Guida all'implementazione

# Sala d'attesa virtuale su AWS

# Sala d'attesa virtuale su AWS: Guida all'implementazione

# Table of Contents

# Assorbi grandi picchi di traffico verso il tuo sito web con la sala d'attesa virtuale attiva AWS

Data di pubblicazione: novembre 2021 (ultimo aggiornamento: novembre 2024)

La AWS soluzione Virtual Waiting Room on aiuta a controllare le richieste degli utenti in arrivo sul tuo sito Web durante grandi picchi di traffico. Crea un'infrastruttura cloud progettata per scaricare temporaneamente il traffico in entrata sul tuo sito Web e offre opzioni per personalizzare e integrare una sala d'attesa virtuale. Questa soluzione può essere integrata con siti Web nuovi o esistenti per scalare senza problemi e gestire improvvisi picchi di traffico.

Esempi di eventi su larga scala che potrebbero generare un aumento del traffico del sito Web includono:

- Inizio della vendita di biglietti per concerti o eventi sportivi
- Vendita antincendio o altra grande vendita al dettaglio, come il Black Friday
- Lancio di un nuovo prodotto con ampi annunci di marketing
- Accesso agli esami e frequenza alle lezioni per test e lezioni online
- Rilascio di posti per appuntamenti medici
- Lancio di un nuovo direct-to-customer servizio che richiede la creazione di un account e i pagamenti

La soluzione funge da area di attesa per i visitatori del sito Web e consente il passaggio del traffico quando la capacità è sufficiente. Il software client utilizzato dai visitatori può essere configurato per consentire in modo trasparente al traffico di attraversare la sala d'attesa fino al raggiungimento della capacità massima del sito Web; a quel punto la sala d'attesa trattiene i visitatori. Quando il sito Web è in grado di generare più traffico, la soluzione genera JSONWeb Token (JWT) che consentono agli utenti di accedere al sito Web. Ad esempio, se avete un evento che dura due ore e il vostro sito Web può elaborare 50 utenti al secondo, ma vi aspettate un volume di 250 al secondo, potete utilizzare questa soluzione per regolare il traffico e consentire agli utenti di mantenere la propria posizione in coda.

Questa soluzione offre le seguenti funzionalità chiave:

- Accodamento strutturato degli utenti sul tuo sito web

- Scalabilità per controllare il traffico per eventi di dimensioni molto grandi
- JSONgenerazione di token web per consentire l'accesso al sito di destinazione
- Tutte le funzionalità sono controllate tramite REST APIs
- Autorizzatore API Gateway chiavi in mano per soluzioni client
- Integrazione autonoma o utilizzo con OpenID

Questa guida all'implementazione descrive le considerazioni architettoniche e i passaggi di configurazione per la distribuzione di Virtual Waiting Room AWS nel cloud Amazon Web Services (AWS). Include collegamenti a AWS CloudFormationmodelli che avviano e configurano i AWS servizi necessari per implementare questa soluzione utilizzando le AWS migliori pratiche di sicurezza e disponibilità.

La guida è destinata ad architetti IT, sviluppatori, DevOps personale, analisti di dati e professionisti delle tecnologie di marketing che hanno esperienza pratica nell'architettura nel cloud. AWS

# Costo

L'utente è responsabile del costo dei AWS servizi utilizzati durante l'esecuzione di questa soluzione. A partire da questa revisione, il costo per l'esecuzione di questa soluzione con le impostazioni predefinite nella regione Stati Uniti orientali (Virginia settentrionale) è di circa 10,00 USD al giorno per stack più i costi per le richieste API e il traffico dati relativi alle dimensioni dell'evento.

## Costo giornaliero per la manutenzione della soluzione senza eventi

| AWS service | Richieste/Orario | Costo [USD] |
|---|---|---|
| Amazon API Gateway | 0 | $0,00 |
| Amazon CloudFront | 0 | $0,00 |
| Amazon CloudWatch | 0 | $0,00 |
| Amazon DynamoDB | 0 | $0,00 |
| Amazon ElastiCache | Ore dei nodi di calcolo (Redis) | ~$6,00 |
| AWS Lambda | Livello gratuito* | $0,00 |
| AWS Secrets Manager | Livello gratuito* | $0,00 |
| Amazon Simple Storage Service (Amazon S3) | Livello gratuito* | $0,00 |
| Amazon Virtual Private Cloud (Amazon VPC) (Amazon VPC) | Orari degli endpoint VPC<br>Orari del gateway NAT | ~$5,00 |
| TOTALE: | | ~$11,00 |

*La stima dei costi si basa su un ambiente pulito. Se utilizzi questo servizio AWS al di fuori di questa soluzione, potresti superare la quota del piano gratuito.

Le tabelle seguenti mostrano i costi stimati per una sala d'attesa da 50.000 utenti e una da 100.000 utenti con una durata dell'evento compresa tra 2 e 4 ore con 500 utenti/secondo in entrata e 1.000

utenti/min in uscita. I prezzi sono soggetti a modifiche. Per tutti i dettagli, consulta la pagina web dei prezzi di ogni servizio utilizzato in questa soluzione. AWS

## Costo stimato per 50.000 utenti della sala d'attesa durante un evento di 2 ore

| AWS service | Dimensioni | Costo [USD] |
|---|---|---|
| Amazon API Gateway | Richieste | $2,00 |
| CloudFront | Richieste, larghezza di banda | $75,00 |
| CloudWatch | Metriche, allarmi, archiviazione | $1,00 |
| CloudWatch Eventi Amazon | Eventi | $1,00 |
| DynamoDB | Unità di lettura/scrittura, archiviazione | $1,00 |
| ElastiCache | Ore per nodo | $8,00 |
| Lambda | Richieste, tempo di calcolo | $1,00 |
| AWS Secrets Manager | Segreti, richieste | $1,00 |
| Amazon S3 | Richieste, archiviazione | $1,00 |
| Amazon VPC | Trasferimento dati, ora dell'endpoint | $2,00 |
| TOTALE | | $94,00 |

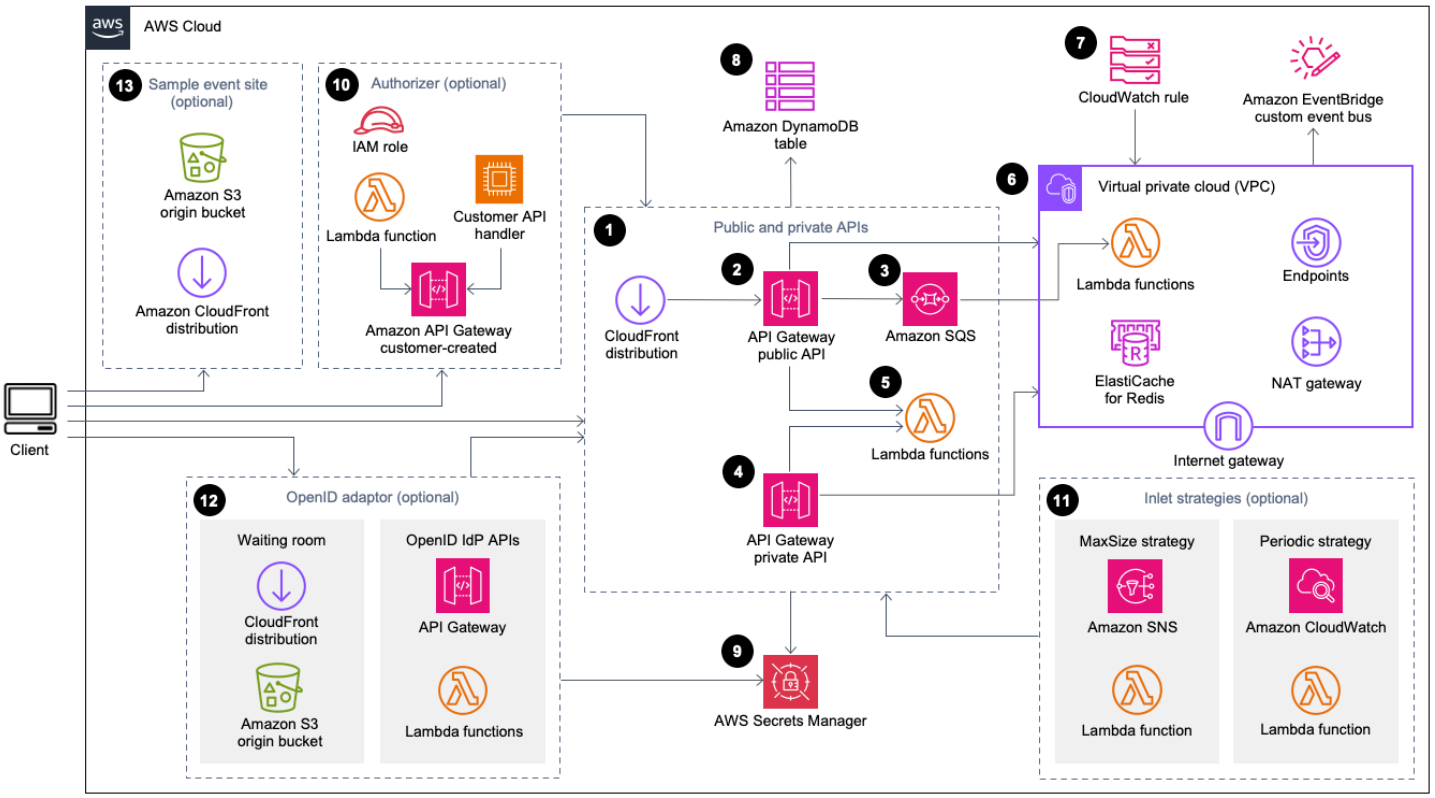
## Costo stimato per 100.000 utenti della sala d'attesa durante un evento di 2 ore

| AWS service | Dimensioni | Costo [USD] |
|---|---|---|

| | | |
|---|---|---|
| Amazon API Gateway | Richieste | $4,00 |
| CloudFront | Richieste, larghezza di banda | $296,00 |
| CloudWatch | Metriche, allarmi, archiviazione | $1,00 |
| CloudWatch Eventi | Eventi | $1,00 |
| DynamoDB | Unità di lettura/scrittura, archiviazione | $4,00 |
| ElastiCache | Ore per nodo | $32,00 |
| Lambda | Richieste, tempo di calcolo | $1,00 |
| AWS Secrets Manager | Segreti, richieste | $1,00 |
| Amazon Simple Queue Service (Amazon SQS) | Richieste | $1,00 |
| Amazon S3 | Richieste, archiviazione | $1,00 |
| Amazon VPC | Trasferimento dati, ora dell'endpoint | $6,00 |
| TOTALE | | 348,00$ |

# Panoramica dell'architettura

L'implementazione di questa soluzione con i modelli richiesti e opzionali, utilizzando i parametri predefiniti, crea il seguente ambiente nel cloud. AWS




Sala d'attesa virtuale sull'architettura AWS

I AWS CloudFormation modelli implementano la seguente infrastruttura:

1. Una CloudFront distribuzione Amazon per fornire API chiamate pubbliche per il cliente.
2. APIRisorse pubbliche di Amazon API Gateway per elaborare le richieste di coda dalla sala d'attesa virtuale, tracciare la posizione della coda e supportare la convalida dei token che consentono l'accesso al sito Web di destinazione.
3. Una coda Amazon Simple Queue Service (AmazonSQS) per regolare il traffico verso la AWS Lambdafunzione che elabora i messaggi in coda. Invece di richiamare la funzione Lambda per ogni richiesta, la coda raggruppa in batch SQS le raffiche di richieste in entrata.
4. APIRisorse private API Gateway per supportare le funzioni amministrative.
5. Funzioni Lambda per convalidare ed elaborare le API richieste pubbliche e private e restituire le risposte appropriate.

