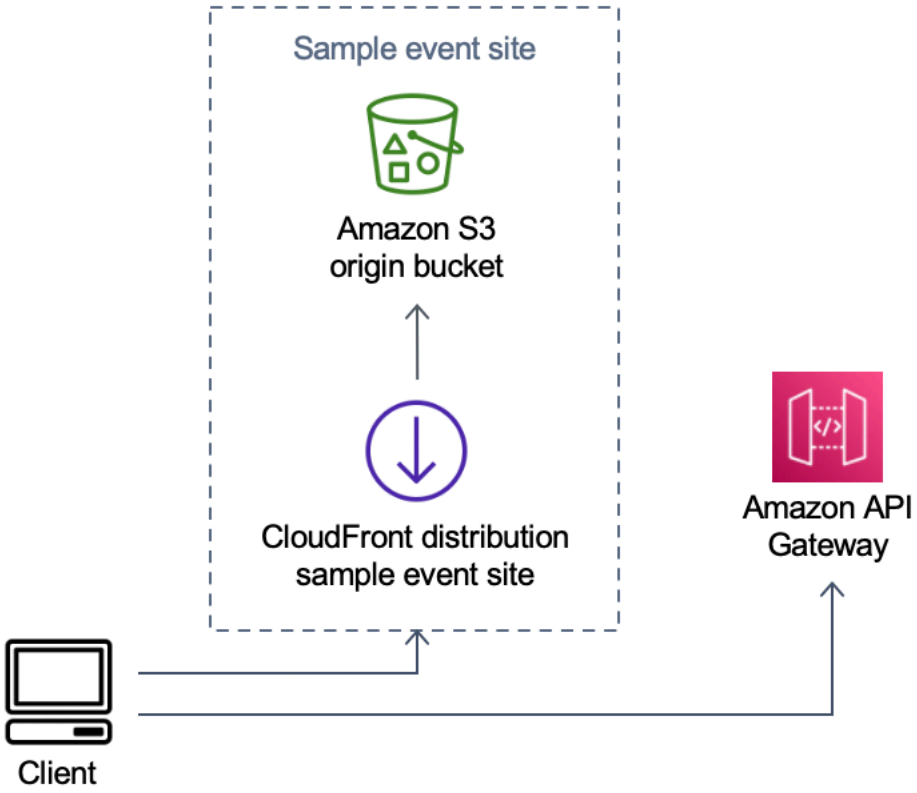
定期入口策略选项：

3. CloudWatch 规则每分钟调用一次 Lambda 函数，将发球计数器增加固定数量。
4. 如果时间介于提供的开始时间和结束时间之间，`PeriodicInletLambda` 函数将按给定大小递增服务计数器。或者，它会检查 CloudWatch 警报，如果警报处于OK状态，则执行增量，否则跳过警报。

等候室示例

除了自定义授权者之外，示例等候室还与公共和私有APIs等候室集成，以演示最小的 end-to-end 等候室解决方案。主网页存储在 S3 存储桶中，并用作其来源 CloudFront。它引导用户完成以下步骤：

1. 在等候室排队等候进入现场。
2. 获取客户的排队位置。
3. 获取等候室的发球位置。
4. 一旦发球位置等于或大于客户头寸，即可获得代币集。
5. 使用该令牌调用受 Lambda 授权方API保护的。



AWS 示例活动网站组件上的虚拟等候室

1. S3 存储桶托管等候室和控制面板的示例内容。
2. CloudFront 分发将 S3 存储桶内容提供给用户。

3. 使用类似购物的资源路径（如和）的API网关部署示例。`/search` `/checkout`API它由堆栈安装并使用令牌授权器进行配置。它旨在作为在等候室保护人的简单方法API的示例。提供有效令牌的请求会被转发到 Lambda，否则会返回错误。除了附加的 Lambda 函数的响应外，没有任何API其他功能。

安全性

当您在 AWS 基础架构上构建系统时，安全责任由您和共同承担 AWS。这种共享模式减轻了您的运营负担，因为您可以 AWS 操作、管理和控制组件，包括主机操作系统、虚拟化层和服务运行设施的物理安全。有关 AWS 安全的更多信息，请访问AWS 云安全。

Elasticache (RedisOSS) 在私有网络内部被分配了一个网络接口。VPC与 Elasticache (RedisOSS) 交互的 Lambda 函数也被分配到内部的网络接口。VPC所有其他资源在共享网络空间中都具有 AWS 网络连接。带有与其他 AWS 服务交互的VPC接口的 Lambda 函数使用VPC终端节点连接到这些服务。

用于创建和验证JSON网络令牌的公钥和私钥是在部署时生成的，并存储在 Secrets Manager 中。用于连接 Elasticache (RedisOSS) 的密码也是在部署时生成的，并存储在 Secrets Manager 中。任何解决方案都无法访问私钥和 Elasticache (RedisOSS) 密码。API

API必须通过以下方式访问公众 CloudFront。该解决方案为 Gate API way 生成一个API密钥，该密钥用作中自定义标头的值 CloudFront。`x-api-key` CloudFront 发出原始请求时包含此标头。有关更多详情，请参阅《Amazon CloudFront 开发者指南》中的向源请求添加自定义标头。

私APIs有配置为需要 AWS IAM授权才能调用。该解决方案创建具有适当权限的`ProtectedAPIGroup`IAM用户组来调用私有用户APIs。添加到该组的IAM用户有权调用私有APIs。

