구현 안내서

AWS 기반 가상 대기실

AWS 기반 가상 대기실: 구현 안내서

Table of Contents

가상 대기실을 켜면 웹사이트로 몰리는 엄청난 트래픽을 흡수할 수 있습니다. AWS

발행일: 2021년 11월 (최종 업데이트: 2024년 6월)

가상 대기실 온 AWS 솔루션은 트래픽이 폭주하는 동안 웹 사이트로 들어오는 사용자 요청을 제어하는 데 도움이 됩니다. 웹 사이트로 들어오는 트래픽을 일시적으로 오프로드하도록 설계된 클라우드 인프라를 생성하고 가상 대기실을 사용자 지정 및 통합할 수 있는 옵션을 제공합니다. 이 솔루션을 신규 또는 기존 웹 사이트와 통합하여 갑자기 급증하는 트래픽을 처리할 수 있도록 원활하게 확장할 수 있습니다.

웹사이트 트래픽을 급증시킬 수 있는 대규모 이벤트의 예는 다음과 같습니다.

- 콘서트 또는 스포츠 이벤트 티켓 판매 시작
- 파이어 세일 또는 기타 대규모 소매 판매 (예: 블랙 프라이데이)
- 광범위한 마케팅 발표를 포함한 신제품 출시
- 온라인 테스트 및 수업을 위한 시험 액세스 및 수업 참석
- 진료 예약 슬롯 출시
- 계정 생성 및 결제가 필요한 새 direct-to-customer 서비스 출시

이 솔루션은 웹사이트 방문자를 위한 저장 공간 역할을 하며 용량이 충분할 때 트래픽이 통과할 수 있도록 합니다. 방문자가 사용하는 클라이언트 소프트웨어는 웹 사이트가 최대 용량이 될 때까지 대기실을 통한 트래픽을 투명하게 허용하도록 구성할 수 있습니다. 이때 대기실은 방문자를 차단합니다. 웹 사이트에 더 많은 트래픽을 수용할 수 있는 용량이 확보되면 솔루션은 사용자가 웹 사이트에 액세스할 수 있도록 하는 JSON 웹 토큰 (JWT) 을 생성합니다. 예를 들어 이벤트가 2시간 동안 지속되고 웹 사이트가 초당 50명의 사용자를 처리할 수 있지만 초당 250명의 사용자를 처리할 것으로 예상한다면 이 솔루션을 사용하여 트래픽을 규제하는 동시에 사용자가 대기열에서 자신의 위치를 유지할 수 있도록 할 수 있습니다.

이 솔루션은 다음과 같은 주요 기능을 제공합니다.

- 웹 사이트에 사용자를 구조적으로 대기시킵니다.
- 대규모 이벤트의 트래픽을 제어할 수 있는 확장성
- 대상 사이트에 들어갈 수 있는 JSON 웹 토큰 생성
- 모든 기능은 REST API를 통해 제어됩니다.

- 클라이언트 솔루션을 위한 턴키 API Gateway 권한 부여자
- 독립형 통합 또는 OpenID와 함께 사용

이 구현 안내서는 Amazon Web Services (AWS) 클라우드에 가상 대기실을 배포하기 위한 아키텍처 고려 사항 및 구성 단계를 설명합니다. AWS 여기에는 보안 및 가용성 AWS 모범 사례를 사용하여 이 솔루션을 배포하는 데 필요한 AWS 서비스를 시작하고 구성하는 AWS CloudFormation템플릿에 대한 링크가 포함되어 있습니다.

이 가이드는 클라우드 아키텍처 설계 실무 경험이 있는 IT 설계자, 개발자, DevOps 직원, 데이터 분석가 및 마케팅 기술 전문가를 대상으로 합니다. AWS

비용

이 솔루션을 실행하는 동안 사용되는 AWS 서비스 비용은 사용자 부담입니다. 이 수정 버전을 기준으로 미국 동부 (버지니아 북부) 지역에서 기본 설정으로 이 솔루션을 실행하는 데 드는 비용은 스택당 하루 약 10.00 USD이며, 여기에 이벤트 규모에 따른 API 요청 및 데이터 트래픽 요금이 추가됩니다.

이벤트 없이 솔루션을 유지 관리하는 데 드는 일일 비용

AWS service	요청/시간	비용 [미국 달러]
Amazon API Gateway	0	0.00 달러
아마존 CloudFront	0	0.00 달러
아마존 CloudWatch	0	0.00 달러
Amazon DynamoDB	0	0.00 달러
아마존 ElastiCache	컴퓨팅 노드 시간 (Redis)	~6.00달러
AWS Lambda	프리 티어*	0.00 달러
AWS Secrets Manager	프리 티어*	0.00 달러
Amazon Simple Storage Service(S3)	프리 티어*	0.00 달러
Amazon Virtual Private Cloud(VPC)	VPC 엔드포인트 시간 NAT 게이트웨이 시간	약 5.00 달러
총액:		~11.00 달러

*예상 비용은 깨끗한 환경을 기준으로 합니다. 이 솔루션 외부에서 이 AWS 서비스를 사용하는 경우 프리 티어 할당량을 초과할 수 있습니다.

다음 표는 50,000명의 사용자와 100,000명의 사용자 대기실에 대한 예상 비용을 보여줍니다. 이벤트 지속 시간은 2~4시간 (초당 500명의 사용자 수신, 1,000명/분 발신 사용자) 입니다. 요금은 변경될 수 있습니다. 자세한 내용은 이 솔루션에서 사용되는 각 서비스의 요금 웹 페이지를 참조하십시오. AWS

2시간 이벤트 기간 동안 대기실 사용자 50,000명에 대한 예상 비용

AWS service	측정기준	비용 [미국 달러]
Amazon API Gateway	요청	2.00 달러
CloudFront	요청, 대역폭	75.00 달러
CloudWatch	지표, 알람, 스토리지	1.00 달러
아마존 CloudWatch 이벤트	이벤트	1.00 달러
DynamoDB	읽기/쓰기 유닛, 스토리지	1.00 달러
ElastiCache	노드 시간	8.00 달러
Lambda	요청, 컴퓨팅 타임	1.00 달러
AWS Secrets Manager	비밀, 요청	1.00 달러
Amazon S3	요청, 스토리지	1.00 달러
Amazon VPC	데이터 전송, 엔드포인트 시간	2.00 달러
총액		94.00 달러

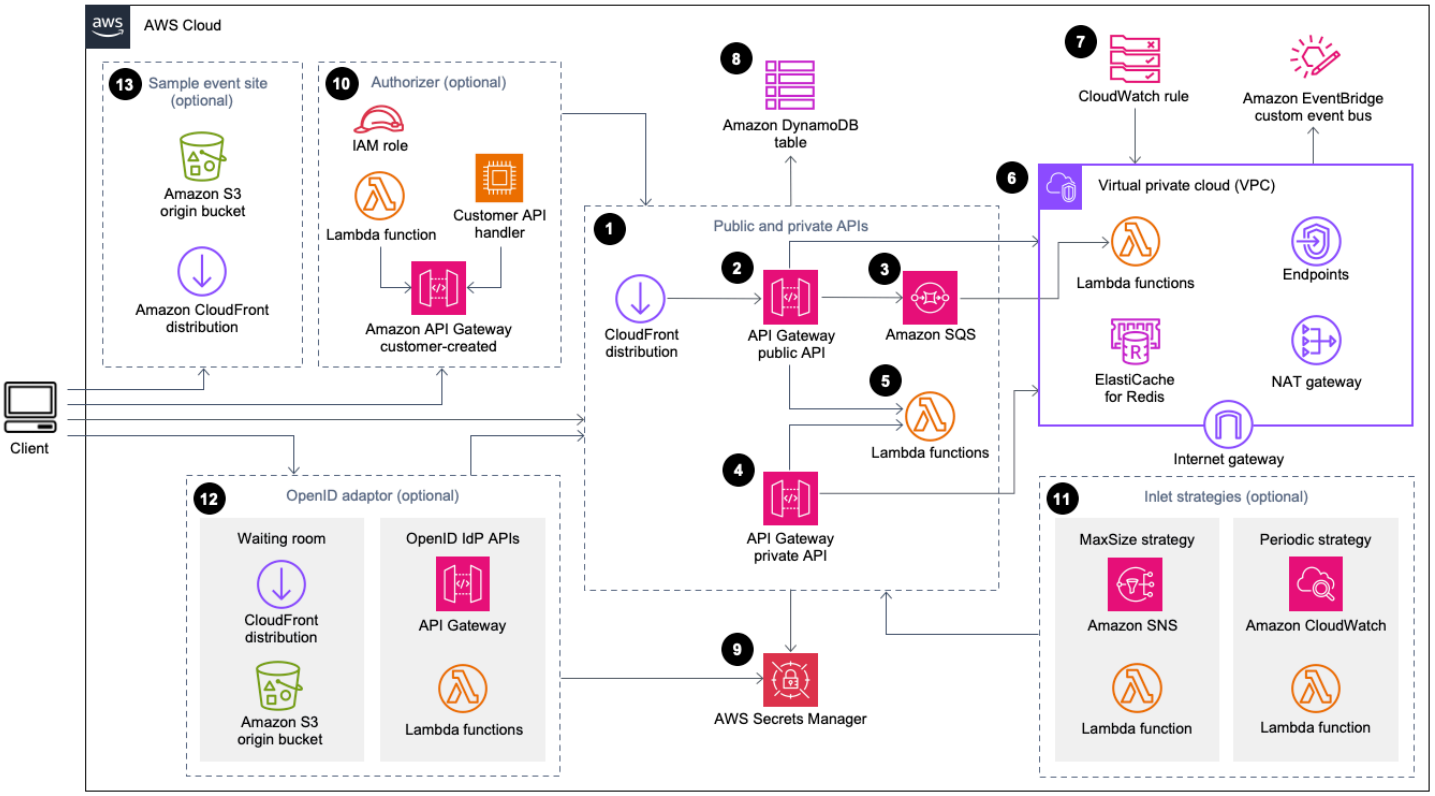
2시간 이벤트 기간 동안 대기실 사용자 10만 명에 대한 예상 비용

AWS service	측정기준	비용 [미국 달러]
Amazon API Gateway	요청	4.00 달러
CloudFront	요청, 대역폭	296.00 달러
CloudWatch	지표, 알람, 스토리지	1.00 달러
CloudWatch 이벤트	이벤트	1.00 달러
DynamoDB	읽기/쓰기 유닛, 스토리지	4.00달러

ElastiCache	노드 시간	32.00 달러
Lambda	요청, 컴퓨팅 타임	1.00 달러
AWS Secrets Manager	비밀, 요청	1.00 달러
Amazon Simple Queue Service(Amazon SQS)	요청	1.00 달러
Amazon S3	요청, 스토리지	1.00 달러
Amazon VPC	데이터 전송, 엔드포인트 시간	6.00 달러
총액		348.00 달러

아키텍처 개요

기본 매개변수를 사용하여 필수 및 선택적 템플릿과 함께 이 솔루션을 배포하면 클라우드에 다음과 같은 환경이 구축됩니다 AWS .



AWS 아키텍처에 따른 가상 대기실

AWS CloudFormation 템플릿은 다음 인프라를 배포합니다.

1. 클라이언트에 공개 API 호출을 전달하기 위한 Amazon CloudFront 배포입니다.
2. Amazon API Gateway 퍼블릭 API 리소스는 가상 대기실의 대기열 요청을 처리하고, 대기열 위치를 추적하고, 대상 웹 사이트에 대한 액세스를 허용하는 토큰의 검증을 지원합니다.
3. Amazon 심플 큐 서비스 (Amazon SQS) 대기열은 대기열 메시지를 처리하는 AWS Lambda함수에 대한 트래픽을 규제합니다. 각 요청에 대해 Lambda 함수를 호출하는 대신, SQS 대기열은 들어오는 요청 버스트를 일괄 처리합니다.
4. API Gateway 프라이빗 API 리소스는 관리 기능을 지원합니다.
5. Lambda 함수는 퍼블릭 및 프라이빗 API 요청을 검증 및 처리하고 적절한 응답을 반환합니다.
6. Amazon VPC (가상 사설 클라우드) 는 Amazon Redis용 클러스터와 직접 상호 작용하는 Lambda 함수를 호스팅합니다. ElastiCache VPC 엔드포인트를 사용하면 VPC의 Lambda 함수가 솔루션 내

의 서비스와 통신할 수 있습니다. 또한 NAT 게이트웨이를 사용하면 VPC의 Lambda 함수가 엔드포인트를 CloudFront 연결하고 필요에 따라 캐시를 무효화할 수 있습니다.