6. Amazon Virtual Private Cloud (VPC) per ospitare le funzioni Lambda che interagiscono direttamente con il cluster Elasticache (Redis). OSS VPCgli endpoint consentono alle funzioni Lambda di comunicare con VPC i servizi all'interno della soluzione. Inoltre, il NAT gateway consente alle funzioni Lambda di connettere gli VPC CloudFront endpoint e invalidare la cache come richiesto.
7. Una CloudWatch regola Amazon per richiamare una funzione Lambda che funziona con un bus EventBridgeAmazon personalizzato per trasmettere periodicamente aggiornamenti di stato.
8. Tabelle Amazon DynamoDB per archiviare token, posizione della coda e server dei dati del contatore.
9. AWS Secrets Manager per archiviare le chiavi per le operazioni con i token e altri dati sensibili.
10. (Facoltativo) Componente di autorizzazione costituito da un ruolo AWS Identity and Access Management(IAM) e una funzione di autorizzazione Lambda da utilizzare con Gateway. API
11. (Facoltativo) Amazon Simple Notification Service (AmazonSNS) e funzioni Lambda per supportare due strategie di ingresso. CloudWatch
12. (Opzionale) Componente adattatore OpenID con funzioni Gateway API e Lambda per consentire a un provider OpenID di autenticare gli utenti sul tuo sito web. CloudFront distribuzione con un bucket Amazon Simple Storage Service (Amazon S3) per la pagina della sala d'attesa per questo componente.
13. (Facoltativo) CloudFront distribuzione con bucket di origine Amazon S3 per l'applicazione Web di esempio per la sala d'attesa.

# Come funziona la soluzione

Questa sezione descrive le fasi di un flusso di lavoro di AWS Virtual Waiting Room ad alto livello. Consulta la Guida per gli sviluppatori GitHub per maggiori dettagli sulla creazione, la personalizzazione e l'integrazione di una sala d'attesa per il tuo sito web.

La sala d'attesa pubblica API può trovarsi dietro il sistema di sicurezza perimetrale del sito oppure può essere disponibile senza alcuna autorizzazione. A seconda dell'approccio utilizzato per integrare la sala d'attesa con il sito Web, all'utente potrebbe essere richiesto di autenticarsi prima sul sito Web prima di poter accedere alla sala d'attesa e ottenere una posizione in coda.

Il software client deve disporre dell'Event ID per entrare nella sala d'attesa ed effettuare altre richieste. Un Event ID è un ID univoco richiesto per la maggior parte delle richieste rivolte al pubblico e al privatoAPIs. L'ID evento viene impostato durante l'installazione dello API stack principale. Durante il funzionamento, l'Event ID può essere fornito come URL parametro o cookie tramite la pagina della sala d'attesa; può essere fornito come parte delle richieste di token di autenticazione o può essere distribuito ai client attraverso un percorso dati diverso.

In alcuni casi il client necessita sia dell'ID evento che dell'ID richiesta per effettuare determinate API chiamate. Il Request ID è un ID univoco rilasciato dalla sala d'attesa che rappresenta uno specifico cliente in fila.

I passaggi seguenti descrivono il flusso di API richieste di ingresso in coda, l'attesa che la coda proceda e l'uscita dalla sala d'attesa con un token di accesso al sito Web.

L'utente entra nella sala d'attesa:

1. All'utente viene presentata una schermata o una pagina che rappresenta il punto di ingresso della sala d'attesa. Scelgono di entrare in coda e il software client (browser, dispositivo mobile, dispositivo) chiama il `assign_queue_num` pubblico API per richiedere una posizione in coda.
2. La API richiesta viene immediatamente consegnata alla SQS coda Amazon da API Gateway.
3. La `assign_queue_num` API chiamata ritorna quando la richiesta viene inserita nella coda. Il client riceve un ID di richiesta univoco che può essere utilizzato in seguito per recuperare la posizione della coda, l'ora della richiesta e un token di accesso.
4. La funzione `AssignQueueNum` Lambda riceve batch composti da un massimo di dieci richieste dalla coda. SQS Il servizio Lambda suddivide le chiamate per elaborare più batch di richieste.

5. La funzione `AssignQueueNum` Lambda convalida ogni messaggio nel relativo batch, incrementa il contatore di coda in Elasticache (Redis) e archivia ogni richiesta in Elasticache (RedisOSS) con la posizione di coda associata. OSS
6. Ogni messaggio viene eliminato man mano che viene elaborato correttamente. I messaggi relativi a una condizione di errore vengono rielaborati una volta in un batch successivo. Dopo un secondo errore, vengono inviati a un dead-letter-queue dispositivo collegato a un CloudWatchallarme.
7. Il client può iniziare il polling `queue_num` API dopo aver ricevuto l'ID della richiesta dalla `assign_queue_num` chiamata. Il client invia l'ID evento e l'ID della richiesta `queue_num` API e riceve una posizione numerica in coda o una risposta che indica che la richiesta non è stata ancora elaborata. Il client potrebbe dover effettuare questa chiamata più di una volta durante eventi di grandi dimensioni. La funzione `GetQueueNum` Lambda viene richiamata da API Gateway e restituisce la posizione numerica del client nella coda da DynamoDB.

L'utente attende nella sala d'attesa:

8. Dopo che il cliente ha raggiunto la sua posizione in coda, può iniziare a fare il polling `serving_num` API a intervalli regolari. `serving_num`APIviene chiamato con l'ID evento e restituisce la posizione di servizio corrente della coda. La risposta di `serving_num` API indica al cliente quando può spostarsi dalla sala d'attesa al sito di destinazione effettivo dove può avvenire la transazione finale. La funzione `GetServingNum` Lambda restituisce l'attuale posizione di servizio della sala d'attesa.
9. Quando la posizione di servizio è uguale o superiore alla posizione in coda (richiesta) del client, il client può richiedere un JSON Web Token (JWT) al pubblico. API Il token può essere utilizzato con il sito di destinazione per finalizzare la transazione. `generate_token`APIViene chiamato con l'ID evento e l'ID della richiesta. APIGateway richiama la funzione `GenerateToken` Lambda con i parametri.
10. La funzione `GenerateToken` Lambda convalida la richiesta e verifica se questo token è stato generato in precedenza. La funzione Lambda interroga la tabella DynamoDB per trovare un token corrispondente. Se trovato, quel token viene restituito al chiamante e non viene rigenerato. Questo processo impedisce l'utilizzo di un singolo ID di richiesta per generare più token diversi con nuovi tempi di scadenza.
11. Se il token non viene trovato in DynamoDB, la funzione Lambda recupera le chiavi per creare il token e salva il token in DynamoDB con l'ID evento e l'ID di richiesta del client. La funzione Lambda scrive un evento su per EventBridge segnalare che è stato generato un nuovo token. La

funzione Lambda incrementa un contatore Elasticache (RedisOSS) che tiene traccia del numero di token generati per l'evento.

12. Se `queue_pos_expiry` è attivata, il client può interrogare il tempo rimanente prima della scadenza chiamando la funzione `queue_pos_expiry` API che richiama la funzione `GetQueuePositionExpiryTime` Lambda.

L'utente lascia la sala d'attesa:

13. Quando il client riceve il token, entra nel sito di destinazione per iniziare la transazione. A seconda del modo in cui l'infrastruttura supporta l'integrazioneJWT, il client potrebbe dover presentare il token in un'intestazione di richiesta, un cookie o in un altro modo. L'authorizer per API Gateway può essere utilizzato per convalidare il token incluso nella richiesta di un client. Qualsiasi libreria commerciale o open source per la convalida e la gestione JWTs può essere utilizzata con Virtual Waiting Room sui token. AWS Se il token è valido, il cliente può continuare la transazione.
14. Dopo che il cliente ha completato la transazione, API viene chiamato un privato per aggiornare lo stato del token del client e questa operazione viene completata in DynamoDB.

Scadenza della posizione in coda:

15. Quando questa funzione è attivata, l'ID di richiesta corrispondente a una particolare posizione in coda è idoneo a generare un token solo per un intervallo di tempo specificato.

Incrementa il contatore di servitù alla scadenza della posizione di coda:

16. Quando questa funzione è attivata, il contatore di servizio viene automaticamente incrementato in base alle posizioni di coda scadute che non sono state in grado di generare token.

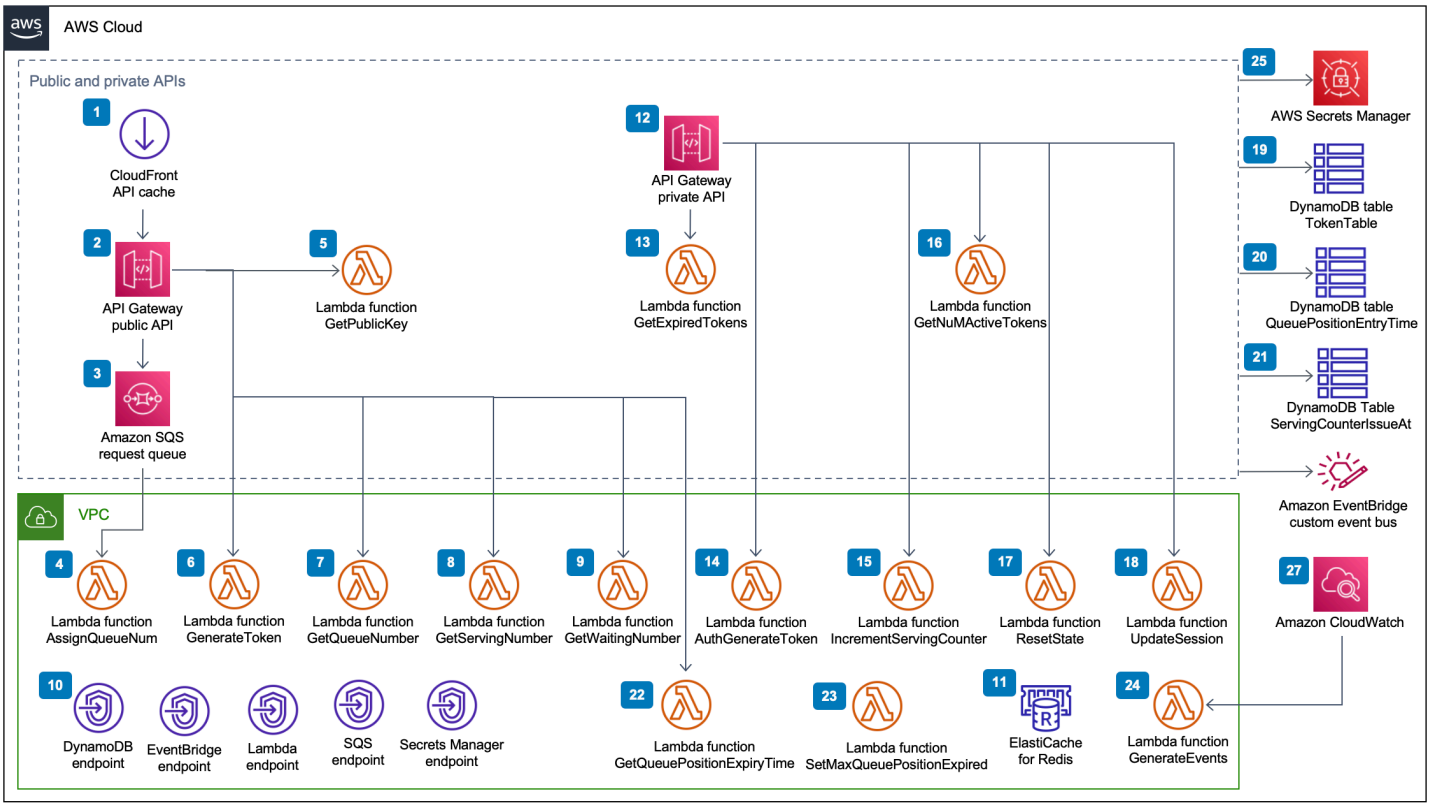
# Componenti della soluzione

## Sala d'attesa pubblica e privata APIs

Lo scopo principale della AWS soluzione Virtual Waiting Room on è controllare la generazione di JSON Web Token (JWT) per i client in modo controllato per evitare esplosioni di nuovi utenti che potrebbero sovraccaricare il sito Web di destinazione. JWTsPuò essere utilizzato per la protezione del sito, impedendo l'accesso alle pagine Web fino all'ottenimento del token della sala d'attesa e anche per API l'autorizzazione all'accesso.