IAM在角色中使用的策略以及附加到解决方案创建的各种资源的权限仅授予执行必要任务所需的权限。

对于解决方案生成的 S3 存储桶、SQS队列和SNS主题等资源，只要有可能，就会激活静态和传输期间的加密。

监控

核心API堆栈包括多个 CloudWatch 警报，可以在解决方案运行时监控这些警报以检测问题。堆栈会针对 Lambda 函数错误和限制条件创建警报，`ALARM`如果在一分钟内出现错误或限制条件，则该堆栈会`OK`将警报状态从更改为。

该堆栈还会为 4XX 和 5XX 状态代码的每个API网关部署创建警报。`ALARM`如果在一API分钟内返回 4XX 或 5XX 状态码，则警报的状态将从变为。`OK`

这些警报在没有错误或限制一分钟后恢复到`OK`状态。

IAM角色

AWS Identity and Access Management (IAM) 角色允许客户向 AWS 云上的服务和用户分配精细的访问策略和权限。此解决方案创建的IAM角色可向解决方案的 AWS Lambda 功能授予创建区域资源的访问权限。

Amazon CloudFront

该`virtual-waiting-room-on-aws.template` CloudFormation 模板创建了等候室APIs的核心公共和私有内容，还为公众API部署了 CloudFront 分发版。 CloudFront 缓存来自公众的响应API，从而减少API网关和执行工作的 Lambda 函数的负载。

该解决方案还有一个可选的等候室模板示例，用于部署托管在亚马逊简单存储服务 (Amazon S3) 存储桶中的简单 Web 应用程序。为了帮助减少延迟和提高安全性，部署了带有原始访问身份的 Amazon CloudFront 分配，该身份是提供对解决方案网站存储桶内容的公开访问权限的 CloudFront 用户。有关更多信息，请参阅《亚马逊 CloudFront 开发者指南》中的使用源站访问身份限制对 Amazon S3 内容的访问。

安全组

在此解决方案中创建VPC的安全组旨在控制和隔离流向 Elasticache (RedisOSS) 的网络流量。需要与 Elasticache (RedisOSS) 通信的 Lambda 与 Elasticache (Redis) 位于同一个安全组中。OSS我们建议您在部署启动并运行后查看安全组并根据需要进一步限制访问权限。

设计注意事项

部署选项

如果这是第一次安装，或者您不确定要安装什么，请部署`virtual-waiting-room-on-aws-getting-started.template`嵌套 CloudFormation模板，该模板将安装核心、授权者和等候室示例模板。这为您提供了一个最小的等候室，流程简单。

受支持的协议

AWS 解决方案上的虚拟等候室可以与以下内容集成：

- JSONWeb 令牌验证库和工具
- 现有API网关部署
- RESTAPI客户
- OpenID 客户和提供商

等候室入口策略

入口策略封装了将客户从等候室转移到网站所需的逻辑和数据。入口策略可以作为 Lambda 函数、容器、Amazon EC2 实例或任何其他计算资源来实现。只要它可以将等候室称为公用和私有即可，它就不必是云资源APIs。入口策略接收有关等候室、网站或其他外部指标的事件，以帮助其决定何时可以发行代币并进入网站。入口策略有几种方法。你采用哪一种取决于你可用的资源以及受保护的网站设计的限制。

入口策略采取的主要操作是将 `increment_serving_num` Amazon API Gateway 设为私API有，其相对值表示还有多少客户可以进入该网站。本节介绍两种样本入口策略。它们可以按原样使用，也可以自定义，也可以采用完全不同的方法。

MaxSize

使用该 MaxSize 策略，将 `MaxSizeInlet` Lambda 函数配置为可以同时使用网站的最大客户数。这是一个固定值。客户发出 Amazon SNS 通知，该通知调用 Lamb `MaxSizeInlet` da 函数以根据消息有效负载增加服务计数器。SNS消息的来源可以来自任何地方，包括网站上的代码或观察网站利用率的监控工具。

MaxSizeInletLambda 函数预计会收到一条消息，其中可能包括：

- exited :已完成的交易数量
- 待标记IDs为已完成的请求清单
- IDs要标记为已放弃的请求清单

此数据用于确定服务计数器要增加多少。在某些情况下，根据当前的客户端数量，可能没有额外的容量来增加计数器。

定期

使用周期策略时， CloudWatch 规则每分钟调用 Lambd PeriodicInlet a 函数，将发球计数器增加固定数量。定期入口使用事件开始时间、结束时间和增量进行参数化。或者，此策略还会检查 CloudWatch 警报，如果警报处于OK状态，则执行增量，否则会跳过警报。站点集成商可以将利用率指标连接到警报，并使用该警报来暂停定期入口。此策略仅在当前时间介于开始时间和结束时间之间时更改发球位置，并且可以选择指定警报处于OK状态。

定制和扩展解决方案

贵组织的站点管理员必须决定在等候室中使用的集成方法。有两个选项：

1. 直接使用APIs和API网关授权方进行基本集成。
2. 通过身份提供商集成 OpenID。

除了上述集成之外，您可能还需要配置域名重定向。您还负责部署自定义的等候室网站页面。

Virtual Waiting Room on AWS 解决方案旨在通过两种机制进行扩展： EventBridge 用于单向事件通知和RESTAPIs用于双向通信。

