Guida per l'utente

AWS Messaggistica push per l'utente finale

AWS Messaggistica push per l'utente finale: Guida per l'utente

Table of Contents

Che cos'è la messaggistica push per l'utente AWS finale?

> **Note**
>
> Le funzionalità di notifica push di Amazon Pinpoint sono ora denominate AWS End User Messaging.

Con AWS End User Messaging Push, puoi coinvolgere gli utenti delle tue app inviando notifiche push tramite un canale di notifica push. Supportiamo Apple Push Notification Service (APNs), Firebase Cloud Messaging (FCM), Amazon Device Messaging (ADM) e Baidu Push.

Argomenti

Sei un utente AWS End User Messaging Push per la prima volta?

Se sei un utente alle prime armi di AWS End User Messaging Push, ti consigliamo di iniziare leggendo le seguenti sezioni:

- Configurare un Account AWS
- Guida introduttiva a AWS End User Messaging Push
- Creazione di un'applicazione e attivazione dei canali push

Funzionalità di AWS End User Messaging Push

È possibile inviare notifiche push ad app utilizzando canali distinti per i servizi di notifiche push seguenti:

- Firebase Cloud Messaging () FCM
- Servizio Apple Push Notification () APNs

> **Note**
>
> Puoi utilizzarlo APNs per inviare messaggi a dispositivi iOS come iPhones eiPads, oltre che al browser Safari su dispositivi macOS, come laptop e desktop Mac.

- Baidu Cloud Push
- Messaggistica per dispositivi Amazon (ADM)

Accesso AWS alla messaggistica push per l'utente finale

Spiega brevemente i diversi modi per accedere al servizioCLI, tramite console oAPI.

È possibile gestire AWS End User Messaging Push utilizzando le seguenti interfacce:

AWS Console End User Messaging Push

L'interfaccia web in cui è possibile creare e gestire le risorse AWS End User Messaging Push. Se ti sei registrato a Account AWS, puoi accedere alla console AWS End User Messaging Push da AWS Management Console.

AWS Command Line Interface

Interagisci con i AWS servizi utilizzando i comandi nella shell della riga di comando. AWS Command Line Interface È supportato su Windows, macOS e Linux. Per ulteriori informazioni su AWS CLI, vedere la Guida per AWS Command Line Interface l'utente. I comandi AWS End User Messaging Push sono disponibili nel AWS CLI Command Reference.

AWS SDKs

Se sei uno sviluppatore di software che preferisce creare applicazioni utilizzando specifiche lingue APIs anziché inviare una richiesta su HTTP oHTTPS, AWS fornisce librerie, codice di esempio, tutorial e altre risorse. Queste librerie forniscono funzioni di base che automatizzano le attività, come la firma crittografica delle richieste, il ritentativo delle richieste e la gestione delle risposte agli errori. Queste funzioni contribuiscono a rendere più efficiente l'avvio. Per ulteriori informazioni, consulta Strumenti per creare in AWS.

Disponibilità regionale

AWS End User Messaging Push è disponibile Regioni AWS in diversi paesi in Nord America, Europa, Asia e Oceania. In ogni regione, AWS mantiene più zone di disponibilità. Queste zone di disponibilità sono fisicamente isolate l'una dall'altra, ma sono unite da connessioni di rete private a bassa latenza, a velocità effettiva elevata e altamente ridondanti. Queste zone di disponibilità vengono utilizzate per fornire livelli molto elevati di disponibilità e ridondanza, riducendo al minimo la latenza.

Per ulteriori informazioni Regioni AWS, consulta Specificare quali contenuti Regioni AWS il tuo account può utilizzare in. Riferimenti generali di Amazon Web Services Per un elenco di tutte le regioni in cui è attualmente disponibile AWS End User Messaging Push e l'endpoint per ciascuna regione, consulta Endpoints e quote per Amazon Pinpoint API and AWS service endpoint in. Riferimenti generali di Amazon Web Services Per ulteriori informazioni sul numero di zone di disponibilità presenti in ciascuna regione, consulta Infrastruttura globale AWS.

Configurare un Account AWS

Prima di poter usare AWS End User Messaging Push Per inviare notifiche push alla tua app, devi prima ottenere un Account AWS con IAM autorizzazioni sufficienti. Questo Account AWS può essere utilizzato anche per altri servizi nel AWS ecosistema.

Argomenti

Iscriviti per un Account AWS

Se non hai un Account AWS, completa i seguenti passaggi per crearne uno.

Per iscriverti a un Account AWS

1. Apri la https://portal.aws.amazon.com/billing/registrazione.
2. Segui le istruzioni online.

   Nel corso della procedura di registrazione riceverai una telefonata, durante la quale sarà necessario inserire un codice di verifica attraverso la tastiera del telefono.

   Quando ti iscrivi a un Account AWS, un Utente root dell'account AWSviene creato. L'utente root ha accesso a tutti Servizi AWS e le risorse presenti nell'account. Come best practice di sicurezza, assegna l'accesso amministrativo a un utente e utilizza solo l'utente root per eseguire attività che richiedono l'accesso di un utente root.

AWS ti invia un'email di conferma dopo il completamento della procedura di registrazione. In qualsiasi momento, puoi visualizzare l'attività corrente del tuo account e gestirlo accedendo a https://aws.amazon.com/e scegliendo Il mio account.

Crea un utente con accesso amministrativo

Dopo esserti registrato per un Account AWS, proteggi il tuo Utente root dell'account AWS, abilita AWS IAM Identity Center e crea un utente amministrativo in modo da non utilizzare l'utente root per le attività quotidiane.

Proteggi i tuoi Utente root dell'account AWS

1. Accedi a AWS Management Consolecome proprietario dell'account selezionando Utente root e inserendo il Account AWS indirizzo email. Nella pagina successiva, inserisci la password.

   Per informazioni sull'accesso tramite utente root, consulta Accesso come utente root in Accedi ad AWS Guida per l'utente.
2. Attiva l'autenticazione a più fattori (MFA) per il tuo utente root.

   Per istruzioni, consulta Abilitare un MFA dispositivo virtuale per il Account AWS utente root (console) nella Guida per l'IAMutente.

Crea un utente con accesso amministrativo

1. Abilita IAM Identity Center.

   Per istruzioni, vedi Abilitazione AWS IAM Identity Center nella AWS IAM Identity Center Guida per l'utente.
2. In IAM Identity Center, concedi l'accesso amministrativo a un utente.

   Per un tutorial sull'utilizzo di IAM Identity Center directory come fonte di identità, vedi Configurare l'accesso utente con l'impostazione predefinita IAM Identity Center directory nella AWS IAM Identity Center Guida per l'utente.

Accesso come utente amministratore

- Per accedere con il tuo utente IAM Identity Center, utilizza l'accesso URL che ti è stato inviato al tuo indirizzo e-mail quando hai creato l'utente IAM Identity Center.

  Per informazioni sull'accesso tramite un utente di IAM Identity Center, vedi Accesso a AWS accedere al portale in Accedi ad AWS Guida per l'utente.

Assegna l'accesso a ulteriori utenti

1. In IAM Identity Center, crea un set di autorizzazioni che segua la migliore pratica di applicazione delle autorizzazioni con privilegi minimi.

Per istruzioni, vedere Creare un set di autorizzazioni nella AWS IAM Identity Center Guida per l'utente.

2. Assegna al gruppo prima gli utenti e poi l'accesso con autenticazione unica (Single Sign-On).

   Per istruzioni, consulta Aggiungere gruppi nella AWS IAM Identity Center Guida per l'utente.

Guida introduttiva a AWS End User Messaging Push

Per configurare AWS End User Messaging Push in modo che possa inviare notifiche push alle tue app, devi prima fornire le credenziali che autorizzano AWS End User Messaging Push a inviare messaggi alla tua app. Le credenziali fornite dipendono dal sistema di notifica push che utilizzi:

- Per le credenziali del servizio Apple Push Notification (APN), consulta Ottenere una chiave di crittografia e un ID di chiave da Apple e Ottenere un certificato di provider da Apple nella documentazione per sviluppatori Apple.
- Per le credenziali di Firebase Cloud Messaging (FCM) che possono essere ottenute tramite la console Firebase, vedi Firebase Cloud Messaging.
- Per le credenziali Baidu, vedi Baidu.
- Per le credenziali di Amazon Device Messaging (ADM), consulta Ottenere credenziali.

Creazione di un'applicazione e attivazione dei canali push

Prima di poter utilizzare AWS End User Messaging Push per inviare notifiche push, devi prima creare un'applicazione e abilitare il canale delle notifiche push.

Contestuale

Applicazione

Un'applicazione è un contenitore di archiviazione per tutte le impostazioni AWS End User Messaging Push. L'applicazione memorizza anche le impostazioni dei canali, delle campagne e dei percorsi di Amazon Pinpoint.

Chiave

Una chiave di firma privata utilizzata da AWS End User Messaging Push per firmare crittograficamente i token di autenticazione. APNs Si può ottenere la chiave di firma dal proprio account sviluppatore Apple.

Se fornisci una chiave di firma,AWS End User Messaging Push utilizza un token con cui autenticarsi APNs per ogni notifica push che invii. Con la chiave di firma, puoi inviare notifiche push agli ambienti di APNs produzione e sandbox.

A differenza dei certificati, la chiave di firma non scade. La chiave viene fornita una sola volta e non è necessario rinnovarla. È possibile utilizzare la stessa chiave di firma per più app. Per ulteriori informazioni, consulta Comunicare APNs utilizzando i token di autenticazione nell'Aiuto di Xcode.

Certificate

Un TLS certificato che AWS End User Messaging Push utilizza per l'autenticazione APNs quando invii notifiche push. Un APNs certificato può supportare sia gli ambienti di produzione che quelli sandbox oppure può supportare solo l'ambiente sandbox. Si può ottenere il certificato dal proprio account sviluppatore Apple.

Un certificato scade dopo un anno. Quando ciò accade, è necessario creare un nuovo certificato, da fornire quindi a AWS End User Messaging Push per rinnovare l'invio delle notifiche push. Per ulteriori informazioni, consulta Comunicare APNs utilizzando un TLS certificato nella Guida di Xcode.

Prerequisiti

Prima di poter utilizzare qualsiasi canale push, sono necessarie credenziali valide per il servizio push. Per ulteriori informazioni sull'ottenimento delle credenziali, consulta. Guida introduttiva a AWS End User Messaging Push

Procedura

Segui queste istruzioni per creare un'applicazione e abilitare uno qualsiasi dei canali push. Per completare questa procedura è necessario solo inserire il nome di un'applicazione. È possibile abilitare o disabilitare qualsiasi canale push in un secondo momento.

1. Apri la console AWS End User Messaging Push all'indirizzo https://console.aws.amazon.com/push-notifications/.
2. Scegli Crea applicazione.
3. Per il nome dell'applicazione, inserisci il nome dell'applicazione.
4. (Facoltativo) Segui questo passaggio opzionale per abilitare il servizio Apple Push Notification (APNs).

    a. Per il servizio Apple Push Notification (APNs), seleziona Abilita.

    b. Per il tipo di autenticazione predefinito, scegli una delle seguenti opzioni:

        i. Se scegli Credenziali chiave, fornisci le seguenti informazioni dal tuo account sviluppatore Apple. AWS End User Messaging Push richiede queste informazioni per creare token di autenticazione.

            - ID chiave: ID assegnato alla chiave di firma.
            - Identificatore del bundle: ID assegnato all'app iOS.
            - Identificatore del team: ID assegnato al team dell'account sviluppatore Apple.
            - Chiave di autenticazione: file .p8 scaricato dall'account sviluppatore Apple quando crei una chiave di autenticazione.

        ii. Se si sceglie Certificate credentials (Credenziali certificato), è necessario fornire le seguenti informazioni:

            - SSLcertificato: il file.p12 per il certificato. TLS
            - Password certificato: se hai assegnato una password al certificato, immettila qui.

- Tipo di certificato: seleziona il tipo di certificato da utilizzare.

5. (Facoltativo) Segui questo passaggio opzionale per abilitare Firebase Cloud Messaging (). FCM

   a. Per Firebase Cloud Messaging () FCM seleziona Abilita.
   b. Per il tipo di autenticazione predefinito, scegli una delle seguenti opzioni:

      i. Per le credenziali del token (consigliato) scegli Scegli i file, quindi scegli il tuo JSON file di servizio.
      ii. Per le credenziali chiave, inserisci la chiave nella API chiave.

6. (Facoltativo) Segui questo passaggio opzionale per abilitare Baidu Cloud Push.

   a. Per Baidu Cloud Push seleziona Abilita.
   b. Per APIchiave inserisci la tua API chiave.
   c. Per chiave segreta inserisci la tua chiave segreta.

7. (Facoltativo) Segui questo passaggio facoltativo per abilitare Amazon Device Messaging.

   a. Per Amazon Device Messaging seleziona Abilita.
   b. Per Client ID inserisci il tuo ID cliente.
   c. Per Client secret, inserisci il tuo client secret.

8. Scegli Crea applicazione.

Disabilitazione dei canali push

Segui queste istruzioni per disabilitare uno qualsiasi dei canali push.

1. Apri la console AWS End User Messaging Push all'indirizzo https://console.aws.amazon.com/push-notifications/.
2. Scegliete l'applicazione che contiene le vostre credenziali push.
3. (Facoltativo) Per il servizio Apple Push Notification (APNs), deselezionate Abilita.
4. (Facoltativo) Per Firebase Cloud Messaging (FCM), deselezionate Abilita.
5. (Facoltativo) Per Baidu Cloud Push, seleziona Enable.
6. (Facoltativo) Per Amazon Device Messaging, seleziona Abilita.
7. Scegli Save changes (Salva modifiche).

Invio di un messaggio

L' AWS End User Messaging Push API può inviare notifiche push transazionali a identificatori di dispositivi specifici. Questa sezione contiene esempi di codice completi che è possibile utilizzare per inviare notifiche push tramite AWS End User Messaging Push API utilizzando un. AWS SDK

È possibile utilizzare questi esempi per inviare notifiche push tramite qualsiasi servizio di notifica push supportato da AWS End User Messaging Push. Attualmente, AWS End User Messaging Push supporta i seguenti canali: Firebase Cloud Messaging (FCM), Apple Push Notification Service (APNs), Baidu Cloud Push e Amazon Device Messaging (). ADM

Per altri esempi di codice su endpoint, segmenti e canali, consulta Esempi di codice.

