사용자 가이드

AWS 최종 사용자 메시지 푸시

AWS 최종 사용자 메시지 푸시: 사용자 가이드

Table of Contents

AWS 최종 사용자 메시지 푸시란?

> **Note**
>
> Amazon Pinpoint의 푸시 알림 기능을 이제 AWS 최종 사용자 메시징이라고 합니다.

AWS 최종 사용자 메시지 푸시를 사용하면 푸시 알림 채널을 통해 푸시 알림을 전송하여 앱 사용자의 참여를 유도할 수 있습니다. Apple 푸시 알림 서비스 (APNs), Firebase 클라우드 메시징 (FCM), Amazon 디바이스 메시징 (ADM), Baidu Push를 지원합니다.

주제

AWS 최종 사용자 메시지 푸시를 처음 사용하시나요?

AWS 최종 사용자 메시지 푸시를 처음 사용하는 경우 먼저 다음 섹션을 읽는 것이 좋습니다.

- 설정하기 AWS 계정
- AWS 최종 사용자 메시지 푸시 시작하기
- 애플리케이션 생성 및 푸시 채널 활성화

AWS 최종 사용자 메시지 푸시의 특징

다음 푸시 알림 서비스에 대해 별도의 채널을 사용하여 앱에 푸시 알림을 보낼 수 있습니다.

- Firebase 클라우드 메시징 () FCM
- Apple 푸시 알림 서비스 () APNs

> **Note**
>
> 를 APNs 사용하여 및 와 같은 iOS 장치뿐만 아니라 Mac 랩톱 iPhones 및 iPads 데스크톱과 같은 macOS 장치의 Safari 브라우저에도 메시지를 보낼 수 있습니다.

- Baidu 클라우드 푸시
- 아마존 디바이스 메시징 (ADM)

AWS 최종 사용자 메시지 푸시에 액세스

콘솔 또는 콘솔을 통해 서비스에 액세스할 수 있는 다양한 방법을 간략하게 API 설명하십시오. CLI

다음 인터페이스를 사용하여 AWS 최종 사용자 메시지 푸시를 관리할 수 있습니다.

AWS 최종 사용자 메시지 푸시 콘솔

: AWS 최종 사용자 메시지 푸시 리소스를 만들고 관리하는 웹 인터페이스입니다. 에 가입한 경우 에서 AWS 최종 사용자 메시지 푸시 콘솔에 액세스할 수 AWS Management Console있습니다. AWS 계정

AWS Command Line Interface

: 명령줄 셸의 명령을 사용하여 AWS 서비스와 상호 작용할 수 있습니다. AWS Command Line Interface 는 윈도우, macOS, 리눅스에서 지원됩니다. 에 AWS CLI대한 자세한 내용은 AWS Command Line Interface 사용 설명서를 참조하십시오. 명령 참조에서 AWS 최종 사용자 메시지 푸시 명령을 찾을 수 있습니다.AWS CLI

AWS SDKs

: 요청을 제출하는 APIs 대신 언어별 응용 프로그램을 빌드하는 것을 HTTP 선호하거나 라이브러리, 샘플 코드HTTPS, 자습서 및 기타 리소스를 AWS 제공하는 소프트웨어 개발자인 경우 이러한 라이브러리는 요청에 암호로 서명하고, 요청을 재시도하고, 오류 응답을 처리하는 등 작업을 자동화하는 기본 기능을 제공합니다. 이러한 함수를 사용하면 작업을 더 효율적으로 시작할 수 있습니다. 자세한 내용은 AWS기반의 도구를 참조하세요.

리전별 가용성

AWS 최종 사용자 메시지 푸시는 북미, 유럽, 아시아 및 오세아니아의 여러 AWS 리전 지역에서 사용할 수 있습니다. 각 지역에서 여러 가용 영역을 AWS 유지 관리합니다. 이러한 가용 영역은 물리적으로 서

로 분리되어 있지만, 지연 시간이 짧고 처리량과 중복성이 우수한 프라이빗 네트워크 연결로 통합됩니다. 이러한 가용 영역은 매우 높은 수준의 가용성과 중복성을 제공하는 동시에 지연 시간을 최소화하는 데 사용됩니다.

자세히 AWS 리전알아보려면 에서 사용할 수 있는 AWS 리전 계정 지정을 참조하십시오. Amazon Web Services 일반 참조 현재 AWS 최종 사용자 메시지 푸시를 사용할 수 있는 모든 지역 목록과 각 지역의 엔드포인트는 Amazon Pinpoint 및 서비스 엔드포인트의 엔드포인트 API 및AWS 할당량을 참조하십시오. Amazon Web Services 일반 참조 각 리전에서 사용할 수 있는 가용 영역 수에 대한 자세한 내용은 AWS 글로벌 인프라를 참조하세요.

설정하기 AWS 계정

AWS 최종 사용자 메시지 푸시를 사용하여 앱에 푸시 알림을 보내려면 먼저 충분한 IAM 권한이 AWS 계정 있는 알림을 받아야 합니다. 이는 AWS 생태계의 다른 서비스에도 사용될 AWS 계정 수 있습니다.

주제

가입해 보세요. AWS 계정

계정이 없는 경우 다음 단계를 완료하여 계정을 만드세요. AWS 계정

가입하려면 AWS 계정

1. https://portal.aws.amazon.com/billing/등록 열기.
2. 온라인 지시 사항을 따릅니다.

   등록 절차 중 전화를 받고 전화 키패드로 확인 코드를 입력하는 과정이 있습니다.

   에 가입하면 AWS 계정 루트 사용자팬이 생성됩니다. AWS 계정루트 사용자에게는 계정의 모든 AWS 서비스 및 리소스에 액세스할 권한이 있습니다. 보안 모범 사례는 사용자에게 관리 액세스 권한을 할당하고, 루트 사용자만 사용하여 루트 사용자 액세스 권한이 필요한 작업을 수행하는 것입니다.

AWS 가입 절차가 완료된 후 확인 이메일을 보냅니다. 언제든지 https://aws.amazon.com/으로 이동하고 내 계정을 선택하여 현재 계정 활동을 보고 계정을 관리할 수 있습니다.

관리자 액세스 권한이 있는 사용자 생성

등록한 AWS 계정후에는 일상적인 작업에 루트 사용자를 사용하지 않도록 관리 사용자를 보호하고 AWS IAM Identity Center활성화하고 생성하십시오 AWS 계정 루트 사용자.

보안을 유지하세요. AWS 계정 루트 사용자

1. 루트 사용자를 선택하고 AWS 계정 이메일 주소를 입력하여 계정 AWS Management Console소유자로 로그인합니다. 다음 페이지에서 비밀번호를 입력합니다.

루트 사용자를 사용하여 로그인하는 데 도움이 필요하면 AWS 로그인 User Guide의 루트 사용자로 로그인을 참조하십시오.

2. 루트 사용자에 대한 다단계 인증 (MFA) 을 켜십시오.

   지침은 사용 설명서의 AWS 계정 IAM루트 사용자 (콘솔) 용 가상 MFA 기기 활성화를 참조하십시오.

관리자 액세스 권한이 있는 사용자 생성

1. IAMID 센터를 활성화합니다.

   지침은 AWS IAM Identity Center 사용 설명서의 AWS IAM Identity Center설정을 참조하세요.

2. IAMID 센터에서 사용자에게 관리 액세스 권한을 부여하십시오.

   를 ID 소스로 사용하는 방법에 대한 자습서는 사용 설명서의 기본값으로 IAM Identity Center 디렉터리AWS IAM Identity Center 사용자 액세스 구성을 참조하십시오. IAM Identity Center 디렉터리

관리 액세스 권한이 있는 사용자로 로그인

- IAMIdentity Center 사용자로 로그인하려면 IAM Identity Center 사용자를 생성할 때 이메일 주소로 전송된 URL 로그인을 사용하십시오.

  IAMIdentity Center 사용자를 사용하여 로그인하는 데 도움이 필요하면 사용 설명서의 AWS 액세스 포털에 로그인을 참조하십시오.AWS 로그인

추가 사용자에게 액세스 권한 할당

1. IAMIdentity Center에서 최소 권한 권한 적용의 모범 사례를 따르는 권한 집합을 생성하십시오.

   지침은AWS IAM Identity Center 사용 설명서의 Create a permission set를 참조하세요.

2. 사용자를 그룹에 할당하고, 그룹에 Single Sign-On 액세스 권한을 할당합니다.

   지침은AWS IAM Identity Center 사용 설명서의 Add groups를 참조하세요.

AWS 최종 사용자 메시지 푸시 시작하기

앱에 푸시 알림을 보낼 수 있도록 AWS 최종 사용자 메시지 푸시를 설정하려면 먼저 AWS 최종 사용자 메시지 푸시가 앱에 메시지를 보낼 수 있도록 승인하는 자격 증명을 제공해야 합니다. 제공한 자격 증명은 사용하는 푸시 알림 시스템에 따라 다릅니다.

- Apple 푸시 알림 서비스 (APN) 자격 증명에 대해서는 Apple 개발자 설명서에서 Apple로부터 암호화 키 및 키 ID 받기 및 Apple로부터 공급자 인증서 받기를 참조하십시오.
- Firebase 콘솔을 통해 얻을 수 있는 Firebase 클라우드 메시징 (FCM) 자격 증명에 대해서는 Firebase 클라우드 메시징을 참조하십시오.
- Baidu 자격 증명에 대해서는 Baidu를 참조하십시오.
- Amazon 디바이스 메시징 (ADM) 자격 증명에 대해서는 자격 증명 획득을 참조하십시오.

애플리케이션 생성 및 푸시 채널 활성화

AWS 최종 사용자 메시지 푸시를 사용하여 푸시 알림을 보내려면 먼저 애플리케이션을 만들고 푸시 알림 채널을 활성화해야 합니다.

상황에 맞는

애플리케이션

애플리케이션은 모든 AWS 최종 사용자 메시지 푸시 설정을 저장하는 컨테이너입니다. 이 애플리케이션은 또한 Amazon Pinpoint 채널, 캠페인 및 여정 설정을 저장합니다.

Key(키)

AWS 최종 사용자 메시지 푸시가 인증 토큰에 암호로 서명하는 데 사용하는 개인 서명 키입니다. APNs 이 서명 키는 Apple 개발자 계정에서 얻을 수 있습니다.

서명 키를 제공하면AWS 최종 사용자 메시지 푸시는 보내는 모든 푸시 알림에 APNs 대해 토큰을 사용하여 인증합니다. 서명 키를 사용하여 APNs 프로덕션 및 샌드박스 환경에 푸시 알림을 보낼 수 있습니다.

서명 키는 인증서와 달리 만료되지 않습니다. 키는 한 번만 입력하면 되고, 나중에 갱신할 필요가 없습니다. 또한 동일한 서명 키를 여러 앱에 사용할 수 있습니다. 자세한 내용은 Xcode 도움말의 인증 토큰을 APNs 사용한 통신을 참조하십시오.

Certificate

푸시 알림을 보낼 APNs 때 AWS 최종 사용자 메시지 푸시가 인증하는 데 사용하는 TLS 인증서입니다. APNs인증서는 프로덕션 환경과 샌드박스 환경을 모두 지원하거나 샌드박스 환경만 지원할 수 있습니다. 이 인증서는 Apple 개발자 계정에서 얻을 수 있습니다.

인증서는 1년 후 만료됩니다. 이 경우 새 인증서를 만든 다음 AWS 최종 사용자 메시지 푸시에 제공하여 푸시 알림 전송을 갱신해야 합니다. 자세한 내용은 Xcode 도움말의 TLS인증서를 APNs 사용하여 통신하기를 참조하십시오.

사전 조건

푸시 채널을 사용하려면 먼저 푸시 서비스에 대한 유효한 자격 증명이 있어야 합니다. 자격 증명 획득에 대한 자세한 내용은 을 참조하십시오AWS 최종 사용자 메시지 푸시 시작하기.

절차

다음 지침에 따라 애플리케이션을 생성하고 푸시 채널을 활성화하십시오. 이 절차를 완료하려면 애플리케이션 이름만 입력하면 됩니다. 나중에 푸시 채널을 활성화하거나 비활성화할 수 있습니다.

1. 에서 AWS 최종 사용자 메시지 푸시 콘솔을 엽니다 https://console.aws.amazon.com/push-notifications/.
2. 애플리케이션 생성을 선택합니다.
3. 애플리케이션 이름에 애플리케이션 이름을 입력합니다.
4. (선택 사항) 이 선택적 단계에 따라 Apple 푸시 알림 서비스를 활성화하십시오 (APNs).
   a. Apple 푸시 알림 서비스 (APNs) 의 경우 활성화를 선택합니다.
   b. 기본 인증 유형에서는 다음 중 하나를 선택합니다.
      i. 키 자격 증명을 선택하는 경우 Apple 개발자 계정에서 다음 정보를 제공하십시오. AWS 최종 사용자 메시지 푸시는 인증 토큰을 생성하기 위해 이 정보를 필요로 합니다.
         - 키 ID - 서명 키에 할당된 ID입니다.
         - 번들 식별자 - iOS 앱에 할당된 ID입니다.
         - 팀 식별자 – Apple 개발자 계정 팀에 할당된 ID입니다.
         - 인증 키 - 인증 키를 생성할 때 Apple 개발자 계정에서 다운로드하는 .p8 파일입니다.
      ii. 인증서 자격 증명을 선택한 경우 다음 정보를 제공합니다.
         - SSL인증서 — 인증서의.p12 파일입니다. TLS
         - 인증서 암호 - 인증서에 암호를 할당했으면 여기에 입력합니다.
         - 인증서 유형 - 사용할 인증서 유형을 선택합니다.
5. (선택사항) 이 선택적 단계에 따라 Firebase 클라우드 메시징 () 을 사용 설정하세요. FCM
   a. Firebase 클라우드 메시징 (FCM) 의 경우 활성화를 선택합니다.
   b. 기본 인증 유형에서는 다음 중 하나를 선택합니다.
      i. 토큰 자격 증명 (권장) 의 경우 파일 선택을 선택한 다음 서비스 JSON 파일을 선택합니다.
      ii. 키 자격 증명의 경우 키 입력 API키를 입력합니다.
6. (선택 사항) 이 선택적 단계에 따라 Baidu Cloud Push를 활성화하십시오.

a. Baidu 클라우드 푸시의 경우 활성화를 선택합니다.
b. APIキー에 API 키를 입력합니다.
c. 비밀 키에는 비밀 키를 입력합니다.