配额

虚拟等候室开启的主要规模限制 AWS 是已安装区域的 Lambda 限制限制。 AWS 当安装到具有默认 Lambda 并发运行配额的 AWS 账户中时，虚拟等候室 AWS 解决方案每秒最多可以处理 500 个客户请求队列中的职位。每秒 500 个客户端的速率基于该解决方案，所有 Lambda 函数并发配额限制都是独家提供的。如果账户中的区域与其他调用 Lambda 函数的解决方案共享，则 AWS 解决方案的虚拟等候室应至少有 1,000 个可用的并发调用。您可以使用 CloudWatch 指标来绘制账户中随时间变化的

Lambda 并发调用次数来做出决定。您可以使用 S ervice Quotas 控制台请求增加配额。提高 Lambda 限制仅在实际发生额外调用时才会提高每月账户费用。

每秒每增加 500 个客户端，将您的油门限制提高 1,000。

预计每秒传入用户数	建议的并发执行配额
0-500	1,000（默认）
501-1,000	2000
1,001-1,500	3000

Lambda 的固定突发限制为 3,000 次并发调用。有关更多信息，请参阅 Lambda 函数扩展。如果返回的错误代码表示暂时出现节流情况，则客户端代码应该预料并重试一些API调用。等候室客户端示例包含此代码，以此作为如何设计用于高容量和高突发事件的客户端的示例。

此解决方案还兼容 Lambda 预留和预配置并发以及自定义配置步骤。有关详细信息，请参阅管理 Lambda 预留并发。

可以进入等候室、接收令牌并继续进行交易的用户上限受到 Elasticache (Redis) 计数器OSS上限的限制。计数器用于等候室的服务位置和跟踪解决方案的摘要状态。Elasticache（RedisOSS）中使用的计数器的上限为 9,223,372,036,854,775,807。DynamoDB 表用于存储发放给等候室用户的每个令牌的副本。DynamoDB 对表格的大小没有实际限制。

区域部署

该解决方案使用的服务在所有 AWS 区域均受支持。有关按地区划分的最新 AWS 服务可用性，请参阅AWS 区域服务列表。

AWS CloudFormation 模板

为了实现自动部署，此解决方案使用以下 AWS CloudFormation 模板，您可以在部署前下载这些模板。

如果这是第一次安装，或者您不确定要安装什么，请部署模板，该`virtual-waiting-room-on-aws-getting-started.template` AWS CloudFormation 模板将安装核心、授权方和等候室代码模板示例。这使您可以通过简单的流程测试正在运行的等候室。

View template

virtual-waiting-room-on-aws-api-gateway-cw-logs-role.templ ate：使用此模板在账户级别向 Gate API way 添加默认角色ARN以获得日志权限。 CloudWatch 有关您的账户是否需要部署此模板的详细信息，请参阅先决条件。

View template

virtual-waiting-room-on-aws-getting-started .templat e：使用此嵌套模板安装核心、授权方和样本等候室堆栈。

View template

virtual-waiting-room-on-aws.templat e：使用此核心模板安装用于创建等候室活动的核心公共RESTAPIs、私有和云服务。将此模板安装在需要等候室RESTAPIs、Elasticache (RedisOSS) 和 DynamoDB 表的账户和区域。

View template

virtual-waiting-room-on-aws-authorizers.temp late：使用此模板安装 Lambda 授权器，该授权器旨在验证等候室发放的令牌，旨在保护最终用户。APIs需要核心堆栈。部署此堆栈需要一些来自核心堆栈的输出作为参数。这是一个可选的模板。

View template

virtual-waiting-room-on-aws-openid.tem plate：使用此模板安装 OpenID 身份提供程序，以便在等候室与授权接口集成。需要核心堆栈。部署此堆栈需要来自核心堆栈的一些输出。这是一个可选的模板。

View template

virtual-waiting-room-on-aws-sample-inlet-strategy .template：使用此模板安装用于目标站点和等候室之间的样本入口策略。入口策略有助于封装逻辑，以确定何时允许更多用户进入目标站点。需要核心堆栈。部署此堆栈需要核心堆栈的输出。这是一个可选的模板。

View template

virtual-waiting-room-on-aws-sample.temp le：使用此模板为等候室和目标站点安装最小 Web 和 API Gateway 配置示例。需要核心和授权堆栈。部署此堆栈需要将核心堆栈和授权方堆栈的输出作为参数。这是一个可选的模板。

自动部署

在启动解决方案之前，请查看本指南中讨论的成本、架构、网络安全和其他注意事项。按照本节中的 step-by-step 说明配置解决方案并将其部署到您的账户。

部署时间：大约 30 分钟（仅限入门堆栈）

先决条件

- AWS 账户控制台权限等同于管理员访问权限。
- 从API网关激活 CloudWatch 日志记录：
  - 登录 APIGateway 控制台并选择您计划安装堆栈的区域。

    如果您已在此区域中APIs定义：

    1. 选择任何API。
    2. 从左侧导航栏中选择 “设置”。
    3. 检查CloudWatch 日志角色ARN字段中的值。
    - 如果没有ARN，请安装virtual-waiting-room-on-aws-api-gateway-cw-logs-role.template。
    - 如果有ARN，请从启动入门堆栈开始。

    如果此区域中未APIs定义任何现有内容，请安装virtual-waiting-room-on-aws-api-gateway-cw-logs-role.template。
- 了解要保护的目标站点的架构和实施细节。

