



Panduan Developer

AWS Pusat Partner



AWS Pusat Partner: Panduan Developer

Table of Contents

.....	vi
Selamat datang	1
Bekerja dengan AWS SDK	1
Pengaturan dan otentikasi	2
Menghubungkan Anda Akun AWS ke Partner Central	2
Mengatur IAM	2
Mengautentikasi panggilan API	3
Menandatangani panggilan Anda dengan agen pengguna khusus	4
Mitra agen Pusat MCP Server	7
Gambaran umum	7
Kemampuan agen	7
Manfaat utama	8
Contoh penggunaan	8
Penciptaan peluang	9
Kloning peluang	9
Wawasan pipa	9
Ringkasan peluang	10
Generasi permainan penjualan	10
Pembuatan profil pelanggan	10
Rekomendasi solusi	10
Rekomendasi pendanaan	11
Rekomendasi langkah selanjutnya	11
Kemajuan peluang	11
Pengalaman agen untuk orientasi mitra	12
Otomatisasi profil mitra	12
Panduan pengaturan penjual	12
Panduan kepatuhan PRM	13
Memulai	13
Memulai	13
Prasyarat	13
Langkah 1: Siapkan izin IAM	14
Langkah 2: Hubungkan klien MCP Anda	21
Menandatangani panggilan Anda dengan header MCP	22
Langkah 3: Verifikasi pengaturan Anda	24

Langkah 4: Jalankan tugas pertama Anda	24
Pertimbangan keamanan	27
Langkah selanjutnya	27
Referensi Konfigurasi	27
Titik akhir	27
Izin IAM	28
Lingkungan katalog	33
Manajemen sesi	33
Unggahan file	33
Kode eror	34
Batas tarif	35
Jenis acara Streaming (SSE)	35
Langkah selanjutnya	36
Alat Referensi	36
Ikhtisar alat	37
SendMessage	37
GetSession	45
Penanganan kesalahan	47
Tentang API	49
Menggunakan API	49
Menggunakan API Akun	49
Menggunakan API Penjualan	63
Menggunakan API Manfaat	119
Menggunakan API Saluran	133
Menggunakan API Pengukuran Pendapatan	135
Wilayah yang Didukung	147
Wilayah untuk API Akun	147
Wilayah untuk API Penjualan	147
Wilayah untuk API Manfaat	148
Wilayah untuk API Saluran	148
Wilayah untuk API Pengukuran Pendapatan	149
Izin	149
Akses untuk API Akun	150
Akses untuk API Penjualan	153
Akses untuk API Manfaat	159
Akses untuk API Saluran	162

Akses untuk API Pengukuran Pendapatan	170
Kuota	179
Kuota untuk API Akun	180
Kuota untuk API Penjualan	183
Kuota untuk API Manfaat	185
Kuota untuk API Saluran	187
Kuota untuk API Pengukuran Pendapatan	192
Pencatatan log	194
Logging API Akun	194
Pencatatan API Penjualan	198
Pencatatan API Manfaat	199
Logging API Saluran	201
Mencatat API Pengukuran Pendapatan	204
Notifikasi	207
Pemberitahuan untuk AWS API Rekening Sentral Mitra	207
Pemberitahuan untuk AWS API Penjualan Sentral Mitra	219
Pemberitahuan untuk AWS API Manfaat Sentral Mitra	234
Pemberitahuan untuk AWS API Saluran Pusat Mitra	247
Pengujian di kotak pasir	258
Akses ke lingkungan kotak pasir	258
Detail penting tentang lingkungan kotak pasir	258
Menguji di kotak pasir untuk API Akun	259
Menguji di kotak pasir untuk API Penjualan	262
Menguji di kotak pasir untuk API Manfaat	266
Menguji di kotak pasir untuk API Saluran	271
Menguji di kotak pasir untuk API Pengukuran Pendapatan	275
Contoh kode	280
Hal-hal mendasar	281
Tindakan	281
Skenario	347
Perbarui entitas terkait dari suatu peluang	348
Catatan rilis	354
Riwayat dokumen	386

Referensi AWS Partner Central API direstrukturisasi. Untuk informasi selengkapnya tentang operasi API yang didukung, lihat [Referensi AWS Partner Central API](#).

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.

Selamat datang di AWS Panduan Pengembang Pusat Mitra

Panduan pengembang ini menjelaskan bagaimana AWS Partner dapat menggunakan API layanan untuk mengintegrasikan dan mengelola aktivitas build, market, sell, dan grow AWS.

Topik

- [Penggunaan AWS Mitra Central API dengan AWS SDK](#)
- [Pengaturan dan otentikasi](#)

Penggunaan AWS Mitra Central API dengan AWS SDK

AWS kit pengembangan perangkat lunak (SDK) tersedia untuk banyak bahasa pemrograman populer. Setiap SDK menyediakan API, contoh kode, dan dokumentasi yang memudahkan developer untuk membangun aplikasi dalam bahasa pilihan mereka.

Dokumentasi SDK	Contoh kode
AWS SDK untuk C++	AWS SDK untuk C++ contoh kode
AWS CLI	AWS CLI contoh kode
AWS SDK untuk Go	AWS SDK untuk Go contoh kode
AWS SDK untuk Java	AWS SDK untuk Java contoh kode
AWS SDK untuk JavaScript	AWS SDK untuk JavaScript contoh kode
AWS SDK untuk Kotlin	AWS SDK untuk Kotlin contoh kode
AWS SDK untuk .NET	AWS SDK untuk .NET contoh kode
AWS SDK untuk PHP	AWS SDK untuk PHP contoh kode
Alat AWS untuk PowerShell	Alat AWS untuk PowerShell contoh kode
AWS SDK untuk Python (Boto3)	AWS SDK untuk Python (Boto3) contoh kode
AWS SDK untuk Ruby	AWS SDK untuk Ruby contoh kode

Dokumentasi SDK	Contoh kode
AWS SDK for Rust	AWS SDK for Rust contoh kode
AWS SDK for SAP ABAP	AWS SDK for SAP ABAP contoh kode
AWS SDK for Swift	AWS SDK for Swift contoh kode

Ketersediaan contoh

Tidak dapat menemukan apa yang Anda butuhkan? Minta contoh kode menggunakan tautan Berikan umpan balik di bagian bawah halaman ini.

Pengaturan dan otentikasi

Menyiapkan dan mengautentikasi dengan API Pusat AWS Mitra melibatkan tiga langkah. Berikut ikhtisar prosesnya:

1. Tautkan akun AWS Marketplace Penjual Anda ke Partner Central.
2. Siapkan izin menggunakan IAM.
3. Mengautentikasi panggilan API Anda menggunakan Signature Version 4 (SigV4).

Menghubungkan Anda Akun AWS ke Partner Central

Menautkan Anda Akun AWS ke Partner Central adalah prasyarat untuk menggunakan API. Untuk informasi selengkapnya, lihat [Menautkan akun Pusat AWS Mitra dengan akun AWS Marketplace penjual](#). Anda harus masuk ke Partner Central dengan akun yang memiliki izin alliance-lead atau cloud-administrator, navigasikan ke bagian tersebut **Account Linking**, dan ikuti petunjuknya.

Mengatur IAM

Untuk menggunakan API Pusat AWS Mitra, Anda memerlukan peran AWS Identity and Access Management(IAM) atau pengguna IAM untuk mulai melakukan panggilan. Untuk informasi selengkapnya, lihat [Kapan saya menggunakan IAM?](#) . Ikuti langkah-langkah untuk [Membuat peran IAM](#) dan [Membuat pengguna IAM](#) di Akun AWS panduan Anda untuk ini. Anda harus membuat IAM

ini Role/User di akun Central-linked AWS Marketplace Penjual Mitra Anda. role/user Pembuatan IAM tidak menimbulkan biaya apa pun.

1. Buat IAM Role/User

Masuk ke Konsol Manajemen AWS, navigasikan ke layanan IAM, dan ikuti langkah-langkah untuk membuat peran IAM atau pengguna IAM.

2. Tetapkan Kebijakan:

Lampirkan kebijakan terkelola atau buat kebijakan khusus sesuai kebutuhan. Untuk mengubah atau memperluas izin, terapkan kebijakan tambahan ke Peran IAM alih-alih menyalin dan menggabungkan konten dari `AWSPartnerCentralOpportunityManagement` dengan izin lainnya. Hindari menduplikasi kebijakan terkelola, karena hal itu akan mencegah Anda mendapatkan akses ke fitur baru secara otomatis saat dirilis, dan Anda harus memperbarui kebijakan secara manual di masa mendatang. Untuk detail selengkapnya tentang kebijakan akses, lihat [dokumentasi Kontrol Akses](#).

Mengelola AWS Marketplace penawaran

Untuk mengelola AWS Marketplace penawaran dan menautkannya ke peluang, mitra harus memberikan izin peran IAM untuk mengakses API Katalog. Pastikan role/user memiliki izin, seperti `aws-marketplace:ListEntities` dan `aws-marketplace:SearchAgreements`.

Mengautentikasi panggilan API

AWS Partner Central API menggunakan Signature Version 4 (SigV4) untuk otentikasi. Berikut cara mengimplementasikannya:

Menggunakan AWS SDK

AWS SDK secara otomatis menangani penandatanganan permintaan. Berikan AWS kredensial Anda, dan SDK melakukan sisanya.

1. Untuk Java, lihat [Menyediakan kredensial sementara ke AWS SDK](#) for Java.
2. [Untuk Python \(Boto3\)](#), lihat [Kredensial](#).
3. Untuk JavaScript (Node.js), lihat [Menyetel kredensial di Node.js](#)
4. Untuk .NET, lihat [Resolusi kredensial dan profil](#).
5. Untuk bahasa pemrograman lain dan contoh lainnya, lihat [Tools to Build on AWS](#).

Otentikasi tanpa menggunakan AWS SDK

Jika AWS SDK tidak tersedia untuk bahasa pemrograman pilihan Anda, otentikasi melibatkan pembuatan permintaan kanonik secara manual, menandatangani permintaan, dan menangani token sesi. AWS menawarkan panduan komprehensif untuk [menggunakan penandatanganan SiGv4](#). Namun, harap dicatat bahwa menggunakan AWS SDK direkomendasikan karena penandatanganan permintaan manual meningkatkan kompleksitas dan memerlukan pengelolaan token keamanan yang cermat.

Setiap permintaan untuk protokol AWSJson1_0 HARUS dikirim ke URL root (/) menggunakan metode HTTP "POST" menggunakan header berikut.

Header yang Diperlukan

Setiap permintaan API untuk protokol AWSJson1_0 HARUS dikirim ke URL root (/) menggunakan metode HTTP "POST" menggunakan header berikut.

Header	Diperlukan	Deskripsi
Content-Type	true	Header ini memiliki nilai statis <code>application/x-amz-json-1.0</code> .
Content-Length	true	Content-Length Header standar ditentukan oleh RFC 9110 #section -8.6 .
X-Amz-Target	benar untuk permintaan	Misalnya, nilai untuk <code>CreatePartner</code> pengoperasian layanan <code>PartnerCentralAccount</code> adalah <code>PartnerCentralAccount.CreatePartner</code> .

Menandatangani panggilan Anda dengan agen pengguna khusus

Saat membuat permintaan API ke AWS Partner Central, sebaiknya `X-Amzn-User-Agent` sertakan header untuk membantu AWS mengidentifikasi sumber aplikasi klien, memantau penggunaan, dan kinerja audit. AWS menggunakan header ini untuk membedakan jenis aplikasi klien yang melakukan panggilan dan untuk mengumpulkan wawasan tentang tingkat keberhasilan implementasi klien yang berbeda.

Header user-agent kustom

Nama header: X-Amzn-User-Agent

Tujuan: Membedakan jenis klien yang membuat permintaan API, mengkategorikan sumber interaksi.

Format: {Integrator}|{Product}

Nilai Contoh: AWS|AWS Partner CRM Connector

Menyertakan header ini di setiap permintaan memungkinkan AWS untuk menganalisis pola permintaan, melacak integrasi, dan meningkatkan pengalaman API untuk sistem klien yang berbeda.

Menggunakan header khusus di SDK

Untuk menyertakan X-Amzn-User-Agent header dalam panggilan SDK, Anda dapat mengubah perilaku permintaan klien sebelum melakukan panggilan API. Di bawah ini adalah contoh API penjualan menggunakan AWS SDK for Python (Boto3):

```
import boto3

# Define service and endpoint details
service_name = "partnercentral-selling"
endpoint_url = "https://partnercentral-selling.us-east-1.api.aws"

# Create a boto3 client for Partner Central
partner_central_client = boto3.client(
    service_name=service_name,
    region='us-east-1',
    endpoint_url=endpoint_url
)

# Function to add the custom User-Agent header

def add_version_header(params, **kwargs):
    params["headers"]['X-Amzn-User-Agent'] = 'AWS|AWS Partner CRM Connector'

# Register the event to modify the request before the call is made
partner_central_client.meta.events.register(
    f'before-call.{service_name}.*', add_version_header
)
```

```
# Now, whenever an API call is made using this client, the custom User-Agent header  
will be included
```

Contoh ini menunjukkan cara mendaftarkan peristiwa di Boto3 SDK untuk menambahkan X-Amzn-User-Agent header secara otomatis ke setiap permintaan API. Pendekatan yang sama dapat diterapkan pada AWS SDK lain dengan memodifikasi mekanisme intersepsi permintaan masing-masing.

 Note

Format X-Amzn-User-Agent header telah disederhanakan. Kami merekomendasikan untuk menggunakan format baru. Format sebelumnya (CompanyName | ProductName | CRMName | ProductVersion) tetap didukung untuk kompatibilitas mundur. Integrasi yang ada akan terus bekerja tanpa modifikasi.

Mitra agen Pusat MCP Server

Agen Pusat Mitra MCP Server menyediakan alat Pusat Mitra melalui Model Context Protocol (MCP), memungkinkan agen dan alat AI Anda untuk menemukan dan berinteraksi dengan manajemen peluang, wawasan pelanggan, dan program pendanaan melalui bahasa alami.

Server menangani otentikasi, manajemen sesi, dan persetujuan manusia-in-the-loop untuk operasi penulisan yang mempertahankan kontrol sambil merampingkan alur kerja.

Gambaran umum

Agen Pusat Mitra MCP Server adalah layanan yang dikelola sepenuhnya dan AWS dihosting yang menghubungkan klien MCP-compatible AI ke agen Pusat Mitra. Ini menggunakan JSON-RPC 2.0 melalui HTTPS dengan otentikasi SiGv4 dan mendukung Server-Sent Acara (SSE) untuk respons streaming waktu nyata.

Server mendukung percakapan multi-putaran, lampiran file untuk analisis dokumen, dan alur kerja persetujuan bawaan yang memerlukan persetujuan eksplisit Anda sebelum operasi penulisan dijalankan.

Kemampuan agen

- Wawasan saluran pipa — Dapatkan intelijen percakapan tentang saluran penjualan Anda, termasuk peluang berisiko, perkembangan panggung, dan analisis yang hilang secara tertutup
- Pembuatan peluang — Ciptakan peluang baru melalui percakapan bahasa alami, dengan mengunggah catatan rapat, proposal, atau transkrip panggilan, atau dengan mengkloning peluang yang ada
- Ringkasan peluang — Hasilkan ringkasan ringkas dan sekilas dari setiap kesepakatan yang mencakup tahap, pengeluaran, tanggal penutupan, dan banyak lagi
- Generasi permainan penjualan - Bangun strategi penjualan yang disesuaikan dengan menggabungkan detail peluang, konteks industri, dan rekomendasi AWS solusi
- Pembuatan profil pelanggan — Menghasilkan profil perusahaan menggunakan informasi yang tersedia untuk umum yang mencakup industri, model bisnis, geografi, dan perkembangan terkini
- Rekomendasi solusi — solusi terdaftar Cross-reference Anda terhadap persyaratan peluang untuk menemukan kecocokan terbaik

- Rekomendasi pendanaan — Mengevaluasi peluang terhadap program AWS pendanaan yang tersedia, memperkirakan jumlah, dan membuat permintaan dana
- Rekomendasi langkah berikutnya - Dapatkan rencana aksi yang diprioritaskan berdasarkan standar AWS penjualan bersama dan panduan perkembangan tahap
- Kemajuan peluang — Unggah dokumen pendukung, ekstrak data yang relevan, dan peluang kemajuan melalui tahapan pipeline
- AI-assisted daftar produk — Hasilkan daftar AWS Marketplace produk berkualitas tinggi dari aset digital Anda yang ada, skor kekuatan daftar berdasarkan AWS Marketplace standar, dan terima rekomendasi tingkat lapangan untuk meningkatkan kemampuan ditemukan. Untuk informasi selengkapnya, lihat [daftar AI-assisted produk](#) di Panduan AWS Marketplace Penjual.

Manfaat utama

- Akses percakapan ke Pusat Mitra - Ajukan pertanyaan dalam bahasa alami alih-alih menavigasi alur kerja konsol yang kompleks
- Lebih banyak waktu menjual — Ciptakan peluang melalui percakapan singkat alih-alih mengisi formulir multi-langkah, yang mengurangi entri data dan memungkinkan tim penjualan mitra menghabiskan lebih banyak waktu untuk menjual, dengan agen merekomendasikan perbaikan sehingga mitra mengirimkan peluang berkualitas lebih tinggi dan meningkatkan kebersihan pipa
- Human-in-the-loop keamanan - Semua operasi tulis memerlukan persetujuan eksplisit Anda sebelum eksekusi
- Multi-turn percakapan — Perbaiki kueri Anda dalam sesi tanpa mengulangi konteks
- Analisis file - Lampirkan dokumen (PDF, DOCX, XLSX, CSV, gambar) untuk agen untuk dianalisis bersama pertanyaan Anda
- Manajemen akses terpusat - Kontrol akses melalui kebijakan IAM dengan izin halus
- Pengujian kotak pasir - Uji alur kerja di lingkungan kotak pasir yang terisolasi sebelum menyentuh data produksi
- Respons streaming - Dapatkan umpan balik melalui SSE saat agen memproses permintaan Anda

Contoh penggunaan

Semua AI-generated wawasan menyertakan ID Sesi untuk ketertelusuran. Data diisolasi ke peluang mitra yang masuk sendiri. Semua konten memiliki label pengungkapan yang jelas dan diatur oleh Kebijakan [AI yang AWS Bertanggung Jawab](#).

Penciptaan peluang

Agen menciptakan peluang baru dari deskripsi bahasa alami atau dokumen yang diunggah (PDF, DOCX, XLSX, TXT). Ini mengekstrak nama pelanggan, ruang lingkup proyek, tanggal penutupan yang diharapkan, dan bidang wajib lainnya, memperkaya detail pelanggan dari data web yang tersedia untuk umum, memvalidasi draf terhadap persyaratan kesiapan AWS, dan menciptakan peluang melalui Partner Central Selling API setelah Anda menyetujui.

- “Ciptakan peluang untuk Acme Corp — mereka ingin memigrasikan gudang data mereka ke Redshift, menargetkan tanggal penutupan akhir Q3, perkiraan pengeluaran AWS bulanan \$40K”
- “Ini catatan panggilan saya dari pertemuan kemarin dengan GlobalTech - buat peluang dari transkrip ini”
- “Gunakan PDF proposal ini untuk menciptakan peluang bagi migrasi SAP pelanggan”

Kloning peluang

Agen menciptakan peluang baru dari yang sudah ada, menggabungkan detail pelanggan atau proyek baru yang Anda berikan, dan memvalidasi bahwa setidaknya satu bidang pembeda telah berubah sebelum pengiriman sehingga duplikat tidak ditolak oleh tinjauan AWS.

- “Peluang klon O1234567890 untuk pelanggan baru - Globex, beban kerja yang sama, tanggal penutupan target akhir Q4”
- “Ciptakan peluang seperti O9876543210 tetapi untuk lingkungan produksi pelanggan, bukan kotak pasir”

Wawasan pipa

Dapatkan intelijen percakapan tentang saluran penjualan Anda. Agen menganalisis perkembangan tahap, tenggat waktu, kesepakatan yang macet, dan pola tertutup yang hilang untuk memunculkan apa yang paling penting.

- “Peluang apa yang perlu saya perhatikan minggu ini?”
- “Berapa banyak peluang yang akan ditutup bulan depan?”
- “Apa alasan utama kami kehilangan peluang dalam 6 bulan terakhir?”

Ringkasan peluang

Agen mensintesis nama perusahaan, industri, panggung, AWS pengeluaran bulanan yang diharapkan, tanggal penutupan target, dan detail kunci lainnya ke dalam ringkasan singkat — tidak perlu memindai bidang formulir individual.

- “Beri saya ringkasan kesempatan O1234567890”
- “Apa status saat ini dan detail kunci untuk kesepakatan Acme Corp?”

Generasi permainan penjualan

Agen membangun strategi penjualan yang disesuaikan dengan menggabungkan detail peluang, konteks industri pelanggan, dan rekomendasi AWS solusi yang relevan ke dalam permainan penjualan yang siap digunakan.

- “Hasilkan permainan penjualan untuk peluang O1234567890”
- “Apa pendekatan terbaik untuk menjual migrasi cloud ke pelanggan layanan keuangan ini?”
- “Bangun saya strategi penjualan untuk kesepakatan analitik GlobalTech data”

Pembuatan profil pelanggan

Agen menghasilkan profil perusahaan menggunakan informasi yang tersedia untuk umum - meliputi klasifikasi industri, model bisnis (B2B/B2C/hybrid), keberadaan geografis, ukuran perusahaan, fokus pasar, dan perkembangan bisnis terkini. Profil diberi label “Dihasilkan dengan data yang tersedia untuk umum dan wawasan AWS AI.”

- “Buat profil pelanggan untuk Acme Corp”
- “Apa yang kita ketahui tentang industri dan model bisnis pelanggan ini?”
- “Kumpulkan ikhtisar perusahaan untuk pertemuan saya yang akan datang dengan GlobalTech”

Rekomendasi solusi

Agen mereferensikan silang solusi terdaftar Anda terhadap persyaratan peluang, menunjukkan nama solusi, deskripsi, dan apakah itu sudah melekat pada peluang.

- “Manakah dari solusi kami yang paling cocok dengan peluang O1234567890?”

- “Apakah solusi analisis data kami sudah melekat pada kesepakatan ini?”
- “Merekomendasikan solusi untuk pelanggan yang ingin memigrasikan beban kerja SAP mereka”

Rekomendasi pendanaan

Agan mengevaluasi setiap peluang terhadap program AWS pendanaan yang tersedia berdasarkan rincian peluang dan kriteria kelayakan program. Ketika kecocokan ditemukan, itu menampilkan nama program, deskripsi, dan penalaran terperinci. Anda kemudian dapat memperkirakan jumlah pendanaan, membuat draf permintaan dana yang diisi secara otomatis, atau mempelajari lebih lanjut tentang program percakapan. Ketersediaan anggaran SCA (Strategic Collaboration Agreement) muncul ketika relevan, dan semua tindakan menghormati izin IAM.

- “Apakah saya memenuhi syarat untuk program pendanaan apa pun pada peluang O6789012345?”
- “Perkirakan jumlah pendanaan untuk POC dengan pelanggan ini”
- “Buat aplikasi manfaat MAP untuk kesempatan ini”

Rekomendasi langkah selanjutnya

Agan mengevaluasi peluang terhadap panduan perkembangan tahap, membandingkan data saat ini dengan kriteria peluang yang memenuhi syarat, dan mengidentifikasi dengan tepat informasi apa yang AWS masih perlu Anda kumpulkan. Hasilnya adalah rencana aksi yang diprioritaskan yang didasarkan pada AWS standar penjualan bersama.

- “Apa yang harus saya lakukan selanjutnya untuk memajukan peluang O1234567890?”
- “Apakah kesempatan ini siap untuk diserahkan? Bidang apa yang hilang?”
- “Apa persyaratan untuk memindahkan kesepakatan ini dari Prospek ke Qualified?”

Kemajuan peluang

Agan menerima dokumen pendukung (transkrip rapat, catatan panggilan, ringkasan email), mengekstrak informasi yang relevan, memetakannya ke bidang peluang, mengevaluasi persyaratan tahap, dan memperbarui peluang. Jika kesenjangan tetap ada, ia mengembalikan rincian persyaratan yang puas vs. tidak puas.

- “Ini catatan panggilan saya - perbarui peluang O1234567890 dengan detail yang relevan”

- “Saya melampirkan transkrip pertemuan. Kemajuan kesempatan ini berdasarkan apa yang kita diskusikan.”
- “Tinjau ringkasan email ini dan beri tahu saya bidang peluang mana yang memuaskan”

Pengalaman agen untuk orientasi mitra

Agen mengotomatiskan orientasi mitra ke Partner Central dan memberikan bantuan terpandu untuk penyiapan penjual Marketplace dan kepatuhan PRM — semuanya melalui bahasa alami.

Kemampuan agen:

- Otomatisasi profil mitra - Pindai situs web Anda, isi otomatis profil mitra Anda, dan kelola visibilitas, kontak pemimpin aliansi, domain pelatihan, dan koneksi akun
- Panduan penyiapan penjual — Step-by-step panduan untuk pendaftaran penjual Marketplace termasuk formulir pajak, perbankan, kepatuhan (KYC/BAV/SU), katalog ESC, dan peran terkait layanan
- Panduan kepatuhan PRM — Menilai kesiapan PRM, mengambil kode produk untuk penandaan pendapatan, dan memverifikasi koneksi akun anak perusahaan untuk atribusi pendapatan konsolidasi

Otomatisasi profil mitra

Agen memindai situs web perusahaan Anda untuk mengekstrak dan mengisi profil mitra Anda secara otomatis, lalu memandu Anda melalui celah yang tersisa. Ini dapat memperbarui bidang profil, mengubah visibilitas, mengelola kontak pemimpin aliansi Anda, menautkan domain sertifikasi pelatihan, dan menangani undangan koneksi akun.

- “Bisakah Anda melihat situs web saya dan mengisi profil saya?”
- “Jadikan profil kami publik sehingga pelanggan AWS dapat menemukan kami”
- “Perbarui kontak pemimpin aliansi kami ke [nama] di [email]”
- “Connect akun anak perusahaan EMEA kami”

Panduan pengaturan penjual

Agen berjalan melalui pendaftaran penjual Marketplace end-to-end, menentukan formulir pajak yang tepat berdasarkan negara dan jenis entitas Anda, menjelaskan pengaturan perbankan dan pencairan

berdasarkan wilayah, dan mengidentifikasi persyaratan kepatuhan (KYC, Verifikasi Rekening Bank, verifikasi Pengguna Sekunder) yang berlaku untuk Anda.

- “Saya ingin mulai menjual di Marketplace — di mana saya mulai?”
- “Formulir pajak apa yang saya butuhkan? Saya sebuah perusahaan di Jerman”
- “Apakah saya membutuhkan KYC? Saya menjual ke pelanggan di Prancis”
- “Bagaimana cara mengatur pembayaran jika saya berada di luar AS?”

Panduan kepatuhan PRM

Agen memeriksa apakah akun Anda PRM-ready — memverifikasi bahwa akun Anda terhubung, daftar ada, dan kode produk Anda tersedia untuk penandaan. Untuk mitra dengan akun anak perusahaan, ini memverifikasi bahwa semua akun penjual ditautkan dengan benar untuk atribusi pendapatan konsolidasi.

- “Apakah saya patuh PRM?”
- “Di mana saya menemukan kode produk saya untuk penandaan?”
- “Beri tahu saya semua langkah yang perlu saya lakukan agar sesuai dengan PRM”

Memulai

Siap untuk mengatur agen Pusat Mitra MCP Server? Buka panduan [Memulai](#) untuk petunjuk pengaturan langkah demi langkah.

Memulai dengan Mitra Central Agent MCP Server

Panduan ini memandu Anda melalui pengaturan akses terprogram ke Server MCP Agen Pusat Mitra menggunakan klien MCP khusus. Server menggunakan HTTPS langsung dengan otentikasi SiGv4 — tidak diperlukan proxy atau plugin IDE.

Prasyarat

Sebelum Anda mulai, pastikan Anda memiliki:

- Akun Partner Central yang aktif (dimigrasikan ke AWS konsol)

- AWS Akun dengan izin IAM untuk Partner Central
- AWS CLI diinstal dan dikonfigurasi dengan kredensial
- Akses ke wilayah us-east-1 (Virginia N.)
- Konektivitas HTTPS ke `partnercentral-agents-mcp.us-east-1.api.aws`
- Dukungan TLS 1.2+ di klien HTTP Anda
- MCP-compatible Klien yang mendukung penandatanganan JSON-RPC permintaan 2.0 dan SiGv4

Langkah 1: Siapkan izin IAM

Server MCP Agen Pusat Mitra memerlukan izin IAM pada dua tingkatan: akses protokol (untuk berkomunikasi dengan titik akhir MCP) dan akses data (untuk melakukan operasi Pusat Mitra).

Melampirkan kebijakan IAM

Untuk melampirkan kebijakan ke identitas IAM Anda menggunakan Konsol AWS Manajemen:

1. Buka [konsol IAM](#).
2. Di panel navigasi kiri, pilih Pengguna, Grup pengguna, atau Peran tergantung pada identitas yang ingin Anda lampirkan kebijakan, lalu pilih nama pengguna, grup, atau peran tertentu.
3. Pilih tab Izin.
4. Pilih Lampirkan kebijakan (atau Tambahkan izin jika ini pertama kalinya).
5. Dalam daftar kebijakan, cari dan pilih kebijakan terkelola yang ingin dilampirkan (misalnya, kebijakan kustom yang Anda buat dari blok JSON di bawah).
6. Pilih Lampirkan kebijakan (atau Berikutnya dan kemudian Tambahkan izin) untuk mengonfirmasi.

Izin segera berlaku. Anda dapat melampirkan beberapa kebijakan ke identitas yang sama.

Direkomendasikan: Gunakan kebijakan terkelola

Cara termudah untuk memberikan akses protokol MCP adalah dengan melampirkan kebijakan `AWSMcpServiceActionsFullAccess` terkelola ke identitas IAM Anda. Kebijakan ini mencakup semua izin yang diperlukan untuk berinteraksi dengan server MCP.

Untuk kontrol halus, Anda dapat menggunakan kunci `aws:IsMcpServiceAction` kondisi dalam kebijakan IAM Anda untuk cakupan izin khusus untuk tindakan layanan MCP.

Izin minimum untuk akses protokol MCP

Minimal, identitas IAM Anda memerlukan tindakan ini untuk berinteraksi dengan server MCP:

Tindakan	Deskripsi
<code>partnercentral:UseSession</code>	Diperlukan untuk membuat, memperbarui, dan mengambil sesi percakapan

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "partnercentral:UseSession"
      ],
      "Resource": "*",
      "Condition": {
        "Bool": {
          "aws:IsMcpServiceAction": "true"
        }
      }
    }
  ]
}
```

Izin akses data

Untuk benar-benar melakukan operasi Partner Central melalui agen, Anda memerlukan izin tambahan berdasarkan kasus penggunaan Anda.

Manajemen peluang:

```
{
  "Effect": "Allow",
  "Action": [
    "partnercentral:List*",
    "partnercentral:Get*",
```

```
        "partnercentral:CreateOpportunity",
        "partnercentral:UpdateOpportunity",
        "partnercentral:SubmitOpportunity",
        "partnercentral:AssignOpportunity",
        "partnercentral:AssociateOpportunity",
        "partnercentral:DisassociateOpportunity"
    ],
    "Resource": "*"
}
```

Program pendanaan:

```
{
  "Effect": "Allow",
  "Action": [
    "partnercentral:ListBenefitAllocations",
    "partnercentral:ListBenefitApplications",
    "partnercentral:CreateBenefitApplication",
    "partnercentral:GetBenefitApplication",
    "partnercentral:UpdateBenefitApplication",
    "partnercentral:SubmitBenefitApplication",
    "partnercentral:AmendBenefitApplication",
    "partnercentral:CancelBenefitApplication",
    "partnercentral:RecallBenefitApplication",
    "partnercentral:AssociateBenefitApplicationResource",
    "partnercentral:DisassociateBenefitApplicationResource"
  ],
  "Resource": "*"
}
```

Akses Marketplace:

```
{
  "Effect": "Allow",
  "Action": [
    "aws-marketplace:DescribeEntity",
    "aws-marketplace:DescribeAgreement",
    "aws-marketplace:SearchAgreements",
    "aws-marketplace:ListEntities"
  ],
  "Resource": "*"
}
```

Orientasi Pusat Mitra:

```
{
  "Effect": "Allow",
  "Action": [
    "partnercentral:ListPartners",
    "partnercentral:GetPartner",
    "partnercentral:GetProfileVisibility",
    "partnercentral:GetAllianceLeadContact",
    "partnercentral:GetProfileUpdateTask",
    "partnercentral:StartProfileUpdateTask",
    "partnercentral:CancelProfileUpdateTask",
    "partnercentral:PutProfileVisibility",
    "partnercentral:PutAllianceLeadContact",
    "partnercentral:SendEmailVerificationCode",
    "partnercentral:AssociateAwsTrainingCertificationEmailDomain",
    "partnercentral:DisassociateAwsTrainingCertificationEmailDomain",
    "partnercentral:GetAccountConnections",
    "partnercentral:GetConnectionInvitations",
    "partnercentral:CreateConnectionInvitation",
    "partnercentral:CancelConnectionInvitation",
    "partnercentral:RespondConnectionInvitation",
    "partnercentral:CancelConnection",
    "partnercentral:ManageConnectionPreferences"
  ],
  "Resource": "*"
}
```

Untuk penyiapan penjual Marketplace, tambahkan juga:

```
{
  "Effect": "Allow",
  "Action": [
    "aws-marketplace:ListEntities",
    "aws-marketplace:DescribeEntity",
    "aws-marketplace:DescribeChangeSet",
    "aws-marketplace:StartChangeSet"
  ],
  "Resource": "*"
}
```

Untuk manajemen peran terkait layanan (Penjualan Kembali Authorizations/CPPO):

```
{
  "Effect": "Allow",
  "Action": [
    "iam:GetRole",
    "iam:CreateServiceLinkedRole"
  ],
  "Resource": "*"
}
```

Kebijakan akses penuh

Untuk pengembangan dan pengujian, Anda dapat menggabungkan semua izin ke dalam satu kebijakan:

```
aws iam create-policy \
  --policy-name PartnerCentralAgentsFullAccess \
  --policy-document '{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": [
          "partnercentral:UseSession",
          "partnercentral:List*",
          "partnercentral:Get*",
          "partnercentral:CreateOpportunity",
          "partnercentral:UpdateOpportunity",
          "partnercentral:SubmitOpportunity",
          "partnercentral:AssignOpportunity",
          "partnercentral:AssociateOpportunity",
          "partnercentral:DisassociateOpportunity",
          "partnercentral:CreateResourceSnapshot",
          "partnercentral:CreateResourceSnapshotJob",
          "partnercentral:StartResourceSnapshotJob",
          "partnercentral:CreateEngagement",
          "partnercentral:CreateEngagementInvitation",
          "partnercentral:RejectEngagementInvitation",
          "partnercentral:StartEngagementByAcceptingInvitationTask",
          "partnercentral:StartEngagementFromOpportunityTask",
          "partnercentral:CreateBenefitApplication",
          "partnercentral:UpdateBenefitApplication",
          "partnercentral:SubmitBenefitApplication",
          "partnercentral:AmendBenefitApplication",
```



```

        "partnercentral:CancelBenefitApplication",
        "partnercentral:RecallBenefitApplication",
        "partnercentral:AssociateBenefitApplicationResource",
        "partnercentral:DisassociateBenefitApplicationResource",
        "partnercentral:ListPartners",
        "partnercentral:StartProfileUpdateTask",
        "partnercentral:CancelProfileUpdateTask",
        "partnercentral:PutProfileVisibility",
        "partnercentral:PutAllianceLeadContact",
        "partnercentral:SendEmailVerificationCode",
        "partnercentral:AssociateAwsTrainingCertificationEmailDomain",
        "partnercentral:DisassociateAwsTrainingCertificationEmailDomain",
        "partnercentral:CreateConnectionInvitation",
        "partnercentral:CancelConnectionInvitation",
        "partnercentral:RespondConnectionInvitation",
        "partnercentral:CancelConnection",
        "partnercentral:ManageConnectionPreferences"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:DescribeAgreement",
        "aws-marketplace:SearchAgreements",
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeChangeSet",
        "aws-marketplace:StartChangeSet"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "iam:GetRole",
        "iam:CreateServiceLinkedRole"
    ],
    "Resource": "*"
}
]
}'

```

Read-only kebijakan

Untuk lingkungan produksi atau kasus penggunaan hanya-baca, batasi izin untuk membaca operasi:

```
aws iam create-policy \
  --policy-name PartnerCentralAgentReadOnly \
  --policy-document '{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": [
          "partnercentral:UseSession",
          "partnercentral:List*",
          "partnercentral:Get*",
          "partnercentral:ListPartners",
          "partnercentral:GetPartner",
          "partnercentral:GetProfileVisibility",
          "partnercentral:GetAllianceLeadContact",
          "partnercentral:GetProfileUpdateTask",
          "partnercentral:GetAccountConnections",
          "partnercentral:GetConnectionInvitations"
        ],
        "Resource": "*"
      },
      {
        "Effect": "Allow",
        "Action": [
          "aws-marketplace:DescribeEntity",
          "aws-marketplace:DescribeAgreement",
          "aws-marketplace:SearchAgreements",
          "aws-marketplace:ListEntities",
          "aws-marketplace:DescribeChangeSet"
        ],
        "Resource": "*"
      },
      {
        "Effect": "Allow",
        "Action": [
          "iam:GetRole"
        ],
        "Resource": "*"
      }
    ]
  }
```

```
}'
```

Langkah 2: Hubungkan klien MCP Anda

Server MCP Agen Pusat Mitra menggunakan HTTPS langsung dengan penandatanganan permintaan SiGv4. Tidak ada lapisan proxy — klien MCP Anda mengirimkan permintaan JSON-RPC 2.0 langsung ke titik akhir.

Titik akhir

```
https://partnercentral-agents-mcp.us-east-1.api.aws/mcp
```

Autentikasi

Semua permintaan harus ditandatangani dengan [AWS Signature Versi 4](#) menggunakan:

- Nama layanan: `partnercentral-agents-mcp`
- Wilayah: `us-east-1`

Inisialisasi koneksi MCP

Kirim `initialize` permintaan untuk membuat protokol:

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "initialize",
  "params": {
    "protocolVersion": "2025-03-26",
    "capabilities": {},
    "clientInfo": {
      "name": "my-partner-client",
      "version": "1.0.0"
    }
  }
}
```

Tanggapan yang diharapkan:

```
{
```

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "protocolVersion": "2025-03-26",
    "capabilities": {
      "tools": {
        "listChanged": false
      }
    },
    "serverInfo": {
      "name": "PartnerCentralAgentMCPServer",
      "version": "1.0.0"
    }
  }
}
```

Daftar alat yang tersedia

```
{
  "jsonrpc": "2.0",
  "id": 2,
  "method": "tools/list",
  "params": {}
}
```

Menandatangani panggilan Anda dengan header MCP

Saat mengajukan permintaan ke agen Mitra Pusat MCP, sebaiknya sertakan header MCP kustom menggunakan metode berikut untuk membantu AWS mengidentifikasi sumber aplikasi klien, memantau penggunaan, dan kinerja audit. AWS menggunakan header ini untuk membedakan jenis aplikasi klien yang melakukan panggilan dan untuk mengumpulkan wawasan tentang tingkat keberhasilan implementasi klien yang berbeda.

Metode 1: bidang `_meta` (programmatic/stateless MCP)

Untuk kode yang secara langsung membangun `tools/call` permintaan MCP, berikan `_meta` bidang pada permintaan.

```
{
  "method": "tools/call",
  "params": {
```

```

    "name": "sendMessage",
    "arguments": {
      "content": [
        {
          "type": "text",
          "text": "List my open opportunities with expected close date in Q1
2026"
        }
      ],
      "catalog": "AWS"
    },
    "_meta": {
      "integrator": "<Integrator's Company Name / Direct>",
      "sourceProduct": "<Integrator's Application Name>"
    }
  }
}

```

Metode 2: ClientInfo (agen kustom berbasis sesi)

Untuk klien MCP kustom yang membuat sesi, berikan info header MCP di dalam bidang: `clientInfo`

```

{
  "method": "initialize",
  "params": {
    "protocolVersion": "2024-11-05",
    "clientInfo": {
      "integrator": "<Integrator's Company Name / Direct>",
      "sourceProduct": "<Integrator's Application Name>"
    }
  }
}

```

Bidang `clientInfo`:

- `integrator`— Nama perusahaan atau “Langsung” untuk mitra

Contoh: AWS

- `sourceProduct`— Product/agent nama

Contoh: AWS CRM Connector

Metode 3: Parameter URL (hanya MCP yang dihosting)

Hanya untuk klien MCP yang di-host di mana integrator tidak dapat memodifikasi bidang protokol. Gunakan parameter URL:

URL Server: `https://mcp.partnercentral.aws?appId=<Integrator's Company Name / Direct>`

Langkah 3: Verifikasi pengaturan Anda

Kirim pesan sederhana untuk mengonfirmasi semuanya berfungsi. Gunakan Sandbox katalog untuk pengujian:

```
{
  "jsonrpc": "2.0",
  "id": 3,
  "method": "tools/call",
  "params": {
    "name": "sendMessage",
    "arguments": {
      "content": [
        {
          "type": "text",
          "text": "Hello, what can you help me with?"
        }
      ],
      "catalog": "Sandbox"
    }
  }
}
```

Jika Anda menerima respons dengan `"status": "complete"` dan balasan teks dari agen, pengaturan Anda berfungsi dengan benar. Respons juga akan mencakup `sessionId` yang dapat Anda gunakan untuk pesan tindak lanjut.

Langkah 4: Jalankan tugas pertama Anda

Tanyakan peluang Anda

```
{
  "jsonrpc": "2.0",
  "id": 4,
```

```

    "method": "tools/call",
    "params": {
      "name": "sendMessage",
      "arguments": {
        "sessionId": "session-xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx",
        "content": [
          {
            "type": "text",
            "text": "List my open opportunities with expected revenue over
$50K"
          }
        ],
        "catalog": "AWS"
      }
    }
  }
}

```

Periksa kelayakan pendanaan

```

{
  "jsonrpc": "2.0",
  "id": 5,
  "method": "tools/call",
  "params": {
    "name": "sendMessage",
    "arguments": {
      "sessionId": "session-xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx",
      "content": [
        {
          "type": "text",
          "text": "Am I eligible for MAP funding for opportunity
01234567890?"
        }
      ],
      "catalog": "AWS"
    }
  }
}

```

Ambil riwayat sesi

```

{

```

```
{
  "jsonrpc": "2.0",
  "id": 6,
  "method": "tools/call",
  "params": {
    "name": "getSession",
    "arguments": {
      "sessionId": "session-xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx",
      "catalog": "AWS"
    }
  }
}
```

Orientasi mitra

```
{
  "jsonrpc": "2.0",
  "id": 7,
  "method": "tools/call",
  "params": {
    "name": "sendMessage",
    "arguments": {
      "sessionId": "session-xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx",
      "content": [
        {
          "type": "text",
          "text": "Help me onboard to Partner Central"
        }
      ],
      "catalog": "AWS"
    }
  }
}
```

Tugas orientasi lainnya untuk dicoba:

- “Bimbing saya melalui proses wawancara pajak”
- “Bisakah Anda melihat situs web saya dan mengisi profil mitra saya?”
- “Apa yang masih harus saya lakukan untuk siap menjual di Marketplace?”

Pertimbangan keamanan

- Jangan meneruskan AWS kredensial melalui parameter alat MCP. Otentikasi ditangani oleh penandatanganan permintaan SiGv4 di lapisan transport.
- Gunakan katalog Sandbox untuk pengujian dan pengembangan. "Sandbox" Katalog menyediakan lingkungan terisolasi yang tidak mempengaruhi data mitra produksi.
- Menerapkan kebijakan IAM dengan hak istimewa paling rendah dalam produksi. Gunakan kebijakan hanya-baca untuk memantau dan melaporkan kasus penggunaan. Hanya berikan izin menulis ketika pengguna perlu memperbarui peluang atau mengirimkan aplikasi pendanaan.
- Tinjau operasi tulis dengan hati-hati. Server menggunakan persetujuan manusia-in-the-loop untuk semua operasi penulisan. Ketika tindakan tulis diusulkan, tinjau parameter sebelum menyetujui.
- Data sesi bersifat sementara. Sesi kedaluwarsa 48 jam setelah pembuatan. Jangan mengandalkan sesi untuk penyimpanan data jangka panjang.
- Unggahan file masuk ke bucket S3 fana. File yang diunggah disimpan sementara dan tidak disimpan secara permanen. Jangan mengunggah file yang berisi kredensial, rahasia, atau informasi sensitif lainnya.

Langkah selanjutnya

- [Referensi Konfigurasi](#) - Referensi lengkap untuk titik akhir, tindakan IAM, manajemen sesi, dan kode kesalahan
- [Referensi Alat](#) - Dokumentasi terperinci untuk sendMessage dan getSession alat

Referensi Konfigurasi

Halaman ini menyediakan referensi lengkap untuk menghubungkan dan mengonfigurasi Server MCP Agen Pusat Mitra.

Titik akhir

Properti	Nilai
URL	<code>https://partnercentral-agents-mcp.us-east-1.api.aws/mcp</code>

Properti	Nilai
Region	us-east-1 (Virginia N.) hanya
Protokol	JSON-RPC 2.0 melalui HTTPS
Streaming	Server-Sent Acara (SSE)
Autentikasi	AWS Versi Tanda Tangan 4
Nama layanan SiGv4	partnercentral-agents-mcp
Persyaratan TLS	TLS 1.2 atau lebih tinggi

Izin IAM

Semua tindakan IAM menggunakan `partnercentral:` awalan kecuali dinyatakan lain.

Akses protokol MCP

Cara termudah untuk memberikan akses protokol MCP adalah dengan melampirkan kebijakan `AWSMcpServiceActionsFullAccess` terkelola ke identitas IAM Anda. Kebijakan ini mencakup semua izin yang diperlukan untuk berinteraksi dengan server MCP. Untuk kontrol halus, Anda dapat menggunakan kunci `aws:IsMcpServiceAction` kondisi dalam kebijakan IAM Anda untuk cakupan izin khusus untuk tindakan layanan MCP.

Tindakan	Deskripsi
<code>partnercentral:UseSession</code>	Membuat, memperbarui, dan mengambil sesi percakapan

Manajemen peluang

Tindakan	Deskripsi
<code>partnercentral:List*</code>	Buat daftar peluang, solusi, dan sumber daya Pusat Mitra lainnya

Tindakan	Deskripsi
<code>partnercentral:Get*</code>	Ambil detail untuk peluang individu dan sumber daya lainnya
<code>partnercentral:CreateOpportunity</code>	Ciptakan peluang
<code>partnercentral:UpdateOpportunity</code>	Ubah bidang peluang (tahap, tanggal tutup, pendapatan, dll.)
<code>partnercentral:SubmitOpportunity</code>	Kirimkan kesempatan untuk AWS ditinjau
<code>partnercentral:AssignOpportunity</code>	Tetapkan kesempatan kepada perwakilan mitra
<code>partnercentral:AssociateOpportunity</code>	Tautkan peluang ke sumber daya lain (solusi, dll.)
<code>partnercentral:DisassociateOpportunity</code>	Hapus tautan antara peluang dan sumber daya lain
<code>partnercentral:StartEngagementFromOpportunityTask</code>	Kirimkan kesempatan untuk memulai keterlibatan dari sebuah kesempatan

Program pendanaan

Tindakan	Deskripsi
<code>partnercentral:ListBenefitAllocations</code>	Buat daftar alokasi manfaat yang tersedia untuk akun Anda
<code>partnercentral:ListBenefitApplications</code>	Daftar aplikasi manfaat yang dikirimkan
<code>partnercentral:CreateBenefitApplication</code>	Buat aplikasi manfaat baru (MAP, POC, WMP)
<code>partnercentral:GetBenefitApplication</code>	Mengambil rincian aplikasi manfaat tertentu

Tindakan	Deskripsi
<code>partnercentral:UpdateBenefitApplication</code>	Perbarui aplikasi manfaat yang tertunda
<code>partnercentral:AssociateBenefitApplicationResource</code>	Tautkan sumber daya ke aplikasi manfaat
<code>partnercentral:DisassociateBenefitApplicationResource</code>	Hapus tautan sumber daya dari aplikasi manfaat

Akses Marketplace

Tindakan	Deskripsi
<code>aws-marketplace:DescribeEntity</code>	Mengambil rincian entitas pasar
<code>aws-marketplace:SearchAgreements</code>	Cari perjanjian pasar
<code>aws-marketplace:ListEntities</code>	Daftar entitas pasar

Agen Orientasi

Manajemen akun mitra (**partnercentral**):

Tindakan	Deskripsi
<code>partnercentral:ListPartners</code>	Daftar pendaftaran mitra untuk akun
<code>partnercentral:GetPartner</code>	Ambil registrasi mitra dan detail profil
<code>partnercentral:GetProfileVisibility</code>	Ambil visibilitas profil saat ini () PUBLIC/PRIVATE
<code>partnercentral:GetAllianceLeadContact</code>	Ambil informasi kontak pemimpin aliansi

Tindakan	Deskripsi
<code>partnercentral:GetProfileUpdateTask</code>	Periksa status pembaruan profil asinkron yang tertunda
<code>partnercentral:StartProfileUpdateTask</code>	Kirim pembaruan profil mitra (async)
<code>partnercentral:CancelProfileUpdateTask</code>	Membatalkan tugas pembaruan profil yang tertunda
<code>partnercentral:PutProfileVisibility</code>	Ubah visibilitas profil menjadi PUBLIK atau PRIBADI
<code>partnercentral:PutAllianceLeadContact</code>	Perbarui kontak utama aliansi (nama, judul, email)
<code>partnercentral:SendEmailVerificationCode</code>	Kirim kode verifikasi untuk perubahan berbasis email domain/contact
<code>partnercentral:AssociateAwsTrainingCertificationEmailDomain</code>	Menautkan domain email untuk pelacakan sertifikasi pelatihan
<code>partnercentral:DisassociateAwsTrainingCertificationEmailDomain</code>	Menghapus domain email sertifikasi pelatihan yang ditautkan
<code>partnercentral:GetAccountConnections</code>	Buat daftar koneksi akun aktif (mis., Tautan anak perusahaan)
<code>partnercentral:GetConnectionInvitations</code>	Daftar undangan koneksi terkirim dan terima yang tertunda
<code>partnercentral:CreateConnectionInvitation</code>	Kirim undangan koneksi ke AWS akun lain
<code>partnercentral:CancelConnectionInvitation</code>	Membatalkan undangan koneksi keluar yang tertunda

Tindakan	Deskripsi
<code>partnercentral:RespondConnectionInvitation</code>	Menerima atau menolak undangan koneksi masuk
<code>partnercentral:CancelConnection</code>	Membatalkan koneksi akun aktif
<code>partnercentral:ManageConnectionPreferences</code>	Dapatkan atau perbarui preferensi berbagi antar akun yang terhubung

Pengaturan penjual Marketplace (**aws-marketplace**):

Tindakan	Deskripsi
<code>aws-marketplace:ListEntities</code>	Daftar entitas Marketplace (misalnya, entitas penjual)
<code>aws-marketplace:DescribeEntity</code>	Mengambil rincian lengkap dari entitas Marketplace
<code>aws-marketplace:DescribeChangeSet</code>	Memeriksa status set perubahan yang dikirimkan
<code>aws-marketplace:StartChangeSet</code>	Kirim set perubahan Marketplace (mis., Pembaruan profil penjual)

IAM — peran terkait layanan (): **iam**

Tindakan	Deskripsi
<code>iam:GetRole</code>	Periksa apakah peran terkait layanan Otorisasi Penjualan Kembali Marketplace ada
<code>iam:CreateServiceLinkedRole</code>	Buat <code>AWSServiceRoleForMarketplaceResaleAuthorization</code> peran

Lingkungan katalog

Katalog	Deskripsi
"AWS"	Lingkungan produksi. Semua operasi memengaruhi data mitra langsung.
"Sandbox"	Lingkungan pengujian yang terisolasi. Tidak berdampak pada data produksi. Gunakan untuk pengembangan dan validasi.

Manajemen sesi

Properti	Nilai
Format ID Sesi	UUID v4 dengan session- awalan (mis.,) session-550e8400-e29b-41d4-a716-446655440000
Pembuatan sesi	Otomatis pada sendMessage panggilan pertama tanpa sessionId
Kedaluwarsa sesi	48 jam dari pembuatan sesi (kedaluwarsa absolut, bukan berbasis inaktivitas)

Unggahan file

Properti	Nilai
Maks file per pesan	3
Batas ukuran gambar	3,75 MB
Batas ukuran dokumen	4,5 MB
Ekstensi yang diizinkan	doc, docx, pdf, png, jpeg, xlsx, csv, txt

Properti	Nilai
Ember S3 (produksi)	aws-partner-central-marketplace-ephemeral-wri teonly-file
Unggah jalur	s3://{bucket}/{aws-account-id}/
Persyaratan URI S3	Harus menyertakan parameter <code>versionId</code> kueri

Unggah alur kerja

1. Unggah file Anda ke bucket S3 yang sesuai di bawah awalan ID AWS akun Anda.
2. Perhatikan URI S3 termasuk yang `versionId` dikembalikan oleh unggahan.
3. Sertakan file sebagai blok document konten dalam `sendMessage` panggilan Anda.

Contoh format URI S3:

```
s3://aws-partner-central-marketplace-ephemeral-writeonly-files/123456789012/my-  
document.pdf?versionId=abc123
```

Kode error

Code	Nama	Deskripsi
-32001	AUTHENTICATION_FAI LURE	Sigv4 signature tidak valid atau kredensialnya telah kedaluwarsa
-31004	TOOL_PERMISSION_DE NIED	Identitas IAM tidak memiliki <code>partnercentral:</code> tindakan yang diperlukan untuk operasi ini

Code	Nama	Deskripsi
-32002	ACCESS_DENIED	Akses umum ditolak (akun tidak terdaftar, ketidakcocokan wilayah, dll.)
-32004	LIMIT_EXCEEDED	Batas tarif atau kuota terlampaui. Coba lagi dengan backoff eksponensial.
-30001	RESOURCE_NOT_FOUND	Sumber daya yang diminta (sesi, peluang, dll.) Tidak ada
-32600	INVALID_REQUEST	JSON-RPC Permintaan cacat atau parameter tidak valid
-32603	INTERNAL_ERROR	Server-side kesalahan. Coba lagi permintaannya.

Batas tarif

Server memberlakukan batas tarif per akun:

Operasi	Tingkat berkelanjutan	Burst
sendMessage	2 permintaan per menit	10
Semua operasi lainnya	10 permintaan per menit	20

Jika Anda melebihi batas ini, server mengembalikan kode kesalahan -32004 (LIMIT_EXCEEDED). Terapkan backoff eksponensial dengan jitter di klien Anda untuk menangani pembatasan laju dengan anggun.

Jenis acara Streaming (SSE)

Ketika `stream` diatur ke `true` dalam `sendMessage` panggilan, server mengembalikan Server-Sent Peristiwa dengan jenis acara ini:

Jenis Acara	Deskripsi
<code>stream_start</code>	Koneksi aliran dibuat
<code>assistant-response.start</code>	Agen telah mulai menghasilkan respon
<code>assistant-response.delta</code>	Potongan teks tambahan dari respons agen
<code>assistant-response.completed</code>	Agen telah selesai menghasilkan respons
<code>server-tool-use</code>	Agen menggunakan alat internal (operasi baca)
<code>server-tool-response</code>	Hasil pemanggilan alat internal
<code>tool_approval_request</code>	Agen meminta persetujuan manusia untuk operasi menulis
<code>stream_end</code>	Penutupan koneksi aliran
<code>done</code>	Acara terakhir yang menunjukkan aliran SSE selesai

Langkah selanjutnya

- [Referensi Alat](#) - Dokumentasi terperinci untuk `sendMessage` dan `getSession` alat

Alat Referensi

Server MCP Agen Pusat Mitra mengekspos dua alat MCP: `sendMessage` untuk semua interaksi agen, dan `getSession` untuk mengambil status sesi. Semua operasi Partner Central - pertanyaan peluang, aplikasi pendanaan, analisis dokumen - ditangani melalui bahasa alami melalui `sendMessage`

Ikhtisar alat

Alat	Deskripsi	Kategori
sendMessage	Kirim pesan ke agen AI Pusat Mitra. Mendukung teks, lampiran file, dan tanggapan persetujuan manusia-in-the-loop.	Membaca/Menulis
getSession	Ambil status sesi termasuk riwayat percakapan, peristiwa, dan metadata.	Read-only

SendMessage

Alat utama untuk semua interaksi agen AI Pusat Mitra. Gunakan alat ini untuk mengajukan pertanyaan, meminta tindakan, melampirkan dokumen untuk analisis, dan menanggapi permintaan persetujuan untuk operasi penulisan.

Agen mempertahankan konteks percakapan dalam sesi, sehingga Anda dapat mengajukan pertanyaan lanjutan tanpa mengulangi konteks sebelumnya.

Parameter

- **content(wajib)** - Array blok konten. Setiap blok harus menyertakan **type** bidang yang menentukan struktur blok. Anda dapat menyertakan beberapa blok dalam satu pesan (mis., Teks +lampiran dokumen).

Jenis blok konten:

Tipe	Bidang	Deskripsi
text	type(diperlukan), text (wajib)	Teks pesan pengguna dikirim ke agen
document	type(wajib), filename (wajib), s3Uri (wajib)	Lampiran file untuk agen untuk dianalisis. s3UriHarus

Tipe	Bidang	Deskripsi
		menyertakan <code>versionId</code> parameter.
<code>tool_approval_response</code>	<code>type(wajib)</code> , <code>toolUseId(wajib)</code> , <code>decision(wajib)</code> , <code>message(optional)</code>	Tanggapan terhadap permintaan persetujuan manusia-in-the-loop

- `catalog(wajib)` - Lingkungan target untuk operasi.

Nilai yang valid: "AWS" (produksi), "Sandbox" (pengujian)

- `sessionId(optional)` - UUID v4 mengidentifikasi sesi yang ada untuk melanjutkan. Abaikan untuk membuat sesi baru. Format: `session-xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxxx`.

Default: Sesi baru dibuat secara otomatis.

- `stream(optional)` - Aktifkan streaming Server-Sent Acara (SSE) untuk pengiriman respons waktu nyata.

Nilai valid: `true`, `false`

Default: `false`

Respons

Respons meliputi:

Bidang	Deskripsi
<code>sessionId</code>	Pengenal sesi untuk pesan tindak lanjut
<code>status</code>	Status respons: "complete" , "requires_approval" , atau "error"
<code>content</code>	Array blok konten respons dari agen

Contoh

Pesan teks dasar (sesi baru)

Permintaan:

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "method": "tools/call",
  "params": {
    "name": "sendMessage",
    "arguments": {
      "content": [
        {
          "type": "text",
          "text": "List my open opportunities with expected close date in Q1
2026"
        }
      ],
      "catalog": "AWS"
    }
  }
}
```

Respons:

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "result": {
    "content": [
      {
        "type": "text",
        "text": "I found 12 open opportunities with expected close dates in Q1
2026. Here's a summary:\n\n1. **01234567890** - Acme Corp Cloud Migration - $250,000 -
Qualified stage\n2. **01234567891** - GlobalTech Data Analytics - $180,000 - Prospect
stage\n..."
      }
    ],
    "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
    "status": "complete"
  }
}
```

```
}
```

Follow-up pesan (sesi yang ada)

Permintaan:

```
{
  "jsonrpc": "2.0",
  "id": 2,
  "method": "tools/call",
  "params": {
    "name": "sendMessage",
    "arguments": {
      "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
      "content": [
        {
          "type": "text",
          "text": "Tell me more about 01234567890. Is it ready for
submission?"
        }
      ],
      "catalog": "AWS"
    }
  }
}
```

Lampiran file

Unggah dokumen ke S3 terlebih dahulu, lalu referensikan di pesan:

```
{
  "jsonrpc": "2.0",
  "id": 3,
  "method": "tools/call",
  "params": {
    "name": "sendMessage",
    "arguments": {
      "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
      "content": [
        {
          "type": "text",
          "text": "Review this customer proposal and suggest which
opportunity it aligns with"
        }
      ]
    }
  }
}
```

```

        },
        {
            "type": "document",
            "filename": "acme-proposal.pdf",
            "s3Uri": "s3://aws-partner-central-marketplace-ephemeral-writeonly-
files/123456789012/acme-proposal.pdf?versionId=abc123def456"
        }
    ],
    "catalog": "AWS"
}
}
}

```

Kendala unggahan file:

- Maksimal 3 file per pesan
- Batas ukuran gambar: 3,75 MB
- Batas ukuran dokumen: 4,5 MB
- Ekstensi yang diizinkan: docdocx,pdf,png,,jpeg,xlsx, csv, txt
- File harus diunggah ke `s3://{bucket}/{your-aws-account-id}/`
- URI S3 harus menyertakan `versionId` parameter kueri

Human-in-the-loop alur kerja persetujuan

Ketika agen perlu melakukan operasi tulis (misalnya, memperbarui peluang, mengirimkan aplikasi pendanaan), ia mengembalikan `"requires_approval"` status dengan rincian tindakan yang diusulkan. Anda harus merespons dengan blok `tool_approval_response` konten.

Langkah 1 — Agen meminta persetujuan:

```

{
  "jsonrpc": "2.0",
  "id": 4,
  "result": {
    "content": [
      {
        "type": "text",
        "text": "I'd like to update opportunity 01234567890 with the following
changes:\n- Target close date: 2026-03-31\n- Expected revenue: $300,000\n- Stage:
Qualified\n\nPlease approve, reject, or override this action."
      }
    ]
  }
}

```

```

    },
    {
      "type": "tool_approval_request",
      "toolUseId": "tool-use-98765",
      "toolName": "update_opportunity_enhanced",
      "parameters": {
        "opportunityId": "01234567890",
        "targetCloseDate": "2026-03-31",
        "expectedRevenue": 300000,
        "stage": "Qualified"
      }
    }
  ],
  "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
  "status": "requires_approval"
}

```

Langkah 2 — Menyetujui tindakan:

```

{
  "jsonrpc": "2.0",
  "id": 5,
  "method": "tools/call",
  "params": {
    "name": "sendMessage",
    "arguments": {
      "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
      "content": [
        {
          "type": "tool_approval_response",
          "toolUseId": "tool-use-98765",
          "decision": "approve"
        }
      ],
      "catalog": "AWS"
    }
  }
}

```

Langkah 2 (alternatif) — Tolak tindakan:

```

{

```



```

    "jsonrpc": "2.0",
    "id": 5,
    "method": "tools/call",
    "params": {
      "name": "sendMessage",
      "arguments": {
        "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
        "content": [
          {
            "type": "tool_approval_response",
            "toolUseId": "tool-use-98765",
            "decision": "reject",
            "message": "The expected revenue should be $250,000, not $300,000"
          }
        ],
        "catalog": "AWS"
      }
    }
  }
}

```

Langkah 2 (alternatif) - Ganti dengan respons khusus:

```

{
  "jsonrpc": "2.0",
  "id": 5,
  "method": "tools/call",
  "params": {
    "name": "sendMessage",
    "arguments": {
      "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
      "content": [
        {
          "type": "tool_approval_response",
          "toolUseId": "tool-use-98765",
          "decision": "override",
          "message": "Use expected revenue of $250,000 and keep the stage as
Prospect instead"
        }
      ],
      "catalog": "AWS"
    }
  }
}

```

Nilai keputusan persetujuan:

Keputusan	Perilaku
"approve"	Jalankan alat dengan parameter yang diusulkan
"reject"	Jangan jalankan alat. Opsional message menjelaskan alasannya.
"override"	Berikan respons khusus atau instruksi yang dimodifikasi melalui message

Streaming dengan SSE

Aktifkan streaming untuk menerima potongan respons tambahan saat agen memproses permintaan Anda:

Permintaan:

```
{
  "jsonrpc": "2.0",
  "id": 6,
  "method": "tools/call",
  "params": {
    "name": "sendMessage",
    "arguments": {
      "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
      "content": [
        {
          "type": "text",
          "text": "Analyze my pipeline and identify opportunities at risk"
        }
      ],
      "catalog": "AWS",
      "stream": true
    }
  }
}
```

Server merespons dengan aliran peristiwa SSE:

```
event: stream_start
data: {"sessionId": "session-550e8400-e29b-41d4-a716-446655440000"}

event: assistant-response.start
data: {}

event: server-tool-use
data: {"toolName": "analyze_pipeline", "parameters": {}}

event: server-tool-response
data: {"toolName": "analyze_pipeline", "result": {"opportunitiesAnalyzed": 47,
  "atRisk": 5}}

event: assistant-response.delta
data: {"text": "I analyzed your pipeline of 47 opportunities and identified "}

event: assistant-response.delta
data: {"text": "5 that are at risk of slipping:\n\n"}

event: assistant-response.delta
data: {"text": "1. **02345678901** - Close date is past due by 15 days\n"}

event: assistant-response.completed
data: {"status": "complete"}

event: stream_end
data: {}
```

GetSession

Ambil status sesi percakapan saat ini, termasuk riwayat percakapan lengkap, peristiwa, dan metadata. Gunakan ini untuk memeriksa status sesi, meninjau interaksi masa lalu, atau melanjutkan percakapan.

Parameter

- **sessionId(wajib)** — UUID sesi untuk mengambil. Format: `session-xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx`.
- **catalog(wajib)** — Lingkungan tempat sesi tersebut dimiliki.

Nilai valid: "AWS", "Sandbox"

Respons

Bidang	Tipe	Deskripsi
sessionId	string	Pengidentifikasi sesi
createdAt	string	Stempel waktu pembuatan sesi ISO 8601
lastActivity	string	Stempel waktu ISO 8601 dari aktivitas terakhir
sequenceNumber	integer	Nomor urutan peristiwa saat ini
stateType	string	Status sesi saat ini
events	array	Riwayat percakapan lengkap (pesan pengguna, tanggapan agen, penggunaan alat)
variables	object	Variabel sesi dan metadata
eventCount	integer	Jumlah total acara dalam sesi

Contoh

Permintaan:

```
{
  "jsonrpc": "2.0",
  "id": 7,
  "method": "tools/call",
  "params": {
    "name": "getSession",
    "arguments": {
      "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
      "catalog": "AWS"
    }
  }
}
```

```
}
```

Respons:

```
{
  "jsonrpc": "2.0",
  "id": 7,
  "result": {
    "content": [
      {
        "type": "text",
        "text": "{\"sessionId\":\"session-550e8400-e29b-41d4-a716-446655440000\", \"createdAt\":\"2026-01-15T10:30:00Z\", \"lastActivity\":\"2026-01-15T11:45:00Z\", \"sequenceNumber\":8, \"stateType\":\"END_TURN\", \"eventCount\":8, \"events\":[...], \"variables\":{}}"
      }
    ]
  }
}
```

Penanganan kesalahan

Semua kesalahan mengikuti format kesalahan JSON-RPC 2.0:

```
{
  "jsonrpc": "2.0",
  "id": 1,
  "error": {
    "code": -32001,
    "message": "Authentication failed. Verify your SigV4 credentials and ensure they have not expired."
  }
}
```

Lihat [the section called “Code error”](#) daftar lengkap kode kesalahan dan artinya.

Strategi coba lagi yang disarankan

- Untuk -32004 (LIMIT_EXCEEDED): Coba lagi dengan backoff eksponensial mulai dari 1 detik
- Untuk -32603 (INTERNAL_ERROR): Coba lagi hingga 3 kali dengan backoff eksponensial
- Untuk -32001 (AUTHENTICATION_FAILURE): Segarkan kredensi dan coba lagi

- Untuk semua kesalahan lainnya: Jangan coba lagi secara otomatis - periksa pesan kesalahan dan perbaiki permintaan

Tentang AWS API Pusat Mitra

Bagian berikut memberikan informasi umum tentang API.

Topik

- [Menggunakan AWS API Pusat Mitra](#)
- [Didukung AWS Wilayah](#)
- [Izin](#)
- [Kuota untuk AWS API Pusat Mitra](#)
- [Penebangan untuk AWS API Pusat Mitra](#)
- [Pemberitahuan untuk AWS API Pusat Mitra](#)
- [Pengujian di kotak pasir](#)

Menggunakan AWS API Pusat Mitra

Bagian berikut memberikan informasi tentang penggunaan API Pusat AWS Mitra.

Topik

- [Menggunakan AWS API Akun Sentral Mitra](#)
- [Menggunakan AWS API Penjualan Sentral Mitra](#)
- [Menggunakan AWS API Manfaat Sentral Mitra](#)
- [Menggunakan AWS API Saluran Pusat Mitra](#)
- [Menggunakan API Pengukuran Pendapatan](#)

Menggunakan AWS API Akun Sentral Mitra

Mengelola akun mitra

Akun mitra terdaftar dan ditautkan dengan AWS akun yang ada untuk memberikan IAM-based pengalaman penuh. Pendekatan ini menghilangkan kebutuhan akan kredensi tingkat pengguna, memastikan bahwa semua pendaftaran dan operasi dikelola melalui entitas IAM dalam akun. AWS AWS Partner Central memungkinkan integrasi tanpa batas dengan AWS layanan, dan

model ini menyediakan manajemen entitas mitra, mendukung beberapa katalog, dan menetapkan sistem hak tingkat akun AWS.

Mitra dapat menggunakan API akun mitra yang tersedia untuk mendaftar, mengelola, dan memperbarui akun mereka:

API Registrasi Mitra

Mitra dapat mendaftarkan akun mereka menggunakan AWS akun mereka untuk Keterlibatan Mitra. Dengan menggunakan Create API, mitra akan memberikan informasi prospek aliansi yang diperlukan, menerima persyaratan APN, dan memberikan nama resmi yang unik.

API Profil Mitra

Mitra mengelola profil yang berisi detail perusahaan (nama, deskripsi, situs web, logo) dengan kontrol public/private visibilitas, validasi asinkron, dan pelacakan status yang memungkinkan satu pembaruan pada satu waktu.

API Manajemen Domain

Mitra dapat mendaftarkan dan memverifikasi domain bisnis melalui validasi email untuk menetapkan identitas organisasi dan pelatihan dan sertifikasi karyawan asosiasi.

API Koneksi Mitra

Mitra menghubungkan AWS akun mereka dengan AWS akun lain untuk berbagai tujuan, seperti berkolaborasi dalam penjualan bersama dengan mitra lain, berbagi data Penawaran AWS Marketplace antar akun, memberi wewenang kepada distributors/channel mitra untuk distribute/re menjual produk mereka, dan banyak lagi.

API Verifikasi Mitra

Verifikasi mitra merupakan prasyarat wajib untuk pendaftaran akun mitra. Sebelum mitra dapat membuat akun mitra, mereka harus menyelesaikan verifikasi bisnis dan proses verifikasi identitas. Verifikasi ini memvalidasi bahwa bisnis terdaftar secara hukum dan mengkonfirmasi identitas orang yang mendaftarkan akun.AWS

Topik

- [Bekerja dengan Registrasi Mitra](#)
- [Bekerja dengan Profil Mitra](#)

- [Bekerja dengan Manajemen Domain](#)
- [Bekerja dengan Koneksi Akun Mitra](#)
- [Bekerja dengan Verifikasi Mitra](#)

Bekerja dengan Registrasi Mitra

Registrasi Mitra

Mitra dapat mendaftarkan akun mereka menggunakan AWS akun mereka untuk Keterlibatan Mitra. Dengan menggunakan Create API, mitra akan memberikan informasi prospek aliansi yang diperlukan, menerima persyaratan APN, dan memberikan nama resmi yang unik.

Selama Registrasi

1. Pembuatan Entitas Mitra Sinkron — Pelanggan memulai proses pendaftaran mitra dengan memanggil Create API, yang melakukan validasi input dan membuat Entitas Mitra dalam permintaan yang sama.
2. Persyaratan Informasi Pimpinan Aliansi — Selama pendaftaran mitra, pelanggan harus memberikan informasi kontak pemimpin aliansi untuk komunikasi terkait Jaringan AWS Mitra (APN).
3. Syarat & Ketentuan Penegakan — Pelanggan harus menerima Syarat dan Ketentuan APN saat pendaftaran. Penerimaan diperlukan untuk menyelesaikan pendaftaran dan mengakses fitur apa pun. Ketentuan ini berlaku untuk semua manfaat dan fungsi program APN.
4. Tag-On-Create Support - Sebagai bagian dari AWS Consistent Authorization Experience (CAE) dan Tag-Based Authorization, pelanggan dapat menambahkan tag pada saat pembuatan entitas mitra.
5. Name-Based Validasi Hukum - Pendaftaran akan menegakkan keunikan berdasarkan nama resmi mitra, memastikan tidak ada pendaftaran duplikat di bawah entitas yang sama.

Pendaftaran Pos

1. Dukungan Manajemen Tag - Setelah pendaftaran, mitra dapat menandai, membatalkan tag, dan mencantumkan tag untuk entitas mitra yang ada.
2. Ambil Entitas Mitra - Setelah entitas mitra dibuat, pelanggan dapat menggunakan List API untuk mengambil ID Mitra dan kemudian menggunakan Get API untuk mengambil detail entitas mitra tertentu.

Ringkasan API

1. **CreatePartner API:** Pelanggan memulai proses pendaftaran mitra dengan memanggil CreatePartner API, yang melakukan validasi input dan membuat Entitas Mitra dalam permintaan yang sama.
2. **ListPartners API:** Setelah pendaftaran, pelanggan dapat menggunakan List API untuk mengambil daftar entitas mitra.
3. **GetPartner API:** Setelah pendaftaran, pelanggan dapat menggunakan Get API untuk mengambil detail entitas mitra tertentu.
4. **PutAllianceLeadContact API:** Memperbarui informasi kontak utama aliansi utama. Operasi ini memungkinkan mitra untuk mentransfer penunjukan pemimpin aliansi utama atau memodifikasi detail kontak sambil mempertahankan kontinuitas organisasi.
5. **GetAllianceLeadContact API:** Memperbarui informasi kontak utama aliansi utama. Operasi ini memungkinkan mitra untuk mentransfer penunjukan pemimpin aliansi utama atau memodifikasi detail kontak sambil mempertahankan kontinuitas organisasi.

Bekerja dengan Profil Mitra

Manajemen profil

Mitra mengelola profil yang berisi detail perusahaan (nama, deskripsi, situs web, logo) dengan kontrol public/private visibilitas, validasi asinkron, dan pelacakan status yang memungkinkan satu pembaruan pada satu waktu.

1. **Informasi Profil** — Profil mitra harus menyertakan detail penting perusahaan seperti nama tampilan, deskripsi, URL situs web, logo, IndustrySegments dan PrimarySolutionType.
2. **Dukungan Multilingual** - Profil mitra mendukung berbagai bahasa. Setiap versi profil yang dilokalkan berisi atribut untuk lokal.
3. **Kontrol Visibilitas Profil** — Mitra dapat mengonfigurasi visibilitas profil mereka (publik atau pribadi).
4. **Penyimpanan Profil** — Informasi profil akan disimpan dalam objek akun mitra.
5. **Manajemen Status Profil** - Pembaruan profil divalidasi secara asinkron sebelum dipublikasikan. Mitra dapat memeriksa status pembaruan profil terbaru (IN_PROGRESS, SUKSES, GAGAL, dan DIBATALKAN). Hanya profil yang disetujui yang terlihat oleh publik.
6. **Minta Kontrol Konkurensi** - Hanya satu permintaan pembaruan profil yang diizinkan pada satu waktu. Jika pembaruan profil sudah berlangsung, pelanggan harus secara eksplisit membatalkan

pembaruan yang sedang berlangsung melalui `CancelPartnerProfileUpdateTask` API sebelum memulai yang baru.

7. **Pembatalan Tugas Pembaruan** — Mitra dapat membatalkan pembaruan profil yang sedang berlangsung dengan memanggil API. `CancelPartnerProfileUpdateTask` Pembatalan hanya diperbolehkan saat pembaruan dalam status `IN_PROGRESS` dan harus diselesaikan sebelum memulai pembaruan baru.

Ringkasan API

1. **StartPutProfileTask API**: API ini menerima pembaruan profil mitra dan melakukan validasi input sinkron dasar.
2. **GetProfileUpdateTask API**: API ini memungkinkan Partner untuk mengambil status pembaruan profil saat ini. Ini dimaksudkan untuk digunakan setelah pelanggan memulai pembaruan profil melalui `StartPutPartnerProfile` API, memungkinkan Mitra untuk memeriksa apakah permintaan mereka `IN_PROGRESS`, `GAGAL`, `BERHASIL` atau `DIBATALKAN`.
3. **CancelProfileTask API**: Memungkinkan mitra untuk secara eksplisit membatalkan tugas pembaruan profil yang saat ini dalam status `IN_PROGRESS`. Ini termasuk fase pemeriksaan otomatis dan peninjauan manual. Membatalkan permintaan memungkinkan mitra untuk memulai pembaruan profil baru tanpa menunggu proses pemeriksaan saat ini selesai.
4. **PutProfileVisibility API**: Dengan memanggil API ini, mitra dapat mengontrol visibilitas profil terlepas dari lokalnya. Hal ini memungkinkan mitra untuk membuat profil publik atau pribadi tanpa memodifikasi konten.
5. **GetProfileVisibility API**: Dengan memanggil `UpdatePartnerProfileVisibility` API, mitra dapat mengontrol visibilitas profil terlepas dari lokalnya. Hal ini memungkinkan mitra untuk membuat profil publik atau pribadi tanpa memodifikasi konten.

Bekerja dengan Manajemen Domain

Manajemen Domain

Mitra dapat mendaftarkan dan memverifikasi domain bisnis melalui validasi email untuk menetapkan identitas organisasi dan pelatihan dan sertifikasi karyawan asosiasi.

Ringkasan API

1. `SendEmailVerificationCode` API: Memulai proses verifikasi email dengan mengirimkan kode verifikasi ke alamat email yang ditentukan. Digunakan untuk pembuatan kontak baru dan pembaruan alamat email.
2. `AssociateAwsTrainingCertificationEmailDomain` API: Mengaitkan domain email dengan pelatihan AWS dan sertifikasi untuk akun mitra, memungkinkan verifikasi otomatis sertifikasi karyawan.
3. `DisassociateAwsTrainingCertificationEmailDomain` API: Menghapus hubungan antara domain email dan pelatihan AWS serta sertifikasi untuk akun mitra.

Bekerja dengan Koneksi Akun Mitra

Mitra dapat mengelola koneksi antara akun mereka sendiri atau koneksi dengan AWS akun mitra lain menggunakan API Koneksi AWS Mitra. Prosesnya melibatkan dua fase:

1. Fase Undangan Koneksi: Memulai dan menanggapi permintaan koneksi
2. Fase Koneksi Aktif: Mengelola koneksi yang mapan dan aktivitas bisnis terkait

Membuat dan Mengirim Undangan Koneksi

Langkah 1: Buat Undangan Koneksi

Langkah pertama dalam membangun koneksi mitra adalah membuat dan mengirim undangan koneksi menggunakan `CreateConnectionInvitation` tindakan. Saat membuat undangan, Anda harus menentukan:

- **Katalog:** AWS Katalog (AWS atau Sandbox) tempat koneksi akan dikelola
- **ConnectionType:** Jenis hubungan yang ingin Anda bangun. Pilih salah satu cara berikut:
 - **OPPORTUNITY_COLLABORATION:** Gunakan jenis ini saat Anda ingin mengirim permintaan koneksi ke akun mitra lain sehingga Anda dapat mengirimi mereka undangan keterlibatan peluang untuk kolaborasi peluang
 - **SUBSIDIARY:** Gunakan jenis ini ketika Anda ingin menghubungkan beberapa akun penjual AWS Marketplace yang Anda miliki ke akun “utama” yang telah Anda tautkan dengan APN atau di mana Anda terdaftar sebagai Mitra
- **ReceiverIdentifier:** ID profil mitra publik penerima (untuk jenis koneksi Opportunity Collaboration) atau ID AWS akun (untuk jenis koneksi Anak Perusahaan)

- Email: Alamat email kontak Anda untuk komunikasi
- Nama: Nama Anda sebagai pengirim
- Pesan: Deskripsi yang menjelaskan tujuan permintaan koneksi

Saat dibuat, undangan memasuki status Pending, menunggu tindakan dari penerima.

Important

Dengan mengirimkan undangan koneksi, Anda setuju untuk mengungkapkan ID AWS akun Anda kepada penerima jika mereka menerima undangan tersebut. Pengungkapan ini diperlukan untuk membangun koneksi tingkat akun.

Langkah 2: Pantau Undangan Anda yang Terkirim

Anda dapat melacak status undangan yang Anda kirim menggunakan `ListConnectionInvitations` tindakan dengan `ParticipantType` parameter yang disetel ke `SENDER`. Hal ini memungkinkan Anda untuk:

- Lihat semua undangan keluar
- Filter berdasarkan status (`PENDING`, `ACCEPTED`, `REJECTED`, `EXPIRED`)
- Filter berdasarkan jenis koneksi
- Periksa undangan khusus ke mitra lain

Langkah 3: Batalkan Undangan (Opsional)

Jika Anda perlu menarik undangan yang tertunda sebelum diterima, Anda dapat menggunakan `CancelConnectionInvitation` tindakan tersebut. Perhatikan bahwa:

- Anda hanya dapat membatalkan undangan dalam status Tertunda
- Setelah undangan diterima dan koneksi dibuat, itu tidak dapat dibatalkan
- Untuk mengakhiri koneksi yang dibuat, Anda harus menggunakan `CancelConnection` tindakan sebagai gantinya

Menerima dan Menanggapi Undangan Koneksi

Langkah 1: Daftar Undangan Masuk

Untuk melihat undangan koneksi yang Anda terima, gunakan `ListConnectionInvitations` tindakan dengan `ParticipantType` parameter yang disetel ke `RECEIVER`. Anda dapat memfilter berdasarkan:

- Tipe koneksi
- Status (`PENDING`, `ACCEPTED`, `REJECTED`)
- Pengenal pengirim

Langkah 2: Tinjau Detail Undangan

Gunakan `GetConnectionInvitation` tindakan untuk melihat detail lengkap undangan tertentu, termasuk:

- Nama pengirim dan email kontak
- Pesan undangan yang menjelaskan tujuan
- Jenis koneksi yang diminta
- Tanggal kedaluwarsa
- Informasi profil publik pengirim

Langkah 3: Terima atau Tolak Undangan

Setelah meninjau detail undangan, Anda memiliki dua opsi:

Opsi A: Terima Undangan

Gunakan `AcceptConnectionInvitation` tindakan untuk membuat koneksi. Ketika Anda menerima:

- Koneksi baru dibuat dalam status Aktif
- Status undangan berubah menjadi Diterima
- ID AWS akun Anda diungkapkan kepada pengirim
- Anda dapat mulai melakukan aktivitas bisnis yang diaktifkan oleh jenis koneksi

 Important

Persetujuan Penting: Dengan menerima undangan koneksi, Anda setuju untuk mengungkapkan ID AWS akun Anda kepada pengirim. Pengungkapan ini diperlukan untuk membangun koneksi tingkat akun.

Opsi B: Tolak Undangan

Gunakan `RejectConnectionInvitation` tindakan jika Anda tidak ingin membuat koneksi. Anda dapat secara opsional memberikan alasan penolakan. Setelah ditolak:

- Status undangan berubah menjadi Ditolak
- Tidak ada koneksi yang dibuat
- Undangan tidak akan muncul lagi di daftar undangan yang tertunda

Kedaluwarsa Otomatis

Jika tidak ada tindakan yang diambil dalam periode kedaluwarsa, sistem secara otomatis mentransisikan undangan ke status Kedaluwarsa. Undangan kedaluwarsa tidak dapat diterima atau ditolak.

Mengelola Koneksi Aktif

Melihat Koneksi Anda

Setelah koneksi dibuat, kedua belah pihak dapat melihat dan mengelolanya menggunakan tindakan berikut:

ListConnections: Ambil daftar koneksi Anda dengan opsi pemfilteran:

- Filter berdasarkan jenis dan status koneksi (mis., `OPPORTUNITY_COLLABORATION:ACTIVE`)
- Filter menurut pengenalan pihak lain
- Lihat informasi ringkasan untuk setiap koneksi

GetConnection: Ambil detail lengkap dari koneksi tertentu, termasuk:

- Semua jenis koneksi yang terkait dengan hubungan mitra ini

- Status setiap jenis koneksi (Aktif atau Terminasi)
- Informasi kontak dari undangan asli
- AWS ID akun dan ID profil publik kedua belah pihak
- Cap waktu pembuatan dan penghentian untuk setiap jenis koneksi

Menambahkan Jenis Koneksi ke Koneksi yang Ada

Jika Anda sudah memiliki koneksi aktif dengan pasangan dan ingin membangun jenis hubungan baru, Anda dapat mengirim undangan koneksi baru dengan jenis koneksi yang berbeda. Setelah diterima:

- Jenis koneksi baru ditambahkan ke koneksi yang ada
- Kedua jenis koneksi tetap dapat dikelola secara independen
- ID koneksi yang sama dipertahankan untuk hubungan

Note

Hanya ada satu koneksi aktif dari jenis tertentu antara dua akun dalam katalog tertentu kapan saja.

Mengakhiri Jenis Koneksi

Salah satu pihak dapat mengakhiri jenis koneksi tertentu menggunakan `CancelConnection` tindakan. Saat mengakhiri:

- Tentukan ID koneksi dan jenis koneksi yang akan dihentikan
- Berikan alasan untuk penghentian
- Status jenis koneksi berubah menjadi CANCELED
- Jenis koneksi lain dalam koneksi yang sama tetap tidak terpengaruh

Terminasi Koneksi Lengkap

Ketika semua jenis koneksi dalam koneksi dihentikan, koneksi itu sendiri bertransisi ke status Terminated. Koneksi yang sepenuhnya terputus:

- Tidak dapat diaktifkan kembali, tetapi Anda dapat mengirim permintaan koneksi baru untuk melanjutkan bisnis dengan akun itu
- Tetap dalam sistem untuk pencatatan sejarah
- Dapat dilihat menggunakan `GetConnection` API
- Membutuhkan undangan baru untuk membangun kembali hubungan

Memantau Acara Koneksi

Mitra harus memantau peristiwa terkait koneksi menggunakan Amazon EventBridge agar tetap mendapat informasi tentang:

- Undangan koneksi masuk baru
- Perubahan status undangan terkirim (ACCEPTED,,REJECTED) EXPIRED
- Pengakhiran koneksi yang diprakarsai oleh pihak lain

Setelah menerima acara, gunakan tindakan Get API yang sesuai untuk mengambil detail terbaru:

- `GetConnectionInvitation` untuk pembaruan undangan
- `GetConnection` untuk pembaruan koneksi

Kemungkinan Peristiwa

- Undangan Koneksi Dibuat: Memberi tahu akun penerima undangan koneksi masuk.
- Undangan Koneksi Dibatalkan: Memberi tahu akun penerima saat pengirim membatalkan undangan koneksi masuk.
- Undangan Koneksi Kedaluwarsa: Memberi tahu akun pengirim dan penerima saat undangan koneksi kedaluwarsa tanpa tindakan.
- Undangan Koneksi Ditolak: Memberi tahu akun pengirim ketika undangan koneksi mereka ditolak oleh penerima.
- Undangan Koneksi Diterima: Memberi tahu akun pengirim ketika undangan koneksi mereka diterima dan koneksi dibuat.
- Koneksi Dibatalkan: Memberi tahu kedua pihak yang terhubung saat semua koneksi terputus sepenuhnya.

Memahami Negara Koneksi

Negara Undangan Koneksi

Status	Deskripsi	Bisa Transisi Ke
PENDING	Status awal saat dibuat; menunggu tindakan penerima	ACCEPTED, REJECTED, EXPIRED, CANCELED
ACCEPTED	Penerima diterima; koneksi dibuat	Tidak ada (status terminal)
REJECTED	Penerima ditolak; berisi alasan penolakan opsional	Tidak ada (status terminal)
EXPIRED	System-expired karena tidak ada tindakan dalam periode kedaluwarsa	Tidak ada (status terminal)
CANCELED	Pengirim menarik undangan sebelum diterima	Tidak ada (status terminal)

Negara Jenis Koneksi

Status	Deskripsi
ACCEPTED	Jenis koneksi operasional; kegiatan bisnis terkait diaktifkan
CANCELED	Jenis koneksi diakhiri oleh salah satu pihak

Praktik Terbaik

1. Clear Communication: Berikan pesan terperinci dan jelas dalam undangan koneksi Anda yang menjelaskan tujuan dan kolaborasi yang diharapkan.
2. Respons Tepat Waktu: Pantau dan tanggapilah undangan koneksi dengan segera untuk menghindari kedaluwarsa otomatis.

3. Tinjauan Reguler: Tinjau koneksi aktif Anda secara berkala untuk memastikannya tetap relevan dengan kebutuhan bisnis Anda saat ini.
4. Pengakhiran yang Tepat: Saat mengakhiri hubungan bisnis, gunakan `CancelConnection` tindakan dengan alasan yang jelas untuk menjaga komunikasi profesional.
5. Pemantauan Acara: Siapkan `EventBridge` aturan untuk secara otomatis menerima pemberitahuan perubahan status koneksi, untuk membantu tetap mendapat informasi tentang pembaruan penting.
6. Manajemen Jenis Koneksi: Pertimbangkan jenis koneksi mana yang diperlukan sebelum membuat koneksi, karena setiap jenis memungkinkan kemampuan bisnis yang berbeda.
7. Kesadaran Keamanan: Ingatlah bahwa membuat koneksi mengungkapkan ID AWS akun antar pihak. Hanya terhubung dengan mitra terpercaya.

Jenis Koneksi dan Tujuannya

Jenis Koneksi	Tujuan	Kasus Penggunaan
OPPORTUNITY_COLLABORATION	Memungkinkan kolaborasi dalam peluang penjualan bersama dengan mitra lain	Bila Anda ingin mitra lain memiliki akses ke peluang Anda atau sebaliknya, gunakan jenis koneksi ini. Koneksi ini memungkinkan Anda mengirim undangan keterlibatan peluang ke mitra yang terhubung.
SUBSIDIARY	Menetapkan koneksi antara beberapa akun penjual AWS Marketplace Anda dan akun Mitra utama Anda	Gunakan saat Anda memiliki beberapa Akun Penjual AWS Marketplace yang ingin Anda sambungkan ke akun utama tempat Anda terdaftar sebagai mitra dan penjual.

Bekerja dengan Verifikasi Mitra

Verifikasi Mitra Pengelola

Verifikasi mitra merupakan prasyarat wajib untuk pendaftaran akun mitra. Sebelum mitra dapat membuat akun mitra, mereka harus menyelesaikan verifikasi bisnis dan proses verifikasi identitas. Verifikasi ini memvalidasi bahwa bisnis terdaftar secara hukum dan mengkonfirmasi identitas orang yang mendaftarkan akun.AWS

Mitra dapat memanfaatkan API verifikasi yang tersedia untuk memulai dan memantau proses verifikasi:

Selama Verifikasi

1. **Inisiasi Verifikasi Bisnis** — Mitra memulai verifikasi bisnis dengan menghubungi `StartVerification` API dengan rincian pendaftaran bisnis termasuk nama resmi, ID pendaftaran, kode negara, dan yurisdiksi pendirian.
2. **Inisiasi Verifikasi Identitas** — Mitra memulai verifikasi identitas dengan memanggil `StartVerification` API dengan informasi pendaftar. API menampilkan URL penyelesaian aman tempat pendaftar menyelesaikan alur kerja verifikasi identitas menggunakan identifikasi yang dikeluarkan pemerintah.
3. **Pemantauan Status Verifikasi** — Mitra menggunakan `GetVerification` API untuk mengambil status terkini dan detail proses verifikasi. API menampilkan jenis verifikasi, status, stempel waktu, dan detail spesifik verifikasi.
4. **Penyelesaian Verifikasi** — Verifikasi bisnis dan verifikasi identitas harus mencapai `SUCCEEDED` status sebelum mitra dapat melanjutkan ke pembuatan akun mitra. Verifikasi yang gagal atau kedaluwarsa harus diselesaikan sebelum pendaftaran akun.

Posting Verifikasi

1. **Pembuatan Akun Mitra** — Setelah verifikasi berhasil, mitra melanjutkan untuk membuat akun mitra mereka menggunakan `CreatePartner` API yang dijelaskan dalam dokumentasi Akun Mitra.
2. **Pengambilan Rekaman Verifikasi** — Mitra dapat mengambil detail verifikasi kapan saja menggunakan `GetVerification` API dengan ID verifikasi yang dikembalikan selama inisiasi.

Ringkasan API

1. **StartVerificationAPI**: Mitra memulai proses verifikasi dengan memanggil `StartVerification` API. Untuk verifikasi bisnis, API memvalidasi detail pendaftaran bisnis. Untuk verifikasi identitas, API menghasilkan URL penyelesaian terbatas waktu bagi pendaftar untuk menyelesaikan verifikasi identitas.
2. **GetVerificationAPI**: Setelah memulai verifikasi, mitra menggunakan `GetVerification` API untuk mengambil status dan detail verifikasi. API menampilkan status saat ini, stempel waktu, dan informasi spesifik verifikasi.

Menggunakan AWS API Penjualan Sentral Mitra

Referensi API Pusat AWS Mitra ini dirancang untuk membantu [AWS Mitra](#) mengintegrasikan sistem manajemen hubungan pelanggan (CRM) dengan Partner Central. API mengotomatiskan interaksi dengan Partner Central, yang membantu memastikan keterlibatan yang efektif dalam kegiatan bisnis bersama.

API menyediakan fungsionalitas AWS API standar. Akses dengan menggunakan [Tindakan](#) API atau dengan menggunakan AWS SDK yang disesuaikan dengan bahasa pemrograman atau platform Anda. Untuk informasi selengkapnya, lihat [Memulai AWS](#) dan [Alat untuk Dibangun AWS](#).

Fitur yang ditawarkan oleh AWS API Pusat Mitra

1. Manajemen peluang: Memfasilitasi pengelolaan peluang penjualan bersama dengan AWS menggunakan tindakan API seperti `CreateOpportunity`, `UpdateOpportunity`, `ListOpportunities`, `GetOpportunity`, dan `AssignOpportunity`.
2. AWS manajemen rujukan: Memfasilitasi menerima rujukan yang dibagikan dengan AWS menggunakan tindakan seperti `ListEngagementInvitations`, `GetEngagementInvitation`, `StartEngagementByAcceptingInvitation`, dan `RejectEngagementInvitation`.
3. Asosiasi entitas: Entitas terkait asosiasi seperti AWS Produk, Solusi Mitra, AWS Marketplace Solusi, AWS Marketplace Produk, dan Penawaran AWS Marketplace Pribadi dengan peluang menggunakan tindakan `AssociateOpportunity` dan `DisassociateOpportunity`.
4. Lihat detail AWS peluang: Gunakan `GetAWSOpportunitySummary` tindakan untuk mengambil ringkasan peluang waktu nyata yang terkait dengan AWS peluang Anda.
5. Solusi daftar: Menyediakan API daftar untuk mencantumkan solusi yang ditawarkan mitra `ListSolutions`.

6. Langganan acara: Mitra dapat berlangganan pembaruan real-time tentang peluang dengan mendengarkan acara seperti Peluang Dibuat, Peluang Diperbarui, Undangan Keterlibatan Diterima, Undangan Keterlibatan Ditolak, dan Undangan Keterlibatan yang Dibuat menggunakan Amazon EventBridge.

Wilayah dan titik akhir yang didukung

API AWS Partner Central tersedia di Wilayah AS Timur (Virginia N.).

Region	Titik akhir
us-east-1	partnercentral-selling.us-east-1.api.aws

Mitra dapat menguji dan memvalidasi integrasi API di lingkungan kotak pasir yang aman. Ini memungkinkan Anda menguji tindakan API tanpa memengaruhi data langsung. Untuk informasi selengkapnya, lihat [Penguji di kotak pasir untuk AWS API Penjualan Sentral Mitra](#).

Menjual entitas API

AWS Entitas Partner Central mewakili komponen bisnis utama yang digunakan dalam keterlibatan kerja sama antara mitra dan AWS. Entitas ini merangkum informasi yang terkait dengan peluang, solusi, produk, dan penawaran, memungkinkan kolaborasi dan pengelolaan kegiatan penjualan bersama yang lancar. Bagian berikut memberikan deskripsi entitas inti dalam API Penjualan Pusat Mitra.

Peluang

Peluang adalah penjualan atau kesepakatan potensial yang diidentifikasi dan dikejar secara aktif oleh bisnis. Ini adalah prospek atau prospek yang memenuhi syarat dengan kebutuhan khusus yang dapat ditangani oleh produk atau layanan perusahaan. Peluang biasanya dilacak dalam pipa penjualan atau sistem CRM dan membentuk fondasi pendapatan masa depan. Manajemen peluang yang efektif melibatkan pengasuhan prospek melalui proses penjualan, dari kualifikasi awal hingga penutupan. Untuk informasi selengkapnya, lihat [Bekerja dengan peluang](#) dan [tipe Data](#).

Solusi Mitra

Merupakan Solusi Mitra (disebut sebagai penawaran di AWS Partner Central), yang merupakan produk perangkat lunak atau praktik konsultasi yang dibuat dan disampaikan oleh AWS Mitra. Solusi

Mitra membantu pelanggan mengatasi tantangan bisnis tertentu atau mencapai tujuan tertentu dengan menggunakan AWS layanan. Untuk informasi lebih lanjut, lihat [Apa itu solusi?](#)

AWS produk

Merupakan AWS layanan atau produk tertentu. AWS menawarkan berbagai macam produk dan layanan yang dirancang untuk menyediakan solusi infrastruktur yang terukur, andal, dan hemat biaya. Mitra dapat memperoleh daftar AWS Produk terbaru dari [halaman impor massal di Partner Central](#) (Mulai Impor >AWS Produk dan Solusi). Untuk informasi selengkapnya, Anda dapat [mempelajari tentang AWS Produk](#) atau [melihat daftar semua AWS Produk](#).

AWS Marketplace penawaran pribadi

AWS Marketplace Penawaran pribadi adalah fitur yang memungkinkan AWS Marketplace penjual untuk menawarkan harga dan persyaratan tertentu kepada AWS pelanggan individu. Melalui penawaran pribadi, penjual dapat menegosiasikan harga khusus, jadwal pembayaran, dan persyaratan lisensi pengguna akhir. AWS pelanggan dapat memperoleh solusi perangkat lunak yang memenuhi persyaratan spesifik mereka, sementara juga mungkin mendapat manfaat dari persyaratan atau harga yang lebih menguntungkan dibandingkan dengan penawaran standar. Proses penawaran pribadi melibatkan penjual membuat penawaran unik dengan persyaratan yang disesuaikan, yang kemudian dibagikan secara pribadi dengan AWS pelanggan yang ditunjuk untuk ditinjau dan diterima. Untuk informasi selengkapnya, lihat [Penawaran pribadi di AWS Marketplace](#).

Undangan pertunangan

Undangan Keterlibatan mengacu pada permintaan formal dari mitra AWS untuk berkolaborasi pada rujukan tertentu. Hal ini memungkinkan AWS dan mitra untuk bekerja sama untuk mendorong peluang ke depan. Undangan dapat diterima atau ditolak oleh pasangan.

Topik

- [Bekerja dengan peluang Anda](#)
- [Bekerja dengan peluang dari AWS](#)
- [Bekerja dengan pembaruan peluang](#)
- [Bekerja dengan peluang multipartner](#)
- [Megasosiasikan, memisahkan, dan menetapkan peluang](#)
- [Bekerja dengan prospek Anda](#)
- [Bekerja dengan wawasan penyesuaian ukuran kesepakatan](#)
- [Praktik terbaik](#)

Bekerja dengan peluang Anda

Apa itu Peluang?

Selama tahap awal proses penjualan, perwakilan penjualan menilai apakah individu yang tertarik (disebut lead) memiliki potensi untuk menjadi pelanggan. Fase penilaian dan validasi ini disebut sebagai Kualifikasi. Setelah prospek dianggap memenuhi syarat dan dianggap memiliki probabilitas yang lebih tinggi untuk mengkonversi ke pelanggan, itu kemudian diklasifikasikan sebagai Peluang.

Bekerja dengan Peluang Anda

Mitra dapat mengelola peluang yang dibuat dalam sistem CRM mereka dan menyinkronkannya dengan AWS Partner Central menggunakan AWS Partner Central Selling API. Hal ini memungkinkan mitra untuk melacak dan mengelola peluang dari inisiasi hingga penutupan.

Menciptakan Peluang

Langkah pertama dalam mengelola peluang adalah menciptakan peluang menggunakan `CreateOpportunity` tindakan. Ini menciptakan peluang dengan `Lifecycle.ReviewStatus` set `kePending Submission`. Namun, kesempatan tersebut belum diserahkan AWS untuk validasi.

Setelah kesempatan dibuat, Anda harus mengaitkan setidaknya satu Solusi Mitra, AWS Marketplace Solusi, atau AWS Marketplace Produk dengan peluang menggunakan `AssociateOpportunity` tindakan tersebut. Tindakan ini dengan jelas mendefinisikan peluang apa yang coba dijual. Anda dapat melihat daftar lengkap solusi yang tersedia di akun Anda menggunakan `ListSolutions` API, dan Anda dapat mengaitkan antara satu dan sepuluh solusi.

Secara opsional, Anda juga dapat mengaitkan AWS Produk yang relevan dengan peluang menggunakan `AssociateOpportunity` tindakan. Langkah ini membantu tim AWS penjualan memahami AWS produk apa yang diharapkan untuk dijual bersamaan dengan peluang.

Setelah solusi atau produk yang diperlukan dikaitkan dan, secara opsional, AWS Produk ditautkan, Anda dapat memulai keterlibatan pada peluang dengan menggunakan `StartEngagementFromOpportunityTask` tindakan. Ini adalah saat Anda mengirimkan kesempatan untuk memulai pertunangan.

Proses Peninjauan

Setelah memulai keterlibatan pada kesempatan menggunakan `StartEngagementFromOpportunityTask` tindakan, peluang memasuki fase AWS validasi, dan

`Lifecycle.ReviewStatus` diatur ke `Submitted`. Tidak ada perubahan yang dapat dilakukan pada kesempatan sampai proses peninjauan selesai.

Selama fase validasi ini, AWS pastikan bahwa detail peluang akurat dan lengkap. Sementara validasi sedang berlangsung, `Lifecycle.ReviewStatus` diatur ke `In-Review`.

Jika ada perubahan atau detail tambahan yang diperlukan dari mitra, AWS tetapkan `Lifecycle.ReviewStatus` ke `Action Required`, dan setiap pembaruan yang diperlukan dikomunikasikan melalui `Lifecycle.ReviewComments` bidang.

Setelah peluang melewati validasi, `Lifecycle.ReviewStatus` perubahan menjadi `Approved`, membuat peluang siap untuk kegiatan co-selling.

Mitra harus memantau `Opportunity Updated` acara menggunakan Amazon EventBridge. Ini akan memberi tahu mereka tentang perubahan status atau umpan balik dari AWS. Setelah menerima acara, mitra dapat menggunakan tindakan `GetOpportunity` API untuk mengambil detail peluang terbaru dan memverifikasi `Lifecycle.ReviewStatus` bidang.

Menyelesaikan Masalah Validasi

Jika `Lifecycle.ReviewStatus` disetel ke `Action Required`, mitra perlu mengatasi masalah yang disorot oleh AWS. Untuk mengatasinya, mitra dapat memperbarui peluang menggunakan tindakan `UpdateOpportunity` API.

Selama `Action Required` negara bagian, hanya bidang tertentu yang dapat diedit. Bidang ini meliputi:

1. `Customer.Account.Address.City`
2. `Customer.Account.Address.Country`
3. `Customer.Account.Address.PostalCode`
4. `Customer.Account.Address.StateOrRegion`
5. `Customer.Account.Address.StreetAddress`
6. `Customer.Account.WebsiteUrl`
7. `LifeCycle.TargetCloseDate`
8. `Project.ExpectedCustomerSpend.Amount`
9. `Project.ExpectedCustomerSpend.Currency`
10. `Project.CustomerBusinessProblem`
11. `PartnerOpportunityIdentifier`

Setelah membuat perubahan yang diperlukan, peluang memasuki kembali fase validasi, dan proses berulang sampai peluang Lifecycle.ReviewStatus diatur ke atau. Approved Disqualified

Post-Approval Pembaruan

Setelah ada kesempatan Approved, mitra dapat terus memperbarui peluang sesuai kebutuhan menggunakan UpdateOpportunity tindakan, memfasilitasi kegiatan co-selling yang mulus.

Mitra harus terus memantau Opportunity Updated peristiwa melalui Amazon EventBridge untuk tetap diperbarui pada setiap perubahan. Untuk informasi selengkapnya tentang melacak AWS pembaruan, lihat bagian “Bekerja dengan pembaruan peluang”. Mitra juga dapat memperbarui bidang tertentu berdasarkan aturan validasi bisnis.

Bekerja dengan peluang dari AWS

1. Menerima AWS Peluang

Peluang dibagi dengan mitra ketika seorang eksekutif AWS penjualan melampirkan mitra ke peluang dalam sistem AWS CRM. Ini disebut sebagai AWS Peluang, berbeda dari peluang yang dibuat di akun mitra sendiri.

Ketika AWS Peluang memiliki mitra yang terpasang, AWS buat Undangan Keterlibatan yang berisi subset data dari AWS Peluang. Mitra akan menerima acara Undangan Keterlibatan yang Dibuat.

2. Meninjau Undangan Keterlibatan

Undangan Keterlibatan berisi informasi penting seperti

`Project.TitleProject.CustomerUseCase,Lifecycle.Stage,Project.CustomerBusinessProb` dan beberapa bidang tambahan. Mitra dapat menggunakan data ini untuk memutuskan apakah akan mengejar peluang.

Namun, bidang berikut dari AWS Peluang tidak termasuk dalam Undangan Keterlibatan:

```
Customer.Account.Address.StreetAddress
Customer.Account.Address.City
Customer.Account.Address.PostalCode
Customer.Contact
Customer.Account.AWSAccountId
Project.OtherSolutionDescription
RelatedEntityIdentifiers
```

3. Menangani Undangan Keterlibatan

Setelah menerima acara yang Dibuat Undangan Keterlibatan, mitra dapat menggunakan `GetEngagementInvitation` tindakan tersebut untuk mengambil detail AWS Peluang.

Jika pasangan memutuskan untuk tidak mengejar kesempatan, mereka dapat menolak undangan menggunakan `RejectEngagementInvitation` tindakan, bersama dengan yang diperlukan `RejectionReason`. Setelah ditolak, akses ke peluang hilang.

Untuk menerima undangan dan melanjutkan, mitra harus menggunakan `StartEngagementByAcceptingInvitationTask` tindakan tersebut. Ini adalah tindakan asinkron yang secara berurutan melakukan tugas-tugas berikut:

1. Menerima Undangan Keterlibatan.
2. Menciptakan peluang baru di akun mitra menggunakan data dari AWS Peluang.
3. Termasuk rincian tambahan yang diperlukan untuk mengidentifikasi pelanggan di akun mitra.

Setelah selesai, acara `Opportunity Created` dipicu dengan ID Peluang yang sesuai. Mitra kemudian dapat menggunakan ID ini dalam `GetOpportunity` tindakan untuk mengambil rincian lengkap dari peluang.

Jika mitra tidak menggunakan acara, mereka dapat menelepon `StartEngagementByAcceptingInvitationTask` lagi dengan muatan yang sama untuk memeriksa status terbaru.

4. Mengelola AWS Kesempatan Post-Acceptance

Setelah peluang ada di akun mitra, itu dapat dikelola dan diperbarui seperti peluang lain dalam sistem. Mitra dapat menggunakan tindakan seperti `UpdateOpportunity` membuat perubahan yang diperlukan.

Untuk detail selengkapnya tentang cara melacak AWS pembaruan dan mengelola pembaruan peluang, lihat [Bekerja dengan pembaruan peluang](#) bagian ini.

Bekerja dengan pembaruan peluang

Memperbarui peluang

Mitra dapat menggunakan `UpdateOpportunity` tindakan untuk memperbarui peluang, dengan aturan khusus yang mengatur bidang mana yang diperbarui, dan kapan mereka diperbarui:

1. Pembaruan tidak dapat dilakukan jika `Lifecycle.ReviewStatus` ada `Submitted` atau `In-Review`.
2. Sebelum pengiriman, mitra dapat melakukan pembaruan, tetapi tidak AWS akan memprosesnya kecuali `StartEngagementFromOpportunityTask` tindakan tersebut dipanggil.
3. Ketika ada peluang `Submitted` atau `In-review` status, semua pembaruan diblokir.
4. Jika peluang dalam `Action Required` status, AWS buka pilih bidang untuk pembaruan. Bidang ini meliputi:
 - `Customer.Account.Address.City`
 - `Customer.Account.Address.Country`
 - `Customer.Account.Address.PostalCode`
 - `Customer.Account.Address.StateOrRegion`
 - `Customer.Account.Address.StreetAddress`
 - `Customer.Account.WebsiteUrl`
 - `LifeCycle.TargetCloseDate`
 - `Project.ExpectedCustomerSpend.Amount`
 - `Project.ExpectedCustomerSpend.CurrencyCode`
 - `Project.ExpectedCustomerSpend.EstimationURL`
 - `Project.ExpectedCustomerSpend.Frequency`
 - `Project.ExpectedCustomerSpend.TargetCompany`
 - `Project.CustomerBusinessProblem`
 - `PartnerOpportunityIdentifier`
5. Setelah proses peninjauan (yaitu, kapan `Lifecycle.ReviewStatus` disetel ke `Approved`), bidang berikut tidak dapat diperbarui:
 - `Customer.Account.Address.Country`
 - `Customer.Account.Address.PostalCode`
 - `Customer.Account.Industry`
 - `Customer.Account.WebsiteUrl`
 - `Project.CustomerBusinessProblem`
 - `PartnerOpportunityIdentifier`

6. Untuk semua bidang lainnya, pembaruan dapat dilakukan menggunakan `UpdateOpportunity` tindakan. Namun, pembatasan tambahan mungkin berlaku berdasarkan aturan bisnis untuk program atau jenis peluang tertentu. Untuk detail selengkapnya, lihat validasi tingkat lapangan.
7. Untuk semua pembaruan yang dilakukan melalui UI dan API, acara `Opportunity Updated` dihasilkan.

Menerima pembaruan dari AWS pada peluang

AWS biasanya memperbarui AWS peluang, dan setiap kali pembaruan dibuat, acara `Opportunity Updated` dihasilkan.

Untuk mengambil pembaruan terbaru dari AWS, mitra perlu memanggil dua tindakan terpisah

1. `GetOpportunity` untuk mengambil rincian peluang mitra.
2. `GetAWSOpportunitySummary` untuk mengambil ringkasan real-time dari data peluang AWS

Sebagian besar pembaruan rutin dari AWS akan tersedia melalui `GetAWSOpportunitySummary` respons. Namun, kadang-kadang AWS dapat memperbarui atribut dalam peluang mitra secara langsung.

Untuk menggunakan pembaruan dari AWS

1. Minta `GetAWSOpportunitySummary` tindakan untuk mengambil detail terbaru dari Peluang AWS
2. Jika perubahan perlu tercermin dalam peluang mitra, gunakan `UpdateOpportunity` tindakan untuk menyalin data yang relevan ke peluang mitra.

Mitra dapat memilih untuk mengotomatiskan proses ini sebagai mekanisme pembaruan langsung atau menerapkan proses tinjauan manual untuk memvalidasi dan memperbarui data.

Bekerja dengan peluang multipartner

AWS Partner s dapat bekerja sama pada peluang dengan AWS sebagai peserta aktif.

Topik

- [Keterlibatan dan snapshot](#)

- [Membuat kebijakan khusus untuk ResourceSnapshotJobRole](#)
- [Mengundang mitra ke kesempatan](#)
- [Mengambil detail undangan keterlibatan](#)
- [Menanggapi undangan pertunangan](#)
- [Meninjau dan memperbarui peluang multipartner](#)
- [Menggunakan snapshot untuk menerima pembaruan mitra](#)
- [Melihat anggota keterlibatan](#)
- [Memantau status pekerjaan snapshot sumber daya](#)

Keterlibatan dan snapshot

Keterlibatan adalah sumber daya yang dimiliki oleh AWS untuk kolaborasi antara beberapa AWS Partner akun dan AWS peluang pelanggan. Keterlibatan memfasilitasi berbagi informasi yang aman di antara mitra dan kolaborasi sambil mempertahankan kepemilikan dan kontrol peluang individu. Keterlibatan mencakup peluang yang berasal dari keduanya AWS dan mitra, dengan AWS sebagai peserta aktif. Mitra dapat mengundang AWS atau mitra lain untuk bergabung dengan keterlibatan menggunakan `EngagementInvitations`. Ketika mitra yang diundang menerima, mereka menerima kesempatan mereka sendiri dalam keterlibatan yang sama.

Anggota keterlibatan berbagi kemajuan melalui snapshot—titik waktu, salinan yang tidak dapat diubah dari bidang tertentu dari sumber daya yang mendasarinya seperti peluang. Snapshot dibuat dalam keterlibatan dan dibagikan dengan anggota. Saat sumber daya yang mendasarinya berubah, pemilik dapat membuat revisi snapshot baru untuk mencerminkan pembaruan. AWS menyediakan pekerjaan snapshot — pekerjaan milik pelanggan yang secara otomatis membuat revisi baru saat sumber daya berubah. Setelah dibagikan, snapshot tetap dapat diakses secara permanen oleh anggota keterlibatan.

Membuat kebijakan khusus untuk ResourceSnapshotJobRole

Berkolaborasi dalam peluang multi-mitra membutuhkan berbagi snapshot peluang Anda dengan mitra lain dalam keterlibatan. Untuk mempertahankan akses ke detail peluang terbaru, Anda perlu membuat peran khusus yang disebut `ResourceSnapshotJobRole`. Peran ini memungkinkan sistem untuk membuat snapshot peluang Anda atas nama Anda dan mengambil snapshot dari mitra lain dalam keterlibatan yang sama. Tanpa peran ini, pembaruan apa pun yang Anda buat untuk peluang Anda tidak akan terlihat oleh mitra lain dalam keterlibatan, dan mereka akan terus melihat snapshot data peluang Anda yang sudah ketinggalan zaman.

 Note

Anda dapat menggunakan nama selain `ResourceSnapshotJobRole`. Jika Anda menggunakan nama yang berbeda, ganti semua contoh `ResourceSnapshotJobRole` dalam kebijakan berikut dengan nama kebijakan Anda.

Untuk membuat peran ini, buka konsol IAM Anda dan buat `ResourceSnapshotJobRole` peran dengan kebijakan kepercayaan khusus berikut:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "resource-snapshot-job.partnercentral-selling.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Setelah membuat peran ini, Anda perlu melampirkan kebijakan [AWSPartnerCentralSellingResourceSnapshotJobExecutionRolePolicy](#) AWS terkelola, atau melampirkan kebijakan izin berikut:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "partnercentral:CreateResourceSnapshot"
      ],

```

```

        "Resource": [
            "arn:aws:partnercentral:*::catalog/AWS/engagement/*"
        ],
    },
    {
        "Effect": "Allow",
        "Action": [
            "partnercentral:GetOpportunity"
        ],
        "Resource": [
            "arn:aws:partnercentral:*:111122223333:catalog/AWS/opportunity/*"
        ]
    }
]
}

```

Ganti {account} dengan ID akun AWS Anda.

Setelah Anda membuat ResourceSnapshotJobRole peran dan melampirkan izin, Anda perlu mengatur ResourceSnapshotJobRole untuk organisasi Anda. Gunakan [PutSellingSystemSettings](#) tindakan untuk menetapkan peran yang Anda buat sebagai ResourceSnapshotJobRole untuk perusahaan Anda (diidentifikasi oleh AWS akun Anda):

```

aws-cli/2.13.5 Python/3.11.4 Linux/4.14.255-314-253.539.amzn2.x86_64
Host: partner-central.aws.amazon.com
Content-Type: application/json

{
  "ResourceSnapshotJobRoleIdentifier": "arn:aws:iam::{account}:role/
ResourceSnapshotJobRole"
}

```

Ganti {account} dengan ID AWS akun Anda, dan ResourceSnapshotJobRole dengan nama peran yang Anda buat. Peran ini akan secara implisit diasumsikan oleh pekerjaan snapshot sumber daya untuk mengakses peluang Anda dan membuat snapshot untuk dibagikan dengan mitra lain dalam keterlibatan.

Mengundang mitra ke kesempatan

Mitra dapat berkolaborasi pada peluang yang berasal dari kedua mitra dan AWS dengan memulai undangan keterlibatan. Anda dapat mengundang maksimal sembilan mitra ke suatu kesempatan.

Untuk mengundang mitra ke kesempatan

1. Ikuti langkah-langkah dalam [Menemukan dan terhubung dengan mitra](#) di Panduan Penjualan Pusat AWS Mitra. Anda harus terhubung dengan mitra sebelum Anda dapat mengundang mereka ke peluang. Koneksi ini memberikan akses timbal balik ke ID akun masing-masing, memastikan undangan mencapai akun mitra yang dimaksud.
2. Gunakan [StartEngagementFromOpportunityTask](#) tindakan untuk menciptakan keterlibatan dan mengaitkan peluang dengannya. Tindakan asinkron ini melakukan tugas-tugas berikut secara berurutan:
 1. Menciptakan keterlibatan.
 2. Mengaitkan kesempatan dengan keterlibatan.
 3. Memanggil [CreateEngagementInvitation](#) tindakan untuk AWS.
 4. Mengajukan kesempatan untuk AWS.
 5. Mulai ResourceSnapshotJob untuk membuat snapshot.
3. Undang mitra lain untuk bergabung.
 - a. Untuk mendapatkan ID keterlibatan yang terkait dengan peluang Anda, gunakan [ListEngagementFromOpportunityTasks](#) tindakan, yang difilter oleh yang TaskIdentifier dikembalikan pada langkah sebelumnya.
 - b. Gunakan ID keterlibatan mitra penerima dan ID akun untuk mengirim undanganCreateEngagementInvitation.

Undangan yang berhasil mengirimkan undangan pertunangan yang dibuat acara ke mitra penerima.

Mengambil detail undangan keterlibatan

Saat Anda menerima acara yang dibuat undangan keterlibatan, gunakan [GetEngagementInvitation](#) tindakan untuk mengambil detail undangan. Tanggapan tersebut mencakup informasi penting tentang peluang pelanggan yang terkait dengan keterlibatan.

Tinjau detail undangan, khususnya InvitationMessage, Project.Title, Project.CustomerUseCase, dan Project.CustomerBusinessProblem bidang. Informasi ini memberikan konteks tentang peluang pelanggan dan harapan mitra yang mengundang untuk kolaborasi.

Gunakan detail ini untuk mengevaluasi apakah Anda ingin mengejar peluang dengan menerima atau menolak undangan pertunangan.

Menanggapi undangan pertunangan

Ketika mitra mengirimkan Anda undangan pertunangan, itu akan menciptakan acara yang dibuat undangan keterlibatan yang memberi tahu Anda tentang undangan tersebut. Anda memiliki opsi untuk menerima atau menolak undangan. Keputusan ini menentukan keterlibatan Anda dalam peluang multipartner.

Untuk menolak undangan pertunangan:

Gunakan [RejectEngagementInvitation](#) tindakan untuk menolak undangan. Anda harus memberikan `RejectionReason` parameter yang menjelaskan keputusan Anda. Setelah ditolak, Anda kehilangan akses ke detail undangan, dan acara yang ditolak undangan pertunangan memberi tahu mitra pengirim bahwa Anda telah menolak undangan mereka.

Untuk menerima undangan pertunangan:

Untuk menerima undangan dan melanjutkan dengan peluang multipartner, gunakan [StartEngagementByAcceptingInvitationTask](#) tindakan. Tindakan asinkron ini melakukan tugas-tugas berikut secara berurutan:

1. Menerima undangan pertunangan.
2. Menciptakan peluang baru di akun mitra Anda menggunakan data dari peluang mitra pengirim. Anda akan menerima acara yang dibuat kesempatan.
3. Termasuk rincian tambahan yang diperlukan untuk mengidentifikasi pelanggan di akun Anda.
4. Menerbitkan acara yang diterima undangan keterlibatan yang memberi tahu mitra bahwa Anda telah menerima undangan dan telah ditambahkan ke pertunangan.

Undangan pertunangan kedaluwarsa

Jika Anda tidak menanggapi undangan pertunangan dalam waktu lima belas hari, itu akan berakhir, dan Anda tidak dapat berpartisipasi dalam peluang multipartner. Acara kedaluwarsa undangan pertunangan memberi tahu mitra pengirim bahwa undangan telah kedaluwarsa.

Note

Jika Anda masih ingin berkolaborasi, mitra pengirim harus memulai undangan keterlibatan baru.

Meninjau dan memperbarui peluang multipartner

Ketika mitra memulai keterlibatan pada suatu kesempatan, status peninjauan peluang yang baru dibuat di akun mitra penerima ditentukan oleh status peluang mitra pengirim.

Peluang dengan status ulasan yang dikirimkan:

Jika peluang mitra pengirim telah `Lifecycle.ReviewStatus` ditetapkan `Submitted`, proses berikut terjadi:

1. Peluang baru yang dibuat di akun mitra penerima akan `Lifecycle.ReviewStatus` ditetapkan `Submitted`.
2. `SubmittedStatus` tetap sampai kesempatan mitra pengirim adalah `Approved`.
3. Setelah peluang mitra pengirim divalidasi dan `Lifecycle.ReviewStatus` diatur ke `Approved`, peluang mitra lain akan secara otomatis mewarisi status. `Approved`

Proses ini memastikan rincian peluang yang konsisten di seluruh peluang multipartner, menghilangkan kebutuhan untuk beberapa ulasan.

Setelah selesai, acara yang dibuat peluang dipicu dengan ID peluang yang sesuai. Mitra dapat menggunakan ID ini dengan `GetOpportunity` tindakan untuk mengambil detail peluang lengkap.

Peluang dengan status ulasan yang disetujui:

Jika mitra pengirim memulai keterlibatan pada kesempatan dengan `Lifecycle.ReviewStatus` set `toApproved`, proses berikut terjadi:

1. Peluang baru yang dibuat di akun mitra penerima akan `Lifecycle.ReviewStatus` ditetapkan `Approved`.
2. Acara yang dibuat peluang dipicu dengan ID peluang yang sesuai.
3. Mitra dapat menggunakan [GetOpportunity](#) tindakan dengan ID peluang untuk mengambil detail peluang penuh.

Namun, peluang yang disetujui mungkin tidak memiliki beberapa detail spesifik mitra yang tidak disalin dari peluang mitra pengirim. Mitra penerima harus memperbarui detail ini menggunakan [UpdateOpportunity](#) tindakan ketika tahap peluang diperbarui ke `Qualified` atau tahap selanjutnya:

- `Project.CustomerUseCase`

- `Project.DeliveryModels`
- `Solution`
- `PrimaryNeedsFromAws`
- `LifeCycle.TargetCloseDate`
- `Project.ExpectedCustomerSpend.Amount`
- `Project.ExpectedCustomerSpend.Currency`
- `Marketing.source`
- `Marketing.AwsFundingUsed`

Setelah peluang dibuat di akun mitra penerima, itu dapat dikelola dan diperbarui seperti peluang lain dalam sistem mitra. Mitra dapat menggunakan [UpdateOpportunity](#) tindakan untuk membuat perubahan atau memberikan informasi lebih lanjut tentang keterlibatan mereka dalam kesempatan.

Menggunakan snapshot untuk menerima pembaruan mitra

Dalam suatu keterlibatan, mitra mempertahankan dan memperbarui peluang mereka secara mandiri. Ketika seseorang merevisi sumber daya keterlibatan, seperti menambahkan peluang, snapshot sumber daya keterlibatan yang dibuat akan dipublikasikan.

Untuk tetap mendapat informasi tentang perubahan dan mengakses informasi terkini dari peluang mitra lain, Anda dapat menggunakan tindakan berikut:

- [ListEngagementResourceAssociations](#)— Gunakan tindakan ini untuk mengambil ID keterlibatan yang terkait dengan peluang.
- [ListResourceSnapshots](#)— Gunakan tindakan ini untuk mengambil daftar lengkap semua snapshot peluang yang terkait dengan keterlibatan, memberikan ikhtisar snapshot yang tersedia di semua mitra.
- [GetResourceSnapshot](#)— Gunakan tindakan ini untuk mendapatkan ringkasan real-time dari snapshot peluang tertentu, memungkinkan Anda untuk melihat informasi terbaru tanpa langsung mengakses peluang mitra lain.

Jika Anda mengidentifikasi perubahan atau pembaruan yang relevan dari peluang mitra lain yang harus tercermin dalam diri Anda sendiri, gunakan [UpdateOpportunity](#) tindakan tersebut. Tindakan ini memfasilitasi secara selektif memasukkan data terkait ke dalam peluang Anda, memastikan keselarasan dan konsistensi di seluruh keterlibatan.

Melihat anggota keterlibatan

Keterlibatan pada peluang multimitra dapat memiliki hingga sepuluh mitra. Setiap kali mitra baru menerima undangan keterlibatan, acara tambahan anggota keterlibatan dipublikasikan, memberi tahu semua anggota saat ini tentang perubahan tersebut.

Untuk melihat semua anggota yang berkolaborasi dalam suatu keterlibatan, ikuti langkah-langkah berikut:

1. Gunakan [ListEngagementResourceAssociation](#) tindakan untuk mengambil ID keterlibatan yang terkait dengan peluang.
2. Berikan ID dari langkah 1 hingga [ListEngagementMembers](#) tindakan untuk mengambil rincian mitra dari anggota keterlibatan.

Note

Hanya anggota pertunangan yang dapat meminta `ListEngagementMembers` tindakan tersebut.

Memantau status pekerjaan snapshot sumber daya

Saat bekerja dengan peluang multimitra, Anda harus membuat peran yang memungkinkan Anda membuat snapshot peluang Anda untuk mitra lain dan mengambil snapshot peluang dari mitra lain dalam suatu keterlibatan. Tanpa peran ini, pembaruan apa pun yang Anda buat untuk peluang Anda tidak akan tersedia untuk mitra lain dalam keterlibatan, dan mereka akan terus melihat cuplikan peluang Anda yang sudah ketinggalan zaman.

Untuk melacak status pekerjaan snapshot sumber daya yang terkait dengan peluang multimitra dalam suatu keterlibatan, ikuti langkah-langkah berikut:

1. Gunakan [ListEngagementResourceAssociation](#) tindakan untuk mengambil ID keterlibatan yang terkait dengan peluang.
2. Berikan ID dari langkah 1 hingga [ListResourceSnapshotJob](#) tindakan untuk menghasilkan daftar semua pekerjaan snapshot yang dimiliki oleh penelepon dalam keterlibatan. Ambil ID pekerjaan dari respons.
3. Berikan ID pekerjaan ke [GetResourceSnapshotJob](#) tindakan untuk melacak status pekerjaan dan melihat apakah itu berjalan.

ResourceSnapshotJobOperasi ini menerbitkan metrik ke Amazon CloudWatch untuk operasi asinkron. CloudWatch memproses data menjadi metrik yang dapat dibaca, mendekati waktu nyata untuk membantu Anda memantau kinerja dan kesehatan layanan.

Metrik berikut tersedia:

- Kesalahan — Menghitung masalah internal selama pelaksanaan pekerjaan. Gunakan metrik ini untuk memantau stabilitas layanan dan mengatur alarm untuk ambang kesalahan.
- Kesalahan — Melacak masalah terkait pelanggan selama pemrosesan pekerjaan. Metrik ini membantu mengidentifikasi masalah dengan input atau konfigurasi pelanggan.
- Doa - Merupakan jumlah total pemanggilan pekerjaan, termasuk upaya yang berhasil dan gagal. Gunakan ini untuk melacak tren penggunaan layanan dari waktu ke waktu.

 Note

Metrik dilaporkan CloudWatch hanya jika ada aktivitas dalam ResourceSnapshotJob layanan. Jika tidak ada pekerjaan atau tidak ada data untuk metrik tertentu, metrik itu tidak dilaporkan.

Untuk memastikan kelancaran ResourceSnapshotJob operasi:

- Gunakan metrik untuk memverifikasi bahwa layanan berfungsi seperti yang diharapkan.
- Buat CloudWatch alarm untuk memantau metrik dan memicu tindakan (seperti mengirim notifikasi) saat metrik melebihi rentang yang dapat diterima.
- Siapkan pemantauan proaktif untuk mengidentifikasi dan menyelesaikan masalah.

Untuk informasi selengkapnya tentang penggunaan CloudWatch metrik, lihat [Panduan CloudWatch Pengguna Amazon](#).

Mengasosiasikan, memisahkan, dan menetapkan peluang

Peluang dapat dikaitkan atau dilepaskan dengan Solusi Mitra,AWS Produk,AWS Marketplace Solusi,AWS Marketplace Produk,AWS Marketplace Penawaran, dan Set AWS Marketplace Penawaran pada setiap tahap siklus hidup peluang.

Mengaitkan peluang dengan entitas lain

Entitas terkait diambil dari `GetOpportunity` metode dalam `RelatedEntityIdentifiers` objek. `RelatedEntityIdentifiers` dapat diperbarui menggunakan `AssociateOpportunity`. Perhatikan bahwa bidang ini tidak dapat diperbarui menggunakan `CreateOpportunity` metode `UpdateOpportunity` or.

Solusi

Sebelum keterlibatan dengan AWS mulai menggunakan `StartEngagementFromOpportunityTask` tindakan, wajib untuk mengaitkan setidaknya satu dan hingga sepuluh Solusi Mitra. AWS Referral mungkin atau mungkin tidak mengandung Solusi Mitra.

Untuk melihat solusi yang ada, gunakan `ListSolutions` tindakan.

Mitra dapat membuat, memperbarui, dan mengelola solusi mereka di Build bagian [AWS Partner Central](#).

Important

Untuk memajukan tahap peluang ke Comited atau Diluncurkan, Anda harus mengaitkan solusi atau daftar AWS Marketplace produk yang valid dengan peluang tersebut.

Note

Solusi Mitra Lama (S-XXXXpengidentifikasi) yang menggunakan tipe `Solutions` entitas masih didukung. Namun, kami merekomendasikan untuk mengaitkan tipe `AwsMarketplaceSolutions` entitas dengan peluang karena tipe `Solution` entitas akan tidak digunakan lagi di masa mendatang.

AWS produk

Hingga 20 AWS Produk dapat dikaitkan dengan peluang. Untuk melihat daftar AWS Produk yang tersedia, gunakan daftar [AWS Produk yang dihosting di GitHub](#). Asosiasi dengan AWS Produk secara eksklusif dilakukan dengan menggunakan `AssociateOpportunity` tindakan. Untuk mengganti atau menghapus Produk, gunakan `DisassociateOpportunity` tindakan.

Solusi

Peluang dapat dikaitkan dengan solusi dari akun penjual Anda. Gunakan `AssociateOpportunity` tindakan dengan `RelatedEntityType` set ke `AwsMarketplaceSolutions`.

Entitas harus dalam status Publik atau Terbatas. Layanan menolak draft atau entitas yang tidak aktif.

Untuk mengaitkan solusi, ARN diperlukan. Contoh berikut menunjukkan format ARN untuk solusi:

```
arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/Solution/soln-ab1c2de3fgh4i
```

Anda juga dapat mengaitkan solusi dari akun anak perusahaan yang terhubung ke AWS Marketplace akun utama Anda melalui Koneksi Akun Mitra. Masukkan ARN lengkap, termasuk ID akun anak perusahaan.

`ListSolutionsTindakan` kembali `AwsMarketplaceSolutionArn` pada setiap ringkasan solusi, dan mendukung pemfilteran oleh ARN.

AWS Marketplace Produk

Peluang dapat dikaitkan dengan AWS Marketplace Produk dari akun penjual Anda. Gunakan `AssociateOpportunity` tindakan dengan `RelatedEntityType` set ke `AwsMarketplaceProducts`.

Entitas harus dalam status Publik atau Terbatas. Layanan menolak draft atau entitas yang tidak aktif.

Untuk mengaitkan suatu produk, ARN diperlukan. ARN mencakup jenis produk (misalnya,, `AmiProductSaaSProduct`, `ContainerProduct`). Contoh berikut menunjukkan format ARN untuk produk AMI:

```
arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/AmiProduct/prod-ab1c2de3fgh4i
```

Anda juga dapat mengaitkan produk dari akun anak perusahaan yang terhubung ke AWS Marketplace akun utama Anda melalui Koneksi Akun Mitra. Masukkan ARN lengkap, termasuk ID akun anak perusahaan.

AWS Marketplace penawaran dan penawaran set

AWS Marketplace penawaran

Peluang dapat dikaitkan dengan Penawaran AWS Marketplace Pribadi. Untuk melihat penawaran yang tersedia, gunakan `ListEntities` dari API AWS Marketplace Katalog. Saat ini, Anda hanya dapat mengaitkan penawaran dari akun AWS Marketplace Penjual yang ditautkan ke AWS Partner Central.

Untuk mengaitkan penawaran pribadi ARN diperlukan. Sampel:

```
arn:aws:aws-marketplace:us-east-1:123123123123:AWSMarketplace/Offer/offer-dtn3example1tg
```

AWS Marketplace set penawaran

Peluang juga dapat dikaitkan dengan Set AWS Marketplace Penawaran. Set penawaran memungkinkan pengelompokan beberapa penawaran pasar terkait bersama untuk pengemasan solusi yang komprehensif. Untuk melihat set penawaran yang tersedia, gunakan `ListEntities` dari API AWS Marketplace Katalog.

Untuk mengaitkan set penawaran, ARN diperlukan. Sampel:

```
arn:aws:aws-marketplace:us-east-1:123123123123:AWSMarketplace/OfferSet/offerset-dtn3example1tg
```

Penting: Peluang dapat dikaitkan dengan salah satu Penawaran ATAU satu Set Penawaran, tetapi tidak keduanya. Hanya satu dari jenis entitas ini yang dapat dikaitkan dengan peluang pada suatu waktu.

Memutuskan peluang dari entitas lain

Gunakan `DisassociateOpportunity` tindakan untuk memutuskan tautan peluang dari jenis entitas berikut:

- Solusi
- AWS Produk
- AWS Marketplace Solusi
- AWS Marketplace Produk
- AWS Marketplace Penawaran
- AWS Marketplace Set Penawaran

Bergantung pada keadaan peluang, aturan validasi yang berbeda berlaku ketika Anda memutuskan tautan objek terkait ini.

Menetapkan peluang

Gunakan `AssignOpportunity` untuk mengubah pemilik peluang. Anda dapat mengatur salah satu pengguna Pusat Mitra Anda untuk menjadi pemilik peluang.

Bekerja dengan prospek Anda

Apa itu Lead?

Dalam penjualan bisnis-ke-bisnis (B2B), prospek mewakili pelanggan potensial yang telah menyatakan minatnya pada produk atau layanan perusahaan tetapi belum sepenuhnya memenuhi syarat sebagai peluang penjualan. Prospek adalah tahap awal dari pipa penjualan dan membutuhkan pengasuhan dan kualifikasi sebelum mereka dapat dikonversi menjadi peluang untuk co-selling aktif dengan AWS.

Prospek yang dibagikan AWS dengan mitra didasarkan pada sinyal keterlibatan pelanggan seperti kehadiran webinar, partisipasi kampanye, atau pertanyaan solusi mitra. Prospek ini mencakup konteks berharga seperti informasi perusahaan pelanggan, masalah bisnis, dan detail keterlibatan untuk membantu mitra memprioritaskan dan memenuhi syarat peluang potensial.

Bekerja dengan Undangan Lead

Mitra menerima undangan prospek dari AWS saat calon pelanggan menyatakan minatnya pada solusi atau layanan mitra. Proses undangan prospek memungkinkan mitra untuk mengevaluasi prospek sebelum berkomitmen untuk keterlibatan, memastikan alokasi sumber daya yang efisien dan tingkat konversi yang lebih tinggi.

Menerima Undangan Lead

AWS membuat undangan prospek dan membagikannya dengan mitra melalui API Penjualan. Saat undangan prospek baru tersedia, AWS kirimkan Undangan Keterlibatan dengan konteks Lead ke mitra yang memenuhi syarat.

Mitra harus memantau `Engagement Invitation Created` acara menggunakan Amazon EventBridge. Pemberitahuan acara ini mencakup `payloadType` bidang yang disetel ke `LeadInvitation`, memungkinkan mitra untuk membedakan undangan prospek dari undangan peluang. Setelah menerima acara, mitra dapat mengambil detail undangan untuk mengevaluasi prospek.

Daftar Undangan Utama

Mitra dapat melihat semua undangan prospek mereka menggunakan tindakan `ListEngagementInvitations` API. Tindakan ini mengambil undangan di mana prinsipal panggilan adalah pengirim atau penerima.

Untuk memfilter secara khusus untuk undangan prospek, mitra harus menggunakan `payloadType` filter dengan nilai `LeadInvitation`. Filter ini memastikan bahwa hanya undangan prospek yang dikembalikan, tidak termasuk undangan peluang dari hasil.

Undangan utama dapat berada di negara bagian berikut:

- Tertunda: Menunggu penerimaan atau penolakan mitra
- Diterima: Mitra telah menerima undangan dan mendapatkan akses ke detail prospek lengkap
- Ditolak: Mitra telah menolak undangan
- Kedaluwarsa: Undangan telah melewati tanggal kedaluwarsa tanpa tindakan

Mengevaluasi Undangan Lead

Sebelum menerima undangan prospek, mitra harus mengambil informasi undangan terperinci menggunakan tindakan `GetEngagementInvitation` API. Ini memberikan konteks penting untuk membuat keputusan menerima atau menolak informasi.

Muatan undangan utama meliputi:

Informasi Pelanggan (tersedia pra-penerimaan):

- Nama perusahaan, industri, dan negara
- URL Situs Web
- Segmen pasar (Perusahaan, Besar, Menengah, Kecil, Mikro)
- AWS tingkat kedewasaan (Mengevaluasi, Single-Account, Multi-Account)

Interaksi Pelanggan:

- Jenis sumber dan informasi kampanye
- Tindakan pelanggan diambil (penyelesaian formulir, kehadiran webinar, dll.)
- Deskripsi masalah bisnis yang disediakan oleh pelanggan
- Kategori kasus penggunaan

- Judul kontak (memberikan konteks Otoritas untuk kualifikasi BANT)

Important

Informasi kontak pelanggan (nama, email, nomor telepon) tidak terlihat pada tahap undangan. Detail kontak menjadi tersedia hanya setelah menerima undangan, memastikan privasi pelanggan dan mencegah perilaku pertanian timbal.

Menerima Undangan Lead

Ketika mitra memutuskan untuk mengejar prospek, mereka harus menerima undangan menggunakan tindakan `AcceptEngagementInvitation` API. Tindakan ini menambahkan mitra ke keterlibatan dan memberikan akses ke detail prospek lengkap, termasuk informasi kontak pelanggan.

Setelah penerimaan:

- Mitra ditambahkan sebagai anggota pertunangan
- Informasi kontak pelanggan menjadi terlihat melalui konteks Lead engagement
- Lead diklasifikasikan sebagai Partner Qualified Lead (PQL) dalam sistem AWS
- Prospek muncul di daftar prospek aktif mitra
- Mitra dapat mulai memelihara prospek dan membangun kontak pelanggan

Mitra dapat menerima beberapa undangan prospek secara terprogram dengan `AcceptEngagementInvitation` meminta setiap undangan, memungkinkan pemrosesan massal yang efisien dari prospek berkualitas tinggi.

Menolak Undangan Lead

Jika prospek tidak selaras dengan kemampuan, kapasitas, atau fokus bisnis mitra, mitra dapat menolak undangan menggunakan tindakan `RejectEngagementInvitation` API. Saat menolak undangan prospek, mitra harus memberikan alasan penolakan untuk membantu AWS meningkatkan perutean dan pencocokan prospek. Alasan penolakan umum meliputi:

- Kurangnya cakupan geografis
- Keahlian teknis yang tidak memadai untuk kasus penggunaan
- Kendala kapasitas saat ini

- Ketidakcocokan industri pelanggan
- Ketidaksejajaran anggaran

Setelah penolakan:

- Prospek dihapus dari antrian undangan tertunda mitra
- Informasi kontak pelanggan tidak pernah dibagikan dengan mitra
- Lead diklasifikasikan sebagai Partner Rejected Lead (PRL) dalam sistem AWS
- Undangan tetap terlihat di daftar undangan yang ditolak mitra untuk referensi

Prospek yang ditolak mungkin memenuhi syarat untuk penugasan kembali ke mitra lain, asalkan persyaratan persetujuan pelanggan terpenuhi.

Mengelola Leads yang Diterima

Setelah menerima undangan prospek, mitra dapat mengakses informasi prospek lengkap, memelihara hubungan pelanggan, dan melacak prospek melalui tahap kualifikasi.

Melihat Detail Lead

Mitra mengakses detail prospek lengkap menggunakan tindakan `GetEngagement API`. Keterlibatan berisi konteks Lead dengan informasi komprehensif tentang pelanggan dan interaksi mereka dengan AWS.

Konteks Lead meliputi:

Profil Pelanggan Lengkap:

- Semua informasi perusahaan dari undangan asli
- Detail kontak lengkap untuk semua interaksi pelanggan (nama, email, telepon, judul bisnis)
- AWS tingkat kematangan dan segmen pasar

Sejarah Interaksi:

- Beberapa titik kontak pelanggan dikonsolidasikan menjadi satu keterlibatan
- Setiap interaksi mencakup informasi sumber, tindakan pelanggan, masalah bisnis, dan detail kontak

- Interaksi dicantumkan secara kronologis untuk memberikan tampilan perjalanan pelanggan yang lengkap

Status Kualifikasi:

- Status kualifikasi prospek saat ini (Status tidak memenuhi syarat, memenuhi syarat, didiskualifikasi, atau kustom)
- Mitra dapat memperbarui status ini saat mereka maju melalui proses kualifikasi mereka

Pandangan komprehensif ini memungkinkan mitra untuk memahami riwayat keterlibatan lengkap pelanggan di beberapa AWS kampanye dan titik kontak, daripada memperlakukan setiap interaksi sebagai prospek yang terpisah.

Daftar Prospek Aktif

Partner dapat mengambil semua lead aktif mereka menggunakan aksi `ListEngagements` API dengan `contextType` filter yang disetel ke `Lead`. Ini mengembalikan keterlibatan di mana mitra adalah anggota dan keterlibatan berisi konteks `Lead`.

Tanggapan daftar mencakup informasi kunci seperti:

- ID Keterlibatan dan judul
- Nama perusahaan pelanggan
- Skor keterlibatan saat ini
- Status kualifikasi
- Jumlah interaksi
- Cap waktu pembuatan dan modifikasi

Mitra dapat menggunakan daftar ini untuk membangun dasbor, melacak kesehatan saluran pipa, dan mengidentifikasi prospek yang memerlukan perhatian.

Memperkaya Prospek dengan Wawasan Prospek

Mitra dapat memperkaya prospek yang diterima dengan AWS wawasan untuk memprioritaskan dan memenuhi syarat mereka dengan lebih baik sebelum berinvestasi dalam penjangkauan. Pengayaan dilakukan secara asinkron melalui tugas pencarian calon, yang menambah keterlibatan prospek dengan sinyal yang diturunkan untuk mendukung keputusan kualifikasi.AWS

Memulai tugas pencarian calon

Mitra memulai pengayaan menggunakan aksi API `StartProspectingFromEngagementTask`. Tindakan ini menerima hingga 100 pengidentifikasi keterlibatan dalam satu permintaan, memungkinkan mitra untuk memperkaya prospek dalam batch. Tugas berjalan secara asinkron dan segera mengembalikan `TaskId` dan `TaskArn` yang digunakan mitra untuk melacak kemajuan. Yang pertama `TaskStatus` adalah salah satu dari `PENDING`, `IN_PROGRESS`, `COMPLETED`, atau `FAILED`.

Mitra secara opsional dapat memberikan label `TaskName` untuk tugas dan `ClientToken` untuk memastikan idempotensi di seluruh percobaan ulang.

Polling untuk hasil

Karena prospecting bersifat asinkron, mitra melakukan polling untuk penyelesaian menggunakan aksi `GetProspectingFromEngagementTask` API dengan yang dikembalikan di awal.

`TaskId` Respons melaporkan status tugas keseluruhan dan hasil per-keterlibatan untuk setiap pengidentifikasi yang dikirimkan.

Setiap keterlibatan diproses secara independen, sehingga keterlibatan individu dapat berhasil atau gagal tanpa mempengaruhi yang lain dalam tugas yang sama. Untuk setiap keterlibatan, hasilnya meliputi:

- Pengenal keterlibatan: Keterlibatan yang diproses.
- Status: `PENDING`, `IN_PROGRESS`, `COMPLETED`, atau `FAILED`.
- Pengidentifikasi konteks prospek: Diisi saat keterlibatan berhasil diproses. Pengidentifikasi ini mereferensikan konteks prospek yang diperkaya yang ditambahkan ke keterlibatan dan dapat digunakan dalam operasi selanjutnya.
- Kode alasan dan pesan: Diisi hanya ketika keterlibatan gagal, memberikan kode kegagalan yang disebutkan dan deskripsi yang dapat dibaca manusia dengan langkah-langkah pemulihan yang disarankan.

Memantau banyak tugas

Untuk melacak pengayaan di banyak prospek, mitra menggunakan aksi `ListProspectingFromEngagementTasks` API. Tindakan ini mencantumkan semua tugas pencarian calon yang dimulai oleh akun pemanggil dan mendukung filter opsional berdasarkan pengenal tugas, nama tugas, atau rentang waktu mulai, dengan pengurutan yang dapat dikonfigurasi. Responsnya diberi paginasi; gunakan `NextToken` nilai dari setiap respons untuk mengambil halaman berikutnya.

Ketika pengayaan selesai dengan sukses, AWS wawasan ditambahkan ke keterlibatan sebagai konteks prospek. Mitra dapat mengambil detail yang diperkaya dengan `GetEngagement` dan menggunakannya untuk menetapkan status kualifikasi prospek dan memutuskan mana yang mengarah ke kemajuan menuju konversi.

Memperbarui Informasi Lead

Saat mitra memelihara prospek dan mengumpulkan informasi tambahan, mereka dapat memperbarui detail prospek menggunakan tindakan `UpdateEngagementContext` API. Tindakan ini memungkinkan mitra untuk memodifikasi konteks Lead dalam keterlibatan.

Mitra dapat memperbarui:

Status Kualifikasi: Mitra harus memperbarui status kualifikasi saat mereka maju melalui proses kualifikasi internal mereka:

- Tidak memenuhi syarat: Status default setelah menerima prospek; menunjukkan prospek memerlukan evaluasi
- Memenuhi syarat: Memimpin memenuhi kriteria kualifikasi dan menunjukkan potensi tinggi untuk konversi ke peluang
- Didiskualifikasi: Timbal tidak memenuhi kriteria atau tidak cocok untuk solusi mitra
- Negara Kustom: Mitra dapat menentukan status kualifikasi mereka sendiri agar sesuai dengan proses internal mereka

Metadata Utama: Mitra dapat menambah atau memperbarui informasi tambahan yang relevan dengan kualifikasi dan proses pengasuhan mereka.

Memperbarui status kualifikasi membantu mitra melacak perkembangan prospek, menghasilkan laporan pipeline yang akurat, dan mengidentifikasi prospek yang siap dikonversi ke peluang.

Memantau AWS Pembaruan

AWS dapat memperbarui informasi prospek berdasarkan sinyal keterlibatan pelanggan baru atau penilaian keterlibatan yang disempurnakan. Saat AWS memperbarui keterlibatan, mitra menerima `Engagement Updated` acara melalui `Amazon EventBridge`.

Pembaruan ini mungkin termasuk:

- Skor keterlibatan yang disempurnakan berdasarkan aktivitas pelanggan baru

- Interaksi pelanggan tambahan dari kampanye atau titik sentuh baru
- Informasi pelanggan yang diperbarui

Mitra harus memantau peristiwa ini dan menelepon `GetEngagement` untuk mengambil detail prospek terbaru. Ini memastikan mitra memiliki informasi terkini untuk memprioritaskan dan memelihara prospek.

`Engagement UpdatedAcara` ini mencakup `contextTypes` bidang, memungkinkan mitra untuk memfilter secara khusus untuk prospek (keterlibatan dengan konteks `Lead`).

Mengubah Lead menjadi Peluang

Ketika prospek telah memenuhi syarat dan menunjukkan niat membeli yang serius, mitra dapat mengubahnya menjadi peluang untuk kolaborasi co-selling formal dengan AWS. Konversi ini menandai transisi dari pengasuhan prospek (terutama didorong oleh mitra) ke manajemen peluang (kolaboratif dengan AWS).

Pendekatan yang Direkomendasikan: API Kenyamanan

Mitra dapat merampingkan proses pembuatan dan penautan peluang dengan menggunakan tindakan API `StartOpportunityFromEngagementTask` kenyamanan. API tugas ini mengatur beberapa tindakan secara otomatis:

1. Menciptakan draf peluang dengan informasi awal dari konteks utama
2. Menautkan peluang ke keterlibatan sumber melalui snapshot sumber daya
3. Menambahkan `CustomerProject` konteks ke keterlibatan

Metode kenyamanan ini mengurangi jumlah panggilan API yang diperlukan dan memastikan pelacakan atribusi yang tepat. Mitra yang menggunakan pendekatan ini masih perlu:

- Lengkapi detail peluang menggunakan `UpdateOpportunity`
- Solusi Mitra Rekanan menggunakan `AssociateOpportunity`
- Kirimkan kesempatan menggunakan `StartEngagementFromOpportunityTask` saat siap

API kenyamanan sangat berguna bagi mitra dengan alur kerja lead-to-opportunity otomatis yang ingin meminimalkan kompleksitas integrasi.

Pendekatan Alternatif: Step-by-step API

Membuat Peluang Draf

Langkah pertama dalam mengonversi prospek adalah membuat peluang draf menggunakan tindakan `CreateOpportunity` API. Ini menciptakan peluang dengan `Lifecycle.ReviewStatus` set `kePending Submission`. Pada tahap ini, peluang belum diserahkan AWS untuk validasi.

Mitra harus mengisi kesempatan dengan informasi yang dikumpulkan selama pengasuhan prospek, termasuk:

- Detail akun pelanggan
- Informasi proyek dan masalah bisnis
- Pengeluaran pelanggan yang diharapkan
- Target tanggal penutupan
- Detail relevan lainnya yang dikumpulkan selama kualifikasi

Peluang draf memungkinkan mitra untuk menyiapkan informasi lengkap sebelum mengirimkan AWS, memastikan tingkat persetujuan yang lebih tinggi dan validasi yang lebih cepat.

Menghubungkan Peluang untuk Memimpin Keterlibatan

Setelah membuat peluang draf, mitra harus menautkannya ke keterlibatan prospek sumber menggunakan tindakan `CreateResourceSnapshot` API. Langkah ini sangat penting untuk:

- Pelacakan Atribusi: Menetapkan rantai asal dari undangan prospek melalui penutupan peluang, memungkinkan perhitungan ROI yang akurat untuk kampanye pemasaran dan sumber prospek.
- Metrik Konversi: Memungkinkan AWS dan mitra untuk mengukur tingkat konversi prospek ke peluang dan mengidentifikasi sumber prospek yang berhasil.
- Pelestarian Konteks: Mempertahankan riwayat perjalanan pelanggan yang lengkap, termasuk semua interaksi asli dan sinyal keterlibatan.

Saat membuat snapshot sumber daya, mitra menentukan:

- ID keterlibatan (keterlibatan prospek sumber)
- Pengidentifikasi peluang
- Jenis sumber daya (Peluang)

Tindakan ini secara otomatis menambahkan CustomerProject konteks pada keterlibatan, menandakan bahwa prospek telah diubah menjadi peluang aktif. Keterlibatan sekarang berisi konteks Lead asli (melestarikan sejarah) dan CustomerProject konteks baru (menunjukkan peluang aktif).

Menyelesaikan Detail Peluang

Setelah peluang dibuat dan ditautkan, mitra harus menyelesaikan persyaratan peluang tambahan sebelum diserahkan. Lihat tindakan AssociateOpportunity API untuk detailnya.

Memperbarui Detail Proyek: Mitra dapat menggunakan tindakan UpdateOpportunity API untuk menyempurnakan informasi proyek, masalah bisnis pelanggan, pengeluaran yang diharapkan, dan detail relevan lainnya yang dikumpulkan selama kualifikasi prospek.

Mengirimkan Kesempatan

Setelah semua informasi yang diperlukan selesai, mitra mengirimkan kesempatan untuk AWS validasi menggunakan API StartEngagementFromOpportunityTask kenyamanan. Ini mengalihkan peluang dari status draf ke proses AWS peninjauan.