> **Note**
>
> Quando invii notifiche push tramite il servizio Firebase Cloud Messaging (FCM), usa il nome del servizio `GCM` nella chiamata all' AWS End User Messaging Push. API Il servizio Google Cloud Messaging (GCM) è stato interrotto da Google il 10 aprile 2018. Tuttavia, AWS End User Messaging Push API utilizza il nome del `GCM` servizio per i messaggi inviati tramite il FCM servizio al fine di mantenere la compatibilità con il API codice scritto prima dell'interruzione del servizio. GCM

GCM (AWS CLI)

L'esempio seguente utilizza send-messages per inviare una GCM notifica Push con. AWS CLI Replace (Sostituisci) *token* con il token univoco del dispositivo e *611e3e3cdd47474c9c1399a50example* con l'identificatore dell'applicazione.

```
aws pinpoint send-messages \
--application-id 611e3e3cdd47474c9c1399a50example \
--message-request file://myfile.json \
--region us-west-2

Contents of myfile.json:
{
  "Addresses": {
    "token": {
```

```
      "ChannelType" : 'GCM'
    }
  },
  "MessageConfiguration": {
    "GCMMessage": {
      "Action": "URL",
      "Body": "This is a sample message",
      "Priority": "normal",
      "SilentPush": True,
      "Title": "My sample message",
      "TimeToLive": 30,
      "Url": "https://www.example.com"
      }
  }
}
```

L'esempio seguente utilizza send-messages per inviare una notifica GCM push, utilizzando tutte le chiavi legacy, con. AWS CLI Replace (Sostituisci) *token* con il token univoco del dispositivo e *611e3e3cdd47474c9c1399a50example* con l'identificatore dell'applicazione.

```
aws pinpoint send-messages \
--application-id 611e3e3cdd47474c9c1399a50example \
--message-request
'{
  "MessageConfiguration": {
    "GCMMessage":{
      "RawContent": "{\"notification\": {\n \"title\": \"string\",\n \"body\":
 \"string\",\n \"android_channel_id\": \"string\",\n \"body_loc_args\": [\n \"string
\"\n ],\n \"body_loc_key\": \"string\",\n \"click_action\": \"string\",\n \"color\":
 \"string\",\n \"icon\": \"string\",\n \"sound\": \"string\",\n \"tag\": \"string
\",\n \"title_loc_args\": [\n \"string\"\n ],\n \"title_loc_key\": \"string\"\n },
\"data\":{\"message\":\"hello in data\"} }",
      "TimeToLive" : 309744
     }
   },
  "Addresses": {
    "token": {
      "ChannelType":"GCM"
      }
   }
}'
\ --region us-east-1
```

L'esempio seguente utilizza send-messages per inviare una notifica GCM push con payload di FCMv1 messaggi utilizzando. AWS CLI Replace (Sostituisci) *token* con il token univoco del dispositivo e *611e3e3cdd47474c9c1399a50example* con l'identificatore dell'applicazione.

```
aws pinpoint send-messages \
--application-id 6a2dafd84bec449ea75fb773f4c41fa1 \
--message-request
'{
  "MessageConfiguration": {
    "GCMMessage":{
      "RawContent": "{\n \"fcmV1Message\": \n {\n \"message\" :{\n  \"notification
\": {\n \"title\": \"string\",\n \"body\": \"string\"\n },\n \"android\": {\n
 \"priority\": \"high\",\n \"notification\": {\n \"title\": \"string\",\n \"body
\": \"string\",\n \"icon\": \"string\",\n \"color\": \"string\",\n \"sound\":
 \"string\",\n \"tag\": \"string\",\n \"click_action\": \"string\",\n \"body_loc_key
\": \"string\",\n \"body_loc_args\": [\n \"string\"\n ],\n \"title_loc_key
\": \"string\",\n \"title_loc_args\": [\n \"string\"\n ],\n \"channel_id\":
 \"string\",\n \"ticker\": \"string\",\n \"sticky\": true,\n \"event_time\":
 \"2024-02-06T22:11:55Z\",\n \"local_only\": true,\n \"notification_priority\":
 \"PRIORITY_UNSPECIFIED\",\n \"default_sound\": false,\n \"default_vibrate_timings
\": true,\n \"default_light_settings\": false,\n \"vibrate_timings\": [\n \"22s
\"\n ],\n \"visibility\": \"VISIBILITY_UNSPECIFIED\",\n \"notification_count\": 5,
\n \"light_settings\": {\n \"color\": {\n \"red\": 1,\n \"green\": 2,\n \"blue\":
 3,\n \"alpha\": 6\n },\n \"light_on_duration\": \"112s\",\n \"light_off_duration
\": \"1123s\"\n },\n \"image\": \"string\"\n },\n \"data\": {\n \"dataKey1\":
 \"priority message\",\n \"data_key_3\": \"priority message\",\n \"dataKey2\":
 \"priority message\",\n \"data_key_5\": \"priority message\"\n },\n \"ttl\":
 \"10023.32s\"\n },\n \"apns\": {\n \"payload\": {\n \"aps\": {\n \"alert\": {\n
 \"subtitle\": \"string\",\n \"title-loc-args\": [\n \"string\"\n ],\n \"title-loc-
key\": \"string\",\n \"launch-image\": \"string\",\n \"subtitle-loc-key\": \"string
\",\n \"subtitle-loc-args\": [\n \"string\"\n ],\n \"loc-args\": [\n \"string
\"\n ],\n \"loc-key\": \"string\",\n \"title\": \"string\",\n \"body\": \"string
\"\n },\n \"thread-id\": \"string\",\n \"category\": \"string\",\n \"content-
available\": 1,\n \"mutable-content\": 1,\n \"target-content-id\": \"string\",\n
 \"interruption-level\": \"string\",\n \"relevance-score\": 25,\n \"filter-criteria
\": \"string\",\n \"stale-date\": 6483,\n \"content-state\": {},\n \"timestamp\":
 673634,\n \"dismissal-date\": 4,\n \"attributes-type\": \"string\",\n \"attributes
\": {},\n \"sound\": \"string\",\n \"badge\": 5\n }\n }\n },\n \"webpush\": {\n
 \"notification\": {\n \"permission\": \"granted\",\n \"maxActions\": 2,\n \"actions
\": [\n \"title\"\n ],\n \"badge\": \"URL\",\n \"body\": \"Hello\",\n \"data\": {\n
 \"hello\": \"hey\"\n },\n \"dir\": \"auto\",\n \"icon\": \"icon\",\n \"image\":
 \"image\",\n \"lang\": \"string\",\n \"renotify\": false,\n \"requireInteraction\":
 true,\n \"silent\": false,\n \"tag\": \"tag\",\n \"timestamp\": 1707259524964,\n
```

```
 \"title\": \"hello\",\n \"vibrate\": [\n 100,\n 200,\n 300\n ]\n },\n \"data\": {\n
 \"data1\": \"priority message\",\n \"data2\": \"priority message\",\n \"data12\":
 \"priority message\",\n \"data3\": \"priority message\"\n }\n },\n \"data\": {\n
 \"data7\": \"priority message\",\n \"data5\": \"priority message\",\n \"data8\":
 \"priority message\",\n \"data9\": \"priority message\"\n }\n }\n \n}\n}",
      "TimeToLive" : 309744
    }
  },
  "Addresses": {
    token: {
      "ChannelType":"GCM"
    }
   }
}'
\ --region us-east-1
```

se si utilizza `ImageUrl` field forGCM, pinpoint invia il campo come notifica dei dati, con la chiave `pinpoint.notification.imageUrl` indicata, il che può impedire il rendering dell'immagine immediatamente. Utilizza RawContent o aggiungi la gestione delle chiavi dati, ad esempio l'integrazione della tua app con. AWS Amplify

Safari (AWS CLI)

Puoi utilizzare AWS End User Messaging Push per inviare messaggi a computer macOS che utilizzano il browser web Safari di Apple. Per inviare un messaggio al browser Safari, devi specificare il contenuto del messaggio in formato RAW e includere un attributo specifico nel payload del messaggio. Puoi farlo creando un modello di notifica push con un payload di messaggi non elaborati o specificando il contenuto del messaggio non elaborato direttamente in un messaggio della campagna, nella Amazon Pinpoint User Guide.

> **Note**
>
> Questo attributo speciale è necessario per l'invio a computer laptop e desktop macOS che utilizzano il browser Web Safari. Non è necessario per l'invio a dispositivi iOS come iPhones eiPads.

Per inviare un messaggio ai browser web Safari, devi specificare il payload del messaggio in formato RAW. Il payload dei messaggi in formato RAW deve includere un array `url-args` all'interno dell'oggetto `aps`. L'array `url-args` è necessario per inviare notifiche push al browser Web Safari. Tuttavia, è accettabile che l'array contenga un singolo elemento vuoto.

L'esempio seguente utilizza send-messages per inviare una notifica al browser web Safari con. AWS CLI Replace (Sostituisci) *token* con il token univoco del dispositivo e *611e3e3cdd47474c9c1399a50example* con l'identificatore dell'applicazione.

```
aws pinpoint send-messages \
--application-id 611e3e3cdd47474c9c1399a50example \
--message-request
'{
  "Addresses": {
    "token":
    {
      "ChannelType":"APNS"
    }
  },
  "MessageConfiguration": {
    "APNSMessage": {
        "RawContent":
          "{\"aps\": {\"alert\": { \"title\": \"Title of my message\", \"body\":
 \"This is a push notification for the Safari web browser.\"},\"content-available\":
 1,\"url-args\": [\"\"]}}"
     }
  }
}'
\ --region us-east-1
```

Per ulteriori informazioni sulle notifiche push di Safari, consulta l'argomento relativo alla configurazione delle notifiche push di Safari sul sito Web di Apple per gli sviluppatori.

APNS (AWS CLI)

L'esempio seguente utilizza send-messages per inviare una notifica APNS push con. AWS CLI Replace (Sostituisci) *token* con il token univoco del dispositivo, *611e3e3cdd47474c9c1399a50example* con l'identificatore dell'applicazione e *GAME_INVITATION* con un identificatore univoco.

```
aws pinpoint send-messages \
--application-id 611e3e3cdd47474c9c1399a50example \
--message-request
'{
    "Addresses": {
     "token":
    {
      "ChannelType":"APNS"
```

```
    }
  },
  "MessageConfiguration": {
    "APNSMessage": {
      "RawContent": "{\"aps\" : {\"alert\" : {\"title\" : \"Game Request\",
\"subtitle\" : \"Five Card Draw\",\"body\" : \"Bob wants to play poker\"},\"category
\" : \"GAME_INVITATION\"},\"gameID\" : \"12345678\"}"
      }
    }
}'
\ --region us-east-1
```

JavaScript (Node.js)

Utilizza questo esempio per inviare notifiche push utilizzando il modulo AWS SDK JavaScript in Node.js. L'esempio presuppone che tu abbia già installato e configurato SDK il modulo JavaScript in Node.js.

Questo esempio presuppone anche che tu stia utilizzando un file di credenziali condivise per specificare la chiave di accesso e la chiave di accesso segreta per un utente esistente. Per ulteriori informazioni, vedere Impostazione delle credenziali nel modulo JavaScript in Node.js Developer Guide.AWS SDK

```
'use strict';

const AWS = require('aws-sdk');

// The AWS Region that you want to use to send the message. For a list of
// AWS Regions where the API is available
const region = 'us-east-1';

// The title that appears at the top of the push notification.
var title = 'Test message sent from End User Messaging Push.';

// The content of the push notification.
var message = 'This is a sample message sent from End User Messaging Push by using
 the '
            + 'AWS SDK for JavaScript in Node.js';

// The application ID that you want to use when you send this
// message. Make sure that the push channel is enabled for the project that
// you choose.
var applicationId = 'ce796be37f32f178af652b26eexample';
```

```
// An object that contains the unique token of the device that you want to send
// the message to, and the push service that you want to use to send the message.
var recipient = {
  'token': 'a0b1c2d3e4f5g6h7i8j9k0l1m2n3o4p5q6r7s8t9u0v1w2x3y4z5a6b7c8d8e9f0',
  'service': 'GCM'
  };

// The action that should occur when the recipient taps the message. Possible
// values are OPEN_APP (opens the app or brings it to the foreground),
// DEEP_LINK (opens the app to a specific page or interface), or URL (opens a
// specific URL in the device's web browser.)
var action = 'URL';

// This value is only required if you use the URL action. This variable contains
// the URL that opens in the recipient's web browser.
var url = 'https://www.example.com';

// The priority of the push notification. If the value is 'normal', then the
// delivery of the message is optimized for battery usage on the recipient's
// device, and could be delayed. If the value is 'high', then the notification is
// sent immediately, and might wake a sleeping device.
var priority = 'normal';

// The amount of time, in seconds, that the push notification service provider
// (such as FCM or APNS) should attempt to deliver the message before dropping
// it. Not all providers allow you specify a TTL value.
var ttl = 30;

// Boolean that specifies whether the notification is sent as a silent
// notification (a notification that doesn't display on the recipient's device).
var silent = false;

function CreateMessageRequest() {
  var token = recipient['token'];
  var service = recipient['service'];
  if (service == 'GCM') {
    var messageRequest = {
      'Addresses': {
        [token]: {
          'ChannelType' : 'GCM'
        }
      },
      'MessageConfiguration': {
```

```
        'GCMMessage': {
          'Action': action,
          'Body': message,
          'Priority': priority,
          'SilentPush': silent,
          'Title': title,
          'TimeToLive': ttl,
          'Url': url
        }
      }
    };
  } else if (service == 'APNS') {
    var messageRequest = {
      'Addresses': {
        [token]: {
          'ChannelType' : 'APNS'
        }
      },
      'MessageConfiguration': {
        'APNSMessage': {
          'Action': action,
          'Body': message,
          'Priority': priority,
          'SilentPush': silent,
          'Title': title,
          'TimeToLive': ttl,
          'Url': url
        }
      }
    };
  } else if (service == 'BAIDU') {
    var messageRequest = {
      'Addresses': {
        [token]: {
          'ChannelType' : 'BAIDU'
        }
      },
      'MessageConfiguration': {
        'BaiduMessage': {
          'Action': action,
          'Body': message,
          'SilentPush': silent,
          'Title': title,
          'TimeToLive': ttl,
```

```
          'Url': url
        }
      }
    };
  } else if (service == 'ADM') {
    var messageRequest = {
      'Addresses': {
        [token]: {
          'ChannelType' : 'ADM'
        }
      },
      'MessageConfiguration': {
        'ADMMessage': {
          'Action': action,
          'Body': message,
          'SilentPush': silent,
          'Title': title,
          'Url': url
        }
      }
    };
  }

  return messageRequest
}

function ShowOutput(data){
  if (data["MessageResponse"]["Result"][recipient["token"]]["DeliveryStatus"]
      == "SUCCESSFUL") {
    var status = "Message sent! Response information: ";
  } else {
    var status = "The message wasn't sent. Response information: ";
  }
  console.log(status);
  console.dir(data, { depth: null });
}

function SendMessage() {
  var token = recipient['token'];
  var service = recipient['service'];
  var messageRequest = CreateMessageRequest();

  // Specify that you're using a shared credentials file, and specify the
  // IAM profile to use.
```

```
  var credentials = new AWS.SharedIniFileCredentials({ profile: 'default' });
  AWS.config.credentials = credentials;

  // Specify the AWS Region to use.
  AWS.config.update({ region: region });

  //Create a new Pinpoint object.
  var pinpoint = new AWS.Pinpoint();
  var params = {
    "ApplicationId": applicationId,
    "MessageRequest": messageRequest
  };

  // Try to send the message.
  pinpoint.sendMessages(params, function(err, data) {
    if (err) console.log(err);
    else     ShowOutput(data);
  });
}

SendMessage()
```