部署概述

使用以下步骤在上部署此解决方案 AWS。有关详细说明，请访问每个步骤的链接。

第 1 步。启动入门堆栈

- 将 AWS CloudFormation 模板启动到您的 AWS 账户。
- 查看模板参数并根据需要输入或调整默认值。

第 2 步。（可选）测试等候室

- 生成 AWS 密钥以呼叫受IAM保护者APIs。
- 打开样品等候室的控制面板。
- 测试样品等候室。

第 1 步。启动入门堆栈

此自动 AWS CloudFormation 模板部署了核心、授权者和样本等候室模板，允许您查看和测试正在运行的等候室。在启动堆栈之前，您必须阅读并理解先决条件。

> **Note**
>
> 运行此解决方案时使用的 AWS 服务费用由您承担。有关更多详细信息，请访问本指南中的 “成本” 部分，并参阅本解决方案中使用的每项 AWS 服务的定价网页。

1. 登录AWS Management Console并选择按钮以启动 `virtual-waiting-room-on-aws-getting-started.template` AWS CloudFormation 模板。

   Launch solution

   或者，您可以下载模板作为自己实现的起点。
2. 默认情况下，该模板在美国东部（弗吉尼亚州北部）区域启动。要在其他 AWS 区域启动解决方案，请使用控制台导航栏中的区域选择器。
3. 在创建堆栈页面上，确认 Amazon S3 URL 文本框中的模板URL是否正确，然后选择下一步。
4. 在指定堆栈详细信息页面上，为您的解决方案堆栈分配一个名称。有关命名字符限制的信息，请参阅《AWS Identity and Access Management 用户指南》中的IAM和STS限制。
5. 在 “参数” 下，查看此解决方案模板的参数并根据需要进行修改。该解决方案使用以下默认值。

参数	默认值	描述
活动编号	`Sample`	此等候室实例的唯一 ID，建议GUID采用格式。
有效期	`3600`	令牌有效期（以秒为单位）。

参数	默认值	描述
启用事件生成	`false`	如果设置为`true`，则每分钟将与等候室相关的指标写入其事件总线
Elasticache (Redis) 端口 OSS	`1785`	用于连接到 Elasticache (RedisOSS) 服务器的端口号。建议不要使用默认的 Elasticache (RedisOSS) 端口。6379
EnableQueuePositionExpiry	`true`	如果设置为`false`，则不应用队列位置到期时间。
QueuePositionExpiryPeriod	`900`	这是以秒为单位的时间间隔，超过该时间间隔，队列位置就没有资格生成令牌。
IncrSvcOnQueuePosi tionExpiry	`false`	如果设置为`true`，则根据未成功生成令牌的过期队列位置自动向前推送计数器。

6. 请选择 Next（下一步）。
7. 在 配置堆栈选项 页面上，请选择 下一步。
8. 在 Review 页面上，审核并确认设置。选中确认模板创建 AWS Identity and Access Management (IAM) 资源的复选框。
9. 选择 Create stack（创建堆栈）以部署堆栈。

   您可以在 AWS CloudFormation 控制台的 “状态” 列中查看堆栈的状态。您将在大约 30 分钟后收到 CREATE _ COMPLETE 状态。

第 2 步。（可选）测试等候室

如果您部署了入门堆栈，则以下步骤可帮助您测试等候室的功能。要完成测试，您需要具有调用核心堆栈APIs中IAM受保护的权限的 AWS 密钥。

生成 AWS 密钥以致电受IAM保护者 APIs

1. 在部署aws-virtual-waiting-room-getting-started.template CloudFormation 模板的 AWS 账户中@@ 创建或使用IAM用户。
2. 向IAM用户授予编程访问权限。为IAM用户创建一组新的访问密钥时，请在出现密钥文件时下载密钥文件。您需要IAM用户的访问密钥 ID 和私有访问密钥来测试等候室。
3. 将IAM用户添加到模板创建的 P rotectedAPIGroup IAM 用户组。

打开样品等候室的控制面板

1. 登录AWS CloudFormation 控制台并选择解决方案的入门堆栈。
2. 选择输出选项卡。
3. 在键列下 ControlPanelURL，找到并选择相应的值。
4. 在新选项卡或浏览器窗口中打开控制面板。
5. 在 “控制面板” 中，展开 “配置” 部分。
6. 输入您在生成密钥中检索到的访问密钥 ID 和私有访问 AWS 密钥以呼叫IAM安全密钥APIs。端点和事件 ID 是从URL参数中填写的。
7. 选择 “使用”。在您提供凭据后，该按钮即会激活。

测试样品等候室

1. 在AWS CloudFormation 控制台中，选择解决方案的入门堆栈。
2. 选择输出选项卡。
3. 在键列下 WaitingRoomURL，找到并选择相应的值。
4. 打开等候室，然后选择 “预订” 进入等候室。
5. 导航回带有控制面板的浏览器选项卡。
6. 在 “增量供应计数器” 下，选择 “更改”。这允许 100 名用户从等候室移动到目标站点。
7. 导航回等候室并选择 “立即结账”！现在，您将被重定向到目标站点。
8. 选择 “立即购买”，在目标站点完成交易。