7. 사용자 지정 Amazon 버스와 함께 작동하여 상태 업데이트를 주기적으로 브로드캐스트하는 Lambda 함수를 호출하는 EventBridgeAmazon CloudWatch 규칙입니다.
8. 토큰, 대기열 위치 및 서빙 카운터 데이터를 저장하는 Amazon DynamoDB 테이블.
9. AWS Secrets Manager는 토큰 작업을 위한 키와 기타 민감한 데이터를 저장합니다.
10. (선택 사항) API Gateway와 함께 사용하기 위한 AWS Identity and Access Management(IAM) 역할 및 Lambda 권한 부여자 함수로 구성된 권한 부여자 구성 요소.
11. (선택 사항) 아마존 심플 알림 서비스 (Amazon SNS) 및 Lambda 함수는 두 가지 유입 전략을 지원합니다. CloudWatch
12. (선택 사항) OpenID 공급자가 웹 사이트에 대한 사용자를 인증할 수 있도록 해주는 API Gateway 및 Lambda 함수를 포함하는 OpenID 어댑터 구성 요소. CloudFront Amazon 심플 스토리지 서비스 (Amazon S3) 버킷을 사용하여 이 구성 요소의 대기실 페이지를 배포합니다.
13. (선택 사항) 샘플 대기실 웹 애플리케이션용 Amazon S3 오리진 버킷을 사용한 CloudFront 배포.

솔루션 작동 방식

이 섹션에서는 AWS 가상 대기실 워크플로의 단계를 개괄적으로 설명합니다. 웹 사이트의 대기실 구축, 사용자 지정 및 통합에 GitHub 대한 자세한 내용은 개발자 안내서를 참조하십시오.

대기실의 공개 API는 사이트 경계 보안 뒤에 위치하거나 승인 없이 사용할 수 있습니다. 대기실을 웹 사이트와 통합하는 데 사용하는 접근 방식에 따라 사용자는 먼저 웹 사이트 인증을 거쳐야 대기실로 이동하여 대기열에 들어갈 수 있습니다.

대기실에 들어가서 다른 요청을 하려면 클라이언트 소프트웨어에 이벤트 ID가 있어야 합니다. 이벤트 ID는 공개 및 비공개 API에 대한 대부분의 요청에 필요한 고유 ID입니다. 이벤트 ID는 코어 API 스택 설치 중에 설정됩니다. 작동 중에 이벤트 ID는 대기실 페이지를 통해 URL 매개 변수 또는 쿠키로 제공될 수 있습니다. 이벤트 ID는 인증 토큰 클레임의 일부로 제공되거나 다른 데이터 경로를 통해 클라이언트에 배포될 수 있습니다.

클라이언트가 특정 API를 호출하기 위해 이벤트 ID와 요청 ID가 모두 필요한 경우가 있습니다. 요청 ID는 대기실에서 발급된 고유 ID로, 줄을 선 특정 클라이언트를 나타냅니다.

다음 단계는 대기열 입력, 대기열 진행 대기, 웹 사이트에 대한 액세스 토큰을 사용하여 대기실 나가기를 위한 API 요청의 흐름을 설명합니다.

사용자가 대기실에 입장합니다.

1. 대기실 진입점을 나타내는 화면 또는 페이지가 사용자에게 표시됩니다. 대기열에 들어가기로 선택하면 클라이언트 소프트웨어 (브라우저, 모바일, 장치) 가 `assign_queue_num` 공개 API를 호출하여 대기열 위치를 요청합니다.
2. API 요청은 API Gateway를 통해 Amazon SQS 대기열로 즉시 전송됩니다.
3. 요청이 대기열에 배치되면 `assign_queue_num` API 호출이 반환됩니다. 클라이언트는 나중에 대기열 위치, 요청 시간 및 액세스 토큰을 검색하는 데 사용할 수 있는 고유한 요청 ID를 받습니다.
4. `AssignQueueNumLambda` 함수는 SQS 대기열에서 최대 10개의 요청 배치를 수신합니다. Lambda 서비스는 호출을 팬아웃하여 여러 배치 요청을 처리합니다.
5. `AssignQueueNumLambda` 함수는 배치에서 각 메시지를 검증하고, Redis의 대기열 카운터를 ElastiCache 늘리고, 각 요청을 관련 대기열 위치와 함께 ElastiCache Redis에 저장합니다.
6. 각 메시지는 성공적으로 처리되면 삭제됩니다. 오류 상황과 관련된 메시지는 이후 일괄 처리에서 한 번 재처리됩니다. 두 번째 실패 후에는 dead-letter-queue 연결된 CloudWatch경보로 전송됩니다.
7. 클라이언트는 `assign_queue_num` 호출로부터 요청 ID를 받은 후 `queue_num` API 폴링을 시작할 수 있습니다. 클라이언트는 이벤트 ID와 요청 ID를 `queue_num` API로 전송하고 대기열 위치 또는

요청이 아직 처리되지 않았음을 나타내는 응답을 받습니다. 대규모 이벤트 중에는 클라이언트가 이 호출을 두 번 이상 해야 할 수도 있습니다. GetQueueNumLambda 함수는 API Gateway에 의해 호출되며 DynamoDB로부터 대기열 내 클라이언트의 숫자 위치를 반환합니다.

사용자가 대기실에서 대기합니다.

8. 클라이언트가 대기열에 들어가면 일정한 간격으로 serving_num API 폴링을 시작할 수 있습니다. serving_numAPI는 이벤트 ID와 함께 호출되며 대기열의 현재 제공 위치를 반환합니다. serving_numAPI의 응답은 클라이언트가 대기실에서 최종 트랜잭션이 발생할 수 있는 실제 대상 사이트로 이동할 수 있는 시기를 클라이언트에게 알려줍니다. GetServingNumLambda 함수는 대기실의 현재 서비스 위치를 반환합니다.
9. 서비스 위치가 클라이언트의 대기열 (요청) 위치와 같거나 크면 클라이언트는 퍼블릭 API에서 JSON 웹 토큰 (JWT) 을 요청할 수 있습니다. 토큰은 대상 사이트에서 트랜잭션을 마무리하는 데 사용할 수 있습니다. generate_tokenAPI는 이벤트 ID 및 요청 ID와 함께 호출됩니다. API Gateway는 파라미터와 함께 GenerateToken Lambda 함수를 호출합니다.
10. GenerateTokenLambda 함수는 요청을 검증하고 이 토큰이 이전에 생성되었는지 확인합니다. Lambda 함수는 DynamoDB 테이블에서 일치하는 토큰을 쿼리합니다. 토큰이 발견되면 해당 토큰은 호출자에게 반환되며 재생성되지 않습니다. 이 프로세스는 단일 요청 ID를 사용하여 만료 시간이 새로운 여러 개의 서로 다른 토큰을 생성하는 것을 방지합니다.
11. DynamoDB에서 토큰을 찾을 수 없는 경우, Lambda 함수는 키를 검색하여 토큰을 생성하고 이벤트 ID 및 클라이언트의 요청 ID와 함께 DynamoDB에 토큰을 저장합니다. Lambda 함수는 새 토큰이 생성되었음을 알리는 이벤트를 EventBridge 기록합니다. Lambda 함수는 이벤트에 대해 생성된 토큰 수를 추적하는 for Redis 카운터를 증가시킵니다 ElastiCache .
12. queue_pos_expiry가 켜져 있는 경우 클라이언트는 Lambda GetQueuePositionExpiryTime 함수를 호출하는 queue_pos_expiry API를 호출하여 만료 전 남은 시간을 쿼리할 수 있습니다.

사용자가 대기실을 떠납니다.

13. 클라이언트는 토큰을 받으면 대상 사이트에 들어가 트랜잭션을 시작합니다. 인프라에서 JWT와의 통합을 지원하는 방식에 따라 클라이언트는 요청 헤더, 쿠키 또는 다른 방법으로 토큰을 제시해야 할 수 있습니다. API Gateway의 권한 부여자를 사용하여 클라이언트 요청에 포함된 토큰을 검증할 수 있습니다. JWT의 검증 및 관리를 위한 모든 상용 또는 오픈 소스 라이브러리를 토큰의 가상 대기실과 함께 사용할 수 있습니다. AWS 토큰이 유효하면 클라이언트는 거래를 계속할 수 있습니다.
14. 클라이언트가 트랜잭션을 완료하면 프라이빗 API가 호출되어 클라이언트 토큰 상태를 업데이트하고 DynamoDB에서 완료됩니다.

대기열 위치 만료:

15.이 기능이 활성화되면 특정 대기열 위치에 해당하는 요청 ID는 지정된 시간 간격 동안만 토큰을 생성할 수 있습니다.

대기열 위치 만료 시 서빙 카운터 증분:

16.이 기능을 활성화하면 토큰을 생성할 수 없었던 만료된 대기열 위치를 기준으로 서빙 카운터가 자동으로 증가합니다.

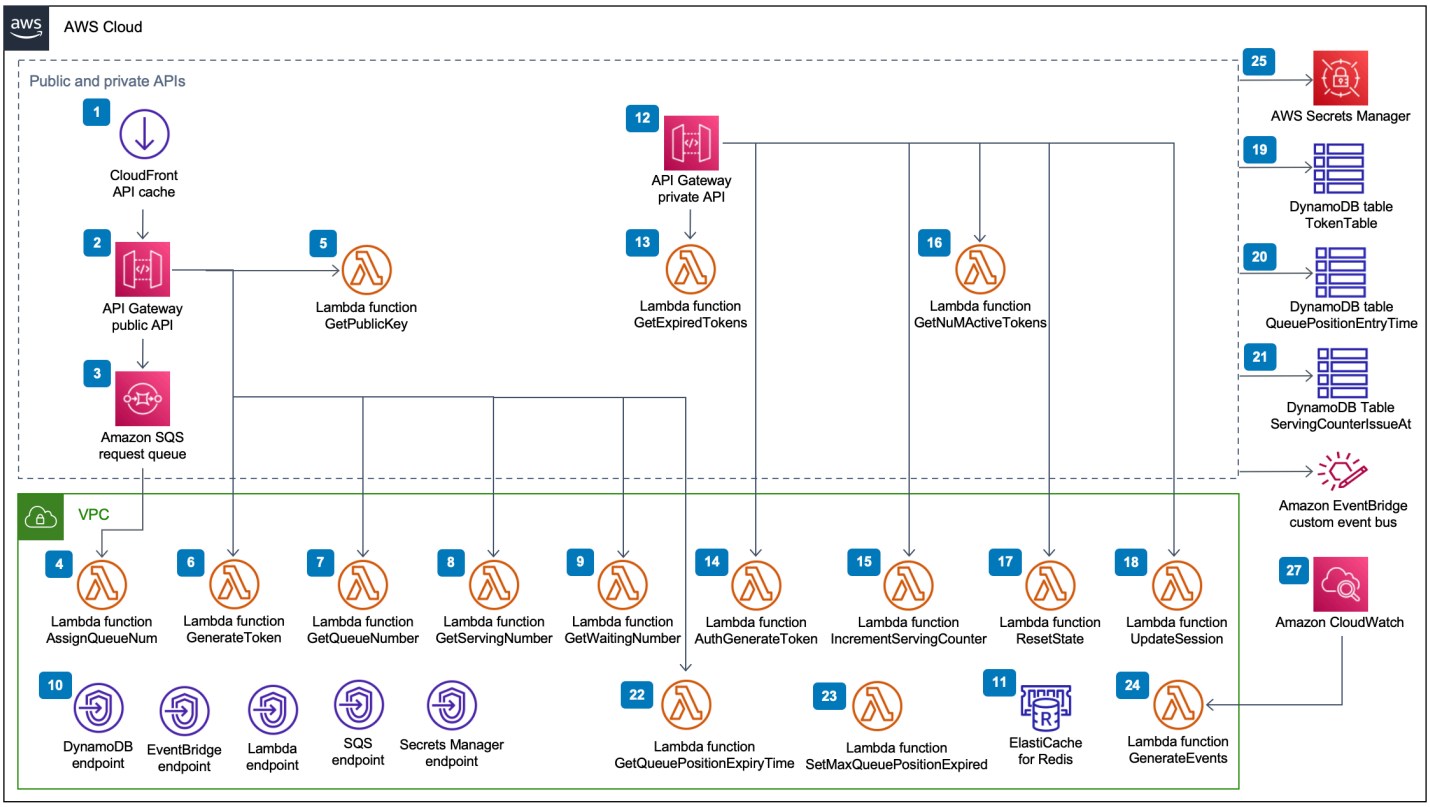
솔루션 구성 요소

대기실 공개 및 비공개 API

Virtual Waiting Room on AWS 솔루션의 주요 목적은 대상 웹 사이트에 과부하가 걸릴 수 있는 신규 사용자 폭주를 방지하기 위해 통제된 방식으로 클라이언트를 위한 JSON 웹 토큰 (JWT) 생성을 제어하는 것입니다. JWT는 사이트 보호, 대기실 토큰을 받을 때까지 웹 페이지에 대한 액세스를 차단하고 API 액세스 권한 부여에도 사용할 수 있습니다.