Python

Utilizza questo esempio per inviare notifiche push utilizzando AWS SDK for Python (Boto3). Questo esempio presuppone che tu abbia già installato e configurato SDK for Python (Boto3).

Questo esempio presuppone anche che tu stia utilizzando un file di credenziali condivise per specificare la chiave di accesso e la chiave di accesso segreta per un utente esistente. Per ulteriori informazioni, vedete Credenziali nel riferimento AWS SDKfor Python (APIBoto3).

```
import json
import boto3
from botocore.exceptions import ClientError

# The AWS Region that you want to use to send the message. For a list of
# AWS Regions where the API is available
region = "us-east-1"

# The title that appears at the top of the push notification.
title = "Test message sent from End User Messaging Push."

# The content of the push notification.
```

```
message = ("This is a sample message sent from End User Messaging Push by using the
 "
           "AWS SDK for Python (Boto3).")

# The application ID to use when you send this message.
# Make sure that the push channel is enabled for the project or application
# that you choose.
application_id = "ce796be37f32f178af652b26eexample"

# A dictionary that contains the unique token of the device that you want to send
 the
# message to, and the push service that you want to use to send the message.
recipient = {
    "token": "a0b1c2d3e4f5g6h7i8j9k0l1m2n3o4p5q6r7s8t9u0v1w2x3y4z5a6b7c8d8e9f0",
    "service": "GCM"
    }

# The action that should occur when the recipient taps the message. Possible
# values are OPEN_APP (opens the app or brings it to the foreground),
# DEEP_LINK (opens the app to a specific page or interface), or URL (opens a
# specific URL in the device's web browser.)
action = "URL"

# This value is only required if you use the URL action. This variable contains
# the URL that opens in the recipient's web browser.
url = "https://www.example.com"

# The priority of the push notification. If the value is 'normal', then the
# delivery of the message is optimized for battery usage on the recipient's
# device, and could be delayed. If the value is 'high', then the notification is
# sent immediately, and might wake a sleeping device.
priority = "normal"

# The amount of time, in seconds, that the push notification service provider
# (such as FCM or APNS) should attempt to deliver the message before dropping
# it. Not all providers allow you specify a TTL value.
ttl = 30

# Boolean that specifies whether the notification is sent as a silent
# notification (a notification that doesn't display on the recipient's device).
silent = False

# Set the MessageType based on the values in the recipient variable.
def create_message_request():
```

```
    token = recipient["token"]
    service = recipient["service"]

    if service == "GCM":
        message_request = {
            'Addresses': {
                token: {
                    'ChannelType': 'GCM'
                }
            },
            'MessageConfiguration': {
                'GCMMessage': {
                    'Action': action,
                    'Body': message,
                    'Priority' : priority,
                    'SilentPush': silent,
                    'Title': title,
                    'TimeToLive': ttl,
                    'Url': url
                }
            }
        }
    elif service == "APNS":
        message_request = {
            'Addresses': {
                token: {
                    'ChannelType': 'APNS'
                }
            },
            'MessageConfiguration': {
                'APNSMessage': {
                    'Action': action,
                    'Body': message,
                    'Priority' : priority,
                    'SilentPush': silent,
                    'Title': title,
                    'TimeToLive': ttl,
                    'Url': url
                }
            }
        }
    elif service == "BAIDU":
        message_request = {
```

```
            'Addresses': {
                token: {
                    'ChannelType': 'BAIDU'
                }
            },
            'MessageConfiguration': {
                'BaiduMessage': {
                    'Action': action,
                    'Body': message,
                    'SilentPush': silent,
                    'Title': title,
                    'TimeToLive': ttl,
                'Url': url
                }
            }
        }
    elif service == "ADM":
        message_request = {
            'Addresses': {
                token: {
                    'ChannelType': 'ADM'
                }
            },
            'MessageConfiguration': {
                'ADMMessage': {
                    'Action': action,
                    'Body': message,
                    'SilentPush': silent,
                    'Title': title,
                    'Url': url
                }
            }
        }
    else:
        message_request = None

    return message_request

# Show a success or failure message, and provide the response from the API.
def show_output(response):
    if response['MessageResponse']['Result'][recipient["token"]]['DeliveryStatus']
 == "SUCCESSFUL":
        status = "Message sent! Response information:\n"
    else:
```

```
        status = "The message wasn't sent. Response information:\n"
    print(status, json.dumps(response,indent=4))

# Send the message through the appropriate channel.
def send_message():

    token = recipient["token"]
    service = recipient["service"]
    message_request = create_message_request()

    client = boto3.client('pinpoint',region_name=region)

    try:
        response = client.send_messages(
            ApplicationId=application_id,
            MessageRequest=message_request
        )
    except ClientError as e:
        print(e.response['Error']['Message'])
    else:
        show_output(response)

send_message()
```

Risorse aggiuntive

- Per ulteriori informazioni sui modelli di canali Push, consulta Creazione di modelli di notifica push nella Guida per l'utente di Amazon Pinpoint.

Ricezione di notifiche push nell'applicazione

I seguenti argomenti descrivono come modificare l'app Swift, Android, React Native o Flutter in modo che riceva notifiche push.

Argomenti

Configurazione delle notifiche push Swift

Le notifiche push per le app iOS vengono inviate utilizzando il servizio Apple Push Notification (APNs). Per poter inviare notifiche push ai dispositivi iOS, è necessario creare un ID app nel portale Apple Developer e creare i certificati richiesti. Puoi trovare ulteriori informazioni sul completamento di questi passaggi in Configurazione dei servizi di notifica push nella documentazione di AWS Amplify.

Lavorare con i token APNs

Come best practice, è consigliabile sviluppare l'app in modo che i token di dispositivo dei clienti vengano rigenerati quando l'app viene reinstallata.

Se un destinatario aggiorna il dispositivo a una nuova versione principale di iOS (ad esempio, da iOS 12 a iOS 13) e successivamente reinstalla l'app, questa genera un nuovo token. Se l'app non aggiorna il token, per inviare la notifica viene utilizzato il token precedente. Di conseguenza, il servizio Apple Push Notification (APNs) rifiuta la notifica, poiché il token ora non è valido. Quando tenti di inviare la notifica, ricevi un messaggio di notifica di errore da. APNs

Configurazione delle notifiche push per Android

Le notifiche push per le app Android vengono inviate utilizzando Firebase Cloud Messaging (FCM), che sostituisce Google Cloud Messaging (). GCM Prima di poter inviare notifiche push ai dispositivi

Android, devi ottenere le credenziali. FCM È quindi possibile utilizzare quelle credenziali per creare un progetto Android e avviare un'app di esempio in grado di ricevere notifiche push. Puoi trovare ulteriori informazioni sul completamento di questi passaggi nella sezione Notifiche push della documentazione di AWS Amplify.

Configurazione delle notifiche push di Flutter

Le notifiche push per le app Flutter vengono inviate utilizzando Firebase Cloud Messaging (FCM) per Android e per APNs iOS. Per ulteriori informazioni sull'esecuzione di questa procedura, consulta la sezione relativa alle notifiche push nella documentazione di AWS Amplify Flutter.

Configurazione delle notifiche push per React Native

Le notifiche push per le app React Native vengono inviate utilizzando Firebase Cloud Messaging (FCM) per Android e per APNs iOS. Puoi trovare ulteriori informazioni sul completamento di questi passaggi nella sezione Notifiche push della documentazione di AWS Amplify. JavaScript

Crea un'applicazione in AWS End User Messaging Push

Per iniziare a inviare notifiche push in AWS End User Messaging Push, devi creare un'applicazione. Quindi, è necessario abilitare i canali delle notifiche push da utilizzare fornendo le credenziali appropriate.

È possibile creare nuove applicazioni e configurare canali di notifica push utilizzando la console AWS End User Messaging Push. Per ulteriori informazioni, consulta Creazione di un'applicazione e attivazione dei canali push.

È inoltre possibile creare e configurare l'applicazione utilizzando il APIAWS SDK, an o il AWS Command Line Interface(AWS CLI). Per creare un'applicazione, utilizzate la `Apps` risorsa. Per configurare i canali delle notifiche push, usa le risorse seguenti:

- APNscanale per inviare messaggi agli utenti di dispositivi iOS utilizzando il servizio Apple Push Notification.
- ADMcanale per inviare messaggi agli utenti dei dispositivi Amazon Kindle Fire.
- Canale Baidu per l'invio di messaggi agli utenti di Baidu.
- GCMcanale per inviare messaggi a dispositivi Android utilizzando Firebase Cloud Messaging (FCM), che sostituisce Google Cloud Messaging (). GCM

Gestione delle notifiche push

Dopo aver ottenuto le credenziali necessarie per inviare notifiche push, puoi aggiornare l'applicazione in modo che sia in grado di ricevere notifiche push. Per ulteriori informazioni, consulta Notifiche push: Guida introduttiva nella documentazione. AWS Amplify

Eliminazione di un'applicazione

Questa procedura rimuove l'applicazione dall'account e tutte le risorse dell'applicazione.

Contestuale

Applicazione

Un'applicazione è un contenitore di archiviazione per tutte le impostazioni AWS End User Messaging Push. L'applicazione memorizza anche le impostazioni dei canali, delle campagne e dei percorsi di Amazon Pinpoint.

Procedura

1. Apri la console AWS End User Messaging Push all'indirizzo. https://console.aws.amazon.com/push-notifications/
2. Scegliete un'applicazione, quindi scegliete Elimina.
3. Nella finestra Elimina applicazione, inserisci **`delete`** e quindi scegli Elimina.

> **Important**
>
> Vengono eliminati anche tutti i canali, le campagne, i percorsi o i segmenti di Amazon Pinpoint.

Best practice

Anche quando operi nell'interesse dei clienti è possibile che si verifichino situazioni che impattano sull'efficienza del recapito dei tuoi messaggi. Le seguenti sezioni contengono raccomandazioni utili ad garantire che le comunicazioni e-mail raggiungano i destinatari previsti.

Invio di un volume elevato di notifiche push

Prima di inviare un volume elevato di notifiche push, assicurati che il tuo account sia configurato per supportare i tuoi requisiti di throughput. Per impostazione predefinita, tutti gli account sono configurati per inviare 25.000 messaggi al secondo. Se è necessario inviare più di 25.000 messaggi in un secondo, puoi richiedere un aumento della quota. Per ulteriori informazioni, consulta Quote per la messaggistica AWS push per l'utente finale.

Assicurati che il tuo account sia configurato correttamente con le credenziali di ciascuno dei provider di notifiche push che intendi utilizzare, ad FCM esempio o. APNs

Infine, elabora un modo per gestire le eccezioni. Ogni servizio di notifica push fornisce diversi messaggi di eccezione. Per gli invii transazionali, ricevi un codice di stato principale pari a 200 per la API chiamata, con un codice di stato per endpoint pari a 400 (errore permanente) se il token o il certificato della piattaforma corrispondente (ad esempio FCMAPN) viene considerato non valido durante l'invio dei messaggi.

Sicurezza in AWS Push di messaggistica per l'utente finale

Sicurezza nel cloud presso AWS è la massima priorità. Come un AWS cliente, trae vantaggio da data center e architetture di rete progettati per soddisfare i requisiti delle organizzazioni più sensibili alla sicurezza.

La sicurezza è una responsabilità condivisa tra AWS e tu. Il modello di responsabilità condivisa descrive questo aspetto come sicurezza del cloud e sicurezza nel cloud:

- Sicurezza del cloud — AWS è responsabile della protezione dell'infrastruttura in esecuzione AWS servizi in Cloud AWS. AWS fornisce inoltre servizi che è possibile utilizzare in modo sicuro. I revisori esterni testano e verificano regolarmente l'efficacia della nostra sicurezza nell'ambito del AWS Programmi di conformità . Per conoscere i programmi di conformità applicabili a AWS Messaggistica push per l'utente finale, vedere AWS Servizi rientranti nell'ambito del programma di conformità .
- Sicurezza nel cloud: la tua responsabilità è determinata dal AWS servizio che utilizzi. Sei anche responsabile di altri fattori, tra cui la riservatezza dei dati, i requisiti della tua azienda e le leggi e normative vigenti.