Il modello principale installa un sistema pubblico API e privato (IAM-autorizzato) API utilizzato per la maggior parte delle operazioni di Virtual Waiting Room. AWS Il pubblico API è configurato con una CloudFront distribuzione con più politiche di memorizzazione nella cache in base al percorso. API Vengono creati una tabella DynamoDB EventBridge e un bus eventi. Il modello ne aggiunge una nuova VPC con due zone di disponibilità (AZs), un cluster Elasticache (RedisOSS) in entrambe AZs e diverse funzioni Lambda. Le funzioni Lambda che interagiscono con Elasticache (RedisOSS) dispongono di interfacce di rete all'interno di e VPC tutte le altre funzioni Lambda hanno una connettività di rete predefinita. Il nucleo APIs è il livello più basso di interazione con la soluzione. Altre funzioni Lambda, l'istanza Amazon Elastic Compute Cloud (AmazonEC2) e i contenitori possono fungere da estensioni e richiamare il core APIs per creare sale d'attesa, controllare il traffico in ingresso e reagire agli eventi generati dalla soluzione.

Inoltre, lo stack principale genera un allarme per tutti gli errori di funzione Lambda e le condizioni di accelerazione, nonché allarmi per API ogni implementazione del Gateway per i codici di stato 4XX e 5XX.




Sala d'attesa virtuale su AWS componente pubblico e privato APIs

1. CloudFront la distribuzione fornisce API chiamate pubbliche per il client e memorizza nella cache i risultati laddove appropriato.
2. Amazon API Gateway API elabora pubblicamente le richieste di coda dalla sala d'attesa virtuale, monitora la posizione della coda e supporta la convalida dei token che consentono l'accesso al sito Web di destinazione.
3. SQSqueue regola il traffico verso la AWS Lambda funzione che elabora i messaggi in coda.
4. La funzione `AssignQueueNum` Lambda convalida ogni messaggio ricevuto nel batch, incrementa il contatore di coda in Elasticache (Redis) e archivia ogni richiesta in Elasticache (RedisOSS) con la posizione di coda associata. OSS
5. La funzione `GetPublicKey` Lambda recupera il valore della chiave pubblica da Secrets Manager.
6. La funzione `GenerateToken` Lambda genera un messaggio JWT per una richiesta valida a cui è stato consentito di completare la transazione nel sito di destinazione. Scrive un evento nel bus degli eventi personalizzato della sala d'attesa che indica che è stato generato un token. Se in precedenza è stato generato un token per questa richiesta, non viene generato alcun nuovo token.

7. La funzione `GetQueueNumber` Lambda recupera e restituisce la posizione numerica del client nella coda da Elasticache (Redis). OSS
8. La funzione `GetServingNumber` Lambda recupera e restituisce il numero attualmente servito dalla sala d'attesa da Elasticache (Redis). OSS
9. La funzione `GetWaitingNum` Lambda restituisce il numero attualmente in coda nella sala d'attesa e non ha ancora ricevuto un token.
10. VPCgli endpoint consentono alle funzioni Lambda di comunicare con VPC i servizi all'interno della soluzione.
11. Il cluster Elasticache (RedisOSS) archivia tutte le richieste di accesso alla sala d'attesa con un ID evento valido. Memorizza inoltre diversi contatori come il numero di richieste in coda, il numero attualmente servito, il numero di token generati, il numero di sessioni completate e il numero di sessioni abbandonate.
12. APIAPIRisorse private Gateway per supportare le funzioni amministrative. I privati APIs sono AWS IAM autenticati.
13. La funzione `GetExpiredTokens` Lambda restituisce un elenco di richieste IDs con token scaduti.
14. La funzione `AuthGenerateToken` Lambda genera un token per una richiesta valida a cui è stato consentito di completare la transazione nel sito di destinazione. L'emittente e il periodo di validità di un token inizialmente impostati durante l'implementazione dello stack principale possono essere ignorati. Scrive un evento nel bus degli eventi personalizzato della sala d'attesa che indica che è stato generato un token. Se il token è stato precedentemente generato per questa richiesta, non viene generato alcun nuovo token.
15. La funzione `IncrementServingCounter` Lambda incrementa il bancone di servizio della sala d'attesa memorizzato in Elasticache (RedisOSS) con un incremento in base al valore.
16. La funzione `GetNumActiveTokens` Lambda interroga DynamoDB per il numero di token che devono ancora scadere, non sono stati utilizzati per completare la transazione e non sono stati contrassegnati come abbandonati.
17. La funzione `ResetState` Lambda reimposta tutti i contatori memorizzati in Elasticache (Redis). OSS Inoltre, elimina e ricrea le tabelle `TokenTableQueuePositionEntryTime`, e DynamoDB`ServingCounterIssuedAt`. Inoltre, esegue l'invalidazione della cache. CloudFront
18. La funzione `UpdateSession` Lambda aggiorna lo stato di una sessione (token) memorizzata nella tabella DynamoDB`TokenTable`. Lo stato della sessione è indicato da un numero intero. Le sessioni impostate su uno stato di 1 indicano completate e `-1` indicano abbandonate. Scrive un evento nel bus degli eventi personalizzato della sala d'attesa indicante che una sessione è stata aggiornata.

19 La tabella `TokenTable` DynamoDB memorizza i dati dei token.

20 La tabella `QueuePositionEntryTime` DynamoDB memorizza i dati sulla posizione della coda e sull'ora di immissione.

21 La tabella `ServingCounterIssuedAt` DynamoDB memorizza gli aggiornamenti del contatore di servizio.

22 La funzione `GetQueuePositionExpireTime` Lambda viene richiamata quando il client richiede il tempo di scadenza della posizione di coda rimanente.

23 La funzione `SetMaxQueuePositionExpired` Lambda imposta la posizione massima della coda scaduta corrispondente ai valori della tabella. `ServingCounterIssuedAt` Viene eseguita ogni minuto se il `IncrSvcOnQueuePositionExpiry` parametro è impostato su `true` durante la distribuzione dello stack principale.

24 La funzione `GenerateEvents` Lambda scrive diverse metriche della sala d'attesa nel bus eventi personalizzato della sala d'attesa. Viene eseguita ogni minuto se il parametro Enable Events Generation è impostato su `true` durante l'implementazione dello stack principale.

25 AWS Secrets Manager archivia le chiavi per le operazioni con i token e altri dati sensibili.

26 Amazon EventBridge Custom Event Bus riceve un evento ogni volta che viene generato un token e una sessione viene aggiornata nella tabella `TokenTable` DynamoDB. Riceve anche eventi quando il bancone di servizio viene spostato nella `SetMaxQueuePositionExpired` Lambda. Viene scritto con varie metriche relative alla sala d'attesa, se attivato durante l'implementazione del core stack.

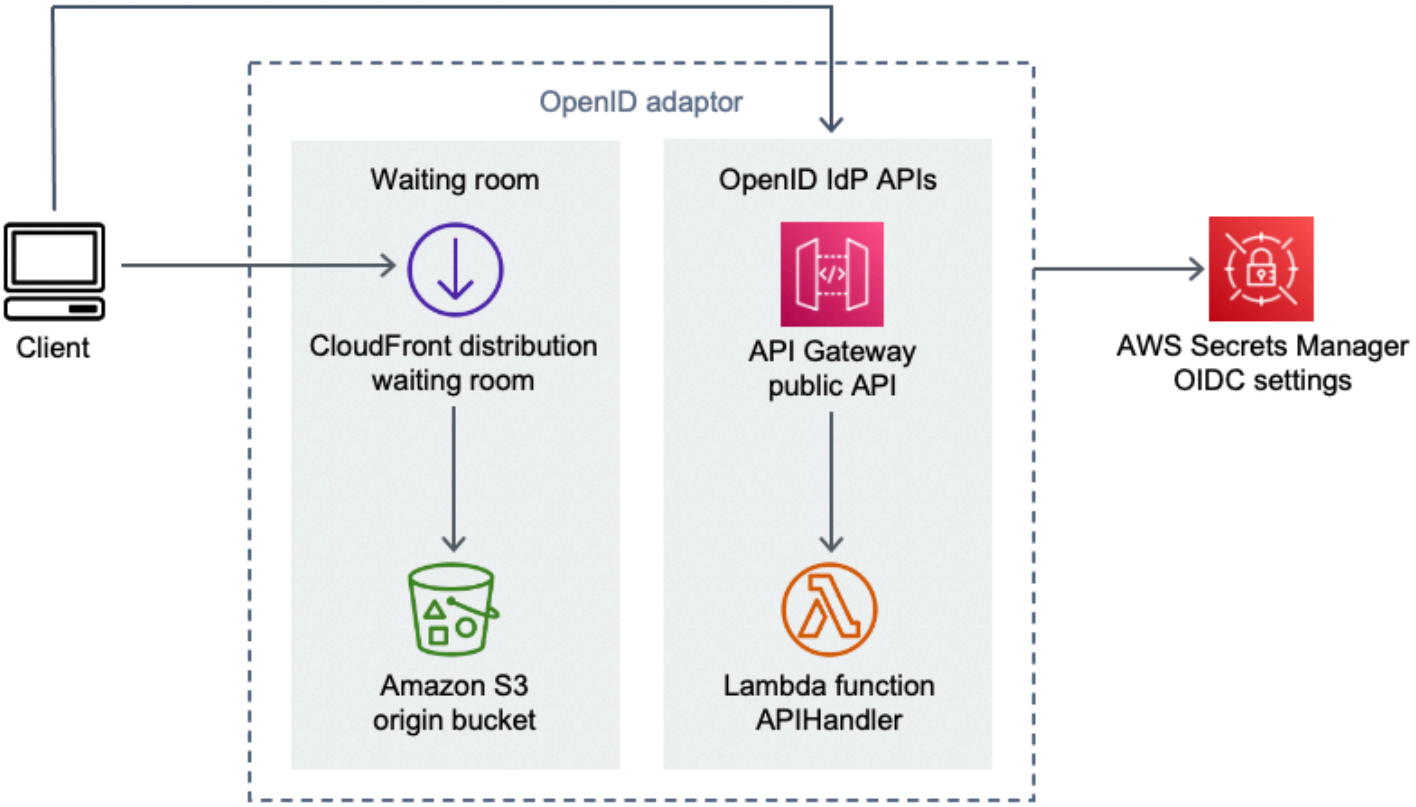
27 La regola CloudWatch degli eventi Amazon viene creata se il parametro Enable Events Generation è impostato su true durante la distribuzione dello stack principale. Questa regola evento avvia la funzione `GenerateEvents` Lambda ogni minuto.

# Authorizers

La soluzione include uno stack di autorizzatori API Gateway Lambda. Lo stack è composto da un IAM ruolo e una funzione Lambda. La funzione `APIGatewayAuthorizer` Lambda è un autorizzatore per API Gateway in grado di convalidare la firma e le affermazioni di un token emesso dalla Virtual Waiting Room su. AWS API La funzione Lambda fornita con lo stack può essere utilizzata per proteggere il cloud APIs fino a quando un utente non ha attraversato la sala d'attesa e non riceve un token di accesso. L'autorizzatore recupera e memorizza automaticamente nella cache la chiave pubblica e la configurazione dal core per la verifica dei token. API Può essere utilizzato senza modifiche e può essere installato in qualsiasi AWS regione che supporti. AWS Lambda

# Adattatore OpenID

Lo stack di adattatori OpenID implementa funzioni Gateway API e Lambda che fungono da provider di identità OpenID. L'adattatore OpenID fornisce un set di funzionalità OIDC compatibili APIs che possono essere utilizzate con i software di hosting web esistenti che supportano i provider di OIDC identità, come AWS Elastic Load Balancers WordPress, o come provider di identità federato per Amazon Cognito o servizi simili. L'adattatore consente a un cliente di utilizzare la sala d'attesa nel flusso Authn/Authz quando utilizza un software di off-the-shelf web hosting con opzioni di integrazione limitate. Lo stack installa anche una CloudFront distribuzione con un bucket Amazon S3 come origine e un altro bucket S3 per la registrazione delle richieste. L'adattatore OpenID fornisce una pagina di sala d'attesa di esempio, simile a quella fornita nello stack di sala d'attesa di esempio, ma progettata per un flusso di autenticazione OpenID. Il processo di autenticazione prevede l'individuazione di una posizione nella coda della sala d'attesa e l'attesa che la posizione di servizio sia uguale o superiore a quella del cliente. La pagina della sala d'attesa OpenID reindirizza al sito di destinazione, che utilizza OpenID per completare l'acquisizione del token e la configurazione della sessione API per il client. APIGli endpoint di questa soluzione vengono mappati direttamente alle specifiche di flusso name-for-name ufficiali di OpenID Connect 1.0,. Per i dettagli, fare riferimento a OpenID Connect Core 1.0 Authentication.