部署单独的堆栈

核心堆栈是获得等候室主要功能所需的唯一堆栈。所有其他堆栈都是可选的。如果您还没有办法验证等候室发放的令牌或保护任何可能已经拥有的代币，APIs请启动授权者堆栈。如果您需要 OpenID 身份提供者来将等候室与授权接口集成，请启动 OpenID 堆栈。样本入口策略堆栈提供了几个示例，说明如何以及何时允许更多用户进入你想要保护的网站。

1. 启动核心堆栈

部署用时：大约 20 分钟

此自动 AWS CloudFormation 模板 AWS 在 AWS 云端部署虚拟等候室。在启动堆栈之前，您必须完成先决条件。

> **Note**
>
> 运行此解决方案时使用的 AWS 服务费用由您承担。有关更多详细信息，请访问本指南中的 “成本” 部分，并参阅本解决方案中使用的每项 AWS 服务的定价网页。

1. 登录AWS Management Console并选择按钮以启动`aws-virtual-waiting-room-on-aws.template` AWS CloudFormation 模板。

   Launch solution

   或者，您可以下载模板作为自己实现的起点。
2. 默认情况下，该模板在美国东部（弗吉尼亚州北部）区域启动。要在其他 AWS 区域启动解决方案，请使用控制台导航栏中的区域选择器。
3. 在创建堆栈页面上，确认 Amazon S3 URL 文本框中的模板URL是否正确，然后选择下一步。
4. 在指定堆栈详细信息页面上，为您的解决方案堆栈分配一个名称。有关命名字符限制的信息，请参阅《AWS Identity and Access Management 用户指南》中的IAM和STS限制。
5. 在 “参数” 下，查看此解决方案模板的参数并根据需要对其进行修改。该解决方案使用以下默认值。

参数	默认值	描述
活动编号	Sample	此等候室实例的唯一 ID，建议GUID采用格式。
有效期	3600	令牌有效期（以秒为单位）。
启用事件生成	false	如果设置为true，则每分钟将与等候室相关的指标写入其事件总线。
Elasticache (Redis) 端口 OSS	1785	用于连接到 Elasticache (RedisOSS) 服务器的端口号。建议不要使用默认的 Elasticache (RedisOSS) 端口。6379
EnableQueuePositionExpiry	true	如果设置为false，则不应用队列位置到期时间。
QueuePositionExpiryPeriod	900	这是以秒为单位的时间间隔，超过该时间间隔，队列位置就没有资格生成令牌。
IncrSvcOnQueuePosi tionExpiry	false	如果设置为true，则根据未成功生成令牌的过期队列位置自动向前推送计数器。

6. 请选择 Next（下一步）。
7. 在 配置堆栈选项 页面上，请选择 下一步。
8. 在 Review 页面上，审核并确认设置。选中确认模板创建 AWS Identity and Access Management (IAM) 资源的复选框。
9. 选择 Create stack（创建堆栈）以部署堆栈。

   您可以在 AWS CloudFormation 控制台的 “状态” 列中查看堆栈的状态。您将在大约 20 分钟后收到 CREATE _ COMPLETE 状态。

2.（可选）启动授权者堆栈

部署用时：大约五分钟

1. 登录AWS Management Console并选择按钮以启动 `aws-virtual-waiting-room-on-aws-authorizers.template` AWS CloudFormation 模板。

   Launch solution

   或者，您可以下载模板作为自己实现的起点。
2. 默认情况下，该模板在美国东部（弗吉尼亚州北部）区域启动。要在其他 AWS 区域启动解决方案，请使用控制台导航栏中的区域选择器。
3. 在创建堆栈页面上，确认 Amazon S3 URL 文本框中的模板URL是否正确，然后选择下一步。
4. 在指定堆栈详细信息页面上，为您的解决方案堆栈分配一个名称。有关命名字符限制的信息，请参阅《AWS Identity and Access Management 用户指南》中的IAM和STS限制。
5. 在“参数”下，查看此解决方案模板的参数并根据需要对其进行修改。该解决方案使用以下默认值。

参数	默认值	描述
公共API端点	*<Requires input>*	虚拟等候室的公共端点APIs。
等候室事件 ID	`Sample`	等候室的事件 ID。
发行人 URI	*<Requires input>*	公钥和令牌URI的发行者。

6. 请选择 Next（下一步）。
7. 在 配置堆栈选项 页面上，请选择 下一步。
8. 在 Review 页面上，审核并确认设置。选中确认模板创建 AWS Identity and Access Management (IAM) 资源的复选框。
9. 选择 Create stack（创建堆栈）以部署堆栈。

   您可以在 AWS CloudFormation 控制台的“状态”列中查看堆栈的状态。大约五分钟后，您应该会收到 CREATE _ COMPLETE 状态。

3. （可选）启动 OpenID 堆栈

部署用时：大约五分钟

1. 登录AWS Management Console并选择按钮以启动 `aws-virtual-waiting-room-on-aws-openid.template` AWS CloudFormation 模板。

   Launch solution

   或者，您可以下载模板作为自己实现的起点。
2. 默认情况下，该模板在美国东部（弗吉尼亚州北部）区域启动。要在其他 AWS 区域启动解决方案，请使用控制台导航栏中的区域选择器。
3. 在创建堆栈页面上，确认 Amazon S3 URL 文本框中的模板URL是否正确，然后选择下一步。
4. 在指定堆栈详细信息页面上，为您的解决方案堆栈分配一个名称。有关命名字符限制的信息，请参阅《AWS Identity and Access Management 用户指南》中的IAM和STS限制。
5. 在“参数”下，查看此解决方案模板的参数并根据需要对其进行修改。该解决方案使用以下默认值。