Questa documentazione aiuta a capire come applicare il modello di responsabilità condivisa durante l'utilizzo AWS Messaggistica push per l'utente finale. I seguenti argomenti mostrano come configurare AWS End User Messaging Push per raggiungere i tuoi obiettivi di sicurezza e conformità. Imparerai anche a usarne altri AWS servizi che ti aiutano a monitorare e proteggere i tuoi AWS Risorse push per la messaggistica con l'utente finale.

Argomenti

Protezione dei dati in AWS Messaggistica Push per l'utente finale

Il AWS modello di responsabilità condivisa modello di di si applica alla protezione dei dati in AWS Messaggistica push per l'utente finale. Come descritto in questo modello, AWS è responsabile della protezione dell'infrastruttura globale che gestisce tutte le Cloud AWS. L'utente è responsabile del mantenimento del controllo sui contenuti ospitati su questa infrastruttura. L'utente è inoltre responsabile delle attività di configurazione e gestione della sicurezza per Servizi AWS che usi. Per ulteriori informazioni sulla privacy dei dati, consulta la sezione Privacy dei dati FAQ. Per informazioni sulla protezione dei dati in Europa, consulta la AWS Modello di responsabilità condivisa e post sul GDPR blog sul AWS Blog sulla sicurezza.

Ai fini della protezione dei dati, ti consigliamo di proteggere Account AWS credenziali e configura singoli utenti con AWS IAM Identity Center oppure AWS Identity and Access Management (IAM). In tal modo, a ogni utente verranno assegnate solo le autorizzazioni necessarie per svolgere i suoi compiti. Ti suggeriamo, inoltre, di proteggere i dati nei seguenti modi:

- Usa l'autenticazione a più fattori (MFA) con ogni account.
- UsaSSL/TLSper comunicare con AWS risorse. Richiediamo TLS 1.2 e consigliamo TLS 1.3.
- Configurazione API e registrazione delle attività degli utenti con AWS CloudTrail. Per informazioni sull'utilizzo dei CloudTrail percorsi di acquisizione AWS attività, vedi Lavorare con i CloudTrail sentieri in AWS CloudTrail Guida per l'utente.
- Utilizzo AWS soluzioni di crittografia, insieme a tutti i controlli di sicurezza predefiniti all'interno Servizi AWS.
- Utilizza i servizi di sicurezza gestiti avanzati, come Amazon Macie, che aiutano a individuare e proteggere i dati sensibili archiviati in Amazon S3.
- Se sono necessari FIPS 140-3 moduli crittografici convalidati per l'accesso AWS tramite un'interfaccia a riga di comando o unAPI, utilizza un endpoint. FIPS Per ulteriori informazioni sugli FIPS endpoint disponibili, vedere Federal Information Processing Standard (FIPS) 140-3.

Ti consigliamo vivamente di non inserire mai informazioni riservate o sensibili, ad esempio gli indirizzi e-mail dei clienti, nei tag o nei campi di testo in formato libero, ad esempio nel campo Nome. Ciò include quando si lavora con AWS Messaggistica per l'utente finale, Push o altro Servizi AWS utilizzando la consoleAPI, AWS CLI, oppure AWS SDKs. I dati inseriti nei tag o nei campi di testo in formato libero utilizzati per i nomi possono essere utilizzati per i la fatturazione o i log di diagnostica. Se fornisci un URL a un server esterno, ti consigliamo vivamente di non includere le informazioni sulle credenziali URL per convalidare la tua richiesta a quel server.

Crittografia dei dati

AWS I dati push di messaggistica con l'utente finale sono crittografati in transito e a riposo. Quando invii dati a AWS End User Messaging Push, crittografa i dati non appena li riceve e li archivia. Quando recuperi dati da AWS End User Messaging Push, ti trasmette i dati utilizzando gli attuali protocolli di sicurezza.

Crittografia a riposo

AWS End User Messaging Push crittografa tutti i dati che archivia per te. Ciò include i dati di configurazione, i dati degli utenti e degli endpoint, i dati di analisi e tutti i dati aggiunti o importati AWS Messaggistica push per l'utente finale. Per crittografare i tuoi dati, AWS End User Messaging Push utilizza sistemi interni AWS Key Management Service (AWS KMS) chiavi che il servizio possiede e gestisce per conto dell'utente. Queste chiavi vengono ruotate su base regolare. Per informazioni su AWS KMS, vedi il AWS Key Management Service Guida per gli sviluppatori.

Crittografia in transito

AWS End User Messaging Push utilizza HTTPS Transport Layer Security (TLS) 1.2 o versione successiva per comunicare con client e applicazioni. Per comunicare con altri AWS servizi, AWS Usi di End User Messaging Push HTTPS e TLS 1.2. Inoltre, quando crei e gestisci AWS Messaggistica con l'utente finale Risorse push utilizzando la console, un AWS SDK, oppure AWS Command Line Interface, tutte le comunicazioni sono protette utilizzando HTTPS e TLS 1.2.

Gestione delle chiavi

Per crittografare i tuoi AWS Dati push di messaggistica per l'utente finale, AWS End User Messaging Push utilizza sistemi interni AWS KMS chiavi che il servizio possiede e gestisce per tuo conto. Queste chiavi vengono ruotate su base regolare. Non puoi fornire e utilizzare le tue AWS KMS o altre chiavi per crittografare i dati archiviati AWS Messaggistica push per l'utente finale.

Riservatezza del traffico Internet

La privacy del traffico internetwork si riferisce alla protezione delle connessioni e del traffico tra AWS End User Messaging Push e i client e le applicazioni locali e tra AWS End User Messaging Push e altro AWS risorse nella stessa AWS Regione. Le seguenti funzionalità e pratiche possono aiutarti a garantire la privacy del traffico su Internet per AWS Messaggistica push per l'utente finale.

Traffico tra AWS End User Messaging Push e client e applicazioni locali

Per stabilire una connessione privata tra AWS È possibile utilizzare End User Messaging Push e client e applicazioni sulla rete locale AWS Direct Connect. Ciò consente di collegare la rete a un AWS Direct Connect localizzazione mediante un cavo Ethernet standard in fibra ottica. Un'estremità del cavo è collegata al router. L'altra estremità è collegata a un AWS Direct Connect router. Per ulteriori informazioni, consulta Cos'è AWS Direct Connect? nel AWS Direct Connect Guida per l'utente.

Per aiutare a proteggere l'accesso a AWS Messaggistica per l'utente finale Pubblicata tramite push throughAPIs, ti consigliamo di attenerti a AWS Requisiti push di messaggistica con l'utente finale per API le chiamate. AWS End User Messaging Push richiede ai client di utilizzare Transport Layer Security (TLS) 1.2 o versione successiva. I client devono inoltre supportare suite di crittografia con Perfect Forward Secrecy (PFS), come Ephemeral Diffie-Hellman () o Elliptic Curve Diffie-Hellman Ephemeral (). DHE ECDHE La maggior parte dei sistemi moderni, come Java 7 e versioni successive, supporta tali modalità.

Inoltre, le richieste devono essere firmate utilizzando un ID chiave di accesso e una chiave di accesso segreta associata a AWS Identity and Access Management (IAM) principale per il tuo AWS conto. In alternativa, puoi usare il AWS Security Token Service (AWS STS) per generare credenziali di sicurezza temporanee per firmare le richieste.

Traffico tra AWS Messaggistica per l'utente finale, Push e altro AWS risorse

Per proteggere le comunicazioni tra AWS End User Messaging Push e altro AWS risorse nella stessa AWS Regione, AWS Per impostazione predefinita, utilizza End User Messaging Push HTTPS e TLS 1.2.

Gestione delle identità e degli accessi per AWS Messaggistica push per l'utente finale

AWS Identity and Access Management (IAM) è un Servizio AWS che aiuta un amministratore a controllare in modo sicuro l'accesso a AWS risorse. IAMgli amministratori controllano chi può essere autenticato (effettuato l'accesso) e autorizzato (dispone delle autorizzazioni) all'uso AWS Risorse push per la messaggistica con l'utente finale. IAMè un Servizio AWS che puoi utilizzare senza costi aggiuntivi.

Argomenti

Destinatari

Come si usa AWS Identity and Access Management (IAM) differisce, a seconda del lavoro che svolgi AWS Messaggistica push per l'utente finale.

Utente del servizio: se si utilizza il AWS Servizio End User Messaging Push per svolgere il tuo lavoro, l'amministratore ti fornisce le credenziali e le autorizzazioni di cui hai bisogno. Man mano che ne usi di più AWS Funzionalità push di messaggistica con l'utente finale Per svolgere il tuo lavoro, potresti aver bisogno di autorizzazioni aggiuntive. La comprensione della gestione dell'accesso ti consente di richiedere le autorizzazioni corrette all'amministratore. Se non riesci ad accedere a una funzionalità in AWS End User Messaging Push, vediRisoluzione dei problemi AWS Messaggistica con l'utente finale: identità e accesso push.

Amministratore del servizio: se sei responsabile di AWS Messaggistica con l'utente finale Le risorse push della tua azienda, probabilmente hai pieno accesso a AWS Messaggistica push per l'utente finale. Spetta a te determinare quale AWS End User Messaging: funzionalità e risorse push a cui gli utenti del servizio devono accedere. È quindi necessario inviare richieste all'IAMamministratore per modificare le autorizzazioni degli utenti del servizio. Consulta le informazioni contenute in questa pagina per comprendere i concetti di base diIAM. Per saperne di più su come la tua azienda può utilizzare IAM con AWS End User Messaging Push, vediIn che modo AWS End User Messaging Push funziona con IAM.

IAMamministratore: se sei un IAM amministratore, potresti voler saperne di più su come scrivere politiche per gestire l'accesso a AWS Messaggistica push per l'utente finale. Per visualizzare un esempio AWS Messaggistica con l'utente finale Policy push basate sull'identità utilizzabili inIAM, vedi. Esempi di politiche basate sull'identità per AWS Messaggistica push per l'utente finale

Autenticazione con identità

L'autenticazione è il modo in cui si accede a AWS utilizzando le tue credenziali di identità. Devi essere autenticato (aver effettuato l'accesso a AWS) come Utente root dell'account AWS, come IAM utente o assumendo un IAM ruolo.

Puoi accedere a AWS come identità federata utilizzando le credenziali fornite tramite una fonte di identità. AWS IAM Identity Center Gli utenti (IAMIdentity Center), l'autenticazione Single Sign-On della tua azienda e le tue credenziali Google o Facebook sono esempi di identità federate. Quando accedi come identità federata, l'amministratore aveva precedentemente configurato la federazione delle identità utilizzando i ruoli. IAM Quando accedi AWS utilizzando la federazione, assumi indirettamente un ruolo.

A seconda del tipo di utente che sei, puoi accedere a AWS Management Console o il AWS portale di accesso. Per ulteriori informazioni sull'accesso a AWS, vedi Come accedere al tuo Account AWS nella Accedi ad AWS Guida per l'utente.

Se accedi AWS programmaticamente, AWS fornisce un kit di sviluppo software (SDK) e un'interfaccia a riga di comando (CLI) per firmare crittograficamente le richieste utilizzando le credenziali dell'utente. Se non usi AWS strumenti, devi firmare tu stesso le richieste. Per ulteriori informazioni sull'utilizzo del metodo consigliato per firmare autonomamente le richieste, vedi Firma AWS APIrichieste nella Guida IAM per l'utente.

A prescindere dal metodo di autenticazione utilizzato, potrebbe essere necessario specificare ulteriori informazioni sulla sicurezza. Ad esempio, AWS consiglia di utilizzare l'autenticazione a più fattori (MFA) per aumentare la sicurezza del proprio account. Per ulteriori informazioni, consulta Autenticazione a più fattori nel AWS IAM Identity Center Guida per l'utente e utilizzo dell'autenticazione a più fattori () MFA in AWS nella Guida per l'utente di IAM.

Account AWS utente root

Quando crei un Account AWS, inizi con un'unica identità di accesso con accesso completo a tutti Servizi AWS e le risorse presenti nell'account. Questa identità è denominata Account AWS utente root ed è accessibile effettuando l'accesso con l'indirizzo e-mail e la password utilizzati per creare l'account. Si consiglia vivamente di non utilizzare l'utente root per le attività quotidiane. Conserva le credenziali dell'utente root e utilizzale per eseguire le operazioni che solo l'utente root può eseguire. Per l'elenco completo delle attività che richiedono l'accesso come utente root, consulta Attività che richiedono le credenziali dell'utente root nella Guida per l'IAMutente.

Identità federata

Come procedura ottimale, richiedi agli utenti umani, compresi gli utenti che richiedono l'accesso come amministratore, di utilizzare la federazione con un provider di identità per accedere Servizi AWS utilizzando credenziali temporanee.

Un'identità federata è un utente dell'elenco utenti aziendale, un provider di identità Web, il AWS Directory Service, la directory Identity Center o qualsiasi utente che accede Servizi AWS utilizzando le credenziali fornite tramite una fonte di identità. Quando le identità federate accedono Account AWS, assumono ruoli e i ruoli forniscono credenziali temporanee.

Per la gestione centralizzata degli accessi, si consiglia di utilizzare AWS IAM Identity Center. È possibile creare utenti e gruppi in IAM Identity Center oppure connettersi e sincronizzarsi con un set di utenti e gruppi nella propria fonte di identità per utilizzarli su tutti i Account AWS e applicazioni. Per informazioni su IAM Identity Center, vedi Cos'è IAM Identity Center? nel AWS IAM Identity Center Guida per l'utente.