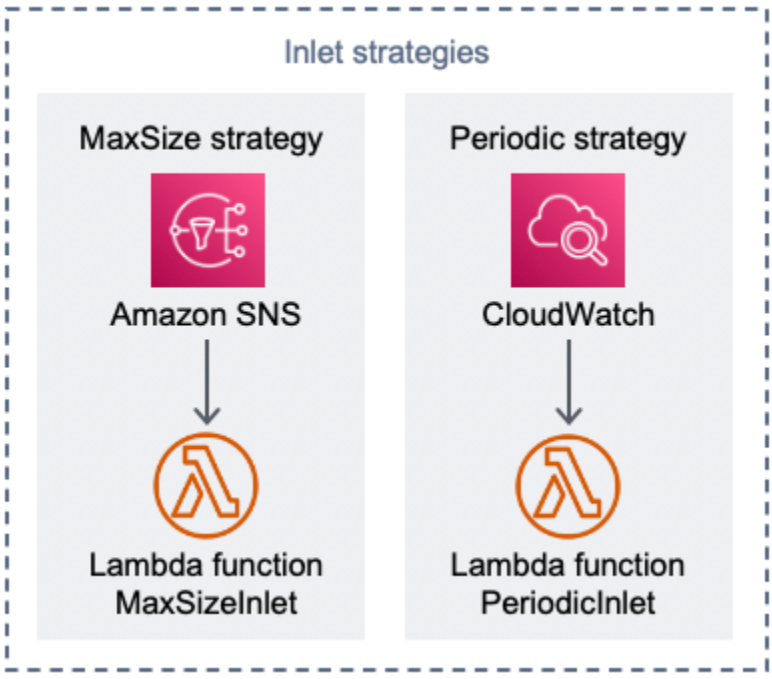
Sala d'attesa virtuale sul AWS componente adattatore OpenID

1. CloudFront la distribuzione fornisce il contenuto del bucket S3 all'utente.
2. Il bucket S3 ospita pagine di esempio per le sale d'attesa.
3. Amazon API Gateway API fornisce un set di funzionalità OIDC compatibili APIs che possono essere utilizzate con i software di hosting Web esistenti che supportano la funzione di autorizzazione Lambda del provider di OIDC identità.
4. La funzione `APIHandler` Lambda gestisce le richieste per tutti i percorsi di risorse del API Gateway. Diverse funzioni Python all'interno dello stesso modulo sono mappate su ciascun percorso. API Ad esempio, il percorso della `/authorize` risorsa in API Gateway richiama la funzione `authorize()` Lambda.
5. OIDCLe impostazioni sono memorizzate in Secrets Manager.

## Esempi di strategie di ingresso

Le strategie di ingresso determinano quando il banco di servizio della soluzione deve passare ad accogliere più utenti nel sito di destinazione. Per ulteriori informazioni concettuali sulle strategie di ingresso nelle sale d'attesa, consulta Considerazioni di progettazione.

Esistono due esempi di strategie di ingresso fornite dalla soluzione: e periodica. MaxSize




Componente delle strategie Virtual Waiting Room on AWS Inlet

Opzione strategica di ingresso Max Size:

1. Un client emette una SNS notifica Amazon che richiama la funzione `MaxSizeInlet` Lambda per aumentare il contatore di servizi in base al payload del messaggio.
2. La funzione `MaxSizeInlet` Lambda prevede di ricevere un messaggio indicante che utilizza per determinare di quanto incrementare il contatore di servizio.

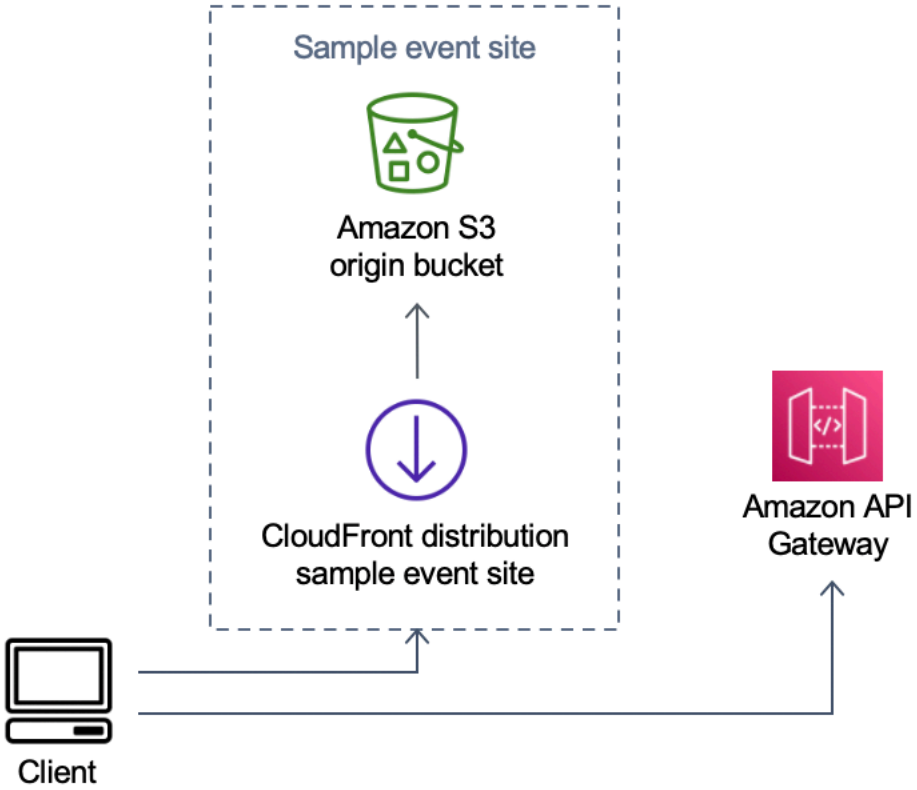
Opzione di strategia di ingresso periodica:

3. Una CloudWatch regola richiama una funzione Lambda ogni minuto per aumentare il contatore di servizio di una quantità fissa.
4. La funzione `PeriodicInlet` Lambda incrementa il contatore di servizio in base alla dimensione specificata se il tempo è compreso tra l'ora di inizio e quella di fine fornita. Facoltativamente, controlla un CloudWatch allarme e, se l'allarme è attivo, esegue l'incremento, altrimenti `OK` lo salta.

# Esempio di sala d'attesa

La sala d'attesa campione si integra con quella pubblica e privata oltre APIs all'autorizzazione personalizzata per dimostrare una soluzione minimale per le sale end-to-end d'attesa. La pagina Web principale viene archiviata in un bucket S3 e utilizzata come origine per. CloudFront Guida l'utente attraverso i seguenti passaggi:

1. Mettiti in fila nella sala d'attesa per entrare nel sito.
2. Ottieni la posizione del cliente in fila.
3. Ottieni la posizione di servizio della sala d'attesa.
4. Ottieni un set di token quando la posizione di servizio è uguale o superiore a quella del cliente.
5. Usa il token per chiamare un sistema di API autorizzazione protetto da Lambda.




Sala d'attesa virtuale su AWS Sample Componente del sito dell'evento

1. Il bucket S3 ospita il contenuto di esempio per la sala d'attesa e il pannello di controllo.
2. CloudFront la distribuzione fornisce il contenuto del bucket S3 all'utente.
3. Esempio di implementazione di API Gateway con percorsi di risorse simili a quelli dello shopping come e. `/search` `/checkout` Questo API viene installato dallo stack e configurato con il token authorizer. È inteso come esempio di un modo semplice per proteggere un uomo in sala API d'attesa. Le richieste che presentano un token valido vengono inoltrate a Lambda, altrimenti viene restituito un errore. Non vi è alcuna funzionalità API oltre alla risposta della funzione Lambda allegata.

# Sicurezza

Quando crei sistemi sull' AWS infrastruttura, le responsabilità in materia di sicurezza vengono condivise tra te e AWS. Questo modello condiviso riduce il carico operativo in quanto AWS gestisce, gestisce e controlla i componenti, tra cui il sistema operativo host, il livello di virtualizzazione e la sicurezza fisica delle strutture in cui operano i servizi. Per ulteriori informazioni sulla AWS sicurezza, visita AWS Cloud Security.

A Elasticache (RedisOSS) viene assegnata un'interfaccia di rete all'interno di quella privata. VPC Alle funzioni Lambda che interagiscono con Elasticache (RedisOSS) vengono inoltre assegnate interfacce di rete all'interno di un. VPC Tutte le altre risorse dispongono di connettività di rete nello spazio di rete condiviso. AWS Le funzioni Lambda con VPC interfacce che interagiscono con altri AWS servizi utilizzano gli VPC endpoint per connettersi a questi servizi.

Le chiavi pubbliche e private utilizzate per creare e convalidare i token JSON Web vengono generate al momento della distribuzione e archiviate in Secrets Manager. La password utilizzata per connettersi a Elasticache (RedisOSS) viene generata anche al momento della distribuzione e archiviata in Secrets Manager. La chiave privata e la password Elasticache (RedisOSS) non sono accessibili tramite alcuna soluzione. API

È API necessario accedere al pubblico tramite. CloudFront La soluzione genera una API chiave per API Gateway, che viene utilizzata come valore di un'intestazione personalizzata`x-api-key`, in CloudFront. CloudFront include questa intestazione quando si effettuano richieste di origine. Per ulteriori dettagli, consulta la sezione Aggiungere intestazioni personalizzate alle richieste di origine nella Amazon CloudFront Developer Guide.

I dati privati APIs sono configurati per richiedere AWS IAM l'autorizzazione all'invocazione. La soluzione crea il gruppo di `ProtectedAPIGroup` IAM utenti con le autorizzazioni appropriate per richiamare il privato. APIs Un IAM utente aggiunto a questo gruppo è autorizzato a richiamare il privato. APIs

IAMle politiche utilizzate nei ruoli e nelle autorizzazioni associate a varie risorse create dalla soluzione concedono solo le autorizzazioni necessarie per eseguire le attività necessarie.

Per risorse come bucket S3, SQS code e SNS argomenti generati dalla soluzione, la crittografia a riposo e durante il transito viene attivata laddove possibile.

# Monitoraggio

Lo API stack principale include diversi CloudWatch allarmi che possono essere monitorati per rilevare problemi mentre la soluzione è operativa. Lo stack crea un allarme per gli errori della funzione Lambda e le condizioni dell'acceleratore e modifica lo stato dell'allarme `OK` da `ALARM` a se si verifica un errore o una condizione di accelerazione in un periodo di un minuto.

Lo stack crea anche allarmi per ogni API implementazione del Gateway per i codici di stato 4XX e 5XX. Lo stato dell'allarme cambia da `OK` a `ALARM` se viene restituito un codice di stato 4XX o 5XX entro un periodo di un minuto. API

Questi allarmi tornano allo `OK` stato attivo dopo un minuto senza errori o acceleratori.