Persyaratan Validasi: Keterlibatan harus memiliki CustomerProject konteks sebelum pengiriman. Konteks ini dibuat secara otomatis saat menggunakan CreateResourceSnapshot untuk menautkan peluang ke keterlibatan. Jika konteks ini hilang, kiriman akan gagal.

Setelah pengajuan:

- Kesempatan Lifecycle.ReviewStatus berubah menjadi Submitted
- Lead diklasifikasikan sebagai Partner Sales Qualified Lead (PSQL) dalam sistem AWS
- AWS memulai validasi untuk memastikan detail peluang akurat dan lengkap
- Tidak ada perubahan yang dapat dilakukan pada kesempatan sampai proses peninjauan selesai

Peluang kemudian mengikuti alur kerja validasi standar yang dijelaskan dalam dokumentasi “Bekerja dengan peluang Anda”, berkembang melalui status seperti, Tindakan yang Diperlukan, Disetujui In-Review, atau Didiskualifikasi.

Bekerja dengan wawasan penyesuaian ukuran kesepakatan

Apa itu Ukuran Kesepakatan?

Deal Sizing adalah kemampuan dalam Peluang Keterlibatan Pelanggan APN (ACE) di AWS Partner Central yang menggunakan AI untuk memberikan perkiraan ukuran transaksi otomatis dan rekomendasi AWS layanan saat Mitra membuat atau memperbarui peluang.

AI memperkirakan perkiraan MRR

Wawasan ukuran kesepakatan memberikan median rentang pendapatan berulang bulanan (MRR) yang diperkirakan berdasarkan AWS peluang historis Mitra, kasus penggunaan, dan deskripsi bisnis. Mitra harus meninjau dan memperbarui total MRR karena Mitra mengumpulkan lebih banyak informasi tentang kesepakatan dan kemajuan melalui siklus penjualan.

AI-Recommended AWS Produk

AI-recommended produk diidentifikasi berdasarkan deskripsi masalah bisnis pelanggan dan rincian peluang. AI merekomendasikan AWS produk yang relevan selaras dengan persyaratan teknis dan kasus penggunaan. Mitra harus meninjau saran ini dan menyesuaikan pemilihan produk agar sesuai dengan kebutuhan spesifik pelanggan.

Wawasan yang Ditingkatkan dengan URL Kalkulator Harga

Mitra juga dapat mengimpor URL Kalkulator AWS Harga untuk secara otomatis mengisi pilihan layanan dan menerima wawasan yang disempurnakan termasuk indikator kelayakan Migration Acceleration Program (MAP), rekomendasi pengoptimalan, dan analisis penghematan biaya potensial.

Memahami Source-Separated Data

Deal Sizing memberikan wawasan dari dua sumber terpisah untuk memberi Anda visibilitas komprehensif ke dalam nilai peluang:

- AWS Sumber: AI-recommended produk berdasarkan pola dari peluang sejarah yang serupa
- Sumber Mitra: Perkiraan Anda sendiri yang diimpor dari Kalkulator AWS Harga

Menangani Varians Antar Sumber

Ketika perkiraan yang diberikan mitra berbeda secara signifikan dari AWS rekomendasi, Mitra harus:

1. Tinjau detail peluang untuk memastikan semua informasi yang relevan ditangkap
2. Verifikasi konfigurasi URL Kalkulator Harga sesuai dengan persyaratan aktual
3. Pertimbangkan AWS rekomendasi sebagai titik data yang diinformasikan oleh peluang historis serupa
4. Membuat keputusan berdasarkan konteks dan persyaratan pelanggan tertentu

Mengakses Wawasan Ukuran Kesepakatan

Partner mengakses data Deal Sizing melalui tindakan `GetAwsOpportunitySummary` API, yang mengembalikan `AwsOpportunitySummary` objek yang diperluas yang berisi perkiraan ukuran kesepakatan, rekomendasi produk, dan wawasan pengoptimalan.

Ketika Data Ukuran Kesepakatan Menjadi Tersedia

Wawasan Ukuran Kesepakatan menjadi tersedia setelah peluang dibuat dengan detail pelanggan dan proyek yang memadai. Sistem menganalisis atribut peluang dan menghasilkan:

1. AI memperkirakan MRR: Secara otomatis dihitung berdasarkan karakteristik peluang
2. AWS rekomendasi produk: Layanan yang relevan dengan masalah bisnis pelanggan dan persyaratan teknis
3. Wawasan yang disempurnakan (saat URL Kalkulator Harga disediakan): Kelayakan MAP, rekomendasi pengoptimalan, dan analisis penghematan biaya

Note

Wawasan Ukuran Kesepakatan tidak tersedia untuk semua peluang. Kelayakan tergantung pada faktor-faktor seperti riwayat mitra dan kelengkapan detail peluang yang diberikan.

Memberikan Masukan Ukuran Kesepakatan

Mitra menyediakan masukan untuk Ukuran Kesepakatan melalui tindakan `UpdateOpportunity` API `CreateOpportunity` dan API. Sistem menggunakan atribut peluang untuk menghasilkan rekomendasi.

Memberikan Total Nilai Kontrak (TCV)

Mitra yang memperkirakan Total Nilai Kontrak daripada Pendapatan Berulang Bulanan dapat memberikan nilai transaksi mereka secara langsung. AWS secara otomatis mengonversi TCV ke AWS MRR menggunakan AI-powered model konversi.

Untuk ukuran kesepakatan di TCV, berikan `ExpectedContractDuration` as Months (nilai yang valid berkisar dari 1 hingga 144 bulan) dan `TargetCompany` sebagai `Self`. `Frequency` `None`

```
{
  "Project": {
    "ExpectedContractDuration": { "Term": "Months", "Value": "36" },
    "ExpectedCustomerSpend": [
      {
        "Amount": "1200000.00",
        "CurrencyCode": "USD",
        "Frequency": "None",
        "TargetCompany": "Self"
      }
    ]
  }
}
```

AWS memperoleh estimasi AWS MRR dan mengembalikan kedua entri pada: `GetOpportunity`

```
{
  "Project": {
    "ExpectedContractDuration": { "Term": "Months", "Value": "36" },
    "ExpectedCustomerSpend": [
      {
        "Amount": "5600.00",
        "CurrencyCode": "USD",
        "Frequency": "Monthly",
        "TargetCompany": "AWS"
      },
      {
        "Amount": "1200000.00",
        "CurrencyCode": "USD",
        "Frequency": "None",
        "TargetCompany": "Self"
      }
    ]
  }
}
```

```

    ]
  }
}
```

Dalam contoh di atas, array `ExpectedCustomerSpend` pertama berisi estimasi AWS MRR (`TargetCompany: "AWS", Frequency: "Monthly"`) dan array kedua adalah Total Contract Value (`TargetCompany: "Self", Frequency: "None"`) mitra.

Aturan Validasi TCV

Semua kesalahan validasi TCV mengembalikan kode `INVALID_VALUE` kesalahan dengan. Reason: `"BUSINESS_VALIDATION_FAILED"` Faktor yang membedakan adalah `FieldName` dan `Message`:

Kondisi	Bidang	Pesan
Entri mandiri tanpa <code>ExpectedContractDuration</code>	<code>project.expectedContractDuration</code>	<code>ExpectedContractDuration</code> diperlukan ketika entri Diri hadir
<code>ExpectedContractDuration</code> tanpa entri Diri (Buat saja)	<code>project.expectedCustomerSpend</code>	Entri mandiri <code>ExpectedContractDuration</code> diperlukan saat hadir
Beberapa entri Mandiri	<code>project.expectedCustomerSpend</code>	Hanya satu entri Mandiri yang diizinkan
Tidak valid <code>TargetCompany</code> saat durasi hadir	<code>project.expectedCustomerSpend.targetCompany</code>	<code>TargetCompany</code> harus 'AWS' atau 'Self'
Ketidakcocokan mata uang antara AWS dan Self	<code>project.expectedCustomerSpend.currencyCode</code>	<code>CurrencyCode</code> harus cocok antar entri
Masuk sendiri + <code>EstimationUrl</code> bersama	<code>project.expectedCustomerSpend.estimationUrl</code>	Masuk sendiri dan <code>EstimationUrl</code> tidak bisa hidup berdampingan

Note

AktifUpdateOpportunity, jika ExpectedContractDuration ada tanpa entri Diri tetapi nilai kesepakatan mitra ada di penyimpanan, sistem merekonstruksi entri Diri secara otomatis - tidak ada kesalahan yang dilemparkan.

Beralih Antara TCV dan MRR Manual

Untuk beralih dari mode TCV kembali ke MRR manual, setel secara eksplisit ExpectedContractDuration ke null dan berikan hanya entri: AWS

```
{
  "Project": {
    "ExpectedContractDuration": null,
    "ExpectedCustomerSpend": [
      {
        "Amount": "8000.00",
        "CurrencyCode": "USD",
        "Frequency": "Monthly",
        "TargetCompany": "AWS"
      }
    ]
  }
}
```

Note

Menghilangkan ExpectedContractDuration dari muatan (tidak mengirim bidang) berarti “tidak ada perubahan” - data TCV yang ada dipertahankan. Mengirim secara eksplisit null berarti “hapus mode TCV.”

TCV dan URL Kalkulator Harga

Jika mitra menyediakan Self entri (mode TCV) danEstimationUrl, API mengembalikan kesalahan validasi. Masuk sendiri dan EstimationUrl tidak dapat hidup berdampingan dalam permintaan yang sama.

Memberikan Estimasi MRR Manual

Mitra dapat memasukkan perkiraan MRR secara manual menggunakan `Project.ExpectedCustomerSpend[].Amount` bidang:

Note

`CurrencyCodeBidang` menerima USD dan EUR. Jika AWS Partition adalah `aws-eusc` (AWS European Sovereign Cloud), kode mata uangnya harus. EUR

```
{
  "Project": {
    "ExpectedCustomerSpend": [
      {
        "Amount": "25000.00",
        "CurrencyCode": "USD",
        "Frequency": "Monthly",
        "TargetCompany": "AWS"
      }
    ]
  }
}
```

Bidang ini tersedia untuk semua Mitra dan dapat digunakan untuk menerima perkiraan AI dengan menetapkan nilai yang sama atau untuk mengganti dengan Partner-provided nilai.

Menyediakan URL Kalkulator Harga

Mitra secara opsional dapat menyediakan URL Kalkulator AWS Harga melalui `Project.ExpectedCustomerSpend[].EstimationUrl` bidang untuk secara otomatis mengimpor konfigurasi layanan dan menerima wawasan yang disempurnakan termasuk indikator kelayakan MAP, rekomendasi pengoptimalan, dan analisis penghematan biaya.

```
{
  "Project": {
    "ExpectedCustomerSpend": [
      {
        "Amount": null,
        "CurrencyCode": "USD",
```

```
        "EstimationUrl": "https://calculator.aws/#/estimate?id=abc123",
        "Frequency": "Monthly",
        "TargetCompany": "AWS"
    }
]
}
```

Ketika URL Kalkulator Harga yang valid disediakan, sistem secara otomatis menghitung jumlah MRR dari konfigurasi kalkulator dan mengisi bidang Jumlah dan data pengeluaran tingkat produk terperinci di sumber Mitra. `AwsProductsSpendInsightsBySource` Mitra harus menetapkan Jumlah ke null saat memasok `EstimationUrl` - sistem akan menghitungnya secara otomatis.

URL dalam format dan URL yang tidak valid yang tidak dapat diselesaikan akan ditolak dengan file. `ValidationException`

Berapa Jumlah dan `EstimationUrl` Bekerja Bersama

Anda memiliki fleksibilitas dalam cara Anda memberikan estimasi pendapatan berulang bulanan (MRR) untuk peluang. API secara cerdas menangani berbagai kombinasi jumlah manual dan URL Kalkulator AWS Harga untuk memastikan bahwa peluang Anda selalu dapat dibuat dengan sukses.

Hanya menggunakan jumlah manual

Jika Anda hanya menyediakan bidang Jumlah, API akan menyimpan estimasi manual Anda dan disetel `EstimationUrl` ke null. Pendekatan ini berguna ketika Anda memiliki perkiraan MRR tertentu dari percakapan pelanggan atau perhitungan Anda sendiri.

Contoh permintaan:

```
{
  "Project": {
    "ExpectedCustomerSpend": [{
      "Amount": "25000.00",
      "CurrencyCode": "USD",
      "Frequency": "Monthly",
      "TargetCompany": "AWS"
    }]
  }
}
```

Hanya menggunakan sebuah AWS URL Kalkulator Harga

Bila Anda hanya menyediakan EstimationUrl bidang:

- URL yang valid — API menghitung jumlah MRR dari konfigurasi kalkulator Anda dan menyimpan URL dan jumlah yang dihitung.
- URL tidak valid — API menampilkan rincian tentang mengapa URL tidak valid. `ValidationException`

Contoh permintaan:

```
{
  "Project": {
    "ExpectedCustomerSpend": [{
      "EstimationUrl": "https://calculator.aws/#/estimate?id=abc123",
      "CurrencyCode": "USD",
      "Frequency": "Monthly",
      "TargetCompany": "AWS"
    }]
  }
}
```

Contoh respons kesalahan (URL tidak valid):

```
{
  "ErrorList": [{
    "Code": "INVALID_VALUE",
    "FieldName": "project.expectedCustomerSpend.estimationUrl",
    "Message": "Invalid Estimation URL: https://calculator.aws/#/estimate?id=invalid"
  }],
  "Message": "The provided Estimation URL is invalid",
  "Reason": "INVALID_VALUE"
}
```

Memberikan Jumlah dan EstimationUrl

Saat Anda menyediakan kedua bidang, API memprioritaskan memastikan bahwa peluang Anda berhasil dibuat:

1. Jika kedua nilai tidak berubah - tidak ada pembaruan yang dilakukan pada bidang ini

2. URL tidak valid - API menyimpan Jumlah yang Anda berikan dan disetel `EstimationUrl` ke null, memungkinkan pembuatan kesempatan Anda untuk dilanjutkan.
3. URL valid dan jumlah yang dihitung sesuai dengan Jumlah yang Anda berikan — Kedua nilai disimpan.
4. URL valid tetapi jumlah yang dihitung tidak sesuai dengan Jumlah yang Anda berikan — API menyimpan Jumlah yang Anda berikan dan disetel `EstimationUrl` ke null tanpa menampilkan kesalahan.

Note

Jika kedua nilai tidak cocok dengan yang sudah ada, API pertama-tama mencoba menghitung jumlah dari URL. Jumlah yang Anda berikan secara manual selalu diutamakan ketika ada masalah ketidakcocokan atau validasi.

Manfaat utama: URL yang tidak valid tidak pernah memblokir pembuatan peluang

Pendekatan ini memastikan bahwa URL Kalkulator AWS Harga yang tidak valid atau tidak dapat diakses tidak pernah mencegah Anda membuat atau memperbarui peluang. Jumlah yang Anda berikan secara manual selalu diutamakan ketika ada masalah konflik atau validasi, memberi Anda kendali penuh atas data peluang Anda.

Memahami Model `AwsOpportunitySummary` Data yang Diperluas

Bagian ini menjelaskan struktur respons yang dikembalikan oleh aksi `GetAwsOpportunitySummary` API.

Note

AI-forecasted MRR dikembalikan `Project.ExpectedCustomerSpend[] .Amount`

Tindakan `GetAwsOpportunitySummary` API mengembalikan wawasan ukuran kesepakatan melalui `Insights.AwsProductsSpendInsightsBySource` struktur baru yang menyediakan analisis pengeluaran komprehensif dengan atribusi sumber.

Pemisahan Sumber: AWS vs Data Mitra

Ukuran Kesepakatan memisahkan wawasan berdasarkan sumber untuk memberikan transparansi dan mencegah penghitungan ganda:

- AWS Sumber: Rekomendasi produk AI dari AWS
- Sumber Mitra: Belanjaan data dan detail produk yang diimpor dari URL Kalkulator Harga yang disediakan mitra

Pemisahan ini memungkinkan Mitra untuk:

1. Bandingkan perkiraan mereka dengan AWS rekomendasi
2. Validasi asumsi ukuran mereka
3. Hindari menggembungkan total dengan menggabungkan kedua sumber secara tidak sengaja
4. Pertahankan visibilitas ke sumber data

Ikhtisar Struktur

```
{
  "Insights": {
    "AwsProductsSpendInsightsBySource": {
      "<AWS | Partner>": {
        "CurrencyCode": "string",
        "Frequency": "string",
        "TotalAmount": "string",
        "TotalOptimizedAmount": "string",
        "TotalPotentialSavingsAmount": "string",
        "TotalAmountByCategory": { },
        "AwsProducts": [ ]
      }
    }
  },
  "Project": {
    "ExpectedCustomerSpend": [ ]
  },
  "RelatedEntityIds": {
    "AwsProducts": [ ]
  }
}
```

Referensi Lapangan

AwsProductsSpendInsightsBySource Peta

Source-separated menghabiskan wawasan yang memberikan analisis independen untuk AWS rekomendasi dan perkiraan mitra.

`Insights.AwsProductsSpendInsightsBySourceBidang` adalah peta yang berisi hingga dua kunci:

Key	Tipe	Deskripsi
AWS	AwsProductsSpendInsights	AI-generated wawasan termasuk produk yang direkomendasikan dari AWS
Mitra	AwsProductsSpendInsights	Partner-sourced wawasan yang berasal dari URL Kalkulator Harga termasuk biaya layanan terperinci dan pengoptimalan

Note

Hanya sumber dengan data yang tersedia yang disertakan dalam respons. Peluang baru biasanya dimulai dengan hanya AWS sumber yang dihuni.

AwsProductInsights Objek

Analisis pengeluaran komprehensif untuk satu sumber (AWS atau Mitra) termasuk jumlah total, penghematan pengoptimalan, perincian kategori program, dan wawasan tingkat produk terperinci.

Setiap objek sumber berisi bidang-bidang berikut:

Bidang	Tipe	Deskripsi
CurrencyCode	string	Kode mata uang ISO 4217. Nilai yang didukung adalah “USD” dan “EUR”.
Frekuensi	string	Menunjukkan seberapa sering pelanggan diharapkan untuk menghabiskan jumlah yang diproyeksikan. Hanya nilai Bulanan yang diizinkan untuk bidang Frekuensi, mewakili pengeluaran bulanan berulang.
TotalAmount	string	Total perkiraan pengeluaran untuk sumber ini sebelum pengoptimalan
TotalOptimizedAmount	string	Total perkiraan pengeluaran setelah menerapkan pengoptimalan yang direkomendasikan
TotalPotentialSavingsAmount	string	Penghematan terukur dapat dicapai melalui penerapan pengoptimalan
TotalAmountByCategory	object	Menghabiskan jumlah yang dipetakan ke AWS program dan jalur modernisasi
AwsProducts	array	Product-level rincian termasuk biaya dan rekomendasi optimasi

Batasan Status Saat Ini: Untuk AWS sumber TotalAmount, TotalOptimizedAmount,, dan TotalPotentialSavingsAmount dapat dihilangkan ketika rekomendasi pengeluaran per produk tidak tersedia. Mitra harus referensi `Project.ExpectedCustomerSpend[] .Amount` untuk perkiraan AWS MRR total dalam kasus ini.

TotalAmountByCategory Objek

Peta menghabiskan jumlah untuk AWS program dan inisiatif strategis:

Kategori Kunci	Deskripsi
MAP Memenuhi Syarat	Pengeluaran untuk layanan yang memenuhi syarat untuk pendanaan Program Migration Acceleration
CAT Memenuhi Syarat	Layanan yang mendukung alokasi dan pelacakan biaya
Pathway - Pindah ke Cloud Native	Layanan selaras dengan modernisasi cloud-native
Pathway - Pindah ke Managed Services	Layanan yang mendukung adopsi layanan terkelola
Pathway - Pindah ke Open Source	Alternatif layanan sumber terbuka
Jalur - Pindah ke Kontainer	Container-based opsi layanan
Pathway - Pindah ke Tanpa Server	Layanan komputasi tanpa server
Pathway - Pindah ke Database	Layanan modernisasi basis data
Pathway - Pindah ke Analytics	Layanan analitik dan pemrosesan data

Note

Jumlah kategori total dapat melebihi TotalAmount karena produk individu dapat termasuk dalam beberapa kategori.

AwsProducts Array

Daftar AWS layanan dengan indikator kelayakan program (MAP, jalur modernisasi), perkiraan biaya, dan rekomendasi pengoptimalan.

Setiap objek produk berisi:

Bidang	Tipe	Diperlukan	Deskripsi
ProductCode	string	Ya	AWS Pengidentifikasi produk Partner Central yang digunakan untuk asosiasi peluang
ServiceCode	string	Tidak	Kode layanan Kalkulator Harga (tautan ke URL kalkulator asli)
Kategori	array	Ya	Daftar kategori program dan jalur produk ini memenuhi syarat untuk
Jumlah	string	Tidak	Biaya layanan dasar sebelum pengoptimalan (mungkin nol untuk rekomendasi yang bersumber)AWS
OptimizedAmount	string	Tidak	Biaya layanan setelah menerapkan pengoptimalan (mungkin nol untuk AWS rekomendasi yang bersumber)

Bidang	Tipe	Diperlukan	Deskripsi
PotentialSavingsAmount	string	Tidak	Service-specific pengurangan biaya melalui pengoptimalan (mungkin nol untuk AWS rekomendasi yang bersumber)
Pengoptimalan	array	Tidak	Daftar rekomendasi pengoptimalan khusus untuk produk ini

Penanganan Null: Untuk produk yang AWS bersumber, Jumlah OptimizedAmount, dan PotentialSavingsAmount bidang mungkin nol jika rekomendasi pembelanjaan per produk tidak tersedia. Sistem menyediakan MRR total melalui `Project.ExpectedCustomerSpend[0].Amount` sebagai gantinya.

Objek Optimasi

Setiap rekomendasi pengoptimalan berisi:

Bidang	Tipe	Deskripsi
Deskripsi	string	Human-readable penjelasan tentang strategi optimasi
SavingsAmount	string	Penghematan biaya terukur dapat dicapai dengan menerapkan optimasi ini

Contoh

Contoh 1: AWS rekomendasi Saja (Keadaan Saat Ini)

Contoh ini menunjukkan respons khas ketika hanya rekomendasi AI yang tersedia dan jumlah pengeluaran per produk belum dapat direkomendasikan.

```
{
  "Catalog": "AWS",
  "Insights": {
    "AwsProductsSpendInsightsBySource": {
      "AWS": {
        "CurrencyCode": "USD",
        "Frequency": "Monthly",
        "AwsProducts": [
          {
            "ProductCode": "AmazonEC2",
            "Categories": ["MAP Eligible", "Cost Allocation Tagging"],
            "Amount": null,
            "OptimizedAmount": null,
            "PotentialSavingsAmount": null,
            "Optimizations": []
          },
          {
            "ProductCode": "AWSLambda",
            "Categories": ["Cost Allocation Tagging", "Pathway - Move to Cloud
Native"],
            "Amount": null,
            "OptimizedAmount": null,
            "PotentialSavingsAmount": null,
            "Optimizations": []
          },
          {
            "ProductCode": "AmazonRDS",
            "Categories": ["MAP Eligible", "Cost Allocation Tagging", "Pathway - Move
to Managed Services"],
            "Amount": null,
            "OptimizedAmount": null,
            "PotentialSavingsAmount": null,
            "Optimizations": []
          }
        ]
      }
    }
  }
}
```

```

},
"Project": {
  "ExpectedCustomerSpend": [
    {
      "Amount": "28000.00",
      "CurrencyCode": "USD",
      "EstimationUrl": null,
      "Frequency": "Monthly",
      "TargetCompany": "AWS"
    }
  ]
},
"RelatedEntityIds": {
  "AwsProducts": [
    "AmazonEC2",
    "AWSLambda",
    "AmazonRDS"
  ]
}
}

```

Poin Utama: AWS sumber menunjukkan produk yang direkomendasikan dengan kategori tetapi tidak ada jumlah per produk. Total estimasi MRR tersedia melalui `Project.ExpectedCustomerSpend[].Amount`.

Contoh 2: Impor Kalkulator Harga Mitra

Contoh ini menunjukkan wawasan yang disempurnakan saat mitra telah menyediakan URL Kalkulator Harga yang valid.

```

{
  "Catalog": "AWS",
  "Insights": {
    "AwsProductsSpendInsightsBySource": {
      "Partner": {
        "CurrencyCode": "USD",
        "Frequency": "Monthly",
        "TotalAmount": "32500.00",
        "TotalOptimizedAmount": "30000.00",
        "TotalPotentialSavingsAmount": "2500.00",
        "TotalAmountByCategory": {
          "MAP Eligible": "22500.00",
          "Cost Allocation Tagging": "32500.00",

```

```

    "Pathway - Move to Cloud Native": "5000.00",
    "Pathway - Move to Managed Services": "7500.00"
  },
  "AwsProducts": [
    {
      "ServiceCode": "ec2",
      "Categories": ["MAP Eligible", "Cost Allocation Tagging"],
      "Amount": "15000.00",
      "OptimizedAmount": "13500.00",
      "PotentialSavingsAmount": "1500.00",
      "Optimizations": [
        {
          "Description": "Convert to 3-year reserved instances",
          "SavingsAmount": "1000.00"
        },
        {
          "Description": "Migrate to Graviton-based instances",
          "SavingsAmount": "500.00"
        }
      ]
    },
    {
      "ServiceCode": "lambda",
      "Categories": ["Cost Allocation Tagging", "Pathway - Move to Cloud
Native"],
      "Amount": "5000.00",
      "OptimizedAmount": "5000.00",
      "PotentialSavingsAmount": "0.00",
      "Optimizations": []
    },
    {
      "ServiceCode": "AmazonRDS",
      "Categories": ["MAP Eligible", "Cost Allocation Tagging", "Pathway - Move
to Managed Services"],
      "Amount": "7500.00",
      "OptimizedAmount": "6500.00",
      "PotentialSavingsAmount": "1000.00",
      "Optimizations": [
        {
          "Description": "Optimize Multi-AZ for dev/test environments",
          "SavingsAmount": "1000.00"
        }
      ]
    }
  ],

```

```

    {
      "ServiceCode": "AmazonS3",
      "Categories": ["MAP Eligible", "Cost Allocation Tagging"],
      "Amount": "5000.00",
      "OptimizedAmount": "5000.00",
      "PotentialSavingsAmount": "0.00",
      "Optimizations": []
    }
  ]
}
},
"Project": {
  "ExpectedCustomerSpend": [
    {
      "Amount": "32500.00",
      "CurrencyCode": "USD",
      "EstimationUrl": "https://calculator.aws/#/estimate?id=abc123",
      "Frequency": "Monthly",
      "TargetCompany": "AWS"
    }
  ]
},
"RelatedEntityIds": {
  "AwsProducts": []
}
}

```

Poin Utama: Sumber mitra berisi detail pengeluaran lengkap dengan jumlah per produk.

Rekomendasi optimasi memberikan strategi pengurangan biaya yang dapat ditindaklanjuti. Rincian kategori menunjukkan kelayakan MAP (\$22.500 dari total \$32,500). Produk termasuk ServiceCode menautkan kembali ke konfigurasi Kalkulator Harga.

Contoh 3: Kedua Sumber Hadir (Skenario Perbandingan)

Contoh ini menunjukkan bagaimana AWS rekomendasi dan perkiraan mitra muncul bersama, memungkinkan perbandingan.

```

{
  "Catalog": "AWS",
  "Insights": {
    "AwsProductsSpendInsightsBySource": {
      "AWS": {

```

```

    "CurrencyCode": "USD",
    "Frequency": "Monthly",
    "AwsProducts": [
      {
        "ProductCode": "AmazonEC2",
        "Categories": ["MAP Eligible", "Cost Allocation Tagging"],
        "Amount": null,
        "OptimizedAmount": null,
        "PotentialSavingsAmount": null,
        "Optimizations": []
      },
      {
        "ProductCode": "AWSLambda",
        "Categories": ["Cost Allocation Tagging", "Pathway - Move to Cloud
Native"],
        "Amount": null,
        "OptimizedAmount": null,
        "PotentialSavingsAmount": null,
        "Optimizations": []
      },
      {
        "ProductCode": "EBS",
        "Categories": ["MAP Eligible", "Cost Allocation Tagging"],
        "Amount": null,
        "OptimizedAmount": null,
        "PotentialSavingsAmount": null,
        "Optimizations": []
      },
      {
        "ProductCode": "RDS",
        "Categories": ["MAP Eligible", "Cost Allocation Tagging", "Pathway - Move
to Managed Services"],
        "Amount": null,
        "OptimizedAmount": null,
        "PotentialSavingsAmount": null,
        "Optimizations": []
      }
    ]
  },
  "Partner": {
    "CurrencyCode": "USD",
    "Frequency": "Monthly",
    "TotalAmount": "32500.00",
    "TotalOptimizedAmount": "30000.00",

```

```

    "TotalPotentialSavingsAmount": "2500.00",
    "TotalAmountByCategory": {
      "MAP Eligible": "22500.00",
      "Cost Allocation Tagging": "32500.00",
      "Pathway - Move to Cloud Native": "5000.00",
      "Pathway - Move to Managed Services": "7500.00"
    },
    "AwsProducts": [
      {
        "ServiceCode": "ec2",
        "Categories": ["MAP Eligible", "Cost Allocation Tagging"],
        "Amount": "15000.00",
        "OptimizedAmount": "13500.00",
        "PotentialSavingsAmount": "1500.00",
        "Optimizations": [
          {
            "Description": "Convert to 3-year reserved instances",
            "SavingsAmount": "1000.00"
          }
        ]
      },
      {
        "ServiceCode": "lambda",
        "Categories": ["Cost Allocation Tagging", "Pathway - Move to Cloud
Native"],
        "Amount": "5000.00",
        "OptimizedAmount": "5000.00",
        "PotentialSavingsAmount": "0.00",
        "Optimizations": []
      },
      {
        "ServiceCode": "amazonRDS",
        "Categories": ["MAP Eligible", "Cost Allocation Tagging", "Pathway - Move
to Managed Services"],
        "Amount": "7500.00",
        "OptimizedAmount": "6500.00",
        "PotentialSavingsAmount": "1000.00",
        "Optimizations": [
          {
            "Description": "Optimize Multi-AZ for dev/test environments",
            "SavingsAmount": "1000.00"
          }
        ]
      }
    ],
  },

```



```

        {
            "ServiceCode": "amazonS3",
            "Categories": ["MAP Eligible", "Cost Allocation Tagging"],
            "Amount": "5000.00",
            "OptimizedAmount": "5000.00",
            "PotentialSavingsAmount": "0.00",
            "Optimizations": []
        }
    ]
}
},
"Project": {
    "ExpectedCustomerSpend": [
        {
            "Amount": "28000.00",
            "CurrencyCode": "USD",
            "EstimationUrl": "https://calculator.aws/#/estimate?id=abc123",
            "Frequency": "Monthly",
            "TargetCompany": "AWS"
        }
    ]
},
"RelatedEntityIds": {
    "AwsProducts": [
        "AmazonEC2",
        "AWSLambda",
        "EBS",
        "RDS"
    ]
}
}

```

Poin Utama: Kedua sumber hadir: AWS rekomendasi (total \$28K) dan kalkulator Mitra (total \$32.5K). AWS rekomendasi menunjukkan 4 produk; Kalkulator mitra menunjukkan 4 produk (3 tumpang tindih). Mitra dapat membandingkan AWS rekomendasi dengan perkiraan terperinci mereka. AWS total tersedia melalui `Project.ExpectedCustomerSpend[].Amount`.

Praktik Terbaik

Menangani Nilai Null

1. **EstimationUrl Bidang:** Berharap EstimationUrl menjadi nol untuk sebagian besar Mitra. Ini adalah perilaku standar dan tidak boleh diperlakukan sebagai kesalahan. Tampilkan opsi entri MRR manual sebagai jalur utama.
2. **Kategori Hilang:** Jika tidak TotalAmountByCategory ada untuk AWS sumber, ini menunjukkan rekomendasi per produk tidak tersedia. Rincian kategori tersedia untuk Partner-sourced data saat URL Kalkulator Harga disediakan.

Rekomendasi Optimasi

1. **Tampilkan Wawasan yang Dapat Ditindaklanjuti:** Tampilkan deskripsi pengoptimalan dan jumlah penghematan secara jelas untuk membantu Mitra memahami peluang pengurangan biaya.
2. **Penghematan Agregat:** Jumlahkan PotentialSavingsAmount seluruh produk untuk menunjukkan potensi pengoptimalan total.
3. **Prioritaskan berdasarkan Dampak:** Urutkan pengoptimalan berdasarkan SavingsAmount untuk membantu Mitra fokus pada rekomendasi dengan dampak tertinggi.

Pemantauan untuk Pembaruan

1. **Poll Berkala:** Polling GetAwsOpportunitySummary secara berkala (direkomendasikan: setiap 5 menit) selama alur kerja pembuatan peluang.
2. **Tangani Pembuatan Async:** Wawasan Ukuran Kesepakatan mungkin tidak segera tersedia setelah pembuatan peluang. Menerapkan status pemuatan yang sesuai dan coba kembali logika.

Pola Integrasi API

1. **Memanfaatkan Integrasi yang Ada:** Mitra yang sudah menelepon GetAwsOpportunitySummary hanya perlu menggunakan bidang tambahan dalam respons—tidak diperlukan panggilan API baru.
2. **Degradasi Anggun:** Desain UI untuk menangani skenario di mana data Ukuran Kesepakatan belum tersedia atau tidak lengkap.

Kualitas dan Akurasi Data

1. Validasi URL Kalkulator Harga: Pastikan URL valid sebelum dikirimkan untuk menghindari tanggapan nol. `EstimationUrl`
2. Berikan Rich Opportunity Details: Informasi peluang yang lebih lengkap (detail pelanggan, deskripsi proyek, persyaratan teknis) menghasilkan rekomendasi AI yang lebih akurat.
3. Varians Tinjauan: Ketika perkiraan mitra berbeda secara signifikan dari AWS rekomendasi, tinjau detail peluang untuk memastikan semua konteks yang relevan ditangkap.

Praktik terbaik

Bereaksi terhadap peristiwa

Saat menangani peristiwa dari API Pusat AWS Mitra, pastikan logika pemrosesan Anda idempoten untuk menangani kejadian duplikat. Alih-alih membuat [GetOpportunity](#) panggilan langsung untuk setiap acara, pertimbangkan untuk mengelompokkan atau secara selektif mengambil detail berdasarkan kebutuhan aplikasi Anda. [Untuk operasi tanpa gangguan, waspadalah terhadap Kuota.](#)

Menerapkan penguncian optimis

Penguncian optimis mencegah penggantian data yang tidak diinginkan selama pembaruan bersamaan. Berikut skenario tipikal:

1. Mitra mengambil kesempatan dari sistem CRM mereka.
2. Pengguna A memperbarui kesempatan di AWS Partner Central.
3. Pengguna B memperbarui kesempatan yang sama pada saat yang sama melalui integrasi CRM.
4. Jika data berubah, sistem CRM mencoba mengunggah data tetapi mengembalikan file. `ConflictException`
5. Pengguna meninjau kesalahan dan secara manual menyelesaikan data yang bertentangan.

Untuk menghindari skenario ini, semua [UpdateOpportunity](#) permintaan harus menyertakan `LastModifiedDate` parameter, yang dapat Anda peroleh dari sebelumnya [CreateOpportunity](#), [UpdateOpportunity](#), dan [GetOpportunity](#) tindakan. Pembaruan hanya berhasil jika `LastModifiedDate` cocok dengan sistem kami. Jika tidak, Anda harus mengambil yang terbaru `LastModifiedDate` menggunakan [GetOpportunity](#) dan mencoba kembali pembaruan.

Sinkronisasi data antara CRM dan AWS Pusat Partner

Sangat penting untuk menjaga sistem Anda tetap sinkron dengan data terbaru dari Partner Central. Berikut ini adalah dua strategi untuk memastikan sistem Anda mencerminkan data terbaru:

Menggunakan acara (disarankan)

1. Muat data menggunakan [ListOpportunities](#).
2. Berlangganan acara peluang.
3. Menanggapi peluang atau perubahan baru.
 - Saat Anda menerima `Opportunity Created`, `Opportunity Updated`, atau `Engagement Invitation Accepted` acara, gunakan `GetOpportunity` untuk mengambil data terbaru.
 - Saat Anda menerima `Engagement Invitation Rejected` acara, hapus peluang yang sesuai.

Menggunakan `ListOpportunities` polling

1. Muat data menggunakan [ListOpportunities](#).
2. Pilih frekuensi polling, pastikan tidak terlalu sering untuk menghindari kehabisan kuota [baca](#) harian Anda.
3. Identifikasi yang terbaru `LastModifiedDate` dari data yang Anda simpan, pastikan itu berasal dari AWS.
4. Gunakan stempel waktu di `AfterLastModifiedDate` filter saat menelepon. [ListOpportunities](#)

```
{
  "FilterList": [
    {
      "Name": "AfterLastModifiedDate",
      "ValueList": [ "2023-05-01T20:37:46Z" ] // Replace with actual timestamp
    }
  ]
}
```

5. AWS akan mengembalikan peluang yang dibuat atau diperbarui setelah nilai yang ditunjukkan pada stempel waktu.
6. Ulangi semua halaman yang dikembalikan menggunakan `NextToken`, dan perbarui data sistem Anda menggunakan [GetOpportunity](#).

```
{
  "NextToken": "AAMA-EFRSN...PZa942D",
  "FilterList": [
    {
      "Name": "AfterLastModifiedDate",
      "ValueList": [ "2023-05-01T20:37:46Z" ] // Replace with actual timestamp
of your last synced data
    }
  ]
}
```

Menggunakan AWS API Manfaat Sentral Mitra

AWS Partner Central Benefits API merampingkan cara mitra menemukan, mengajukan, dan mengelola manfaat di semua Program AWS Mitra. Dengan memusatkan manajemen manfaat ke dalam satu antarmuka yang intuitif, layanan ini menciptakan meritokrasi transparan yang memberi penghargaan kepada mitra sambil menyederhanakan operasi.

API menyediakan fungsionalitas AWS API standar. Akses dengan menggunakan [Tindakan](#) API atau dengan menggunakan AWS SDK yang disesuaikan dengan bahasa pemrograman atau platform Anda. Untuk informasi selengkapnya, lihat [Memulai AWS](#) dan [Alat untuk Dibangun AWS](#).

Apa itu Manfaat?

Dalam Jaringan AWS Mitra (APN), Benefit mewakili keuntungan atau kemampuan yang dapat diperoleh mitra AWS untuk mendukung pertumbuhan bisnis dan kesuksesan pelanggan mereka. Manfaat dirancang untuk memberi penghargaan kepada mitra berdasarkan kualifikasi dan kontribusi mereka sambil memberikan nilai nyata yang membantu mitra meningkatkan praktik mereka AWS.

Manfaat dapat mengambil berbagai bentuk, termasuk:

- Insentif keuangan - Kredit, pencairan uang tunai, dan program pendanaan seperti Migration Acceleration Program (MAP) atau Marketing Development Funds (MDF)
- Manfaat akses - Pratinjau akses ke layanan, alat, atau sumber daya khusus baru
- Pengakuan - Lencana digital, sertifikasi, penunjukan kompetensi, dan status tingkat mitra
- Sumber daya teknis - Bantuan arsitektur solusi, dukungan teknis, dan ProServe keterlibatan
- Dukungan pemasaran - Co-marketing peluang, status mitra unggulan, dan publikasi kisah sukses

Menemukan Manfaat yang Tersedia

Mitra memulai perjalanan keuntungan mereka dengan menemukan manfaat apa yang tersedia bagi mereka dan memahami persyaratan kelayakan. Benefits API menyediakan kemampuan penemuan yang komprehensif melalui Benefit Discovery API.

Daftar Manfaat yang Tersedia

Mitra dapat melihat semua manfaat yang tersedia menggunakan tindakan `ListBenefits` API. Tindakan ini mengambil katalog manfaat dengan kemampuan penyaringan opsional untuk membantu mitra menemukan peluang yang relevan dengan cepat.

Untuk menemukan manfaat secara efisien, mitra dapat memfilter berdasarkan:

- Program - Filter berdasarkan program AWS mitra tertentu seperti MAP (Migration Acceleration Program), MDF (Marketing Development Funds), ISV Workload Migration, Innovation Sandbox, Proof of Concept, atau Partner Initiative Funding
- Jenis Pemenuhan - Filter berdasarkan bagaimana manfaat disampaikan: CREDITS, CASH, DISCOUNT, ACCESSRECOGNITION, atau RESOURCE
- Status - Filter berdasarkan status manfaat: ACTIVE atau DEPRECATED

`ListBenefitsAPI` mengembalikan ringkasan manfaat yang berisi informasi penting seperti ID manfaat, nama, deskripsi, program terkait, dan jenis pemenuhan. Tampilan ringkasan ini memungkinkan mitra untuk dengan cepat memindai manfaat yang tersedia dan mengidentifikasi yang paling relevan dengan tujuan bisnis mereka.

Contoh pendekatan penyaringan:

Mitra yang berfokus pada insentif keuangan dapat menyaring berdasarkan jenis pemenuhan CREDITS dan CASH untuk melihat peluang pendanaan. Mitra yang mencari pemberdayaan teknis dapat memfilter berdasarkan jenis pemenuhan RESOURCE atau ACCESS untuk menemukan manfaat dukungan teknis dan akses alat.

Melihat Informasi Manfaat Terperinci

Setelah mitra mengidentifikasi manfaat yang diminati, mereka dapat mengambil detail lengkap menggunakan tindakan `GetBenefit` API. Ini memberikan informasi yang komprehensif termasuk:

- Kriteria kelayakan - Persyaratan terperinci yang harus dipenuhi oleh mitra agar memenuhi syarat untuk mendapatkan manfaat

- Skema permintaan manfaat - Informasi spesifik dan mitra dokumentasi yang perlu diberikan saat melamar
- Rincian hibah - Apa yang akan diterima mitra setelah disetujui
- Program terkait - Program AWS mitra mana yang didukung manfaat

Skema permintaan manfaat sangat penting karena mendefinisikan struktur dan bidang yang diperlukan untuk aplikasi manfaat. Mitra harus meninjau skema ini dengan cermat sebelum membuat aplikasi manfaat untuk memastikan mereka mengumpulkan semua informasi yang diperlukan di muka.

 Important

Penentuan kelayakan untuk setiap manfaat ditangani oleh klien atau lapisan UI. Untuk manfaat terkait pendanaan, kelayakan biasanya didasarkan pada kinerja kartu skor mitra dan partisipasi program.

Topik

- [Bekerja dengan Aplikasi Manfaat](#)
- [Bekerja dengan Alokasi Manfaat](#)

Bekerja dengan Aplikasi Manfaat

Aplikasi Manfaat memodelkan permintaan mitra untuk manfaat tertentu. Ini menangkap semua informasi yang diperlukan untuk mengevaluasi dan memproses permintaan sesuai dengan kondisi khusus manfaat. Siklus hidup aplikasi manfaat berlangsung melalui beberapa status mulai dari pembuatan draf hingga persetujuan atau penolakan akhir.

Membuat Aplikasi Manfaat

Mitra memulai proses permintaan manfaat dengan membuat aplikasi manfaat menggunakan tindakan `CreateBenefitApplication` API. Saat pembuatan, aplikasi memasuki `PENDING_SUBMISSION` status, memungkinkan mitra untuk menyiapkan informasi lengkap sebelum mengirimkan untuk ditinjau.

Saat membuat aplikasi manfaat, mitra harus menyediakan:

- Benefit Identifier - ID atau ARN dari manfaat yang diminta

- Jenis Pemenuhan - Bagaimana mitra mengharapkan untuk menerima manfaat (CREDITS,,,CASH, DISCOUNT ACCESSRECOGNITION, atau) RESOURCE
- Detail Aplikasi Manfaat - Dokumen JSON yang berisi informasi khusus manfaat sebagaimana didefinisikan oleh skema permintaan manfaat
- Token Klien - Token idempotensi unik untuk mencegah pengiriman duplikat

Mitra secara opsional dapat menyediakan:

- Nama dan Deskripsi - Human-readable pengidentifikasi untuk aplikasi
- Kontak Mitra - Informasi kontak untuk orang yang mengelola permintaan manfaat ini (maksimal 1 kontak)
- Lampiran File - Dokumentasi pendukung seperti rencana proyek, SOW, bukti biaya, atau survei kepuasan pelanggan (maksimum 10 file)
- Sumber Daya Terkait - Tautan ke peluang terkait atau alokasi manfaat yang ada (maksimum 10 sumber daya)
- Tag - Key-value pasangan untuk organisasi sumber daya dan pelacakan (maksimum 200 tag)

CreateBenefitApplicationAPI melakukan validasi lunak, hanya memeriksa jenis dan pola bidang dasar. Validasi logika bisnis penuh terjadi selama pengiriman, memungkinkan mitra untuk menyimpan aplikasi yang tidak lengkap dan kembali nanti untuk menyelesaikannya.

Praktik Terbaik: Mitra harus menggunakan skema permintaan manfaat GetBenefit untuk memahami dengan tepat informasi apa yang diperlukan sebelum membuat aplikasi. Ini mengurangi kemungkinan kegagalan pengiriman karena data yang hilang atau tidak valid.

Memperbarui Aplikasi Manfaat

Mitra dapat memodifikasi draf aplikasi manfaat menggunakan tindakan UpdateBenefitApplication API. Hal ini memungkinkan mitra untuk memperbaiki informasi, menambahkan dokumentasi, atau memperbaiki kesalahan sebelum pengiriman.

Saat memperbarui aplikasi manfaat, mitra harus menyediakan:

- Identifier - ID atau ARN dari aplikasi manfaat untuk diperbarui
- Revisi - Nomor revisi saat ini untuk penguncian optimis
- Detail Aplikasi Manfaat - Dokumen JSON yang lengkap dan diperbarui

Mitra harus menyerahkan objek aplikasi manfaat lengkap, meskipun hanya bidang tertentu yang berubah. Praktik terbaik adalah terlebih dahulu mengambil detail aplikasi terbaru menggunakan `GetBenefitApplication`, memodifikasi bidang yang diperlukan, kemudian mengirimkan muatan lengkap yang diperbarui ke `UpdateBenefitApplication`.

Optimistic Locking: Bidang revisi memastikan bahwa pembaruan diterapkan hanya jika aplikasi tidak berubah sejak terakhir diambil. Jika revisi tidak cocok dengan nilai database saat ini, pembaruan ditolak dengan kesalahan konflik. Ini mencegah mitra dari secara tidak sengaja menimpa perubahan yang dibuat oleh proses atau sistem lain.

Pembaruan hanya dapat dilakukan saat aplikasi dalam `PENDING_SUBMISSION` status. Setelah dikirimkan, aplikasi memasuki proses peninjauan dan tidak dapat lagi diperbarui melalui API ini.

Mengaitkan Sumber Daya dengan Aplikasi Manfaat

Mitra dapat menautkan aplikasi manfaat ke sumber daya terkait menggunakan tindakan `AssociateBenefitApplicationResource` API. Ini menciptakan hubungan yang berharga antara manfaat dan konteks bisnis di mana mereka digunakan.

Jenis sumber daya yang didukung:

- **PELUANG** - Tautkan aplikasi manfaat ke peluang pelanggan tertentu di. Ini sangat berguna untuk manfaat khusus peluang seperti pendanaan MAP atau kredit POC yang terkait dengan keterlibatan pelanggan.
- **BENEFIT_ALLOCATION** - Tautkan ke alokasi manfaat yang ada, memungkinkan mitra untuk merantai manfaat atau menunjukkan bagaimana manfaat sebelumnya dimanfaatkan

Asosiasi sumber daya hanya dapat terjadi sebelum pengiriman. Setelah aplikasi manfaat diajukan (status berubah menjadi `IN_REVIEW`), tidak ada asosiasi lebih lanjut yang diizinkan. Mitra dapat mengaitkan hingga 10 sumber daya dengan setiap aplikasi manfaat.

Untuk memisahkan sumber daya, mitra menggunakan tindakan `DisassociateBenefitApplicationResource` API. Seperti asosiasi, disosiasi hanya dapat terjadi dalam `PENDING_SUBMISSION` status.

Important

Mitra harus memiliki izin yang sesuai untuk mengakses aplikasi manfaat dan membaca sumber daya tertentu yang terkait. API memvalidasi izin ini selama operasi asosiasi.

Mengirimkan Aplikasi Manfaat

Ketika aplikasi manfaat selesai dan siap untuk AWS ditinjau, mitra mengirimkannya menggunakan tindakan `SubmitBenefitApplication` API. Pengajuan memicu transisi status dari `PENDING_SUBMISSION` ke `IN_REVIEW` dan memulai alur kerja persetujuan khusus manfaat.

Setelah diserahkan, API melakukan validasi komprehensif termasuk:

- Validasi bidang wajib - Memastikan semua bidang wajib dalam detail aplikasi manfaat hadir
- Validasi format bidang - Memverifikasi bahwa nilai bidang cocok dengan pola dan tipe data yang diharapkan
- Validasi aturan bisnis - Menerapkan logika dan kendala bisnis khusus manfaat
- Validasi sumber daya - Mengonfirmasi bahwa semua sumber daya terkait valid dan dapat diakses
- Validasi file - Memverifikasi bahwa semua lampiran file telah menyelesaikan pemrosesan dengan sukses

Jika validasi gagal, API mengembalikan kode kesalahan `ValidationException` terperinci dan pesan yang menunjukkan bidang mana yang memerlukan koreksi. Mitra harus mengatasi masalah ini menggunakan `UpdateBenefitApplication` sebelum mencoba pengiriman lagi.

Persyaratan Pemrosesan File: Aplikasi manfaat tidak dapat dikirimkan jika ada file terlampir yang masih dalam `PENDING` status. Mitra harus menunggu pemrosesan file selesai (perubahan status ke `SUCCEEDED`) sebelum mengirimkan. Jika file gagal diproses (status `FAILED`), mitra harus mengunggah versi yang dikoreksi.

Setelah pengiriman berhasil:

- Status aplikasi berubah menjadi `IN_REVIEW`
- Tim pemilik manfaat diberitahu tentang pengajuan baru
- Mitra tidak dapat lagi memodifikasi detail aplikasi melalui API pembaruan standar
- Aplikasi memasuki alur kerja persetujuan yang ditentukan yang dapat mencakup persetujuan bisnis, persetujuan teknis, dan tahap persetujuan keuangan

Mitra dapat melacak kemajuan pengiriman dengan mengambil aplikasi dan memantau bidang Tahap, yang menunjukkan tahap persetujuan saat ini (misalnya, “Persetujuan Bisnis”, “Persetujuan Teknologi”, “Persetujuan Keuangan”).

Mengelola Aplikasi yang Dikirim

Setelah dikirimkan, mitra memiliki opsi terbatas untuk mengelola aplikasi manfaat, tetapi API menyediakan tindakan spesifik untuk skenario umum.

Mengingat Aplikasi Manfaat

Jika mitra menemukan kesalahan atau perlu membuat perubahan setelah pengiriman, mereka dapat mengingat aplikasi menggunakan tindakan `RecallBenefitApplication` API. Recall mentransisikan aplikasi kembali ke `PENDING_SUBMISSION` status, memungkinkan pembaruan dan pengiriman ulang.