7. (선택 사항) 이 선택적 단계에 따라 Amazon 디바이스 메시징을 활성화하십시오.

a. Amazon 디바이스 메시징의 경우 활성화를 선택합니다.
b. 클라이언트 ID에 클라이언트 ID를 입력합니다.
c. 클라이언트 비밀번호에는 클라이언트 비밀번호를 입력합니다.

8. 애플리케이션 생성을 선택합니다.

푸시 채널 비활성화

푸시 채널을 비활성화하려면 다음 지침을 따르십시오.

1. 에서 AWS 최종 사용자 메시지 푸시 콘솔을 엽니다 https://console.aws.amazon.com/push-notifications/.
2. 푸시 자격 증명이 포함된 애플리케이션을 선택합니다.
3. (선택 사항) Apple 푸시 알림 서비스의 경우 (APNs) 활성화를 선택 취소합니다.
4. (선택사항) Firebase 클라우드 메시징의 경우 (FCM) 활성화를 선택 취소합니다.
5. (선택사항) Baidu Cloud 푸시의 경우 활성화를 선택 취소합니다.
6. (선택 사항) Amazon 디바이스 메시징의 경우 활성화를 선택 취소합니다.
7. 변경 사항 저장(Save changes)을 선택합니다.

메시지 전송

AWS 최종 사용자 메시지 푸시는 특정 장치 식별자에게 트랜잭션 푸시 알림을 보낼 API 수 있습니다. 이 섹션에는 를 사용하여 AWS 최종 사용자 메시지 푸시를 통해 푸시 알림을 보내는 데 사용할 수 있는 전체 코드 예제가 포함되어 있습니다. API AWS SDK

이 예제를 사용하여 AWS 최종 사용자 메시지 푸시가 지원하는 모든 푸시 알림 서비스를 통해 푸시 알림을 보낼 수 있습니다. 현재 AWS 최종 사용자 메시지 푸시는 Firebase 클라우드 메시징 (FCM), Apple 푸시 알림 서비스 (APNs), Baidu 클라우드 푸시, Amazon 디바이스 메시징 (ADM) 채널을 지원합니다.

엔드포인트, 세그먼트, 채널에 대한 추가 코드 예제는 코드 예제를 참조하십시오.

> **Note**
>
> Firebase 클라우드 메시징 (FCM) 서비스를 통해 푸시 알림을 보내는 경우 AWS 최종 사용자 메시지 푸시를 호출할 때 서비스 이름을 GCM 사용하세요. API 구글은 2018년 4월 10일에 Google 클라우드 메시징 (GCM) 서비스를 중단했습니다. 하지만 AWS 최종 사용자 메시지 푸시는 GCM 서비스를 중단하기 전에 작성된 API 코드와의 호환성을 유지하기 위해 서비스를 통해 보내는 메시지에 FCM 서비스 이름을 API 사용합니다. GCM

GCM (AWS CLI)

다음 예에서는 send-messages를 사용하여 와 함께 GCM 푸시 알림을 보냅니다. AWS CLI Replace *token* 디바이스의 고유 토큰과 함께 *611e3e3cdd47474c9c1399a50example* 애플리케이션 식별자와 함께

```
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"
      }
  }
}
```

다음 예제에서는 send-messages를 사용하여 모든 레거시 키를 사용하여 와 함께 GCM 푸시 알림을 보냅니다. AWS CLI Replace *token* 디바이스의 고유 토큰과 함께 *611e3e3cdd47474c9c1399a50example* 애플리케이션 식별자와 함께

```
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
```

다음 예제에서는 send-messages를 사용하여 를 사용하여 FCMv1 메시지 페이로드와 함께 GCM 푸시 알림을 보냅니다. AWS CLI Replace *token* 디바이스의 고유 토큰을 사용하고 *611e3e3cdd47474c9c1399a50example* 애플리케이션 식별자와 함께

```
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
```

ImageUrlfield for를 사용하는 경우GCM, pinpoint는 해당 필드를 데이터 알림으로 보내는 데pinpoint.notification.imageUrl, 키가 있으면 이미지가 즉시 렌더링되는 것을 방지할 수 있습니다. 앱 통합과 같은 데이터 키 처리를 RawContent 사용하거나 추가하세요. AWS Amplify

Safari (AWS CLI)

AWS 최종 사용자 메시지 푸시를 사용하여 Apple의 Safari 웹 브라우저를 사용하는 macOS 컴퓨터에 메시지를 보낼 수 있습니다. Safari 브라우저로 메시지를 보내려면 원시 메시지 콘텐츠를 지정하고 메시지 페이로드에 특정 속성을 포함해야 합니다. Amazon Pinpoint User Guide에서 원시 메시지 페이로드가 포함된 푸시 알림 템플릿을 생성하거나 캠페인 메시지에 직접 원시 메시지 콘텐츠를 지정하여 이를 수행할 수 있습니다.

> **Note**
>
> 이 특별한 속성은 Safari 웹 브라우저를 사용하는 macOS 랩톱 및 데스크톱 컴퓨터로 전송하는 데 필요합니다. iPhones 및 와 같은 iOS 장치로 전송하는 경우에는 필요하지 않습니다 iPads.

Safari 웹 브라우저로 메시지를 보내려면 원시 메시지 페이로드를 지정해야 합니다. 원시 메시지 페이로드의 aps 객체 내에 url-args 배열이 포함되어야 합니다. Safari 웹 브라우저에 푸시 알림을 보내려면 url-args 배열이 필요합니다. 하지만 배열에 비어 있는 단일 요소가 포함되어도 괜찮습니다.

다음 예제에서는 send-messages를 사용하여 를 사용하여 Safari 웹 브라우저에 알림을 보냅니다. AWS CLI Replace *token* 디바이스의 고유 토큰을 사용하여 *611e3e3cdd47474c9c1399a50example* 애플리케이션 식별자와 함께

```
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
```

Safari 푸시 알림에 대한 자세한 내용은 Apple 개발자 웹 사이트에서 Safari 푸시 알림 구성을 참조하세요.

APNS (AWS CLI)

다음 예제에서는 send-messages를 사용하여 와 함께 APNS 푸시 알림을 보냅니다. AWS CLI Replace *token* 디바이스의 고유 토큰을 사용하여 *611e3e3cdd47474c9c1399a50example* 애플리케이션 식별자와 함께,*GAME_INVITATION* 고유 식별자 사용.

```
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)

이 예제를 사용하면 Node.js JavaScript 내 for를 사용하여 푸시 알림을 보낼 수 있습니다. AWS SDK 이 JavaScript 예제에서는 Node.js 에서 양식을 이미 설치하고 구성했다고 가정합니다. SDK

또한 이 예제에서는 공유 자격 증명 파일을 사용하여 기존 사용자의 액세스 키 및 보안 액세스 키를 지정한다고 가정합니다. 자세한 AWS SDK내용은 JavaScript Node.js 개발자 안내서의 양식에서 자격 증명 설정을 참조하십시오.

```
'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

AWS SDK for Python (Boto3)를 사용하여 푸시 알림을 보내려면 이 예를 사용하세요. 이 예제에서는 Python (Boto3) SDK 용으로 이미 설치 및 구성했다고 가정합니다.

또한 이 예제에서는 공유 자격 증명 파일을 사용하여 기존 사용자의 액세스 키 및 보안 액세스 키를 지정한다고 가정합니다. 자세한 내용은 AWS SDKfor Python (Boto3) API 참조의 자격 증명을 참조하십시오.

```
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()
```

추가 리소스

- 푸시 채널 템플릿에 대한 자세한 내용은 Amazon Pinpoint 사용 설명서의 푸시 알림 템플릿 생성을 참조하십시오.

애플리케이션에서 푸시 알림 받기

다음 항목에서는 푸시 알림을 수신하도록 Swift, Android, React Native 또는 Flutter 앱을 수정하는 방법을 설명합니다.

주제

Swift 푸시 알림 설정

iOS 앱의 푸시 알림은 Apple 푸시 알림 서비스 (APNs) 를 사용하여 전송됩니다. iOS 디바이스에 푸시 알림을 전송하려면 먼저 Apple 개발자 포털에서 앱 ID를 만들고 필요한 인증서를 생성해야 합니다. 이러한 단계를 완료하는 방법에 대한 자세한 내용은 AWS Amplify 설명서의 푸시 알림 서비스 설정을 참조하십시오.

토큰을 사용한 APNs 작업

앱을 다시 설치할 때 고객의 디바이스 토큰이 재생성되도록 앱을 개발하는 것이 가장 좋습니다.

수신자가 디바이스를 iOS의 새로운 메이저 버전으로 업그레이드(예: iOS 12에서 iOS 13으로 업그레이드)한 후 앱을 다시 설치하면 앱은 새 토큰을 생성합니다. 앱이 토큰을 새로 고치지 않으면 알림을 전송하는 데 이전 토큰이 사용됩니다. 결과적으로 Apple 푸시 알림 서비스 (APNs) 는 토큰이 유효하지 않기 때문에 알림을 거부합니다. 알림을 보내려고 하면 에서 APNs 메시지 실패 알림을 받게 됩니다.

Android 푸시 알림 설정

Android 앱의 푸시 알림은 Google 클라우드 메시징 (FCM) 을 대체하는 Firebase 클라우드 메시징 (GCM) 을 사용하여 전송됩니다. Android 기기에 푸시 알림을 보내려면 먼저 사용자 인증 정보를 FCM 얻어야 합니다. 이 자격 증명을 사용하여 Android 프로젝트를 생성하고 푸시 알림을 수신할 수 있는 샘

플 앱을 실행할 수 있습니다. 이러한 단계를 완료하는 방법에 대한 자세한 내용은 AWS Amplify 설명서의 푸시 알림 섹션에서 확인할 수 있습니다.

Flutter 푸시 알림 설정

Flutter 앱의 푸시 알림은 안드로이드와 APNs iOS의 경우 Firebase 클라우드 메시징 (FCM) 을 사용하여 전송됩니다. 이러한 단계의 완료에 대한 자세한 내용은 AWS Amplify Flutter 설명서의 푸시 알림 섹션을 참조하세요.

React Native 푸시 알림 설정

React Native 앱의 푸시 알림은 Android 및 APNs iOS용 Firebase 클라우드 메시징 (FCM) 을 사용하여 전송됩니다. 이러한 단계를 완료하는 방법에 대한 자세한 내용은 AWS Amplify JavaScript 설명서의 푸시 알림 섹션에서 확인할 수 있습니다.

AWS 최종 사용자 메시지 푸시에서 애플리케이션을 생성하십시오.

AWS 최종 사용자 메시지 푸시에서 푸시 알림 전송을 시작하려면 애플리케이션을 만들어야 합니다. 그런 다음, 해당되는 자격 증명을 제공하여 사용할 푸시 알림 채널을 활성화해야 합니다.

AWS 최종 사용자 메시지 푸시 콘솔을 사용하여 새 애플리케이션을 만들고 푸시 알림 채널을 설정할 수 있습니다. 자세한 내용은 애플리케이션 생성 및 푸시 채널 활성화 단원을 참조하십시오.

APIAWS SDK, an 또는 AWS Command Line Interface(AWS CLI) 를 사용하여 애플리케이션을 만들고 설정할 수도 있습니다. 애플리케이션을 만들려면 Apps 리소스를 사용하십시오. 푸시 알림 채널을 구성하려면 아래 리소스를 사용합니다.

- APNsApple 푸시 알림 서비스를 사용하여 iOS 기기 사용자에게 메시지를 보내는 채널입니다.
- ADMAmazon Kindle Fire 디바이스 사용자에게 메시지를 보내는 채널입니다.
- Baidu 채널은 Baidu 사용자에게 메시지를 전송하는 데 사용됩니다.
- GCMGoogle 클라우드 메시징 () 을 대체하는 Firebase 클라우드 메시징 (FCM) 을 사용하여 채널을 통해 Android 기기에 메시지를 보낼 수 있습니다. GCM

푸시 알림 처리

푸시 알림을 보내는 데 필요한 자격 증명을 획득한 후에는 푸시 알림을 받을 수 있도록 애플리케이션을 업데이트할 수 있습니다. 자세한 내용은 설명서의 푸시 알림 - 시작하기를 참조하십시오. AWS Amplify

애플리케이션 삭제

이 절차를 수행하면 계정 및 애플리케이션의 모든 리소스에서 애플리케이션이 제거됩니다.

상황에 맞는

애플리케이션

애플리케이션은 모든 AWS 최종 사용자 메시지 푸시 설정을 저장하는 컨테이너입니다. 이 애플리케이션은 또한 Amazon Pinpoint 채널, 캠페인 및 여정 설정을 저장합니다.