IAM users and groups

Un IAMutente è un'identità all'interno del tuo Account AWS che dispone di autorizzazioni specifiche per una singola persona o applicazione. Laddove possibile, consigliamo di fare affidamento su credenziali temporanee anziché creare IAM utenti con credenziali a lungo termine come password e chiavi di accesso. Tuttavia, se hai casi d'uso specifici che richiedono credenziali a lungo termine con IAM gli utenti, ti consigliamo di ruotare le chiavi di accesso. Per ulteriori informazioni, consulta Ruotare regolarmente le chiavi di accesso per i casi d'uso che richiedono credenziali a lungo termine nella Guida per l'utente. IAM

Un IAMgruppo è un'identità che specifica un insieme di utenti. IAM Non è possibile eseguire l'accesso come gruppo. È possibile utilizzare gruppi per specificare le autorizzazioni per più utenti alla volta. I gruppi semplificano la gestione delle autorizzazioni per set di utenti di grandi dimensioni. Ad esempio, è possibile assegnare un nome a un gruppo IAMAdminse concedere a tale gruppo le autorizzazioni per IAM amministrare le risorse.

Gli utenti sono diversi dai ruoli. Un utente è associato in modo univoco a una persona o un'applicazione, mentre un ruolo è destinato a essere assunto da chiunque ne abbia bisogno. Gli utenti dispongono di credenziali a lungo termine permanenti, mentre i ruoli forniscono credenziali temporanee. Per ulteriori informazioni, consulta Quando creare un IAM utente (anziché un ruolo) nella Guida per l'IAMutente.

IAMruoli

Un IAMruolo è un'identità all'interno del tuo Account AWS che dispone di autorizzazioni specifiche. È simile a un IAM utente, ma non è associato a una persona specifica. È possibile assumere temporaneamente un IAM ruolo nel AWS Management Console cambiando ruolo. Puoi assumere un ruolo chiamando un AWS CLI oppure AWS APIoperazione o utilizzando un comando personalizzatoURL. Per ulteriori informazioni sui metodi di utilizzo dei ruoli, vedere Utilizzo IAM dei ruoli nella Guida per l'IAMutente.

IAMi ruoli con credenziali temporanee sono utili nelle seguenti situazioni:

- Accesso utente federato: per assegnare le autorizzazioni a una identità federata, è possibile creare un ruolo e definire le autorizzazioni per il ruolo. Quando un'identità federata viene autenticata, l'identità viene associata al ruolo e ottiene le autorizzazioni da esso definite. Per informazioni sui ruoli per la federazione, vedere Creazione di un ruolo per un provider di identità di terze parti nella Guida per l'IAMutente. Se utilizzi IAM Identity Center, configuri un set di autorizzazioni. Per controllare a cosa possono accedere le identità dopo l'autenticazione, IAM Identity Center correla il set di autorizzazioni a un ruolo in. IAM Per informazioni sui set di autorizzazioni, consulta Set di autorizzazioni nella AWS IAM Identity Center Guida per l'utente.
- Autorizzazioni IAM utente temporanee: un IAM utente o un ruolo può assumere un IAM ruolo per assumere temporaneamente autorizzazioni diverse per un'attività specifica.
- Accesso su più account: puoi utilizzare un IAM ruolo per consentire a qualcuno (un responsabile fidato) di un altro account di accedere alle risorse del tuo account. I ruoli sono lo strumento principale per concedere l'accesso multi-account. Tuttavia, con alcuni Servizi AWS, è possibile allegare una policy direttamente a una risorsa (anziché utilizzare un ruolo come proxy). Per conoscere la differenza tra i ruoli e le politiche basate sulle risorse per l'accesso tra account diversi, consulta la sezione Accesso alle risorse su più account IAM nella Guida per l'utente. IAM
- Accesso a più servizi: alcuni Servizi AWS usa le funzionalità in altri Servizi AWS. Ad esempio, quando effettui una chiamata in un servizio, è normale che quel servizio esegua applicazioni in Amazon EC2 o archivi oggetti in Amazon S3. Un servizio può eseguire questa operazione utilizzando le autorizzazioni dell'entità chiamante, utilizzando un ruolo di servizio o utilizzando un ruolo collegato al servizio.
  - Sessioni di accesso diretto (FAS): quando utilizzi un IAM utente o un ruolo per eseguire azioni in AWS, sei considerato un preside. Quando si utilizzano alcuni servizi, è possibile eseguire un'operazione che attiva un'altra operazione in un servizio diverso. FASutilizza le autorizzazioni del principale che chiama un Servizio AWS, in combinazione con la richiesta Servizio AWS per effettuare richieste ai servizi a valle. FASle richieste vengono effettuate solo quando un servizio

riceve una richiesta che richiede interazioni con altri Servizi AWS o risorse da completare. In questo caso è necessario disporre delle autorizzazioni per eseguire entrambe le azioni. Per i dettagli FAS delle politiche relative alle richieste, consulta Forward access sessions.

- Ruolo di servizio: un ruolo di servizio è un IAMruolo che un servizio assume per eseguire azioni per conto dell'utente. Un IAM amministratore può creare, modificare ed eliminare un ruolo di servizio dall'internoIAM. Per ulteriori informazioni, vedere Creazione di un ruolo per delegare le autorizzazioni a un Servizio AWS nella Guida per l'utente di IAM.
- Ruolo collegato al servizio: un ruolo collegato al servizio è un tipo di ruolo di servizio collegato a un Servizio AWS. Il servizio può assumere il ruolo di eseguire un'azione per conto dell'utente. I ruoli collegati ai servizi vengono visualizzati nel tuo Account AWS e sono di proprietà del servizio. Un IAM amministratore può visualizzare, ma non modificare le autorizzazioni per i ruoli collegati al servizio.

- Applicazioni in esecuzione su Amazon EC2: puoi utilizzare un IAM ruolo per gestire le credenziali temporanee per le applicazioni in esecuzione su un'EC2istanza e in fase di creazione AWS CLI oppure AWS APIrichieste. Ciò è preferibile alla memorizzazione delle chiavi di accesso all'interno dell'EC2istanza. Per assegnare un AWS assegnare un ruolo a un'EC2istanza e renderlo disponibile a tutte le relative applicazioni, è necessario creare un profilo di istanza collegato all'istanza. Un profilo di istanza contiene il ruolo e consente ai programmi in esecuzione sull'EC2istanza di ottenere credenziali temporanee. Per ulteriori informazioni, consulta Usare un IAM ruolo per concedere le autorizzazioni alle applicazioni in esecuzione su EC2 istanze Amazon nella Guida per l'IAMutente.

Per sapere se utilizzare IAM ruoli o IAM utenti, consulta Quando creare un IAM ruolo (anziché un utente) nella Guida per l'IAMutente.

Gestione dell'accesso con policy

Puoi controllare l'accesso in AWS creando politiche e allegandole a AWS identità o risorse. Una politica è un oggetto in AWS che, se associato a un'identità o a una risorsa, ne definisce le autorizzazioni. AWS valuta queste politiche quando un principale (utente, utente root o sessione di ruolo) effettua una richiesta. Le autorizzazioni nelle policy determinano l'approvazione o il rifiuto della richiesta. La maggior parte delle politiche viene archiviata in AWS come JSON documenti. Per ulteriori informazioni sulla struttura e il contenuto dei documenti relativi alle JSON politiche, vedere Panoramica delle JSON politiche nella Guida per l'IAMutente.

Gli amministratori possono utilizzare AWS JSONpolitiche per specificare chi ha accesso a cosa. In altre parole, quale principale può eseguire azioni su quali risorse e in quali condizioni.

Per impostazione predefinita, utenti e ruoli non dispongono di autorizzazioni. Per concedere agli utenti il permesso di eseguire azioni sulle risorse di cui hanno bisogno, un IAM amministratore può creare IAM politiche. L'amministratore può quindi aggiungere le IAM politiche ai ruoli e gli utenti possono assumerli.

IAMle politiche definiscono le autorizzazioni per un'azione indipendentemente dal metodo utilizzato per eseguire l'operazione. Ad esempio, supponiamo di disporre di una policy che consente l'operazione `iam:GetRole`. Un utente con tale criterio può ottenere informazioni sul ruolo da AWS Management Console, il AWS CLI, o il AWS API.

Policy basate su identità

I criteri basati sull'identità sono documenti relativi ai criteri di JSON autorizzazione che è possibile allegare a un'identità, ad esempio un IAM utente, un gruppo di utenti o un ruolo. Tali policy definiscono le azioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una politica basata sull'identità, consulta Creazione di politiche nella Guida per l'utente. IAM IAM

Le policy basate su identità possono essere ulteriormente classificate come policy inline o policy gestite. Le policy inline sono integrate direttamente in un singolo utente, gruppo o ruolo. Le politiche gestite sono politiche autonome che puoi allegare a più utenti, gruppi e ruoli all'interno del Account AWS. Le politiche gestite includono AWS politiche gestite e politiche gestite dai clienti. Per informazioni su come scegliere tra una politica gestita o una politica in linea, consulta Scelta tra politiche gestite e politiche in linea nella Guida per l'IAMutente.

Policy basate su risorse

Le politiche basate sulle risorse sono documenti di JSON policy allegati a una risorsa. Esempi di politiche basate sulle risorse sono le policy di trust dei IAM ruoli e le policy dei bucket di Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. Quando è collegata a una risorsa, una policy definisce le azioni che un principale può eseguire su tale risorsa e a quali condizioni. È necessario specificare un principale in una policy basata sulle risorse. I principali possono includere account, utenti, ruoli, utenti federati o Servizi AWS.

Le policy basate sulle risorse sono policy inline che si trovano in tale servizio. Non puoi usare AWS politiche gestite da IAM una politica basata sulle risorse.

Liste di controllo degli accessi () ACLs

Le liste di controllo degli accessi (ACLs) controllano quali principali (membri dell'account, utenti o ruoli) dispongono delle autorizzazioni per accedere a una risorsa. ACLssono simili alle politiche basate sulle risorse, sebbene non utilizzino il formato del documento di policy. JSON

Amazon S3, AWS WAF e Amazon VPC sono esempi di servizi che supportanoACLs. Per ulteriori informazioniACLs, consulta la panoramica di Access control list (ACL) nella Amazon Simple Storage Service Developer Guide.

Altri tipi di policy

AWS supporta tipi di policy aggiuntivi e meno comuni. Questi tipi di policy possono impostare il numero massimo di autorizzazioni concesse dai tipi di policy più comuni.

- Limiti delle autorizzazioni: un limite di autorizzazioni è una funzionalità avanzata in cui si impostano le autorizzazioni massime che una politica basata sull'identità può concedere a un'entità (utente o ruolo). IAM IAM È possibile impostare un limite delle autorizzazioni per un'entità. Le autorizzazioni risultanti sono l'intersezione delle policy basate su identità dell'entità e i relativi limiti delle autorizzazioni. Le policy basate su risorse che specificano l'utente o il ruolo nel campo `Principal`sono condizionate dal limite delle autorizzazioni. Un rifiuto esplicito in una qualsiasi di queste policy sostituisce l'autorizzazione. Per ulteriori informazioni sui limiti delle autorizzazioni, consulta Limiti delle autorizzazioni per le entità nella Guida per l'utente. IAM IAM
- Politiche di controllo del servizio (SCPs): SCPs sono JSON politiche che specificano le autorizzazioni massime per un'organizzazione o un'unità organizzativa (OU) in AWS Organizations. AWS Organizations è un servizio per il raggruppamento e la gestione centralizzata di più Account AWS di cui è proprietaria la tua azienda. Se abiliti tutte le funzionalità di un'organizzazione, puoi applicare le politiche di controllo del servizio (SCPs) a uno o tutti i tuoi account. I SCP limiti e le autorizzazioni per le entità presenti negli account dei membri, inclusi tutti Utente root dell'account AWS. Per ulteriori informazioni su Organizations andSCPs, vedere Service control policies nel AWS Organizations Guida per l'utente.
- Policy di sessione: le policy di sessione sono policy avanzate che vengono trasmesse come parametro quando si crea in modo programmatico una sessione temporanea per un ruolo o un utente federato. Le autorizzazioni della sessione risultante sono l'intersezione delle policy basate su identità del ruolo o dell'utente e le policy di sessione. Le autorizzazioni possono anche provenire da una policy basata su risorse. Un rifiuto esplicito in una qualsiasi di queste policy sostituisce l'autorizzazione. Per ulteriori informazioni, consulta le politiche di sessione nella Guida IAM per l'utente.

Più tipi di policy

Quando più tipi di policy si applicano a una richiesta, le autorizzazioni risultanti sono più complicate da comprendere. Per scoprire come AWS determina se consentire una richiesta quando sono coinvolti più tipi di policy, consulta Logica di valutazione delle policy nella Guida per l'IAMutente.

In che modo AWS End User Messaging Push funziona con IAM

Prima di utilizzare IAM per gestire l'accesso a AWS End User Messaging Push, scopri quali IAM funzionalità sono disponibili per l'uso AWS Messaggistica push per l'utente finale.

IAMfunzionalità utilizzabili con AWS Messaggistica Push per l'utente finale

IAMfunzionalità	AWS Supporto Push per la messaggistica con l'utente finale
Policy basate su identità	Sì
Policy basate su risorse	Sì
Azioni di policy	Sì
Risorse relative alle policy	Sì
Chiavi di condizione delle policy	Sì
ACLs	No
ABAC(tag nelle politiche)	Parziale
Credenziali temporanee	Sì
Autorizzazioni del principale	Sì
Ruoli di servizio	Sì
Ruoli collegati al servizio	No

Per avere una visione di alto livello di come AWS Messaggistica per l'utente finale, Push e altro AWS i servizi funzionano con la maggior parte delle IAM funzionalità, vedi AWS servizi compatibili con IAM la Guida per l'IAMutente.

Politiche basate sull'identità per AWS Messaggistica push per l'utente finale

Supporta le policy basate su identità: sì

Le politiche basate sull'identità sono documenti relativi alle politiche di JSON autorizzazione che è possibile allegare a un'identità, ad esempio un IAM utente, un gruppo di utenti o un ruolo. Tali policy definiscono le azioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una politica basata sull'identità, consulta Creazione di politiche nella Guida per l'utente. IAM IAM

Con le politiche IAM basate sull'identità, puoi specificare azioni e risorse consentite o negate, nonché le condizioni in base alle quali le azioni sono consentite o negate. Non è possibile specificare l'entità principale in una policy basata sull'identità perché si applica all'utente o al ruolo a cui è associato. Per ulteriori informazioni su tutti gli elementi che è possibile utilizzare in una JSON politica, vedere il riferimento agli elementi IAM JSON della politica nella Guida per l'IAMutente.