Saat mengingat aplikasi, mitra harus menyediakan:

- Identifier - ID atau ARN dari aplikasi manfaat untuk ditarik kembali
- Alasan - Penjelasan opsional untuk penarikan (maksimum 1000 karakter)

Bidang alasan memungkinkan penelusuran yang lebih baik dan membantu AWS memahami masalah umum yang mengarah pada penarikan kembali, menginformasikan perbaikan masa depan untuk proses aplikasi manfaat.

Setelah penarikan kembali, mitra dapat menggunakan `UpdateBenefitApplication` untuk membuat perubahan yang diperlukan, lalu mengirim ulang menggunakan `SubmitBenefitApplication` saat siap.

Important

Ingat biasanya hanya tersedia selama tahap peninjauan awal. Aplikasi yang telah berkembang ke tahap persetujuan selanjutnya atau telah disetujui mungkin tidak memenuhi syarat untuk ditarik kembali.

Mengubah Aplikasi Manfaat

Untuk koreksi kecil setelah pengiriman, mitra dapat menggunakan tindakan `AmendBenefitApplication` API untuk memperbarui bidang tertentu tanpa mengingat seluruh aplikasi. Ini sangat berguna ketika AWS pengulas meminta klarifikasi atau koreksi selama proses peninjauan.

Amandemen menggunakan Patch-style pendekatan JSON di mana mitra menentukan:

- Path - Ekspresi JsonPath mengidentifikasi bidang untuk memperbarui (misalnya,) `$.CreditDisbursementDetails.AwsAccountIdForCredits`
- Nilai - Nilai baru untuk bidang
- Operasi - Operasi untuk melakukan (saat REPLACE ini hanya didukung)

Mitra dapat mengirimkan hingga 10 amandemen dalam satu panggilan API. Setiap amandemen harus menyertakan nomor revisi yang diperbarui untuk penguncian optimis.

Amandemen harus digunakan untuk koreksi kecil. Untuk perubahan besar, mitra harus menggunakan `RecallBenefitApplication` untuk mengembalikan aplikasi ke status draf untuk pembaruan komprehensif.

Membatalkan Aplikasi Manfaat

Jika mitra tidak lagi membutuhkan manfaat atau ingin menarik aplikasi mereka, mereka dapat membatalkannya menggunakan tindakan `CancelBenefitApplication` API. Pembatalan mentransisikan aplikasi ke `CANCELED` status, mengakhiri proses aplikasi secara permanen.

Saat membatalkan aplikasi, mitra harus menyediakan:

- Identifier - ID atau ARN dari aplikasi manfaat untuk dibatalkan
- Alasan - Penjelasan opsional untuk pembatalan (maksimum 1000 karakter)

Setelah dibatalkan, aplikasi tidak dapat diaktifkan kembali. Jika pasangan kemudian memutuskan mereka membutuhkan manfaat, mereka harus membuat aplikasi manfaat baru.

Alasan umum untuk pembatalan meliputi:

- Peluang pelanggan hilang atau tertunda
- Kendala kapasitas mitra berubah
- Prioritas bisnis bergeser
- Manfaat tidak lagi diperlukan untuk tujuan yang dimaksudkan

Melihat Detail Aplikasi Manfaat

Mitra dapat mengambil informasi lengkap tentang aplikasi manfaat menggunakan tindakan `GetBenefitApplication` API. Ini memberikan pandangan komprehensif dari aplikasi termasuk:

Metadata Aplikasi:

- Pengenal unik (ID) dan Nama Sumber Daya Amazon (ARN)
- Pengidentifikasi manfaat terkait
- Nama dan deskripsi aplikasi
- Status saat ini (PENDING_SUBMISSION, IN_REVIEW, ACTION_REQUIRED, APPROVED, REJECTED, atau CANCELED)
- Tahap pemrosesan saat ini

Informasi Status:

- Alasan Status - Human-readable penjelasan status saat ini
- Kode Alasan Status - Kode terstruktur yang menunjukkan masalah atau persyaratan tertentu (mis., “Daftar Periksa Tidak Lengkap”, “Rencana Proyek Hilang”, “Desain Menang Hilang”)

Konten Aplikasi:

- Detail aplikasi manfaat lengkap dokumen JSON
- Informasi kontak mitra
- Lampiran file dengan status pemrosesan
- Sumber daya terkait (peluang atau alokasi)
- Tag sumber daya

Informasi Audit:

- Stempel waktu pembuatan
- Stempel waktu modifikasi terakhir
- Nomor revisi saat ini

Kode alasan status sangat berharga ketika aplikasi dalam ACTION_REQUIRED status. Kode-kode ini memberikan panduan spesifik dan dapat ditindaklanjuti tentang apa yang perlu ditangani oleh mitra agar aplikasi dapat dilanjutkan.

Daftar Aplikasi Manfaat

Mitra dapat melihat semua aplikasi manfaat mereka menggunakan tindakan `ListBenefitApplications` API. Ini mengembalikan daftar ringkasan aplikasi paginasi dengan kemampuan penyaringan yang kuat.

Mitra dapat memfilter aplikasi dengan:

- Program - Lihat aplikasi untuk program tertentu (MAP, MDF, Sandbox, POC, dll.)
- Jenis Pemenuhan - Filter dengan metode pengiriman (CREDITS,CASH,ACCESS, dll.)
- Benefit Identifier - Lihat aplikasi untuk manfaat tertentu
- Status - Filter berdasarkan status aplikasi (PENDING_SUBMISSION,IN_REVIEW,APPROVED,REJECTED,CANCELED)
- Tahap - Filter berdasarkan tahap persetujuan (Persetujuan Bisnis, Persetujuan Teknologi, Persetujuan Keuangan)
- ARN Sumber Daya Terkait - Temukan aplikasi yang terkait dengan peluang atau alokasi tertentu

Tanggapan daftar mencakup informasi ringkasan penting seperti:

- ID Aplikasi, ARN, dan nama
- ID manfaat terkait
- Program dan jenis pemenuhan
- Status dan panggung saat ini
- Cap waktu pembuatan dan modifikasi
- Sumber daya terkait
- Nomor revisi saat ini
- Bidang yang dipilih dari detail aplikasi manfaat (sebagaimana didefinisikan oleh pemilik manfaat)

Mitra dapat mengonfigurasi pengurutan hasil menggunakan parameter `Urutkan`, dengan opsi untuk mengurutkan berdasarkan tanggal pembuatan, status, program, atau pengenalan manfaat dalam urutan naik atau turun.

Dashboard Building: `ListBenefitApplications` API dirancang untuk mendukung pembuatan dasbor mitra. Dengan memfilter dan menyortir aplikasi, mitra dapat membangun tampilan seperti:

- Aplikasi yang membutuhkan tindakan (ACTION_REQUIREDstatus)
- Aplikasi yang baru saja dikirimkan (diurutkan berdasarkan tanggal pembuatan, IN_REVIEW status)
- Aplikasi yang disetujui menunggu alokasi (APPROVEDstatus)
- Aplikasi berdasarkan program (disaring oleh program tertentu)

Bekerja dengan Alokasi Manfaat

Alokasi Manfaat merupakan manfaat yang telah diberikan kepada mitra. Ketika aplikasi manfaat disetujui, AWS buat satu atau lebih alokasi manfaat yang memberi mitra kredit aktual, pendanaan, akses, atau pemenuhan manfaat lainnya.

Memahami Alokasi Manfaat

Alokasi manfaat dapat mewakili berbagai jenis pemenuhan:

- Financial Disbursements (CASH) - Pembayaran keuangan langsung kepada mitra dengan pelacakan pencairan
- AWS Kredit (KREDIT) - Kode kredit berlaku untuk AWS akun dengan pelacakan penggunaan
- Alokasi Konsumabel - jumlah Pre-approved pendanaan atau dompet dengan pelacakan pemanfaatan
- Access Grants (ACCESS) - Akses ke sistem, alat, atau sumber daya
- Pengakuan (RECOGNITION) - Lencana digital, sertifikasi, atau penunjukan status
- Sumber Daya (SUMBER DAYA) - Dukungan teknis, layanan konsultasi, atau keterlibatan ProServe

Setiap alokasi manfaat memiliki siklus hidup yang dilacak melalui bidang Statusnya:

- AKTIF - Alokasi tersedia untuk digunakan
- TIDAK AKTIF - Alokasi sementara tidak tersedia
- DIPENUHI - Alokasi telah sepenuhnya digunakan atau dikirimkan

Alokasi manfaat juga mencakup informasi temporal yang menentukan kapan mereka menjadi efektif (StartsAt) dan kapan mereka kedaluwarsa (ExpiresAt), memungkinkan mitra untuk memahami kerangka waktu untuk memanfaatkan manfaat.

Daftar Alokasi Manfaat

Mitra dapat melihat semua alokasi manfaat mereka menggunakan tindakan `ListBenefitAllocations` API. Ini mengembalikan daftar ringkasan alokasi paginasi dengan kemampuan penyaringan yang komprehensif.

Mitra dapat memfilter alokasi dengan:

- Pengenal Manfaat - Lihat alokasi untuk manfaat tertentu
- Benefit Application Identifier - Lihat alokasi yang dihasilkan dari aplikasi tertentu
- Jenis Pemenuhan - Filter berdasarkan jenis alokasi (CREDITS,, CASHACCESS, dll.)
- Status - Filter berdasarkan status alokasi (ACTIVE,INACTIVE,FULFILLED)

Tanggapan daftar mencakup ringkasan alokasi yang berisi:

- ID Alokasi, ARN, dan nama
- ID manfaat terkait dan ID aplikasi manfaat (jika ada)
- Jenis pemenuhan
- Status saat ini
- Stempel waktu pembuatan
- Tanggal mulai

Tampilan daftar ini memungkinkan mitra untuk membangun dasbor pelacakan alokasi dan mengidentifikasi:

- Alokasi aktif tersedia untuk penggunaan segera
- Alokasi mendekati kedaluwarsa
- Alokasi terpenuhi untuk pelacakan historis
- Nilai alokasi total menurut program atau jenis pemenuhan

Melihat Informasi Alokasi Terperinci

Partner dapat mengambil detail alokasi lengkap menggunakan tindakan `GetBenefitAllocation` API. Ini memberikan informasi komprehensif tentang alokasi termasuk rincian spesifik pemenuhan yang bervariasi berdasarkan jenis alokasi.

Informasi Alokasi Inti:

- Pengenal unik (ID) dan Nama Sumber Daya Amazon (ARN)
- Nama dan status alokasi
- Alasan status (penjelasan status saat ini)
- ID manfaat terkait dan ID aplikasi manfaat
- Jenis pemenuhan
- Cap waktu pembuatan, pembaruan, mulai, dan kedaluwarsa

Detail Pemenuhan (Type-Specific):

FulfillmentDetail Bidang berisi struktur informasi yang berbeda berdasarkan jenis pemenuhan:

Detail Pencairan Uang Tunai (Jenis TUNAI)

Untuk pembayaran tunai langsung, rincian pemenuhan meliputi:

- Jumlah yang Disbursed - Jumlah total dana yang dicairkan, termasuk nilai nilai dan kode mata uang
- Detail Penerbitan (jika berlaku) - Informasi tentang pesanan pembelian atau acara penerbitan:
 - ID Penerbitan - Pengenal unik untuk pencairan
 - Jumlah Penerbitan - Jumlah dalam mata uang lokal
 - Diterbitkan Pada - Stempel waktu pencairan

Struktur ini mendukung pencairan tunggal dan skenario pendanaan yang telah disetujui sebelumnya di mana beberapa penerbitan dapat terjadi terhadap alokasi tunggal.

Rincian Kredit (Jenis KREDIT)

Untuk AWS kredit, rincian pemenuhan meliputi:

- Jumlah yang Dialokasikan - Total jumlah kredit yang dialokasikan
- Jumlah yang Diterbitkan - Total jumlah kredit yang diterbitkan (mungkin kurang dari yang dialokasikan untuk penerbitan bertahap)
- Kode Kredit - Array rincian kode kredit individu:
 - AWS ID Akun - Akun tempat kredit diterapkan

- Nilai - Nilai moneter dari kode kredit
- AWS Kode Kredit - String kode kredit yang sebenarnya
- Status - Status kode kredit (ACTIVE, INACTIVE, FULFILLED)
- Diterbitkan Pada - Ketika kode kredit dikeluarkan
- Kedaluwarsa Pada - Saat kode kredit kedaluwarsa

Pelacakan kredit terperinci ini memungkinkan mitra untuk memantau pemanfaatan kredit di beberapa akun dan memahami kredit mana yang tersedia, digunakan sebagian, atau dikonsumsi sepenuhnya.

Detail Alokasi Habis Pakai () Pre-Approved Amounts/Wallets

Untuk kumpulan pendanaan yang telah disetujui sebelumnya, rincian pemenuhan meliputi:

- Jumlah yang Dialokasikan - Jumlah total dalam alokasi
- Jumlah Sisa - Jumlah yang tidak digunakan masih tersedia
- Jumlah yang Digunakan - Jumlah yang sudah dikonsumsi
- Detail Penerbitan (jika berlaku) - Informasi pesanan pembelian untuk alokasi

Alokasi yang dapat dikonsumsi memungkinkan mitra untuk menarik dana sesuai kebutuhan untuk kegiatan yang memenuhi syarat, dengan melacak saldo yang tersisa secara real-time.

Detail Akses (Tipe ACCESS)

Untuk hibah akses, rincian pemenuhan meliputi:

- Deskripsi - Penjelasan terperinci tentang akses yang diberikan dan cara memanfaatkannya

Alokasi akses mungkin menyediakan akses ke layanan pratinjau, alat khusus, atau sumber daya terbatas. Bidang deskripsi memberikan instruksi tentang bagaimana mitra dapat mengaktifkan dan menggunakan akses yang diberikan.

Menerapkan Alokasi untuk Aplikasi Manfaat Baru

ApplicableBenefitIds Bidang dalam alokasi manfaat mengidentifikasi manfaat lain yang dapat memanfaatkan alokasi ini. Hal ini memungkinkan rantai manfaat di mana mitra dapat menggunakan alokasi yang ada untuk mengajukan manfaat tambahan.

Misalnya, pasangan mungkin:

1. Menerima alokasi pendanaan MAP yang telah disetujui sebelumnya
2. Buat aplikasi manfaat untuk proyek migrasi pelanggan individu
3. Kaitkan aplikasi tersebut dengan alokasi yang telah disetujui sebelumnya
4. Menarik dana dari alokasi saat proyek disetujui

Pendekatan ini merampingkan proses aplikasi manfaat untuk mitra dengan pendanaan yang telah disetujui sebelumnya, mengurangi siklus peninjauan untuk permintaan proyek individu.

Menggunakan AWS API Saluran Pusat Mitra

AWS Partner Central Channel API memungkinkan Anda mengelola bisnis penjualan AWS kembali resmi Anda secara terprogram. API ini memungkinkan Penyedia AWS Solusi, AWS Distributor, dan Penjual AWS Distribusi untuk:

- Laporkan akun AWS manajemen yang digunakan untuk program penjualan kembali ke Partner Central
- Kirim undangan ke AWS akun untuk menerima asosiasi mitra
- Buat hubungan yang diperlukan untuk melakukan onboard AWS akun pelanggan akhir yang dijual kembali dan menerapkan diskon mitra

Dengan API Saluran Pusat AWS Mitra, Anda dapat membangun otomatisasi khusus untuk:

- Mengonboard AWS akun milik mitra baru ke program penjualan kembali APN
- Mengonboard akun pelanggan akhir baru untuk menjual kembali program
- Mempercepat kualifikasi orientasi pelanggan dan discount

Bekerja dengan Akun Manajemen Program (PMA)

Akun Manajemen Program (PMA) mengaitkan akun AWS manajemen Anda dengan pendaftaran Program Saluran Anda. Akun manajemen program diaktifkan swalayan menggunakan jabatan saluran. Saat Anda membuat dan mengaktifkan PMA, manfaat dan diskon program saluran diterapkan ke semua faktur yang dikirimkan ke akun AWS manajemen terkait. Gunakan API ini untuk melaporkan dan mengelola AWS akun yang menangani penjualan kembali dan Diskon Saluran:

- `CreateProgramManagementAccount`: Tambahkan akun baru untuk manajemen penagihan pelanggan
- `UpdateProgramManagementAccount`: Ubah akun yang ada
- `DeleteProgramManagementAccount`: Hapus akun dari program penjualan kembali
- `ListProgramManagementAccounts`: Lihat akun yang ada dan statusnya

Bekerja dengan jabat tangan saluran

Jabat tangan saluran memungkinkan persetujuan bersama yang aman untuk operasi manajemen saluran yang kritis. AWS Partner Central menggunakan jabat tangan saluran untuk memverifikasi persetujuan dari akun AWS manajemen untuk tiga tujuan berbeda: mengaktifkan Akun Manajemen Program (PMA), menetapkan periode layanan, dan mengakhiri periode layanan lebih awal. Ada tiga tipe:

- `PROGRAM_MANAGEMENT_ACCOUNT`: Manajemen Program Jabat tangan akun memverifikasi persetujuan saat mengaktifkan PMA untuk menerapkan manfaat program saluran
- `START_SERVICE_PERIOD`: Jabat tangan pembuatan periode layanan menetapkan kesepakatan bersama untuk komitmen transfer penagihan dengan periode pemberitahuan minimum atau persyaratan tetap
- `REVOKE_SERVICE_PERIOD`: Jabat tangan penghentian periode layanan memungkinkan penghentian dini periode layanan aktif ketika kedua belah pihak menyetujui. Semua jenis jabat tangan memerlukan penerimaan dari akun AWS manajemen target untuk diterapkan.

Gunakan API ini untuk mengelola jabat tangan saluran untuk aktivasi PMA dan manajemen periode layanan:

- `CreateChannelHandshake`: Buat jabat tangan untuk undangan PMA, pembentukan periode layanan, atau penghentian periode layanan
- `AcceptChannelHandshake`: Terima jabat tangan dari AWS akun target (dapat digunakan oleh mitra atau pelanggan tergantung pada jenis jabat tangan)
- `RejectChannelHandshake`: Tolak jabat tangan dari akun target AWS
- `CancelChannelHandshake`: Batalkan jabat tangan yang tertunda sebelum diterima, ditolak, atau kedaluwarsa
- `ListChannelHandshakes`: Lacak dan lihat jabat tangan dengan pemfilteran berdasarkan jenis dan status

Bekerja dengan hubungan saluran

Hubungan memungkinkan Anda untuk mengelola pelanggan akhir, organisasi internal, atau penjual hilir. Setiap hubungan meliputi:

- Akun AWS manajemen AWS organisasi terkait
- Metadata Jaringan AWS Mitra yang diperlukan untuk kualifikasi channel discount

Saat Anda membuat hubungan, semua penggunaan dari AWS organisasi terkait menerima manfaat Program Saluran terkait. Gunakan API ini untuk melaporkan penjual dan pelanggan akhir:

- `CreateRelationship`: Pelanggan akhir baru, penjual distribusi, atau akun internal yang baru AWS
- `GetRelationship`: Lihat detail hubungan tertentu
- `ListRelationships`: Lihat semua hubungan
- `UpdateRelationship`: Ubah hubungan yang ada
- `DeleteRelationship`: Hapus hubungan

Menggunakan API Pengukuran Pendapatan

Topik

- [Bekerja dengan ID Atribusi Pendapatan Anda](#)
- [Bekerja dengan Marketplace Revenue Shares Anda](#)

Bekerja dengan ID Atribusi Pendapatan Anda

ID Atribusi Pendapatan adalah pengenal tingkat kesepakatan yang memetakan pendapatan produk AWS Marketplace Anda ke Peluang Pusat Mitra Penawaran AWS and/or AWS Marketplace tertentu. Ini dibangun di atas implementasi Pengukuran Pendapatan Mitra (PRM) tingkat produk: PRM menangkap total pendapatan yang dikaitkan untuk produk Marketplace, dan ID Atribusi Pendapatan memetakan pendapatan tersebut ke transaksi tertentu sesuai dengan persentase alokasi biaya bulanan yang Anda tentukan.

Layanan Atribusi Pendapatan mengekspos API untuk membuat, mengambil, mencantumkan, dan memperbarui ID Atribusi Pendapatan dan mengelola entri alokasi biaya bulannya secara asinkron.

Apa itu ID Atribusi Pendapatan?

ID Atribusi Pendapatan memiliki dua lapisan:

- Atribusi sumber daya yang disebut mitra, dijelaskan mitra yang diidentifikasi oleh ID `ra-` awalan (misalnya,) dan ARN. `ra-aabbccdde001` Setiap atribusi dicakup ke `Catalog` (untuk produksi, AWS Sandbox untuk pengujian) dan akun mitra.
- Satu atau lebih entri alokasi biaya bulanan setiap entri memetakan kombinasi tunggal (ID Penawaran atau ID Peluang, Bulan Penagihan) ke persentase alokasi biaya. Entri dikelola dalam batch secara asinkron melalui pola tugas alokasi.

ID Atribusi Pendapatan dapat digunakan dalam dua cara:

- Sebagai hamparan tingkat kesepakatan pada implementasi PRM tingkat produk Anda yang ada. Buat ID Atribusi Pendapatan, kaitkan Peluang and/or ACE Penawaran Marketplace yang berlaku, dan AWS petakan pendapatan produk Anda yang sudah diukur ke penawaran spesifik tersebut — tidak diperlukan perubahan pada tag sumber daya atau string agen pengguna yang ada.
- Sebagai pengidentifikasi mandiri yang digunakan secara langsung sebagai nilai tag sumber daya (`aws-apn-id=<RA ID>`) atau dalam string user-agent (`APN_1.1/pc_<RA ID>$`) untuk mengaitkan AWS konsumsi ke kesepakatan tertentu sejak awal.

Bekerja dengan ID Atribusi Pendapatan Anda

Mitra dapat mengelola ID Atribusi Pendapatan dan entri alokasi biaya bulanan mereka melalui Layanan Atribusi Pendapatan. Siklus hidup berlangsung melalui dua aliran independen: alur catatan atribusi (buat, perbarui, ambil, daftar) dan alur alokasi (mulai tugas batch, jajak pendapat untuk hasil, ambil entri individual, daftar entri untuk atribusi).

Membuat ID Atribusi Pendapatan

Langkah pertama adalah membuat ID Atribusi Pendapatan menggunakan tindakan `CreateRevenueAttribution` API. Yang dikembalikan `Id` dan `Arn` dapat digunakan segera sebagai nilai tag sumber daya atau dalam string user-agent untuk mengatribusikan konsumsi.AWS

Saat membuat ID Atribusi Pendapatan, mitra harus menyediakan:

- **Catalog**— AWS untuk produksi atau Sandbox untuk pengujian.

- **Name**— nama yang dapat dibaca manusia untuk atribusi. Harus unik dalam Katalog dan akun mitra. Maksimal 128 karakter.

Mitra secara opsional dapat menyediakan:

- **Description**— deskripsi teks bebas dari atribusi. Maksimal 1024 karakter.
- **MarketplaceProduct**— produk AWS Marketplace untuk dikaitkan dengan atribusi ini. Berikan `ProductIdentifier` (ID produk Marketplace 25 karakter) dan `TenancyModel` (`MULTI_TENANT` atau `SINGLE_TENANT`). Jika dihilangkan pada saat pembuatan, atribusi berlaku di semua konsumsi yang diukur berdasarkan ID Atribusi Pendapatan, terlepas dari produk Marketplace mana yang dibeli pelanggan — berguna ketika satu penerapan mencakup lebih dari satu daftar Marketplace.
- **Tags**— hingga 200 pasangan nilai kunci untuk organisasi sumber daya. Kunci tag harus unik dalam permintaan.

Praktik Terbaik: Berikan `MarketplaceProduct.ProductIdentifier` setiap kali transaksi Anda ditambahkan ke daftar Marketplace tertentu. Hal ini memungkinkan AWS untuk memvalidasi bahwa produk dimiliki oleh akun mitra panggilan atau oleh akun anak perusahaan yang terhubung melalui Koneksi Akun Mitra (PAC), dan memunculkan yang diselesaikan `ProductCode` dan `ProductType` (misalnya, `SaaSAMI,ML`) dalam tanggapan. Jika produk tidak dimiliki oleh akun resmi, API kembali `ValidationException` dengan alasan `PRODUCT_NOT_FOUND_OR_NOT_OWNED`.

Respons mengembalikan `MarketplaceProduct` atribut baru `Id` (format `arn:[a-z0-9]{13}`)`Arn`, diselesaikan, dan `Version` nomor awal (yang dimulai pada 1 dan kenaikan pada setiap pembaruan berikutnya).

Menambahkan entri alokasi biaya bulanan

Setelah ID Atribusi Pendapatan dibuat, mitra mengaitkan AWS Marketplace Offers and/or ACE Opportunities dengan mengirimkan sejumlah entri alokasi biaya bulanan melalui tindakan API. `StartRevenueAttributionAllocationsTask` ini adalah operasi asinkron yang menerima hingga 250 perubahan alokasi () per tugas. `CREATE` and/or `UPDATE`

Setiap entri alokasi menentukan:

- ID Penawaran atau ID Peluang — pengenal unik dari kesepakatan yang terkait. Jika Penawaran Marketplace sudah ditautkan ke Peluang ACE, mitra hanya perlu memberikan ID Penawaran; AWS secara otomatis mengaitkan pendapatan ke Peluang yang ditautkan.

- **Bulan Penagihan** — bulan kalender di mana alokasi biaya berlaku (misalnya, 2026-04).
- **Alokasi Biaya%** — porsi dari total AWS konsumsi produk yang dapat diatribusikan pada kesepakatan ini di bulan penagihan ini. Diperlukan untuk produk SaaS multi-penyewa, termasuk komponen yang dihosting mitra untuk penyebaran hibrida, tempat pelanggan berbagi infrastruktur.
- **ID AWS Rekening Pelanggan** — AWS akun pelanggan yang mengkonsumsi produk. Diperlukan untuk produk SaaS multi-penyewa.

Total Alokasi Biaya% di semua entri alokasi untuk ID Atribusi Pendapatan tertentu dan Bulan Penagihan yang sama tidak boleh melebihi 100%. Jika total melebihi 100%, entri yang melanggar ditolak selama validasi bisnis asinkron dengan kode kesalahan per rekaman.

Validasi sinkron: `StartRevenueAttributionAllocationsTask` melakukan validasi bentuk dasar secara sinkron (jenis bidang, pola, ukuran batch). Jika validasi dasar lolos, API mengembalikan status `TaskId IN_PROGRESS` with. Validasi bisnis (pemeriksaan batas, aturan kekekalan, pencarian ketergantungan terhadap Marketplace dan ACE) berjalan secara asinkron dan hasil per rekaman ditemukan melalui `GetRevenueAttributionAllocationsTask`

Untuk memantau tugas yang dikirimkan, jajak pendapat bermitra

`GetRevenueAttributionAllocationsTask` dengan Id Sumber Daya Atribusi Pendapatan atau ARN. Respons berlangsung dari `IN_PROGRESS` ke `COMPLETED` (terlepas dari apakah catatan individu berhasil atau gagal) dan kembali:

- **TaskStatus**—`IN_PROGRESS`, `COMPLETED`, atau `FAILED` (kegagalan tugas tingkat atas).
- **RecordResults**— untuk setiap catatan masukan: yang ditetapkan `AllocationId` (jika berhasil), status per-rekaman (`SUCCEEDED` atau `FAILED`), dan kode kesalahan terstruktur dan pesan jika catatan gagal.

Mitra harus segera mengambil hasil tugas. Setelah tugas selesai, hasil per catatan dapat direkonsiliasi dan entri yang gagal dapat diperbaiki dan dikirim ulang dalam tugas baru.

Mengedit entri alokasi biaya bulanan

Entri alokasi biaya dikelola per bulan penagihan dengan aturan berikut:

- Mitra dapat menambahkan entri bulanan baru untuk bulan penagihan saat ini atau yang akan datang kapan saja.

- Mitra dapat memperbarui entri bulanan untuk bulan penagihan saat ini atau yang akan datang kapan saja.
- Mitra dapat memperbarui entri bulan lalu hingga tanggal 7 bulan berjalan. Jendela ini memungkinkan mitra untuk meninjau penggunaan aktual di AWS Cost Explorer sebelum menyelesaikan alokasi bulan sebelumnya.

Setelah tanggal 7 bulan berjalan, atribusi untuk bulan penagihan sebelumnya tidak dapat lagi diubah. Pembaruan untuk entri bulan saat ini atau masa depan berlaku dari siklus penagihan bulanan berikutnya. Atribusi historis untuk bulan-bulan sebelumnya tidak dihitung ulang secara surut.

Pembaruan untuk entri alokasi dikirimkan melalui tindakan `StartRevenueAttributionAllocationsTask` API yang sama dengan `Operation` disetel ke `UPDATE` untuk setiap entri yang terpengaruh. Pola tugas `async` menangani pembaruan dan kreasi secara seragam.

Memperbarui ID Atribusi Pendapatan

Partner dapat memperbarui ID Atribusi Pendapatan yang ada menggunakan tindakan `UpdateRevenueAttribution` API. `Description` Catatan atribusi itu sendiri tidak dapat diubah —`Name`, `MarketplaceProduct`, dan nilai `tag-on-create` tidak dapat diubah setelah pembuatan. Untuk menandai ulang sumber daya, gunakan operasi AWS penandaan standar pada ARN atribusi.

Saat memperbarui ID Atribusi Pendapatan, mitra harus memberikan:

- **Catalog**— AWS atau `Sandbox`.
- **Identifier**— `ra-` ID atribusi untuk diperbarui.
- **Version**— versi atribusi saat ini untuk penguncian optimis.

Mitra secara opsional dapat menyediakan:

- **Description**— deskripsi teks gratis yang diperbarui. Maksimal 1024 karakter.
- **ClientToken**— token idempotensi untuk pembaruan.

Penguncian Optimis: `Version` Bidang memastikan pembaruan hanya berlaku jika atribusi tidak berubah sejak terakhir kali diambil. Jika yang dikirimkan `Version` tidak cocok dengan versi sumber daya saat ini, pembaruan akan ditolak `ConflictException` dan pesan termasuk nomor versi yang

dikirimkan dan saat ini. Praktik terbaik adalah mengambil versi terbaru `GetRevenueAttribution` sebelum setiap pembaruan.

Melihat detail ID Atribusi Pendapatan

Partner dapat mengambil informasi lengkap untuk satu ID Atribusi Pendapatan menggunakan tindakan `GetRevenueAttribution` API. Ini mengembalikan:

Metadata Sumber Daya:

- `Unique identifier (Id)` dan `Amazon Resource Name (Arn)`.
- `CatalogId` mana atribusi ada.
- `Name` dan `Description`.
- Yang diambil `Version` dan `LatestVersion` (gunakan yang terbaru untuk pembaruan berikutnya).

Atribut Produk Marketplace (jika `MarketplaceProduct` ditetapkan saat pembuatan):

- `ProductId`— ID produk Marketplace yang disediakan oleh mitra.
- `ProductCode`— kode produk AWS Marketplace diselesaikan dari `ProductId`.
- `ProductType`— `SaaS`, `AMI`, `Container`, `ML`, `Data`, atau `Professional Services`.
- `TenancyModel`— `MULTI_TENANT` atau `SINGLE_TENANT`.

Informasi Audit:

- `CreatedDate` dan `LastModifiedDate`.

Mitra secara opsional dapat memberikan permintaan spesifik `Version` untuk mengambil versi historis atribusi. Abaikan `Version` untuk mengembalikan yang terbaru.

Untuk mengambil entri alokasi biaya bulanan tertentu, mitra menggunakan tindakan `GetRevenueAttributionAllocation` API dengan entri. `AllocationId` Ini mengembalikan ID Penawaran atau ID Peluang, Bulan Penagihan, % Alokasi Biaya, ID AWS Akun Pelanggan, dan status entri serta informasi audit.

Daftar ID Atribusi Pendapatan

Mitra dapat melihat semua ID Atribusi Pendapatan di akun mereka menggunakan tindakan `ListRevenueAttributions` API. Ini mengembalikan daftar ringkasan atribusi paginasi dengan kemampuan pemfilteran dan pengurutan.

Mitra dapat memfilter hasil dengan:

- **Catalog**— AWS atau Sandbox (wajib).
- **Identifiers**— daftar hingga 100 ra- ID spesifik untuk diambil.
- **CreatedAfter/CreatedBefore**— filter berdasarkan rentang waktu pembuatan (inklusif).

Mitra dapat mengonfigurasi penyortiran hasil menggunakan `Sort` parameter:

- `SortBy`— `CreatedDate` atau `LastModifiedDate`.
- `SortOrder`— `ASCENDING` atau `DESCENDING`.

Tanggapan tersebut mencakup ringkasan untuk setiap atribusi: `Arn`, `Id`, `CatalogName`, `MarketplaceProduct` atribut yang diselesaikan, `CreatedDate` `LastModifiedDate`, yang terbaru `Version`, dan `TotalRevenueAttributionAssociationCount` (jumlah entri alokasi biaya bulanan aktif yang ditautkan ke atribusi).

Gunakan `MaxResults` (default 25, maksimum 100) dan `NextToken` untuk paginasi melalui set hasil besar. Tanggapan termasuk `NextToken` jika halaman tambahan tersedia.

Daftar entri alokasi biaya bulanan

Mitra dapat mencantumkan entri alokasi biaya bulanan untuk ID Atribusi Pendapatan tertentu menggunakan tindakan API `ListRevenueAttributionAllocations`. Ini mengembalikan daftar ringkasan alokasi paginasi dengan kemampuan pemfilteran.

Mitra dapat memfilter hasil dengan:

- **MarketplaceOfferId**— daftar hanya entri yang terkait dengan Penawaran Marketplace tertentu.
- **AwsPartnerCentralOpportunityId**— daftar hanya entri yang terkait dengan Peluang Pusat Mitra tertentu.
- **BillingMonth**— daftar hanya entri untuk bulan kalender tertentu.

Tanggapan tersebut mencakup informasi ringkasan untuk setiap entri: ID Penawaran atau ID Peluang terkait, ID AWS Akun Pelanggan, Bulan Penagihan,% Alokasi Biaya, nama dan status entri, dan cap waktu audit. `AllocationId`

Dashboard Building: Kombinasi `ListRevenueAttributions` dan

`ListRevenueAttributionAllocations` dirancang untuk mendukung pembuatan dasbor mitra.

Mitra dapat membangun tampilan seperti:

- Semua ID Atribusi Pendapatan dibuat dalam 30 hari terakhir, diurutkan berdasarkan tanggal pembuatan.
- Semua entri alokasi biaya bulanan untuk Penawaran Marketplace tertentu di beberapa ID Atribusi Pendapatan.
- Semua entri alokasi biaya untuk bulan penagihan saat ini, untuk memvalidasi bahwa total tidak melebihi 100% per atribusi.
- Semua ID Atribusi Pendapatan yang memiliki setidaknya satu entri alokasi aktif
(`TotalRevenueAttributionAssociationCount > 0`).

Bekerja dengan Marketplace Revenue Shares Anda

Marketplace Revenue Share adalah masukan untuk produk AWS Marketplace yang menyatakan porsi dari total pendapatan produk yang dikumpulkan melalui AWS Marketplace. AWS menggunakan masukan ini untuk mengkorelasikan Marketplace-collected pendapatan mitra dengan sinyal pendapatan lainnya.

Marketplace Revenue Share Service mengekspos API untuk membuat, mengambil, dan mencantumkan sumber daya bagi Pendapatan Marketplace, dan untuk mengelola alokasi mereka (persentase pembagian pendapatan yang dinyatakan mitra dan jendela tanggal efektif). Operasi AWS penandaan standar didukung pada sumber daya Marketplace Revenue Share.

Apa itu Marketplace Revenue Share?

Marketplace Revenue Share memiliki dua lapisan:

- Saham — masukan untuk satu produk AWS Marketplace, yang diidentifikasi oleh `MarketplaceProductId`. Ada persis satu Marketplace Revenue Share per produk.
- Satu atau lebih alokasi - setiap alokasi menyatakan `RevenueSharePercent` dan jendela tanggal efektif (`EffectiveDate`, opsional). `ExpirationDate` Sebuah saham dapat memiliki banyak alokasi dari waktu ke waktu, tetapi setiap saat hanya satu ACTIVE alokasi yang dapat berlaku.

Alokasi memiliki properti utama ini:

- **Status** — **ACTIVE** atau **INACTIVE**. Pada pembuatan, status default ke. **ACTIVE** Hanya **ACTIVE** alokasi yang berpartisipasi dalam perhitungan pendapatan dan pemeriksaan tumpang tindih.
- **Tumpang tindih invarian** — dalam satu bagian, tidak ada dua **ACTIVE** alokasi yang mungkin memiliki rentang yang tumpang tindih. [EffectiveDate, ExpirationDate] Paling banyak satu **ACTIVE** alokasi terbuka (tidakExpirationDate) diizinkan per saham kapan saja.

Bekerja dengan Marketplace Revenue Shares Anda

Mitra mengelola Marketplace Revenue Shares dan alokasi mereka melalui Marketplace Revenue Share Service. Siklus hidup berlangsung melalui dua aliran: alur berbagi (buat, ambil, daftar, tag) dan alur alokasi (buat, perbarui, hapus lunak, ambil, daftar).

Membuat Marketplace Revenue Share

Langkah pertama adalah membuat Marketplace Revenue Share untuk produk AWS Marketplace menggunakan aksi `CreateMarketplaceRevenueShare` API. Bagian diidentifikasi oleh Marketplace `ProductId` dan berfungsi sebagai wadah untuk semua alokasi berikutnya.

Saat membuat Marketplace Revenue Share, mitra harus menyediakan:

- **Catalog**— AWS untuk produksi atau Sandbox untuk pengujian.
- **ProductId**— ID produk Marketplace 25 karakter. Pola: `[a-z0-9]{25}`.

Mitra secara opsional dapat menyediakan:

- **Tags**— hingga 50 pasangan nilai kunci untuk organisasi sumber daya pada waktu pembuatan. Gunakan `TagResource` untuk menambahkan tag setelah pembuatan.
- **ClientToken**— token idempotensi untuk mencegah duplikat kreasi saat dicoba lagi. Maksimal 64 karakter. SDK menghasilkan token ini secara otomatis.

Validasi Produk: Saat pembuatan, layanan memverifikasi dengan AWS Marketplace bahwa produk dimiliki oleh akun penelepon atau oleh akun anak perusahaan yang terhubung ke penelepon melalui Koneksi Akun Mitra (PAC). Jika produk tidak ada atau akun pemilik tidak diotorisasi untuk pemanggil, API akan kembali `ValidationException` dengan alasan. `PRODUCT_NOT_FOUND_OR_NOT_OWNED`

Satu Saham Per Produk: `CreateMarketplaceRevenueShare` Panggilan kedua untuk hal yang sama `ProductId` dalam `Catalog` pengembalian yang sama `ConflictException`.

Respons mengembalikan share Arn (format `arn:{partition}:partnercentral:{region}:{account-id}:catalog/{catalog}/marketplace-revenue-share/{productId}`),
Version angka awal (yang dimulai dari 1 dan bertambah pada setiap mutasi alokasi), dan tag apa pun yang diterapkan saat pembuatan. `Catalog ProductId`

Menambahkan alokasi

Setelah `Marketplace Revenue Share` dibuat, mitra mendeklarasikan persentase bagi hasil dan jendela tanggal efektif dengan membuat alokasi melalui tindakan `API. CreateMarketplaceRevenueShareAllocation`

Saat membuat alokasi, mitra harus menyediakan:

- **Catalog**— AWS atau `Sandbox`.
- **MarketplaceRevenueShareIdentifier**— pengenalan bagian induk.
- **RevenueSharePercent**— persentase pendapatan produk yang dikumpulkan melalui `Marketplace` disediakan sebagai desimal (misalnya, `0.45` untuk 45%).
- **EffectiveDate**— tanggal kalender ketika alokasi ini menjadi efektif. Format: `YYYY-MM-DD`.

Respons mengembalikan alokasi Id (format `mrsa-[A-Za-z0-9]{13}`), status alokasi (`ACTIVE` saat pembuatan), dan pembagian induk yang terbentur `Version` (bilangan bulat versi tunggal per saham ditabrak pada setiap mutasi alokasi, digunakan untuk penguncian optimis pada pembaruan alokasi berikutnya).

Memperbarui alokasi

Mitra dapat memperbarui alokasi menggunakan tindakan `UpdateMarketplaceRevenueShareAllocation` API. Bidang yang dapat diperbarui bergantung pada apakah alokasi tersebut bertanggal di masa depan, efektif saat ini, atau bertanggal lalu.

Saat memperbarui alokasi, mitra harus menyediakan:

- **Catalog**— AWS atau `Sandbox`.
- **MarketplaceRevenueShareIdentifier**— pengenalan bagian induk.
- **AllocationId**— `mrsa`- ID alokasi untuk memperbarui.

- **MarketplaceRevenueShareVersion**— versi pangsa induk saat ini untuk penguncian optimis pada alokasi. Konflik benjolan kembali `ConflictException`.

Mitra secara opsional dapat menyediakan:

- **RevenueSharePercent**— persentase pembagian pendapatan yang diperbarui. Dapat diedit hanya pada alokasi masa depan.
- **EffectiveDate**— tanggal efektif yang diperbarui. Dapat diedit hanya pada alokasi masa depan.
- **ExpirationDate**— tanggal kedaluwarsa yang diperbarui. Dapat diedit pada alokasi masa depan (nilai apa pun > `EffectiveDate`) dan pada alokasi yang efektif saat ini atau terbuka (nilai apa pun). $\geq$ today
- **Status**— `ACTIVE` atau `INACTIVE`. Pengaturan untuk `INACTIVE` soft-menghapus alokasi. Dapat diedit hanya pada alokasi masa depan.
- **ClientToken**— token idempotensi.

Melihat detail Pembagian Pendapatan Marketplace

Partner dapat mengambil informasi lengkap untuk satu Marketplace Revenue Share menggunakan tindakan `GetMarketplaceRevenueShare` API. Ini mengembalikan `shareArn`, `CatalogProductId`, `Version` (versi induk share saat ini), `CreatedDate` `LastModifiedDate`, dan tag apa pun yang diterapkan ke share.

Partner dapat mengambil satu alokasi menggunakan aksi `GetMarketplaceRevenueShareAllocation` API dengan alokasi. Id Ini mengembalikan:

- Alokasi Id dan bagian induk. `MarketplaceRevenueShareArn`
- `RevenueSharePercent`, `EffectiveDate`, `ExpirationDate`.
- `Status`(`ACTIVE` atau `INACTIVE`).
- Bagian induk `MarketplaceRevenueShareVersion` untuk penguncian optimis pada pembaruan berikutnya.
- `CreatedDate` dan `LastModifiedDate`.

Daftar Saham Pendapatan Marketplace

Partner dapat melihat semua Marketplace Revenue Shares yang dimiliki oleh akun mereka menggunakan tindakan `ListMarketplaceRevenueShares` API. Ini mengembalikan daftar ringkasan berbagi paginasi.

Mitra dapat memfilter hasil dengan:

- **Catalog**— AWS atau Sandbox (wajib).
- **ProductIds**— daftar ID produk Marketplace tertentu untuk diambil.

Tanggapan tersebut mencakup ringkasan untuk setiap saham: `Arn`, `Catalog`, `ProductId`, yang terbaru `Version`, `CreatedDate`, dan `LastModifiedDate`.

Gunakan `MaxResults` dan `NextToken` untuk paginasi melalui set hasil yang besar.

Alokasi daftar

Partner dapat mencantumkan alokasi untuk Marketplace Revenue Share tertentu menggunakan tindakan `ListMarketplaceRevenueShareAllocations` API. Ini mengembalikan daftar ringkasan alokasi paginasi dengan kemampuan pemfilteran.

Mitra dapat memfilter hasil dengan:

- **Status**— `ACTIVE` atau `INACTIVE`.
- Rentang tanggal efektif — `EffectiveDateOnOrAfter/EffectiveDateOnOrBefore` untuk mengambil alokasi yang tanggal efektifnya termasuk dalam jendela tertentu.

Tanggapan tersebut mencakup informasi ringkasan untuk setiap alokasi: `Id`, `RevenueSharePercent`, `EffectiveDate`, `ExpirationDate`, `StatusMarketplaceRevenueShareVersion`, dan cap waktu audit.

Menandai Saham Pendapatan Marketplace

Operasi AWS penandaan standar didukung pada sumber daya Marketplace Revenue Share:

- **TagResource**— menambah atau menempa tag pada Marketplace Revenue Share. Maksimal 50 tag per saham.
- **UntagResource**— hapus tag tertentu dengan kunci.

- **ListTagsForResource**— mengambil semua tag yang saat ini diterapkan ke Marketplace Revenue Share.

Alokasi tidak mendukung tag. Tandai bagian induk untuk mengatur alokasi terkait.

Setiap operasi penandaan menerima share Arn (format: `arn:{partition}:{partnercentral}:{region}:{account-id}:catalog/{catalog}/marketplace-revenue-share/{productId}`) sebagai pengenalan sumber daya. Standar AWS tag-on-create juga didukung melalui bidang pada. Tags `CreateMarketplaceRevenueShare`

Didukung AWS Wilayah

Bagian berikut memberikan informasi tentang Wilayah AWS yang didukung.

Topik

- [Didukung AWS daerah untuk AWS API Rekening Sentral Mitra](#)
- [Didukung AWS daerah untuk AWS API Penjualan Sentral Mitra](#)
- [Didukung AWS daerah untuk AWS API Manfaat Sentral Mitra](#)
- [Didukung AWS daerah untuk AWS API Saluran Pusat Mitra](#)
- [Didukung AWS daerah untuk AWS API Pengukuran Pendapatan Pusat Mitra](#)

Didukung AWS daerah untuk AWS API Rekening Sentral Mitra

Anda dapat mengakses layanan akun AWS Partner Central dari wilayah AWS US East (Virginia N.) mana pun dengan titik akhir berikut.

```
partnercentral-account.us-east-1.api.aws
```

Didukung AWS daerah untuk AWS API Penjualan Sentral Mitra

Anda dapat mengakses layanan penjualan AWS Partner Central dari wilayah AWS US East (Virginia N.) mana pun dengan titik akhir berikut.

```
partnercentral-selling.us-east-1.api.aws
```

Didukung AWS daerah untuk AWS API Manfaat Sentral Mitra

Anda dapat mengakses layanan manfaat AWS Partner Central dari wilayah AWS US East (Virginia N.) mana pun dengan titik akhir berikut.

```
partnercentral-benefits.us-east-1.api.aws
```

Didukung AWS daerah untuk AWS API Saluran Pusat Mitra

Anda dapat mengakses layanan manajemen saluran AWS Partner Central dari wilayah komersial AWS mana pun dengan titik akhir berikut.

```
partnercentral-channel.global.api.aws
```

Menandatangani permintaan dengan Sigv4a

Karena API Saluran Pusat AWS Mitra menggunakan titik akhir global, permintaan ditandatangani dengan Signature Version 4a (Sigv4a), ekstensi SigV4 yang mendukung penandatanganan dengan kriptografi asimetris, memungkinkan tanda tangan yang dapat diverifikasi di beberapa Wilayah. AWS

Menggunakan AWS SDK atau CLI

Jika Anda menggunakan AWS SDK atau AWS CLI, penandatanganan Sigv4a ditangani secara otomatis saat Anda memanggil titik akhir global. Tidak diperlukan konfigurasi tambahan.

Menandatangani permintaan secara manual

Jika Anda menandatangani permintaan secara manual tanpa SDK, perhatikan perbedaan berikut dari SigV4 standar:

- Sigv4a memperoleh keypair Elliptic Curve Digital Signature Algorithm (ECDSA) dari kunci akses rahasia yang ada AWS, daripada menggunakan derivasi kunci. HMAC-based
- Cakupan kredensi digunakan * untuk komponen wilayah, bukan nama Wilayah tertentu, karena tanda tangan valid di seluruh Wilayah.

Untuk informasi selengkapnya tentang cara kerja Sigv4a, lihat [AWS Signature Version 4 untuk permintaan API](#). Untuk contoh kode penandatanganan Sigv4a, lihat proyek [sigv4a-signing-examples](#) di GitHub

Didukung AWS daerah untuk AWS API Pengukuran Pendapatan Pusat Mitra

Anda dapat mengakses layanan Pengukuran Pendapatan Pusat AWS Mitra dari wilayah komersial AWS mana pun dengan titik akhir berikut.

```
partnercentral-prm.global.api.aws
```

Menandatangani permintaan dengan Sigv4a

Karena layanan pengukuran pendapatan menggunakan titik akhir global, permintaan ditandatangani dengan Signature Version 4a (Sigv4a), perpanjangan SigV4 yang mendukung penandatanganan dengan kriptografi asimetris, memungkinkan tanda tangan yang dapat diverifikasi di beberapa Wilayah. AWS

Menggunakan AWS SDK atau CLI

Jika Anda menggunakan AWS SDK atau AWS CLI, penandatanganan Sigv4a ditangani secara otomatis saat Anda memanggil titik akhir global. Tidak diperlukan konfigurasi tambahan.

Menandatangani permintaan secara manual

Jika Anda menandatangani permintaan secara manual tanpa SDK, perhatikan perbedaan berikut dari SigV4 standar:

- Sigv4a memperoleh keypair Elliptic Curve Digital Signature Algorithm (ECDSA) dari kunci akses rahasia yang ada AWS, daripada menggunakan derivasi kunci. HMAC-based
- Cakupan kredensi digunakan * untuk komponen wilayah, bukan nama Wilayah tertentu, karena tanda tangan valid di seluruh Wilayah.

Untuk informasi selengkapnya tentang cara kerja Sigv4a, lihat [AWS Signature Version 4 untuk permintaan API](#). Untuk contoh kode penandatanganan Sigv4a, lihat proyek [sigv4a-signing-examples](#) di GitHub

Izin

Bagian berikut memberikan informasi tentang izin.

Topik

- [Akses untuk API Akun](#)
- [Akses untuk API Penjualan](#)
- [Akses untuk API Manfaat](#)
- [Akses untuk API Saluran](#)
- [Akses untuk API Pengukuran Pendapatan](#)

Akses untuk API Akun

Kontrol akses dan izin dikelola oleh AWS Identity and Access Management(IAM). Bagian ini memberikan panduan untuk mengonfigurasi izin yang diperlukan untuk berinteraksi dengan API Akun.

Prasyarat

Sebelum mengonfigurasi izin, pastikan bahwa Anda Akun AWS ditautkan dan bahwa Anda membuat peran dan pengguna IAM yang diperlukan. Untuk informasi selengkapnya, lihat [Pengaturan dan otentikasi](#).

Penggunaan AWS kebijakan terkelola

AWS menyediakan kebijakan terkelola yang memberikan izin yang diperlukan untuk berinteraksi dengan API Akun. Untuk menyediakan akses yang diperlukan untuk mengelola sumber daya akun, lampirkan `AWSPartnerCentralFullAccess` kebijakan ke identitas IAM Anda. Untuk informasi selengkapnya, lihat [kebijakan AWS terkelola untuk pengguna](#).