코어 템플릿은 대부분의 가상 대기실 운영에 사용되는 공개 API 및 사설 (IAM 인증) API를 설치합니다. AWS 퍼블릭 API는 API 경로를 기반으로 하는 여러 캐싱 정책이 포함된 CloudFront 배포로 구성됩니다. DynamoDB EventBridge 테이블과 이벤트 버스가 생성됩니다. 템플릿은 두 개의 가용 영역 (AZ) 이 있는 새 VPC, 두 AZ 모두에 있는 ElastiCache Redis 클러스터 및 여러 Lambda 함수를 추가합니다. for ElastiCache Redis와 상호 작용하는 Lambda 함수는 VPC 내에 네트워크 인터페이스를 가지며 다른 모든 Lambda 함수는 기본 네트워크 연결을 가집니다. 핵심 API는 솔루션과의 상호 작용이 가장 낮은 계층입니다. 다른 Lambda 함수, Amazon Elastic Compute Cloud (Amazon EC2) 인스턴스 및 컨테이너는 확장 기능으로 작동하고 코어 API를 호출하여 대기실을 구축하고, 유입 트래픽을 제어하고, 솔루션에서 생성된 이벤트에 대응할 수 있습니다.

또한 코어 스택은 모든 Lambda 함수 오류 및 스로틀 조건에 대한 경보뿐만 아니라 4XX 및 5XX 상태 코드에 대한 각 API Gateway 배포에 대한 경보를 생성합니다.



AWS 퍼블릭 및 프라이빗 API 구성 요소의 가상 대기실

1. CloudFront 배포는 클라이언트에 대한 퍼블릭 API 호출을 제공하고 적절한 경우 결과를 캐싱합니다.
2. Amazon API Gateway 퍼블릭 API는 가상 대기실의 대기열 요청을 처리하고, 대기열 위치를 추적하고, 대상 웹 사이트에 대한 액세스를 허용하는 토큰의 검증을 지원합니다.
3. SQS 대기열은 대기열 메시지를 처리하는 AWS Lambda 함수에 대한 트래픽을 규제합니다.
4. `AssignQueueNumLambda` 함수는 수신된 일괄 처리의 각 메시지를 검증하고, Redis의 대기열 카운터를 ElastiCache 늘리고, 각 요청을 관련 대기열 위치와 함께 ElastiCache Redis에 저장합니다.
5. `GetPublicKeyLambda` 함수는 Secrets Manager에서 퍼블릭 키 값을 검색합니다.
6. `GenerateTokenLambda` 함수는 대상 사이트에서 트랜잭션을 완료하도록 허용된 유효한 요청에 대해 JWT를 생성합니다. 토큰이 생성되었다는 이벤트를 대기실의 사용자 지정 이벤트 버스에 기록합니다. 이 요청에 대해 이전에 토큰을 생성한 경우 새 토큰은 생성되지 않습니다.
7. `GetQueueNumberLambda` 함수는 Redis용 대기열에서 클라이언트의 숫자 위치를 검색하고 반환합니다. ElastiCache
8. `GetServingNumberLambda` 함수는 Redis의 대기실에서 현재 서비스 중인 번호를 검색하여 반환합니다. ElastiCache

9. GetWaitingNumLambda 함수는 현재 대기실에 대기하고 있지만 아직 토큰이 발급되지 않은 번호를 반환합니다.
10. VPC 엔드포인트를 사용하면 VPC의 Lambda 함수가 솔루션 내의 서비스와 통신할 수 있습니다.
11. ElastiCache Redis의 경우 클러스터는 대기실에 들어가기 위한 모든 요청을 유효한 이벤트 ID와 함께 저장합니다. 또한 대기열에 있는 요청 수, 현재 제공 중인 요청 수, 생성된 토큰 수, 완료된 세션 수, 중단된 세션 수와 같은 여러 카운터를 저장합니다.
12. API Gateway 프라이빗 API 리소스는 관리 기능을 지원합니다. 프라이빗 API는 AWS IAM 인증을 받았습니다.
13. GetExpiredTokensLambda 함수는 만료된 토큰이 있는 요청 ID 목록을 반환합니다.
14. AuthGenerateTokenLambda 함수는 대상 사이트에서 트랜잭션을 완료하도록 허용된 유효한 요청에 대한 토큰을 생성합니다. 코어 스택 배포 중에 처음 설정된 토큰의 발급자 및 유효 기간은 재정의될 수 있습니다. 토큰이 생성되었다는 이벤트를 대기실의 커스텀 이벤트 버스에 기록합니다. 이 요청에 대한 토큰이 이전에 생성된 경우 새 토큰은 생성되지 않습니다.
15. IncrementServingCounterLambda 함수는 Redis에 저장된 ElastiCache 대기실의 서빙 카운터를 값만큼 증가시켜 줍니다.
16. GetNumActiveTokensLambda 함수는 아직 만료되지 않았고, 트랜잭션을 완료하는 데 사용되지 않았으며, 중단된 것으로 표시되지 않은 토큰 수를 DynamoDB에 쿼리합니다.
17. ResetStateLambda 함수는 Redis에 저장된 모든 카운터를 재설정합니다. ElastiCache 또한TokenTable,QueuePositionEntryTime, DynamoDB ServingCounterIssuedAt 테이블을 삭제하고 다시 생성합니다. 또한 캐시 무효화를 수행합니다. CloudFront
18. UpdateSessionLambda 함수는 DynamoDB 테이블에 저장된 세션 (토큰) 의 TokenTable 상태를 업데이트합니다. 세션 상태는 정수로 표시됩니다. 상태로 설정된 세션은 1 완료됨을 나타내며 중단됨을 -1 나타냅니다. 세션이 업데이트되었다는 이벤트를 대기실의 커스텀 이벤트 버스에 기록합니다.
19. TokenTableDynamoDB 테이블은 토큰 데이터를 저장합니다.
20. QueuePositionEntryTimeDynamoDB 테이블은 대기열 위치 및 진입 시간 데이터를 저장합니다.
21. ServingCounterIssuedAtDynamoDB 테이블은 서빙 카운터에 대한 업데이트를 저장합니다.
22. GetQueuePositionExpireTimeLambda 함수는 클라이언트가 남은 대기열 위치 만료 시간을 요청할 때 호출됩니다.
23. SetMaxQueuePositionExpiredLambda 함수는 테이블 값에 따라 만료된 최대 대기열 위치를 설정합니다. ServingCounterIssuedAt 코어 스택 배포 true 중에 IncrSvcOnQueuePositionExpiry 파라미터가 로 설정된 경우 1분마다 실행됩니다.

24 GenerateEventsLambda 함수는 대기실의 사용자 지정 이벤트 버스에 다양한 대기실 지표를 기록합니다. 코어 스택 배포 true 중에 이벤트 생성 활성화 파라미터가 설정된 경우 1분마다 실행됩니다.

25 AWS Secrets Manager는 토큰 작업을 위한 키와 기타 민감한 데이터를 저장합니다.

26 Amazon EventBridge 사용자 지정 이벤트 버스는 토큰이 생성되고 TokenTable DynamoDB 테이블에서 세션이 업데이트될 때마다 이벤트를 수신합니다. 또한 SetMaxQueuePositionExpired Lambda에서 서빙 카운터가 이동할 때 이벤트를 수신합니다. 코어 스택 배포 중에 활성화되면 다양한 대기실 지표와 함께 기록됩니다.

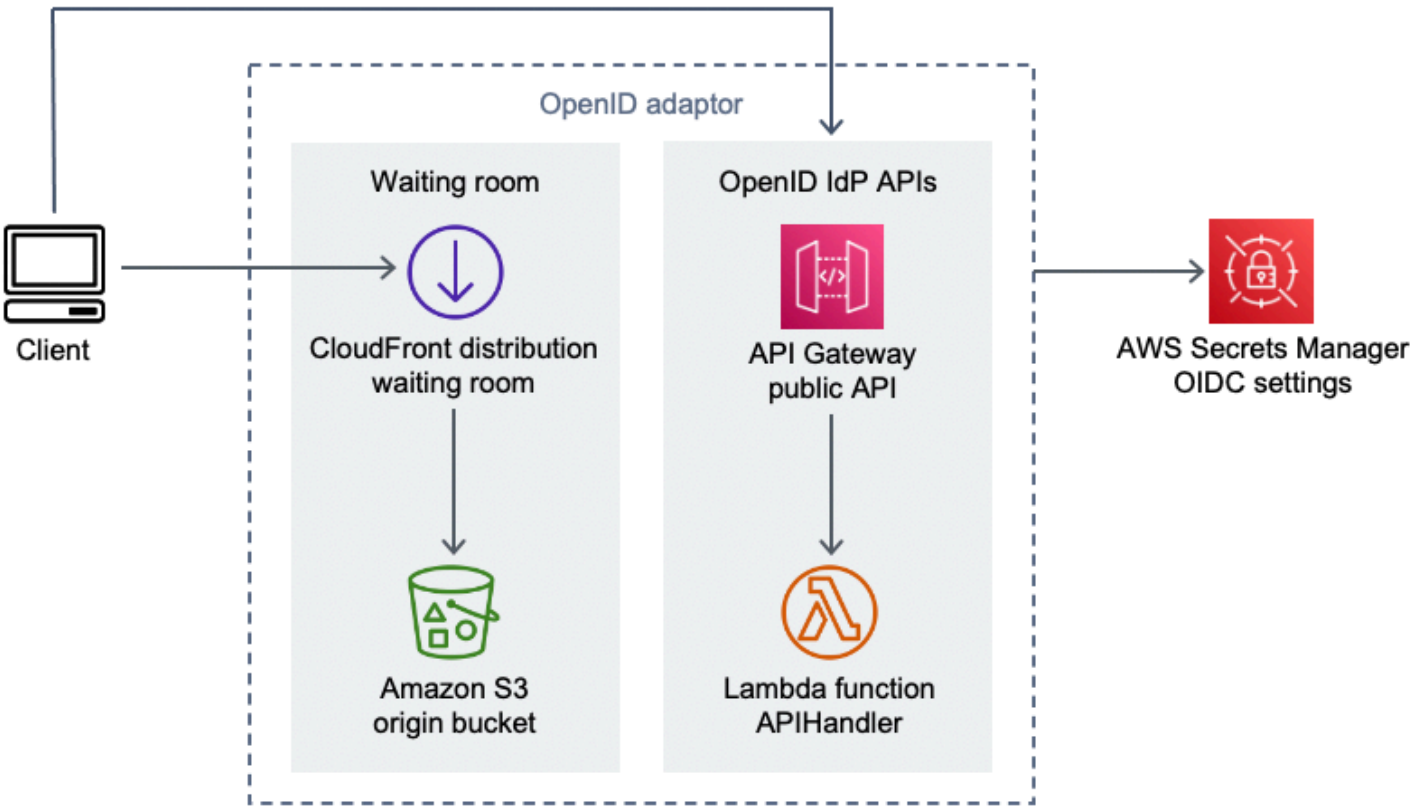
27 Amazon CloudWatch 이벤트 규칙은 코어 스택 배포 중에 이벤트 생성 활성화 파라미터가 true로 설정된 경우 생성됩니다. 이 이벤트 규칙은 GenerateEvents 1분마다 Lambda 함수를 시작합니다.