# IAMruoli

AWS Identity and Access Management (IAM) i ruoli consentono ai clienti di assegnare policy e autorizzazioni di accesso granulari a servizi e utenti sul Cloud. AWS Questa soluzione crea IAM ruoli che garantiscono l'accesso alle AWS Lambda funzioni della soluzione per creare risorse regionali.

# Amazon CloudFront

Il `virtual-waiting-room-on-aws.template` CloudFormation modello, che crea il nucleo pubblico e privato APIs della sala d'attesa, implementa anche una CloudFront distribuzione per il pubblicoAPI. CloudFront memorizza nella cache le risposte del pubblicoAPI, riducendo così il carico sul API Gateway e sulle funzioni Lambda che eseguono il lavoro.

Questa soluzione include anche un modello di sala d'attesa di esempio opzionale che distribuisce una semplice applicazione Web ospitata in un bucket Amazon Simple Storage Service (Amazon S3). Per contribuire a ridurre la latenza e migliorare la sicurezza, viene implementata una CloudFront distribuzione Amazon con un'identità di accesso di origine, ovvero un CloudFront utente che fornisce l'accesso pubblico ai contenuti del bucket del sito Web della soluzione. Per ulteriori informazioni, consulta la sezione Limitazione dell'accesso ai contenuti Amazon S3 utilizzando un'identità Origin Access nella  CloudFront Amazon Developer Guide.

# Gruppi di sicurezza

I gruppi VPC di sicurezza creati in questa soluzione sono progettati per controllare e isolare il traffico di rete verso Elasticache (Redis). OSS Le unità Lambda che devono comunicare con Elasticache

(RedisOSS) vengono inserite nello stesso gruppo di sicurezza di Elasticache (Redis). OSS Ti consigliamo di esaminare i gruppi di sicurezza e di limitare ulteriormente l'accesso, se necessario, una volta che la distribuzione è attiva e funzionante.

# Considerazioni di natura progettuale

## Opzioni di implementazione

Se è la prima volta che installi o non sei sicuro di cosa installare, distribuisci il CloudFormation modello `virtual-waiting-room-on-aws-getting-started.template` annidato, che installa il core, gli autorizzatori e i modelli di sala d'attesa di esempio. Ciò offre una sala d'attesa minimale con un flusso semplice.

## Protocolli supportati

La AWS soluzione Virtual Waiting Room on può essere integrata con quanto segue:

- JSONLibrerie e strumenti di verifica Web Token
- Implementazioni API Gateway esistenti
- RESTAPIclienti
- Client e provider OpenID

## Strategie di ingresso nelle sale d'attesa

Le strategie di ingresso racchiudono la logica e i dati necessari per spostare i clienti dalla sala d'attesa al sito Web. Una strategia di input può essere implementata come funzione Lambda, contenitore, istanza EC2 Amazon o qualsiasi altra risorsa di calcolo. Non è necessario che sia una risorsa cloud purché possa chiamare la sala d'attesa pubblica e privata. APIs La strategia di inlet riceve eventi relativi alla sala d'attesa, al sito Web o ad altri indicatori esterni che la aiutano a decidere quando più clienti possono far emettere token e accedere al sito. Esistono diversi approcci alle strategie di ingresso. La scelta da adottare dipende dalle risorse a tua disposizione e dai vincoli imposti dalla progettazione del sito web da proteggere.

L'azione principale intrapresa dalla strategia di ingresso consiste nel chiamare `increment_serving_num` Amazon API Gateway private API con un valore relativo che indica quanti altri client possono accedere al sito. Questa sezione descrive due strategie di ingresso di esempio. Queste possono essere utilizzate così come sono, personalizzate oppure è possibile utilizzare un approccio completamente diverso.

## MaxSize

Utilizzando la MaxSize strategia, la funzione `MaxSizeInlet` Lambda è configurata con il numero massimo di client che possono utilizzare il sito Web contemporaneamente. Si tratta di un valore fisso. Un client emette una SNS notifica Amazon che richiama la funzione `MaxSizeInlet` Lambda per aumentare il contatore di servizi in base al payload del messaggio. L'origine del SNS messaggio può provenire da qualsiasi luogo, incluso il codice sul sito Web o uno strumento di monitoraggio che osserva il livello di utilizzo del sito.

La funzione `MaxSizeInlet` Lambda prevede di ricevere un messaggio che può includere:

- `exited :`numero di transazioni completate
- elenco delle richieste IDs da contrassegnare come completate
- elenco delle richieste IDs da contrassegnare come abbandonate

Questi dati vengono utilizzati per determinare di quanto incrementare il contatore di servizio. In alcuni casi non esiste una capacità aggiuntiva per incrementare il contatore, in base al numero attuale di clienti.

## Periodic (Periodico)

Quando si utilizza la strategia periodica, una CloudWatch regola richiama la funzione `PeriodicInlet` Lambda ogni minuto per aumentare il contatore di servizio di una quantità fissa. L'ingresso periodico è parametrizzato con l'ora di inizio dell'evento, l'ora di fine e l'importo dell'incremento. Facoltativamente, questa strategia controlla anche un CloudWatch allarme e, se l'allarme è attivo, esegue l'incremento, altrimenti lo salta. `OK` Gli integratori del sito possono collegare una metrica di utilizzo a un allarme e utilizzare tale allarme per mettere in pausa l'ingresso periodico. Questa strategia modifica la posizione di servizio solo quando l'ora corrente è compresa tra l'ora di inizio e quella di fine e, facoltativamente, l'allarme specificato è nello stato. `OK`

# Personalizzazione ed estensione della soluzione

L'amministratore del sito dell'organizzazione deve decidere i metodi di integrazione da utilizzare con la sala d'attesa. Sono disponibili due opzioni:

1. Integrazione di base utilizzando direttamente APIs gli autorizzatori API Gateway.
2. Integrazione OpenID tramite un provider di identità.

Oltre all'integrazione di cui sopra, potrebbe essere necessario configurare il reindirizzamento del nome di dominio. Sei inoltre responsabile dell'implementazione di una pagina personalizzata del sito della sala d'attesa.

La AWS soluzione Virtual Waiting Room on è progettata per essere estesa attraverso due meccanismi: EventBridge per la notifica unidirezionale degli eventi e REST APIs per la comunicazione bidirezionale.

# Quote

La limitazione di scala principale per Virtual Waiting Room on AWS è il limite di accelerazione Lambda per la regione installata. AWS Se installata in un AWS account con la quota di esecuzione simultanea Lambda predefinita, la AWS soluzione Virtual Waiting Room on può gestire fino a 500 client al secondo che richiedono una posizione in coda. La tariffa di 500 client al secondo si basa sulla soluzione che prevede esclusivamente i limiti di quota simultanei per tutte le funzioni Lambda. Se la regione dell'account è condivisa con altre soluzioni che richiamano le funzioni Lambda, la soluzione Virtual Waiting Room AWS on dovrebbe avere almeno 1.000 chiamate simultanee disponibili. Puoi utilizzare le CloudWatch metriche per tracciare un grafico delle chiamate simultanee Lambda nel tuo account nel tempo per prendere una decisione. Puoi utilizzare la console Service Quotas per richiedere aumenti. L'aumento del limite di accelerazione Lambda aumenta i costi mensili dell'account solo se si verificano effettivamente chiamate aggiuntive.

Per ogni 500 client aggiuntivi al secondo, aumenta il limite di accelerazione di 1.000.

| Sono previsti utenti in entrata al secondo | Quota di esecuzione simultanea consigliata |
|---|---|
| 0-500 | 1.000 (impostazione predefinita) |
| 501-1.000 | 2.000 |
| 1.001-1.500 | 3.000 |

Lambda ha un limite di burst fisso di 3.000 chiamate simultanee. Per ulteriori informazioni, consulta la sezione Scalabilità delle funzioni Lambda. Il codice client dovrebbe prevedere e riprovare alcune API chiamate se viene restituito un codice di errore che indica una situazione temporanea di accelerazione. Il client di esempio per la sala d'attesa include questo codice come esempio di come progettare client utilizzati in eventi ad alta capacità e ad alta frequenza.

Questa soluzione è anche compatibile con Lambda Reserved e Provisioned in concomitanza con fasi di configurazione personalizzate. Per i dettagli, consulta Gestione della concorrenza riservata Lambda.

Il limite massimo di utenti che possono entrare nella sala d'attesa, ricevere un token e continuare una transazione è limitato dal limite superiore dei contatori Elasticache (Redis). OSS I contatori vengono utilizzati per la posizione di servizio della sala d'attesa e per tracciare lo stato riepilogativo della soluzione. I contatori utilizzati in Elasticache (RedisOSS) hanno un limite superiore di 9.223.372.036.854.775.807. Una tabella DynamoDB viene utilizzata per archiviare una copia di ogni token rilasciato a un utente della sala d'attesa. DynamoDB non ha limiti pratici alla dimensione di una tabella.

## Implementazioni regionali

I servizi utilizzati da questa soluzione sono supportati in tutte le AWS regioni. Per la disponibilità più aggiornata dei AWS servizi per regione, consulta l'Elenco dei servizi AWS regionali.

# AWS CloudFormation modelli

Per automatizzare l'implementazione, questa soluzione utilizza i seguenti AWS CloudFormation modelli, che è possibile scaricare prima della distribuzione.

Se è la prima volta che installi o non sei sicuro di cosa installare, distribuisci il `virtual-waiting-room-on-aws-getting-started.template` AWS CloudFormation modello, che installa il core, gli autorizzatori e i modelli di codice di esempio per le sale d'attesa. Ciò consente di testare una sala d'attesa funzionante con un semplice flusso.

View template

virtual-waiting-room-on- aws-api-gateway-cw -logs-role.template: utilizza questo modello per aggiungere un ruolo predefinito a API Gateway a livello di account ARN per le autorizzazioni di registrazione. CloudWatch Fai riferimento a Prerequisiti per sapere se il tuo account richiede o meno l'implementazione di questo modello.

View template

virtual-waiting-room-on- aws-getting-started .template: utilizza questo modello annidato per installare gli stack di base, gli autorizzatori e gli esempi di stack di sala d'attesa.

View template

virtual-waiting-room-on-aws.template: utilizza questo modello di base per installare i principali servizi pubblici, privati REST APIs e cloud per la creazione di eventi in sala d'attesa. Installa questo modello nell'account e nella regione in cui ti servono la sala d'attesa RESTAPIs, Elasticache (RedisOSS) e la tabella DynamoDB.

View template

virtual-waiting-room-on-aws-authorizers.template: utilizza questo modello per installare l'autorizzatore Lambda progettato per verificare i token emessi dalla sala d'attesa e destinato a proteggere quelli degli utenti finali. APIs Richiede lo stack principale. Alcuni output dello stack principale sono necessari come parametri per distribuire questo stack. Questo è un modello opzionale.

View template

virtual-waiting-room-on-aws-openid.template: utilizza questo modello per installare un provider di identità OpenID per l'integrazione delle sale d'attesa con le interfacce di autorizzazione. Richiede lo stack principale. Per distribuire questo stack sono necessari alcuni output dello stack principale. Questo è un modello opzionale.

View template

virtual-waiting-room-on- aws-sample-inlet-strategy .template: utilizza questo modello per installare esempi di strategie di ingresso da utilizzare tra un sito di destinazione e la sala d'attesa. Le strategie di ingresso aiutano a incapsulare la logica per determinare quando consentire a più utenti di accedere al sito di destinazione. Richiede lo stack principale. Gli output dello stack principale sono necessari per distribuire questo stack. Questo è un modello opzionale.

View template

virtual-waiting-room-on-aws-sample.template: utilizza questo modello per installare un esempio di configurazione minima per Web e API Gateway per una sala d'attesa e un sito di destinazione. Richiede gli stack core e authorizers. Gli output degli stack core e degli authorizers sono necessari come parametri per distribuire questo stack. Questo è un modello opzionale.