Esempi di policy basate sull'identità per AWS Messaggistica push per l'utente finale

Per visualizzare esempi di AWS Policy basate sull'identità di End User Messaging Push, vedi. Esempi di politiche basate sull'identità per AWS Messaggistica push per l'utente finale

Politiche basate sulle risorse all'interno AWS Messaggistica Push per l'utente finale

Supporta politiche basate sulle risorse: Sì

Le politiche basate sulle risorse sono documenti di JSON policy allegati a una risorsa. Esempi di politiche basate sulle risorse sono le policy di trust dei IAM ruoli e le policy dei bucket di Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. Quando è collegata a una risorsa, una policy definisce le azioni che un principale può eseguire su tale risorsa e a quali condizioni. È necessario specificare un principale in una policy basata sulle risorse. I principali possono includere account, utenti, ruoli, utenti federati o Servizi AWS.

Per abilitare l'accesso tra più account, puoi specificare un intero account o IAM entità in un altro account come principale in una politica basata sulle risorse. L'aggiunta di un principale multi-account

a una policy basata sulle risorse rappresenta solo una parte della relazione di trust. Quando il principale e la risorsa sono diversi Account AWS, un IAM amministratore dell'account affidabile deve inoltre concedere all'entità principale (utente o ruolo) l'autorizzazione ad accedere alla risorsa. L'autorizzazione viene concessa collegando all'entità una policy basata sull'identità. Tuttavia, se una policy basata su risorse concede l'accesso a un principale nello stesso account, non sono richieste ulteriori policy basate su identità. Per ulteriori informazioni, consulta la sezione Cross Account Resource Access IAM nella Guida IAM per l'utente.

Azioni politiche per AWS Messaggistica push per l'utente finale

Supporta le operazioni di policy: si

Gli amministratori possono utilizzare AWS JSONpolitiche per specificare chi ha accesso a cosa. Cioè, quale principale può eseguire operazioni su quali risorse, e in quali condizioni.

L'`Action`elemento di una JSON policy descrive le azioni che è possibile utilizzare per consentire o negare l'accesso a una policy. Le azioni politiche in genere hanno lo stesso nome di quelle associate AWS APIoperazione. Esistono alcune eccezioni, come le azioni di sola autorizzazione che non hanno un'operazione corrispondente. API Esistono anche alcune operazioni che richiedono più operazioni in una policy. Queste operazioni aggiuntive sono denominate operazioni dipendenti.

Includi le operazioni in una policy per concedere le autorizzazioni a eseguire l'operazione associata.

Per visualizzare un elenco di AWS Azioni push di messaggistica per l'utente finale, vedere Azioni definite da AWS End User Messaging Push nel riferimento di autorizzazione del servizio.

Azioni politiche in AWS End User Messaging Push utilizza il seguente prefisso prima dell'azione:

```
mobiletargeting
```

Per specificare più operazioni in una sola istruzione, occorre separarle con la virgola.

```
"Action": [
      "mobiletargeting:action1",
      "mobiletargeting:action2"
         ]
```

Per visualizzare esempi di AWS Policy basate sull'identità di End User Messaging Push, vedi. Esempi di politiche basate sull'identità per AWS Messaggistica push per l'utente finale

Risorse politiche per AWS Messaggistica push per l'utente finale

Supporta le risorse di policy: sì

Gli amministratori possono utilizzare AWS JSONpolitiche per specificare chi ha accesso a cosa. Cioè, quale principale può eseguire operazioni su quali risorse, e in quali condizioni.

L'elemento `Resource` JSON policy specifica l'oggetto o gli oggetti a cui si applica l'azione. Le istruzioni devono includere un elemento `Resource`o un elemento `NotResource`. Come best practice, specifica una risorsa utilizzando il relativo Amazon Resource Name (ARN). Puoi eseguire questa operazione per azioni che supportano un tipo di risorsa specifico, note come autorizzazioni a livello di risorsa.

Per le azioni che non supportano le autorizzazioni a livello di risorsa, ad esempio le operazioni di elenco, utilizza un carattere jolly (*) per indicare che l'istruzione si applica a tutte le risorse.

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

Per visualizzare un elenco di AWS Tipi di risorse End User Messaging Push e relativeARNs, vedi Risorse definite da AWS End User Messaging Push nel riferimento di autorizzazione del servizio. Per sapere con quali azioni è possibile specificare le caratteristiche ARN di ciascuna risorsa, consulta Azioni definite da AWS Messaggistica push per l'utente finale.

Per visualizzare esempi di AWS Policy basate sull'identità di End User Messaging Push, vedi. Esempi di politiche basate sull'identità per AWS Messaggistica push per l'utente finale

Chiavi relative alle condizioni delle politiche per AWS Messaggistica Push per l'utente finale

Supporta le chiavi di condizione delle policy specifiche del servizio: sì

Gli amministratori possono utilizzare AWS JSONpolitiche per specificare chi ha accesso a cosa. Cioè, quale principale può eseguire azioni su quali risorse, e in quali condizioni.

L'elemento `Condition`(o blocco `Condition`) consente di specificare le condizioni in cui un'istruzione è in vigore. L'elemento `Condition`è facoltativo. Puoi compilare espressioni condizionali

che utilizzano operatori di condizione, ad esempio uguale a o minore di, per soddisfare la condizione nella policy con i valori nella richiesta.

Se specificate più `Condition` elementi in un'istruzione o più chiavi in un singolo `Condition` elemento, AWS li valuta utilizzando un'`AND`operazione logica. Se specificate più valori per una singola chiave di condizione, AWS valuta la condizione utilizzando un'`0R`operazione logica. Tutte le condizioni devono essere soddisfatte prima che le autorizzazioni dell'istruzione vengano concesse.

Puoi anche utilizzare variabili segnaposto quando specifichi le condizioni. Ad esempio, è possibile concedere a un IAM utente l'autorizzazione ad accedere a una risorsa solo se è contrassegnata con il relativo nome IAM utente. Per ulteriori informazioni, consulta gli elementi IAM della politica: variabili e tag nella Guida IAM per l'utente.

AWS supporta chiavi di condizione globali e chiavi di condizione specifiche del servizio. Per vedere tutto AWS chiavi di condizione globali, vedi AWS chiavi di contesto della condizione globale nella Guida IAM per l'utente.

Per visualizzare un elenco di AWS Chiavi di condizione Push per la messaggistica con l'utente finale, vedi Condition Keys per AWS End User Messaging Push nel riferimento di autorizzazione del servizio. Per sapere con quali azioni e risorse è possibile utilizzare una chiave di condizione, consulta Azioni definite da AWS Messaggistica push per l'utente finale.

Per visualizzare esempi di AWS Policy basate sull'identità di End User Messaging Push, vedi. Esempi di politiche basate sull'identità per AWS Messaggistica push per l'utente finale

ACLsin AWS Messaggistica Push per l'utente finale

SupportiACLs: no

Le liste di controllo degli accessi (ACLs) controllano quali principali (membri dell'account, utenti o ruoli) dispongono delle autorizzazioni per accedere a una risorsa. ACLssono simili alle politiche basate sulle risorse, sebbene non utilizzino il formato del documento di policy. JSON

ABACcon AWS Messaggistica Push per l'utente finale

Supporti ABAC (tag nelle politiche): Parziale

Il controllo degli accessi basato sugli attributi (ABAC) è una strategia di autorizzazione che definisce le autorizzazioni in base agli attributi. In AWS, questi attributi sono chiamati tag. È possibile allegare tag a IAM entità (utenti o ruoli) e a molte AWS risorse. L'etichettatura di entità e risorse è il primo

passo diABAC. Quindi si progettano ABAC politiche per consentire le operazioni quando il tag del principale corrisponde al tag sulla risorsa a cui sta tentando di accedere.

ABACè utile in ambienti in rapida crescita e aiuta in situazioni in cui la gestione delle politiche diventa complicata.

Per controllare l'accesso basato su tag, fornisci informazioni sui tag nell'elemento condizione di una policy utilizzando le chiavi di condizione `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`o `aws:TagKeys`.

Se un servizio supporta tutte e tre le chiavi di condizione per ogni tipo di risorsa, il valore per il servizio è Yes (Sì). Se un servizio supporta tutte e tre le chiavi di condizione solo per alcuni tipi di risorsa, allora il valore sarà Parziale.

Per ulteriori informazioni suABAC, vedere Cos'è? ABAC nella Guida IAM per l'utente. Per visualizzare un tutorial con i passaggi per la configurazioneABAC, consulta Utilizzare il controllo di accesso basato sugli attributi (ABAC) nella Guida per l'IAMutente.

Utilizzo di credenziali temporanee con AWS Messaggistica Push per l'utente finale

Supporta le credenziali temporanee: sì

Medio Servizi AWS non funzionano quando accedi utilizzando credenziali temporanee. Per ulteriori informazioni, tra cui Servizi AWS lavorare con credenziali temporanee, vedere Servizi AWS che funzionano con IAM la Guida per l'IAMutente.

Stai utilizzando credenziali temporanee se accedi a AWS Management Console utilizzando qualsiasi metodo tranne il nome utente e la password. Ad esempio, quando accedi AWS utilizzando il link Single Sign-On (SSO) della vostra azienda, tale processo crea automaticamente credenziali temporanee. Le credenziali temporanee vengono create in automatico anche quando accedi alla console come utente e poi cambi ruolo. Per ulteriori informazioni sul cambio di ruolo, consulta Passare a un ruolo (console) nella Guida per l'IAMutente.

È possibile creare manualmente credenziali temporanee utilizzando il AWS CLI oppure AWS API. È quindi possibile utilizzare tali credenziali temporanee per accedere AWS. AWS consiglia di generare dinamicamente credenziali temporanee anziché utilizzare chiavi di accesso a lungo termine. Per ulteriori informazioni, vedere Credenziali di sicurezza temporanee in. IAM

Autorizzazioni principali per più servizi per AWS Messaggistica push per l'utente finale

Supporta sessioni di accesso diretto (FAS): Sì

Quando si utilizza un IAM utente o un ruolo per eseguire azioni in AWS, sei considerato un preside. Quando si utilizzano alcuni servizi, è possibile eseguire un'operazione che attiva un'altra operazione in un servizio diverso. FASutilizza le autorizzazioni del principale che chiama un Servizio AWS, in combinazione con la richiesta Servizio AWS per effettuare richieste ai servizi a valle. FASle richieste vengono effettuate solo quando un servizio riceve una richiesta che richiede interazioni con altri Servizi AWS o risorse da completare. In questo caso è necessario disporre delle autorizzazioni per eseguire entrambe le azioni. Per i dettagli FAS delle politiche relative alle richieste, consulta Forward access sessions.

Ruoli di servizio per AWS Messaggistica push per l'utente finale

Supporta i ruoli di servizio: sì

Un ruolo di servizio è un IAMruolo che un servizio assume per eseguire azioni per conto dell'utente. Un IAM amministratore può creare, modificare ed eliminare un ruolo di servizio dall'internoIAM. Per ulteriori informazioni, vedere Creazione di un ruolo per delegare le autorizzazioni a un Servizio AWS nella Guida per l'utente di IAM.

> **Warning**
>
> La modifica delle autorizzazioni per un ruolo di servizio potrebbe interrompersi AWS Funzionalità push di messaggistica per l'utente finale. Modifica i ruoli di servizio solo quando AWS End User Messaging Push fornisce indicazioni in tal senso.

Ruoli collegati ai servizi per AWS Messaggistica push per l'utente finale

Supporta i ruoli collegati ai servizi: no

Un ruolo collegato al servizio è un tipo di ruolo di servizio collegato a un Servizio AWS. Il servizio può assumere il ruolo di eseguire un'azione per conto dell'utente. I ruoli collegati ai servizi vengono visualizzati nel tuo Account AWS e sono di proprietà del servizio. Un IAM amministratore può visualizzare, ma non modificare le autorizzazioni per i ruoli collegati al servizio.

Per informazioni dettagliate sulla creazione o la gestione di ruoli collegati ai servizi, vedere AWS servizi che funzionano con. IAM Trova un servizio nella tabella che include un `Yes` nella colonna Service-linked role (Ruolo collegato ai servizi). Scegli il collegamento Sì per visualizzare la documentazione relativa al ruolo collegato ai servizi per tale servizio.

Esempi di politiche basate sull'identità per AWS Messaggistica push per l'utente finale

Per impostazione predefinita, gli utenti e i ruoli non sono autorizzati a creare o modificare AWS Risorse push di messaggistica per l'utente finale. Inoltre, non possono eseguire attività utilizzando il AWS Management Console, AWS Command Line Interface (AWS CLI), oppure AWS API. Per concedere agli utenti l'autorizzazione a eseguire azioni sulle risorse di cui hanno bisogno, un IAM amministratore può creare IAM politiche. L'amministratore può quindi aggiungere le IAM politiche ai ruoli e gli utenti possono assumerli.

Per informazioni su come creare una politica IAM basata sull'identità utilizzando questi documenti di esempioJSON, consulta Creazione di IAM politiche nella Guida per l'IAMutente.

Per informazioni dettagliate sulle azioni e sui tipi di risorse definiti da AWS End User Messaging Push, incluso il formato di ARNs per ogni tipo di risorsa, consulta Azioni, risorse e chiavi di condizione per AWS End User Messaging Push nel riferimento di autorizzazione del servizio.