参数	默认值	描述
公共API端点	*<Requires input>*	虚拟等候室URL的公共端点APIs。
私有API端点	*<Requires input>*	虚拟等候室URL的私有端点APIs。
API区域	*<Requires input>*	AWS 公共和私人等候室的地区名称APIs。
活动编号	`Sample`	等候室的事件 ID。

6. 请选择 Next（下一步）。
7. 在 配置堆栈选项 页面上，请选择 下一步。
8. 在 Review 页面上，审核并确认设置。选中确认模板创建 AWS Identity and Access Management (IAM) 资源的复选框。
9. 选择 Create stack（创建堆栈）以部署堆栈。

您可以在 AWS CloudFormation 控制台的 “状态” 列中查看堆栈的状态。大约五分钟后，您应该会收到 CREATE _ COMPLETE 状态。

4.（可选）启动样本入口策略堆栈

部署时间：大约两分钟

1. 登录AWS Management Console并选择按钮以启动`aws-virtual-waiting-room-sample-inlet-strategy.template` AWS CloudFormation 模板。

   或者，您可以下载模板作为自己实现的起点。
2. 默认情况下，该模板在美国东部（弗吉尼亚州北部）区域启动。要在其他 AWS 区域启动解决方案，请使用控制台导航栏中的区域选择器。
3. 在创建堆栈页面上，确认 Amazon S3 URL 文本框中的模板URL是否正确，然后选择下一步。
4. 在指定堆栈详细信息页面上，为您的解决方案堆栈分配一个名称。有关命名字符限制的信息，请参阅《AWS Identity and Access Management 用户指南》中的IAM和STS限制。
5. 在 “参数” 下，查看此解决方案模板的参数并根据需要对其进行修改。该解决方案使用以下默认值。

参数	默认值	描述
活动编号	`Sample`	等候室的事件 ID。
私有核心API端点	*`<Requires input>`*	虚拟等候室URL的私有端点APIs。
核心API区域	*`<Requires input>`*	AWS 安装核心API的区域。
入口策略	`Periodic`	有待部署的入口策略。`Periodic`每分钟递增一次服用次数。 `MaxSize`根据下游目标站点在给定时间可以处理的最大事务数来增加服务数量。

参数	默认值	描述
增量依据	*<Requires input>*	发球计数器应每分钟递增多少。如果选择定期入口策略，则为必填项。
开始时间	*<Requires input>*	关于何时开始递增服务数的时间戳（以秒为单位的纪元时间）。如果选择定期入口策略，则为必填项。
结束时间	*<Requires input>*	何时停止增加服务数的时间戳（以秒为单位的纪元时间）。如果留下 0，则发球次数将无限增加。如果选择定期入口策略，则为必填项。
CloudWatch 警报名称	*<Requires input>*	与定期入口策略关联的可选 CloudWatch 警报名称。如果提供且处于警报状态，则服务数量不会增加。仅适用于定期入口策略。
最大尺寸	*<Requires input>*	下游目标站点一次可以处理的最大交易数量（MaxSize 策略）。

6. 请选择 Next（下一步）。
7. 在 配置堆栈选项 页面上，请选择 下一步。
8. 在 Review 页面上，审核并确认设置。选中确认模板创建 AWS Identity and Access Management (IAM) 资源的复选框。
9. 选择 Create stack（创建堆栈）以部署堆栈。

   您可以在 AWS CloudFormation 控制台的 “状态” 列中查看堆栈的状态。大约两分钟后，您应该会收到 CREATE _ COMPLETE 状态。

5. （可选）启动样本等候室堆栈

部署用时：大约五分钟

1. 登录AWS Management Console并选择按钮以启动`aws-virtual-waiting-room-sample.template` AWS CloudFormation 模板。

   Launch solution

   或者，您可以下载模板作为自己实现的起点。
2. 默认情况下，该模板在美国东部（弗吉尼亚州北部）区域启动。要在其他 AWS 区域启动解决方案，请使用控制台导航栏中的区域选择器。
3. 在创建堆栈页面上，确认 Amazon S3 URL 文本框中的模板URL是否正确，然后选择下一步。
4. 在指定堆栈详细信息页面上，为您的解决方案堆栈分配一个名称。有关命名字符限制的信息，请参阅《AWS Identity and Access Management 用户指南》中的IAM和STS限制。
5. 在“参数”下，查看此解决方案模板的参数并根据需要对其进行修改。该解决方案使用以下默认值。

参数	默认值	描述
API网关区域	*<Requires input>*	AWS API网关的区域名称。
授权者 ARN	*<Requires input>*	ARNAPI网关 Lambda 授权者的。
活动编号	`Sample`	等候室的事件 ID。
私有API端点	*<Requires input>*	虚拟等候室URL的私有端点APIs。
公共API端点	*<Requires input>*	虚拟等候室URL的公共端点APIs。

6. 请选择 Next（下一步）。
7. 在 配置堆栈选项 页面上，请选择 下一步。
8. 在 Review 页面上，审核并确认设置。选中确认模板创建 AWS Identity and Access Management (IAM) 资源的复选框。
9. 选择 Create stack（创建堆栈）以部署堆栈。