# Implementazione automatica

Prima di avviare la soluzione, esaminate i costi, l'architettura, la sicurezza della rete e altre considerazioni discusse in questa guida. Segui le step-by-step istruzioni in questa sezione per configurare e distribuire la soluzione nel tuo account.

Tempo di implementazione: circa 30 minuti (solo stack introduttivo)

## Prerequisiti

- AWS autorizzazioni per la console dell'account equivalenti a quelle di Administrator Access.
- Attiva la CloudWatch registrazione da API Gateway:
  - Accedi alla console API Gateway e seleziona la regione in cui intendi installare gli stack.

    Se hai già APIs definito qualcosa in questa regione:

    1. Seleziona qualsiasiAPI.
    2. Dalla barra di navigazione a sinistra, seleziona Impostazioni.
    3. Verifica la presenza di un valore nel ARN campo del ruolo del CloudWatch registro.
    - Se non ce n'èARN, installa il`virtual-waiting-room-on-aws-api-gateway-cw-logs-role.template`.
    - Se ce n'è unoARN, inizia con il lancio dello stack introduttivo.

    Se non ci sono APIs definizioni esistenti in questa regione, installa. `virtual-waiting-room-on-aws-api-gateway-cw-logs-role.template`
- Conoscenza dell'architettura e dei dettagli di implementazione del sito di destinazione da proteggere.

## Panoramica della distribuzione

Utilizza i seguenti passaggi per distribuire questa soluzione su AWS. Per istruzioni dettagliate, segui i collegamenti per ciascuna fase.

Fase 1: Avvia lo stack introduttivo

- Avvia il AWS CloudFormation modello nel tuo account. AWS
- Rivedi i parametri dei modelli e inserisci o modifica i valori predefiniti in base alle esigenze.

Fase 2. (Facoltativo) Prova la sala d'attesa

- Genera AWS chiavi per chiamare la IAM sicurezza. APIs
- Apri il pannello di controllo della sala d'attesa campione.
- Prova la sala d'attesa dei campioni.

# Fase 1: Avvia lo stack introduttivo

Questo AWS CloudFormation modello automatizzato implementa i modelli di base, gli autorizzatori e i modelli di sala d'attesa di esempio che consentono di visualizzare e testare una sala d'attesa funzionante. È necessario leggere e comprendere i prerequisiti prima di avviare lo stack.

> **Note**
>
> L'utente è responsabile del costo dei AWS servizi utilizzati durante l'esecuzione di questa soluzione. Per ulteriori dettagli, visita la sezione Costi di questa guida e consulta la pagina web dei prezzi per ogni AWS servizio utilizzato in questa soluzione.

1. Accedi a AWS Management Consolee seleziona il pulsante per avviare il `virtual-waiting-room-on-aws-getting-started.template` AWS CloudFormation modello.

   Launch solution

   In alternativa, puoi scaricare il modello come punto di partenza per la tua implementazione.
2. Per impostazione predefinita, il modello viene avviato nella regione Stati Uniti orientali (Virginia settentrionale). Per avviare la soluzione in un'altra AWS regione, utilizza il selettore della regione nella barra di navigazione della console.
3. Nella pagina Create stack, verifica che il modello corretto URL sia nella casella di testo Amazon URL S3 e scegli Avanti.
4. Nella pagina Specificare i dettagli dello stack, assegna un nome allo stack di soluzioni. Per informazioni sulle limitazioni relative ai nomi dei caratteri, consulta IAMe STS Limiti nella Guida per l'utente.AWS Identity and Access Management
5. In Parametri, esaminate i parametri per questo modello di soluzione e modificateli se necessario. Questa soluzione utilizza i seguenti valori predefiniti.

| Parametro | Predefinito | Descrizione |
| --- | --- | --- |
| ID evento | `Sample` | ID univoco per questa istanza della sala d'attesa, GUID formato consigliato. |
| Periodo di validità | `3600` | Periodo di validità del token in secondi. |
| Abilita la generazione di eventi | `false` | Se impostato su`true`, le metriche relative alla sala d'attesa vengono scritte nel relativo bus degli eventi ogni minuto |
| Porta Elasticache (Redis) OSS | `1785` | Il numero di porta da utilizzar e per la connessione al server Elasticache (Redis). OSS Si consiglia di non utilizzare la porta Elasticache (Redis) predefinita di. OSS `6379` |
| EnableQueuePositionExpiry | `true` | Se impostato su`false`, il periodo di scadenza della posizione in coda non viene applicato. |
| QueuePositionExpiryPeriod | `900` | È l'intervallo di tempo in secondi oltre il quale una posizione in coda non è idonea a generare un token. |

| Parametro | Predefinito | Descrizione |
| --- | --- | --- |
| IncrSvcOnQueuePosi tionExpiry | `false` | Se impostato su`true`, il contatore di servizio viene automaticamente avanzato in base alle posizioni di coda scadute che non hanno generato correttamente i token. |

6. Scegli Next (Successivo).
7. Nella pagina Configure stack options (Configura opzioni pila), scegliere Next (Successivo).
8. Nella pagina Rivedi, verifica e conferma le impostazioni. Seleziona la casella per confermare che il modello crea AWS Identity and Access Management risorse (). IAM
9. Seleziona Create (Crea) per implementare lo stack.

   Puoi visualizzare lo stato dello stack nella AWS CloudFormation Console nella colonna Stato. Dovresti ricevere COMPLETE lo stato CREATE _ tra circa 30 minuti.

# Fase 2: (Facoltativo) Prova la sala d'attesa

Se hai implementato lo stack introduttivo, i seguenti passaggi ti aiutano a testare la funzionalità della sala d'attesa. Per completare il test, sono necessarie AWS chiavi con autorizzazioni per chiamare lo stack IAM secured in APIs the core.

## Genera AWS chiavi per chiamare la rete protetta IAM APIs

1. Crea o utilizza un IAM utente nell' AWS account in cui è stato distribuito il `aws-virtual-waiting-room-getting-started.template` CloudFormation modello.
2. Concedi all'IAMutente l'accesso programmatico. Quando crei un nuovo set di chiavi di accesso per l'IAMutente, scarica il file della chiave quando viene presentato. Per testare la sala d'attesa sono necessari l'ID della chiave di accesso e la chiave di accesso segreta dell'IAMutente.
3. Aggiungi l'IAMutente al gruppo di rotectedAPIGroup IAM utenti P creato dal modello.

## Apri il pannello di controllo della sala d'attesa di esempio

1. Accedi alla AWS CloudFormation console e seleziona lo stack introduttivo della soluzione.
2. Seleziona la scheda Outputs (Output).
3. Nella colonna Chiave ControlPanelURL, individua e seleziona il valore corrispondente.
4. Apri il pannello di controllo in una nuova scheda o finestra del browser.
5. Nel pannello di controllo, espandi la sezione Configurazione.
6. Inserisci l'ID della chiave di accesso e la chiave di accesso segreta che hai recuperato in Genera AWS chiavi per chiamare la IAM sicurezza APIs. Gli endpoint e l'ID dell'evento vengono compilati dai parametri. URL
7. Scegliete Usa. Il pulsante si attiva dopo aver fornito le credenziali.

## Prova il campione della sala d'attesa

1. Nella AWS CloudFormation console, seleziona lo stack introduttivo della soluzione.
2. Seleziona la scheda Outputs (Output).
3. Nella colonna Chiave WaitingRoomURL, individua e seleziona il valore corrispondente.
4. Apri la sala d'attesa, quindi scegli Prenota per entrare nella sala d'attesa.
5. Torna alla scheda del browser che contiene il pannello di controllo.
6. In Increment Serving Counter, scegli Cambia. Ciò consente a 100 utenti di passare dalla sala d'attesa al sito di destinazione.
7. Torna alla sala d'attesa e scegli Check out now! Ora verrai reindirizzato al sito di destinazione.
8. Scegli Acquista ora per completare la transazione sul sito di destinazione.

# Implementazione di stack separati

Lo stack principale è l'unico stack richiesto per ottenere le funzionalità principali della sala d'attesa. Tutti gli altri stack sono opzionali. Avvia lo stack degli autorizzatori se non disponi già di un modo per convalidare i token emessi dalla sala d'attesa o proteggere quelli che potresti già avere. APIs Avvia lo stack OpenID se hai bisogno di un provider di identità OpenID per l'integrazione delle sale d'attesa con le interfacce di autorizzazione. Lo stack Sample Inlet Strategy fornisce un paio di esempi su come e quando consentire a più utenti di accedere al sito che state cercando di proteggere.

## 1. Avvia lo stack principale

Tempo di implementazione: circa 20 minuti

Questo AWS CloudFormation modello automatizzato implementa Virtual Waiting Room on the AWS AWS Cloud. È necessario completare i prerequisiti prima di avviare lo stack.

> **Note**
>
> L'utente è responsabile del costo dei AWS servizi utilizzati durante l'esecuzione di questa soluzione. Per ulteriori dettagli, visita la sezione Costi di questa guida e consulta la pagina web dei prezzi per ogni AWS servizio utilizzato in questa soluzione.

1. Accedi a AWS Management Consolee seleziona il pulsante per avviare il `aws-virtual-waiting-room-on-aws.template` AWS CloudFormation modello.

   Launch solution In alternativa, puoi scaricare il modello come punto di partenza per la tua implementazione.
2. Per impostazione predefinita, il modello viene avviato nella regione Stati Uniti orientali (Virginia settentrionale). Per avviare la soluzione in un'altra AWS regione, utilizza il selettore della regione nella barra di navigazione della console.
3. Nella pagina Create stack, verifica che il modello corretto URL sia nella casella di testo Amazon URL S3 e scegli Avanti.
4. Nella pagina Specificare i dettagli dello stack, assegna un nome allo stack di soluzioni. Per informazioni sulle limitazioni relative ai nomi dei caratteri, consulta IAMe STS Limiti nella Guida per l'utente.AWS Identity and Access Management

5. In Parametri, esaminate i parametri per questo modello di soluzione e modificateli se necessario. Questa soluzione utilizza i seguenti valori predefiniti.

| Parametro | Predefinito | Descrizione |
| --- | --- | --- |
| ID evento | `Sample` | ID univoco per questa istanza della sala d'attesa, GUID formato consigliato. |
| Periodo di validità | `3600` | Periodo di validità del token in secondi. |
| Abilita la generazione di eventi | `false` | Se impostato su`true`, le metriche relative alla sala d'attesa vengono scritte sul relativo bus degli eventi ogni minuto. |
| Porta Elasticache (Redis) OSS | `1785` | Il numero di porta da utilizzar e per la connessione al server Elasticache (Redis). OSS Si consiglia di non utilizzare la porta Elasticache (Redis) predefinita di. OSS `6379` |
| EnableQueuePositionExpiry | `true` | Se impostato su`false`, il periodo di scadenza della posizione in coda non viene applicato. |
| QueuePositionExpiryPeriod | `900` | È l'intervallo di tempo in secondi oltre il quale una posizione in coda non è idonea a generare un token. |

| Parametro | Predefinito | Descrizione |
| --- | --- | --- |
| IncrSvcOnQueuePosi tionExpiry | `false` | Se impostato su`true`, il contatore di servizio viene automaticamente avanzato in base alle posizioni di coda scadute che non hanno generato correttamente i token. |

6. Scegli Next (Successivo).
7. Nella pagina Configure stack options (Configura opzioni pila), scegliere Next (Successivo).
8. Nella pagina Rivedi, verifica e conferma le impostazioni. Seleziona la casella per confermare che il modello crea AWS Identity and Access Management risorse (). IAM
9. Seleziona Create (Crea) per implementare lo stack.

   Puoi visualizzare lo stato dello stack nella AWS CloudFormation Console nella colonna Stato. Dovresti ricevere COMPLETE lo stato CREATE _ tra circa 20 minuti.

## 2. (Facoltativo) Avvia lo stack Authorizers

Durata dell'implementazione: circa cinque minuti

1. Accedi a AWS Management Consolee seleziona il pulsante per avviare il `aws-virtual-waiting-room-on-aws-authorizers.template` AWS CloudFormation modello.