절차

1. 에서 AWS 최종 사용자 메시지 푸시 콘솔을 엽니다. https://console.aws.amazon.com/push-notifications/
2. 애플리케이션을 선택한 다음 삭제를 선택합니다.
3. 애플리케이션 삭제 창에서 **`delete`** 를 입력한 다음 삭제를 선택합니다.

> **Important**
>
> Amazon Pinpoint 채널, 캠페인, 여정 또는 세그먼트도 모두 삭제됩니다.

모범 사례

고객의 이익을 가장 먼저 생각한다고 해도 메시지 발송률에 영향을 미치는 상황은 언제든지 발생할 수 있습니다. 다음 섹션에서는 푸시 커뮤니케이션이 목표 고객에게 도달하도록 하는 데 도움이 되는 권장 사항에 대해서 살펴봅니다.

대량의 푸시 알림 보내기

대량의 푸시 알림을 보내려면 먼저 계정이 처리량 요구 사항을 지원하도록 구성되어 있는지 확인하세요. 기본적으로 모든 계정은 초당 25,000개의 메시지를 보내도록 구성되어 있습니다. 1초에 25,000개 이상의 메시지를 보내야 하는 경우, 할당량 증가를 요청할 수 있습니다. 자세한 내용은 AWS 최종 사용자 메시지 푸시 할당량 단원을 참조하십시오.

사용하려는 각 푸시 알림 제공자의 자격 증명 (예: FCM 또는APNs) 을 사용하여 계정을 올바르게 구성했는지 확인하십시오.

마지막으로, 예외를 처리하는 방법을 고안하세요. 푸시 알림 서비스마다 제공하는 예외 메시지가 다릅니다. 트랜잭션 전송의 경우 API 호출에 대해 기본 상태 코드인 200을 받게 되며, 메시지 전송 중에 해당 플랫폼 토큰 (예:FCM) 또는 인증서 (예:) 가 유효하지 않은 것으로 확인되면 엔드포인트당 상태 코드 400이 영구 실패로 표시됩니다. APN

AWS 최종 사용자 메시지 푸시의 보안

클라우드 AWS 보안이 최우선 과제입니다. AWS 고객은 가장 보안에 민감한 조직의 요구 사항을 충족하도록 구축된 데이터 센터 및 네트워크 아키텍처의 혜택을 누릴 수 있습니다.

보안은 기업과 귀사 간의 공동 책임입니다. AWS 공동 책임 모델은 이 사항을 클라우드의 보안 및 클라우드 내 보안으로 설명합니다.

- 클라우드 보안 - AWS 클라우드에서 AWS 서비스를 실행하는 인프라를 보호하는 역할을 합니다 AWS 클라우드. AWS 또한 안전하게 사용할 수 있는 서비스를 제공합니다. AWS 최종 사용자 메시지 푸시에 적용되는 규정 준수 프로그램에 대해 자세히 알아보려면 규정 준수 프로그램별 범위 내 AWSAWS 서비스 규정 준수 프로그램별 참조하십시오.
- 클라우드에서의 보안 — 사용하는 AWS 서비스에 따라 책임이 결정됩니다. 또한 귀하는 귀사의 데이터의 민감도, 귀사의 요구 사항, 관련 법률 및 규정을 비롯한 기타 요소에 대해서도 책임이 있습니다.

이 설명서는 AWS 최종 사용자 메시지 푸시를 사용할 때 공동 책임 모델을 적용하는 방법을 이해하는 데 도움이 됩니다. 다음 항목에서는 보안 및 규정 준수 목표를 충족하도록 AWS 최종 사용자 메시지 푸시를 구성하는 방법을 보여줍니다. 또한 AWS 최종 사용자 메시지 푸시 리소스를 모니터링하고 보호하는 데 도움이 되는 다른 AWS 서비스를 사용하는 방법도 알아봅니다.

주제

AWS 최종 사용자 메시지 푸시의 데이터 보호

AWS 공동 책임 모델 AWS 최종 사용자 메시지 푸시의 데이터 보호에 적용됩니다. 이 모델에 설명된 대로 AWS 는 모든 데이터를 실행하는 글로벌 인프라를 보호하는 역할을 AWS 클라우드합니다. 사용자는 인프라에서 호스팅되는 콘텐츠를 관리해야 합니다. 사용하는 AWS 서비스 의 보안 구성과 관리 작업에 대한 책임도 사용자에게 있습니다. 데이터 프라이버시에 대한 자세한 내용은 데이터 프라이버시

를 참조하십시오FAQ. 유럽의 데이터 보호에 대한 자세한 내용은 AWS 공동 책임 모델 및AWS 보안 GDPR 블로그의 블로그 게시물을 참조하십시오.

데이터 보호를 위해 AWS 계정 자격 증명을 보호하고 개별 사용자에게 AWS IAM Identity Center 또는 AWS Identity and Access Management (IAM) 를 설정하는 것이 좋습니다. 이렇게 하면 개별 사용자에게 자신의 직무를 충실히 이행하는 데 필요한 권한만 부여됩니다. 또한 다음과 같은 방법으로 데이터를 보호하는 것이 좋습니다.

- 각 계정마다 다단계 인증 (MFA) 을 사용하십시오.
- SSLTLS/를 사용하여 AWS 리소스와 통신하세요. TLS1.2가 필요하고 TLS 1.3을 권장합니다.
- API를 사용하여 사용자 활동 로깅을 설정합니다 AWS CloudTrail.
- 포함된 모든 기본 보안 제어와 함께 AWS 암호화 솔루션을 사용하십시오 AWS 서비스.
- Amazon S3에 저장된 민감한 데이터를 검색하고 보호하는 데 도움이 되는 Amazon Macie와 같은 고급 관리형 보안 서비스를 사용하세요.
- 명령줄 인터페이스 또는 API an을 AWS 통해 액세스할 때 FIPS 140-2개의 검증된 암호화 모듈이 필요한 경우 엔드포인트를 사용하십시오. FIPS 사용 가능한 FIPS 엔드포인트에 대한 자세한 내용은 연방 정보 처리 표준 () 140-2를 참조하십시오. FIPS

고객의 이메일 주소와 같은 기밀 정보나 중요한 정보는 태그나 이름 필드와 같은 자유 양식 필드에 입력하지 않는 것이 좋습니다. 여기에는 콘솔, API AWS CLI, 또는 를 AWS 서비스 사용하여 AWS 최종 사용자 메시지 푸시 또는 기타 작업을 하는 경우가 포함됩니다. AWS SDKs 이름에 사용되는 태그 또는 자유 형식 텍스트 필드에 입력하는 모든 데이터는 청구 또는 진단 로그에 사용될 수 있습니다. 외부 서버에 URL a를 제공하는 경우 해당 서버에 대한 요청을 URL 검증하기 위해 자격 증명 정보를 에 포함하지 않는 것이 좋습니다.

데이터 암호화

AWS 최종 사용자 메시지 푸시 데이터는 전송 및 저장 시 암호화됩니다. AWS 최종 사용자 메시지 푸시에 데이터를 제출하면 데이터를 수신하고 저장하는 동안 데이터가 암호화됩니다. AWS 최종 사용자 메시지 푸시에서 데이터를 검색하면 최신 보안 프로토콜을 사용하여 데이터를 사용자에게 전송합니다.

저장 중 암호화

AWS 최종 사용자 메시지 푸시는 사용자를 위해 저장하는 모든 데이터를 암호화합니다. 여기에는 구성 데이터, 사용자 및 엔드포인트 데이터, 분석 데이터, AWS 최종 사용자 메시징 푸시에 추가하거나 가져오는 모든 데이터가 포함됩니다. 데이터를 암호화하기 위해 AWS 최종 사용자 메시지 푸시는 서비스가 사용자를 대신하여 소유하고 유지 관리하는 내부 AWS Key Management Service (AWS

KMS) 키를 사용합니다. 이들 키는 정기적으로 교체됩니다. 에 대한 AWS KMS자세한 내용은 AWS Key Management Service 개발자 안내서를 참조하십시오.

전송 중 암호화

AWS 최종 사용자 메시지 푸시는 전송 계층 보안 (TLS) 1.2 이상을 사용하여 HTTPS 클라이언트 및 애플리케이션과 통신합니다. AWS 최종 사용자 메시지 푸시는 다른 AWS 서비스와 통신하기 위해 HTTPS 및 TLS 1.2를 사용합니다. 또한 콘솔 AWS SDK, 또는 를 사용하여 AWS 최종 사용자 메시지 푸시 리소스를 만들고 관리하는 경우 HTTPS 및 TLS 1.2를 사용하여 모든 통신의 AWS Command Line Interface보안을 유지합니다.

키 관리

AWS 최종 사용자 메시지 푸시 데이터를 암호화하기 위해 AWS 최종 사용자 메시지 푸시는 서비스가 사용자를 대신하여 소유하고 유지 관리하는 내부 AWS KMS 키를 사용합니다. 이들 키는 정기적으로 교체됩니다. AWS 최종 사용자 메시지 푸시에 저장하는 데이터를 자체 AWS KMS 키나 다른 키를 프로비저닝하고 사용하여 암호화할 수는 없습니다.

인터네트워크 트래픽 개인 정보 보호

인터넷 트래픽 개인 정보 보호란 AWS 최종 사용자 메시징 푸시와 온-프레미스 클라이언트 및 애플리케이션 간, AWS 최종 사용자 메시징 푸시와 같은 지역의 다른 AWS 리소스 간의 연결 및 트래픽을 보호하는 것을 말합니다. AWS 다음 기능 및 방법은 AWS 최종 사용자 메시지 푸시의 네트워크 간 트래픽 프라이버시를 보장하는 데 도움이 될 수 있습니다.

AWS 최종 사용자 메시지 푸시와 온-프레미스 클라이언트 및 애플리케이션 간의 트래픽

를 사용하여 AWS 최종 사용자 메시지 푸시와 온-프레미스 네트워크의 클라이언트 및 애플리케이션 간에 비공개 연결을 설정할 수 있습니다. AWS Direct Connect이렇게 하면 표준 광섬유 이더넷 케이블을 사용하여 네트워크를 AWS Direct Connect 위치에 연결할 수 있습니다. 케이블의 한쪽 끝이 라우터에 연결되어 있습니다. 다른 쪽 끝은 AWS Direct Connect 라우터에 연결되어 있습니다. 자세한 내용은 AWS Direct Connect사용 설명서의 AWS Direct Connect 이란 무엇입니까? 섹션을 참조하십시오.

APIs게시를 통해 AWS 최종 사용자 메시지 푸시에 안전하게 액세스하려면 API 통화에 대한 AWS 최종 사용자 메시지 푸시 요구 사항을 준수하는 것이 좋습니다. AWS 최종 사용자 메시지 푸시를 사용하려면 클라이언트가 전송 계층 보안 (TLS) 1.2 이상을 사용해야 합니다. 또한 클라이언트는 Ephemeral Diffie-Hellman () 또는 Eliptic Curve Diffie-Hellman Ephemeral (PFS) 과 같이 완벽한 순방향 기밀성 () 을 갖춘 암호 제품군을 지원해야 합니다. DHE ECDHE Java 7 이상의 최신 시스템은 대부분 이러한 모드를 지원합니다.

또한 계정의 () 보안 주체와 연결된 액세스 키 ID 및 비밀 액세스 키를 사용하여 요청에 서명해야 합니다. AWS Identity and Access Management IAM AWS 또는 AWS Security Token Service(AWS STS)를 사용해 임시 보안 자격 증명을 생성하여 요청에 서명할 수 있습니다.

AWS 최종 사용자 메시지 푸시와 다른 AWS 리소스 간의 트래픽

AWS 최종 사용자 메시지 푸시와 동일한 AWS 지역의 다른 AWS 리소스 간의 통신을 보호하기 위해 AWS 최종 사용자 메시지 푸시는 기본적으로 A HTTPS 및 TLS 1.2를 사용합니다.

AWS 최종 사용자 메시지 푸시의 ID 및 액세스 관리

AWS Identity and Access Management (IAM) 는 관리자가 AWS 리소스에 대한 액세스를 안전하게 제어할 수 AWS 서비스 있도록 도와줍니다. IAM관리자는 AWS 최종 사용자 메시지 푸시 리소스를 사용할 수 있는 인증 (로그인) 및 권한 부여 (권한 보유) 를 받을 수 있는 사용자를 제어합니다. IAM추가 비용 없이 사용할 AWS 서비스 수 있습니다.

주제

- 고객
- ID를 통한 인증
- 정책을 사용한 액세스 관리
- AWS 최종 사용자 메시지 푸시의 작동 방식 IAM
- 최종 사용자 메시지 푸시에 대한 AWS ID 기반 정책 예제
- AWS 최종 사용자 메시지 푸시 ID 및 액세스 문제 해결

고객

AWS Identity and Access Management (IAM) 사용 방법은 AWS 최종 사용자 메시지 푸시에서 수행하는 작업에 따라 다릅니다.

서비스 사용자 - AWS 최종 사용자 메시지 푸시 서비스를 사용하여 작업을 수행하는 경우 관리자가 필요한 자격 증명과 권한을 제공합니다. 작업에 더 많은 AWS 최종 사용자 메시지 푸시 기능을 사용함에 따라 추가 권한이 필요할 수 있습니다. 액세스 권한 관리 방식을 이해하면 적절한 권한을 관리자에게 요청할 수 있습니다. AWS 최종 사용자 메시지 푸시의 기능에 액세스할 수 없는 경우 을 참조하십시오AWS 최종 사용자 메시지 푸시 ID 및 액세스 문제 해결.