권한 부여자

솔루션에는 API Gateway Lambda 권한 부여자 스택이 포함되어 있습니다. 스택은 하나의 IAM 역할과 Lambda 함수로 구성됩니다. APIGatewayAuthorizerLambda 함수는 API Gateway의 권한 부여자로, API의 가상 대기실에서 발급한 토큰의 서명과 클레임을 검증할 수 있습니다. AWS 스택과 함께 제공되는 Lambda 함수는 사용자가 대기실을 통과하여 액세스 토큰을 받을 때까지 클라우드 API를 보호하는 데 사용할 수 있습니다. 권한 부여자는 토큰 검증을 위해 핵심 API에서 퍼블릭 키와 구성을 자동으로 검색하고 캐싱합니다. 수정 없이 사용할 수 있으며 지원하는 모든 AWS 지역에 설치할 수 있습니다. AWS Lambda

오픈ID 어댑터

OpenID 어댑터 스택은 OpenID ID 공급자 역할을 하는 API Gateway 및 Lambda 함수를 배포합니다. OpenID 어댑터는 OIDC 호환 API 세트를 제공합니다. 이 API는 AWS Elastic Load Balancers와 같은 OIDC ID 공급자를 지원하는 기존 웹 호스팅 소프트웨어와 함께 사용하거나 Amazon Cognito 또는 유사한 서비스의 페더레이션 ID 공급자로 사용할 수 있습니다. WordPress 이 어댑터를 통해 고객은 통합 옵션이 제한된 off-the-shelf 웹 호스팅 소프트웨어를 사용할 때 AuthN/Authz 플로우의 대기실을 사용할 수 있습니다. 또한 스택은 Amazon S3 버킷 하나를 오리진으로 사용하고 다른 S3 버킷은 요청 로깅을 위한 CloudFront 배포를 설치합니다. OpenID 어댑터는 샘플 대기실 스택에 제공된 것과 비슷하지만 OpenID 인증 흐름에 맞게 설계된 샘플 대기실 페이지를 제공합니다. 인증을 받는 과정에는 대기실 대기열의 위치를 확인하고 서빙 위치가 클라이언트의 대기열 위치와 같거나 더 커질 때까지 기다리는 과정이 포함됩니다. OpenID 대기실 페이지는 OpenID API를 사용하여 클라이언트의 토큰 획득 및 세션 구성을 완료하는 대상 사이트로 다시 리디렉션됩니다. 이 솔루션의 API 엔드포인트는 공식 OpenID Connect 1.0 흐름 name-for-name 사양에 직접 매핑됩니다. 자세한 내용은 OpenID Connect Core 1.0 인증을 참조하십시오.



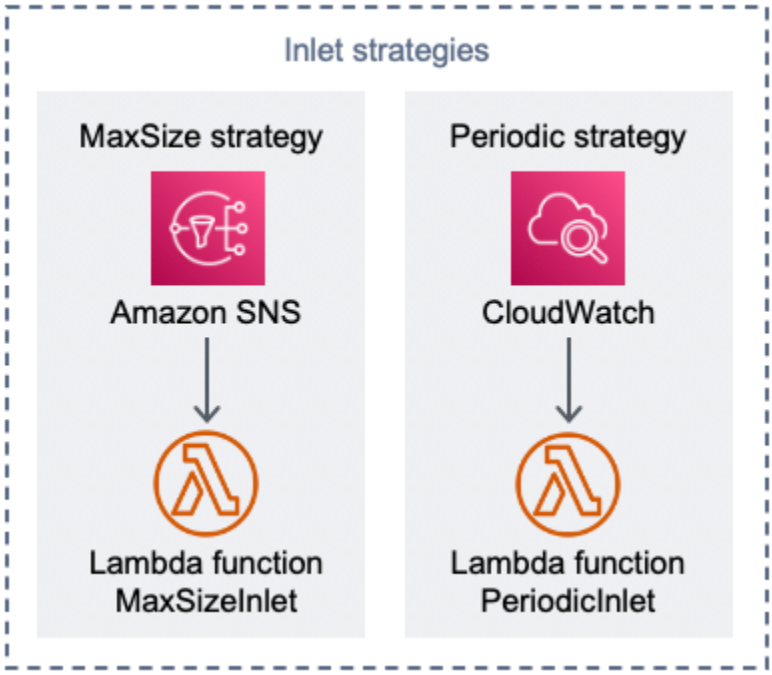
AWS OpenID 어댑터 구성 요소의 가상 대기실

1. CloudFront 배포는 S3 버킷의 콘텐츠를 사용자에게 제공합니다.
2. S3 버킷은 샘플 대기실 페이지를 호스팅합니다.
3. Amazon API Gateway API는 OIDC 자격 증명 공급자의 Lambda 권한 부여 기능을 지원하는 기존 웹 호스팅 소프트웨어와 함께 사용할 수 있는 OIDC 호환 API 세트를 제공합니다.
4. `APIHandler`Lambda 함수는 모든 API Gateway 리소스 경로에 대한 요청을 처리합니다. 동일한 모듈 내의 다른 Python 함수가 각 API 경로에 매핑됩니다. 예를 들어, API Gateway의 `/authorize` 리소스 경로는 Lambda 함수 `authorize()` 내에서 호출됩니다.
5. OIDC 설정은 Secrets Manager에 저장됩니다.

샘플 유입구 전략

유입구 전략은 대상 사이트에 더 많은 사용자를 수용하기 위해 솔루션의 서빙 카운터를 앞당겨야 하는 시기를 결정합니다. 대기실 유입 전략에 대한 자세한 개념 정보는 설계 고려 사항을 참조하십시오.

솔루션에서 제공하는 샘플 유입 전략에는 두 가지가 있습니다. 바로 주기율입니다. MaxSize



AWS 유입 전략 구성 요소의 가상 대기실

최대 크기 유입구 전략 옵션:

1. 클라이언트는 `MaxSizeInlet` Lambda 함수를 호출하여 메시지 페이로드를 기반으로 서비스 카운터를 늘리는 Amazon SNS 알림을 발행합니다.
2. `MaxSizeInletLambda` 함수는 메시지를 수신할 것으로 예상하고, 이 함수를 사용하여 서빙 카운터를 얼마나 늘릴지 결정합니다.

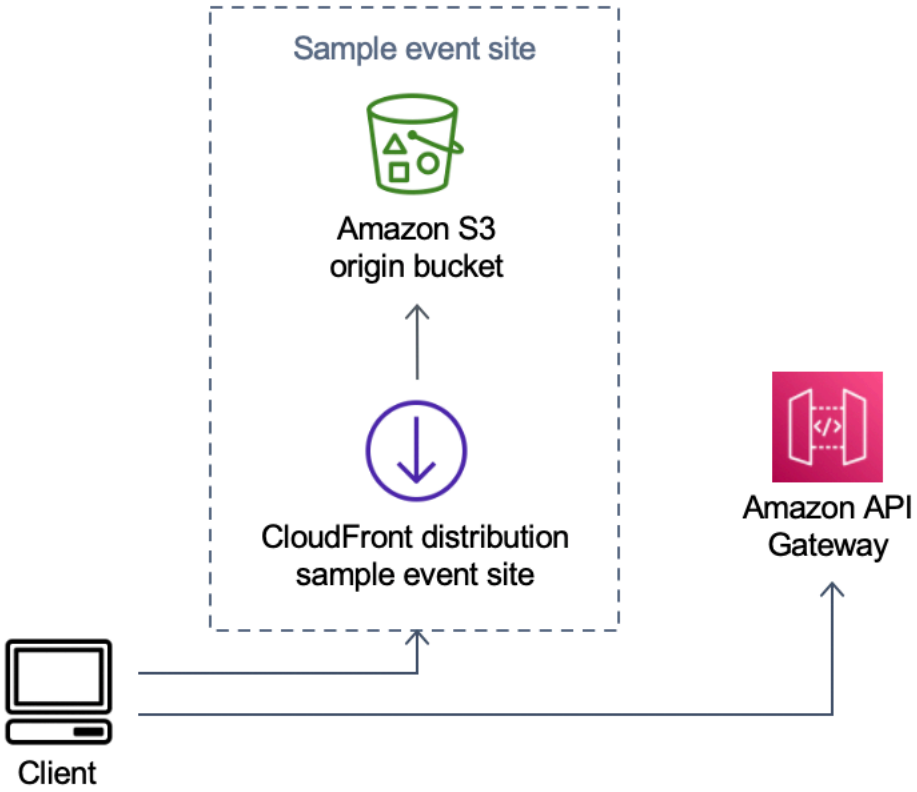
주기적 입력 전략 옵션:

3. CloudWatch 규칙은 1분마다 Lambda 함수를 호출하여 서빙 카운터를 고정된 수량만큼 늘립니다.
4. `PeriodicInletLambda` 함수는 제공된 시작 시간과 종료 시간 사이에 있는 경우 지정된 크기만큼 서빙 카운터를 증가시킵니다. 선택적으로 CloudWatch 경보를 확인하고 경보가 `OK` 상태에 있으면 증분을 수행하고 그렇지 않으면 건너뛰게 됩니다.

샘플 대기실

샘플 대기실은 사용자 지정 권한 부여 기능 외에도 퍼블릭 및 프라이빗 API와 통합되어 최소한의 end-to-end 대기실 솔루션을 보여줍니다. 기본 웹 페이지는 S3 버킷에 저장되며 오리진으로 사용됩니다. CloudFront 사용자에게 다음 단계를 안내합니다.

1. 대기실에서 줄을 서서 사이트에 입장하세요.
2. 줄 서서 고객의 위치를 파악하세요.
3. 대기실의 서빙 위치를 확인하세요.
4. 서빙 포지션이 클라이언트의 포지션과 같거나 그 이상이 되면 토큰 세트를 받으세요.
5. 토큰을 사용하여 Lambda 권한 부여자가 보호하는 API를 호출합니다.



AWS 샘플 이벤트 사이트 구성 요소의 가상 대기실

1. S3 버킷은 대기실 및 제어판의 샘플 콘텐츠를 호스팅합니다.
2. CloudFront 배포는 S3 버킷 콘텐츠를 사용자에게 제공합니다.
3. 및 와 같은 쇼핑과 유사한 리소스 경로를 포함하는 샘플 API Gateway 배포. `/search` `/checkout` 이 API는 스택에 의해 설치되며 토큰 권한 부여자로 구성됩니다. 이는 대기실에서 API를 보호하는 간단한 방법의 예시입니다. 유효한 토큰을 제시하는 요청은 Lambda로 전달되며, 그렇지 않으면 오류가 반환됩니다. 첨부된 Lambda 함수의 응답 외에는 API에 기능이 없습니다.

보안

AWS 인프라에 시스템을 구축할 때는 사용자와 시스템 간에 보안 책임이 분담됩니다 AWS. 이 공유 모델은 호스트 운영 체제, 가상화 계층, 서비스가 AWS 운영되는 시설의 물리적 보안을 비롯한 구성 요소를 운영, 관리 및 제어하므로 운영 부담이 줄어듭니다. AWS 보안에 대한 자세한 내용은 AWS 클라우드 보안을 참조하십시오.

ElastiCache for Redis에는 프라이빗 VPC 내부의 네트워크 인터페이스가 할당됩니다. for ElastiCache Redis와 상호 작용하는 Lambda 함수에도 VPC 내의 네트워크 인터페이스가 할당됩니다. 다른 모든 리소스는 공유 네트워크 공간에서 네트워크 연결을 제공합니다. AWS 다른 AWS 서비스와 상호 작용하는 VPC 인터페이스를 갖춘 Lambda 함수는 VPC 엔드포인트를 사용하여 이러한 서비스에 연결합니다.

JSON 웹 토큰을 생성하고 검증하는 데 사용되는 공개 키와 개인 키는 배포 시 생성되며 Secrets Manager에 저장됩니다. Redis에 연결하는 데 ElastiCache 사용되는 비밀번호도 배포 시 생성되어 Secrets Manager에 저장됩니다. 프라이빗 키와 ElastiCache Redis 비밀번호는 솔루션 API를 통해 액세스할 수 없습니다.

퍼블릭 API는 를 통해 CloudFront 액세스해야 합니다. 솔루션은 에서 사용자 지정 헤더의 값으로 사용되는 API Gateway용 API 키를 생성합니다 CloudFront. `x-api-key` CloudFront 오리진 요청 시 이 헤더를 포함합니다. 자세한 내용은 Amazon CloudFront 개발자 안내서의 오리진 요청에 사용자 지정 헤더 추가를 참조하십시오.

프라이빗 API는 호출 AWS 시 IAM 승인이 필요하도록 구성되어 있습니다. 솔루션은 프라이빗 API를 호출할 수 있는 적절한 권한을 가진 `ProtectedAPIGroup` IAM 사용자 그룹을 생성합니다. 이 그룹에 추가된 IAM 사용자는 프라이빗 API를 호출할 권한이 있습니다.

솔루션에서 생성한 다양한 리소스에 연결된 역할 및 권한에 사용되는 IAM 정책은 필요한 작업을 수행하는 데 필요한 권한만 부여합니다.

솔루션에서 생성된 S3 버킷, SQS 대기열, SNS 주제 등의 리소스의 경우 저장 및 전송 중 암호화가 가능한 경우 활성화됩니다.

모니터링

코어 API 스택에는 솔루션이 작동하는 동안 문제를 탐지하기 위해 모니터링할 수 있는 여러 CloudWatch 경보가 포함되어 있습니다. 스택은 Lambda 함수 오류 및 스로틀 조건에 대한 경보를 생성하고 1분 내에 오류 또는 스로틀 상태가 `OK` 발생하는 `ALARM` 경우로 경보 상태를 변경합니다.