您可以在 AWS CloudFormation 控制台的 “状态” 列中查看堆栈的状态。大约五分钟后，您应该会收到 CREATE _ COMPLETE 状态。

从先前版本更新堆栈

我们建议删除堆栈并为新版本创建一个新堆栈。当前，不支持使用 CloudFormation 堆栈更新迁移到较新版本。卸载此解决方案然后参见 “启动入门堆栈”。

> **Note**
>
> 如果您不积极使用该解决方案来支持正在进行的活动，我们建议您迁移到较新的版本。

性能数据

虚拟等候室 AWS 已使用名为 Locust 的工具进行了负载测试。模拟事件的大小从 10,000 到 100,000 个客户端不等。负载测试环境由以下配置组成：

- Locust 2.x 具有针对云部署的自定义设置 AWS
- 四个 AWS 区域 (us-west-1、us-west-2、us-east-1、us-east-2)
- 每个区域 10 c5.4xlarge 台 Amazon EC2 主机（共 40 台）
- 每台主机 32 个蝗虫进程
- 模拟用户在 1,280 个进程中均匀分布

每个用户进程的 end-to-end API 测试步骤：

1. 致电assign_queue_num并获取请求编号。
2. 使用请求 ID queue_num 进行循环，直到它返回用户的队列位置（短时间）。
3. 循环serving_num直到返回值大于等于用户的队列位置（很长时间）。
4. 很少打电话waiting_room_size来检索等待的用户数量。
5. 致电generate_token并接收 JWT 以在目标站点中使用。

调查发现

通过等候室可以处理的客户数量没有实际的上限。

用户进入等候室的速度会影响 Lambda 函数部署区域的 Lambda 函数并发运行配额。

负载测试无法超过默认的 API Gateway 请求限制，即每秒 10,000 个请求，且使用了缓存策略 CloudFront。

get_queue_numLambda 函数的调用率与进入等候室的用户速率接近 1:1。由于并发限制或突发限制，此 Lambda 函数可能会在用户访问频率较高时受到限制。由大量 Lambda 函数调用引起的限制可能会作为get_queue_num副作用影响其他 Lambda 函数。如果客户端软件能够使用重试/退缩逻辑对此类临时缩放错误做出适当的响应，则整个系统将继续运行。

由核心堆栈在默认配额配置中配置的 CloudFront 分发可以处理容纳 250,000 个用户的等候室，每个用户至少每秒轮询一次 serving_num API。

故障排除

本节提供此解决方案的故障排除信息。

如果本节无法解决您的问题，请联系 AWS Support，我们将为您提供有关如何为该解决方案提出 AWS Support 案例的说明。

来自 API 的 4xx 响应状态

- 这可能是由于事件 ID 或请求 ID 不正确或两者兼而有之。这发生在相关 Lambda 函数的 CloudWatch 日志中。
- 私有 API 已通过 IAM 身份验证，客户端需要有权调用私有 API 的 AWS 密钥。这发生在 API Gateway 的 CloudWatch 日志中。

来自 API 的 5xx 响应状态

- 来自受限制的 Lambda 或 API Gateway 的响应，请查看警报。*<LambdaFunctionName>*ThrottlesAlarm CloudWatch
- 后端配置错误，请查看*<LambdaFunctionName>*ErrorsAlarm CloudWatch 警报和 CloudWatch 日志以了解详细信息。

5XX/ErrorPublicPrivateApiAlarm

- 此警报ALARM状态是 API 在 60 秒内向调用方返回 5XX 状态。
- 此警报在 60 秒内未返回 5xx 状态OK时返回。
- 此警报可以通过 Lambda 函数或 Lambda 运行时向 API Gateway 返回错误来启动。

4XX/ErrorPublicPrivateApiAlarm

- 此警报ALARM状态是 API 在 60 秒内向调用方返回 4XX 状态。
- 此警报会返回OK到 4XX 状态，持续 60 秒。
- 此警报可能由不正确的 API 网址发起。

*<LambdaFunctionName>*ThrottlesAlarm

- 当命名的 Lambda 在 60 秒内遇到并发运行限制时，此警报状态为警报。

- OK如果 60 秒内未遇到任何限制，则此警报将返回。
- 您可能需要提高账户所在地区的并发限制。
- 您可能会遇到 Lambda 的突发限制，这需要在您的客户端上执行一些重试逻辑。

*<LambdaFunctionName>*ErrorsAlarm

- 此警报状态是指ALARM命名的 Lambda 在 60 秒内遇到运行时运行错误。
- OK如果 60 秒内未遇到任何错误，则此警报将返回。
- 这可能是由后端配置错误引起的。
- 这可能是由 Lambda 代码中的错误引起的。

联系我们 AWS Support

如果您有 AWS 开发者支持、AWS 商业支持或 AWS 企业支持，则可以使用支持中心获取有关此解决方案的专家帮助。以下部分提供了说明。

创建案例

1. 登录 Su pport Center。
2. 选择创建案例。

我们能提供什么帮助？

1. 选择 “技术”。
2. 对于 “服务”，选择 “解决方案”。
3. 在 “类别” 中，选择 “其他解决方案”。
4. 在 “严重性” 中，选择与您的用例最匹配的选项。
5. 当您输入 “服务”、“类别” 和 “严重性” 时，界面会填充常见疑难解答问题的链接。如果您无法通过这些链接解决问题，请选择下一步：其他信息。