서비스 관리자 — 회사에서 AWS 최종 사용자 메시지 푸시 리소스를 담당하는 경우 AWS 최종 사용자 메시지 푸시에 대한 전체 액세스 권한을 가지고 있을 것입니다. 서비스 사용자가 액세스해야 하는

AWS 최종 사용자 메시지 푸시 기능 및 리소스를 결정하는 것은 여러분의 몫입니다. 그런 다음 IAM 관리자에게 서비스 사용자의 권한을 변경해 달라는 요청을 제출해야 합니다. 이 페이지의 정보를 검토하여 의 기본 개념을 IAM 이해하십시오. 회사에서 AWS 최종 사용자 메시지 푸시를 사용하는 IAM 방법에 대한 자세한 내용은 을 참조하십시오AWS 최종 사용자 메시지 푸시의 작동 방식 IAM.

IAM관리자 - IAM 관리자인 경우 AWS 최종 사용자 메시지 푸시에 대한 액세스를 관리하기 위한 정책을 작성하는 방법에 대해 자세히 알아보는 것이 좋습니다. 에서 IAM 사용할 수 있는 AWS 최종 사용자 메시지 푸시 ID 기반 정책의 예를 보려면 을 참조하십시오. 최종 사용자 메시지 푸시에 대한 AWS ID 기반 정책 예제

ID를 통한 인증

인증은 ID 자격 증명을 AWS 사용하여 로그인하는 방법입니다. IAM사용자로서 또는 역할을 위임하여 인증 (로그인 AWS) 을 받아야 합니다. AWS 계정 루트 사용자 IAM

ID 소스를 통해 제공된 자격 증명을 사용하여 페더레이션 ID로 로그인할 수 있습니다. AWS AWS IAM Identity Center (IAMID 센터) 사용자, 회사의 싱글 사인온 인증, Google 또는 Facebook 자격 증명이 페더레이션 ID의 예입니다. 페더레이션 ID로 로그인하는 경우 관리자는 이전에 역할을 사용하여 ID 페더레이션을 설정했습니다. IAM 페더레이션을 AWS 사용하여 액세스하는 경우 간접적으로 역할을 수임하는 것입니다.

사용자 유형에 따라 AWS Management Console 또는 AWS 액세스 포털에 로그인할 수 있습니다. 로그인에 대한 자세한 내용은 AWS 로그인 사용 설명서의 내 로그인 방법을 참조하십시오. AWS AWS 계정

AWS 프로그래밍 방식으로 액세스하는 경우 자격 증명을 사용하여 요청에 암호로 서명할 수 있는 소프트웨어 개발 키트 (SDKCLI) 와 명령줄 인터페이스 () 가 AWS 제공됩니다. AWS 도구를 사용하지 않는 경우 요청에 직접 서명해야 합니다. 권장 방법을 사용하여 직접 요청에 서명하는 방법에 대한 자세한 내용은 사용 IAM설명서의 AWS API요청 서명을 참조하십시오.

사용하는 인증 방법에 상관없이 추가 보안 정보를 제공해야 할 수도 있습니다. 예를 들어, 계정 보안을 강화하기 위해 다단계 인증 (MFA) 을 사용할 AWS 것을 권장합니다. 자세한 내용은 사용 설명서의 다단계 인증 및 사용 AWS IAM Identity Center 설명서의 다단계 인증 사용 (MFA) 을 IAM 참조하십시오. AWS

AWS 계정 루트 사용자

계정을 AWS 계정만들 때는 먼저 계정의 모든 AWS 서비스 리소스에 대한 완전한 액세스 권한을 가진 하나의 로그인 ID로 시작합니다. 이 ID를 AWS 계정 루트 사용자라고 하며, 계정을 만들 때 사용한 이메일 주소와 비밀번호로 로그인하여 액세스할 수 있습니다. 일상적인 태스크에 루트 사용자를 사용하지 않을 것을 강력히 권장합니다. 루트 사용자 보안 인증 정보를 보호하고 루트 사용자만 수행할 수 있는

태스크를 수행하는 데 사용하세요. 루트 사용자로 로그인해야 하는 작업의 전체 목록은 사용 설명서의 루트 사용자 자격 증명이 필요한 작업을 참조하십시오. IAM

페더레이션 자격 증명

가장 좋은 방법은 관리자 액세스가 필요한 사용자를 비롯한 수동 AWS 서비스 사용자가 ID 공급자와의 페더레이션을 사용하여 임시 자격 증명을 사용하여 액세스하도록 하는 것입니다.

페더레이션 ID는 기업 사용자 디렉토리, 웹 ID 공급자, Identity Center 디렉터리의 사용자 또는 ID 소스를 통해 제공된 자격 증명을 사용하여 액세스하는 AWS 서비스 모든 사용자를 말합니다. AWS Directory Service페더레이션 ID에 AWS 계정액세스하면 이들이 역할을 맡고 역할은 임시 자격 증명을 제공합니다.

중앙 집중식 액세스 관리를 위해 AWS IAM Identity Center(을)를 사용하는 것이 좋습니다. IAMIdentity Center에서 사용자 및 그룹을 만들거나 자체 ID 소스의 사용자 및 그룹 집합에 연결하고 동기화하여 AWS 계정 모든 사용자 및 애플리케이션에서 사용할 수 있습니다. ID 센터에 대한 자세한 내용은 IAM ID 센터란 IAM 무엇입니까? 를 참조하십시오. AWS IAM Identity Center 사용 설명서에서.

IAM 사용자 및 그룹

IAM사용자란 단일 사용자 또는 애플리케이션에 대한 특정 권한을 가진 사용자 내의 ID입니다. AWS 계정 가능하면 암호 및 액세스 키와 같은 장기 자격 증명을 가진 IAM 사용자를 만드는 대신 임시 자격 증명을 사용하는 것이 좋습니다. 하지만 특정 사용 사례에서 IAM 사용자의 장기 자격 증명이 필요한 경우에는 액세스 키를 교체하는 것이 좋습니다. 자세한 내용은 사용 설명서의 장기 자격 증명이 필요한 사용 사례에 대한 정기적인 액세스 키 IAM 교체를 참조하십시오.

IAM그룹은 IAM 사용자 컬렉션을 지정하는 ID입니다. 사용자는 그룹으로 로그인할 수 없습니다. 그룹을 사용하여 여러 사용자의 권한을 한 번에 지정할 수 있습니다. 그룹을 사용하면 대규모 사용자 집합의 권한을 더 쉽게 관리할 수 있습니다. 예를 들어 이름을 지정한 IAMAdmins그룹을 만들고 해당 그룹에 IAM 리소스를 관리할 권한을 부여할 수 있습니다.

사용자는 역할과 다릅니다. 사용자는 한 사람 또는 애플리케이션과 고유하게 연결되지만, 역할은 해당 역할이 필요한 사람이라면 누구나 수임할 수 있습니다. 사용자는 영구적인 장기 보안 인증 정보를 가지고 있지만, 역할은 임시 보안 인증만 제공합니다. 자세히 알아보려면 사용 설명서의 역할 대신 IAM 사용자를 만드는 시기를 참조하십시오. IAM

IAM역할

IAM역할은 특정 권한을 AWS 계정 가진 사용자 내의 ID입니다. IAM사용자와 비슷하지만 특정인과 관련이 있는 것은 아닙니다. 역할을 AWS Management Console 전환하여 에서 일시적으로 IAM 역할을

맡을 수 있습니다. AWS CLI or AWS API 작업을 호출하거나 사용자 지정을 사용하여 역할을 수임할 수 URL 있습니다. 역할 사용 방법에 대한 자세한 내용은 사용 IAM설명서의 IAM역할 사용을 참조하십시오.

IAM임시 자격 증명이 있는 역할은 다음과 같은 상황에서 유용합니다.

- 페더레이션 사용자 액세스 - 페더레이션 ID에 권한을 부여하려면 역할을 생성하고 해당 역할의 권한을 정의합니다. 페더레이션 ID가 인증되면 역할이 연결되고 역할에 정의된 권한이 부여됩니다. 페더레이션을 위한 역할에 대한 자세한 내용은 IAM사용 설명서의 타사 ID 제공자를 위한 역할 생성을 참조하십시오. IAMIdentity Center를 사용하는 경우 권한 집합을 구성합니다. ID가 인증된 후 액세스할 수 있는 대상을 제어하기 위해 IAM Identity Center는 권한 집합을 역할의 상관 관계와 연결합니다. IAM 권한 세트에 대한 자세한 내용은 AWS IAM Identity Center 사용 설명서의 권한 세트를 참조하십시오.
- 임시 IAM 사용자 권한 - IAM 사용자 또는 역할이 역할을 맡아 특정 작업에 대해 일시적으로 다른 권한을 부여받을 수 있습니다. IAM
- 계정 간 액세스 - IAM 역할을 사용하여 다른 계정의 사용자 (신뢰할 수 있는 사용자) 가 계정의 리소스에 액세스하도록 허용할 수 있습니다. 역할은 계정 간 액세스를 부여하는 기본적인 방법입니다. 하지만 일부 AWS 서비스경우에는 역할을 프록시로 사용하는 대신 정책을 리소스에 직접 연결할 수 있습니다. 계정 간 액세스에 대한 역할과 리소스 기반 정책의 차이점을 알아보려면 사용 설명서의 교차 계정 리소스 액세스를 참조하십시오. IAM IAM
- 서비스 간 액세스 — 일부는 다른 기능을 AWS 서비스 사용합니다. AWS 서비스예를 들어, 서비스를 호출하면 해당 서비스가 Amazon에서 애플리케이션을 EC2 실행하거나 Amazon S3에 객체를 저장하는 것이 일반적입니다. 서비스는 직접적으로 호출하는 보안 주체의 권한을 사용하거나, 서비스 역할을 사용하거나, 또는 서비스 연결 역할을 사용하여 이 태스크를 수행할 수 있습니다.
  - 전달 액세스 세션 (FAS) — IAM 사용자 또는 역할을 사용하여 작업을 수행하는 AWS경우 보안 주체로 간주됩니다. 일부 서비스를 사용하는 경우 다른 서비스에서 다른 작업을 시작하는 작업을 수행할 수 있습니다. FAS전화를 거는 주체의 권한을 다운스트림 서비스에 AWS 서비스 요청하라는 요청과 결합하여 사용합니다. AWS 서비스 FAS요청은 다른 서비스 AWS 서비스 또는 리소스와의 상호 작용이 필요한 요청을 서비스가 수신한 경우에만 이루어집니다. 이 경우 두 작업을 모두 수행할 수 있는 권한이 있어야 합니다. FAS요청 시 적용되는 정책 세부 정보는 전달 액세스 세션을 참조하십시오.
  - 서비스 역할 - 서비스 역할은 서비스가 사용자를 대신하여 작업을 수행하는 것으로 간주하는 IAM 역할입니다. IAM관리자는 내부에서 IAM 서비스 역할을 만들고, 수정하고, 삭제할 수 있습니다. 자세한 내용은 사용 설명서의 역할 만들기를 참조하여 권한을 위임하십시오IAM. AWS 서비스
  - 서비스 연결 역할 - 서비스 연결 역할은 에 연결된 서비스 역할 유형입니다. AWS 서비스서비스는 사용자를 대신하여 작업을 수행하기 위해 역할을 수임할 수 있습니다. 서비스 연결 역할은 사용자

에게 AWS 계정 표시되며 해당 서비스가 소유합니다. IAM관리자는 서비스 연결 역할에 대한 권한을 볼 수 있지만 편집할 수는 없습니다.

- Amazon에서 실행 중인 애플리케이션 EC2 — IAM 역할을 사용하여 EC2 인스턴스에서 실행되고 AWS API 요청을 보내는 애플리케이션의 임시 자격 증명을 관리할 수 있습니다. AWS CLI EC2인스턴스 내에 액세스 키를 저장하는 것보다 이 방법이 더 좋습니다. EC2인스턴스에 AWS 역할을 할당하고 모든 애플리케이션에서 사용할 수 있게 하려면 인스턴스에 연결된 인스턴스 프로필을 만들어야 합니다. 인스턴스 프로필에는 역할이 포함되며, 이를 통해 EC2 인스턴스에서 실행 중인 프로그램이 임시 자격 증명을 얻을 수 있습니다. 자세한 내용은 사용 설명서의 IAM역할을 사용하여 Amazon EC2 인스턴스에서 실행되는 애플리케이션에 권한 부여를 IAM 참조하십시오.

IAM역할을 사용할지 IAM 사용자를 사용할지 알아보려면 사용 설명서의 IAM 역할 생성 시기 (사용자 대신) 를 IAM참조하십시오.

정책을 사용한 액세스 관리

정책을 만들고 이를 AWS ID 또는 리소스에 AWS 연결하여 액세스를 제어할 수 있습니다. 정책은 ID 또는 리소스와 연결될 때 AWS 해당 권한을 정의하는 객체입니다. AWS 주도자 (사용자, 루트 사용자 또는 역할 세션) 가 요청할 때 이러한 정책을 평가합니다. 정책에서 권한은 요청이 허용되거나 거부되는지를 결정합니다. 대부분의 정책은 JSON 문서로 AWS 저장됩니다. JSON정책 문서의 구조 및 내용에 대한 자세한 내용은 IAM사용 설명서의 JSON 정책 개요를 참조하십시오.