또한 스택은 4XX 및 5XX 상태 코드에 대한 각 API Gateway 배포에 대한 경보를 생성합니다. 1분 내에 API에서 0K 4XX 또는 5XX 상태 코드가 ALARM 반환되면 경보 상태가 에서 로 바뀝니다.

이러한 경보는 1분 동안 오류나 병목 0K 현상이 없는 상태로 돌아옵니다.

IAM 역할

AWS Identity and Access Management (IAM) 역할을 통해 고객은 클라우드의 서비스와 사용자에게 세분화된 액세스 정책 및 권한을 할당할 수 있습니다. AWS 이 솔루션은 솔루션의 AWS Lambda 기능에 지역 리소스를 생성할 수 있는 액세스 권한을 부여하는 IAM 역할을 생성합니다.

아마존 CloudFront

대기실의 핵심 퍼블릭 및 프라이빗 API를 생성하는 `virtual-waiting-room-on-aws.template` CloudFormation 템플릿은 퍼블릭 API용 CloudFront 배포판도 배포합니다. CloudFront 퍼블릭 API의 응답을 캐싱하여 작업을 수행하는 API Gateway 및 Lambda 함수의 부하를 줄입니다.

이 솔루션에는 Amazon Simple Storage Service (Amazon S3) 버킷에 호스팅된 간단한 웹 애플리케이션을 배포하는 샘플 대기실 템플릿도 옵션으로 제공됩니다. 지연 시간을 줄이고 보안을 개선하기 위해 Amazon CloudFront 배포는 솔루션의 웹 사이트 버킷 콘텐츠에 대한 공개 액세스를 제공하는 CloudFront 사용자인 원본 액세스 ID와 함께 배포됩니다. 자세한 내용은 Amazon CloudFront 개발자 안내서의 원본 액세스 ID를 사용하여 Amazon S3 콘텐츠에 대한 액세스 제한을 참조하십시오.

보안 그룹

이 솔루션에서 만든 VPC 보안 그룹은 Redis용 네트워크 트래픽을 제어하고 격리하도록 설계되었습니다. ElastiCache Redis용 Lambda와 통신해야 하는 Lambda는 ElastiCache Redis용 Lambda와 동일한 보안 그룹에 배치됩니다. ElastiCache 보안 그룹을 검토하고 배포가 시작되고 실행되면 필요에 따라 액세스를 추가로 제한하는 것이 좋습니다.

설계 고려 사항

배포 옵션

처음 설치하는 경우나 무엇을 설치해야 할지 잘 모르겠으면 코어, 권한 부여자 및 샘플 대기실 CloudFormation 템플릿을 설치하는 `virtual-waiting-room-on-aws-getting-started.template` 중첩 템플릿을 배포하십시오. 이렇게 하면 간단한 흐름으로 최소한의 대기실을 확보할 수 있습니다.

지원되는 프로토콜

가상 대기실 온 AWS 솔루션은 다음과 통합될 수 있습니다.

- JSON 웹 토큰 검증 라이브러리 및 도구
- 기존 API Gateway 배포
- REST API 클라이언트
- OpenID 클라이언트 및 제공업체

대기실 유입 전략

유입 전략은 고객을 대기실에서 웹 사이트로 이동시키는 데 필요한 로직과 데이터를 캡슐화합니다. 유입구 전략은 Lambda 함수, 컨테이너, Amazon EC2 인스턴스 또는 기타 컴퓨팅 리소스로 구현될 수 있습니다. 대기실 퍼블릭 및 프라이빗 API를 호출할 수 있는 한 클라우드 리소스일 필요는 없습니다. 유입구 전략은 대기실, 웹 사이트 또는 기타 외부 지표에 대한 이벤트를 수신하여 더 많은 클라이언트가 토큰을 발행하고 사이트에 진입할 시기를 결정하는 데 도움이 됩니다. 인렛 전략에는 여러 가지 접근 방식이 있습니다. 어떤 방법을 채택할지는 사용 가능한 리소스와 보호 대상 웹 사이트 디자인의 제약에 따라 달라집니다.

인렛 전략에서 취하는 주요 조치는 사이트에 들어갈 수 있는 클라이언트 수를 나타내는 상대 값을 사용하여 `increment_serving_num` Amazon API Gateway 프라이빗 API를 호출하는 것입니다. 이 섹션에서는 두 가지 샘플 인렛 전략을 설명합니다. 그대로 사용하거나 사용자 정의할 수도 있고 완전히 다른 접근 방식을 사용할 수도 있습니다.

MaxSize

이 MaxSize 전략을 사용하면 `MaxSizeInlet` Lambda 함수는 웹 사이트를 동시에 사용할 수 있는 최대 클라이언트 수로 구성됩니다. 이는 고정된 값입니다. 클라이언트는 `MaxSizeInlet` Lambda 함수를 호출하여 메시지 페이로드를 기반으로 서비스 카운터를 늘리는 Amazon SNS 알림을 발행합니다. SNS 메시지의 소스는 웹 사이트의 코드나 사이트 사용률 수준을 관찰하는 모니터링 도구를 포함하여 어디에서든 올 수 있습니다.

`MaxSizeInletLambda` 함수는 다음을 포함할 수 있는 메시지를 수신할 것으로 예상합니다.

- `exited :`완료된 트랜잭션 수
- 완료로 표시할 요청 ID 목록
- 중단된 것으로 표시할 요청 ID 목록

이 데이터는 서빙 카운터를 얼마나 늘릴지 결정하는 데 사용됩니다. 현재 클라이언트 수를 기준으로 카운터를 늘릴 수 있는 추가 용량이 없는 경우가 있을 수 있습니다.

주기적

주기적 전략을 사용하는 경우 CloudWatch 규칙은 1분마다 `PeriodicInlet` Lambda 함수를 호출하여 서빙 카운터를 고정된 수량만큼 늘립니다. 주기적 입력은 이벤트 시작 시간, 종료 시간 및 증가량을 사용하여 파라미터화됩니다. 선택적으로 이 전략은 경보를 검사하여 CloudWatch 경보가 `OK` 상태에 있으면 증분을 수행하고 그렇지 않으면 경보를 건너뛰게 됩니다. 사이트 통합자는 사용률 지표를 경보에 연결하고 해당 경보를 사용하여 주기적 입력을 일시 중지할 수 있습니다. 이 전략은 현재 시간이 시작 시간과 종료 시간 사이이고 선택적으로 지정된 알람이 해당 상태에 있는 동안에만 서비스 제공 위치만 변경합니다. `OK`

솔루션 사용자 지정 및 확장

조직의 사이트 관리자는 대기실에서 사용할 통합 방법을 결정해야 합니다. 두 가지 옵션이 있습니다.

1. API 및 API Gateway 권한 부여자를 사용하여 직접 기본 통합
2. ID 공급자를 통한 OpenID 통합

위의 통합 외에도 도메인 이름 리디렉션을 구성해야 할 수 있습니다. 또한 사용자 지정된 대기실 사이트 페이지를 배포해야 합니다.

Virtual Waiting Room on AWS 솔루션은 두 가지 메커니즘, 즉 단방향 이벤트 알림을 EventBridge 위한 것과 양방향 통신을 위한 REST API를 통해 확장할 수 있도록 설계되었습니다.

할당량

가상 대기실의 기본 규모 AWS 제한은 설치된 지역의 Lambda 스로틀 한도입니다. AWS 기본 Lambda 동시 실행 할당량이 있는 AWS 계정에 설치하면 솔루션의 가상 대기실에서 AWS 대기열 위치를 요청하는 클라이언트를 초당 최대 500개까지 처리할 수 있습니다. 초당 500 클라이언트 속도는 모든 Lambda 함수 동시 할당량 한도가 독점적으로 제공되는 솔루션을 기반으로 합니다. 계정의 지역을 Lambda 함수를 호출하는 다른 솔루션과 공유하는 경우 솔루션의 가상 대기실에는 최소 AWS 1,000건의 동시 호출을 사용할 수 있어야 합니다. CloudWatch 지표를 사용하여 시간 경과에 따른 계정의 Lambda 동시 호출을 차트로 나타내어 결정을 내릴 수 있습니다. Service Quotas 콘솔을 사용하여 증가를 요청할 수 있습니다. Lambda 스로틀 한도를 늘리면 추가 호출이 실제로 발생하는 경우에만 월별 계정 요금이 인상됩니다.

초당 500개의 클라이언트가 추가될 때마다 스로틀 한도를 1,000씩 늘리십시오.

초당 유입 사용자 수 예상	권장 동시 실행 할당량
0-500	1,000 (기본값)
501-1,000	2,000
1,001-1,500	3,000

Lambda의 동시 호출 버스트 한도는 3,000건으로 고정되어 있습니다. 자세한 내용은 Lambda 함수 스케일링을 참조하십시오. 일시적인 스로틀 상황을 나타내는 오류 코드가 반환되는 경우 클라이언트 코드는 일부 API 호출을 예상하고 재시도해야 합니다. 샘플 대기실 클라이언트에는 대용량 및 고용량 버스트 이벤트에 사용되는 클라이언트를 설계하는 방법의 예로 이 코드가 포함되어 있습니다.

이 솔루션은 사용자 지정 구성 단계가 있는 Lambda 예약 및 프로비저닝된 동시성과도 호환됩니다. 자세한 내용은 Lambda 예약 동시성 관리를 참조하십시오.

대기실에 입장하여 토큰을 받고 거래를 계속할 수 있는 사용자의 상한은 Redis 카운터의 상한으로 제한됩니다. ElastiCache 카운터는 대기실 서비스 위치 및 솔루션의 요약 상태 추적에 사용됩니다. Redis에 사용되는 카운터의 상한은 ElastiCache 9,223,372,036,854,775,807입니다. DynamoDB 테이블은 대기실 사용자에게 발급된 각 토큰의 사본을 저장하는 데 사용됩니다. DynamoDB는 테이블 크기에 실질적인 제한이 없습니다.

지역별 배포

이 솔루션에서 사용하는 서비스는 모든 AWS 지역에서 지원됩니다. 지역별 최신 AWS 서비스 가용성은 AWS 지역 서비스 목록을 참조하십시오.

AWS CloudFormation 템플리트

배포를 자동화하기 위해 이 솔루션은 배포 전에 다운로드할 수 있는 다음 AWS CloudFormation 템플릿을 사용합니다.

처음 설치하는 경우이거나 무엇을 설치해야 할지 잘 모르겠으면 `virtual-waiting-room-on-aws-getting-started.template` AWS CloudFormation 템플릿을 배포하여 코어, 권한 부여자 및 샘플 대기실 코드 템플릿을 설치합니다. 이를 통해 간단한 흐름으로 실제 대기실을 테스트할 수 있습니다.

View template

virtual-waiting-room-on- aws-api-gateway-cw -logs-role.template: 이 템플릿을 사용하면 계정 수준에서 API Gateway에 로깅 권한에 대한 기본 역할 ARN을 추가할 수 있습니다. CloudWatch 계정에 이 템플릿의 배포가 필요한지 여부에 대한 자세한 내용은 사전 요구 사항을 참조하십시오.

View template

virtual-waiting-room-on- aws-getting-started .template: 이 중첩된 템플릿을 사용하여 코어, 권한 부여자 및 샘플 대기실 스택을 설치합니다.

View template

virtual-waiting-room-on-aws.template: 이 핵심 템플릿을 사용하여 대기실 이벤트 생성을 위한 핵심 퍼블릭 및 프라이빗 REST API와 클라우드 서비스를 설치할 수 있습니다. 대기실 REST API, ElastiCache Redis 용 및 DynamoDB 테이블이 필요한 계정 및 지역에 이 템플릿을 설치합니다.

View template

virtual-waiting-room-on-aws-authorizers.template: 이 템플릿을 사용하여 대기실에서 발급한 토큰을 검증하도록 설계되고 최종 사용자의 API를 보호하도록 설계된 Lambda 권한 부여자를 설치합니다. 코어 스택이 필요합니다. 이 스택을 배포하려면 코어 스택의 일부 출력이 매개변수로 필요합니다. 이 템플릿은 선택적 템플릿입니다.

View template

virtual-waiting-room-on-aws-openid.template: 이 템플릿을 사용하여 인증 인터페이스와의 대기실 통합을 위

한 OpenID ID 공급자를 설치할 수 있습니다. 코어 스택이 필요합니다. 이 스택을 배포하려면 코어 스택의 일부 출력이 필요합니다. 이 템플릿은 선택적 템플릿입니다.