   Launch solution In alternativa, puoi scaricare il modello come punto di partenza per la tua implementazione.
2. Per impostazione predefinita, il modello viene avviato nella regione Stati Uniti orientali (Virginia settentrionale). Per avviare la soluzione in un'altra AWS regione, utilizza il selettore della regione nella barra di navigazione della console.
3. Nella pagina Create stack, verifica che il modello corretto URL sia nella casella di testo Amazon URL S3 e scegli Avanti.

4. Nella pagina Specificare i dettagli dello stack, assegna un nome allo stack di soluzioni. Per informazioni sulle limitazioni relative ai nomi dei caratteri, consulta IAMe STS Limiti nella Guida per l'utente.AWS Identity and Access Management
5. In Parametri, esaminate i parametri per questo modello di soluzione e modificateli se necessario. Questa soluzione utilizza i seguenti valori predefiniti.

| Parametro | Predefinito | Descrizione |
|---|---|---|
| APIEndpoint pubblico | *<Requires input>* | Endpoint pubblico per la sala d'attesa virtuale. APIs |
| ID evento della sala d'attesa | `Sample` | ID evento della sala d'attesa. |
| Emittente URI | *<Requires input>* | Emittente URI delle chiavi e dei token pubblici. |

6. Scegli Next (Successivo).
7. Nella pagina Configure stack options (Configura opzioni pila), scegliere Next (Successivo).
8. Nella pagina Rivedi, verifica e conferma le impostazioni. Seleziona la casella per confermare che il modello crea risorse AWS Identity and Access Management ()IAM.
9. Seleziona Create (Crea) per implementare lo stack.

   Puoi visualizzare lo stato dello stack nella AWS CloudFormation Console nella colonna Stato. Dovresti ricevere COMPLETE lo stato CREATE _ tra circa cinque minuti.

## 3. (Facoltativo) Avvia lo stack OpenID

Durata dell'implementazione: circa cinque minuti

1. Accedi a AWS Management Consolee seleziona il pulsante per avviare il `aws-virtual-waiting-room-on-aws-openid.template` AWS CloudFormation modello.

   Launch solution In alternativa, puoi scaricare il modello come punto di partenza per la tua implementazione.

2. Per impostazione predefinita, il modello viene avviato nella regione Stati Uniti orientali (Virginia settentrionale). Per avviare la soluzione in un'altra AWS regione, utilizza il selettore della regione nella barra di navigazione della console.
3. Nella pagina Create stack, verifica che il modello corretto URL sia nella casella di testo Amazon URL S3 e scegli Avanti.
4. Nella pagina Specificare i dettagli dello stack, assegna un nome allo stack di soluzioni. Per informazioni sulle limitazioni relative ai nomi dei caratteri, consulta IAMe STS Limiti nella Guida per l'utente.AWS Identity and Access Management
5. In Parametri, esaminate i parametri per questo modello di soluzione e modificateli se necessario. Questa soluzione utilizza i seguenti valori predefiniti.

| Parametro | Predefinito | Descrizione |
|---|---|---|
| APIEndpoint pubblico | *<Requires input>* | Endpoint pubblico URL per la sala d'attesa virtuale. APIs |
| Endpoint privato API | *<Requires input>* | Endpoint privato URL per la sala d'attesa virtuale. APIs |
| APIRegione | *<Requires input>* | AWS nome della regione per la sala d'attesa pubblica e privataAPIs. |
| ID evento | `Sample` | ID evento della sala d'attesa. |

6. Scegli Next (Successivo).
7. Nella pagina Configure stack options (Configura opzioni pila), scegliere Next (Successivo).
8. Nella pagina Rivedi, verifica e conferma le impostazioni. Seleziona la casella che conferma che il modello crea AWS Identity and Access Management (IAM) risorse.
9. Seleziona Create (Crea) per implementare lo stack.

   Puoi visualizzare lo stato dello stack nella AWS CloudFormation Console nella colonna Stato. Dovresti ricevere COMPLETE lo stato CREATE _ tra circa cinque minuti.

# 4. (Facoltativo) Avvia lo stack Sample Inlet Strategy

Tempo di implementazione: circa due minuti

1. Accedi a AWS Management Consolee seleziona il pulsante per avviare il `aws-virtual-waiting-room-sample-inlet-strategy.template` AWS CloudFormation modello.

   Launch solution

   In alternativa, puoi scaricare il modello come punto di partenza per la tua implementazione.
2. Per impostazione predefinita, il modello viene avviato nella regione Stati Uniti orientali (Virginia settentrionale). Per avviare la soluzione in un'altra AWS regione, utilizza il selettore della regione nella barra di navigazione della console.
3. Nella pagina Create stack, verifica che il modello corretto URL sia nella casella di testo Amazon URL S3 e scegli Avanti.
4. Nella pagina Specificare i dettagli dello stack, assegna un nome allo stack di soluzioni. Per informazioni sulle limitazioni relative ai nomi dei caratteri, consulta IAMe STS Limiti nella Guida per l'utente.AWS Identity and Access Management
5. In Parametri, esaminate i parametri per questo modello di soluzione e modificateli se necessario. Questa soluzione utilizza i seguenti valori predefiniti.

| Parametro | Predefinito | Descrizione |
|---|---|---|
| ID evento | `Sample` | ID evento della sala d'attesa. |
| APIEndpoint principale privato | *`<Requires input>`* | Endpoint privato URL per la sala d'attesa virtuale. APIs |
| Regione centrale API | *`<Requires input>`* | AWS Regione in cui API è installato il core. |
| Strategia di ingresso | `Periodic` | Strategia di ingresso da implementare. `Periodic`aumenta il numero di porzioni ogni minuto. `MaxSize`incrementa il numero di servizi in base al |

| Parametro | Predefinito | Descrizione |
| --- | --- | --- |
| | | numero massimo di transazioni che il sito di destinazione a valle può gestire in un determinato momento. |
| Incrementa per | *<Requires input>* | Di quanto deve essere incrementato il contatore di servizio ogni minuto. Necessario se si seleziona la strategia di ingresso periodica. |
| Ora di inizio | *<Requires input>* | Indicazione temporale su quando iniziare ad incrementare il numero di porzioni (epoca in secondi). Obbligatorio se si seleziona una strategia di ingresso periodica. |
| End Time (Ora di fine) | *<Requires input>* | Indicazione temporale su quando interrompere l'incremento del numero di porzioni (epoca in secondi). Se lasciato 0, il numero di porzioni viene incrementato a tempo indeterminato. Necessario se si seleziona una strategia di ingresso periodica. |

| Parametro | Predefinito | Descrizione |
| --- | --- | --- |
| CloudWatch Nome dell'allarme | *<Requires input>* | Nome opzionale CloudWatch dell'allarme da associare alla strategia di ingresso periodica. Se fornito e in stato di allarme, il numero di servizio non viene incrementato. Applicabile solo alla strategia di ingresso periodica. |
| Dimensione massima | *<Requires input>* | Il numero massimo di transazioni che il sito di destinazione a valle può elaborare alla volta (MaxSize Strategia). |

6. Scegli Next (Successivo).
7. Nella pagina Configure stack options (Configura opzioni pila), scegliere Next (Successivo).
8. Nella pagina Rivedi, verifica e conferma le impostazioni. Seleziona la casella che conferma che il modello crea AWS Identity and Access Management (IAM) risorse.
9. Seleziona Create (Crea) per implementare lo stack.

   Puoi visualizzare lo stato dello stack nella AWS CloudFormation Console nella colonna Stato. Dovresti ricevere COMPLETE lo stato CREATE _ tra circa due minuti.

# 5. (Facoltativo) Avvia lo stack di esempio per le sale d'attesa

Durata dell'implementazione: circa cinque minuti

1. Accedi a AWS Management Consolee seleziona il pulsante per avviare il `aws-virtual-waiting-room-sample.template` AWS CloudFormation modello.

   Launch solution

   In alternativa, puoi scaricare il modello come punto di partenza per la tua implementazione.

2. Per impostazione predefinita, il modello viene avviato nella regione Stati Uniti orientali (Virginia settentrionale). Per avviare la soluzione in un'altra AWS regione, utilizza il selettore della regione nella barra di navigazione della console.
3. Nella pagina Create stack, verifica che il modello corretto URL sia nella casella di testo Amazon URL S3 e scegli Avanti.
4. Nella pagina Specificare i dettagli dello stack, assegna un nome allo stack di soluzioni. Per informazioni sulle limitazioni relative ai nomi dei caratteri, consulta IAMe STS Limiti nella Guida per l'utente.AWS Identity and Access Management
5. In Parametri, esaminate i parametri per questo modello di soluzione e modificateli se necessario. Questa soluzione utilizza i seguenti valori predefiniti.

| Parametro | Predefinito | Descrizione |
|---|---|---|
| APIRegione del gateway | *<Requires input>* | AWS Nome della regione del API gateway. |
| Autorizzatore ARN | *<Requires input>* | ARNdell'autorizzatore API Gateway Lambda. |
| ID evento | `Sample` | ID evento della sala d'attesa. |
| APIEndpoint privato | *<Requires input>* | Endpoint privato URL per la sala d'attesa virtuale. APIs |
| Endpoint pubblico API | *<Requires input>* | Endpoint pubblico URL per la sala d'attesa virtuale. APIs |

6. Scegli Next (Successivo).
7. Nella pagina Configure stack options (Configura opzioni pila), scegliere Next (Successivo).
8. Nella pagina Rivedi, verifica e conferma le impostazioni. Seleziona la casella per confermare che il modello crea AWS Identity and Access Management (IAM) risorse.
9. Seleziona Create (Crea) per implementare lo stack.

   Puoi visualizzare lo stato dello stack nella AWS CloudFormation Console nella colonna Stato. Dovresti ricevere COMPLETE lo stato CREATE _ tra circa cinque minuti.

# Aggiornamento dello stack da una versione precedente

Ti consigliamo di eliminare lo stack e di creare un nuovo stack per la nuova versione. Attualmente, la migrazione alla versione più recente tramite CloudFormation stack update non è supportata. Vedi Disinstalla la soluzione quindi Avvia lo stack introduttivo.

> **Note**
>
> Ti consigliamo di passare a una versione più recente quando non utilizzi attivamente la soluzione per supportare un evento in corso.

# Dati di prestazioni

Virtual Waiting Room on è AWS stato sottoposto a test di carico con uno strumento chiamato Locust. Le dimensioni degli eventi simulati variavano da 10.000 a 100.000 clienti. L'ambiente di test di carico consisteva nella seguente configurazione:

- Locust 2.x con personalizzazioni per le implementazioni cloud AWS
- Quattro AWS regioni (,,,) `us-west-1` `us-west-2` `us-east-1` `us-east-2`
- 10 host `c5.4xlarge` Amazon EC2 per regione (40 in totale)
- 32 processi Locust per host
- Gli utenti simulati sono stati distribuiti uniformemente tra i 1.280 processi

Le fasi di test dell' end-to-end API per ogni processo utente:

1. Chiama `assign_queue_num` e ricevi un ID di richiesta.
2. Esegui il loop `queue_num` con l'ID della richiesta finché non restituisce la posizione in coda dell'utente (breve periodo).
3. Esegui il ciclo `serving_num` finché il valore restituito non è **>=** posizione in coda dell'utente (lungo periodo).
4. Chiama raramente `waiting_room_size` per recuperare il numero di utenti in attesa.
5. Chiama `generate_token` e ricevi un JWT da utilizzare nel sito di destinazione.

# Risultati

Non esiste un limite massimo pratico al numero di clienti che possono essere processati nella sala d'attesa.

La velocità con cui gli utenti entrano nella sala d'attesa influisce sulle quote di esecuzione simultanea della funzione Lambda per la regione in cui viene distribuita.

Il test di carico non è stato in grado di superare i limiti di richiesta API Gateway predefiniti di 10.000 richieste al secondo con le politiche di caching utilizzate con CloudFront.