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

기본적으로, 사용자와 역할에는 어떠한 권한도 없습니다. IAM관리자는 IAM 정책을 생성하여 필요한 리소스에서 작업을 수행할 수 있는 권한을 사용자에게 부여할 수 있습니다. 그러면 관리자가 역할에 IAM 정책을 추가할 수 있으며, 사용자는 역할을 수임할 수 있습니다.

IAM정책은 작업을 수행하는 데 사용하는 방법에 관계없이 작업에 대한 권한을 정의합니다. 예를 들어, `iam:GetRole` 작업을 허용하는 정책이 있다고 가정합니다. 해당 정책을 사용하는 사용자는 AWS Management Console, AWS CLI, 또는 에서 역할 정보를 가져올 수 AWS API 있습니다.

보안 인증 기반 정책

ID 기반 정책은 IAM 사용자, 사용자 그룹 또는 역할과 같은 ID에 연결할 수 있는 JSON 권한 정책 문서입니다. 이러한 정책은 사용자와 역할이 어떤 리소스와 어떤 조건에서 어떤 태스크를 수행할 수 있는지를 제어합니다. ID 기반 정책을 만드는 방법을 알아보려면 사용 설명서의 IAM정책 생성을 참조하십시오. IAM

보안 인증 기반 정책은 인라인 정책 또는 관리형 정책으로 한층 더 분류할 수 있습니다. 인라인 정책은 단일 사용자, 그룹 또는 역할에 직접 포함됩니다. 관리형 정책은 내 여러 사용자, 그룹 및 역할에 연결할 수 있는 독립형 정책입니다. AWS 계정관리형 정책에는 AWS 관리형 정책과 고객 관리형 정책이 포함됩니다. 관리형 정책과 인라인 정책 중에서 선택하는 방법을 알아보려면 IAM사용 설명서의 관리형 정책과 인라인 정책 중 선택을 참조하십시오.

리소스 기반 정책

리소스 기반 정책은 리소스에 연결하는 JSON 정책 문서입니다. 리소스 기반 정책의 예로는 IAM 역할 신뢰 정책과 Amazon S3 버킷 정책이 있습니다. 리소스 기반 정책을 지원하는 서비스에서 서비스 관리자는 이러한 정책을 사용하여 특정 리소스에 대한 액세스를 통제할 수 있습니다. 정책이 연결된 리소스의 경우, 정책은 지정된 보안 주체가 해당 리소스와 어떤 조건에서 어떤 태스크를 수행할 수 있는지를 정의합니다. 리소스 기반 정책에서 보안 주체를 지정해야 합니다. 보안 주체에는 계정, 사용자, 역할, 연동 사용자 등이 포함될 수 있습니다. AWS 서비스

리소스 기반 정책은 해당 서비스에 있는 인라인 정책입니다. 리소스 기반 IAM 정책에서는 AWS 관리형 정책을 사용할 수 없습니다.

액세스 제어 목록 () ACLs

액세스 제어 목록 (ACLs) 은 리소스에 액세스할 수 있는 권한을 가진 주체 (계정 구성원, 사용자 또는 역할) 를 제어합니다. ACLs정책 문서 형식을 사용하지는 않지만 리소스 기반 정책과 JSON 비슷합니다.

지원하는 서비스의 VPC 예로는 Amazon S3와 Amazon이 ACLs 있습니다. AWS WAF자세한 내용은 Amazon 심플 스토리지 서비스 개발자 안내서의 액세스 제어 목록 (ACL) 개요를 참조하십시오. ACLs

기타 정책 타입

AWS 일반적이지 않은 추가 정책 유형을 지원합니다. 이러한 정책 타입은 더 일반적인 정책 타입에 따라 사용자에게 부여되는 최대 권한을 설정할 수 있습니다.

- 권한 경계 - 권한 경계는 ID 기반 정책이 IAM 엔티티 (IAM사용자 또는 역할) 에 부여할 수 있는 최대 권한을 설정하는 고급 기능입니다. 개체에 대한 권한 경계를 설정할 수 있습니다. 그 결과로 얻는 권한은 개체의 보안 인증 기반 정책과 그 권한 경계의 교집합입니다. `Principal` 필드에서 사용자나 역할을 지정하는 리소스 기반 정책은 권한 경계를 통해 제한되지 않습니다. 이러한 정책 중 하나에 포함된 명시적 거부는 허용을 재정의합니다. 권한 경계에 대한 자세한 내용은 사용 IAM설명서의 IAM 엔티티의 권한 경계를 참조하십시오.
- 서비스 제어 정책 (SCPs) - SCPs 조직 또는 OU (조직 단위) 에 대한 최대 권한을 지정하는 JSON AWS Organizations정책입니다. AWS Organizations 기업이 소유한 여러 AWS 계정 개를 그룹화하

고 중앙에서 관리하는 서비스입니다. 조직의 모든 기능을 사용하도록 설정하면 일부 또는 모든 계정에 서비스 제어 정책 (SCPs) 을 적용할 수 있습니다. 각 항목을 포함하여 구성원 계정의 엔티티에 대한 권한을 SCP AWS 계정 루트 사용자제한합니다. Organizations 및 SCPs 에 대한 자세한 내용은 AWS Organizations 사용 설명서의 SCPs작업 방식을 참조하십시오.

- 세션 정책 - 세션 정책은 역할 또는 페더레이션 사용자에 대해 임시 세션을 프로그래밍 방식으로 생성할 때 파라미터로 전달하는 고급 정책입니다. 결과적으로 얻는 세션의 권한은 사용자 또는 역할의 보안 인증 기반 정책의 교차와 세션 정책입니다. 또한 권한을 리소스 기반 정책에서 가져올 수도 있습니다. 이러한 정책 중 하나에 포함된 명시적 거부는 허용을 재정의합니다. 자세한 내용은 IAM사용 설명서의 세션 정책을 참조하십시오.

여러 정책 타입

여러 정책 유형이 요청에 적용되는 경우, 결과 권한은 이해하기가 더 복잡합니다. 여러 정책 유형이 관련된 경우 요청을 허용할지 여부를 AWS 결정하는 방법을 알아보려면 IAM사용 설명서의 정책 평가 로직을 참조하십시오.

AWS 최종 사용자 메시지 푸시의 작동 방식 IAM

를 IAM 사용하여 최종 사용자 메시지 푸시에 대한 액세스를 관리하기 전에 AWS 최종 사용자 메시지 푸시와 함께 AWS 사용할 수 있는 IAM 기능에 대해 알아보세요.

IAM AWS 최종 사용자 메시지 푸시와 함께 사용할 수 있는 기능

IAM기능	AWS 최종 사용자 메시지 푸시 지원
ID 기반 정책	예
리소스 기반 정책	예
정책 작업	예
정책 리소스	예
정책 조건 키	예
ACLs	아니요
ABAC(정책의 태그)	부분

IAM기능	AWS 최종 사용자 메시지 푸시 지원
임시 보안 인증	예
보안 주체 권한	예
서비스 역할	예
서비스 연결 역할	아니요

AWS 최종 사용자 메시지 푸시 및 기타 AWS 서비스가 대부분의 IAM 기능과 어떻게 작동하는지 자세히 보려면 IAM사용 IAM 설명서에서 함께 작동하는AWS 서비스를 참조하십시오.

AWS 최종 사용자 메시지 푸시에 대한 ID 기반 정책

자격 증명 기반 정책 지원: 예

ID 기반 정책은 IAM 사용자, 사용자 그룹 또는 역할과 같은 ID에 연결할 수 있는 JSON 권한 정책 문서입니다. 이러한 정책은 사용자와 역할이 어떤 리소스와 어떤 조건에서 어떤 태스크를 수행할 수 있는지를 제어합니다. ID 기반 정책을 만드는 방법을 알아보려면 사용 설명서의 IAM정책 생성을 참조하십시오. IAM

IAMID 기반 정책을 사용하면 허용 또는 거부된 작업 및 리소스는 물론 작업이 허용되거나 거부되는 조건을 지정할 수 있습니다. 보안 인증 기반 정책에서는 보안 주체가 연결된 사용자 또는 역할에 적용되므로 보안 주체를 지정할 수 없습니다. JSON정책에서 사용할 수 있는 모든 요소에 대해 알아보려면 사용 IAM설명서의 IAM JSON 정책 요소 참조를 참조하십시오.

AWS 최종 사용자 메시지 푸시에 대한 ID 기반 정책 예제

AWS 최종 사용자 메시지 푸시 ID 기반 정책의 예를 보려면 을 참조하십시오. 최종 사용자 메시지 푸시에 대한 AWS ID 기반 정책 예제

최종 사용자 메시지 푸시 내의 AWS 리소스 기반 정책

리소스 기반 정책 지원: 예

리소스 기반 정책은 리소스에 연결하는 JSON 정책 문서입니다. 리소스 기반 정책의 예로는 IAM 역할 신뢰 정책과 Amazon S3 버킷 정책이 있습니다. 리소스 기반 정책을 지원하는 서비스에서 서비스 관리자는 이러한 정책을 사용하여 특정 리소스에 대한 액세스를 통제할 수 있습니다. 정책이 연결된 리소스

의 경우, 정책은 지정된 보안 주체가 해당 리소스와 어떤 조건에서 어떤 태스크를 수행할 수 있는지를 정의합니다. 리소스 기반 정책에서 보안 주체를 지정해야 합니다. 보안 주체에는 계정, 사용자, 역할, 연동 사용자 등이 포함될 수 있습니다. AWS 서비스

계정 간 액세스를 활성화하려면 다른 계정의 전체 계정 또는 IAM 엔티티를 리소스 기반 정책의 보안 주체로 지정할 수 있습니다. 리소스 기반 정책에 크로스 계정 보안 주체를 추가하는 것은 트러스트 관계 설정의 절반밖에 되지 않는다는 것을 유념하십시오. 보안 주체와 리소스가 다른 AWS 계정경우 신뢰할 수 있는 계정의 IAM 관리자는 보안 주체 (사용자 또는 역할) 에게 리소스에 액세스할 수 있는 권한도 부여해야 합니다. 엔터티에 ID 기반 정책을 연결하여 권한을 부여합니다. 하지만 리소스 기반 정책이 동일 계정의 보안 주체에 액세스를 부여하는 경우, 추가 자격 증명 기반 정책이 필요하지 않습니다. 자세한 내용은 IAM사용 설명서의 계정 간 리소스 액세스를 참조하십시오. IAM

AWS 최종 사용자 메시지 푸시에 대한 정책 조치

정책 작업 지원: 예

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

정책 `Action` 요소는 JSON 정책에서 액세스를 허용하거나 거부하는 데 사용할 수 있는 작업을 설명합니다. 정책 작업은 일반적으로 관련 AWS API 작업과 이름이 같습니다. 일치하는 작업이 없는 권한 전용 작업과 같은 몇 가지 예외가 있습니다. API 정책에서 여러 작업이 필요한 몇 가지 작업도 있습니다. 이러한 추가 작업을 일컬어 종속 작업이라고 합니다.

연결된 작업을 수행할 수 있는 권한을 부여하기 위한 정책에 작업을 포함하십시오.

AWS 최종 사용자 메시지 푸시 작업 목록을 보려면 서비스 권한 부여 참조의 AWS 최종 사용자 메시지 푸시에 의해 정의된 작업을 참조하십시오.

AWS 최종 사용자 메시지 푸시의 정책 작업은 작업 앞에 다음 접두사를 사용합니다.

```
mobiletargeting
```

단일 문에서 여러 작업을 지정하려면 다음과 같이 쉼표로 구분합니다.

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

AWS 최종 사용자 메시지 푸시 ID 기반 정책의 예를 보려면 을 참조하십시오. 최종 사용자 메시지 푸시에 대한 AWS ID 기반 정책 예제

AWS 최종 사용자 메시지 푸시를 위한 정책 리소스

정책 리소스 지원: 예

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

`Resource`JSON정책 요소는 작업이 적용되는 하나 또는 여러 개의 객체를 지정합니다. 문장에는 `Resource`또는 `NotResource`요소가 반드시 추가되어야 합니다. Amazon 리소스 이름 (ARN) 을 사용하여 리소스를 지정하는 것이 가장 좋습니다. 리소스 수준 권한이라고 하는 특정 리소스 유형을 지원하는 작업에 대해 이 태스크를 수행할 수 있습니다.

작업 나열과 같이 리소스 수준 권한을 지원하지 않는 작업의 경우, 와일드카드(*)를 사용하여 해당 문이 모든 리소스에 적용됨을 나타냅니다.

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

AWS 최종 사용자 메시징 푸시 리소스 유형 및 해당 ARNs 유형의 목록을 보려면 서비스 권한 부여 참조의 AWS 최종 사용자 메시징 푸시로 정의된 리소스를 참조하십시오. 각 리소스의 어떤 작업을 지정할 수 있는지 알아보려면 AWS 최종 사용자 메시지 푸시로 정의된 작업을 참조하십시오. ARN

AWS 최종 사용자 메시지 푸시 ID 기반 정책의 예를 보려면 을 참조하십시오. 최종 사용자 메시지 푸시에 대한 AWS ID 기반 정책 예제

AWS 최종 사용자 메시지 푸시의 정책 조건 키

서비스별 정책 조건 키 지원: 예

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

`Condition` 요소(또는 `Condition` 블록)를 사용하면 정책이 발효되는 조건을 지정할 수 있습니다. `Condition` 요소는 옵션입니다. 같거나 작음과 같은 조건 연산자를 사용하여 정책의 조건을 요청의 값과 일치시키는 조건식을 생성할 수 있습니다.