View template

virtual-waiting-room-on- aws-sample-inlet-strategy .template: 이 템플릿을 사용하면 대상 사이트와 대기실 간에 사용할 샘플 유입구 전략을 설치할 수 있습니다. 유입구 전략은 로직을 캡슐화하여 대상 사이트에 더 많은 사용자를 허용할 시기를 결정하는 데 도움이 됩니다. 코어 스택이 필요합니다. 이 스택을 배포하려면 코어 스택의 출력이 필요합니다. 이 템플릿은 선택적 템플릿입니다.

View template

virtual-waiting-room-on-aws-sample.template: 이 템플릿을 사용하여 대기실 및 대상 사이트에 대한 샘플 최소 웹 및 API Gateway 구성을 설치합니다. 코어 및 권한 부여자 스택이 필요합니다. 이 스택을 배포하려면 코어 및 권한 부여자 스택의 출력이 매개변수로 필요합니다. 이 템플릿은 선택적 템플릿입니다.

배포 자동화

솔루션을 시작하기 전에 이 가이드에서 설명하는 비용, 아키텍처, 네트워크 보안 및 기타 고려 사항을 검토하십시오. 이 섹션의 step-by-step 지침에 따라 솔루션을 구성하고 계정에 배포하십시오.

배포 시간: 약 30분 (시작 스택만 해당)

사전 조건

- AWS 계정 콘솔 권한은 관리자 액세스와 동일합니다.
- API Gateway에서 CloudWatch 로깅을 활성화합니다.
  - API Gateway 콘솔에 로그인하고 스택을 설치할 지역을 선택합니다.

    이 지역에 정의된 기존 API가 있는 경우:

    1. 아무 API나 선택하세요.
    2. 왼쪽 탐색창에서 설정을 선택합니다.
    3. CloudWatch 로그 역할 ARN 필드에서 값을 확인합니다.
    - ARN이 없는 경우 를 설치합니다. `virtual-waiting-room-on-aws-api-gateway-cw-logs-role.template`
    - ARN이 있는 경우 시작 스택을 시작하는 것부터 시작하십시오.

    이 지역에 정의된 기존 API가 없는 경우 를 설치하십시오. `virtual-waiting-room-on-aws-api-gateway-cw-logs-role.template`
- 보호할 대상 사이트의 아키텍처 및 구현 세부 정보에 대한 지식

배포 개요

다음 단계를 사용하여 이 솔루션을 배포하십시오 AWS. 자세한 지침은 각 단계에 대한 링크를 따르십시오.

단계 1. 시작 스택을 실행하세요.

- 계정에서 AWS CloudFormation 템플릿을 실행합니다. AWS
- 템플릿 매개변수를 검토하고 필요에 따라 기본값을 입력하거나 조정합니다.

단계 2. (선택 사항) 대기실 테스트

- IAM 보안 API를 호출하기 위한 AWS 키를 생성합니다.
- 샘플 대기실의 제어판을 엽니다.
- 샘플 대기실을 테스트해 보세요.

단계 1. 시작 스택을 실행하세요.

이 자동화된 AWS CloudFormation 템플릿은 코어, 권한 부여자 및 샘플 대기실 템플릿을 배포하며, 이를 통해 실제 대기실을 보고 테스트할 수 있습니다. 스택을 시작하기 전에 사전 요구 사항을 읽고 이해해야 합니다.

> **Note**
>
> 이 솔루션을 실행하는 동안 사용되는 AWS 서비스 비용은 사용자가 부담해야 합니다. 자세한 내용은 이 가이드의 비용 섹션을 방문하고 이 솔루션에서 사용되는 각 AWS 서비스의 요금 웹 페이지를 참조하십시오.

1. 에 AWS Management Console로그인하고 버튼을 선택하여 `virtual-waiting-room-on-aws-getting-started.template` AWS CloudFormation 템플릿을 실행합니다.

   Launch solution

   또는 템플릿을 다운로드하여 직접 구현하기 위한 시작점으로 사용할 수도 있습니다.
2. 이 템플릿은 기본적으로 미국 동부(버지니아 북부) 리전에서 시작됩니다. 다른 AWS 지역에서 솔루션을 실행하려면 콘솔 탐색 표시줄의 지역 선택기를 사용하십시오.
3. 스택 생성 페이지에서 Amazon S3 URL 텍스트 상자에 올바른 템플릿 URL이 있는지 확인하고 다음을 선택합니다.
4. 스택 세부 정보 지정 페이지에서 솔루션 스택 이름을 할당합니다. 이름 지정 문자 제한에 대한 자세한 내용은 사용 AWS Identity and Access Management 설명서의 IAM 및 STS 제한을 참조하십시오.
5. 매개 변수에서 이 솔루션 템플릿의 매개 변수를 검토하고 필요에 따라 수정하십시오. 이 솔루션은 다음과 같은 기본값을 사용합니다.

파라미터	기본값	설명
이벤트 ID	`Sample`	이 대기실 인스턴스의 고유 ID (GUID 형식 권장)
유효 기간	`3600`	토큰 유효 기간 (초)
이벤트 생성 활성화	`false`	로 `true` 설정하면 대기실과 관련된 지표가 1분마다 이벤트 버스에 기록됩니다.
레디 스 포트	`1785`	Redis 서버 ElastiCache 연결에 사용할 포트 번호입니다. Redis 포트의 ElastiCache 기본값을 사용하지 않는 것이 좋습니다. 6379
EnableQueuePositionExpiry	`true`	로 `false` 설정하면 대기열 위치 만료 기간이 적용되지 않습니다.
QueuePositionExpiryPeriod	`900`	대기열 위치에서 토큰을 생성할 수 없는 시간 간격 (초) 입니다.
IncrSvcOnQueuePositionExpiry	`false`	로 설정하면 토큰을 `true` 성공적으로 생성하지 못한 만료된 대기열 위치를 기준으로 서빙 카운터가 자동으로 상향 조정됩니다.

6. 다음을 선택합니다.
7. Configure stack options(스택 옵션 구성) 페이지에서 Next(다음)를 선택합니다.
8. 검토 페이지에서 설정을 검토하고 확인합니다. 템플릿이 AWS Identity and Access Management (IAM) 리소스를 생성한다는 것을 확인하는 확인란을 선택합니다.
9. [스택 생성(Create stack)]을 선택하여 스택을 배포합니다.

AWS CloudFormation 콘솔의 상태 열에서 스택 상태를 볼 수 있습니다. 약 30분 후에 CREATE_COMPLETE 상태를 받게 됩니다.

단계 2. (선택 사항) 대기실 테스트

시작 스택을 배포한 경우 다음 단계를 통해 대기실 기능을 테스트할 수 있습니다. 테스트를 완료하려면 코어 스택에서 IAM 보안 API를 호출할 권한이 있는 AWS 키가 필요합니다.

IAM 보안 API를 호출하기 위한 AWS 키를 생성하십시오.

1. `aws-virtual-waiting-room-getting-started.template` CloudFormation 템플릿이 배포된 AWS 계정에서 IAM 사용자를 만들거나 사용하십시오.
2. IAM 사용자에게 프로그래밍 액세스 권한을 부여합니다. IAM 사용자를 위한 새 액세스 키 세트를 생성할 때 키 파일이 제시되면 해당 키 파일을 다운로드하십시오. 대기실을 테스트하려면 IAM 사용자의 액세스 키 ID와 보안 액세스 키가 필요합니다.
3. 템플릿에서 생성한 ProtectedAPIGroup IAM 사용자 그룹에 IAM 사용자를 추가합니다.

샘플 대기실의 제어판을 엽니다.

1. AWS CloudFormation 콘솔에 로그인하고 솔루션의 시작 스택을 선택합니다.
2. 출력 탭을 선택합니다.
3. 키 열에서 ControlPanelURL을 찾아 해당 값을 선택합니다.
4. 새 탭 또는 브라우저 창에서 제어판을 엽니다.
5. 제어판에서 구성 섹션을 확장합니다.
6. IAM 보안 API를 호출하려면 키 생성에서 검색한 액세스 키 ID와 보안 액세스 AWS 키를 입력합니다. 엔드포인트와 이벤트 ID는 URL 파라미터에서 입력됩니다.
7. [사용] 을 선택합니다. 자격 증명을 제공한 후 버튼이 활성화됩니다.

샘플 대기실을 테스트하세요.

1. AWS CloudFormation 콘솔에서 솔루션의 시작 스택을 선택합니다.
2. 출력 탭을 선택합니다.
3. 키 열에서 WaitingRoomURL을 찾아 해당 값을 선택합니다.

4. 대기실을 연 다음 예약을 선택하여 대기실에 입장합니다.
5. 제어판이 있는 브라우저 탭으로 다시 이동합니다.
6. 인크리먼트 서빙 카운터에서 변경을 선택합니다. 이렇게 하면 100명의 사용자가 대기실에서 대상 사이트로 이동할 수 있습니다.
7. 대기실로 돌아가서 지금 체크아웃을 선택하세요! 이제 대상 사이트로 리디렉션됩니다.
8. 지금 구매를 선택하여 대상 사이트에서 거래를 완료하세요.

개별 스택 배포

코어 스택은 대기실의 주요 기능을 사용하는 데 필요한 유일한 스택입니다. 다른 모든 스택은 선택 사항입니다. 대기실에서 발급한 토큰을 검증하거나 이미 보유하고 있을 수 있는 API를 보호할 방법이 아직 없다면 Authorizers 스택을 실행하세요. 인증 인터페이스와의 대기실 통합을 위해 OpenID ID 공급자가 필요한 경우 OpenID 스택을 시작하십시오. 샘플 유입구 전략 스택은 보호하려는 사이트에 더 많은 사용자를 언제 어떻게 허용해야 하는지에 대한 몇 가지 예를 제공합니다.

1. 코어 스택을 실행하세요.

배포에 소요되는 시간: 약 20분

이 자동화된 AWS CloudFormation 템플릿은 AWS 클라우드에 가상 대기실을 배포합니다. AWS 스택을 시작하기 전에 사전 요구 사항을 완료해야 합니다.

> **Note**
>
> 이 솔루션을 실행하는 동안 사용되는 AWS 서비스 비용은 사용자가 부담해야 합니다. 자세한 내용은 이 가이드의 비용 섹션을 방문하고 이 솔루션에서 사용되는 각 AWS 서비스의 요금 웹 페이지를 참조하십시오.

1. 에 AWS Management Console로그인하고 버튼을 선택하여 `aws-virtual-waiting-room-on-aws.template` AWS CloudFormation 템플릿을 실행합니다.

   Launch solution 또는 템플릿을 다운로드하여 직접 구현하기 위한 시작점으로 사용할 수도 있습니다.
2. 이 템플릿은 기본적으로 미국 동부(버지니아 북부) 리전에서 시작됩니다. 다른 AWS 지역에서 솔루션을 실행하려면 콘솔 탐색 표시줄의 지역 선택기를 사용하십시오.
3. 스택 생성 페이지에서 Amazon S3 URL 텍스트 상자에 올바른 템플릿 URL이 있는지 확인하고 다음을 선택합니다.
4. 스택 세부 정보 지정 페이지에서 솔루션 스택 이름을 할당합니다. 이름 지정 문자 제한에 대한 자세한 내용은 사용 AWS Identity and Access Management 설명서의 IAM 및 STS 제한을 참조하십시오.

5. 매개 변수에서 이 솔루션 템플릿의 매개 변수를 검토하고 필요에 따라 수정하십시오. 이 솔루션은 다음과 같은 기본값을 사용합니다.

파라미터	기본값	설명
이벤트 ID	`Sample`	이 대기실 인스턴스의 고유 ID, GUID 형식 권장.
유효 기간	`3600`	토큰 유효 기간 (초).
이벤트 생성 활성화	`false`	로 `true` 설정하면 대기실과 관련된 지표가 1분마다 이벤트 버스에 기록됩니다.
레디 스 포트	`1785`	Redis 서버 ElastiCache 연결에 사용할 포트 번호입니다. Redis 포트의 ElastiCache 기본값을 사용하지 않는 것이 좋습니다. 6379
EnableQueuePositionExpiry	`true`	로 `false` 설정하면 대기열 위치 만료 기간이 적용되지 않습니다.
QueuePositionExpiryPeriod	`900`	대기열 위치에서 토큰을 생성할 수 없는 시간 간격 (초) 입니다.
IncrSvcOnQueuePositionExpiry	`false`	로 설정하면 토큰을 `true` 성공적으로 생성하지 못한 만료된 대기열 위치를 기준으로 서빙 카운터가 자동으로 상향 조정됩니다.