其他信息

1. 在 “主题” 中，输入总结您的问题或问题的文本。
2. 在描述中，详细描述问题。

3. 选择 “附加文件”。
4. 附上处理请求 AWS Support 所需的信息。

帮助我们更快地解决您的问题

1. 输入所需的信息。
2. 选择下一步：立即解决或联系我们。

立即解决或联系我们

1. 查看 “立即解决” 解决方案。
2. 如果您无法使用这些解决方案解决问题，请选择 “联系我们”，输入所需信息，然后选择 “提交”。

其他资源

AWS 服务	
• AWS CloudFormation	• Amazon DynamoDB
• Amazon Simple Storage Service	• 亚马逊API网关
• AWS Lambda	• AWS Secrets Manager
• 亚马逊 CloudFront	• Amazon Simple Queue Service
• 亚马逊 EventBridge	• 亚马逊 CloudWatch
• Elasticache (Redis) OSS	• Amazon Comprehend
• Amazon Virtual Private Cloud	• AWS Identity and Access Management

卸载此解决方案

你可以使用 AWS Management Console 或卸载 AWS 解决方案上的虚拟等候室 AWS Command Line Interface。您必须手动删除此解决方案创建的各种资源用于存储日志的 S3 存储桶。 AWS 解决方案实施不会自动删除这些 S3 存储桶，因此在删除解决方案后，您仍然可以查看日志事件。

如果您已手动将 IAM 用户添加到解决方案创建`ProtectedAPIGroup`的 IAM 用户组，请在卸载解决方案之前将该 IAM 用户从 IAM 用户组中移除。否则，IAM 用户组和关联的 IAM 策略将无法删除。

对于部署的每个堆栈，请按照以下说明进行操作。

使用 AWS Management Console

1. 登录 AWS CloudFormation 控制台。
2. 在堆栈页面上，选择此解决方案的安装堆栈。
3. 选择删除。

使用 AWS Command Line Interface

确定 AWS Command Line Interface (AWS CLI) 在您的环境中是否可用。有关安装说明，请参阅什么是 AWS Command Line Interface？ 在《AWS CLI 用户指南》中。确认可用后，运行以下命令。 AWS CLI

```
$ aws cloudformation delete-stack --stack-name <installation-stack-name>
```

删除 Amazon S3 存储桶

如果您决定删除 AWS CloudFormation 堆栈以防止意外丢失数据，则此解决方案配置为保留解决方案创建的 Amazon S3 存储桶（用于在可选区域进行部署）。卸载解决方案后，如果您不需要保留数据，则可以手动删除此 S3 存储桶。按照以下步骤删除 Amazon S3 存储桶。

1. 登录 Amazon S3 控制台。
2. 在左侧导航窗格中，选择桶。
3. 找到 *<stack-name>* S3 桶。
4. 选择 S3 存储桶，然后选择删除。

要使用删除 S3 存储桶 AWS CLI，请运行以下命令：

```
$ aws s3 rb s3://<bucket-name> --force
```

源代码

访问我们的GitHub存储库，下载此解决方案的源文件并与其他人共享您的自定义设置。

贡献者

- 吉姆·塔里奥
- Thyag Ramachandran
- 琼·摩根
- 贾斯汀·皮特尔
- Allen Moheimani
- 加维特·辛格
- Bassem Wanis

修订

Date	更改
2021 年 11 月	初始版本
2022 年 9 月	版本 1.1：根据过期队列位置自动增加服务计数器增量。将一些 Elasticache (Redis) OSS 用法迁移到 DynamoDB。用于获取剩余队列位置到期时间的公共API端点。有关更多信息，请参阅 GitHub 存储库中的 CHANGELOG.md 文件。
2023 年 4 月	版本 1.1.1：缓解了所有新 S3 存储桶的 S3 对象所有权（ACLs已禁用）的新默认设置所造成的影响。有关更多信息，请参阅 GitHub 存储库中的 CHANGELOG.md 文件。
2023 年 11 月	版本 1.1.2：更新了软件包版本以解决安全漏洞。有关更多信息，请参阅 GitHub 存储库中的 CHANGELOG.md 文件。
2024 年 3 月	版本 1.1.3：解决了三个问题：过期的队列位置在等候室大小中仍然存在，即使在重置后仍`queue_num` API返回旧结果，以及 OpenID 适配器出现间歇性故障。`/userInfo` API 有关更多信息，请参阅 GitHub 存储库中的 CHANGELOG.md 文件。
2024 年 4 月	版本 1.1.4：更新了软件包版本以解决安全漏洞。有关更多信息，请参阅 GitHub 存储库中的 CHANGELOG.md 文件。
2024 年 6 月	版本 1.1.5：更新了软件包版本以解决安全漏洞。有关更多信息，请参阅 GitHub 存储库中的 CHANGELOG.md 文件。

Date	更改
2024 年 8 月	版本 1.1.6：更新了软件包版本以解决安全漏洞。有关更多信息，请参阅 GitHub 存储库中的 CHANGELOG.md 文件。
2024 年 8 月	版本 1.1.7：更新了软件包版本以解决安全漏洞。有关更多信息，请参阅 GitHub 存储库中的 CHANGELOG.md 文件。
2024 年 9 月	版本 1.1.8：更新了软件包版本以解决安全漏洞。有关更多信息，请参阅 GitHub 存储库中的 CHANGELOG.md 文件。

版权声明

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。