Menetapkan kebijakan untuk peran dan pengguna IAM

Ikuti langkah-langkah ini untuk menetapkan kebijakan untuk peran dan pengguna IAM:

1. Masuk ke Konsol Manajemen AWS.
2. Arahkan ke layanan IAM.
3. Pilih peran atau pengguna, dan pilih peran IAM atau pengguna yang ingin Anda lampirkan kebijakan.
4. Lampirkan `AWSPartnerCentralFullAccess` kebijakan ke peran IAM atau pengguna yang dipilih.

Untuk informasi selengkapnya, lihat [Menambahkan dan menghapus izin identitas IAM](#).

Mengelola izin menggunakan tombol kondisi

Kunci kondisi dalam kebijakan IAM memberikan izin tingkat sumber daya kapan harus menerapkan kebijakan pernyataan. Anda dapat menggunakan tombol kondisi untuk menentukan kondisi yang menentukan kapan izin tertentu diizinkan atau ditolak.

Untuk informasi selengkapnya, lihat [elemen kebijakan IAM JSON: Operator kondisi](#).

Ikhtisar tombol kondisi

Kunci syarat	Deskripsi	Tindakan yang berlaku	Nilai valid
Partnercentral:Katalog	memfilter akses berdasarkan jenis entitas katalog terkait	semua tindakan	AWS, kotak pasir

Ringkasan izin yang diperlukan

Ringkasan izin yang diperlukan

Tindakan	Deskripsi
partnercentral: AcceptConnectionInvitation	memungkinkan menerima undangan koneksi
partnercentral: AssociateAwsTrainingCertificationEmailDomain	memungkinkan mengaitkan domain email sertifikasi AWS pelatihan
partnercentral: CancelConnection	memungkinkan pembatalan koneksi
partnercentral: CancelConnectionInvitation	memungkinkan pembatalan undangan koneksi
partnercentral: CancelProfileUpdateTask	memungkinkan pembatalan tugas pembaruan profil
partnercentral: CreateConnectionInvitation	memungkinkan membuat undangan koneksi
partnercentral: CreatePartner	memungkinkan menciptakan mitra

Tindakan	Deskripsi
partnercentral: DisassociateAwsTrainingCertificationEmailDomain	memungkinkan memisahkan domain email sertifikasi AWS pelatihan
partnercentral: GetAllianceLeadContact	memungkinkan mengambil rincian kontak pemimpin aliansi
partnercentral: GetConnection	memungkinkan mengambil detail koneksi
partnercentral: GetConnectionInvitation	memungkinkan mengambil detail undangan koneksi
partnercentral: GetConnectionPreferences	memungkinkan mengambil preferensi koneksi
partnercentral: GetPartner	memungkinkan mengambil detail mitra
partnercentral: GetProfileUpdateTask	memungkinkan mengambil detail tugas pembaruan profil
partnercentral: GetProfileVisibility	memungkinkan mengambil pengaturan visibilitas profil
partnercentral: GetVerification	memungkinkan mengambil detail verifikasi
partnercentral: ListConnectionInvitations	memungkinkan daftar undangan koneksi
partnercentral: ListConnections	memungkinkan daftar koneksi
partnercentral: ListPartners	memungkinkan daftar mitra
partnercentral: PutAllianceLeadContact	memungkinkan memperbarui detail kontak pemimpin aliansi
partnercentral: PutProfileVisibility	memungkinkan memperbarui pengaturan visibilitas profil
partnercentral: RejectConnectionInvitation	memungkinkan menolak undangan koneksi
partnercentral: SendEmailVerificationCode	memungkinkan pengiriman kode verifikasi email

Tindakan	Deskripsi
partnercentral: StartProfileUpdateTask	memungkinkan memulai tugas pembaruan profil
partnercentral: StartVerification	memungkinkan memulai proses verifikasi
partnercentral: UpdateConnectionPreferences	memungkinkan memperbarui preferensi koneksi

Akses untuk API Penjualan

Kontrol akses dan izin dikelola oleh AWS Identity and Access Management(IAM). Bagian ini memberikan panduan untuk mengonfigurasi izin yang diperlukan untuk berinteraksi dengan API, termasuk izin yang diperlukan untuk mencantumkan entitas.AWS Marketplace

Prasyarat

Sebelum mengonfigurasi izin, pastikan bahwa Anda Akun AWS ditautkan ke Partner Central dan Anda membuat peran dan pengguna IAM yang diperlukan. Untuk informasi selengkapnya, lihat [Pengaturan dan Otentikasi](#).

Penggunaan AWS kebijakan terkelola

AWS menyediakan kebijakan terkelola yang memberikan izin yang diperlukan untuk berinteraksi dengan API. Untuk menyediakan akses yang diperlukan untuk mengelola peluang, lampirkan AWSPartnerCentralOpportunityManagement kebijakan ke identitas IAM Anda. Untuk informasi selengkapnya, lihat [kebijakan AWS terkelola untuk pengguna Pusat AWS Mitra](#).

AWSPartnerCentralOpportunityManagement kebijakan

Kebijakan ini memberikan akses penuh ke tindakan manajemen peluang Partner Central.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "OpportunityManagement",
```

```

    "Effect": "Allow",
    "Action": [
        "partnercentral:AcceptEngagementInvitation",
        "partnercentral:AssignOpportunity",
        "partnercentral:AssociateOpportunity",
        "partnercentral:CreateEngagement",
        "partnercentral:CreateEngagementInvitation",
        "partnercentral:CreateOpportunity",
        "partnercentral:CreateResourceSnapshot",
        "partnercentral:CreateResourceSnapshotJob",
        "partnercentral>DeleteResourceSnapshotJob",
        "partnercentral:DisassociateOpportunity",
        "partnercentral:GetAwsOpportunitySummary",
        "partnercentral:GetEngagement",
        "partnercentral:GetEngagementInvitation",
        "partnercentral:GetOpportunity",
        "partnercentral:GetResourceSnapshot",
        "partnercentral:GetResourceSnapshotJob",
        "partnercentral:ListEngagementByAcceptingInvitationTasks",
        "partnercentral:ListEngagementFromOpportunityTasks",
        "partnercentral:ListEngagementInvitations",
        "partnercentral:ListEngagementMembers",
        "partnercentral:ListEngagementResourceAssociations",
        "partnercentral:ListEngagements",
        "partnercentral:ListOpportunities",
        "partnercentral:ListResourceSnapshotJobs",
        "partnercentral:ListResourceSnapshots",
        "partnercentral:ListSolutions",
        "partnercentral:RejectEngagementInvitation",
        "partnercentral:StartEngagementByAcceptingInvitationTask",
        "partnercentral:StartEngagementFromOpportunityTask",
        "partnercentral:StartResourceSnapshotJob",
        "partnercentral:StopResourceSnapshotJob",
        "partnercentral:SubmitOpportunity",
        "partnercentral:UpdateOpportunity"
    ],
    "Resource": "*"
},
{
    "Sid": "ListingAWSMarketplaceEntities",
    "Effect": "Allow",
    "Action": ["aws-marketplace:ListEntities"],
    "Resource": "*"
},

```



```
{
  "Sid": "AWSMarketplaceOffersAccess",
  "Effect": "Allow",
  "Action": ["aws-marketplace:DescribeEntity"],
  "Resource": [
    "arn:aws:aws-marketplace::*:AWSMarketplace/Offer/*",
    "arn:aws:aws-marketplace::*:AWSMarketplace/OfferSet/*"
  ]
}
```

Kebijakan kustom

Jika kebijakan terkelola tidak memenuhi kebutuhan Anda, buat kebijakan IAM khusus yang memberikan izin yang diperlukan untuk kasus penggunaan Anda. Contoh berikut adalah kebijakan khusus yang memberikan izin untuk mencantumkan AWS Marketplace entitas:

Example Contoh kebijakan kustom

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "partnercentral:ListOpportunities",
        "aws-marketplace:ListEntities"
      ],
      "Resource": "*"
    }
  ]
}
```

Kebijakan permisif khusus

Kebijakan ini menyediakan akses luas ke tindakan penjualan Partner Central, termasuk fitur yang dapat ditambahkan di masa mendatang tanpa memerlukan pembaruan kebijakan. Dengan menggunakan aksi `wild cardpartnercentral:*`, kebijakan ini secara otomatis memberikan akses ke fitur penjualan Partner Central baru saat tersedia, sehingga mengurangi kebutuhan akan pembaruan manual. Kebijakan ini juga mencakup izin untuk berinteraksi dengan AWS Marketplace entitas, yang membantu memastikan akses dipertahankan untuk tindakan penjualan dan Marketplace.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "partnercentral:*"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeEntity"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:SearchAgreements",
        "aws-marketplace:DescribeAgreement"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals":
```

```
{
  "aws-marketplace:PartyType": "Proposer"
}
```

Menetapkan kebijakan untuk peran dan pengguna IAM

Ikuti langkah-langkah ini untuk menetapkan kebijakan untuk peran dan pengguna IAM:

1. Masuk ke Konsol Manajemen AWS.
2. Arahkan ke layanan IAM.
3. Pilih peran atau pengguna, dan pilih peran IAM atau pengguna yang ingin Anda lampirkan kebijakan.
4. Lampirkan `AWSPartnerCentralOpportunityManagement` kebijakan atau kebijakan khusus Anda ke peran atau pengguna IAM yang dipilih.

Untuk informasi selengkapnya, lihat [Menambahkan dan menghapus izin identitas IAM](#).

Mengelola izin menggunakan tombol kondisi

Kunci kondisi dalam kebijakan IAM memberikan izin tingkat sumber daya kapan harus menerapkan kebijakan pernyataan. Anda dapat menggunakan tombol kondisi untuk menentukan kondisi yang menentukan kapan izin tertentu diizinkan atau ditolak.

Untuk informasi selengkapnya, lihat [elemen kebijakan IAM JSON: Operator kondisi](#).

Ikhtisar tombol kondisi

Kunci syarat	Deskripsi	Tindakan yang berlaku	Nilai valid
Partnercentral:Katalog	memfilter akses berdasarkan jenis entitas katalog terkait	semua tindakan	AWS, kotak pasir

Kunci syarat	Deskripsi	Tindakan yang berlaku	Nilai valid
aws-pasar: PartyType	memfilter akses berdasarkan jenis partai (misalnya, pengusul)	SearchAgreements, DescribeAgreement	pengusul

Ringkasan izin yang diperlukan

Ringkasan izin yang diperlukan

Tindakan	Deskripsi
Partnercentral: CreateOpportunity	memungkinkan menciptakan peluang
Partnercentral: UpdateOpportunity	memungkinkan memperbarui peluang
Partnercentral: ListOpportunities	memungkinkan peluang daftar
Partnercentral: GetOpportunity	memungkinkan mengambil detail peluang
Partnercentral: ListSolutions	memungkinkan daftar solusi
Partnercentral: AssociateOpportunity	memungkinkan mengasosiasikan peluang dengan entitas lain
Partnercentral: DisassociateOpportunity	memungkinkan memisahkan peluang dari entitas lain
Partnercentral: AcceptEngagementInvitation	memungkinkan menerima undangan keterlibatan
Partnercentral: RejectEngagementInvitation	memungkinkan menolak undangan keterlibatan
Partnercentral: GetEngagementInvitation	memungkinkan mengambil detail undangan keterlibatan
Partnercentral: ListEngagementInvitations	memungkinkan daftar undangan keterlibatan

Tindakan	Deskripsi
Partnercentral: SubmitOpportunity	memungkinkan pengajuan peluang
Partnercentral: GetAwsOpportunitySummary	memungkinkan mengambil ringkasan AWS peluang
Partnercentral: StartProspectingFromEngagementTask	Menciptakan tugas prospeksi dari keterlibatan
Partnercentral: GetProspectingFromEngagementTask	Mengembalikan rincian tugas pencarian calon
Partnercentral: ListProspectingFromEngagementTasks	Mengembalikan daftar tugas prospeksi
aws-pasar: ListEntities	memungkinkan AWS Marketplace entitas daftar
aws-pasar: DescribeEntity	memungkinkan mendeskripsikan entitas AWS Marketplace
aws-pasar: SearchAgreements	memungkinkan pencarian perjanjian di AWS Marketplace
aws-pasar: DescribeAgreement	memungkinkan mendeskripsikan perjanjian di AWS Marketplace

Akses untuk API Manfaat

Kontrol akses dan izin dikelola oleh AWS Identity and Access Management(IAM). Bagian ini memberikan panduan untuk mengonfigurasi izin yang diperlukan untuk berinteraksi dengan API Manfaat.

Prasyarat

Sebelum mengonfigurasi izin, pastikan bahwa Anda Akun AWS ditautkan dan bahwa Anda membuat peran dan pengguna IAM yang diperlukan. Untuk informasi selengkapnya, lihat [Pengaturan dan otentikasi](#).

Penggunaan AWS kebijakan terkelola

AWS menyediakan kebijakan terkelola yang memberikan izin yang diperlukan untuk berinteraksi dengan API Manfaat. Untuk menyediakan akses yang diperlukan untuk mengelola sumber daya manfaat, lampirkan `AWSPartnerCentralFullAccess` kebijakan ke identitas IAM Anda. Untuk informasi selengkapnya, lihat [kebijakan AWS terkelola untuk pengguna](#).

Menetapkan kebijakan untuk peran dan pengguna IAM

Ikuti langkah-langkah ini untuk menetapkan kebijakan untuk peran dan pengguna IAM:

1. Masuk ke Konsol Manajemen AWS.
2. Arahkan ke layanan IAM.
3. Pilih peran atau pengguna, dan pilih peran IAM atau pengguna yang ingin Anda lampirkan kebijakan.
4. Lampirkan `AWSPartnerCentralFullAccess` kebijakan ke peran IAM atau pengguna yang dipilih.

Untuk informasi selengkapnya, lihat [Menambahkan dan menghapus izin identitas IAM](#).

Mengelola izin menggunakan tombol kondisi

Kunci kondisi dalam kebijakan IAM memberikan izin tingkat sumber daya kapan harus menerapkan kebijakan pernyataan. Anda dapat menggunakan tombol kondisi untuk menentukan kondisi yang menentukan kapan izin tertentu diizinkan atau ditolak.

Untuk informasi selengkapnya, lihat [elemen kebijakan IAM JSON: Operator kondisi](#).

Ikhtisar tombol kondisi

Kunci syarat	Deskripsi	Tindakan yang berlaku	Nilai valid
Partnercentral:Katalog	memfilter akses berdasarkan jenis entitas katalog terkait	semua tindakan	AWS, kotak pasir

Ringkasan izin yang diperlukan

Ringkasan izin yang diperlukan

Tindakan	Deskripsi
partnercentral: AmendBenefitApplication	memungkinkan mengubah aplikasi manfaat
partnercentral: AssociateBenefitApplicationResource	memungkinkan mengaitkan sumber daya dengan aplikasi manfaat
partnercentral: CancelBenefitApplication	memungkinkan pembatalan aplikasi manfaat
partnercentral: CreateBenefitApplication	memungkinkan pembuatan aplikasi manfaat
partnercentral: DisassociateBenefitApplicationResource	memungkinkan memisahkan sumber daya dari aplikasi manfaat
partnercentral: GetBenefit	memungkinkan pengambilan detail manfaat
partnercentral: GetBenefitAllocation	memungkinkan pengambilan detail alokasi manfaat
partnercentral: GetBenefitApplication	memungkinkan pengambilan detail aplikasi manfaat
partnercentral: ListBenefitAllocations	memungkinkan daftar alokasi manfaat
partnercentral: ListBenefitApplications	memungkinkan daftar aplikasi manfaat
partnercentral: ListBenefits	memungkinkan daftar manfaat
partnercentral: RecallBenefitApplication	memungkinkan penarikan kembali aplikasi manfaat
partnercentral: SubmitBenefitApplication	memungkinkan pengajuan aplikasi manfaat
partnercentral: UpdateBenefitApplication	memungkinkan memperbarui aplikasi manfaat

Akses untuk API Saluran

Kontrol akses dan izin dikelola oleh AWS Identity and Access Management(IAM). Bagian ini memberikan panduan untuk mengonfigurasi izin yang diperlukan untuk berinteraksi dengan API Saluran Pusat AWS Mitra.

Pengaturan akun manajemen saluran

Fitur manajemen saluran di AWS Partner Central memerlukan peran IAM untuk digunakan dalam dua jenis AWS akun:

1. [AWS Mitra Central terhubung Akun AWS:](#)

Ini Akun AWS terkait dengan akun AWS Partner Network dan Partner Central Anda. Hanya ada satu AWS Partner Central yang ditautkan Akun AWS per mitra. Sebagai aktivitas penyiapan satu kali, Anda akan membuat peran IAM dalam hal ini Akun AWS menggunakan kebijakan terkelola saluran Partner Central.

2. [Akun manajemen program Akun AWS:](#)

Ini Akun AWS digunakan sebagai Bill-Transfer akun untuk mengelola penagihan dan pembayaran untuk konsumsi pelanggan akhir. Akun AWS Ini dilaporkan sebagai akun manajemen program di manajemen saluran AWS Partner Central. Anda mungkin memiliki beberapa akun manajemen program, dan Anda perlu membuat peran IAM di setiap laporan Akun AWS Anda sebagai akun manajemen program.

Untuk setiap Akun AWS jenis, Anda perlu mengaitkan peran IAM yang berbeda dengan kebijakan izin dan kepercayaan khusus.

Akses untuk AWS Mitra Central terhubung Akun AWS

Di dalam Pusat AWS Mitra yang ditautkan Akun AWS, buat peran IAM untuk mengelola sumber daya saluran, yang akan diberikan kepada pengguna Partner Central. Peran IAM ini memungkinkan pengguna untuk melihat dan mengelola sumber daya manajemen saluran di AWS Partner Central. Nama peran harus dimulai dengan `PartnerCentralRoleFor`, dan nama yang disarankan adalah `PartnerCentralRoleForChannel`. Untuk mengonfigurasi peran, lengkapi yang berikut ini:

- Tambahkan kebijakan kepercayaan khusus berikut untuk mengizinkan admin cloud AWS Partner Central dan lead aliansi untuk [memetakan peran IAM ke pengguna Partner Central](#):


```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "partnercentral-account-management.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

- Kaitkan kebijakan yang [AWSPartnerCentralChannelManagement](#) dikelola dengan PartnerCentralRoleForChannel peran tersebut.

Akses untuk akun manajemen program AWS akun (s)

Dalam setiap Akun AWS laporan yang dilaporkan sebagai akun manajemen program di AWS Partner Central, buat peran IAM berikut, dengan nama yang diperlukan di bawah ini:

- PartnerCentralChannelHandshakeApprovalManagement: Manfaatkan peran ini untuk mengelola jabat tangan antara Partner Central dan manajemen program Anda. Akun AWS Peran IAM ini dapat dimanfaatkan untuk merespons untuk mengaktifkan akun manajemen program Anda di AWS Partner Central. Saat membuat peran ini, kaitkan kebijakan AWSPartnerCentralChannelHandshakeApprovalManagement terkelola.
- PartnerCentralChannelBillingTransferReadOnly: Manfaatkan peran lintas akun ini untuk memungkinkan visibilitas ke status transfer penagihan dari Partner Central. AWS Peran ini akan membagikan status transfer penagihan dari Anda Akun AWS ke AWS Partner Central untuk melihat status transfer penagihan secara terpusat. Jika peran ini dibuat, pengguna di Pusat AWS Mitra dapat melihat status transfer penagihan di Pusat AWS Mitra. Namun, peran ini tidak memberikan izin kepada pengguna AWS Partner Central untuk mengakses transfer AWS penagihan di konsol Billing and Cost Management.
- Kebijakan kepercayaan:
 - Dalam JSON di bawah ini, ganti {PartnerCentralLinkedAccountID} dengan ID Akun AWS 12 digit yang ditautkan Pusat AWS Mitra Anda yang sebenarnya saat membuat peran.

```
{
```

```

"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::{PartnerCentralLinkedAccountId}:root"
    },
    "Action": "sts:AssumeRole"
  }
]
}

```

- Kebijakan izin:

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "BillingTransferReadOnly",
      "Effect": "Allow",
      "Action": [
        "organizations:DescribeResponsibilityTransfer",
        "organizations:ListHandshakesForOrganization",
        "organizations:ListInboundResponsibilityTransfers",
        "organizations:ListOutboundResponsibilityTransfers"
      ],
      "Resource": "*"
    }
  ]
}

```

- **PartnerCentralChannelBillingTransferManagement(optional):** Manfaatkan peran lintas akun ini untuk memungkinkan pembuatan dan pengelolaan transfer penagihan dari Partner Central.AWS Peran ini akan memungkinkan pengguna AWS Partner Central mengakses peran IAM yang berisi kebijakan **AWSPartnerCentralChannelManagement** terkelola untuk mengelola transfer penagihan di akun manajemen program.Akun AWS Dengan menggunakan peran lintas akun ini, pengguna AWS Partner Central akan dapat mengakses dan mengelola transfer AWS penagihan di konsol Billing and Cost Management menggunakan tombol AWS“Kelola AWS dalam Penagihan” di manajemen saluran Partner Central.
- Kebijakan kepercayaan:

- Dalam JSON di bawah ini, ganti {PartnerCentralLinkedAccountId} dengan ID Akun AWS 12 digit yang ditautkan Pusat AWS Mitra Anda yang sebenarnya saat membuat peran.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::{PartnerCentralLinkedAccountId}:root"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

- Kebijakan izin:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "BillingTransferManagement",
      "Effect": "Allow",
      "Action": [
        "billingconductor:CreateBillingGroup",
        "billingconductor:ListPricingPlans",
        "organizations:AcceptHandshake",
        "organizations:CancelHandshake",
        "organizations:DeclineHandshake",
        "organizations:DescribeResponsibilityTransfer",
        "organizations:InviteOrganizationToTransferResponsibility",
        "organizations:ListHandshakesForAccount",
        "organizations:ListHandshakesForOrganization",
        "organizations:ListInboundResponsibilityTransfers",
        "organizations:ListOutboundResponsibilityTransfers",
        "organizations:TerminateResponsibilityTransfer",
        "organizations:UpdateResponsibilityTransfer"
      ],
      "Resource": "*"
    }
  ]
}
```

```
}
```

Penggunaan AWS kebijakan terkelola

AWS menyediakan kebijakan terkelola yang memberikan izin yang diperlukan untuk berinteraksi dengan API Saluran. Untuk menyediakan akses yang diperlukan untuk mengelola semua sumber daya saluran, lampirkan `AWSPartnerCentralChannelManagement` kebijakan ke identitas IAM Anda. Untuk menyediakan akses yang diperlukan untuk melihat dan menanggapi permintaan jabatan channel, lampirkan `AWSPartnerCentralChannelHandshakeApprovalManagement` kebijakan tersebut ke identitas IAM Anda. Untuk informasi selengkapnya, lihat [kebijakan AWS terkelola untuk pengguna Pusat AWS Mitra](#).

AWSPartnerCentralChannelManagement kebijakan

Kebijakan ini memberikan akses penuh ke tindakan manajemen saluran Partner Central.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ChannelManagement",
      "Effect": "Allow",
      "Action": [
        "partnercentral:CreateProgramManagementAccount",
        "partnercentral:UpdateProgramManagementAccount",
        "partnercentral>DeleteProgramManagementAccount",
        "partnercentral:ListProgramManagementAccounts",
        "partnercentral:GetProgramManagementAccount",
        "partnercentral:CreateRelationship",
        "partnercentral:UpdateRelationship",
        "partnercentral>DeleteRelationship",
        "partnercentral:GetRelationship",
        "partnercentral:ListRelationships",
        "partnercentral:CreateChannelHandshake",
        "partnercentral:AcceptChannelHandshake",
        "partnercentral:RejectChannelHandshake",
        "partnercentral:CancelChannelHandshake",
        "partnercentral:ListChannelHandshakes"
      ],
      "Resource": "*",
      "Condition": {
```

```

        "StringEquals": {
            "partnercentral:Catalog": [
                "AWS",
                "Sandbox"
            ]
        }
    },
    {
        "Sid": "ChannelBillingTransferRoleAccess",
        "Effect": "Allow",
        "Action": [
            "sts:AssumeRole"
        ],
        "Resource": [
            "arn:aws:iam::*:role/PartnerCentralChannelBillingTransferManagement",
            "arn:aws:iam::*:role/PartnerCentralChannelBillingTransferReadOnly"
        ]
    },
    {
        "Sid": "TaggingAccess",
        "Effect": "Allow",
        "Action": [
            "partnercentral:TagResource",
            "partnercentral:UntagResource",
            "partnercentral:ListTagsForResource"
        ],
        "Resource": [
            "arn:aws:partnercentral:*:*:catalog/*/program-management-account/*",
            "arn:aws:partnercentral:*:*:catalog/*/channel-handshake/*"
        ],
        "Condition": {
            "StringEquals": {
                "partnercentral:Catalog": [
                    "AWS",
                    "Sandbox"
                ]
            }
        }
    }
]
}

```

AWSPartnerCentralChannelHandshakeApprovalManagement kebijakan

Kebijakan ini memberikan akses untuk melihat dan menanggapi permintaan jabat tangan channel. Kebijakan ini harus diterapkan pada peran dalam AWS akun yang menerima permintaan jabat tangan saluran.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ChannelHandshakeManagement",
      "Effect": "Allow",
      "Action": [
        "partnercentral:ListChannelHandshakes",
        "partnercentral:AcceptChannelHandshake",
        "partnercentral:RejectChannelHandshake"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "partnercentral:Catalog": [
            "AWS",
            "Sandbox"
          ]
        }
      }
    }
  ]
}
```

Menetapkan kebijakan untuk peran dan pengguna IAM

Ikuti langkah-langkah ini untuk menetapkan kebijakan untuk peran dan pengguna IAM:

1. Masuk ke Konsol Manajemen AWS.
2. Arahkan ke layanan IAM.
3. Pilih peran atau pengguna, dan pilih peran IAM atau pengguna yang ingin Anda lampirkan kebijakan.
4. Lampirkan `AWSPartnerCentralChannelManagement` kebijakan atau kebijakan khusus Anda ke peran atau pengguna IAM yang dipilih.

Untuk informasi selengkapnya, lihat [Menambahkan dan menghapus izin identitas IAM](#).

Mengelola izin menggunakan tombol kondisi

Kunci kondisi dalam kebijakan IAM memberikan izin tingkat sumber daya kapan harus menerapkan kebijakan pernyataan. Anda dapat menggunakan tombol kondisi untuk menentukan kondisi yang menentukan kapan izin tertentu diizinkan atau ditolak.

Untuk informasi selengkapnya, lihat [elemen kebijakan IAM JSON: Operator kondisi](#).

Ikhtisar tombol kondisi

Kunci syarat	Deskripsi	Tindakan yang berlaku	Nilai valid
Partnercentral:Katalog	Memfilter akses berdasarkan jenis entitas katalog terkait	Semua tindakan Manajemen Saluran	AWS, Kotak Pasir
partnercentral: ChannelHandshakeType	Memfilter akses berdasarkan jenis jabatan tangan saluran	CreateChannelHandshake, AcceptChannelHandshake, RejectChannelHandshake, CancelChannelHandshake, ListChannelHandshakes	PROGRAM_MANAGEMENT_ACCOUNT, START_SERVICE_PERIOD, REVOKE_SERVICE_PERIOD

Ringkasan izin yang diperlukan

Ringkasan izin yang diperlukan

Tindakan	Deskripsi
partnercentral: CreateProgramManagementAccount	memungkinkan pembuatan akun manajemen program
partnercentral: UpdateProgramManagementAccount	memungkinkan memperbarui akun manajemen program

Tindakan	Deskripsi
partnercentral: DeleteProgramManagementAccount	memungkinkan menghapus akun manajemen program
partnercentral: ListProgramManagementAccounts	memungkinkan daftar akun manajemen program
partnercentral: GetProgramManagementAccount	memungkinkan pengambilan detail akun manajemen program
partnercentral: CreateRelationship	memungkinkan menciptakan hubungan
partnercentral: UpdateRelationship	memungkinkan memperbarui hubungan
partnercentral: DeleteRelationship	memungkinkan menghapus hubungan
partnercentral: GetRelationship	memungkinkan mengambil detail hubungan
partnercentral: ListRelationships	memungkinkan daftar hubungan
partnercentral: CreateChannelHandshake	memungkinkan membuat jabat tangan saluran
partnercentral: AcceptChannelHandshake	memungkinkan menerima jabat tangan saluran
partnercentral: RejectChannelHandshake	memungkinkan menolak jabat tangan saluran
partnercentral: CancelChannelHandshake	memungkinkan pembatalan jabat tangan saluran
partnercentral: ListChannelHandshakes	memungkinkan daftar jabat tangan saluran

Akses untuk API Pengukuran Pendapatan

Kontrol akses dan izin dikelola oleh AWS Identity and Access Management(IAM). Bagian ini memberikan panduan untuk mengonfigurasi izin yang diperlukan untuk berinteraksi dengan API Pengukuran Pendapatan, termasuk izin yang diperlukan untuk mencantumkan entitas.AWS Marketplace

Prasyarat

Sebelum mengonfigurasi izin, pastikan bahwa Anda Akun AWS ditautkan ke Partner Central dan Anda membuat peran dan pengguna IAM yang diperlukan. Untuk informasi selengkapnya, lihat [Pengaturan dan Otentikasi](#).

Penggunaan AWS kebijakan terkelola

AWS menyediakan kebijakan terkelola yang memberikan izin yang diperlukan untuk berinteraksi dengan API Pengukuran Pendapatan. Dua kebijakan terkelola tersedia tergantung pada kepribadian Anda:

- Mitra — lampirkan `AWSPartnerCentralRevenueAttributionManagement` kebijakan ke identitas IAM Anda.
- Pelanggan — lampirkan `AWSRevenueAttributionManagement` kebijakan ke identitas IAM Anda.

Untuk informasi selengkapnya, lihat [kebijakan AWS terkelola untuk pengguna Pusat AWS Mitra](#).

`AWSPartnerCentralRevenueAttributionManagement` kebijakan

Kebijakan ini menyediakan akses yang diperlukan untuk aktivitas pengelolaan atribusi pendapatan untuk AWS akun yang terdaftar di AWS Partner Central.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "RevenueMeasurementCreation",
      "Effect": "Allow",
      "Action": [
        "partnercentral:CreateRevenueAttribution",
        "partnercentral:CreateMarketplaceRevenueShare",
        "partnercentral:CreateMarketplaceRevenueShareAllocation"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "partnercentral:Catalog": [
            "AWS",
            "Sandbox"
          ]
        }
      }
    }
  ]
}
```

```

        ]
      }
    },
    {
      "Sid": "RevenueMeasurementListing",
      "Effect": "Allow",
      "Action": [
        "partnercentral:ListRevenueAttributions",
        "partnercentral:ListRevenueAttributionAllocations",
        "partnercentral:ListMarketplaceRevenueShares",
        "partnercentral:ListMarketplaceRevenueShareAllocations"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "partnercentral:Catalog": [
            "AWS",
            "Sandbox"
          ]
        }
      }
    },
    {
      "Sid": "RevenueAttributionManagement",
      "Effect": "Allow",
      "Action": [
        "partnercentral:GetRevenueAttribution",
        "partnercentral:UpdateRevenueAttribution",
        "partnercentral:GetRevenueAttributionAllocation",
        "partnercentral:StartRevenueAttributionAllocationsTask",
        "partnercentral:GetRevenueAttributionAllocationsTask"
      ],
      "Resource": "arn:aws:partnercentral:*:*:catalog/*/revenue-attribution/*",
      "Condition": {
        "StringEquals": {
          "partnercentral:Catalog": [
            "AWS",
            "Sandbox"
          ]
        }
      }
    }
  ],
  {

```

```

        "Sid": "MarketplaceRevenueShareManagement",
        "Effect": "Allow",
        "Action": [
            "partnercentral:GetMarketplaceRevenueShare",
            "partnercentral:GetMarketplaceRevenueShareAllocation",
            "partnercentral:UpdateMarketplaceRevenueShareAllocation"
        ],
        "Resource": "arn:aws:partnercentral:*:*:catalog/*/marketplace-revenue-
share/*",
        "Condition": {
            "StringEquals": {
                "partnercentral:Catalog": [
                    "AWS",
                    "Sandbox"
                ]
            }
        }
    },
    {
        "Sid": "TaggingAccess",
        "Effect": "Allow",
        "Action": [
            "partnercentral:TagResource",
            "partnercentral:UntagResource",
            "partnercentral:ListTagsForResource"
        ],
        "Resource": [
            "arn:aws:partnercentral:*:*:catalog/*/revenue-attribution/*",
            "arn:aws:partnercentral:*:*:catalog/*/marketplace-revenue-share/*"
        ],
        "Condition": {
            "StringEquals": {
                "partnercentral:Catalog": [
                    "AWS",
                    "Sandbox"
                ]
            }
        }
    },
    {
        "Sid": "OpportunityAccess",
        "Effect": "Allow",
        "Action": [
            "partnercentral:ListOpportunities",

```

```

        "partnercentral:GetOpportunity"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "partnercentral:Catalog": [
                "AWS",
                "Sandbox"
            ]
        }
    }
},
{
    "Sid": "PartnerResourceAccess",
    "Effect": "Allow",
    "Action": [
        "partnercentral:ListPartners",
        "partnercentral:GetPartner"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "partnercentral:Catalog": [
                "AWS",
                "Sandbox"
            ]
        }
    }
},
{
    "Sid": "ListingAWSMarketplaceEntities",
    "Effect": "Allow",
    "Action": [
        "aws-marketplace:ListEntities"
    ],
    "Resource": "*"
},
{
    "Sid": "AWSMarketplaceEntityAccess",
    "Effect": "Allow",
    "Action": [
        "aws-marketplace:DescribeEntity"
    ],
    "Resource": [

```

```

        "arn:aws:aws-marketplace:*:*:AWSMarketplace*/Offer/*"
    ],
    },
    {
        "Sid": "AmazonQPartnerAssistantAccess",
        "Effect": "Allow",
        "Action": [
            "q:StartConversation",
            "q:SendMessage",
            "q:GetConversation",
            "q:ListConversations",
            "q:PassRequest"
        ],
        "Resource": "*"
    }
]
}

```

AWSRevenueAttributionManagement kebijakan

Kebijakan ini menyediakan akses yang diperlukan untuk kegiatan pengelolaan atribusi pendapatan untuk AWS akun yang tidak terdaftar di AWS Partner Central.

```

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "RevenueAttributionCreation",
            "Effect": "Allow",
            "Action": [
                "partnercentral:CreateRevenueAttribution"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "AWS",
                        "Sandbox"
                    ]
                }
            }
        },
        {

```

```

        "Sid": "RevenueAttributionListing",
        "Effect": "Allow",
        "Action": [
            "partnercentral:ListRevenueAttributions"
        ],
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "partnercentral:Catalog": [
                    "AWS",
                    "Sandbox"
                ]
            }
        }
    },
    {
        "Sid": "RevenueAttributionManagement",
        "Effect": "Allow",
        "Action": [
            "partnercentral:GetRevenueAttribution",
            "partnercentral:UpdateRevenueAttribution"
        ],
        "Resource": "arn:aws:partnercentral:*:*:catalog/*/revenue-attribution/*",
        "Condition": {
            "StringEquals": {
                "partnercentral:Catalog": [
                    "AWS",
                    "Sandbox"
                ]
            }
        }
    },
    {
        "Sid": "TaggingAccess",
        "Effect": "Allow",
        "Action": [
            "partnercentral:TagResource",
            "partnercentral:UntagResource",
            "partnercentral:ListTagsForResource"
        ],
        "Resource": [
            "arn:aws:partnercentral:*:*:catalog/*/revenue-attribution/*"
        ],
        "Condition": {

```

```
    "StringEquals": {
      "partnercentral:Catalog": [
        "AWS",
        "Sandbox"
      ]
    }
  }
}
```

Menetapkan kebijakan untuk peran dan pengguna IAM

Ikuti langkah-langkah ini untuk menetapkan kebijakan untuk peran dan pengguna IAM:

1. Masuk ke Konsol Manajemen AWS.
2. Arahkan ke layanan IAM.
3. Pilih peran atau pengguna, dan pilih peran IAM atau pengguna yang ingin Anda lampirkan kebijakan.
4. Lampirkan `AWSPartnerCentralRevenueAttributionManagement` kebijakan (untuk mitra) atau `AWSRevenueAttributionManagement` kebijakan (untuk pelanggan) ke peran atau pengguna IAM yang dipilih.

Untuk informasi selengkapnya, lihat [Menambahkan dan menghapus izin identitas IAM](#).

Mengelola izin menggunakan tombol kondisi

Kunci kondisi dalam kebijakan IAM memberikan izin tingkat sumber daya kapan harus menerapkan kebijakan pernyataan. Anda dapat menggunakan tombol kondisi untuk menentukan kondisi yang menentukan kapan izin tertentu diizinkan atau ditolak.

Untuk informasi selengkapnya, lihat [elemen kebijakan IAM JSON: Operator kondisi](#).

Ikhtisar tombol kondisi

Kunci syarat	Deskripsi	Tindakan yang berlaku	Nilai valid
Partnercentral:Katalog	Memfilter akses berdasarkan jenis entitas katalog terkait	Semua tindakan	AWS, Kotak pasir

Ringkasan izin yang diperlukan

Ringkasan izin yang diperlukan

Tindakan	Deskripsi
partnercentral: CreateRevenueAttribution	Memungkinkan membuat catatan atribusi pendapatan baru
partnercentral: UpdateRevenueAttribution	Memungkinkan memperbarui catatan atribusi pendapatan yang ada
partnercentral: GetRevenueAttribution	Memungkinkan mengambil rincian atribusi pendapatan tertentu
partnercentral: ListRevenueAttributions	Memungkinkan mencantumkan atribusi pendapatan dengan filter opsional
partnercentral: GetRevenueAttributionAllocation	Memungkinkan mengambil alokasi tunggal dengan ID-nya
partnercentral: ListRevenueAttributionAllocations	Memungkinkan daftar alokasi berkomitmen dengan dukungan penyaringan
partnercentral: StartRevenueAttributionAllocationsTask	Memungkinkan mengirimkan batch hingga 250 perubahan alokasi untuk pemrosesan asinkron
partnercentral: GetRevenueAttributionAllocationsTask	Memungkinkan pengambilan status tugas alokasi yang dikirimkan sebelumnya

Tindakan	Deskripsi
partnercentral: CreateMarketplaceRevenueShare	Memungkinkan menciptakan sumber daya bagi hasil pasar baru
partnercentral: GetMarketplaceRevenueShare	Memungkinkan mengambil rincian dari pangsa pendapatan pasar tertentu
partnercentral: ListMarketplaceRevenueShares	Memungkinkan daftar pembagian pendapatan pasar dengan filter opsional
partnercentral: CreateMarketplaceRevenueShareAllocation	Memungkinkan pembuatan alokasi bagi hasil pasar baru
partnercentral: GetMarketplaceRevenueShareAllocation	Memungkinkan mengambil rincian alokasi bagi hasil pasar tertentu
partnercentral: UpdateMarketplaceRevenueShareAllocation	Memungkinkan memperbarui alokasi bagi hasil pasar yang ada
partnercentral: ListMarketplaceRevenueShareAllocations	Memungkinkan daftar alokasi di bawah pangsa pendapatan pasar
partnercentral: TagResource	Mengizinkan menambahkan atau menimpa tag untuk sumber daya
partnercentral: UntagResource	Memungkinkan menghapus tag dari sumber daya
partnercentral: ListTagsForResource	Mengizinkan tag daftar yang terkait dengan sumber daya

Kuota untuk AWS API Pusat Mitra

Bagian berikut memberikan informasi tentang kuota layanan.

Topik

- [Kuota untuk AWS API Rekening Sentral Mitra](#)

- [Kuota untuk AWS API Penjualan Sentral Mitra](#)
- [Kuota untuk AWS API Manfaat Sentral Mitra](#)
- [Kuota untuk AWS API Saluran Pusat Mitra](#)
- [Kuota untuk AWS API Pengukuran Pendapatan Pusat Mitra](#)

Kuota untuk AWS API Rekening Sentral Mitra

API AWS Rekening Sentral Mitra memiliki kuota berikut.

Kuota tambahan

Kuota tambahan

Nama tampilan	Katalog	Deskripsi	Nilai default
Buka undangan koneksi per akun	AWS	Jumlah maksimum undangan koneksi terbuka yang dapat Anda pertahankan dengan akun mitra di katalog AWS	1.000
Koneksi aktif per akun	AWS	Jumlah maksimum koneksi aktif yang dapat Anda pertahankan dengan akun mitra di AWS katalog	1.000
Domain email per mitra	AWS	Jumlah maksimum domain email yang dapat dikaitkan dengan akun mitra untuk sertifikasi AWS pelatihan dalam katalog AWS	50
Tingkat undangan koneksi per akun	AWS	Jumlah maksimum undangan koneksi per	50

Nama tampilan	Katalog	Deskripsi	Nilai default
		hari yang dapat Anda kirim dalam katalog AWS	
Tingkat tugas pembaruan profil per akun	AWS	Jumlah maksimum tugas pembaruan profil per hari yang dapat Anda mulai dalam katalog AWS	10
Tingkat pembaruan visibilitas profil per akun	AWS	Jumlah maksimum pembaruan visibilitas profil per hari yang dapat Anda mulai di katalog AWS	5
Tarif kode verifikasi email per alamat email	AWS	Jumlah maksimum kode verifikasi email per hari yang dapat Anda minta di katalog	5
Buka undangan koneksi per akun	Kotak pasir	Jumlah maksimum undangan koneksi terbuka yang dapat Anda pertahankan dengan akun mitra di katalog Sandbox	1.000
Koneksi aktif per akun	Kotak pasir	Jumlah maksimum koneksi aktif yang dapat Anda pertahankan dengan akun mitra di katalog Sandbox	1.000

Nama tampilan	Katalog	Deskripsi	Nilai default
Domain email per mitra	Kotak pasir	Jumlah maksimum domain email yang dapat dikaitkan dengan akun mitra untuk sertifikasi AWS pelatihan dalam katalog Sandbox	50
Tingkat undangan koneksi per akun	Kotak pasir	Jumlah maksimum undangan koneksi per hari yang dapat Anda kirim dalam katalog Sandbox	50
Tingkat tugas pembaruan profil per akun	Kotak pasir	Jumlah maksimum tugas pembaruan profil per hari yang dapat Anda mulai di katalog Sandbox	10
Tingkat pembaruan visibilitas profil per akun	Kotak pasir	Jumlah maksimum pembaruan visibilitas profil per hari yang dapat Anda lakukan di katalog Sandbox	5
Tarif kode verifikasi email per alamat email	Kotak pasir	Jumlah maksimum kode verifikasi email per hari yang dapat Anda minta di katalog Sandbox. Tidak ada email yang dikirim dari katalog Sandbox.	N/A

Memahami dan mengelola kuota

Pembatasan tarif

Ketika batas tarif API tercapai, layanan akan merespons dengan `ThrottlingException`. Untuk menangani pembatasan tarif dengan lebih baik, AWS merekomendasikan untuk menerapkan strategi backoff dan coba lagi eksponensial dalam aplikasi Anda.

Meminta peningkatan kuota

Jika kuota default tidak memenuhi persyaratan Anda, Anda dapat meminta kenaikan kuota melalui halaman [Service Quotas](#). Konsol Service Quotas adalah antarmuka berbasis browser yang dapat Anda gunakan untuk melihat dan mengelola kuota layanan Anda. Anda dapat mengakses Service Quotas dari Konsol Manajemen AWS halaman mana pun dengan memilihnya di bilah navigasi atas, atau dengan mencari Service Quotas di halaman Konsol Manajemen AWS.

Kuota untuk AWS API Penjualan Sentral Mitra

AWS Partner Central selling API memberlakukan kuota untuk memastikan penggunaan yang wajar dan untuk melindungi layanan dari penyalahgunaan. Di bawah ini adalah kuota terperinci untuk berbagai operasi API dan asosiasi per peluang.

Kuota operasi API

Tipe	Operasi API	Kuota (per akun mitra)
Baca tindakan	GetOpportunity	10 per detik; 100.000 per 24 jam
	GetAwsOpportunitySummary	
	ListOpportunities	
	ListSolutions	
	GetEngagementInvitation	
	ListEngagementInvitations	
Prospeksi tindakan baca	GetProspectingFromEngagementTask	100 per detik

Tipe	Operasi API	Kuota (per akun mitra)
Prospeksi tindakan baca	ListProspectingFromEngagementTasks	50 per detik
Tulis tindakan	CreateOpportunity	1 per detik; 10.000 per 24 jam
	UpdateOpportunity	
	AssociateOpportunity	
	DisassociateOpportunity	
	RejectEngagementInvitation	
	AssignOpportunity	
	StartEngagementFromOpportunityTask	
	StartEngagementByAcceptingInvitationTask	
Prospeksi tindakan menulis	StartProspectingFromEngagementTask	2 per detik
Tindakan menulis keterlibatan	CreateEngagement	15 per detik

Kuota asosiasi per peluang

Entitas terkait	Kuota
AWS produk	20 per kesempatan
Solusi Mitra	10 per kesempatan
AWS Marketplace solusi	10 per kesempatan
AWS Marketplace produk	10 per kesempatan

Entitas terkait	Kuota
AWS Marketplace penawaran pribadi	1 per kesempatan

Memahami dan mengelola kuota

Pembatasan tarif

Ketika batas tarif API tercapai, layanan akan merespons dengan `ThrottlingException`. Untuk menangani pembatasan tarif dengan lebih baik, AWS merekomendasikan untuk menerapkan strategi backoff dan coba lagi eksponensial dalam aplikasi Anda.

Jendela waktu untuk kuota

Kuota harian diatur ulang pada periode 24 jam bergulir. Permintaan Anda akan dibatasi misalnya jika Anda telah melakukan 10.000 tindakan tulis dalam 24 jam terakhir dan mencoba melakukan permintaan 10.001. Pastikan pola penggunaan aplikasi Anda mempertimbangkan hal ini untuk mencegah pelambatan yang tidak disengaja.

Meminta peningkatan kuota

Jika kuota default tidak memenuhi persyaratan Anda, Anda dapat meminta kenaikan kuota melalui halaman [Service Quotas](#). Konsol Service Quotas adalah antarmuka berbasis browser yang dapat Anda gunakan untuk melihat dan mengelola kuota layanan Anda. Anda dapat mengakses Service Quotas dari Konsol Manajemen AWS halaman mana pun dengan memilihnya di bilah navigasi atas, atau dengan mencari Service Quotas di halaman Konsol Manajemen AWS.

Kuota untuk AWS API Manfaat Sentral Mitra

AWS Partner Central Benefits API memiliki kuota berikut.

Kuota permintaan

Kuota permintaan

Operasi API	Kuota (per AWS akun)
ListBenefits	20 per detik

Operasi API	Kuota (per AWS akun)
GetBenefit	20 per detik
AmendBenefitApplication	10 per detik
CancelBenefitApplication	10 per detik
CreateBenefitApplication	10 per detik
GetBenefitApplication	20 per detik
ListBenefitApplications	30 per detik
RecallBenefitApplication	10 per detik
SubmitBenefitApplication	10 per detik
UpdateBenefitApplication	10 per detik
GetBenefitAllocation	20 per detik
ListBenefitAllocations	20 per detik
AssociateBenefitApplicationResource	10 per detik
DisassociateBenefitApplicationResource	10 per detik

Kuota tambahan

Kuota tambahan

Nama tampilan	Katalog	Deskripsi	Nilai default
Aplikasi manfaat	AWS	Jumlah maksimum aplikasi manfaat yang dapat Anda buat di AWS katalog	10.000
Aplikasi manfaat	Kotak pasir	Jumlah maksimum aplikasi manfaat yang	10.000

Nama tampilan	Katalog	Deskripsi	Nilai default
		dapat Anda buat di katalog Sandbox	

Memahami dan mengelola kuota

Pembatasan tarif

Ketika batas tarif API tercapai, layanan akan merespons dengan `ThrottlingException`. Untuk menangani pembatasan tarif dengan lebih baik, AWS merekomendasikan untuk menerapkan strategi backoff dan coba lagi eksponensial dalam aplikasi Anda.

Meminta peningkatan kuota

Jika kuota default tidak memenuhi persyaratan Anda, Anda dapat meminta kenaikan kuota melalui halaman [Service Quotas](#). Konsol Service Quotas adalah antarmuka berbasis browser yang dapat Anda gunakan untuk melihat dan mengelola kuota layanan Anda. Anda dapat mengakses Service Quotas dari Konsol Manajemen AWS halaman mana pun dengan memilihnya di bilah navigasi atas, atau dengan mencari Service Quotas di halaman Konsol Manajemen AWS.

Kuota untuk AWS API Saluran Pusat Mitra

AWS Partner Central Channel API memiliki kuota berikut.

Kuota permintaan

Kuota permintaan

Operasi API	Kuota (per AWS akun)
CreateProgramManagementAccount	3 per detik
UpdateProgramManagementAccount	3 per detik
ListProgramManagementAccounts	5 per detik
DeleteProgramManagementAccount	3 per detik
CreateChannelHandshake	3 per detik

Operasi API	Kuota (per AWS akun)
ListChannelHandshakes	5 per detik
AcceptChannelHandshake	3 per detik
RejectChannelHandshake	3 per detik
CancelChannelHandshake	3 per detik
CreateRelationship	3 per detik
GetRelationship	5 per detik
ListRelationships	5 per detik
UpdateRelationship	3 per detik
DeleteRelationship	3 per detik

Kuota tambahan

Kuota tambahan

Nama tampilan	Katalog	Deskripsi	Nilai default
Akun manajemen program	AWS	Jumlah maksimum akun manajemen program yang dapat Anda buat di AWS katalog	50
Hubungan per akun manajemen program	AWS	Jumlah maksimum hubungan yang dapat Anda buat per akun manajemen program dalam AWS katalog	1.000

Nama tampilan	Katalog	Deskripsi	Nilai default
Jabat tangan akun manajemen program per akun	AWS	Jumlah maksimum jabat tangan akun manajemen program aktif dalam katalog AWS	100
Mulai periode layanan jabat tangan per akun	AWS	Jumlah maksimum jabat tangan aktif untuk menetapkan n periode layanan dalam katalog AWS	200
Mencabut jabat tangan periode layanan per akun	AWS	Jumlah maksimum jabat tangan aktif untuk mencabut periode layanan dalam katalog AWS	200
Tingkat jabat tangan akun manajemen program per hari	AWS	Jumlah maksimum jabat tangan akun manajemen program yang dapat Anda kirim ke AWS akun yang sama per hari di katalog AWS	5
Tingkat awal periode layanan jabat tangan per hari	AWS	Jumlah maksimum jabat tangan periode layanan awal yang dapat Anda kirim ke AWS akun yang sama per hari di katalog AWS	5

Nama tampilan	Katalog	Deskripsi	Nilai default
Tarif jabat tangan periode layanan mencabut per hari	AWS	Jumlah maksimum jabat tangan periode layanan pencabulan yang dapat Anda kirim ke AWS akun yang sama per hari di katalog AWS	5
Akun manajemen program	Kotak pasir	Jumlah maksimum akun manajemen program yang dapat Anda buat di katalog Sandbox	50
Hubungan per akun manajemen program	Kotak pasir	Jumlah maksimum hubungan yang dapat Anda buat per akun manajemen program di katalog Sandbox	1.000
Jabat tangan akun manajemen program per akun	Kotak pasir	Jumlah maksimum jabat tangan akun manajemen program aktif di katalog Sandbox	100
Mulai periode layanan jabat tangan per akun	Kotak pasir	Jumlah maksimum jabat tangan aktif untuk menetapkan periode layanan dalam katalog Sandbox	200

Nama tampilan	Katalog	Deskripsi	Nilai default
Mencabut jabat tangan periode layanan per akun	Kotak pasir	Jumlah maksimum jabat tangan aktif untuk mencabut periode layanan dalam katalog Sandbox	200
Tingkat jabat tangan akun manajemen program per hari	Kotak pasir	Jumlah maksimum jabat tangan akun manajemen program yang dapat Anda kirim ke AWS akun yang sama per hari di katalog Sandbox	5
Tingkat awal periode layanan jabat tangan per hari	Kotak pasir	Jumlah maksimum jabat tangan periode layanan awal yang dapat Anda kirim ke AWS akun yang sama per hari di katalog Sandbox	5
Tarif jabat tangan periode layanan mencabut per hari	Kotak pasir	Jumlah maksimum jabat tangan periode layanan pencabulan yang dapat Anda kirim ke AWS akun yang sama per hari di katalog Sandbox	5

Memahami dan mengelola kuota

Pembatasan tarif

Ketika batas tarif API tercapai, layanan akan merespons dengan `ThrottlingException`. Untuk menangani pembatasan tarif dengan lebih baik, AWS merekomendasikan untuk menerapkan strategi backoff dan coba lagi eksponensial dalam aplikasi Anda.

Meminta peningkatan kuota

Jika kuota default tidak memenuhi persyaratan Anda, Anda dapat meminta kenaikan kuota melalui halaman [Service Quotas](#). Konsol Service Quotas adalah antarmuka berbasis browser yang dapat Anda gunakan untuk melihat dan mengelola kuota layanan Anda. Anda dapat mengakses Service Quotas dari Konsol Manajemen AWS halaman mana pun dengan memilihnya di bilah navigasi atas, atau dengan mencari Service Quotas di halaman Konsol Manajemen AWS.

Kuota untuk AWS API Pengukuran Pendapatan Pusat Mitra

AWS Partner Central Revenue Measurement API memberlakukan kuota untuk memastikan penggunaan yang wajar dan untuk melindungi layanan dari penyalahgunaan. Di bawah ini adalah kuota terperinci untuk berbagai operasi API.

Kuota operasi API

Tipe	Operasi API	Kuota (per akun mitra)
Baca tindakan	<code>GetRevenueAttribution</code>	50 per detik
	<code>GetRevenueAttributionAllocation</code>	
	<code>GetRevenueAttributionAllocationsTask</code>	
	<code>GetMarketplaceRevenueShare</code>	
	<code>GetMarketplaceRevenueShareAllocation</code>	

Tipe	Operasi API	Kuota (per akun mitra)
Daftar tindakan	ListRevenueAttributions	10 per detik
	ListRevenueAttributionAllocations	
	ListMarketplaceRevenueShares	
	ListMarketplaceRevenueShareAllocations	
	ListTagsForResource	
Tulis tindakan	CreateRevenueAttribution	2 per detik
	UpdateRevenueAttribution	
	CreateMarketplaceRevenueShare	
	CreateMarketplaceRevenueShareAllocation	
	UpdateMarketplaceRevenueShareAllocation	
	TagResource	
	UntagResource	
Tindakan tulis async	StartRevenueAttributionAllocationsTask	1 per detik

Memahami dan mengelola kuota

Pembatasan tarif

Ketika batas tarif API tercapai, layanan akan merespons dengan `ThrottlingException`. Untuk menangani pembatasan tingkat dengan lebih baik, AWS merekomendasikan untuk menerapkan strategi backoff dan coba lagi eksponensial dalam aplikasi Anda.

Meminta peningkatan kuota

Jika kuota default tidak memenuhi persyaratan Anda, Anda dapat meminta kenaikan kuota melalui halaman [Service Quotas](#). Konsol Service Quotas adalah antarmuka berbasis browser yang dapat Anda gunakan untuk melihat dan mengelola kuota layanan Anda. Anda dapat mengakses Service Quotas dari Konsol Manajemen AWS halaman mana pun dengan memilihnya di bilah navigasi atas, atau dengan mencari Service Quotas di halaman Konsol Manajemen AWS.

Penebangan untuk AWS API Pusat Mitra

Bagian berikut memberikan informasi tentang logging.

Topik

- [Pencatatan AWS API Rekening Sentral Mitra](#)
- [Pencatatan AWS API Penjualan Sentral Mitra](#)
- [Pencatatan AWS API Manfaat Sentral Mitra](#)
- [Pencatatan AWS API Saluran Pusat Mitra](#)
- [Pencatatan AWS API Atribusi Pendapatan Pusat Mitra](#)

Pencatatan AWS API Rekening Sentral Mitra

[AWS CloudTrail](#) adalah layanan yang memungkinkan tata kelola, kepatuhan, audit operasional, dan audit risiko Anda. Akun AWS Dengan AWS CloudTrail, Anda dapat log, terus memantau, dan mempertahankan aktivitas akun yang terkait dengan tindakan di seluruh AWS infrastruktur Anda. AWS Aktivitas API Rekening Sentral Mitra dicatat sebagai peristiwa di CloudTrail. Anda dapat membuat jejak, konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3.

Gambaran umum

API Akun Pusat AWS Mitra terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di AWS Partner Central. CloudTrail menangkap semua panggilan API untuk API Akun Sentral AWS Mitra sebagai peristiwa. Panggilan yang diambil termasuk panggilan dari Pusat AWS Mitra dan dari panggilan kode ke operasi API Akun Pusat AWS Mitra.

Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara berkelanjutan ke bucket Amazon S3, termasuk peristiwa untuk API Akun Pusat AWS Mitra. Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol dalam Riwayat acara.

Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat ke API Akun Pusat AWS Mitra, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Memahami AWS Entri file log API Akun Sentral Mitra

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3. Saat jejak Anda melacak peristiwa API Akun Pusat AWS Mitra, CloudTrail proses peristiwa sebagai file log di semua wilayah. Setiap file log dapat berisi satu atau lebih peristiwa.

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `CreatePartner` tindakan pada API Akun Sentral AWS Mitra:

```
{
  "eventVersion": "1.11",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "EX_PRINCIPAL_ID",
    "arn": "arn:aws:iam::123456789012:user/CloudTrailTestUser",
    "accountId": "123456789012",
    "accessKeyId": "ABCDEFGHJKLMNOP1234",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROEXAMPLE52AGFT725JGDZ",
        "arn": "arn:aws:iam::123456789012:role/ExampleRole",
        "accountId": "123456789012",
        "userName": "ExampleRole"
      }
    }
  },
```

```
        "attributes": {
            "creationDate": "2025-11-07T19:45:28Z",
            "mfaAuthenticated": "false"
        }
    },
    "eventTime": "2025-11-07T19:45:58Z",
    "eventSource": "partnercentral-account.amazonaws.com",
    "eventName": "CreatePartner",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "PostmanRuntime/7.18.0",
    "requestParameters": {
        "catalog": "AWS",
        "clientToken": "abcdef12-3456-7890-bcde-f123456789ab",
        "legalName": "ExampleLegalName",
        "primarySolutionType": "VALUE_ADDED_RESALE_AWS_SERVICES",
        "allianceLeadContact": {
            "firstName": "ExampleFirstName",
            "lastName": "ExampleLastName",
            "email": "example@domain.com",
            "businessTitle": "ExampleBusinessTitle"
        }
    },
    "emailVerificationCode": "ExampleVerificationCode",
    "tags": [
        {
            "key": "office",
            "value": "1"
        }
    ]
},
"responseElements": {
    "catalog": "AWS",
    "arn": "arn:aws:partnercentral:us-east-1:123456789012:catalog/AWS/partner/EXAMPLE_PARTNER_ID",
    "id": "EXAMPLE_PARTNER_ID",
    "legalName": "ExampleLegalName",
    "createdAt": "2025-11-07T19:45:57.867007329Z",
    "profile": {
        "primarySolutionType": "VALUE_ADDED_RESALE_AWS_SERVICES"
    },
    "allianceLeadContact": {
        "firstName": "ExampleFirstName",
        "lastName": "ExampleLastName",
```

```

        "email": "example@domain.com",
        "businessTitle": "ExampleBusinessTitle"
    },
    "requestID": "12345678-1234-5678-9abc-def012345678",
    "eventID": "87654321-4321-8765-cba9-fed098765432",
    "readOnly": false,
    "resources": [
        {
            "accountId": "123456789012",
            "type": "AWS::PartnerCentral::Partner",
            "ARN": "arn:aws:partnercentral:us-east-1:123456789012:catalog/AWS/partner/
EXAMPLE_PARTNER_ID"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management",
    "tlsDetails": {
        "tlsVersion": "TLSv1.3",
        "cipherSuite": "TLS_AES_128_GCM_SHA256",
        "clientProvidedHostHeader": "partnercentral-account.partner.us-east-1.api.aws"
    }
}

```

Dalam contoh ini, `CreatePartner` tindakan dipanggil oleh pengguna IAM bernama `CloudTrailTestUser`. Permintaan dibuat pada 7 November 2025 pukul 19:45:58 UTC. Permintaan tersebut membuat mitra baru dengan ID `EXAMPLE_PARTNER_ID` untuk katalog AWS dengan nama resmi `ExampleLegalName`.

Bidang di AWS Entri file log API Akun Sentral Mitra

Setiap entri dalam file CloudTrail log berisi informasi tentang siapa yang membuat permintaan, sumber daya yang ditindaklanjuti dalam permintaan, dan elemen respons yang dikembalikan oleh API Akun Pusat AWS Mitra. Daftar bidang dalam entri log, seperti `eventVersion`, dan `userIdentityeventTime`, memberikan informasi rinci tentang tindakan tersebut. Misalnya, `sourceIPAddress` bidang menunjukkan alamat IP tempat permintaan dibuat.

Pencatatan AWS API Penjualan Sentral Mitra

[AWS CloudTrail](#) adalah layanan yang memungkinkan tata kelola, kepatuhan, audit operasional, dan audit risiko Anda. Akun AWS Dengan AWS CloudTrail, Anda dapat log, terus memantau, dan mempertahankan aktivitas akun yang terkait dengan tindakan di seluruh AWS infrastruktur Anda. AWS Aktivitas API Pusat Mitra dicatat sebagai peristiwa di CloudTrail. Anda dapat membuat jejak, konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3.

Gambaran umum

API AWS Partner Central terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di AWS Partner Central. CloudTrail menangkap semua panggilan API untuk AWS Partner Central sebagai peristiwa. Panggilan yang diambil termasuk panggilan dari Pusat AWS Mitra dan dari panggilan kode ke operasi API Pusat AWS Mitra.

Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara berkelanjutan ke bucket Amazon S3, termasuk acara untuk AWS Partner Central. Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol dalam Riwayat acara.

Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat ke AWS Partner Central, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Memahami AWS Entri berkas log API Penjualan Pusat Mitra

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3. Saat jejak Anda melacak peristiwa AWS Partner Central, CloudTrail proses peristiwa sebagai file log di semua wilayah. Setiap file log dapat berisi satu atau lebih peristiwa.

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `ListOpportunities` tindakan pada AWS Partner Central:

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "ABCDEFGHJKLMNOP12345",
    "arn": "arn:aws:iam::123456789010:user/CloudTrailTestUser",
```

```
    "accountId": "123456789010",
    "accessKeyId": "ABCDEFGHJKLMNOP1234",
    "userName": "CloudTrailTestUser"
  },
  "eventTime": "2023-10-17T21:49:23Z",
  "eventSource": "partnercentral-selling.amazonaws.com",
  "eventName": "ListOpportunities",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "127.0.0.1",
  "userAgent": "PostmanRuntime/7.18.0",
  "requestParameters": {
    "MaxResults": 20
  },
  "responseElements": null,
  "requestID": "fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1",
  "eventID": "7EXAMPLE-97d6-4139-91e3-01aEXAMPLE48",
  "readOnly": true,
  "eventType": "AwsApiCall",
  "recipientAccountId": "123456789010"
}
```

Dalam contoh ini, `ListOpportunities` tindakan dipanggil oleh pengguna IAM bernama `CloudTrailTestUser`. Tindakan itu dipanggil di Wilayah AWS `us-east-1`, dan permintaan dibuat pada 17 Oktober 2023 pukul 21:49:23 UTC.

Bidang di AWS Entri berkas log API Penjualan Pusat Mitra

Setiap entri dalam file CloudTrail log berisi informasi tentang siapa yang membuat permintaan, sumber daya yang ditindaklanjuti dalam permintaan, dan elemen respons yang dikembalikan oleh AWS Partner Central. Daftar bidang dalam entri log, seperti `eventVersion`, dan `userIdentityeventTime`, memberikan informasi rinci tentang tindakan tersebut. Misalnya, `sourceIPAddress` bidang menunjukkan alamat IP tempat permintaan dibuat.

Pencatatan AWS API Manfaat Sentral Mitra

[AWS CloudTrail](#) adalah layanan yang memungkinkan tata kelola, kepatuhan, audit operasional, dan audit risiko Anda. Akun AWS Dengan AWS CloudTrail, Anda dapat log, terus memantau, dan mempertahankan aktivitas akun yang terkait dengan tindakan di seluruh AWS infrastruktur Anda. AWS Aktivitas API Partner Central Benefits dicatat sebagai peristiwa di CloudTrail. Anda dapat membuat jejak, konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3.

Gambaran umum

AWS Partner Central Benefits API terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di AWS Partner Central. CloudTrail menangkap semua panggilan API untuk AWS Partner Central Benefits API sebagai event. Panggilan yang diambil mencakup panggilan dari Pusat AWS Mitra dan dari panggilan kode ke operasi API Manfaat Pusat AWS Mitra.

Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara berkelanjutan ke bucket Amazon S3, termasuk peristiwa untuk API Manfaat Pusat AWS Mitra. Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol dalam Riwayat acara.

Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat untuk AWS Partner Central Benefits API, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Memahami AWS Entri file log API Manfaat Sentral Mitra

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3. Saat jejak Anda melacak peristiwa API AWS Partner Central Benefits, CloudTrail proses peristiwa sebagai file log di semua wilayah. Setiap file log dapat berisi satu atau lebih peristiwa.

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `ListBenefitApplications` tindakan pada API Manfaat Pusat AWS Mitra:

```
{
  "eventVersion": "1.11",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "EX_PRINCIPAL_ID",
    "arn": "arn:aws:iam::123456789012:user/CloudTrailTestUser",
    "accountId": "123456789012",
    "accessKeyId": "EXAMPLE_KEY_ID",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:role/TestRole",
        "accountId": "123456789012",
        "userName": "TestRole"
      }
    }
  },
```

```

        "attributes": {
            "creationDate": "2025-10-21T03:08:23Z",
            "mfaAuthenticated": "false"
        }
    },
    "eventTime": "2025-10-21T03:10:59Z",
    "eventSource": "partnercentral-benefits.amazonaws.com",
    "eventName": "ListBenefitApplications",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "python-requests/2.32.4",
    "requestParameters": {
        "catalog": "AWS"
    },
    "responseElements": null,
    "requestID": "12345678-1234-5678-9abc-def012345678",
    "eventID": "87654321-4321-8765-cba9-fed098765432",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}

```

Dalam contoh ini, ListBenefitApplications tindakan dipanggil oleh pengguna IAM bernama Alice. Permintaan dibuat pada 21 Oktober 2025 pukul 03:10:59 UTC. Permintaan mencantumkan aplikasi manfaat untuk katalog AWS.

Bidang di AWS Entri file log API Manfaat Sentral Mitra

Setiap entri dalam file CloudTrail log berisi informasi tentang siapa yang membuat permintaan, sumber daya yang ditindaklanjuti dalam permintaan, dan elemen respons yang dikembalikan oleh AWS Partner Central Benefits API. Daftar bidang dalam entri log, seperti, eventVersion, dan userIdentityeventTime, memberikan informasi rinci tentang tindakan tersebut. Misalnya, sourceIPAddress bidang menunjukkan alamat IP tempat permintaan dibuat.

Pencatatan AWS API Saluran Pusat Mitra

[AWS CloudTrail](#) adalah layanan yang memungkinkan tata kelola, kepatuhan, audit operasional, dan audit risiko Anda. Akun AWS Dengan AWS CloudTrail, Anda dapat log, terus memantau, dan mempertahankan aktivitas akun yang terkait dengan tindakan di seluruh AWS infrastruktur

Anda.AWS Aktivitas API Saluran Pusat Mitra direkam sebagai peristiwa di CloudTrail. Anda dapat membuat jejak, konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3.

Gambaran umum

API Saluran Pusat AWS Mitra terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di Pusat AWS Mitra. CloudTrail menangkap semua panggilan API untuk AWS Partner Central Channel API sebagai peristiwa. Panggilan yang diambil mencakup panggilan dari Pusat AWS Mitra dan dari panggilan kode ke operasi API Saluran Pusat AWS Mitra.

Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail peristiwa secara berkelanjutan ke bucket Amazon S3, termasuk peristiwa untuk API Saluran Pusat AWS Mitra. Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol dalam Riwayat acara.

Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat ke API Saluran Pusat AWS Mitra, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Memahami AWS Entri file log API Saluran Pusat Mitra

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3. Saat jejak Anda melacak peristiwa API Saluran Pusat AWS Mitra, CloudTrail proses peristiwa sebagai file log di semua wilayah. Setiap file log dapat berisi satu atau lebih peristiwa.

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan CreateProgramManagementAccount tindakan pada API Saluran Pusat AWS Mitra:

```
{
  "eventVersion": "1.11",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROEXAMPLE52AGFT725JGDZ:example-user-Isengard",
    "arn": "arn:aws:sts::123456789012:assumed-role/Admin/example-user-Isengard",
    "accountId": "123456789012",
    "accessKeyId": "ASIAEXAMPLE52AGL6IXQLZ5",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
```



```

        "principalId": "AROEXAMPLE52AGFT725JGDZ",
        "arn": "arn:aws:iam::123456789012:role/ExampleRole",
        "accountId": "123456789012",
        "userName": "ExampleRole"
    },
    "attributes": {
        "creationDate": "2025-10-21T17:06:47Z",
        "mfaAuthenticated": "false"
    }
}
},
"eventTime": "2025-10-21T17:07:26Z",
"eventSource": "partnercentral-channel.amazonaws.com",
"eventName": "CreateProgramManagementAccount",
"awsRegion": "us-east-1",
"sourceIPAddress": "127.0.0.1",
"userAgent": "PostmanRuntime/7.18.0",
"requestParameters": {
    "catalog": "AWS",
    "program": "SOLUTION_PROVIDER",
    "displayName": "ExampleDisplayName",
    "accountId": "987654321098",
    "clientToken": "abcdef12-3456-7890-bcde-f123456789ab"
},
"responseElements": {
    "programManagementAccountDetail": {
        "id": "pma-example123456789",
        "arn": "arn:aws:partnercentral:us-east-1:123456789012:catalog/AWS/program-
management-account/pma-example123456789"
    }
},
"requestID": "12345678-1234-5678-9abc-def012345678",
"eventID": "87654321-4321-8765-cba9-fed098765432",
"readOnly": false,
"resources": [
    {
        "accountId": "123456789012",
        "type": "AWS::PartnerCentralChannel::ProgramManagementAccount",
        "ARN": "arn:aws:partnercentral:us-east-1:123456789012:catalog/AWS/program-
management-account/pma-example123456789"
    }
],
"eventType": "AwsApiCall",
"managementEvent": true,

```

```
"recipientAccountId": "123456789012",
"eventCategory": "Management",
"tlsDetails": {
  "clientProvidedHostHeader": "partnercentral-channel.global.api.aws"
}
}
```

Dalam contoh ini, `CreateProgramManagementAccount` tindakan dipanggil oleh peran IAM yang dinamai `ExampleRole` melalui sesi peran yang diasumsikan. Permintaan dibuat pada 21 Oktober 2025 pukul 17:07:26 UTC. Permintaan membuat Akun Manajemen Program baru dengan ID `pma-example123456789` untuk program Penyedia Solusi.

Bidang di AWS Entri file log API Saluran Pusat Mitra

Setiap entri dalam file CloudTrail log berisi informasi tentang siapa yang membuat permintaan, sumber daya yang ditindaklanjuti dalam permintaan, dan elemen respons yang dikembalikan oleh API Saluran Pusat AWS Mitra. Daftar bidang dalam entri log, seperti `eventVersion`, dan `userIdentityeventTime`, memberikan informasi rinci tentang tindakan tersebut. Misalnya, `sourceIPAddress` bidang menunjukkan alamat IP tempat permintaan dibuat.

Pencatatan AWS API Atribusi Pendapatan Pusat Mitra

[AWS CloudTrail](#) adalah layanan yang memungkinkan tata kelola, kepatuhan, audit operasional, dan audit risiko Anda. Akun AWS Dengan AWS CloudTrail, Anda dapat log, terus memantau, dan mempertahankan aktivitas akun yang terkait dengan tindakan di seluruh AWS infrastruktur Anda. AWS Aktivitas API Pengukuran Pendapatan Sentral Mitra dicatat sebagai peristiwa di CloudTrail. Anda dapat membuat jejak, konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3.

Gambaran umum

API Pengukuran Pendapatan Pusat AWS Mitra terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di AWS Partner Central. CloudTrail menangkap semua panggilan API untuk Atribusi Pendapatan Pusat AWS Mitra sebagai peristiwa. Panggilan yang diambil mencakup panggilan dari konsol Pusat AWS Mitra dan dari panggilan kode ke operasi API Pengukuran Pendapatan Pusat AWS Mitra.

Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara berkelanjutan ke bucket Amazon S3, termasuk peristiwa untuk Pengukuran Pendapatan Pusat AWS

Mitra. Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol dalam Riwayat acara.

Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat untuk Pengukuran Pendapatan Pusat AWS Mitra, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Tindakan API Pengukuran Pendapatan Pusat AWS Mitra berikut dicatat CloudTrail:

- `CreateRevenueAttribution`
- `UpdateRevenueAttribution`
- `GetRevenueAttribution`
- `ListRevenueAttributions`
- `TagResource`
- `UntagResource`
- `ListTagsForResource`
- `CreateMarketplaceRevenueShare`
- `GetMarketplaceRevenueShare`
- `ListMarketplaceRevenueShares`
- `CreateMarketplaceRevenueShareAllocation`
- `GetMarketplaceRevenueShareAllocation`
- `UpdateMarketplaceRevenueShareAllocation`
- `ListMarketplaceRevenueShareAllocations`
- `StartRevenueAttributionAllocationsTask`
- `GetRevenueAttributionAllocationsTask`
- `ListRevenueAttributionAllocations`
- `GetRevenueAttributionAllocation`

Memahami AWS Entri file log API Pengukuran Pendapatan Pusat Mitra

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3. Saat jejak Anda melacak peristiwa Pengukuran Pendapatan Pusat AWS Mitra, CloudTrail proses peristiwa sebagai file log di semua wilayah. Setiap file log dapat berisi satu atau lebih peristiwa.

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan CreateRevenueAttribution tindakan pada Atribusi Pendapatan Pusat AWS Mitra:

```
{
  "eventVersion": "1.11",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDAEXAMPLEID",
    "arn": "arn:aws:iam::123456789012:user/CloudTrailTestUser",
    "accountId": "123456789012",
    "accessKeyId": "AKIAEXAMPLEKEY",
    "userName": "CloudTrailTestUser"
  },
  "eventTime": "2026-06-02T11:53:54Z",
  "eventSource": "partnercentral-prm.amazonaws.com",
  "eventName": "CreateRevenueAttribution",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "192.0.2.1",
  "userAgent": "aws-sdk-java/2.20.0",
  "requestParameters": {
    "catalog": "AWS",
    "name": "my-revenue-attribution",
    "marketplaceProduct": {
      "tenancyModel": "MULTI_TENANT"
    }
  },
  "responseElements": {
    "id": "ra-0123456789abcdef0",
    "arn": "arn:aws:partnercentral:us-east-1:123456789012:catalog/AWS/revenue-attribution/ra-0123456789abcdef0",
    "name": "my-revenue-attribution",
    "revision": "1"
  },
  "requestID": "5ce2f907-3850-412a-b1a4-r012r1234567",
  "eventID": "80b39b72-eefe-4578-8db0-01ee2e34ee56",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "recipientAccountId": "123456789012"
}
```

Dalam contoh ini, CreateRevenueAttribution tindakan dipanggil oleh pengguna IAM bernama CloudTrailTestUser. Tindakan itu dipanggil di Wilayah AWS us-east-1, dan permintaan dibuat

pada 2 Juni 2026 pukul 11:53:54 UTC. Sumber daya atribusi pendapatan baru dibuat dengan IDra-0123456789abcdef0.

Bidang di AWS Entri file log API Pengukuran Pendapatan Pusat Mitra

Setiap entri dalam file CloudTrail log berisi informasi tentang siapa yang membuat permintaan, sumber daya yang ditindaklanjuti dalam permintaan, dan elemen respons yang dikembalikan oleh Pengukuran Pendapatan Pusat AWS Mitra. Daftar bidang dalam entri log, seperti, `eventVersion`, dan `userIdentityeventTime`, memberikan informasi rinci tentang tindakan tersebut. Misalnya, `sourceIPAddress` bidang menunjukkan alamat IP tempat permintaan dibuat.

Pemberitahuan untuk AWS API Pusat Mitra

Bagian berikut memberikan informasi tentang notifikasi.

Topik

- [Pemberitahuan untuk AWS API Rekening Sentral Mitra](#)
- [Pemberitahuan untuk AWS API Penjualan Sentral Mitra](#)
- [Pemberitahuan untuk AWS API Manfaat Sentral Mitra](#)
- [Pemberitahuan untuk AWS API Saluran Pusat Mitra](#)

Pemberitahuan untuk AWS API Rekening Sentral Mitra

Pemberitahuan Koneksi Akun Mitra memungkinkan mitra untuk tetap mendapat informasi tentang peristiwa siklus hidup koneksi yang secara langsung memengaruhi hubungan bisnis dan alur kerja operasional mereka. Peristiwa ini sangat penting bagi mitra untuk:

- Bereaksi segera terhadap kolaborasi baru
- Pertahankan kesadaran akan status portofolio koneksi mereka
- Mengambil tindakan yang tepat ketika koneksi dibuat atau dihentikan
- Pastikan kelangsungan bisnis dengan mengetahui kapan hubungan berubah

Topik

- [Lengkapi prasyarat untuk memantau acara](#)
- [Konfigurasi Amazon EventBridge untuk memantau peristiwa](#)

- [Pelajari lebih lanjut tentang peristiwa API koneksi akun](#)

Lengkapi prasyarat untuk memantau acara

Pengguna memerlukan izin IAM yang sesuai untuk mengakses dan mengelola peristiwa yang diterbitkan oleh API koneksi akun. Untuk informasi selengkapnya tentang tindakan, sumber daya, dan kunci kondisi yang tersedia EventBridge, lihat [Menggunakan kondisi kebijakan IAM di Amazon EventBridge](#) di Panduan EventBridge Pengguna Amazon. Salah satu kunci kondisi adalah `events:detail-type`, yang dapat digunakan untuk cakupan izin untuk jenis peristiwa tertentu.

Contoh kebijakan berikut menunjukkan cara menyesuaikan dan cakupan izin untuk acara yang diusulkan. `AllowPutRuleForPartnercentralAccountEvents` Pernyataan tersebut memungkinkan pembuatan aturan, tetapi hanya untuk acara dari `aws.partnercentral-account` sumbernya.

Untuk contoh kebijakan IAM yang mendetail, lihat dokumentasi AWS tentang EventBridge izin.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowPutRuleForPartnerCentralAccountEvents",
      "Effect": "Allow",
      "Action": "events:PutRule",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "events:source": "aws.partnercentral-account.connection"
        }
      }
    }
  ]
}
```

Konfigurasi Amazon EventBridge untuk memantau peristiwa

Untuk memantau peristiwa API koneksi akun, Anda membuat EventBridge aturan yang cocok dengan peristiwa yang ingin Anda tangkap. Anda dapat menggunakan Konsol Manajemen AWS atau AWS SDK untuk membuat dan mengelola aturan. Bagian berikut menjelaskan cara membuat aturan

menggunakan kedua metode. Terlepas dari metode yang Anda gunakan, Anda harus membuat aturan di `us-east-1` Wilayah AS Timur (Virginia N.).

Konsol Manajemen AWS mempersiapkan

Untuk menyiapkan EventBridge aturan menggunakan Konsol Manajemen AWS, ikuti langkah-langkah dalam [Membuat aturan yang bereaksi terhadap peristiwa di Amazon EventBridge](#) di Panduan EventBridge Pengguna Amazon. Saat membuat aturan, Anda harus mengatur bus acara ke default, dan membuat aturan di `us-east-1` Wilayah AS Timur (Virginia N.).

Berikut ini adalah contoh aturan acara:

```
{
  "source": ["aws.partnercentral-account"],
  "detail": {
    "catalog": ["AWS"]
  }
}
```

AWS Pengaturan SDK

Anda dapat menggunakan AWS SDK untuk membuat dan mengelola EventBridge aturan secara terprogram. Untuk informasi selengkapnya, lihat [PutRule](#) di Referensi Amazon EventBridge API.

Contoh berikut menggunakan AWS SDK for Python (Boto3):

```
import boto3

client = boto3.client('events', region_name='us-east-1')

response = client.put_rule(
    Name='MyConnectionInvitationReceivedRule',
    EventPattern=
    '{
      "source": ["aws.partnercentral-account"],
      "detail-type": ["Partner Connection Invitation Received"],
      "detail": {"catalog": ["AWS"]}
    }',
    State='ENABLED'
)
print('Rule ARN:', response['RuleArn'])
```

Pelajari lebih lanjut tentang peristiwa API koneksi akun

Bagian berikut menjelaskan jenis peristiwa API koneksi akun, skenario yang memicunya, dan contoh peristiwa.

Tipe peristiwa

Berikut ini adalah jenis acara dan pemicunya.

- [Undangan Koneksi Mitra Diterima](#): Memberi tahu penerima bahwa mitra lain ingin membuat koneksi
- [Undangan Koneksi Mitra Diterima](#): Memberi tahu pengirim bahwa undangan mereka diterima dan koneksi sekarang aktif
- [Undangan Koneksi Mitra Ditolak](#): Memberi tahu pengirim bahwa undangan mereka ditolak
- [Koneksi Mitra Dibatalkan](#): Memberi tahu peserta lain yang terhubung saat koneksi aktif dihentikan
- [Undangan Koneksi Mitra Dibatalkan](#): Memberi tahu penerima bahwa pengirim telah menarik undangan mereka sebelum tindakan apa pun diambil
- [Undangan Koneksi Mitra Kedaluwarsa](#): Memberi tahu pengirim dan penerima bahwa undangan telah kedaluwarsa

Contoh peristiwa

Bagian berikut memberikan contoh peristiwa yang tercantum sebelumnya di bagian sebelumnya.

Topik

- [Undangan Koneksi Mitra Diterima](#)
- [Undangan Koneksi Mitra Diterima](#)
- [Undangan Koneksi Mitra Ditolak](#)
- [Koneksi Mitra Dibatalkan](#)
- [Undangan Koneksi Mitra Dibatalkan](#)
- [Undangan Koneksi Mitra Kedaluwarsa](#)

Undangan Koneksi Mitra Diterima

Memicu Konteks

Peristiwa ini dihasilkan ketika undangan koneksi berhasil dibuat dan disimpan di penyimpanan data PAC melalui API. `CreateConnectionInvitation` Acara dikirim segera setelah pembuatan undangan selesai dengan sukses dan disimpan di penyimpanan data PAC, tidak hanya saat panggilan API dilakukan.

Acara akan dikirim secara otomatis ketika undangan koneksi baru dibuat. Satu acara per undangan dibuat tanpa duplikat acara untuk undangan yang sama.

Penerima

AWS Akun penerima dalam undangan koneksi.

Penanganan yang diharapkan

Tindakan Pelanggan Khas:

- Pemberitahuan Segera: Memicu peringatan ke tim bisnis tentang peluang kemitraan baru
- Otomatisasi Alur Kerja: Secara otomatis memperbarui sistem manajemen mitra dengan undangan yang tertunda
- Decision Support: Rute undangan ke pengambil keputusan yang sesuai berdasarkan jenis koneksi
- Audit Logging: Catat tanda terima undangan untuk kepatuhan dan pelacakan kemitraan
- Otomatisasi Respons: Untuk mitra tepercaya, berpotensi menerima jenis undangan tertentu secara otomatis

Pola Integrasi Umum:

- Fungsi Lambda → Perbarui dasbor portal mitra
- Antrian SQS → Undangan proses batch untuk ditinjau
- Topik SNS → Email/Slack pemberitahuan ke tim bisnis
- Tujuan API → Webhook ke sistem CRM/partner manajemen eksternal

Contoh

```
{
```

```

"version": "1",
"id": "<event id>",
"detail-type": "Partner Connection Invitation Received",
"source": "aws.partnercentral-account",
"time": "<ISO 8601 date time>",
"region": "us-east-1",
"account": "<corresponding partner AWS account>",
"resources": [ "<<Connection Invitation ARN>>" ],
"detail": {
  "catalog": "AWS",
  "connectionInvitation" :{
    "arn": "<<Connection Invitation ARN>>",
    "id": "pacinv-****",
    "connectionType": "OpportunityCollaboration",
    "inviterEmail": "abc@def.com",
    "invitationMessage": "We'd like to collaborate on a joint solution for cloud
security. Please accept this connection invitation to proceed.",
    "senderCompanyName": "<<Sender Partner Account Name>>",
    "senderProfileId": "pprofile-****",
    "expiresAt": "<ISO 8601 date time>"
  }
}
}

```

Undangan Koneksi Mitra Diterima

Memicu Konteks

Acara ini dihasilkan ketika undangan koneksi berhasil diterima dan koneksi dibuat dan bertahan di penyimpanan data PAC melalui API. `AcceptConnectionInvitation` Acara dikirim segera setelah penerimaan undangan selesai dengan sukses dan koneksi terjalin.

Acara akan dikirim secara otomatis ketika undangan koneksi diterima. Satu acara per undangan diterima tanpa duplikat acara untuk penerimaan yang sama.

Penerima

Kedua AWS akun yang terlibat dalam undangan koneksi: akun pengirim yang awalnya mengirim undangan dan akun penerima yang menerima undangan.

Penanganan yang diharapkan

Tindakan Pelanggan Khas:

- Aktivasi Kemitraan: Memicu alur kerja untuk mengaktifkan aktivitas bisnis Layer 2
- Pembaruan Status: Perbarui sistem manajemen mitra dengan status koneksi aktif
- Pemberitahuan: Beri tahu tim bisnis tentang keberhasilan pembentukan kemitraan
- Penyediaan Akses: Secara otomatis memberikan izin yang sesuai untuk kolaborasi
- Analytics: Lacak tingkat konversi kemitraan dan metrik keberhasilan

Pola Integrasi Umum:

- Fungsi Lambda → Aktifkan izin berbagi data dan perbarui portal mitra
- Antrian SQS → Aktivasi koneksi proses batch untuk pengaturan bisnis
- Topik SNS → Email/Slack pemberitahuan untuk tim bisnis dan teknis
- Tujuan API → Webhook ke sistem CRM untuk memperbarui status kemitraan

Contoh

```
{
  "version": "1",
  "id": "<event id>",
  "detail-type": "Partner Connection Invitation Accepted",
  "source": "aws.partnercentral-account",
  "time": "<ISO 8601 date time>",
  "region": "us-east-1",
  "resources": [ "<<Connection Invitation ARN>>" , "<<Connection ARN>>" ],
  "account": "<corresponding partner AWS account>",
  "detail": {
    "catalog": "AWS",
    "connectionInvitation" :{
      "arn": "<<Connection Invitation ARN>>",
      "id": "pacinv-****",
      "connectionType": "OpportunityCollaboration"
    },
    "connection": {
      "arn": "<<Connection ARN>>",
      "id": "pac-*****",
      "Participant1AccountId": "<<Sender AWS AccountId>>",
      "Participant2AccountId": "<<Receiver AWS AccountId>>",
      "status": "ACTIVE"
    }
  }
}
```

```
}
```

Undangan Koneksi Mitra Ditolak

Memicu Konteks

Peristiwa ini dihasilkan ketika undangan koneksi berhasil ditolak melalui `RejectConnectionInvitation` API. Acara dikirim segera setelah penolakan undangan diproses dan disimpan di penyimpanan data PAC.

Acara akan dikirim secara otomatis ketika undangan koneksi ditolak. Satu acara per undangan ditolak tanpa duplikat acara untuk penolakan yang sama.

Penerima

AWS Akun yang awalnya mengirim undangan koneksi (akun pengirim).

Penanganan yang diharapkan

Tindakan Pelanggan Khas:

- Pembaruan Status: Perbarui sistem manajemen mitra dengan status penolakan
- Follow-up Tindakan: Memicu alur kerja untuk pendekatan kemitraan alternatif
- Pemberitahuan: Beri tahu tim bisnis tentang keputusan kemitraan
- Pembersihan: Hapus referensi undangan yang tertunda dari sistem internal

Pola Integrasi Umum:

- Fungsi Lambda → Perbarui portal mitra dan picu alur kerja tindak lanjut
- Antrian SQS → Penolakan proses batch untuk analisis bisnis
- Topik SNS → Email/Slack pemberitahuan kepada tim pengembangan bisnis
- Tujuan API → Webhook ke sistem CRM untuk memperbarui status peluang

Contoh

```
{  
  "version": "1",
```

```

    "id": "<event id>",
    "detail-type": "Partner Connection Invitation Rejected",
    "source": "aws.partnercentral-account",
    "time": "<ISO 8601 date time>",
    "region": "us-east-1",
    "account": "<corresponding partner AWS account>",
    "resources": [ "<<Connection Invitation ARN>>" ],
    "detail": {
      "catalog": "AWS",
      "connectionInvitation" :{
        "arn": "<<Connection Invitation ARN>>",
        "id": "pacinv-****",
        "connectionType": "OpportunityCollaboration",
        "invitationMessage": "We'd like to collaborate on a joint solution for cloud
security. Please accept this connection invitation to proceed.",
        "receiverProfileId": "pprofile-****"
      }
    }
  }
}

```

Koneksi Mitra Dibatalkan

Memicu Konteks

Peristiwa ini dihasilkan ketika koneksi aktif berhasil dibatalkan dan status aktif diperbarui untuk dihentikan di penyimpanan data PAC melalui API. CancelConnection Acara dikirim segera setelah pembatalan koneksi diproses dan status koneksi diperbarui.

Acara akan dikirim secara otomatis ketika koneksi dihentikan. Satu acara per koneksi dibatalkan tanpa duplikat acara untuk pembatalan yang sama.

Penerima

AWS Akun peserta lain dalam koneksi (bukan orang yang memulai pembatalan).

Penanganan yang diharapkan

Tindakan Pelanggan Khas:

- Pencabutan Akses: Segera cabut izin dan nonaktifkan berbagi data
- Pembaruan Status: Perbarui sistem manajemen mitra dengan status koneksi yang dihentikan
- Pemberitahuan: Peringatkan tim bisnis dan teknis tentang penghentian kemitraan

- Pembersihan: Hapus referensi koneksi dan bersihkan sumber daya bersama
- Analytics: Lacak durasi koneksi dan pola penghentian

Pola Integrasi Umum:

- Fungsi Lambda → Cabut izin dan perbarui status portal mitra
- Antrian SQS → Penghentian koneksi proses batch untuk alur kerja pembersihan
- Topik SNS → Email/Slack pemberitahuan untuk tim bisnis dan teknis
- Tujuan API → Webhook ke sistem CRM untuk memperbarui status kemitraan

Contoh

```
{
  "version": "1",
  "id": "<event id>",
  "detail-type": "Partner Connection Cancelled",
  "source": "aws.partnercentral-account",
  "time": "<ISO 8601 date time>",
  "region": "us-east-1",
  "resources": [ "<<Connection ARN>>" ],
  "account": "<corresponding partner AWS account>",
  "detail": {
    "catalog": "AWS",
    "connection" :{
      "arn": "<<Connection Invitation ARN>>",
      "id": "pac-****",
      "connectionType": "OpportunityCollaboration",
      "canceledBy": "<<AWS account ID>>"
    }
  }
}
```

Undangan Koneksi Mitra Dibatalkan

Memicu Konteks

Peristiwa ini dihasilkan ketika undangan koneksi yang tertunda berhasil dibatalkan melalui `CancelConnectionInvitation` API. Acara dikirim segera setelah pembatalan undangan diproses dan status undangan diperbarui menjadi DIBATALKAN.

Acara akan dikirim secara otomatis ketika undangan koneksi dibatalkan. Satu acara per undangan koneksi dibatalkan tanpa duplikat acara untuk pembatalan yang sama.

Penerima

AWS Akun yang seharusnya menerima undangan koneksi (akun penerima).

Penanganan yang diharapkan

Tindakan Pelanggan Khas:

- **Pembaruan Status:** Perbarui sistem manajemen mitra untuk menghapus referensi undangan yang tertunda
- **Pemberitahuan:** Beri tahu tim bisnis bahwa peluang kemitraan potensial telah ditarik
- **Pembersihan:** Hapus referensi undangan dari sistem pelacakan internal
- **Analytics:** Lacak pola pembatalan undangan untuk wawasan kemitraan

Pola Integrasi Umum:

- **Fungsi Lambda** → Perbarui portal mitra dan hapus pemberitahuan undangan yang tertunda
- **Antrian SQS** → Pembatalan proses batch untuk analisis bisnis
- **Topik SNS** → Email/Slack pemberitahuan kepada tim pengembangan bisnis
- **Tujuan API** → Webhook ke sistem CRM untuk memperbarui status peluang

Contoh

```
{
  "version": "1",
  "id": "<event id>",
  "detail-type": "Partner Connection Invitation Cancelled",
  "source": "aws.partnercentral-account",
  "time": "<ISO 8601 date time>",
  "region": "us-east-1",
  "account": "<corresponding partner AWS account>",
  "resources": [ "<<Connection Invitation ARN>>" ],
  "detail": {
    "catalog": "AWS",
    "connectionInvitation" :{
```

```
"arn": "<<Connection Invitation ARN>>",
"id": "pacinv-****",
"connectionType": "OpportunityCollaboration",
"invitationMessage": "We'd like to collaborate on a joint solution for cloud
security. Please accept this connection invitation to proceed.",
"senderProfileId": "pprofile-****"
}
}
}
```

Undangan Koneksi Mitra Kedaluwarsa

Memicu Konteks

Acara ini dihasilkan ketika undangan koneksi yang tertunda secara otomatis kedaluwarsa setelah mencapai ExpiresAt stempel waktu tanpa diterima atau ditolak oleh penerima. Acara ini secara otomatis dipicu oleh sistem ketika waktu kedaluwarsa undangan tercapai.

Acara akan dikirim secara otomatis ketika undangan koneksi kedaluwarsa. Satu acara per undangan kedaluwarsa tanpa duplikat acara untuk kedaluwarsa yang sama.

Penerima

Kedua AWS akun yang terlibat dalam undangan koneksi: akun pengirim yang awalnya mengirim undangan dan akun penerima yang menerima undangan.

Penanganan yang diharapkan

Tindakan Pelanggan Khas:

- **Pembaruan Status:** Perbarui sistem manajemen mitra untuk menandai undangan sebagai kedaluwarsa
- **Follow-up Tindakan:** Memicu alur kerja untuk keterlibatan kembali atau pendekatan kemitraan alternatif
- **Pemberitahuan:** Beri tahu tim bisnis tentang peluang kemitraan yang terlewatkan
- **Pembersihan:** Hapus referensi undangan yang kedaluwarsa dari sistem pelacakan internal

Pola Integrasi Umum:

- Fungsi Lambda → Perbarui portal mitra dan picu alur kerja tindak lanjut

- Antrian SQS → Kedaluwarsa proses batch untuk analisis bisnis
- Topik SNS → Email/Slack pemberitahuan kepada tim pengembangan bisnis
- Tujuan API → Webhook ke sistem CRM untuk memperbarui status peluang

Contoh

```
{
  "version": "1",
  "id": "<event id>",
  "detail-type": "Partner Connection Invitation Expired",
  "source": "aws.partnercentral-account",
  "time": "<ISO 8601 date time>",
  "region": "us-east-1",
  "resources": [ "<<Connection Invitation ARN>>" ],
  "detail": {
    "catalog": "AWS",
    "connectionInvitation": {
      "arn": "<<Connection Invitation ARN>>",
      "id": "pacinv-****",
      "connectionType": "OpportunityCollaboration",
      "inviterEmail": "abc@def.com",
      "invitationMessage": "We'd like to collaborate on a joint solution for cloud security. Please accept this connection invitation to proceed.",
      "senderProfileId": "pprofile-****",
      "receiverProfileId": "pprofile-****"
    }
  }
}
```

Pemberitahuan untuk AWS API Penjualan Sentral Mitra

Acara untuk API penjualan memberikan pemberitahuan real-time tentang perubahan status atau detail peluang. Acara ini membantu menjaga sistem Anda tetap sinkron dengan AWS Partner Central, dan membantu memastikan respons dan pembaruan tepat waktu.

Topik

- [Lengkapi prasyarat untuk memantau acara](#)
- [Konfigurasi Amazon EventBridge untuk memantau peristiwa](#)
- [Pelajari selengkapnya tentang menjual event API](#)

Lengkapi prasyarat untuk memantau acara

Pengguna memerlukan izin IAM yang sesuai untuk mengakses dan mengelola acara yang diterbitkan oleh API penjualan Partner Central. Untuk informasi selengkapnya tentang tindakan, sumber daya, dan kunci kondisi yang tersedia EventBridge, lihat [Menggunakan kondisi kebijakan IAM di Amazon EventBridge](#) di Panduan EventBridge Pengguna Amazon. Salah satu kunci kondisi adalah `events:detail-type`, yang dapat digunakan untuk cakupan izin untuk jenis peristiwa tertentu.

Contoh kebijakan berikut menunjukkan cara menyesuaikan dan cakupan izin untuk acara yang diusulkan. `AllowPutRuleForPartnercentralSellingEvents` Pernyataan tersebut memungkinkan pembuatan aturan, tetapi hanya untuk acara dari `aws.partnercentral-selling` sumbernya.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowPutRuleForPartnercentralSellingEvents",
      "Effect": "Allow",
      "Action": "events:PutRule",
      "Resource": "*",
      "Condition": {
        "ForAllValues:StringEquals": {
          "events:source": "aws.partnercentral-selling"
        }
      }
    }
  ]
}
```

Konfigurasi Amazon EventBridge untuk memantau peristiwa

Untuk memantau penjualan peristiwa API, Anda membuat EventBridge aturan yang cocok dengan peristiwa yang ingin Anda tangkap. Anda dapat menggunakan Konsol Manajemen AWS atau AWS SDK untuk membuat dan mengelola aturan. Bagian berikut menjelaskan cara membuat aturan menggunakan kedua metode. Terlepas dari metode yang Anda gunakan, Anda harus membuat aturan di `us-east-1` Wilayah AS Timur (Virginia N.).

Konsol Manajemen AWS mempersiapkan

Untuk menyiapkan EventBridge aturan menggunakan Konsol Manajemen AWS, ikuti langkah-langkah dalam [Membuat aturan yang bereaksi terhadap peristiwa di Amazon EventBridge](#) di Panduan EventBridge Pengguna Amazon. Saat membuat aturan, Anda harus mengatur bus acara ke default, dan membuat aturan di us-east-1 Wilayah AS Timur (Virginia N.).

Berikut ini adalah contoh aturan acara:

```
{
  "source": ["aws.partnercentral-selling"],
  "detail": {
    "catalog": ["AWS"]
  }
}
```

AWS Pengaturan SDK

Anda dapat menggunakan AWS SDK untuk membuat dan mengelola EventBridge aturan secara terprogram. Untuk informasi selengkapnya, lihat [PutRule](#) di Referensi Amazon EventBridge API.

Contoh berikut menggunakan AWS SDK for Python (Boto3):

```
import boto3

client = boto3.client('events', region_name='us-east-1')

response = client.put_rule(
    Name='MyOpportunityCreatedRule',
    EventPattern=
    '{
      "source": ["aws.partnercentral-selling"],
      "detail-type": ["Opportunity Created"],
      "detail": {"catalog": ["AWS"]}
    }',
    State='ENABLED'
)
print('Rule ARN:', response['RuleArn'])
```

Pelajari selengkapnya tentang menjual event API

Bagian berikut menjelaskan jenis peristiwa API penjualan, skenario yang memicunya, dan contoh peristiwa.

Tipe peristiwa

Berikut ini adalah jenis acara dan pemicunya.

- [Peluang Dibuat](#): Dipicu ketika peluang baru dibuat.
- [Peluang Diperbarui](#): Dipicu ketika peluang (Peluang atau Ringkasan AWS Peluang yang sesuai) diperbarui.
- [Undangan Keterlibatan Dibuat](#): Dipicu saat undangan AWS Referral dibuat.
- [Undangan Keterlibatan Diterima](#): Dipicu ketika mitra menerima AWS Undangan Keterlibatan, menegaskan minat mereka untuk berkolaborasi dengan AWS kesempatan tersebut.
- [Undangan Keterlibatan Ditolak](#): Dipicu ketika pasangan menolak AWS Undangan Keterlibatan.
- [Undangan Keterlibatan Kedaluwarsa](#): Acara ini dipicu ketika Mitra menolak. Engagement Invitation Ini memberi tahu mitra pengirim dan penerima bahwa undangan telah kedaluwarsa.
- [Anggota Keterlibatan Ditambahkan](#): Acara ini dipicu ketika anggota baru bergabung dengan Keterlibatan setelah menerima undangan. Ini memberi tahu semua anggota Keterlibatan saat ini tentang anggota baru yang ditambahkan ke Keterlibatan.
- [Snapshot Sumber Daya Keterlibatan Dibuat](#): Peristiwa ini dipicu saat revisi baru Snapshot sumber daya (seperti peluang) yang terkait dengan Keterlibatan dibuat. Ini memberi tahu semua anggota Keterlibatan saat ini tentang perubahan pada Snapshot sumber daya terkait.
- [Keterlibatan Dibuat](#): Dipicu saat keterlibatan baru dibuat.
- [Keterlibatan Diperbarui](#): Dipicu saat keterlibatan diperbarui.

Skenario acara

Tabel berikut menjelaskan bagaimana peristiwa beroperasi di bawah skenario yang berbeda. Asumsi berikut dibuat untuk skenario ini:

- AWS juga merupakan mitra dalam skenario ini.
- `ResourceSnapshotJob` konfigurasi dengan benar oleh mitra.
- Bidang yang diperbarui pada peluang juga ada di template peluang untuk Resource Snapshot.