6. 다음을 선택합니다.
7. Configure stack options(스택 옵션 구성) 페이지에서 Next(다음)를 선택합니다.
8. 검토 페이지에서 설정을 검토하고 확인합니다. 템플릿이 AWS Identity and Access Management (IAM) 리소스를 생성한다는 것을 확인하는 확인란을 선택합니다.

9. [스택 생성(Create stack)]을 선택하여 스택을 배포합니다.

   AWS CloudFormation 콘솔의 상태 열에서 스택 상태를 볼 수 있습니다. 약 20분 후에 CREATE_COMPLETE 상태를 받게 됩니다.

2. (선택 사항) 권한 부여자 스택 시작

배포에 소요되는 시간: 약 5분

1. 에 AWS Management Console로그인하고 버튼을 선택하여 템플릿을 실행합니다. `aws-virtual-waiting-room-on-aws-authorizers.template` AWS CloudFormation

   Launch solution 또는 템플릿을 다운로드하여 직접 구현하기 위한 시작점으로 사용할 수도 있습니다.
2. 이 템플릿은 기본적으로 미국 동부(버지니아 북부) 리전에서 시작됩니다. 다른 AWS 지역에서 솔루션을 실행하려면 콘솔 탐색 표시줄의 지역 선택기를 사용하십시오.
3. 스택 생성 페이지에서 Amazon S3 URL 텍스트 상자에 올바른 템플릿 URL이 있는지 확인하고 다음을 선택합니다.
4. 스택 세부 정보 지정 페이지에서 솔루션 스택 이름을 할당합니다. 이름 지정 문자 제한에 대한 자세한 내용은 사용 AWS Identity and Access Management 설명서의 IAM 및 STS 제한을 참조하십시오.
5. 매개 변수에서 이 솔루션 템플릿의 매개 변수를 검토하고 필요에 따라 수정하십시오. 이 솔루션은 다음과 같은 기본값을 사용합니다.

파라미터	기본값	설명
퍼블릭 API 엔드포인트	`<## ##>`	가상 대기실 API용 퍼블릭 엔드포인트.
대기실 이벤트 ID	`Sample`	대기실의 이벤트 ID.
발급자 URI	`<## ##>`	퍼블릭 키와 토큰의 발급자 URI.

6. 다음을 선택합니다.
7. Configure stack options(스택 옵션 구성) 페이지에서 Next(다음)를 선택합니다.

8. 검토 페이지에서 설정을 검토하고 확인합니다. 템플릿이 AWS Identity and Access Management (IAM) 리소스를 생성함을 확인하는 체크박스를 선택합니다.
9. [스택 생성(Create stack)]을 선택하여 스택을 배포합니다.

   AWS CloudFormation 콘솔의 상태 열에서 스택 상태를 볼 수 있습니다. 약 5분 후에 CREATE_COMPLETE 상태를 받게 됩니다.

3. (선택 사항) OpenID 스택 실행

배포에 소요되는 시간: 약 5분

1. 에 AWS Management Console로그인하고 버튼을 선택하여 `aws-virtual-waiting-room-on-aws-openid.template` AWS CloudFormation 템플릿을 실행합니다.

   Launch solution 또는 템플릿을 다운로드하여 직접 구현하기 위한 시작점으로 사용할 수도 있습니다.
2. 이 템플릿은 기본적으로 미국 동부(버지니아 북부) 리전에서 시작됩니다. 다른 AWS 지역에서 솔루션을 실행하려면 콘솔 탐색 표시줄의 지역 선택기를 사용하십시오.
3. 스택 생성 페이지에서 Amazon S3 URL 텍스트 상자에 올바른 템플릿 URL이 있는지 확인하고 다음을 선택합니다.
4. 스택 세부 정보 지정 페이지에서 솔루션 스택 이름을 할당합니다. 이름 지정 문자 제한에 대한 자세한 내용은 사용 AWS Identity and Access Management 설명서의 IAM 및 STS 제한을 참조하십시오.
5. 매개 변수에서 이 솔루션 템플릿의 매개 변수를 검토하고 필요에 따라 수정하십시오. 이 솔루션은 다음과 같은 기본값을 사용합니다.

파라미터	기본값	설명
퍼블릭 API 엔드포인트	*<## ##>*	가상 대기실 API용 퍼블릭 엔드포인트 URL.
프라이빗 API 엔드포인트	*<## ##>*	가상 대기실 API용 프라이빗 엔드포인트 URL.

파라미터	기본값	설명
API 지역	*<## ##>*	AWS 공개 및 비공개 대기실 API의 지역 이름.
이벤트 ID	`Sample`	대기실의 이벤트 ID.

6. 다음을 선택합니다.
7. Configure stack options(스택 옵션 구성) 페이지에서 Next(다음)를 선택합니다.
8. 검토 페이지에서 설정을 검토하고 확인합니다. 템플릿이 AWS Identity and Access Management (IAM) 리소스를 생성함을 확인하는 확인란을 선택합니다.
9. [스택 생성(Create stack)]을 선택하여 스택을 배포합니다.

   AWS CloudFormation 콘솔의 상태 열에서 스택 상태를 볼 수 있습니다. 약 5분 후에 CREATE_COMPLETE 상태를 받게 됩니다.

4. (선택 사항) 샘플 유입구 전략 스택 실행

배포 시간: 약 2분

1. 에 AWS Management Console로그인하고 버튼을 선택하여 `aws-virtual-waiting-room-sample-inlet-strategy.template` AWS CloudFormation 템플릿을 실행합니다.

   Launch solution 또는 템플릿을 다운로드하여 직접 구현하기 위한 시작점으로 사용할 수도 있습니다.
2. 이 템플릿은 기본적으로 미국 동부(버지니아 북부) 리전에서 시작됩니다. 다른 AWS 지역에서 솔루션을 실행하려면 콘솔 탐색 표시줄의 지역 선택기를 사용하십시오.
3. 스택 생성 페이지에서 Amazon S3 URL 텍스트 상자에 올바른 템플릿 URL이 있는지 확인하고 다음을 선택합니다.
4. 스택 세부 정보 지정 페이지에서 솔루션 스택 이름을 할당합니다. 이름 지정 문자 제한에 대한 자세한 내용은 사용 AWS Identity and Access Management 설명서의 IAM 및 STS 제한을 참조하십시오.
5. 매개 변수에서 이 솔루션 템플릿의 매개 변수를 검토하고 필요에 따라 수정하십시오. 이 솔루션은 다음과 같은 기본값을 사용합니다.

파라미터	기본값	설명
이벤트 ID	`Sample`	대기실의 이벤트 ID.
프라이빗 코어 API 엔드포인트	*<## ##>*	가상 대기실 API용 프라이빗 엔드포인트 URL.
코어 API 지역	*<## ##>*	AWS 코어 API가 설치된 지역.
인렛 전략	`Periodic`	유입구 전략 배포 예정. `Periodic`1분마다 서빙 수를 증가시킵니다. `MaxSize`다운스트림 대상 사이트에서 지정된 시간에 처리할 수 있는 최대 트랜잭션 수를 기준으로 서비스 수를 증가시킵니다.
증분 기준	*<## ##>*	1분마다 서빙 카운터를 얼마나 늘려야 하는지. 주기적 유입 전략을 선택할 때 필요합니다.
시작 시간	*<## ##>*	서빙 번호 증가 시작 시점의 타임스탬프 (에포크 시간 (초)). 주기적 입력 전략을 선택할 때 필요합니다.
종료 시간	*<## ##>*	서빙 넘버 증가를 중지해야 하는 시점의 타임스탬프 (에포크 시간 (초)). 0을 그대로 두면 서빙 번호가 무기한 증가합니다. 주기적 유입 전략을 선택할 때 필요합니다.

파라미터	기본값	설명
CloudWatch 알람 이름	*<## ##>*	주기적 유입 전략에 연결할 선택적 CloudWatch 알람 이름. 제공되고 경보 상태인 경우, 서빙 번호는 증가하지 않습니다. 주기적 유입 전략에만 적용됩니다.
최대 크기	*<## ##>*	다운스트림 대상 사이트에서 한 번에 처리할 수 있는 최대 트랜잭션 수 (MaxSize 전략)

6. 다음을 선택합니다.
7. Configure stack options(스택 옵션 구성) 페이지에서 Next(다음)를 선택합니다.
8. 검토 페이지에서 설정을 검토하고 확인합니다. 템플릿이 AWS Identity and Access Management (IAM) 리소스를 생성한다는 것을 확인하는 확인란을 선택합니다.
9. [스택 생성(Create stack)]을 선택하여 스택을 배포합니다.

   AWS CloudFormation 콘솔의 상태 열에서 스택 상태를 볼 수 있습니다. 약 2분 후에 CREATE_COMPLETE 상태를 받게 됩니다.

5. (선택 사항) 샘플 대기실 스택 실행

배포에 소요되는 시간: 약 5분

1. 에 AWS Management Console로그인하고 버튼을 선택하여 `aws-virtual-waiting-room-sample.template` AWS CloudFormation 템플릿을 실행합니다.

   Launch solution 또는 템플릿을 다운로드하여 직접 구현하기 위한 시작점으로 사용할 수도 있습니다.
2. 이 템플릿은 기본적으로 미국 동부(버지니아 북부) 리전에서 시작됩니다. 다른 AWS 지역에서 솔루션을 실행하려면 콘솔 탐색 표시줄의 지역 선택기를 사용하십시오.
3. 스택 생성 페이지에서 Amazon S3 URL 텍스트 상자에 올바른 템플릿 URL이 있는지 확인하고 다음을 선택합니다.

4. 스택 세부 정보 지정 페이지에서 솔루션 스택 이름을 할당합니다. 이름 지정 문자 제한에 대한 자세한 내용은 사용 AWS Identity and Access Management 설명서의 IAM 및 STS 제한을 참조하십시오.
5. 매개 변수에서 이 솔루션 템플릿의 매개 변수를 검토하고 필요에 따라 수정하십시오. 이 솔루션은 다음과 같은 기본값을 사용합니다.

파라미터	기본값	설명
API Gateway 지역	`<## ##>`	AWS API Gateway의 지역 이름.
권한 부여자 ARN	`<## ##>`	API Gateway Lambda 권한 부여자의 ARN.
이벤트 ID	`Sample`	대기실의 이벤트 ID.
프라이빗 API 엔드포인트	`<## ##>`	가상 대기실 API용 프라이빗 엔드포인트 URL.
퍼블릭 API 엔드포인트	`<## ##>`	가상 대기실 API용 퍼블릭 엔드포인트 URL.

6. 다음을 선택합니다.
7. Configure stack options(스택 옵션 구성) 페이지에서 Next(다음)를 선택합니다.
8. 검토 페이지에서 설정을 검토하고 확인합니다. 템플릿이 AWS Identity and Access Management (IAM) 리소스를 생성한다는 것을 확인하는 확인란을 선택합니다.
9. [스택 생성(Create stack)]을 선택하여 스택을 배포합니다.

   AWS CloudFormation 콘솔의 상태 열에서 스택 상태를 볼 수 있습니다. 약 5분 후에 CREATE_COMPLETE 상태를 받게 됩니다.

이전 버전에서 스택 업데이트

스택을 삭제하고 새 버전용 새 스택을 생성하는 것이 좋습니다. 현재 CloudFormation 스택 업데이트를 사용한 새 버전으로의 마이그레이션은 지원되지 않습니다. 솔루션 제거그런 다음 시작 스택을 실행하십시오.

> **Note**
>
> 진행 중인 이벤트를 지원하기 위해 솔루션을 적극적으로 사용하지 않는 경우 새 버전으로 마이그레이션하는 것이 좋습니다.

성능 데이터

가상 대기실 AWS 온은 Locust라는 도구를 사용하여 부하 테스트를 거쳤습니다. 시뮬레이션된 이벤트 크기는 클라이언트 수가 10,000~100,000개에 달했습니다. 부하 테스트 환경은 다음과 같은 구성으로 구성되었습니다.

- 클라우드 배포를 위한 사용자 지정 기능이 포함된 Locust 2.x AWS
- 4개 AWS 지역 (,,,) `us-west-1` `us-west-2` `us-east-1` `us-east-2`
- 지역당 `c5.4xlarge` 아마존 EC2 호스트 10개 (총 40개)
- 호스트당 32개의 로커스트 프로세스
- 시뮬레이션된 사용자는 1,280개 프로세스에 균등하게 분산되었습니다.