Argomenti

- Best practice per le policy
- Utilizzo di AWS Console push di messaggistica per l'utente finale
- Consentire agli utenti di visualizzare le loro autorizzazioni

Best practice per le policy

Le politiche basate sull'identità determinano se qualcuno può creare, accedere o eliminare AWS Messaggistica con l'utente finale Risorse push nel tuo account. Queste azioni possono comportare costi per Account AWS. Quando crei o modifichi politiche basate sull'identità, segui queste linee guida e consigli:

- Inizia con AWS politiche gestite e passaggio alle autorizzazioni con privilegi minimi: per iniziare a concedere autorizzazioni a utenti e carichi di lavoro, utilizza il AWS politiche gestite che concedono autorizzazioni per molti casi d'uso comuni. Sono disponibili nel tuo Account AWS. Si consiglia di ridurre ulteriormente le autorizzazioni definendo AWS politiche gestite dai clienti specifiche per i tuoi casi d'uso. Per ulteriori informazioni, consulta AWS politiche gestite o AWS politiche gestite per le funzioni lavorative nella Guida per IAM l'utente.
- Applica le autorizzazioni con privilegi minimi: quando imposti le autorizzazioni con le IAM politiche, concedi solo le autorizzazioni necessarie per eseguire un'attività. Puoi farlo definendo

le azioni che possono essere intraprese su risorse specifiche in condizioni specifiche, note anche come autorizzazioni con privilegi minimi. Per ulteriori informazioni sull'utilizzo per applicare le autorizzazioni, consulta Politiche e autorizzazioni nella Guida IAM per l'utente. IAM IAM

- Utilizza le condizioni nelle IAM politiche per limitare ulteriormente l'accesso: puoi aggiungere una condizione alle tue politiche per limitare l'accesso ad azioni e risorse. Ad esempio, puoi scrivere una condizione di policy per specificare che tutte le richieste devono essere inviate utilizzandoSSL. È inoltre possibile utilizzare le condizioni per concedere l'accesso alle azioni di servizio se vengono utilizzate tramite uno specifico Servizio AWS, ad esempio AWS CloudFormation. Per ulteriori informazioni, consulta Elementi IAM JSON della politica: Condizione nella Guida IAM per l'utente.
- Usa IAM Access Analyzer per convalidare IAM le tue policy e garantire autorizzazioni sicure e funzionali: IAM Access Analyzer convalida le policy nuove ed esistenti in modo che aderiscano al linguaggio delle IAM policy () e alle best practice. JSON IAM IAMAccess Analyzer fornisce più di 100 controlli delle politiche e consigli pratici per aiutarti a creare policy sicure e funzionali. Per ulteriori informazioni, vedere Convalida delle policy di IAM Access Analyzer nella Guida per l'utente. IAM
- Richiedi l'autenticazione a più fattori (MFA): se disponi di uno scenario che richiede l'utilizzo di IAM utenti o di un utente root Account AWS, attivala MFA per una maggiore sicurezza. Per richiedere MFA quando vengono richiamate API le operazioni, aggiungi MFA delle condizioni alle tue politiche. Per ulteriori informazioni, consulta Configurazione dell'APIaccesso MFA protetto nella Guida per l'IAMutente.

Per ulteriori informazioni sulle procedure consigliate inIAM, consulta la sezione Procedure consigliate in materia di sicurezza IAM nella Guida per l'IAMutente.

Utilizzo di AWS Console push di messaggistica per l'utente finale

Per accedere a AWS La console End User Messaging Push, è necessario disporre di un set minimo di autorizzazioni. Queste autorizzazioni devono consentire di elencare e visualizzare i dettagli relativi a AWS Messaggistica con l'utente finale Risorse push nel tuo Account AWS. Se crei una politica basata sull'identità che è più restrittiva delle autorizzazioni minime richieste, la console non funzionerà come previsto per le entità (utenti o ruoli) con quella politica.

Non è necessario consentire autorizzazioni minime per la console per gli utenti che effettuano chiamate solo verso AWS CLI o il AWS API. Consenti invece l'accesso solo alle azioni che corrispondono all'APIoperazione che stanno cercando di eseguire.

Per garantire che utenti e ruoli possano ancora utilizzare il AWS Console End User Messaging Push, allega anche il `AWSEndUserMessaging` AWS politica gestita per le entità. Per ulteriori informazioni, consulta Aggiungere autorizzazioni a un utente nella Guida per l'IAMutente.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "AWSEndUserMessaging",
   "Effect": "Allow",
   "Action": [
    "mobiletargeting:CreateApp",
                "mobiletargeting:GetApp",
                "mobiletargeting:GetApps",
                            "mobiletargeting:DeleteApp",
                            "mobiletargeting:GetChannels",
                            "mobiletargeting:GetApnsChannel",
                            "mobiletargeting:GetApnsVoipChannel",
                            "mobiletargeting:GetApnsVoipSandboxChannel",
                            "mobiletargeting:GetApnsSandboxChannel",
                "mobiletargeting:GetAdmChannel",
                "mobiletargeting:GetBaiduChannel",
                "mobiletargeting:GetGcmChannel",
                "mobiletargeting:UpdateApnsChannel",
                "mobiletargeting:UpdateApnsVoipChannel",
                "mobiletargeting:UpdateApnsVoipSandboxChannel",
                "mobiletargeting:UpdateBaiduChannel",
                "mobiletargeting:UpdateGcmChannel",
                "mobiletargeting:UpdateAdmChannel"
            ],
   "Resource": [
    "*"
   ]
  }
 ]
}
```

Consentire agli utenti di visualizzare le loro autorizzazioni

Questo esempio mostra come è possibile creare una politica che consenta IAM agli utenti di visualizzare le politiche in linea e gestite allegate alla loro identità utente. Questa politica include le

autorizzazioni per completare questa azione sulla console o utilizzando a livello di codice il AWS CLI oppure AWS API.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Risoluzione dei problemi AWS Messaggistica con l'utente finale: identità e accesso push

Utilizza le seguenti informazioni per aiutarti a diagnosticare e risolvere i problemi più comuni che potresti riscontrare quando lavori con AWS Messaggistica per l'utente finale Push eIAM.

Argomenti

Non sono autorizzato a eseguire alcuna azione in AWS Messaggistica Push per l'utente finale

Se ricevi un errore che indica che non disponi dell'autorizzazione per eseguire un'operazione, le tue policy devono essere aggiornate in modo che ti sei consentito eseguire tale operazione.

L'errore di esempio seguente si verifica quando l'`mateojackson`IAMutente tenta di utilizzare la console per visualizzare i dettagli su una *`my-example-widget`* risorsa fittizia ma non dispone delle autorizzazioni fittizie`mobiletargeting:`*`GetWidget`*.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 mobiletargeting:GetWidget on resource: my-example-widget
```

In questo caso, la policy per l'utente `mateojackson` deve essere aggiornata per consentire l'accesso alla risorsa *`my-example-widget`* utilizzando l'azione `mobiletargeting:`*`GetWidget`*.

Se hai bisogno di aiuto, contatta il AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

Non sono autorizzato a eseguire iam: PassRole

Se ricevi un messaggio di errore indicante che non sei autorizzato a eseguire l'`iam:PassRole`azione, le tue politiche devono essere aggiornate per consentirti di assegnare un ruolo a AWS Messaggistica push per l'utente finale.

Medio Servizi AWS consentono di trasferire un ruolo esistente a quel servizio anziché creare un nuovo ruolo di servizio o un ruolo collegato al servizio. Per eseguire questa operazione, è necessario disporre delle autorizzazioni per trasmettere il ruolo al servizio.

L'errore di esempio seguente si verifica quando un IAM utente denominato `marymajor` tenta di utilizzare la console per eseguire un'azione in AWS Messaggistica push per l'utente finale. Tuttavia, l'azione richiede che il servizio disponga delle autorizzazioni concesse da un ruolo di servizio. Mary non dispone delle autorizzazioni per passare il ruolo al servizio.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

In questo caso, le policy di Mary devono essere aggiornate per poter eseguire l'operazione `iam:PassRole`.

Se hai bisogno di aiuto, contatta il AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

Voglio consentire l'accesso a persone esterne al mio Account AWS per accedere al mio AWS Risorse push per la messaggistica con l'utente finale

È possibile creare un ruolo con il quale utenti in altri account o persone esterne all'organizzazione possono accedere alle tue risorse. È possibile specificare chi è attendibile per l'assunzione del ruolo. Per i servizi che supportano politiche basate sulle risorse o liste di controllo degli accessi (ACLs), puoi utilizzare tali politiche per concedere agli utenti l'accesso alle tue risorse.

Per ulteriori informazioni, consulta gli argomenti seguenti:

- Per sapere se AWS End User Messaging Push supporta queste funzionalità, vedereIn che modo AWS End User Messaging Push funziona con IAM.
- Per scoprire come fornire l'accesso alle tue risorse in tutto il mondo Account AWS di cui sei proprietario, vedi Fornire l'accesso a un IAM utente in un altro Account AWS che possiedi nella Guida per l'IAMutente.
- Per scoprire come fornire l'accesso alle tue risorse a terze parti Account AWS, vedi Fornire l'accesso a Account AWS di proprietà di terzi nella Guida per l'IAMutente.
- Per informazioni su come fornire l'accesso tramite la federazione delle identità, consulta Fornire l'accesso a utenti autenticati esternamente (federazione delle identità) nella Guida per l'IAMutente.

- Per conoscere la differenza tra l'utilizzo di ruoli e politiche basate sulle risorse per l'accesso tra account diversi, consulta la sezione Accesso alle risorse tra account nella Guida per l'utente. IAM IAM

Convalida della conformità per AWS Messaggistica push per l'utente finale

Per sapere se un Servizio AWS rientra nell'ambito di specifici programmi di conformità, vedere Servizi AWS in Scope by Compliance Program e scegli il programma di conformità che ti interessa. Per informazioni generali, vedi AWS Programmi di conformità di conformità.

È possibile scaricare report di audit di terze parti utilizzando AWS Artifact. Per ulteriori informazioni, consulta Scaricamento dei report in AWS Artifact.

La tua responsabilità di conformità durante l'utilizzo Servizi AWS è determinata dalla sensibilità dei dati, dagli obiettivi di conformità dell'azienda e dalle leggi e dai regolamenti applicabili. AWS fornisce le seguenti risorse per contribuire alla conformità:

- Guide introduttive su sicurezza e conformità: queste guide all'implementazione illustrano le considerazioni relative all'architettura e forniscono i passaggi per l'implementazione degli ambienti di base su AWS incentrati sulla sicurezza e la conformità.
- Architettura per la HIPAA sicurezza e la conformità su Amazon Web Services: questo white paper descrive come le aziende possono utilizzare AWS per creare applicazioni idonee. HIPAA

> **Note**
>
> Non tutti Servizi AWS sono HIPAA idonei. Per ulteriori informazioni, consulta la Guida ai servizi HIPAA idonei.

- AWS Risorse per la conformità : questa raccolta di cartelle di lavoro e guide potrebbe riguardare il tuo settore e la tua località.
- AWS Guide alla conformità dei clienti: comprendi il modello di responsabilità condivisa attraverso la lente della conformità. Le guide riassumono le migliori pratiche per la protezione Servizi AWS e mappano le linee guida per i controlli di sicurezza su più framework (tra cui il National Institute of Standards and Technology (NIST), il Payment Card Industry Security Standards Council (PCI) e l'International Organization for Standardization ()). ISO

- Valutazione delle risorse con regole in AWS Config Guida per gli sviluppatori: la AWS Config il servizio valuta la conformità delle configurazioni delle risorse alle pratiche interne, alle linee guida del settore e alle normative.
- AWS Security Hub— Questo Servizio AWS fornisce una visione completa dello stato di sicurezza all'interno AWS. Security Hub utilizza i controlli di sicurezza per valutare i AWS risorse e per verificare la vostra conformità agli standard e alle migliori pratiche del settore della sicurezza. Per un elenco dei servizi e dei controlli supportati, consulta la pagina Documentazione di riferimento sui controlli della Centrale di sicurezza.
- Amazon GuardDuty — Questo Servizio AWS rileva potenziali minacce per il tuo Account AWS, carichi di lavoro, contenitori e dati monitorando l'ambiente alla ricerca di attività sospette e dannose. GuardDuty può aiutarti a soddisfare vari requisiti di conformità, ad esempio PCI DSS soddisfacendo i requisiti di rilevamento delle intrusioni imposti da determinati framework di conformità.
- AWS Audit Manager— Questo Servizio AWS ti aiuta a controllare continuamente i tuoi AWS utilizzo per semplificare la gestione del rischio e la conformità alle normative e agli standard di settore.

Resilienza in AWS Messaggistica push per l'utente finale

Il AWS l'infrastruttura globale è costruita attorno a Regioni AWS e zone di disponibilità. Regioni AWS forniscono più zone di disponibilità fisicamente separate e isolate, collegate con reti a bassa latenza, ad alto throughput e altamente ridondanti. Con le zone di disponibilità, puoi progettare e gestire applicazioni e database che eseguono automaticamente il failover tra zone di disponibilità senza interruzioni. Le zone di disponibilità sono più disponibili, tolleranti ai guasti e scalabili rispetto alle infrastrutture a data center singolo o multiplo tradizionali.

Per ulteriori informazioni sull' Regioni AWS e zone di disponibilità, vedi AWS Infrastruttura globale.

Oltre al AWS infrastruttura globale, AWS End User Messaging Push offre diverse funzionalità per supportare le esigenze di resilienza e backup dei dati.

Sicurezza dell'infrastruttura in AWS Messaggistica Push per l'utente finale

Come servizio gestito, AWS End User Messaging Push è protetto da AWS procedure di sicurezza di rete globali descritte nel white paper Amazon Web Services: Overview of Security Processes.

Usi AWS APIchiamate pubblicate per accedere AWS Messaggistica per l'utente finale Push attraverso la rete. I client devono supportare Transport Layer Security (TLS) 1.2 o versione successiva. I client devono inoltre supportare suite di crittografia con Perfect Forward Secrecy (PFS) come (Ephemeral Diffie-Hellman) o DHE (Elliptic Curve Ephemeral Diffie-Hellman). ECDHE La maggior parte dei sistemi moderni, come Java 7 e versioni successive, supporta tali modalità.

Inoltre, le richieste devono essere firmate utilizzando un ID chiave di accesso e una chiave di accesso segreta associata a un principale. IAM Oppure puoi usare il AWS Security Token Service (AWS STS) per generare credenziali di sicurezza temporanee per firmare le richieste.