한 문에서 여러 Condition 요소를 지정하거나 단일 Condition 요소에서 여러 키를 지정하는 경우, AWS 는 논리적 AND 태스크를 사용하여 평가합니다. 단일 조건 키에 여러 값을 지정하는 경우 는 논리적 OR 연산을 사용하여 조건을 AWS 평가합니다. 명문의 권한을 부여하기 전에 모든 조건을 충족해야 합니다.

조건을 지정할 때 자리 표시자 변수를 사용할 수도 있습니다. 예를 들어 리소스에 IAM 사용자 이름이 태그가 지정된 경우에만 리소스에 대한 액세스 권한을 IAM 사용자에게 부여할 수 있습니다. 자세한 내용은 IAM사용 설명서의 IAM 정책 요소: 변수 및 태그를 참조하십시오.

AWS 글로벌 조건 키 및 서비스별 조건 키를 지원합니다. 모든 AWS 글로벌 조건 키를 보려면 IAM사용 설명서의AWS 글로벌 조건 컨텍스트 키를 참조하십시오.

AWS 최종 사용자 메시지 푸시 조건 키 목록을 보려면 서비스 권한 부여 참조의 AWS 최종 사용자 메시지 푸시의 조건 키를 참조하십시오. 조건 키를 사용할 수 있는 작업 및 리소스를 알아보려면 AWS 최종 사용자 메시지 푸시로 정의된 작업을 참조하십시오.

AWS 최종 사용자 메시지 푸시 ID 기반 정책의 예를 보려면 을 참조하십시오. 최종 사용자 메시지 푸시에 대한 AWS ID 기반 정책 예제

ACLs AWS 최종 사용자 메시지 푸시에서

지원ACLs: 아니요

액세스 제어 목록 (ACLs) 은 리소스에 액세스할 수 있는 권한을 가진 주체 (계정 구성원, 사용자 또는 역할) 를 제어합니다. ACLs정책 문서 형식을 사용하지는 않지만 리소스 기반 정책과 JSON 비슷합니다.

ABAC AWS 최종 사용자 메시지 푸시 사용

지원 ABAC (정책의 태그): 부분

속성 기반 액세스 제어 (ABAC) 는 속성을 기반으로 권한을 정의하는 권한 부여 전략입니다. AWS에서는 이러한 속성을 태그라고 합니다. IAM엔티티 (사용자 또는 역할) 및 여러 AWS 리소스에 태그를 첨부할 수 있습니다. 의 ABAC 첫 번째 단계는 엔티티와 리소스에 태그를 지정하는 것입니다. 그런 다음 보안 주체의 태그가 액세스하려는 리소스의 태그와 일치할 때 작업을 허용하도록 ABAC 정책을 설계합니다.

ABAC빠르게 성장하는 환경에서 유용하며 정책 관리가 복잡해지는 상황에도 도움이 됩니다.

태그에 근거하여 액세스를 제어하려면 `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` 또는 `aws:TagKeys` 조건 키를 사용하여 정책의 조건 요소에 태그 정보를 제공합니다.

서비스가 모든 리소스 유형에 대해 세 가지 조건 키를 모두 지원하는 경우, 값은 서비스에 대해 예입니다. 서비스가 일부 리소스 유형에 대해서만 세 가지 조건 키를 모두 지원하는 경우, 값은 부분적입니다.

에 대한 자세한 내용은 What is ABAC? 를 참조하십시오. ABAC IAM사용 설명서에서. 설정 ABAC 단계가 포함된 자습서를 보려면 사용 IAM설명서의 속성 기반 액세스 제어 사용 (ABAC) 을 참조하십시오.

AWS 최종 사용자 메시지 푸시에 임시 자격 증명 사용

임시 자격 증명 지원: 예

임시 자격 증명을 사용하여 로그인하면 작동하지 AWS 서비스 않는 것도 있습니다. 임시 자격 증명을 사용하는 AWS 서비스 방법을 비롯한 추가 정보는 IAM사용 설명서의 AWS 서비스 해당 자격 증명을 참조하십시오. IAM

사용자 이름과 암호를 제외한 다른 방법을 AWS Management Console 사용하여 로그인하는 경우 임시 자격 증명을 사용하는 것입니다. 예를 들어 회사의 Single Sign-On (SSO) 링크를 AWS 사용하여 액세스하는 경우 이 프로세스에서 자동으로 임시 자격 증명을 생성합니다. 또한 콘솔에 사용자로 로그인한 다음 역할을 전환할 때 임시 보안 인증을 자동으로 생성합니다. 역할 전환에 대한 자세한 내용은 IAM사용 설명서의 역할 전환 (콘솔) 을 참조하십시오.

AWS CLI 또는 를 사용하여 임시 자격 증명을 수동으로 생성할 수 AWS API 있습니다. 그런 다음 해당 임시 자격 증명을 사용하여 액세스할 수 AWS있습니다. AWS 장기 액세스 키를 사용하는 대신 임시 자격 증명을 동적으로 생성할 것을 권장합니다. 자세한 내용은 의 임시 보안 자격 증명을 참조하십시오. IAM

AWS 최종 사용자 메시지 푸시에 대한 서비스 간 보안 주체 권한

순방향 액세스 세션 지원 (FAS): 예

에서 IAM 사용자 또는 역할을 사용하여 작업을 수행하는 AWS경우 사용자는 보안 주체로 간주됩니다. 일부 서비스를 사용하는 경우 다른 서비스에서 다른 작업을 시작하는 작업을 수행할 수 있습니다. FAS를 호출하는 주체의 권한을 다운스트림 서비스에 AWS 서비스 요청하기 위한 요청과 결합하여 사용합니다. AWS 서비스 FAS요청은 다른 서비스 AWS 서비스 또는 리소스와의 상호 작용이 필요한 요청을 서비스가 수신한 경우에만 이루어집니다. 이 경우 두 작업을 모두 수행할 수 있는 권한이 있어야 합니다. FAS요청 시 적용되는 정책 세부 정보는 전달 액세스 세션을 참조하십시오.

AWS 최종 사용자 메시지 푸시의 서비스 역할

서비스 역할 지원: 예

서비스 역할은 서비스가 사용자를 대신하여 작업을 수행하는 것으로 간주하는 IAM역할입니다. IAM관리자는 내부에서 IAM 서비스 역할을 생성, 수정 및 삭제할 수 있습니다. 자세한 내용은 사용 설명서의 역할 만들기를 참조하여 권한을 위임하십시오IAM. AWS 서비스

> **Warning**
>
> 서비스 역할의 권한을 변경하면 AWS 최종 사용자 메시지 푸시 기능이 중단될 수 있습니다. AWS 최종 사용자 메시지 푸시가 이에 대한 지침을 제공하는 경우에만 서비스 역할을 편집하십시오.

AWS 최종 사용자 메시지 푸시의 서비스 연결 역할

서비스 링크 역할 지원: 아니요

서비스 연결 역할은 에 연결된 서비스 역할 유형입니다. AWS 서비스서비스는 사용자를 대신하여 작업을 수행하기 위해 역할을 수임할 수 있습니다. 서비스 연결 역할은 사용자에게 AWS 계정 표시되며 해당 서비스가 소유합니다. IAM관리자는 서비스 연결 역할에 대한 권한을 볼 수 있지만 편집할 수는 없습니다.

서비스 연결 역할을 만들거나 관리하는 방법에 대한 자세한 내용은 함께 작동하는 AWS 서비스를 참조하십시오. IAM 서비스 연결 역할 열에서 Yes(이)가 포함된 서비스를 테이블에서 찾습니다. 해당 서비스에 대한 서비스 연결 역할 설명서를 보려면 Yes(네) 링크를 선택합니다.

최종 사용자 메시지 푸시에 대한 AWS ID 기반 정책 예제

기본적으로 사용자와 역할에는 AWS 최종 사용자 메시징 푸시 리소스를 만들거나 수정할 권한이 없습니다. 또한 AWS Management Console, AWS Command Line Interface (AWS CLI) 또는 를 사용하여 작업을 수행할 수도 없습니다 AWS API. IAM관리자는 IAM 정책을 생성하여 필요한 리소스에서 작업을 수행할 수 있는 권한을 사용자에게 부여할 수 있습니다. 그러면 관리자가 역할에 IAM 정책을 추가할 수 있으며, 사용자는 역할을 수임할 수 있습니다.

이러한 예제 JSON 정책 문서를 사용하여 IAM ID 기반 정책을 만드는 방법을 알아보려면 사용 IAM설명서에서 IAM 정책 생성을 참조하십시오.

각 리소스 유형의 형식을 포함하여 AWS 최종 사용자 메시지 푸시에 정의된 작업 및 리소스 유형에 ARNs 대한 자세한 내용은 서비스 권한 부여 참조의 AWS 최종 사용자 메시지 푸시를 위한 작업, 리소스 및 조건 키를 참조하십시오.

주제

- 정책 모범 사례
- AWS 최종 사용자 메시지 푸시 콘솔 사용
- 사용자가 자신의 고유한 권한을 볼 수 있도록 허용

정책 모범 사례

ID 기반 정책은 누군가가 계정의 AWS 최종 사용자 메시지 푸시 리소스를 생성, 액세스 또는 삭제할 수 있는지 여부를 결정합니다. 이 작업으로 인해 AWS 계정에 비용이 발생할 수 있습니다. ID 기반 정책을 생성하거나 편집할 때는 다음 지침과 권장 사항을 따릅니다.

- AWS 관리형 정책을 시작하고 최소 권한 권한으로 이동 — 사용자와 워크로드에 권한을 부여하려면 여러 일반적인 사용 사례에 권한을 부여하는 AWS 관리형 정책을 사용하세요. 에서 사용할 수 있습니다. AWS 계정사용 사례에 맞는 AWS 고객 관리형 정책을 정의하여 권한을 더 줄이는 것이 좋습니다. 자세한 내용은 IAM사용 설명서의 AWS 관리형 정책 또는 작업 기능에 대한AWS 관리형 정책을 참조하십시오.
- 최소 권한 적용 — IAM 정책으로 권한을 설정하는 경우 작업 수행에 필요한 권한만 부여하십시오. 이렇게 하려면 최소 권한으로 알려진 특정 조건에서 특정 리소스에 대해 수행할 수 있는 작업을 정의합니다. 를 사용하여 권한을 IAM 적용하는 방법에 대한 자세한 내용은 사용 설명서의 정책 및 권한을 참조하십시오. IAM IAM
- IAM정책의 조건을 사용하여 액세스를 추가로 제한 - 정책에 조건을 추가하여 작업 및 리소스에 대한 액세스를 제한할 수 있습니다. 예를 들어, 를 사용하여 모든 요청을 전송하도록 지정하는 정책 조건을 작성할 수 SSL 있습니다. 예를 AWS 서비스들어 특정 작업을 통해 서비스 작업을 사용하는 경우 조건을 사용하여 서비스 작업에 대한 액세스 권한을 부여할 수도 AWS CloudFormation있습니다. 자세한 내용은 IAM사용 설명서의 IAMJSON정책 요소: 조건을 참조하십시오.
- IAMAccess Analyzer를 사용하여 IAM 정책을 검증하여 안전하고 기능적인 권한을 보장합니다. IAM Access Analyzer는 새 정책과 기존 정책을 검증하여 정책이 IAM 정책 언어 (JSON) 및 IAM 모범 사례를 준수하는지 확인합니다. IAMAccess Analyzer는 안전하고 기능적인 정책을 작성하는 데 도움이 되는 100개 이상의 정책 검사와 실행 가능한 권장 사항을 제공합니다. 자세한 내용은 사용 설명서의 IAMAccess Analyzer 정책 검증을 참조하십시오. IAM
- 다단계 인증 필요 (MFA) - 사용자 또는 루트 IAM 사용자가 필요한 시나리오가 있는 경우 보안을 강화하려면 이 기능을 MFA 켜십시오. AWS 계정 API작업 호출 MFA 시기를 요구하려면 정책에 MFA 조건을 추가하세요. 자세한 내용은 IAM사용 설명서의 MFA -보호된 API 액세스 구성을 참조하십시오.

의 모범 사례에 IAM 대한 자세한 내용은 IAM사용 설명서의 보안 모범 사례를 참조하십시오. IAM

AWS 최종 사용자 메시지 푸시 콘솔 사용

AWS 최종 사용자 메시지 푸시 콘솔에 액세스하려면 최소 권한 집합이 있어야 합니다. 이러한 권한을 통해 내 AWS 최종 사용자 메시지 푸시 리소스를 나열하고 해당 리소스에 대한 세부 정보를 볼 수 있어야 AWS 계정합니다. 최소 필수 권한보다 더 제한적인 자격 증명 기반 정책을 만들면 콘솔이 해당 정책에 연결된 엔터티(사용자 또는 역할)에 대해 의도대로 작동하지 않습니다.

AWS CLI 또는 에만 전화를 거는 사용자에게 최소 콘솔 권한을 허용할 필요는 AWS API 없습니다. 대신 수행하려는 작업과 일치하는 API 작업에만 액세스를 허용하세요.