La funzione `get_queue_num` Lambda ha una frequenza di invocazione vicina a 1:1 rispetto alla frequenza degli utenti in entrata nella sala d'attesa. Questa funzione Lambda può essere limitata

durante un elevato numero di utenti in entrata a causa di limiti di concorrenza o limiti di burst. La limitazione causata da un gran numero di chiamate di funzioni `get_queue_num` Lambda può influire su altre funzioni Lambda come effetto collaterale. L'intero sistema continua a funzionare se il software client è in grado di rispondere in modo appropriato a questo tipo di errore di ridimensionamento temporaneo con la logica retry/back-off.

La CloudFront distribuzione configurata dallo stack principale in una configurazione di quote predefinita è in grado di gestire una sala d'attesa con 250.000 utenti, ciascuno dei quali esegue il polling dell'API almeno ogni secondo. `serving_num`

# Risoluzione dei problemi

Questa sezione fornisce informazioni sulla risoluzione dei problemi relativi a questa soluzione.

Se questa sezione non risolve il problema, Contatta AWS Support fornisce istruzioni per aprire un caso AWS Support per questa soluzione.

stato della risposta 4xx dalle API

- Ciò può essere causato da un ID evento o da un ID di richiesta errati o da entrambi. Ciò si verifica nei CloudWatch registri per la funzione Lambda correlata.
- Le API private sono autenticate da IAM e il client necessita di AWS chiavi con diritti per richiamare le API private. Ciò si verifica nei CloudWatch Logs for API Gateway.

stato di risposta 5xx dalle API

- Risposta da Lambda o API Gateway con limitazione, verifica allarme. *<LambdaFunctionName>*`ThrottlesAlarm` CloudWatch
- Configurazione errata sul back-end, controlla Alarm e Logs per i dettagli *<LambdaFunctionName>*`ErrorsAlarm` CloudWatch . CloudWatch

5XX/ErrorPublicPrivateApiAlarm

- Questo stato di allarme `ALARM` si verifica quando l'API restituisce uno stato 5XX al chiamante entro un periodo di 60 secondi.
- Questo allarme ritorna `OK` quando non viene restituito lo stato 5xx per 60 secondi.
- Questo allarme può essere avviato da una funzione Lambda o da un runtime Lambda che restituisce un errore ad API Gateway.

4XX/ErrorPublicPrivateApiAlarm

- Questo stato di allarme si `ALARM` verifica quando l'API restituisce uno stato 4XX al chiamante entro un periodo di 60 secondi.
- Questo allarme ritorna `OK` quando viene ripristinato lo stato 4XX per 60 secondi.
- Questo allarme può essere avviato da un URL API errato.

*<LambdaFunctionName>*ThrottlesAlarm

- Questo stato di allarme è ALARM quando la Lambda denominata incontra un limite di esecuzione simultanea entro un periodo di 60 secondi.
- Questo allarme si attiva OK se non viene rilevata alcuna accelerazione per 60 secondi.
- Potrebbe essere necessario aumentare il limite di concorrenza per la regione del tuo account.
- Potresti incontrare il limite di burst per Lambda, che richiede una logica di ripetizione dei tentativi sul tuo client.

*<LambdaFunctionName>*ErrorsAlarm

- Questo stato di allarme si verifica ALARM quando la Lambda denominata rileva un errore di esecuzione in un periodo di 60 secondi.
- Questo allarme torna attivo OK se non vengono rilevati errori per 60 secondi.
- Ciò può essere causato da un'errata configurazione del backend.
- Ciò può essere causato da un bug nel codice di Lambda.

# Contatto AWS Support

Se disponi di AWS Developer Support, AWS Business Support o AWS Enterprise Support, puoi utilizzare il Support Center per ottenere l'assistenza di esperti su questa soluzione. Le istruzioni per eseguire tali operazioni sono fornite nelle sezioni seguenti.

## Crea un caso

1. Accedi al Support Center.
2. Scegli Crea caso.

## Come possiamo aiutarti?

1. Scegli Tecnico.
2. Per Assistenza, seleziona Soluzioni.
3. Per Categoria, seleziona Altre soluzioni.
4. Per Severità, seleziona l'opzione più adatta al tuo caso d'uso.

5. Quando si inseriscono i campi Servizio, Categoria e Severità, l'interfaccia compila i collegamenti alle domande più comuni per la risoluzione dei problemi. Se non riesci a risolvere la tua domanda con questi link, scegli Passaggio successivo: Informazioni aggiuntive.

## Informazioni aggiuntive

1. In Oggetto, inserisci il testo che riassume la domanda o il problema.
2. Per Descrizione, descrivi il problema in dettaglio.
3. Scegli Allega file.
4. Allega le informazioni AWS Support necessarie per elaborare la richiesta.

## Aiutaci a risolvere il tuo caso più velocemente

1. Inserisci le informazioni richieste.
2. Scegli Passaggio successivo: risolvi ora o contattaci.

## Risolvi subito o contattaci

1. Rivedi le soluzioni Solve now.
2. Se non riesci a risolvere il problema con queste soluzioni, scegli Contattaci, inserisci le informazioni richieste e scegli Invia.

# Risorse aggiuntive

| AWS servizi | |
|---|---|
| • AWS CloudFormation | • Amazon DynamoDB |
| • Amazon Simple Storage Service | • Amazon API Gateway |
| • AWS Lambda | • AWS Secrets Manager |
| • Amazon CloudFront | • Amazon Simple Queue Service |
| • Amazon EventBridge | • Amazon CloudWatch |
| • Elasticache (Redis) OSS | • Amazon Comprehend |
| • Amazon Virtual Private Cloud | • AWS Identity and Access Management |

# Disinstalla la soluzione

È possibile disinstallare la AWS soluzione Virtual Waiting Room on dalla AWS Management Console o utilizzando la AWS Command Line Interface. È necessario eliminare manualmente i bucket S3 utilizzati per archiviare i log da varie risorse create da questa soluzione. AWS Le implementazioni delle soluzioni non eliminano automaticamente questi bucket S3, quindi è ancora possibile rivedere i log degli eventi dopo l'eliminazione della soluzione.

Se hai aggiunto manualmente un utente IAM al gruppo di utenti `ProtectedAPIGroup` IAM creato dalla soluzione, rimuovi l'utente IAM dal gruppo di utenti IAM prima di disinstallare la soluzione. In caso contrario, il gruppo di utenti IAM e la policy IAM associata non verranno eliminati.

Per ciascuno degli stack distribuiti, segui le istruzioni riportate di seguito.

## Usando il AWS Management Console

1. Accedi alla console AWS CloudFormation.
2. Nella pagina Stack, seleziona lo stack di installazione di questa soluzione.
3. Scegli Elimina.

## Usando AWS Command Line Interface

Determina se AWS Command Line Interface (AWS CLI) è disponibile nel tuo ambiente. Per le istruzioni di installazione, consulta What Is the AWS Command Line Interface? nella Guida AWS CLI per l'utente. Dopo aver verificato che AWS CLI sia disponibile, esegui il comando seguente.

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

## Eliminazione dei bucket Amazon S3

Questa soluzione è configurata per conservare il bucket Amazon S3 creato dalla soluzione (per la distribuzione in una regione opt-in) se decidi di eliminare lo stack per prevenire AWS CloudFormation la perdita accidentale di dati. Dopo aver disinstallato la soluzione, puoi eliminare manualmente questo bucket S3 se non hai bisogno di conservare i dati. Segui questi passaggi per eliminare il bucket Amazon S3.

1. Accedere alla console Amazon S3.
2. Scegli Bucket dal riquadro di navigazione a sinistra.
3. Individua i <stack-name>bucket S3.
4. Seleziona il bucket S3 e scegli Elimina.

Per eliminare il bucket S3 utilizzando AWS CLI, esegui il seguente comando:

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

# Codice sorgente

Visita il nostro GitHubrepository per scaricare i file sorgente di questa soluzione e condividere le tue personalizzazioni con altri.

# Collaboratori

- Jim Thario
- Thyag Ramachandran
- Joan Morgan
- Justin Pirtle
- Allen Moheimani
- Garvit Singh
- Bassem Wanis

# Revisioni

| Data | Modifica |
| --- | --- |
| novembre 2021 | Rilascio iniziale |
| Settembre 2022 | Versione 1.1: Incremento automatico del contatore di servizio basato sulle posizioni di coda scadute. Trasferisci parte dell'utilizzo di Elasticache (Redis) OSS su DynamoDB. APIEndpoint pubblico per ottenere il tempo di scadenza della posizione di coda rimanente . Per ulteriori informazioni, consulta il CHANGELOGfile.md nel repository. GitHub |
| Aprile 2023 | Versione 1.1.1: impatto mitigato causato dalle nuove impostazioni predefinite per S3 Object Ownership (ACLsdisabilitate) per tutti i nuovi bucket S3. Per ulteriori informazioni, consulta il CHANGELOGfile.md nel repository. GitHub |
| Novembre 2023 | Versione 1.1.2: versioni aggiornate dei pacchetti per risolvere le vulnerabilità di sicurezza. Per ulteriori informazioni, consulta il CHANGELOGfile.md nel repository. GitHub |
| Marzo 2024 | Versione 1.1.3: sono stati risolti tre problemi: posizioni di coda scadute che persistev ano nelle dimensioni della sala d'attesa, `queue_num` API restituzione di vecchi risultati anche dopo un ripristino e guasti intermittenti negli adattatori OpenID. `/userInfo` API Per ulteriori informazioni, consulta il file.md nel repository. CHANGELOG GitHub |
| aprile 2024 | Versione 1.1.4: versioni aggiornate dei pacchetti per risolvere le vulnerabilità di |

| Data | Modifica |
| --- | --- |
| | sicurezza. Per ulteriori informazioni, consulta il CHANGELOGfile.md nel repository. GitHub |
| Giugno 2024 | Versione 1.1.5: versioni aggiornate dei pacchetti per risolvere le vulnerabilità di sicurezza. Per ulteriori informazioni, consulta il CHANGELOGfile.md nel repository. GitHub |
| agosto 2024 | Versione 1.1.6: versioni aggiornate dei pacchetti per risolvere le vulnerabilità di sicurezza. Per ulteriori informazioni, consulta il CHANGELOGfile.md nel repository. GitHub |
| agosto 2024 | Versione 1.1.7: versioni aggiornate dei pacchetti per risolvere le vulnerabilità di sicurezza. Per ulteriori informazioni, consulta il CHANGELOGfile.md nel repository. GitHub |
| Settembre 2024 | Versione 1.1.8: versioni aggiornate dei pacchetti per risolvere le vulnerabilità di sicurezza. Per ulteriori informazioni, consulta il CHANGELOGfile.md nel repository. GitHub |
| Novembre 2024 | Versione 1.1.9: versioni aggiornate dei pacchetti per risolvere le vulnerabilità di sicurezza. Per ulteriori informazioni, consulta il CHANGELOGfile.md nel repository. GitHub |
| Novembre 2024 | Versione 1.1.10: versioni aggiornate dei pacchetti per risolvere le vulnerabilità di sicurezza. Per ulteriori informazioni, consulta il CHANGELOGfile.md nel repository. GitHub |

# Note

I clienti sono responsabili della propria valutazione indipendente delle informazioni contenute in questo documento. Questo documento: (a) è solo a scopo informativo, (b) rappresenta le AWS attuali offerte e pratiche di prodotto, che sono soggette a modifiche senza preavviso, e (c) non crea alcun impegno o assicurazione da parte delle sue affiliate, fornitori o AWS licenzianti. AWS i prodotti o i servizi sono forniti «così come sono» senza garanzie, dichiarazioni o condizioni di alcun tipo, esplicite o implicite. AWS le responsabilità nei confronti dei propri clienti sono regolate da AWS accordi e il presente documento non fa parte di, né modifica, alcun accordo tra AWS e i suoi clienti.

Virtual Waiting Room on AWS è concesso in licenza secondo i termini della licenza Apache versione 2.0.

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.