Analisi della configurazione e delle vulnerabilità

Come servizio gestito, AWS End User Messaging Push è protetto da AWS procedure di sicurezza di rete globali descritte nel white paper Amazon Web Services: panoramica dei processi di sicurezza. Ciò significa che AWS gestisce ed esegue attività e procedure di sicurezza di base per rafforzare, applicare patch, aggiornare e mantenere in altro modo l'infrastruttura sottostante per l'account e le risorse. Queste procedure sono state riviste e certificate dalle terze parti appropriate.

Best practice di sicurezza

Utilizzo AWS Account Identity and Access Management (IAM) per controllare l'accesso alle API operazioni, in particolare alle operazioni che creano, modificano o eliminano risorse. Per ilAPI, tali risorse includono progetti, campagne e viaggi.

- Crea un utente IAM per ogni persona che gestisce le risorse , incluso l'utente stesso. Non usare AWS credenziali di root per gestire le risorse.
- Assegna a ciascun utente un set minimo di autorizzazioni richieste per eseguire le proprie mansioni.
- Usa IAM i gruppi per gestire efficacemente le autorizzazioni per più utenti.
- Ruota periodicamente le credenziali IAM.

Per ulteriori informazioni sulla sicurezza, consulta Sicurezza in AWS Push di messaggistica per l'utente finale. Per ulteriori informazioni suIAM, vedere AWS Identity and Access Management. Per informazioni sulle IAM best practice, consulta la sezione IAMbest practice.

Monitoraggio dei messaggi push per gli utenti AWS finali

Il monitoraggio è una parte importante per mantenere l'affidabilità, la disponibilità e le prestazioni di AWS End User Messaging Push e delle altre AWS soluzioni. AWSfornisce i seguenti strumenti di monitoraggio per guardare AWS End User Messaging Push, segnalare quando qualcosa non va e intraprendere azioni automatiche se necessario:

- Amazon CloudWatch monitora AWS le tue risorse e le applicazioni su cui esegui AWS in tempo reale. Puoi raccogliere i parametri e tenerne traccia, creare pannelli di controllo personalizzati e impostare allarmi per inviare una notifica o intraprendere azioni quando un parametro specificato raggiunge una determinata soglia. Ad esempio, puoi tenere CloudWatch traccia CPU dell'utilizzo o di altri parametri delle tue EC2 istanze Amazon e avviare automaticamente nuove istanze quando necessario. Per ulteriori informazioni, consulta la Amazon CloudWatch User Guide.
- Amazon CloudWatch Logs ti consente di monitorare, archiviare e accedere ai tuoi file di registro da EC2 istanze Amazon e altre fonti. CloudTrail CloudWatch I log possono monitorare le informazioni contenute nei file di registro e avvisarti quando vengono raggiunte determinate soglie. Puoi inoltre archiviare i dati del log in storage estremamente durevole. Per ulteriori informazioni, consulta la Amazon CloudWatch Logs User Guide.
- Amazon EventBridge può essere utilizzato per automatizzare AWS i tuoi servizi e rispondere automaticamente agli eventi di sistema, come problemi di disponibilità delle applicazioni o modifiche delle risorse. Gli eventi AWS relativi ai servizi vengono forniti quasi EventBridge in tempo reale. Puoi compilare regole semplici che indichino quali eventi sono considerati di interesse per te e quali operazioni automatizzate intraprendere quando un evento corrisponde a una regola. Per ulteriori informazioni, consulta Amazon EventBridge User Guide.
- AWS CloudTrailacquisisce le API chiamate e gli eventi correlati effettuati da o per conto del tuo AWS account e invia i file di registro a un bucket Amazon S3 da te specificato. Puoi identificare quali utenti e account hanno chiamato AWS, l'indirizzo IP di origine da cui sono state effettuate le chiamate e quando sono avvenute le chiamate. Per ulteriori informazioni, consulta la Guida per l'utente AWS CloudTrail.

Monitoraggio dei messaggi push per gli utenti AWS finali con Amazon CloudWatch

È possibile monitorare AWS End User Messaging utilizzando End User Messaging Push CloudWatch, che raccoglie dati grezzi e li elabora in metriche leggibili e quasi in tempo reale.

Queste statistiche vengono conservate per un periodo di 15 mesi, per permettere l'accesso alle informazioni storiche e offrire una prospettiva migliore sulle prestazioni del servizio o dell'applicazione web. È anche possibile impostare allarmi che controllano determinate soglie e inviare notifiche o intraprendere azioni quando queste soglie vengono raggiunte. Per ulteriori informazioni, consulta la Amazon CloudWatch User Guide.

Per un elenco di metriche e dimensioni, consulta Monitoring Amazon Pinpoint CloudWatch with nella Amazon Pinpoint User Guide.

Registrazione delle API chiamate push tramite messaggistica per l'utente AWS finale AWS CloudTrail

AWS End User Messaging Push è integrato con AWS CloudTrail, un servizio che fornisce una registrazione delle azioni intraprese da un utente, ruolo o AWS servizio in AWS End User Messaging Push. CloudTrail acquisisce tutte le API chiamate per AWS End User Messaging Push come eventi. Le chiamate acquisite includono le chiamate dalla console AWS End User Messaging Push e le chiamate in codice alle API operazioni AWS End User Messaging Push. Se crei un trail, puoi abilitare la distribuzione continua di CloudTrail eventi a un bucket Amazon S3, inclusi gli eventi per AWS End User Messaging Push. Se non configuri un percorso, puoi comunque visualizzare gli eventi più recenti nella CloudTrail console nella cronologia degli eventi. Utilizzando le informazioni raccolte da CloudTrail, è possibile determinare la richiesta effettuata a AWS End User Messaging Push, l'indirizzo IP da cui è stata effettuata la richiesta, chi ha effettuato la richiesta, quando è stata effettuata e ulteriori dettagli.

Per ulteriori informazioni CloudTrail, consulta la Guida AWS CloudTrail per l'utente.

AWS Messaggistica con l'utente finale Informazioni push in CloudTrail

CloudTrail è abilitato sul tuo Account AWS quando crei l'account. Quando si verifica un'attività in AWS End User Messaging Push, tale attività viene registrata in un CloudTrail evento insieme ad altri eventi di AWS servizio nella cronologia degli eventi. Puoi visualizzare, cercare e scaricare eventi recenti in Account AWS. Per ulteriori informazioni, consulta Visualizzazione degli eventi con la cronologia degli CloudTrail eventi.

Per una registrazione continua degli eventi in tuo Account AWS, inclusi gli eventi per AWS End User Messaging Push, crea un percorso. Un trail consente di CloudTrail inviare file di log a un bucket Amazon S3. Per impostazione predefinita, quando si crea un percorso nella console, questo sarà valido in tutte le Regioni AWS. Il trail registra gli eventi di tutte le regioni della AWS partizione e

consegna i file di log al bucket Amazon S3 specificato. Inoltre, puoi configurare altri AWS servizi per analizzare ulteriormente e agire in base ai dati sugli eventi raccolti nei log. CloudTrail Per ulteriori informazioni, consulta gli argomenti seguenti:

- Panoramica della creazione di un percorso
- CloudTrail servizi e integrazioni supportati
- Configurazione delle SNS notifiche Amazon per CloudTrail
- Ricezione di file di CloudTrail registro da più regioni e ricezione di file di CloudTrail registro da più account

Tutte le azioni push di messaggistica con l'utente AWS finale vengono registrate CloudTrail e sono documentate nel riferimento sui push API di messaggistica per l'utente AWS finale. Ad esempio, le chiamate a `UpdateApnsChannel` e `GetAdmChannel` le `GetApnsVoipChannel` azioni generano voci nei file di CloudTrail registro.

Ogni evento o voce di log contiene informazioni sull'utente che ha generato la richiesta. Le informazioni di identità consentono di determinare quanto segue:

- Se la richiesta è stata effettuata con credenziali utente root o AWS Identity and Access Management (IAM).
- Se la richiesta è stata effettuata con le credenziali di sicurezza temporanee per un ruolo o un utente federato.
- Se la richiesta è stata effettuata da un altro AWS servizio.

Per ulteriori informazioni, vedete l'CloudTrail userIdentityelemento.

Informazioni sulle voci dei file di registro push di AWS End User Messaging

Un trail è una configurazione che consente la distribuzione di eventi come file di log in un bucket Amazon S3 specificato dall'utente. CloudTrail i file di registro contengono una o più voci di registro. Un evento rappresenta una singola richiesta proveniente da qualsiasi fonte e include informazioni sull'azione richiesta, la data e l'ora dell'azione, i parametri della richiesta e così via. CloudTrail i file di registro non sono una traccia stack ordinata delle API chiamate pubbliche, quindi non vengono visualizzati in un ordine specifico.

Accedere a AWS End User Messaging Push utilizzando un'interfaccia endpoint ()AWS PrivateLink

Puoi utilizzarlo AWS PrivateLink per creare una connessione privata tra te VPC e AWS End User Messaging Push. Puoi accedere a AWS End User Messaging Push come se fosse nel tuoVPC, senza l'uso di un gateway, NAT dispositivo, VPN connessione o AWS Direct Connect connessione Internet. Le istanze in uso VPC non necessitano di indirizzi IP pubblici per accedere a AWS End User Messaging Push.

Stabilisci questa connessione privata creando un endpoint di interfaccia attivato da AWS PrivateLink. In ciascuna sottorete viene creata un'interfaccia di rete endpoint da abilitare per l'endpoint di interfaccia. Si tratta di interfacce di rete gestite dai richiedenti che fungono da punto di ingresso per il traffico destinato all'End User Messaging Push. AWS

Per ulteriori informazioni, consulta Access Servizi AWS through AWS PrivateLink nella Guida.AWS PrivateLink

Considerazioni sulla messaggistica push per l'utente AWS finale

Prima di configurare un endpoint di interfaccia per AWS End User Messaging Push, consulta le considerazioni nella Guida.AWS PrivateLink

AWS End User Messaging Push supporta l'esecuzione di chiamate a tutte le sue API azioni tramite l'endpoint dell'interfaccia.

VPCle policy degli endpoint non sono supportate per AWS End User Messaging Push. Per impostazione predefinita, l'accesso completo a AWS End User Messaging Push è consentito tramite l'interfaccia endpoint. In alternativa, è possibile associare un gruppo di sicurezza alle interfacce di rete dell'endpoint per controllare il traffico verso l'utente AWS finale di messaggistica Push attraverso l'endpoint dell'interfaccia.

Crea un endpoint di interfaccia per AWS End User Messaging Push

Puoi creare un endpoint di interfaccia per AWS End User Messaging Push utilizzando la VPC console Amazon o AWS Command Line Interface (AWS CLI). Per ulteriori informazioni, consulta la sezione Creazione di un endpoint di interfaccia nella Guida per l'utente di AWS PrivateLink .

Crea un endpoint di interfaccia per AWS End User Messaging Push utilizzando il seguente nome di servizio:

```
com.amazonaws.region.pinpoint
```

Se abiliti private DNS per l'endpoint dell'interfaccia, puoi effettuare API richieste a AWS End User Messaging Push utilizzando il nome regionale DNS predefinito. Ad esempio, `com.amazonaws.us-east-1.pinpoint`.

Creazione di una policy dell' endpoint per l'endpoint dell'interfaccia

Una policy per gli endpoint è una IAM risorsa che è possibile allegare a un endpoint di interfaccia. La policy predefinita per gli endpoint consente l'accesso completo all' AWS End User Messaging Push tramite l'endpoint dell'interfaccia. Per controllare l'accesso consentito a AWS End User Messaging Push dal tuoVPC, allega una policy personalizzata per l'endpoint all'endpoint di interfaccia.

Una policy di endpoint specifica le informazioni riportate di seguito:

- I principali che possono eseguire azioni (IAMutenti Account AWS e IAM ruoli).
- Le azioni che possono essere eseguite.
- Le risorse in cui è possibile eseguire le operazioni.

Per ulteriori informazioni, consulta la sezione Controllo dell'accesso ai servizi con policy di endpoint nella Guida di AWS PrivateLink .

Esempio: policy VPC sugli endpoint per le azioni push di messaggistica con l'utente AWS finale

Di seguito è riportato l'esempio di una policy dell'endpoint personalizzata. Quando alleghi questa policy all'endpoint dell'interfaccia, concede l'accesso alle azioni push di messaggistica per l'utente AWS finale elencate per tutti i principali utenti su tutte le risorse.

```
{
   "Statement": [
      {
         "Principal": "*",
         "Effect": "Allow",
         "Action": [
             "mobiletargeting:CreateApp",
             "mobiletargeting:DeleteApp"
```

```
         ],
         "Resource":"*"
      }
   ]
}
```

Quote per la messaggistica AWS push per l'utente finale

Your Account AWS ha delle quote predefinite, precedentemente denominate limiti, per ogni servizio. AWS Salvo diversa indicazione, ogni quota si applica a una regione specifica. Se per alcune quote è possibile richiedere aumenti, altre quote non possono essere modificate.

Per visualizzare le quote per AWS End User Messaging Push, apri la console Service Quotas. Nel riquadro di navigazione, scegli AWSservizi e seleziona Amazon Pinpoint.

Il tuo AWS account ha le seguenti quote relative a AWS End User Messaging Push.

Risorsa	Quota predefinita	Idoneità all'incremento
Numero massimo di notifiche push che possono essere inviate al secondo in una campagna	25.000 notifiche al secondo	Sì, usa la console Service Quotas
Dimensioni del payload dei messaggi Amazon Device Messaging (ADM)	6 KB per messaggio	No
Dimensioni del payload dei messaggi del servizio Apple Push Notification (APNs)	4 KB per messaggio	No
Dimensione del payload dei messaggi sandbox di APNs	4 KB per messaggio	No
Dimensione del payload dei messaggi di Baidu Cloud Push	4 KB per messaggio	No
Dimensione del payload dei messaggi di Firebase Cloud Messaging (FCM)	4 KB per messaggio	No

Cronologia dei documenti per la AWS End User Messaging Push User Guide

La tabella seguente descrive le versioni della documentazione per AWS End User Messaging Push.

Modifica	Descrizione	Data
Versione iniziale	Versione iniziale della Guida per l'utente di AWS End User Messaging Push	24 luglio 2024

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.