사용자와 역할이 AWS 최종 사용자 메시지 푸시 콘솔을 계속 사용할 수 있도록 하려면 `AWSEndUserMessaging` AWS 관리형 정책도 엔티티에 연결하십시오. 자세한 내용은 사용 설명서의 IAM사용자에게 권한 추가를 참조하십시오.

```
{
 "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": [
    "*"
   ]
```

```
  }
 ]
}
```

사용자가 자신의 고유한 권한을 볼 수 있도록 허용

이 예제에서는 IAM 사용자가 자신의 사용자 ID에 연결된 인라인 및 관리형 정책을 볼 수 있도록 허용하는 정책을 만드는 방법을 보여줍니다. 이 정책에는 콘솔에서 또는 OR를 사용하여 프로그래밍 방식으로 이 작업을 완료할 수 있는 AWS CLI 권한이 포함됩니다. 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": "*"
        }
    ]
}
```

AWS 최종 사용자 메시지 푸시 ID 및 액세스 문제 해결

다음 정보를 사용하면 AWS 최종 사용자 메시지 푸시 및 작업 시 발생할 수 있는 일반적인 문제를 진단하고 해결하는 데 도움이 IAM 됩니다.

주제

- AWS 최종 사용자 메시지 푸시에서 작업을 수행할 권한이 없습니다.
- 저는 IAM을 수행할 권한이 없습니다. PassRole
- 외부 사용자가 내 AWS 최종 사용자 메시지 푸시 AWS 계정 리소스에 액세스할 수 있도록 허용하고 싶습니다.

AWS 최종 사용자 메시지 푸시에서 작업을 수행할 권한이 없습니다.

작업을 수행할 수 있는 권한이 없다는 오류가 수신되면 작업을 수행할 수 있도록 정책을 업데이트해야 합니다.

다음 예제 오류는 mateojackson IAM 사용자가 콘솔을 사용하여 가상 *my-example-widget* 리소스에 대한 세부 정보를 보려고 하지만 가상 권한이 없는 경우 발생합니다. mobiletargeting:*GetWidget*

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

이 경우 mobiletargeting:*GetWidget*작업을 사용하여 *my-example-widget*리소스에 액세스할 수 있도록 mateojackson사용자 정책을 업데이트해야 합니다.

도움이 필요하면 관리자에게 문의하세요. AWS 관리자는 로그인 자격 증명을 제공한 사람입니다.

저는 IAM을 수행할 권한이 없습니다. PassRole

작업을 수행할 권한이 없다는 오류가 발생하는 경우 AWS 최종 사용자 메시지 푸시에 역할을 전달할 수 있도록 정책을 업데이트해야 합니다. iam:PassRole

일부 AWS 서비스 서비스에서는 새 서비스 역할 또는 서비스 연결 역할을 만드는 대신 기존 역할을 해당 서비스에 전달할 수 있습니다. 이렇게 하려면 사용자가 서비스에 역할을 전달할 수 있는 권한을 가지고 있어야 합니다.

다음 예제 오류는 라는 IAM 사용자가 콘솔을 사용하여 AWS 최종 사용자 메시지 푸시에서 작업을 `marymajor` 수행하려고 할 때 발생합니다. 하지만 작업을 수행하려면 서비스 역할이 부여한 권한이 서비스에 있어야 합니다. Mary는 서비스에 역할을 전달할 수 있는 권한을 가지고 있지 않습니다.

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

이 경우, Mary가 `iam:PassRole` 작업을 수행할 수 있도록 Mary의 정책을 업데이트해야 합니다.

도움이 필요하면 AWS 관리자에게 문의하세요. 관리자는 로그인 자격 증명을 제공한 사람입니다.

외부 사용자가 내 AWS 최종 사용자 메시지 푸시 AWS 계정 리소스에 액세스할 수 있도록 허용하고 싶습니다.

다른 계정의 사용자 또는 조직 외부의 사람이 리소스에 액세스할 때 사용할 수 있는 역할을 생성할 수 있습니다. 역할을 수임할 신뢰할 수 있는 사람을 지정할 수 있습니다. 리소스 기반 정책 또는 액세스 제어 목록 (ACLs) 을 지원하는 서비스의 경우 이러한 정책을 사용하여 사용자에게 리소스에 대한 액세스 권한을 부여할 수 있습니다.

자세히 알아보려면 다음을 참조하십시오.

- AWS 최종 사용자 메시지 푸시가 이러한 기능을 지원하는지 알아보려면 을 참조하십시오. AWS 최종 사용자 메시지 푸시의 작동 방식 IAM
- 소유한 리소스에 대한 액세스 권한을 AWS 계정 부여하는 방법을 알아보려면 사용 설명서의 다른 IAM AWS 계정 사용자에게 액세스 권한 제공을 IAM 참조하십시오.
- 제3자에게 리소스에 대한 액세스 권한을 제공하는 방법을 알아보려면 IAM사용 설명서의 제3자가 AWS 계정 소유한 리소스에 대한 액세스 제공을 참조하십시오. AWS 계정
- ID 페더레이션을 통해 액세스를 제공하는 방법을 알아보려면 사용 설명서의 외부 인증된 사용자에게 액세스 제공 (ID 페더레이션) 을 IAM 참조하십시오.
- 계정 간 액세스에 대한 역할 사용과 리소스 기반 정책의 차이점을 알아보려면 사용 설명서의 계정 간 리소스 액세스를 참조하십시오. IAM IAM

AWS 최종 사용자 메시지 푸시에 대한 규정 준수 검증

특정 규정 준수 프로그램의 범위 내에 AWS 서비스 있는지 알아보려면AWS 서비스 규정 준수 프로그램의AWS 서비스 범위별, 규정 참조하여 관심 있는 규정 준수 프로그램을 선택하십시오. 일반 정보는 AWS 규정 준수 프로그램AWS 보증 프로그램 규정AWS 참조하십시오.

를 사용하여 AWS Artifact타사 감사 보고서를 다운로드할 수 있습니다. 자세한 내용은 의 보고서 https://docs.aws.amazon.com/artifact/latest/ug/downloading-documents.html 참조하십시오 AWS Artifact.

사용 시 규정 준수 AWS 서비스 책임은 데이터의 민감도, 회사의 규정 준수 목표, 관련 법률 및 규정에 따라 결정됩니다. AWS 규정 준수에 도움이 되는 다음 리소스를 제공합니다.

- 보안 및 규정 준수 퀵 스타트 가이드 - 이 배포 가이드에서는 아키텍처 고려 사항을 설명하고 보안 및 규정 준수에 AWS 중점을 둔 기본 환경을 배포하기 위한 단계를 제공합니다.
- Amazon Web Services의 HIPAA 보안 및 규정 준수를 위한 설계 — 이 백서에서는 기업이 적합한 애플리케이션을 만드는 AWS HIPAA 데 사용할 수 있는 방법을 설명합니다.

> **Note**
>
> 모든 AWS 서비스 사람이 자격이 있는 것은 아닙니다. HIPAA 자세한 내용은 HIPAA적격 서비스 참조를 참조하십시오.

- AWS 규정AWS 준수 리소스 — 이 워크북 및 가이드 모음은 해당 산업 및 지역에 적용될 수 있습니다.
- AWS 고객 규정 준수 가이드 — 규정 준수의 관점에서 공동 책임 모델을 이해하십시오. 이 가이드에서는 보안을 유지하기 위한 모범 사례를 AWS 서비스 요약하고 여러 프레임워크 (국립 표준 기술 연구소 (NIST), 결제 카드 산업 보안 표준 위원회 (), 국제 표준화 기구 ()) 를 포함한PCI) 전반의 보안 제어에 대한 지침을 매핑합니다. ISO
- AWS Config 개발자 안내서의 규칙을 사용하여 리소스 평가 — 이 AWS Config 서비스는 리소스 구성이 내부 관행, 업계 지침 및 규정을 얼마나 잘 준수하는지 평가합니다.
- AWS Security Hub— 이를 AWS 서비스 통해 내부 AWS보안 상태를 포괄적으로 파악할 수 있습니다. Security Hub는 보안 제어를 사용하여 AWS 리소스를 평가하고 보안 업계 표준 및 모범 사례에 대한 규정 준수를 확인합니다. 지원되는 서비스 및 제어 목록은 Security Hub 제어 참조를 참조하십시오.
- Amazon GuardDuty — 환경에 의심스럽고 악의적인 활동이 있는지 AWS 계정모니터링하여 워크로드, 컨테이너 및 데이터에 대한 잠재적 위협을 AWS 서비스 탐지합니다. GuardDuty 특정 규정 준수 프레임워크에서 요구하는 침입 탐지 요구 사항을 충족하는 PCI DSS 등 다양한 규정 준수 요구 사항을 해결하는 데 도움이 될 수 있습니다.
- AWS Audit Manager— 이를 AWS 서비스 통해 AWS 사용량을 지속적으로 감사하여 위험을 관리하고 규정 및 업계 표준을 준수하는 방법을 단순화할 수 있습니다.

AWS 엔드 유저 메시지 푸시의 탄력성

AWS 글로벌 인프라는 가용 영역을 중심으로 구축됩니다 AWS 리전 . AWS 리전 물리적으로 분리되고 격리된 여러 가용 영역을 제공합니다. 이 가용 영역은 지연 시간이 짧고 처리량이 높으며 중복성이 높은 네트워킹으로 연결됩니다. 가용 영역을 사용하면 중단 없이 영역 간에 자동으로 장애 극복 조치가 이루어지는 애플리케이션 및 데이터베이스를 설계하고 운영할 수 있습니다. 가용 영역은 기존의 단일 또는 다중 데이터 센터 인프라보다 가용성, 내결함성, 확장성이 뛰어납니다.

가용 영역에 대한 AWS 리전 자세한 내용은 글로벌 인프라를 참조하십시오AWS .

AWS 글로벌 인프라 외에도 AWS 최종 사용자 메시지 푸시는 데이터 복원력 및 백업 요구 사항을 지원하는 데 도움이 되는 여러 기능을 제공합니다.

AWS 최종 사용자 메시지 푸시의 인프라 보안

AWS 최종 사용자 메시지 푸시는 관리형 서비스로서 Amazon Web Services: 보안 프로세스 개요 백서에 설명된 AWS 글로벌 네트워크 보안 절차에 따라 보호됩니다.

AWS 게시된 API 호출을 사용하여 네트워크를 통해 AWS 최종 사용자 메시지 푸시에 액세스할 수 있습니다. 클라이언트는 전송 계층 보안 (TLS) 1.2 이상을 지원해야 합니다. 또한 클라이언트는 (Epemeral Diffie-Hellman) 또는 (타원 곡선 Epememeral Diffie-HellmanPFS) 과 같은 완벽한 순방향 기밀 DHE () 을 갖춘 암호 제품군을 지원해야 합니다. ECDHE Java 7 이상의 최신 시스템은 대부분 이러한 모드를 지원합니다.

또한 액세스 키 ID 및 보안 주체와 연결된 보안 액세스 키를 사용하여 요청에 서명해야 합니다. IAM 또는 AWS Security Token Service(AWS STS)를 사용하여 임시 보안 인증을 생성하여 요청에 서명할 수 있습니다.

구성 및 취약성 분석

AWS 최종 사용자 메시지 푸시는 관리형 서비스로서 Amazon Web Services: 보안 프로세스 개요 백서에 설명된 AWS 글로벌 네트워크 보안 절차에 따라 보호됩니다. 즉, 계정 및 리소스의 기본 인프라를 강화, 패치, 업데이트 및 유지 관리하기 위한 기본 보안 작업과 절차를 AWS 관리하고 수행합니다. 적합한 제3자가 이 절차를 검토하고 인증하였습니다.

보안 모범 사례

AWS Identity and Access Management (IAM) 계정을 사용하여 API 작업, 특히 리소스를 생성, 수정 또는 삭제하는 작업에 대한 액세스를 제어할 수 있습니다. 의 API 경우 이러한 리소스에는 프로젝트, 캠페인 및 여정이 포함됩니다.

- 본인을 포함하여 리소스를 관리하는 각 개인에 대해 개별 사용자를 생성합니다. AWS 루트 자격 증명을 사용하여 리소스를 관리하지 마세요.
- 각 사용자에게 각자의 임무를 수행하는 데 필요한 최소 권한 집합을 부여합니다.
- IAM그룹을 사용하면 여러 사용자의 권한을 효과적으로 관리할 수 있습니다.
- IAM 자격 증명을 정기적으로 교체합니다.

보안에 대한 자세한 내용은 AWS 최종 사용자 메시지 푸시의 보안을 참조하세요. 에 대한 IAM 자세한 내용은 AWS ID 및 Access Management를 참조하십시오. 모범 사례에 대한 자세한 내용은 IAM 모범 사례를 참조하십시오IAM.

AWS 최종 사용자 메시지 푸시 모니터링

모니터링은 AWS 최종 사용자 메시지 푸시 및 기타 AWS 솔루션의 안정성, 가용성 및 성능을 유지하는 데 있어 중요한 부분입니다. AWS AWS 최종 사용자 메시지 푸시를 감시하고, 문제 발생 시 보고하고, 적절한 경우 자동 조치를 취할 수 있는 다음과 같은 모니터링 도구를 제공합니다.

- Amazon은 실행 중인 AWS 리소스와 애플리케이션을 AWS 실시간으로 CloudWatch 모니터링합니다. 지표를 수집 및 추적하고, 사용자 지정 대시보드를 생성할 수 있으며, 지정된 지표가 지정한 임계값에 도달하면 사용자에게 알리거나 조치를 취하도록 경보를 설정할 수 있습니다. 예를 들어 Amazon EC2 인스턴스의 CPU 사용량 또는 기타 지표를 CloudWatch 추적하고 필요할 때 새 인스턴스를 자동으로 시작할 수 있습니다. 자세한 내용은 Amazon CloudWatch 사용 설명서를 참조하십시오.
- Amazon CloudWatch Logs를 사용하면 Amazon EC2 인스턴스 및 기타 소스에서 로그 파일을 모니터링 CloudTrail, 저장 및 액세스할 수 있습니다. CloudWatch 로그는 로그 파일의 정보를 모니터링하고 특정 임계값이 충족되면 알려줄 수 있습니다. 또한 매우 내구력 있는 스토리지에 로그 데이터를 저장할 수 있습니다. 자세한 내용은 Amazon CloudWatch Logs 사용 설명서를 참조하십시오.
- Amazon을 사용하면 AWS 서비스를 자동화하고 애플리케이션 가용성 문제 또는 리소스 변경과 같은 시스템 이벤트에 자동으로 대응할 EventBridge 수 있습니다. AWS 서비스에서 발생하는 이벤트는 거의 EventBridge 실시간으로 전송됩니다. 원하는 이벤트만 표시하도록 간단한 규칙을 작성한 후 규칙과 일치하는 이벤트 발생 시 실행할 자동화 작업을 지정할 수 있습니다. 자세한 내용은 Amazon EventBridge 사용 설명서를 참조하십시오.
- AWS CloudTrail사용자 계정에서 또는 AWS 계정을 대신하여 이루어진 API 호출 및 관련 이벤트를 캡처하고 지정한 Amazon S3 버킷으로 로그 파일을 전송합니다. 어떤 사용자와 계정이 전화를 걸었는지 AWS, 어떤 소스 IP 주소에서 호출이 이루어졌는지, 언제 호출이 발생했는지 식별할 수 있습니다. 자세한 내용은 AWS CloudTrail 사용 설명서를 참조하십시오.

Amazon을 통한 AWS 최종 사용자 메시지 푸시 모니터링 CloudWatch

원시 데이터를 수집하여 읽을 수 있는 거의 실시간 지표로 처리하는 를 사용하여 AWS CloudWatch 최종 사용자 메시지 푸시를 모니터링할 수 있습니다. 이러한 통계는 15개월간 보관되므로 기록 정보에 액세스하고 웹 애플리케이션 또는 서비스가 어떻게 실행되고 있는지 전체적으로 더 잘 파악할 수 있습니다. 특정 임계값을 주시하다가 해당 임계값이 충족될 때 알림을 전송하거나 조치를 취하도록 경보를 설정할 수도 있습니다. 자세한 내용은 Amazon CloudWatch 사용 설명서를 참조하십시오.

지표 및 측정기준 목록은 아마존 Pinpoint 사용 설명서의 Amazon Pinpoint 모니터링을 참조하십시오. CloudWatch

를 사용하여 AWS 최종 사용자 메시지 푸시 API 콜 로깅 AWS CloudTrail

AWS 최종 사용자 메시지 푸시는 AWS 최종 사용자 메시지 푸시에서 사용자, 역할 또는 AWS 서비스가 수행한 작업의 기록을 제공하는 서비스와 통합되어 있습니다. AWS CloudTrail CloudTrail AWS 최종 사용자 메시지 푸시에 대한 모든 API 호출을 이벤트로 캡처합니다. 캡처된 호출에는 AWS 최종 사용자 메시지 푸시 콘솔의 통화와 AWS 최종 사용자 메시지 푸시 API 작업에 대한 코드 호출이 포함됩니다. 트레일을 생성하면 AWS 최종 사용자 메시지 푸시에 대한 CloudTrail 이벤트를 포함하여 Amazon S3 버킷에 이벤트를 지속적으로 전송할 수 있습니다. 트레일을 구성하지 않아도 CloudTrail 콘솔의 이벤트 기록에서 가장 최근 이벤트를 계속 볼 수 있습니다. 에서 수집한 CloudTrail 정보를 사용하여 AWS 최종 사용자 메시지 푸시에 대한 요청, 요청이 이루어진 IP 주소, 요청한 사람, 요청 시기 및 추가 세부 정보를 확인할 수 있습니다.

자세한 CloudTrail 내용은 AWS CloudTrail 사용 설명서를 참조하십시오.

AWS 최종 사용자 메시지 푸시 정보 CloudTrail

CloudTrail 계정을 만들 AWS 계정 때 활성화됩니다. AWS 최종 사용자 메시지 푸시에서 활동이 발생하면 해당 활동이 CloudTrail 이벤트 기록의 다른 AWS 서비스 이벤트와 함께 이벤트에 기록됩니다. 에서 최근 이벤트를 보고, 검색하고, 다운로드할 수 있습니다 AWS 계정. 자세한 내용은 이벤트 기록으로 CloudTrail 이벤트 보기를 참조하십시오.

AWS 최종 사용자 메시지 푸시 이벤트를 AWS 계정포함하여 내 이벤트의 진행 중인 기록을 보려면 트레일을 생성하십시오. 트레일을 사용하면 CloudTrail Amazon S3 버킷으로 로그 파일을 전송할 수 있습니다. 콘솔에서 추적을 생성하면 기본적으로 모든 AWS 리전에 추적이 적용됩니다. 트레일은 AWS 파티션에 있는 모든 지역의 이벤트를 기록하고 지정한 Amazon S3 버킷으로 로그 파일을 전송합니다. 또한 CloudTrail 로그에서 수집된 이벤트 데이터를 추가로 분석하고 이에 따라 조치를 취하도록 다른 AWS 서비스를 구성할 수 있습니다. 자세한 내용은 다음 자료를 참조하십시오.

- 추적 생성 개요
- CloudTrail 지원되는 서비스 및 통합
- 다음에 대한 Amazon SNS 알림 구성 CloudTrail
- 여러 지역에서 CloudTrail 로그 파일 수신 및 여러 계정으로부터 CloudTrail 로그 파일 수신

모든 AWS 최종 사용자 메시지 푸시 동작은 최종 사용자 메시지 푸시 참조에 의해 CloudTrail 기록되며 AWS 최종 사용자 메시지 푸시 API 참조에 문서화되어 있습니다. 예를 들어`GetAdmChannel`, 에 대한 호출 `UpdateApnsChannel` 및 `GetApnsVoipChannel` 작업은 CloudTrail 로그 파일에 항목을 생성합니다.

모든 이벤트 및 로그 항목에는 요청을 생성한 사용자에 대한 정보가 들어 있습니다. 보안 인증 정보를 이용하면 다음을 쉽게 판단할 수 있습니다.

- 요청이 루트 또는 AWS Identity and Access Management (IAM) 사용자 자격 증명으로 이루어졌는지 여부
- 역할 또는 페더레이션 사용자에 대한 임시 보안 보안 인증을 사용하여 요청이 생성되었는지 여부.
- 다른 AWS 서비스에서 요청했는지 여부.

자세한 내용은 CloudTrail userIdentity요소를 참조하십시오.

AWS 최종 사용자 메시지 푸시 로그 파일 항목 이해

트레일은 지정한 Amazon S3 버킷에 이벤트를 로그 파일로 전송할 수 있는 구성입니다. CloudTrail 로그 파일에는 하나 이상의 로그 항목이 포함되어 있습니다. 이벤트는 모든 소스의 단일 요청을 나타내며 요청된 작업, 작업 날짜 및 시간, 요청 매개 변수 등에 대한 정보를 포함합니다. CloudTrail 로그 파일은 공개 API 호출의 정렬된 스택 트레이스가 아니므로 특정 순서로 표시되지 않습니다.

인터페이스 엔드포인트 (AWS PrivateLink) 를 사용하여 AWS 최종 사용자 메시지 푸시에 액세스

를 AWS PrivateLink 사용하여 최종 사용자 메시지 VPC 푸시와 AWS 최종 사용자 메시지 푸시 간에 비공개 연결을 만들 수 있습니다. 인터넷 게이트웨이, NAT 장치VPC, VPN 연결 또는 AWS Direct Connect 연결을 사용하지 않고도 마치 집에 있는 것처럼 AWS 최종 사용자 메시지 푸시에 액세스할 수 있습니다. 의 인스턴스는 AWS 최종 사용자 메시지 푸시에 액세스하는 데 퍼블릭 IP 주소가 VPC 필요하지 않습니다.

AWS PrivateLink에서 제공되는 인터페이스 엔드포인트를 생성하여 이 프라이빗 연결을 설정합니다. 인터페이스 엔드포인트에 대해 사용 설정하는 각 서브넷에서 엔드포인트 네트워크 인터페이스를 생성합니다. 이는 AWS 최종 사용자 메시지 푸시로 향하는 트래픽의 진입점 역할을 하는 요청자 관리형 네트워크 인터페이스입니다.

자세한 내용은 가이드의 액세스를 참조하십시오 AWS 서비스 . AWS PrivateLinkAWS PrivateLink

AWS 최종 사용자 메시지 푸시에 대한 고려사항

AWS 최종 사용자 메시지 푸시를 위한 인터페이스 엔드포인트를 설정하기 전에 AWS PrivateLink 가이드의 고려 사항을 검토하십시오.

AWS 최종 사용자 메시지 푸시는 인터페이스 엔드포인트를 통해 모든 API 작업에 대한 호출을 지원합니다.

VPC엔드포인트 정책은 AWS 최종 사용자 메시지 푸시에 지원되지 않습니다. 기본적으로 인터페이스 엔드포인트를 통해 AWS 최종 사용자 메시지 푸시에 대한 전체 액세스가 허용됩니다. 또는 보안 그룹을 엔드포인트 네트워크 인터페이스와 연결하여 인터페이스 엔드포인트를 통해 AWS 최종 사용자 메시지 푸시로 전달되는 트래픽을 제어할 수 있습니다.

AWS 최종 사용자 메시지 푸시를 위한 인터페이스 엔드포인트를 생성합니다.

Amazon VPC 콘솔 또는 AWS Command Line Interface (AWS CLI) 를 사용하여 AWS 최종 사용자 메시지 푸시에 대한 인터페이스 엔드포인트를 생성할 수 있습니다. 자세한 내용은 AWS PrivateLink 설명서의 인터페이스 엔드포인트 생성을 참조하십시오.

다음 서비스 이름을 사용하여 AWS 최종 사용자 메시지 푸시용 인터페이스 엔드포인트를 생성합니다.

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

인터페이스 엔드포인트에 DNS 대해 비공개를 활성화하면 기본 지역 DNS 이름을 사용하여 AWS 최종 사용자 메시지 푸시에 API 요청을 보낼 수 있습니다. 예: `com.amazonaws.us-east-1.pinpoint`.

엔드포인트의 엔드포인트 정책 생성

엔드포인트 정책은 인터페이스 엔드포인트에 연결할 수 있는 IAM 리소스입니다. 기본 엔드포인트 정책은 인터페이스 엔드포인트를 통해 AWS 최종 사용자 메시지 푸시에 대한 전체 액세스를 허용합니다. 에서 AWS 최종 사용자 메시지 푸시에 허용되는 액세스를 제어하려면 인터페이스 엔드포인트에 사용자 지정 엔드포인트 정책을 연결하십시오. VPC

엔드포인트 정책은 다음 정보를 지정합니다.

- 작업을 수행할 수 있는 주체 (AWS 계정, IAM 사용자, IAM 역할)
- 수행할 수 있는 작업.
- 작업을 수행할 수 있는 리소스.

자세한 내용은AWS PrivateLink 가이드의 엔드포인트 정책을 사용하여 서비스에 대한 액세스 제어를 참조하세요.

예: AWS 최종 사용자 메시지 푸시 작업에 대한 VPC 엔드포인트 정책

다음은 사용자 지정 엔드포인트 정책의 예입니다. 이 정책을 인터페이스 엔드포인트에 연결하면 모든 리소스의 모든 보안 주체에 대해 나열된 AWS 최종 사용자 메시지 푸시 작업에 대한 액세스 권한이 부여됩니다.

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