각 end-to-end 사용자 프로세스의 API 테스트 단계:

1. 요청 ID를 `assign_queue_num` 호출하고 수신합니다.
2. 요청 ID가 사용자의 대기열 위치를 반환할 때까지 반복 `queue_num` 재생합니다 (짧은 시간).
3. 반환된 값이 >= 사용자 대기열 위치 (장시간) 가 `serving_num` 될 때까지 반복합니다.
4. 대기 중인 사용자 수를 `waiting_room_size` 검색하기 위해 가끔 호출합니다.
5. 대상 사이트에서 사용할 JWT를 `generate_token` 호출하고 수신하십시오.

조사 결과

대기실을 통해 처리할 수 있는 클라이언트 수에는 실질적인 상한선이 없습니다.

사용자가 대기실에 입장하는 속도는 배포된 지역의 Lambda 함수 동시 실행 할당량에 영향을 줍니다.

에서 사용한 CloudFront 캐싱 정책으로는 부하 테스트에서 기본 API Gateway 요청 제한인 초당 10,000개의 요청을 초과할 수 없었습니다.

`get_queue_num`Lambda 함수의 호출 비율은 대기실로 들어오는 사용자의 비율과 거의 1:1 에 가깝습니다. 이 Lambda 함수는 동시 실행 제한 또는 버스트 제한으로 인해 수신 사용자 수가 높은 경우 병목 현상이 발생할 수 있습니다. 다수의 Lambda 함수 호출로 인한 스로틀링은 부작용으로 다른 `get_queue_num` Lambda 함수에 영향을 미칠 수 있습니다. 클라이언트 소프트웨어가 재시도/백오프 로직으로 이러한 유형의 임시 규모 조정 오류에 적절하게 대응할 수 있다면 전체 시스템은 계속 운영됩니다.

기본 할당량 구성의 코어 스택으로 구성된 CloudFront 배포는 각 사용자가 1초에 한 번씩 API를 폴링하는 250,000명의 사용자를 수용하는 대기실을 처리할 수 있습니다. `serving_num`

문제 해결

이 섹션에서는 이 솔루션에 대한 문제 해결 정보를 제공합니다.

이 섹션에서 문제를 해결할 수 없는 경우 AWS Support에 문의하여 이 솔루션에 대한 AWS Support 사례를 여는 방법을 안내해 드립니다.

API의 4xx 응답 상태

- 이는 잘못된 이벤트 ID나 요청 ID 또는 둘 다로 인해 발생할 수 있습니다. 이는 관련 Lambda 함수의 CloudWatch 로그에서 발생합니다.
- 프라이빗 API는 IAM 인증을 거쳤으며 클라이언트에는 프라이빗 API를 호출할 권한이 있는 AWS 키가 필요합니다. 이는 API Gateway의 CloudWatch 로그에서 발생합니다.

API의 5xx 응답 상태

- 병목 현상이 발생한 Lambda 또는 API Gateway의 응답, 알람 확인 *<LambdaFunctionName>*ThrottlesAlarm CloudWatch
- 백엔드의 구성이 잘못되었습니다. 자세한 내용은 경보 및 로그를 확인하십시오. *<LambdaFunctionName>*ErrorsAlarm CloudWatch CloudWatch

ErrorPublic5XX/PrivateApiAlarm

- 이 알람 상태는 API가 60초 이내에 5XX 상태를 호출자에게 반환하는 ALARM 경우입니다.
- 이 경보는 60초 동안 5xx 상태가 반환되지 않을 OK 때 반환됩니다.
- 이 경보는 Lambda 함수 또는 Lambda 런타임에서 API Gateway에 오류를 반환하여 시작할 수 있습니다.

4XX/ErrorPublicPrivateApiAlarm

- 이 알람 상태는 API가 60초 이내에 4XX 상태를 호출자에게 반환하는 ALARM 경우입니다.
- 이 경보는 4XX 상태가 60초 동안 반환되는 시점으로 돌아갑니다OK.
- 이 경보는 잘못된 API URL로 시작될 수 있습니다.

*<LambdaFunctionName>*ThrottlesAlarm

- 명명된 Lambda가 60초 내에 동시 실행 한도에 도달하면 이 경보 상태는 ALARM입니다.
- 60초 동안 병목 현상이 발생하지 OK 않으면 이 경보가 반환됩니다.
- 계정 지역의 동시성 한도를 늘려야 할 수도 있습니다.
- Lambda의 버스트 한도가 발생할 수 있으며, 이 경우 클라이언트에서 일부 재시도 로직이 필요합니다.

*<LambdaFunctionName>*ErrorsAlarm

- 이 경보 상태는 명명된 Lambda에서 60초 이내에 런타임 실행 오류가 발생하는 ALARM 경우입니다.
- 60초 동안 오류가 발생하지 OK 않으면 이 경보가 반환됩니다.
- 이는 백엔드의 잘못된 컨피그레이션으로 인해 발생할 수 있습니다.
- 이는 Lambda 코드의 버그로 인해 발생할 수 있습니다.

연락처 AWS Support

AWS 개발자 지원, AWS 비즈니스 지원 또는 AWS Enterprise Support를 보유한 경우 지원 센터를 사용하여 이 솔루션에 대한 전문가 지원을 받을 수 있습니다. 이후 단원에서는 그 방법에 대해서 설명합니다.

사례 생성

1. Support 센터에 로그인하세요.
2. 사례 생성을 선택합니다.

어떻게 도와드릴까요?

1. 테크니컬을 선택하세요.
2. 서비스에서 솔루션을 선택합니다.
3. 카테고리에서 기타 솔루션을 선택합니다.
4. 심각도에서는 사용 사례에 가장 적합한 옵션을 선택합니다.
5. 서비스, 범주 및 심각도를 입력하면 인터페이스에 일반적인 문제 해결 질문으로 연결되는 링크가 채워집니다. 이러한 링크로 질문을 해결할 수 없는 경우 다음 단계: 추가 정보를 선택하십시오.

추가 정보

1. 제목에 질문이나 문제를 요약한 텍스트를 입력합니다.
2. 설명에는 문제를 자세히 설명하세요.
3. 파일 첨부를 선택합니다.
4. 요청을 처리하는 데 AWS Support 필요한 정보를 첨부합니다.

케이스를 더 빨리 해결할 수 있도록 도와주세요

1. 요청된 정보를 입력합니다.
2. 다음 단계: 지금 해결하거나 문의하기를 선택합니다.

지금 해결하거나 저희에게 문의하세요

1. 지금 해결 솔루션을 검토하세요.
2. 이러한 해결 방법으로 문제를 해결할 수 없는 경우 문의하기를 선택하고 요청된 정보를 입력한 다음 제출을 선택합니다.

추가적인 리소스

AWS 서비스	
• AWS CloudFormation	• Amazon DynamoDB
• Amazon Simple Storage Service(S3)	• Amazon API Gateway
• AWS Lambda	• AWS 보안 관리자
• 아마존 CloudFront	• Amazon Simple Queue Service
• 아마존 EventBridge	• 아마존 CloudWatch
• 아마존 포 ElastiCache 레디스용	• Amazon Comprehend
• Amazon Virtual Private Cloud	• AWS Identity and Access Management

솔루션 제거

OR에서 를 사용하여 가상 대기실 ON AWS 솔루션을 제거할 수 AWS Management Console 있습니다. AWS Command Line Interface이 솔루션에서 생성된 다양한 리소스의 로그를 저장하는 데 사용되는 S3 버킷을 수동으로 삭제해야 합니다. AWS 솔루션 구현에서는 이러한 S3 버킷을 자동으로 삭제하지 않으므로 솔루션이 삭제된 후에도 로그 이벤트를 검토할 수 있습니다.

솔루션에서 생성한 IAM 사용자 그룹에 IAM 사용자를 수동으로 추가한 경우 솔루션을 제거하기 전에 IAM 사용자 그룹에서 IAM 사용자를 제거하십시오. `ProtectedAPIGroup` 그렇지 않으면 IAM 사용자 그룹 및 관련 IAM 정책이 삭제되지 않습니다.

배포된 각 스택에 대해 아래 지침을 따르십시오.

사용 AWS Management Console

1. AWS CloudFormation 콘솔에 로그인합니다.
2. 스택 페이지에서 이 솔루션의 설치 스택을 선택합니다.
3. 삭제를 선택합니다.

사용 AWS Command Line Interface

사용자 환경에서 AWS Command Line Interface (AWS CLI) 를 사용할 수 있는지 확인하십시오. 설치 지침은 무엇입니까를 참조하십시오. AWS Command Line Interface AWS CLI 사용 설명서에서. 를 사용할 수 AWS CLI 있는지 확인한 후 다음 명령을 실행합니다.

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

Amazon S3 버킷 삭제

이 솔루션은 우발적인 데이터 손실을 방지하기 위해 AWS CloudFormation 스택을 삭제하기로 결정한 경우 솔루션에서 생성한 Amazon S3 버킷 (옵트인 지역에 배포용) 을 보존하도록 구성되어 있습니다. 솔루션을 제거한 후 데이터를 보존할 필요가 없는 경우 이 S3 버킷을 수동으로 삭제할 수 있습니다. 다음 단계에 따라 Amazon S3 버킷을 삭제합니다.

1. Amazon S3 콘솔에 로그인합니다.

2. 왼쪽 탐색 창에서 버킷을 선택합니다.
3. *<stack-name>* S3 버킷을 찾습니다.
4. S3 버킷을 선택하고 삭제를 선택합니다.

를 사용하여 AWS CLI S3 버킷을 삭제하려면 다음 명령을 실행합니다.

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

소스 코드

GitHub리포지토리를 방문하여 이 솔루션의 소스 파일을 다운로드하고 사용자 지정 내용을 다른 사람과 공유하십시오.

기여자

- 짐 타리오
- 티아그 라마찬드란
- 조안 모건
- 저스틴 퍼틀
- 앨런 모헤이마니
- 가빗 싱
- 바셈 와니스

개정

날짜	변경 사항
2021년 11월	최초 릴리스
2022년 9월	버전 1.1: 만료된 대기열 위치에 따라 서빙 카운터가 자동으로 증가합니다. 일부 Redis 사용을 DynamoDB로 재배치합니다. 남은 대기열 위치 만료 시간을 확인하기 위한 공개 API 엔드포인트 자세한 내용은 리포지토리의 ChangeLog.md 파일을 참조하십시오. GitHub
2023년 4월	버전 1.1.1: 모든 새 S3 버킷의 S3 객체 소유권(ACL 비활성화) 에 대한 새로운 기본 설정으로 인한 영향이 완화되었습니다. 자세한 내용은 리포지토리의 ChangeLog.md 파일을 참조하십시오. GitHub
2023년 11월	버전 1.1.2: 보안 취약성을 해결하기 위해 패키지 버전이 업데이트되었습니다. 자세한 내용은 저장소의 ChangeLog.md 파일을 참조하십시오. GitHub
2024년 3월	버전 1.1.3: 대기실 크기에서 대기열 위치가 만료된 상태로 유지되는 문제, 재설정 후에도 이전 결과를 반환하는 queue_num API, OpenID 어댑터 API의 간헐적인 오류 등 세 가지 문제를 해결했습니다. /userInfo 자세한 내용은 저장소의 ChangeLog.md 파일을 참조하십시오. GitHub
2024년 4월	버전 1.1.4: 보안 취약성을 해결하기 위해 패키지 버전이 업데이트되었습니다. 자세한 내용은 저장소의 ChangeLog.md 파일을 참조하십시오. GitHub

날짜	변경 사항
2024년 6월	버전 1.1.5: 보안 취약성을 해결하기 위해 패키지 버전이 업데이트되었습니다. 자세한 내용은 저장소의 ChangeLog.md 파일을 참조하십시오. GitHub

고지 사항

고객은 본 문서의 정보를 독립적으로 평가할 책임이 있습니다. 이 문서는 (a) 정보 제공만을 목적으로 하고, (b) 사전 통지 없이 변경될 수 있는 AWS 현재의 제품 제안 및 관행을 나타내며, (c) 해당 계열사, 공급업체 또는 라이선스 제공자로부터 AWS 어떠한 약정이나 보증도 하지 않습니다. AWS 제품 또는 서비스는 명시적이든 묵시적이든 어떠한 종류의 보증, 진술 또는 조건 없이 “있는 그대로” 제공됩니다. AWS 고객에 대한 책임과 책임은 AWS 계약에 의해 통제되며, 본 문서는 고객과 체결한 계약의 일부가 아니며 수정하지도 않습니다. AWS

가상 대기실 AWS 온은 Apache 라이선스 버전 2.0의 조건에 따라 라이선스가 부여됩니다.

AWS 용어집

최신 AWS 용어는 참조의 AWS 용어집을 참조하십시오.AWS 용어집

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.