```
}
```

AWS 최종 사용자 메시지 푸시 할당량

Your AWS 계정 서비스에는 각 서비스에 대한 기본 할당량 (이전에는 한도라고 함) 이 있습니다. AWS 다르게 표시되지 않는 한 리전별로 각 할당량이 적용됩니다. 일부 할당량에 대한 증가를 요청할 수 있으며 다른 할당량은 늘릴 수 없습니다.

AWS 최종 사용자 메시지 푸시의 할당량을 보려면 Service Quotas 콘솔을 여십시오. 탐색 창에서 AWS 서비스를 선택하고 Amazon Pinpoint를 선택합니다.

AWS계정에는 AWS 최종 사용자 메시지 푸시와 관련된 다음과 같은 할당량이 있습니다.

Resource	기본 할당량	증가 가능 여부
단일 캠페인에서 초당 전송 가능한 최대 푸시 알림 수	초당 25,000개	예, Service Quotas 콘솔을 사용하십시오.
Amazon 디바이스 메시징 (ADM) 메시지 페이로드 크기	메시지당 6KB	아니요
Apple 푸시 알림 서비스 (APNs) 메시지 페이로드 크기	메시지당 4KB	아니요
APNs샌드박스 메시지 페이로드 크기	메시지당 4KB	아니요
Baidu Cloud Push 메시지 페이로드 크기	메시지당 4KB	아니요
Firebase 클라우드 메시징 (FCM) 메시지 페이로드 크기	메시지당 4KB	아니요

AWS 최종 사용자 메시지 푸시 사용 설명서의 문서 기록

다음 표에는 AWS 최종 사용자 메시지 푸시에 대한 설명서 릴리스가 설명되어 있습니다.

변경 사항	설명	날짜
최초 릴리스	AWS 최종 사용자 메시지 푸시 사용 설명서의 초기 릴리스	2024년 7월 24일

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