Tindakan oleh Anda sebagai mitra	Acara yang diterima	Jenis peserta
Anda menciptakan peluang	Peluang Diciptakan	
Anda gunakan <code>StartEngagementFromOpportunityTask</code> untuk mengirimkan kesempatan	Peluang Diperbarui, Keterlibatan Dibuat, Snapshot Sumber Daya Keterlibatan Dibuat, Anggota Keterlibatan Ditambahkan, Undangan Keterlibatan Dibuat	
AWS menyetujui kesempatan yang Anda kirimkan	Undangan Keterlibatan Diterima, Anggota Keterlibatan Ditambahkan, Peluang Diperbarui, Snapshot Sumber Daya Keterlibatan Dibuat	
AWS menolak kesempatan yang Anda kirimkan	Undangan Keterlibatan Ditolak, Peluang Diperbarui, Snapshot Sumber Daya Keterlibatan Dibuat	
Anda mengaitkan AWS Marketplace penawaran dengan peluang	Kesempatan Diperbarui	
Anda mengasosiasikan solusi dengan peluang	Peluang Diperbarui, Snapshot Sumber Daya Keterlibatan Dibuat	
Anda memperbarui peluang	Peluang Diperbarui, Snapshot Sumber Daya Keterlibatan Dibuat (Opsional jika <code>ResourceSnapshotJob</code> diatur dan pembaruan bidang ada di template)	

Tindakan oleh Anda sebagai mitra	Acara yang diterima	Jenis peserta
AWS memperbarui kesempatan Anda	Peluang Diperbarui, Snapshot Sumber Daya Keterlibatan Dibuat	
Anda mengundang pasangan lain ke pertunangan	Undangan Keterlibatan Dibuat	Sender
Undangan Anda untuk bergabung dengan pertunangan diterima oleh mitra	Undangan Keterlibatan Diterima, Anggota Keterlibatan Ditambahkan	Sender
Undangan Anda untuk bergabung dengan pertunangan ditolak oleh pasangan	Undangan Pertunangan Ditolak	Sender
Undangan Anda untuk bergabung dengan pertunangan tidak ditindaklanjuti oleh mitra selama 15 hari	Undangan Keterlibatan Kedaluwarsa	Sender
Anda menerima undangan dari mitra lain untuk bergabung dengan pertunangan	Undangan Keterlibatan Dibuat	Penerima
Anda menerima undangan dari mitra lain untuk bergabung dengan pertunangan melalui <code>StartEngagementByAcceptingInvitationTask</code>	Undangan Keterlibatan Diterima, Anggota Keterlibatan Ditambahkan, Peluang Dibuat, Peluang Diperbarui, Snapshot Sumber Daya Keterlibatan Dibuat	Penerima
Anda menolak undangan untuk bergabung dengan pertunangan dari mitra lain	Undangan Pertunangan Ditolak	Penerima

Tindakan oleh Anda sebagai mitra	Acara yang diterima	Jenis peserta
Anda tidak bertindak atas undangan untuk bergabung dengan pertunangan dari mitra lain selama 15 hari	Undangan Keterlibatan Kedaluwarsa	Penerima
Anda membuat keterlibatan	Keterlibatan Dibuat	Sender
Anda memperbarui pertunangan	Keterlibatan Diperbarui	Pengirim, Penerima

Contoh peristiwa

Bagian berikut memberikan contoh peristiwa yang tercantum sebelumnya di bagian sebelumnya.

Topik

- [Peluang Diciptakan](#)
- [Kesempatan Diperbarui](#)
- [Undangan Keterlibatan Dibuat](#)
- [Undangan Keterlibatan Diterima](#)
- [Undangan Pertunangan Ditolak](#)
- [Undangan Keterlibatan Kedaluwarsa](#)
- [Anggota Keterlibatan Ditambahkan](#)
- [Snapshot Sumber Daya Keterlibatan Dibuat](#)
- [Keterlibatan Dibuat](#)
- [Keterlibatan Diperbarui](#)

Peluang Diciptakan

Mitra menggunakan acara ini untuk:

- Beri tahu pengguna mereka bahwa Peluang baru telah dibuat.

- Memicu alur kerja otomatis seperti memperbarui sistem CRM atau memberi tahu tim penjualan tentang kreasi peluang baru.

Contoh berikut menunjukkan acara khas yang Dibuat Peluang.

```
{
  "version": "1",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "source": "aws.partnercentral-selling",
  "detail-type": "Opportunity Created",
  "time": "2023-04-16T15:23:45Z",
  "region": "us-east-1",
  "account": "123456789012",
  "detail": {
    "schemaVersion": "<version number>",
    "catalog": "<Sandbox | AWS>",
    "opportunity": {
      "identifier": "String"
    }
  }
}
```

Kesempatan Diperbarui

Mitra menggunakan acara ini untuk:

- Beri tahu pengguna mereka bahwa Peluang yang ada telah diperbarui.
- Memicu alur kerja otomatis seperti memperbarui sistem CRM atau memberi tahu tim penjualan.

Contoh berikut menunjukkan acara Opportunity Updated yang khas.

```
{
  "version": "1",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "source": "aws.partnercentral-selling",
  "detail-type": "Opportunity Updated",
  "time": "2023-04-16T15:23:45Z",
  "region": "us-east-1",
  "account": "123456789012",
  "detail": {
    "schemaVersion": "<version number>",
```



```

    "catalog": "<Sandbox | AWS>",
    "opportunity": {
        "identifier": "String"
    }
}
}

```

Undangan Keterlibatan Dibuat

Mitra menggunakan acara ini untuk:

- Beri tahu pengguna mereka tentang hal baru yang telah EngagementInvitation mereka terima.
- Memicu alur kerja otomatis untuk menerima atau menolak undangan, seperti memperbarui sistem CRM atau memberi tahu tim penjualan.

Contoh berikut menunjukkan acara khas Undangan Keterlibatan Dibuat.

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Engagement Invitation Created",
  "source": "aws.partnercentral-selling",
  "account": "123456789012",
  "time": "2023-04-15T12:34:56Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-invitation/engi-
v7p8z56whnauo"
  ],
  "detail": {
    "catalog": "AWS",
    "engagementInvitation": {
      "arn": "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-invitation/
engi-v7p8z56whnauo",
      "id": "engi-v7p8z56whnauo",
      "engagementId": "eng-12345678901234",
      "senderAccountId": "string",
      "receiverAccountId": "string",
      "senderCompanyName": "string",
      "expirationDate": "string",
      "participantType": "Enum", //Sender/Receiver
      "payloadType": "LeadInvitation" //OpportunityInvitation|LeadInvitation
    }
  }
}

```

```

    }
  }
}

```

Undangan Keterlibatan Diterima

Mitra menggunakan acara ini untuk:

- Beri tahu pengguna mereka bahwa EngagementInvitation telah diterima.
- Perbarui catatan internal mereka untuk mencerminkan mitra baru dalam Keterlibatan.
- Picu alur kerja otomatis untuk menyinkronkan data atau memberi tahu anggota tim lain tentang anggota Keterlibatan baru.

Contoh berikut menunjukkan acara khas Undangan Keterlibatan Diterima.

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Engagement Invitation Accepted",
  "source": "aws.partnercentral-selling",
  "account": "123456789012",
  "time": "2023-04-16T15:23:45Z",
  "region": "us-east-1",
  "resources": ["arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-invitation/engi-v7p8z56whnauo"],
  "detail": {
    "catalog": "AWS",
    "engagementInvitation": {
      "arn": "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-invitation/engi-v7p8z56whnauo",
      "id": "engi-v7p8z56whnauo",
      "engagementId": "eng-12356whnauo",
      "participantType": "Enum", //Sender/Receiver
      "senderAccountId": "string",
      "receiverAccountId": "string",
      "payloadType": "LeadInvitation" //OpportunityInvitation|LeadInvitation
    }
  }
}

```

Undangan Pertunangan Ditolak

Mitra menggunakan acara ini untuk:

- Beri tahu pengguna mereka bahwa EngagementInvitation telah ditolak.
- Perbarui catatan internal mereka untuk mencerminkan undangan yang ditolak.
- Picu alur kerja yang diperlukan, seperti memberi tahu tim penjualan tentang penolakan.

Contoh berikut menunjukkan acara khas Undangan Keterlibatan Ditolak.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Engagement Invitation Rejected",
  "source": "aws.partnercentral-selling",
  "account": "123456789012",
  "time": "2023-04-17T09:48:27Z",
  "region": "us-east-1",
  "resources": ["arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-invitation/engi-v7p8z56whnauo"],
  "detail": {
    "catalog": "AWS",
    "engagementInvitation": {
      "arn": "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-invitation/engi-v7p8z56whnauo",
      "id": "engi-v7p8z56whnauo",
      "engagementId": "eng-12356whnauo",
      "participantType": "Enum", //Sender/Receiver
      "senderAccountId": "string",
      "receiverAccountId": "string",
      "payloadType": "LeadInvitation" //OpportunityInvitation|LeadInvitation
    }
  }
}
```

Undangan Keterlibatan Kedaluwarsa

Mitra menggunakan acara ini untuk:

- Beri tahu pengguna mereka bahwa EngagementInvitation telah kedaluwarsa dan tidak dapat lagi ditindaklanjuti.

- Perbarui catatan internal mereka untuk mencerminkan undangan yang kedaluwarsa.
- Picu alur kerja yang diperlukan, seperti memberi tahu tim penjualan tentang undangan yang kedaluwarsa.

Contoh berikut menunjukkan acara Kedaluwarsa Undangan Keterlibatan yang khas.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Engagement Invitation Expired",
  "source": "aws.partnercentral-selling",
  "account": "123456789012",
  "time": "2023-04-18T18:20:15Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-invitation/engi-
v7p8z56whnauo"
  ],
  "detail": {
    "catalog" : "AWS",
    "engagementInvitation": {
      "arn": "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-invitation/
engi-v7p8z56whnauo",
      "id": "engi-v7p8z56whnauo",
      "engagementId": "eng-12356whnauo",
      "participantType": "Enum", //Sender/Receiver
      "senderAccountId": "string",
      "receiverAccountId": "string",
      "payloadType": "LeadInvitation" //OpportunityInvitation|LeadInvitation
    }
  }
}
```

Anggota Keterlibatan Ditambahkan

Mitra menggunakan acara ini untuk:

- Beri tahu pengguna mereka tentang perubahan pada keanggotaan Keterlibatan.
- Perbarui catatan internal mereka untuk mencerminkan anggota Keterlibatan baru.
- Picu alur kerja apa pun yang diperlukan, seperti memberi tahu anggota tim tentang perubahan tersebut.

Contoh berikut menunjukkan acara Added Anggota Keterlibatan yang khas.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Engagement Member Added",
  "source": "aws.partnercentral-selling",
  "account": "123456789012",
  "time": "2023-04-16T15:23:45Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement/eng-v7p8z56whnauo"
  ],
  "detail": {
    "catalog" : "AWS",
    "engagement": {
      "id": "eng-v7p8z56whnauo"
    },
    "engagementMember": {
      "accountId": "string",
      "companyName": "string"
    }
  }
}
```

Snapshot Sumber Daya Keterlibatan Dibuat

Mitra menggunakan acara ini untuk melacak perubahan pada sumber daya yang terkait dengan keterlibatan dan memicu alur kerja otomatis, seperti memperbarui sistem CRM atau memberi tahu tim penjualan. Template snapshot menentukan jenis pembaruan:

OpportunitySummaryView

- Beri tahu semua anggota tentang keterlibatan bahwa keterlibatan telah diperbarui.

AwsOpportunitySummaryFullView

- Lacak pembaruan yang dibuat secara khusus oleh AWS ke peluang dalam keterlibatan.
- Pemberitahuan dan snapshot ini hanya dapat dilihat oleh pemilik peluang.
- Termasuk pembaruan ukuran kesepakatan, yang tidak tersedia melalui OpportunitySummaryView template.

Contoh berikut menunjukkan acara khas Engagement Resource Snapshot Created.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Engagement Resource Snapshot Created",
  "source": "aws.partnercentral-selling",
  "account": "123456789012",
  "time": "2023-04-18T18:20:15Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement/eng-v7p8z56whnauo"
  ],
  "detail": {
    "catalog" : "AWS",
    "resourceSnapshot": {
      "arn": "arn:aws:partnercentral-selling:us-east-1::catalog/AWS/engagement/eng-v7p8z56whnauo/resource/Opportunity/o-12312/template/temp-name/snapshot/snapshot-19232",
      "engagementId": "eng-v7p8z56whnauo",
      "resourceType": "Opportunity",
      "resourceId" : "0123211231",
      "createdBy": "123123123123",
      "targetMemberAccounts": ["123123123123", "aws"],
      "resourceSnapshotTemplateName": "temp-name"
    }
  }
}
```

Keterlibatan Dibuat

Mitra menggunakan acara ini untuk:

- Beri tahu pengguna mereka bahwa Keterlibatan baru telah dibuat.
- Memicu alur kerja otomatis seperti memperbarui sistem CRM atau memberi tahu tim penjualan tentang kreasi keterlibatan baru.

Contoh berikut menunjukkan event khas Engagement Created.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Engagement Created",
  "source": "aws.partnercentral-selling",
```

```

    "account": "123456789012",
    "time": "2023-04-18T18:20:15Z",
    "region": "us-east-1",
    "resources": [
      "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement/eng-567a1j5hxp351o"
    ],
    "detail": {
      "catalog": "AWS",
      "engagement": {
        "engagementArn": "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement/eng-567a1j5hxp351o",
        "engagementId": "eng-567a1j5hxp351o",
        "CreatedAt": "2023-04-18T18:20:15Z",
        "CreatedBy": "aws",
        "ContextTypes": ["Lead"]
      }
    }
  }
}

```

Keterlibatan Diperbarui

Mitra menggunakan acara ini untuk:

- Beri tahu pengguna mereka bahwa Keterlibatan yang ada telah diperbarui.
- Memicu alur kerja otomatis seperti memperbarui sistem CRM atau memberi tahu tim penjualan.

Contoh berikut menunjukkan acara Engagement Updated yang khas.

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Engagement Updated",
  "source": "aws.partnercentral-selling",
  "account": "123456789012",
  "time": "2023-04-18T18:20:15Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement/eng-567a1j5hxp351o"
  ],
  "detail": {
    "catalog": "AWS",

```

```
    "engagement": {
      "engagementArn": "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement/
eng-567a1j5hxp351o",
      "engagementId": "eng-567a1j5hxp351o",
      "LastModifiedAt": "2023-04-18T18:20:15Z",
      "LastModifiedBy": "aws",
      "ContextTypes": ["Lead", "CustomerProject"]
    }
  }
}
```

Pemberitahuan untuk AWS API Manfaat Sentral Mitra

Layanan Manfaat menyediakan EventBridge acara untuk keduanya BenefitApplications dan BenefitAllocations. Peristiwa ini memungkinkan pelanggan untuk membangun arsitektur berbasis peristiwa komprehensif yang merespons perubahan di seluruh siklus hidup sumber daya - dari Aplikasi awal hingga Alokasi dan Pemenuhan akhir.

Topik

- [Lengkapi prasyarat untuk memantau acara](#)
- [Konfigurasi Amazon EventBridge untuk memantau peristiwa](#)
- [Pelajari selengkapnya tentang manfaat peristiwa API](#)

Lengkapi prasyarat untuk memantau acara

Pengguna memerlukan izin IAM yang sesuai untuk mengakses dan mengelola peristiwa yang diterbitkan oleh API manfaat. Untuk informasi selengkapnya tentang tindakan, sumber daya, dan kunci kondisi yang tersedia EventBridge, lihat [Menggunakan kondisi kebijakan IAM di Amazon EventBridge](#) di Panduan EventBridge Pengguna Amazon. Salah satu kunci kondisi adalah `events:detail-type`, yang dapat digunakan untuk cakupan izin untuk jenis peristiwa tertentu.

Contoh kebijakan berikut menunjukkan cara menyesuaikan dan cakupan izin untuk acara yang diusulkan. `AllowPutRuleForPartnercentralBenefitsEvents` Pernyataan tersebut memungkinkan pembuatan aturan, tetapi hanya untuk acara dari `aws.partnercentral-benefits` sumbernya.

Untuk contoh kebijakan IAM yang mendetail, lihat dokumentasi AWS tentang EventBridge izin.


```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowPutRuleForPartnerCentralBenefitsEvents",
      "Effect": "Allow",
      "Action": "events:PutRule",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "events:source": "aws.partnercentral-benefits"
        }
      }
    }
  ]
}
```

Konfigurasi Amazon EventBridge untuk memantau peristiwa

Untuk memantau peristiwa API manfaat, Anda membuat EventBridge aturan yang cocok dengan peristiwa yang ingin Anda tangkap. Anda dapat menggunakan Konsol Manajemen AWS atau AWS SDK untuk membuat dan mengelola aturan. Bagian berikut menjelaskan cara membuat aturan menggunakan kedua metode. Terlepas dari metode yang Anda gunakan, Anda harus membuat aturan di us-east-1 Wilayah AS Timur (Virginia N.).

Konsol Manajemen AWS mempersiapkan

Untuk menyiapkan EventBridge aturan menggunakan Konsol Manajemen AWS, ikuti langkah-langkah dalam [Membuat aturan yang bereaksi terhadap peristiwa di Amazon EventBridge](#) di Panduan EventBridge Pengguna Amazon. Saat membuat aturan, Anda harus mengatur bus acara ke default, dan membuat aturan di us-east-1 Wilayah AS Timur (Virginia N.).

Berikut ini adalah contoh aturan acara:

```
{
  "source": ["aws.partnercentral-benefits"],
  "detail": {
    "catalog": ["AWS"]
  }
}
```

AWS Pengaturan SDK

Anda dapat menggunakan AWS SDK untuk membuat dan mengelola EventBridge aturan secara terprogram. Untuk informasi selengkapnya, lihat [PutRule](#) di Referensi Amazon EventBridge API.

Contoh berikut menggunakan AWS SDK for Python (Boto3):

```
import boto3

client = boto3.client('events', region_name='us-east-1')

response = client.put_rule(
    Name='MyBenefitApplicationCreatedRule',
    EventPattern=
    '{
        "source": ["aws.partnercentral-benefits"],
        "detail-type": ["Benefit Application Created"],
        "detail": {"catalog": ["AWS"]}
    }',
    State='ENABLED'
)
print('Rule ARN:', response['RuleArn'])
```

Pelajari selengkapnya tentang manfaat peristiwa API

Bagian berikut menjelaskan manfaat jenis peristiwa API, skenario yang memicunya, dan contoh peristiwa.

Tipe peristiwa

Berikut ini adalah jenis acara dan pemicunya.

Aplikasi Manfaat

- [Aplikasi Manfaat Dibuat](#): Saat Aplikasi Manfaat dibuat
- [Aplikasi Manfaat Dalam Tinjauan](#): Ketika Aplikasi Manfaat diajukan atau ketika Aplikasi beralih dari Tindakan yang Diperlukan kembali ke Dalam Tinjauan, acara ini dipicu.
- [Tindakan Aplikasi Manfaat Diperlukan](#): Peristiwa ini terjadi ketika peninjau internal memperbarui status aplikasi untuk menunjukkan bahwa mitra perlu memberikan informasi tambahan, klarifikasi, atau modifikasi sebelum peninjauan dapat dilanjutkan.

- [Aplikasi Manfaat Ditolak](#): Peristiwa ini terjadi ketika reviewer internal memperbarui status aplikasi untuk menunjukkan bahwa aplikasi tidak memenuhi kriteria atau persyaratan manfaat dan telah ditolak
- [Aplikasi Manfaat Disetujui](#): Acara ini terjadi ketika peninjau internal memperbarui status aplikasi untuk menunjukkan bahwa aplikasi telah disetujui untuk alokasi manfaat.
- [Tahap Aplikasi Manfaat Berubah](#): Peristiwa ini terjadi ketika peninjau internal memperbarui tahap aplikasi selama proses peninjauan seperti Tinjauan Bisnis, Tinjauan, AWS Tinjauan Keuangan, dll.

Alokasi Manfaat

- [Alokasi Manfaat Diaktifkan](#): Ketika Alokasi Manfaat dibuat atau diperbarui kembali ke Diaktifkan oleh layanan internal.
- [Alokasi Manfaat Tidak Aktif](#): Peristiwa ini terjadi ketika reviewer internal memperbarui status alokasi atau ketika Alokasi Manfaat berakhir secara otomatis.
- [Alokasi Manfaat Terpenuhi](#): Acara ini terjadi ketika reviewer internal memperbarui status alokasi untuk menunjukkan bahwa alokasi telah sepenuhnya digunakan.

Contoh peristiwa

Bagian berikut memberikan contoh peristiwa yang tercantum sebelumnya di bagian sebelumnya.

Topik

- [Aplikasi Manfaat Dibuat](#)
- [Aplikasi Manfaat Dalam Tinjauan](#)
- [Tindakan Aplikasi Manfaat Diperlukan](#)
- [Aplikasi Manfaat Ditolak](#)
- [Aplikasi Manfaat Disetujui](#)
- [Tahap Aplikasi Manfaat Berubah](#)
- [Alokasi Manfaat Diaktifkan](#)
- [Alokasi Manfaat Tidak Aktif](#)
- [Alokasi Manfaat Terpenuhi](#)

Aplikasi Manfaat Dibuat

Memicu Konteks

Acara Benefit Application Created dipublikasikan ke Amazon EventBridge ketika Aplikasi Benefit baru berhasil dibuat melalui Layanan Manfaat. Peristiwa ini terjadi ketika panggilan CreateBenefitApplication API berhasil diselesaikan, yang menunjukkan bahwa mitra telah memulai permintaan untuk manfaat tertentu.

Acara ini merupakan awal dari siklus hidup aplikasi manfaat dan memberi pelanggan pemberitahuan segera ketika aplikasi baru dikirimkan ke akun mereka.

Penerima

Akun pemilik akan menerima acara untuk Alokasi Manfaat.

Contoh

```
{
  "id": "7gh88765-k0jh-d08g-i292-2ff1796m030n",
  "detail-type": "Benefit Application Created",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-02-10T15:45:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-application/benappl-12345abcdef123"
  ],
  "detail": {
    "catalog": "AWS",
    "benefitApplication": {
      "id": "benappl-12345abcdef123",
      "fulfillmentTypes": ["CREDITS"],
      "status": "PENDING_SUBMISSION",
      "revision": "hh7e8400-e29b-41d4-a716-446655440012"
    },
    "benefit": {
      "id": "ben-xyz789uvw012",
      "name": "AWS Partner Credits",
      "program": ["AWS Partner Network"]
    }
  }
}
```

```
}
```

Aplikasi Manfaat Dalam Tinjauan

Memicu Konteks

Acara Benefit Application In Review dipublikasikan ke Amazon EventBridge ketika Aplikasi Benefit beralih ke status IN_REVIEW melalui Layanan Manfaat. Peristiwa ini terjadi ketika:

- Mitra mengirimkan aplikasi mereka melalui panggilan SubmitBenefitApplication API
- Transisi aplikasi dari ACTION_REQUIRED kembali ke IN_REVIEW setelah mitra menangani umpan balik
- Aplikasi ditarik kembali dan kemudian dikirimkan kembali

Acara ini menunjukkan bahwa aplikasi telah memasuki proses peninjauan formal dan sedang dievaluasi oleh tim AWS internal.

Penerima

Akun pemilik akan menerima acara untuk Alokasi Manfaat.

Contoh

```
{
  "id": "8hi99876-l1ki-e19h-j303-3gg2807n141o",
  "detail-type": "Benefit Application In Review",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-02-10T16:00:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-application/benappl-12345abcdef123"
  ],
  "detail": {
    "catalog": "AWS",
    "benefitApplication": {
      "id": "benappl-12345abcdef123",
      "fulfillmentTypes": ["CREDITS"],
      "status": "IN_REVIEW",
      "stage": "BUSINESS_REVIEW",
      "revision": "ii8f9511-f30c-52e5-b827-557766551123"
    }
  }
}
```

```

    },
    "benefit": {
      "id": "ben-xyz789uvw012",
      "name": "AWS Partner Credits",
      "program": ["AWS Partner Network"]
    }
  }
}

```

Tindakan Aplikasi Manfaat Diperlukan

Memicu Konteks

Peristiwa Benefit Application Action Required dipublikasikan ke Amazon EventBridge saat Aplikasi Benefit beralih ke status ACTION_REQUIRED melalui Layanan Manfaat. Peristiwa ini terjadi ketika reviewer internal memperbarui status aplikasi melalui UpdateBenefitApplicationInternal API untuk menunjukkan bahwa mitra perlu memberikan informasi tambahan, klarifikasi, atau modifikasi sebelum peninjauan dapat dilanjutkan.

Peristiwa ini merupakan titik kritis dalam siklus hidup aplikasi di mana intervensi mitra diperlukan untuk memajukan aplikasi dalam proses peninjauan.

Penerima

Akun pemilik akan menerima acara untuk Alokasi Manfaat

Contoh

```

{
  "id": "9ij00987-m2lj-f20i-k414-4hh3918o252p",
  "detail-type": "Benefit Application Action Required",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-02-12T10:30:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-application/benappl-12345abcdef123"
  ],
  "detail": {
    "catalog": "AWS",
    "benefitApplication": {
      "id": "benappl-12345abcdef123",

```

```

    "fulfillmentTypes": ["CREDITS"],
    "status": "ACTION_REQUIRED",
    "stage": "BUSINESS_REVIEW",
    "revision": "jj9g0622-g41d-63f6-c938-668877662234"
  },
  "benefit": {
    "id": "ben-xyz789uvw012",
    "name": "AWS Partner Credits",
    "program": ["AWS Partner Network"]
  }
}
}

```

Aplikasi Manfaat Ditolak

Memicu Konteks

Acara yang Ditolak Aplikasi Manfaat dipublikasikan ke Amazon EventBridge saat Aplikasi Manfaat beralih ke status DITOLAK melalui Layanan Manfaat. Peristiwa ini terjadi ketika reviewer internal memperbarui status aplikasi melalui UpdateBenefitApplicationInternal API untuk menunjukkan bahwa aplikasi tidak memenuhi kriteria manfaat atau persyaratan dan telah ditolak.

Peristiwa ini mewakili status terminal dalam siklus hidup aplikasi di mana aplikasi telah ditolak secara resmi dan tidak akan melanjutkan ke alokasi manfaat.

Penerima

Akun pemilik akan menerima acara untuk Alokasi Manfaat.

Contoh

```

{
  "id": "0kl11098-n3mk-g31j-l525-5ii4029p363q",
  "detail-type": "Benefit Application Rejected",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-02-15T14:20:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-application/benappl-12345abcdef123"
  ],
  "detail": {
    "catalog": "AWS",

```

```
"benefitApplication": {
  "id": "benappl-12345abcdef123",
  "fulfillmentTypes": ["CREDITS"],
  "status": "REJECTED",
  "revision": "kk0h1733-h52e-74g7-d049-779988773345"
},
"benefit": {
  "id": "ben-xyz789uvw012",
  "name": "AWS Partner Credits",
  "program": ["AWS Partner Network"]
}
}
```

Aplikasi Manfaat Disetujui

Memicu Konteks

Acara yang Disetujui Aplikasi Manfaat dipublikasikan ke Amazon EventBridge saat Aplikasi Manfaat beralih ke status DISETUJUI melalui Layanan Manfaat. Peristiwa ini terjadi ketika reviewer internal memperbarui status aplikasi melalui UpdateBenefitApplicationInternal API untuk menunjukkan bahwa aplikasi memenuhi semua kriteria manfaat dan persyaratan dan telah disetujui untuk alokasi manfaat.

Acara ini merupakan keberhasilan penyelesaian proses peninjauan aplikasi dan biasanya memicu pembuatan Alokasi Manfaat yang sesuai.

Penerima

Akun pemilik akan menerima acara untuk Alokasi Manfaat.

Contoh

```
{
  "id": "1lm22109-o4nl-h42k-m636-6jj5130q474r",
  "detail-type": "Benefit Application Approved",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-02-14T16:45:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-application/benappl-12345abcdef123"
  ],
```



```

"detail": {
  "catalog": "AWS",
  "benefitApplication": {
    "id": "benappl-12345abcdef123",
    "fulfillmentTypes": ["CREDITS"],
    "status": "APPROVED",
    "revision": "111i2844-i63f-85h8-e150-880099884456"
  },
  "benefit": {
    "id": "ben-xyz789uvw012",
    "name": "AWS Partner Credits",
    "program": ["AWS Partner Network"]
  }
}
}

```

Tahap Aplikasi Manfaat Berubah

Memicu Konteks

Acara Benefit Application Stage Changed dipublikasikan ke Amazon EventBridge ketika tahap peninjauan Aplikasi Benefit diperbarui melalui Layanan Manfaat. Peristiwa ini terjadi ketika reviewer internal memperbarui Tahap aplikasi ke salah satu nilai Enum yang mungkin (FINANCE_REVIEW, BUSINESS_REVIEW, TECH_REVIEW dan _REVIEW).AWS

Tidak seperti perubahan status, perubahan tahap adalah pembaruan informasi hanya-baca yang memberikan visibilitas ke dalam perkembangan alur kerja tinjauan internal tanpa memerlukan tindakan mitra.

Penerima

Akun pemilik akan menerima acara untuk Alokasi Manfaat.

Contoh

```

{
  "id": "2mn33210-p5om-i53l-n747-7kk6241r585s",
  "detail-type": "Benefit Application Stage Changed",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-02-13T11:15:00Z",
  "region": "us-east-1",
  "resources": [

```

```

    "arn:aws:partnercentral:us-east-1:123456789012:benefit-application/
benappl-12345abcdef123"
  ],
  "detail": {
    "catalog": "AWS",
    "benefitApplication": {
      "id": "benappl-12345abcdef123",
      "fulfillmentTypes": ["CREDITS"],
      "status": "IN_REVIEW",
      "stage": "FINANCIAL_REVIEW",
      "revision": "mm2j3955-j74g-96i9-f261-991100995567"
    },
    "benefit": {
      "id": "ben-xyz789uvw012",
      "name": "AWS Partner Credits",
      "program": ["AWS Partner Network"]
    }
  }
}

```

Alokasi Manfaat Diaktifkan

Memicu Konteks

Acara Aktif Alokasi Manfaat dipublikasikan ke Amazon EventBridge saat Alokasi Manfaat beralih ke status AKTIF melalui Layanan Manfaat. Peristiwa ini terjadi ketika reviewer internal memperbarui status alokasi melalui UpdateBenefitAllocationInternal API untuk mengaktifkan kembali alokasi yang sebelumnya tidak aktif.

Peristiwa ini biasanya terjadi ketika alokasi yang dinonaktifkan sementara (karena kedaluwarsa, perubahan kebijakan, atau alasan administratif) dikembalikan ke status aktif, sehingga manfaat tersedia untuk pemanfaatan mitra lagi.

Penerima

AWS Akun milik Mitra akan menerima pesan tersebut. Dengan kata lain, AWS Akun yang memiliki sumber daya.

Contoh Acara

```

{
  "id": "3no44321-q6pn-j64m-o858-8117352s696t",
  "detail-type": "Benefit Allocation Activated",

```

```
"source": "aws.partnercentral-benefits",
"account": "123456789012",
"time": "2025-03-01T08:15:00Z",
"region": "us-east-1",
"resources": [
  "arn:aws:partnercentral:us-east-1:123456789012:benefit-allocation/benalloc-
abc123def456"
],
"detail": {
  "catalog": "AWS",
  "benefitAllocation": {
    "id": "benalloc-abc123def456",
    "fulfillmentType": "CREDITS",
    "status": "ACTIVE"
  },
  "benefitApplication": {
    "id": "benappl-12345abcdef123"
  },
  "benefit": {
    "id": "ben-xyz789uvw012",
    "name": "AWS Partner Credits",
    "program": ["AWS Partner Network"]
  }
}
```

Alokasi Manfaat Tidak Aktif

Memicu Konteks

Acara Alokasi Manfaat Tidak Aktif dipublikasikan ke Amazon EventBridge ketika Alokasi Manfaat beralih ke status TIDAK AKTIF melalui Layanan Manfaat. Peristiwa ini terjadi ketika reviewer internal memperbarui status alokasi melalui UpdateBenefitAllocationInternal API atau ketika Alokasi Manfaat kedaluwarsa secara otomatis.

Peristiwa ini biasanya terjadi ketika alokasi ditangguhkan sementara (karena kedaluwarsa, perubahan kebijakan, masalah kepatuhan, atau alasan administratif), membuat manfaat tidak tersedia untuk pemanfaatan mitra hingga diaktifkan kembali.

Penerima

AWS Akun milik Mitra akan menerima pesan tersebut. Dengan kata lain, AWS Akun yang memiliki sumber daya.

Contoh Acara

```
{
  "id": "4op55432-r7qo-k75n-p969-9mm8463t707u",
  "detail-type": "Benefit Allocation Inactivated",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-06-30T23:59:59Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-allocation/benalloc-abc123def456"
  ],
  "detail": {
    "catalog": "AWS",
    "benefitAllocation": {
      "id": "benalloc-abc123def456",
      "fulfillmentType": "CREDITS",
      "status": "INACTIVE"
    },
    "benefitApplication": {
      "id": "benappl-12345abcdef123"
    },
    "benefit": {
      "id": "ben-xyz789uvw012",
      "name": "AWS Partner Credits",
      "program": ["AWS Partner Network"]
    }
  }
}
```

Alokasi Manfaat Terpenuhi

Memicu Konteks

Acara Terpenuhi Alokasi Manfaat dipublikasikan ke Amazon EventBridge ketika Alokasi Manfaat beralih ke status TERPENUHI melalui Layanan Manfaat. Peristiwa ini terjadi ketika reviewer internal memperbarui status alokasi melalui UpdateBenefitAllocationInternal API untuk menunjukkan bahwa alokasi telah sepenuhnya digunakan atau bahwa persyaratan manfaat telah dipenuhi.

Acara ini merupakan penyelesaian siklus hidup manfaat, yang menunjukkan bahwa mitra telah berhasil memanfaatkan manfaat yang dialokasikan atau bahwa alokasi telah mencapai kesimpulan alaminya.

Penerima

AWS Akun milik Mitra akan menerima pesan tersebut. Dengan kata lain, AWS Akun yang memiliki sumber daya.

Contoh Acara

```
{
  "id": "5pq66543-s8rp-l86o-q070-0nn9574u818v",
  "detail-type": "Benefit Allocation Fulfilled",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-12-15T17:30:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-allocation/benalloc-abc123def456"
  ],
  "detail": {
    "catalog": "AWS",
    "benefitAllocation": {
      "id": "benalloc-abc123def456",
      "fulfillmentType": "CREDITS",
      "status": "FULFILLED"
    },
    "benefitApplication": {
      "id": "benappl-12345abcdef123"
    },
    "benefit": {
      "id": "ben-xyz789uvw012",
      "name": "AWS Partner Credits",
      "program": ["AWS Partner Network"]
    }
  }
}
```

Pemberitahuan untuk AWS API Saluran Pusat Mitra

Acara untuk API channel memberikan notifikasi real-time tentang perubahan status atau detail aktivitas terkait saluran. Acara ini membantu menjaga sistem Anda tetap sinkron dengan AWS Partner Central, dan membantu memastikan respons dan pembaruan tepat waktu.

Prasyarat

Anda memerlukan izin IAM yang sesuai untuk mengakses dan mengelola peristiwa dari API Saluran Pusat AWS Mitra. Untuk informasi tentang tindakan, sumber daya, dan kunci kondisi yang tersedia EventBridge, lihat [Menggunakan kondisi kebijakan IAM di Amazon EventBridge](#) di Panduan EventBridge Pengguna Amazon. Salah satu kunci kondisi adalah `events:detail-type`, yang dapat Anda gunakan untuk cakupan izin untuk jenis peristiwa tertentu.

Contoh kebijakan berikut menunjukkan cara menyesuaikan dan cakupan izin untuk acara. `AllowPutRuleForPartnercentralChannelEvents` Pernyataan tersebut memungkinkan pembuatan aturan, tetapi hanya untuk acara dari `aws.partnercentral-channel` sumbernya.

Untuk contoh kebijakan IAM yang mendetail, lihat dokumentasi AWS tentang EventBridge izin.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowPutRuleForPartnerCentralChannelEvents",
      "Effect": "Allow",
      "Action": "events:PutRule",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "events:source": "aws.partnercentral-channel"
        }
      }
    }
  ]
}
```

Konfigurasi Amazon EventBridge untuk memantau peristiwa

Untuk memantau peristiwa API channel, Anda membuat EventBridge aturan yang cocok dengan peristiwa yang ingin Anda tangkap. Anda dapat menggunakan Konsol Manajemen AWS atau AWS SDK untuk membuat dan mengelola aturan. Bagian berikut menjelaskan cara membuat aturan menggunakan kedua metode. Terlepas dari metode yang Anda gunakan, Anda harus membuat aturan di `us-east-1` Wilayah AS Timur (Virginia N.).

Konsol Manajemen AWS mempersiapkan

Untuk menyiapkan EventBridge aturan menggunakan Konsol Manajemen AWS, ikuti langkah-langkah dalam [Membuat aturan yang bereaksi terhadap peristiwa di Amazon EventBridge](#) di Panduan EventBridge Pengguna Amazon. Saat membuat aturan, Anda harus mengatur bus acara ke default, dan membuat aturan di us-east-1 Wilayah AS Timur (Virginia N.).

Berikut ini adalah contoh aturan acara:

```
{
  "source": ["aws.partnercentral-channel"],
  "detail": {
    "catalog": ["AWS"]
  }
}
```

AWS Pengaturan SDK

Anda dapat menggunakan AWS SDK untuk membuat dan mengelola EventBridge aturan secara terprogram. Untuk informasi selengkapnya, lihat [PutRule](#) di Referensi Amazon EventBridge API.

Contoh berikut menggunakan AWS SDK for Python (Boto3):

```
import boto3

client = boto3.client('events', region_name='us-east-1')

response = client.put_rule(
    Name='ChannelHandshakeCreatedRule',
    EventPattern=
    '{
      "source": ["aws.partnercentral-channel"],
      "detail-type": ["Channel Handshake Created"],
      "detail": {"catalog": ["AWS"]}
    }',
    State='ENABLED'
)
print('Rule ARN:', response['RuleArn'])
```

Pelajari selengkapnya tentang peristiwa API channel

Bagian berikut menjelaskan jenis peristiwa API channel, skenario yang memicunya, dan contoh peristiwa.

Tipe peristiwa

Berikut ini adalah jenis acara dan pemicunya.

- [Jabat Tangan Saluran Dibuat](#): Dipicu saat jabat tangan saluran baru dibuat.
- [Jabat Tangan Saluran Diterima](#): Dipicu saat jabat tangan saluran baru diterima.
- [Jabat Tangan Saluran Ditolak](#): Dipicu saat jabat tangan saluran baru ditolak.

Jabat Tangan Saluran Dibuat

Contoh berikut menunjukkan acara Channel Handshake Created khas untuk jenis jabat tangan yang berbeda.

Mulai Jabat Tangan Periode Layanan:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Channel Handshake Created",
  "source": "aws.partnercentral-channel",
  "account": "789789789789",
  "time": "2025-02-01T00:00:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/ch-abc123def456g"
  ],
  "detail": {
    "requestId": "12345678-1234-1234-1234-123456789abc",
    "catalog": "AWS",
    "associatedResourceIdentifier": "rs-jkl012mno345p",
    "handshakeType": "START_SERVICE_PERIOD",
    "id": "ch-abc123def456g",
    "receiverAccountId": "789789789789",
    "ownerAccountId": "123123123123",
    "senderAccountId": "456456456456",
    "senderDisplayName": "TestDisplayName",
```



```

    "handshakeDetail": {
      "startServicePeriodHandshakeDetail": {
        "servicePeriodType": "MINIMUM_NOTICE_PERIOD",
        "minimumNoticeDays": "14",
        "endDate": null,
        "startDate": "2025-02-02T00:00:00Z",
        "note": "Test note example"
      }
    }
  }
}

```

Mencabut Jabat Tangan Periode Layanan:

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Channel Handshake Created",
  "source": "aws.partnercentral-channel",
  "account": "789789789789",
  "time": "2025-02-01T00:00:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/ch-def456ghi789j"
  ],
  "detail": {
    "requestId": "12345678-1234-1234-1234-123456789abc",
    "catalog": "AWS",
    "associatedResourceIdentifier": "rs-jkl012mno345p",
    "handshakeType": "REVOKE_SERVICE_PERIOD",
    "id": "ch-def456ghi789j",
    "ownerAccountId": "123123123123",
    "receiverAccountId": "789789789789",
    "senderAccountId": "456456456456",
    "senderDisplayName": "TestDisplayName",
    "handshakeDetail": {
      "revokeServicePeriodHandshakeDetail": {
        "servicePeriodType": "MINIMUM_NOTICE_PERIOD",
        "minimumNoticeDays": "14",
        "endDate": null,
        "startDate": "2025-02-02T00:00:00Z",
        "note": "Test note example"
      }
    }
  }
}

```

```

    }
  }
}

```

Jabat Tangan Akun Manajemen Program:

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Channel Handshake Created",
  "source": "aws.partnercentral-channel",
  "account": "456456456456",
  "time": "2025-02-01T00:00:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/ch-ghi789jkl012m"
  ],
  "detail": {
    "requestId": "12345678-1234-1234-1234-123456789abc",
    "catalog": "AWS",
    "associatedResourceIdentifier": "pma-abc123def456g",
    "handshakeType": "PROGRAM_MANAGEMENT_ACCOUNT",
    "id": "ch-ghi789jkl012m",
    "ownerAccountId": "123123123123",
    "receiverAccountId": "456456456456",
    "senderAccountId": "123123123123",
    "senderDisplayName": "TestDisplayName",
    "handshakeDetail": {
      "programManagementAccountHandshakeDetail": {
        "program": "SOLUTION_PROVIDER"
      }
    }
  }
}

```

Jabat Tangan Saluran Diterima

Contoh berikut menunjukkan acara Channel Handshake Accepted yang khas untuk jenis jabat tangan yang berbeda.

Mulai Jabat Tangan Periode Layanan:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Channel Handshake Accepted",
  "source": "aws.partnercentral-channel",
  "account": "123123123123",
  "time": "2025-02-01T00:00:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/ch-abc123def456g"
  ],
  "detail": {
    "requestId": "12345678-1234-1234-1234-123456789abc",
    "catalog": "AWS",
    "associatedResourceIdentifier": "rs-jkl012mno345p",
    "handshakeType": "START_SERVICE_PERIOD",
    "id": "ch-abc123def456g",
    "ownerAccountId": "123123123123",
    "receiverAccountId": "789789789789",
    "senderAccountId": "456456456456",
    "senderDisplayName": "TestDisplayName",
    "handshakeDetail": {
      "startServicePeriodHandshakeDetail": {
        "servicePeriodType": "MINIMUM_NOTICE_PERIOD",
        "minimumNoticeDays": "14",
        "endDate": null,
        "startDate": "2025-02-02T00:00:00Z",
        "note": "Test note example"
      }
    }
  }
}
```

Mencabut Jabat Tangan Periode Layanan:

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Channel Handshake Accepted",
  "source": "aws.partnercentral-channel",
  "account": "123123123123",
  "time": "2025-02-01T00:00:00Z",
```

```

    "region": "us-east-1",
    "resources": [
        "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/
ch-def456ghi789j"
    ],
    "detail": {
        "requestId": "12345678-1234-1234-1234-123456789abc",
        "catalog": "AWS",
        "associatedResourceIdentifier": "rs-jkl012mno345p",
        "handshakeType": "REVOKE_SERVICE_PERIOD",
        "id": "ch-def456ghi789j",
        "ownerAccountId": "123123123123",
        "receiverAccountId": "789789789789",
        "senderAccountId": "456456456456",
        "senderDisplayName": "TestDisplayName",
        "handshakeDetail": {
            "revokeServicePeriodHandshakeDetail": {
                "servicePeriodType": "MINIMUM_NOTICE_PERIOD",
                "minimumNoticeDays": "14",
                "endDate": null,
                "startDate": "2025-02-02T00:00:00Z",
                "note": "Test note example"
            }
        }
    }
}

```

Jabat Tangan Akun Manajemen Program:

```

{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Channel Handshake Accepted",
    "source": "aws.partnercentral-channel",
    "account": "123123123123",
    "time": "2025-02-01T00:00:00Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/
ch-ghi789jkl012m"
    ],
    "detail": {
        "requestId": "12345678-1234-1234-1234-123456789abc",

```

```

    "catalog": "AWS",
    "associatedResourceIdentifier": "pma-abc123def456g",
    "handshakeType": "PROGRAM_MANAGEMENT_ACCOUNT",
    "id": "ch-ghi789jkl012m",
    "ownerAccountId": "123123123123",
    "receiverAccountId": "456456456456",
    "senderAccountId": "123123123123",
    "senderDisplayName": "TestDisplayName",
    "handshakeDetail": {
      "programManagementAccountHandshakeDetail": {
        "program": "SOLUTION_PROVIDER"
      }
    }
  }
}

```

Jabat Tangan Saluran Ditolak

Contoh berikut menunjukkan peristiwa Channel Handshake Rejected khas untuk jenis jabat tangan yang berbeda.

Mulai Jabat Tangan Periode Layanan:

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Channel Handshake Rejected",
  "source": "aws.partnercentral-channel",
  "account": "123123123123",
  "time": "2025-02-01T00:00:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/ch-abc123def456g"
  ],
  "detail": {
    "requestId": "12345678-1234-1234-1234-123456789abc",
    "catalog": "AWS",
    "associatedResourceIdentifier": "rs-jkl012mno345p",
    "handshakeType": "START_SERVICE_PERIOD",
    "id": "ch-abc123def456g",
    "ownerAccountId": "123123123123",
    "receiverAccountId": "789789789789",

```

```

    "senderAccountId": "456456456456",
    "senderDisplayName": "TestDisplayName",
    "handshakeDetail": {
      "startServicePeriodHandshakeDetail": {
        "servicePeriodType": "MINIMUM_NOTICE_PERIOD",
        "minimumNoticeDays": "14",
        "endDate": null,
        "startDate": "2025-02-02T00:00:00Z",
        "note": "Test note example"
      }
    }
  }
}

```

Mencabut Jabat Tangan Periode Layanan:

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Channel Handshake Rejected",
  "source": "aws.partnercentral-channel",
  "account": "123123123123",
  "time": "2025-02-01T00:00:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/ch-def456ghi789j"
  ],
  "detail": {
    "requestId": "12345678-1234-1234-1234-123456789abc",
    "catalog": "AWS",
    "associatedResourceIdentifier": "rs-jkl012mno345p",
    "handshakeType": "REVOKE_SERVICE_PERIOD",
    "id": "ch-def456ghi789j",
    "ownerAccountId": "123123123123",
    "receiverAccountId": "789789789789",
    "senderAccountId": "456456456456",
    "senderDisplayName": "TestDisplayName",
    "handshakeDetail": {
      "revokeServicePeriodHandshakeDetail": {
        "servicePeriodType": "MINIMUM_NOTICE_PERIOD",
        "minimumNoticeDays": "14",
        "endDate": null,

```

```

        "startDate": "2025-02-02T00:00:00Z",
        "note": "Test note example"
    }
}

```

Jabat Tangan Akun Manajemen Program:

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Channel Handshake Rejected",
  "source": "aws.partnercentral-channel",
  "account": "123123123123",
  "time": "2025-02-01T00:00:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/ch-ghi789jkl012m"
  ],
  "detail": {
    "requestId": "12345678-1234-1234-1234-123456789abc",
    "catalog": "AWS",
    "associatedResourceIdentifier": "pma-abc123def456g",
    "handshakeType": "PROGRAM_MANAGEMENT_ACCOUNT",
    "id": "ch-ghi789jkl012m",
    "ownerAccountId": "123123123123",
    "receiverAccountId": "456456456456",
    "senderAccountId": "123123123123",
    "senderDisplayName": "TestDisplayName",
    "handshakeDetail": {
      "programManagementAccountHandshakeDetail": {
        "program": "SOLUTION_PROVIDER"
      }
    }
  }
}

```

Pengujian di kotak pasir

Mitra dapat menggunakan kotak pasir untuk menguji dan memvalidasi interaksi API Pusat AWS Mitra mereka di lingkungan yang aman dan terisolasi, memastikan kelancaran operasi sebelum mempromosikan solusi mereka ke lingkungan produksi. AWS menawarkan kotak pasir dinamis kepada pengguna API Pusat AWS Mitra yang mengembalikan respons yang mirip dengan lingkungan produksi. AWS tidak menyediakan antarmuka pengguna ke lingkungan kotak pasir. Oleh karena itu, mitra perlu mengandalkan respons terprogram untuk menguji solusi mereka.

Akses ke lingkungan kotak pasir

Mitra mendapatkan akses ke lingkungan pengujian segera setelah mereka menautkan Akun AWS ke akun Pusat Mitra. Untuk informasi selengkapnya, lihat [Menautkan AWS akun Anda ke akun AWS Partner Central](#). Setiap permintaan menyertakan catalog parameter, yang menentukan lingkungan data. Ketika catalog diatur ke AWS, itu referensi data produksi, dan ketika diatur ke Sandbox, itu referensi data kotak pasir.

Detail penting tentang lingkungan kotak pasir

1. **Penyegaran data:** Sekali setahun, AWS menyegarkan data di lingkungan kotak pasir (biasanya di awal tahun). Setelah penyegaran ini, Anda mungkin kehilangan beberapa data di lingkungan pengujian Anda.
2. **Lingkup pengujian:** Lingkungan kotak pasir biasanya digunakan untuk pengujian fungsional dan bukan untuk menguji skalabilitas atau kinerja.

Topik

- [Pengujian di kotak pasir untuk AWS API Akun Sentral Mitra](#)
- [Pengujian di kotak pasir untuk AWS API Penjualan Sentral Mitra](#)
- [Pengujian di kotak pasir untuk AWS API Manfaat Sentral Mitra](#)
- [Pengujian di kotak pasir untuk AWS API Saluran Pusat Mitra](#)
- [Pengujian di kotak pasir untuk AWS API Pengukuran Pendapatan Pusat Mitra](#)

Pengujian di kotak pasir untuk AWS API Akun Sentral Mitra

Cara menggunakan kotak pasir

Untuk menggunakan kotak pasir, selesaikan langkah-langkah berikut:

1. Buat peran IAM:

Buat peran IAM dalam Akun AWS tautan dengan akun Pusat AWS Mitra Anda.

2. Tetapkan Kebijakan:

Tetapkan kebijakan berikut ke peran IAM. Untuk informasi selengkapnya, lihat [Menambahkan dan menghapus izin identitas IAM](#).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AccountManagement",
      "Effect": "Allow",
      "Action": [
        "partnercentral:AcceptConnectionInvitation",
        "partnercentral:AssociateAwsTrainingCertificationEmailDomain",
        "partnercentral:CancelConnection",
        "partnercentral:CancelConnectionInvitation",
        "partnercentral:CancelProfileUpdateTask",
        "partnercentral:CreateConnectionInvitation",
        "partnercentral:CreatePartner",
        "partnercentral:DisassociateAwsTrainingCertificationEmailDomain",
        "partnercentral:GetAllianceLeadContact",
        "partnercentral:GetConnection",
        "partnercentral:GetConnectionInvitation",
        "partnercentral:GetConnectionPreferences",
        "partnercentral:GetPartner",
        "partnercentral:GetProfileUpdateTask",
        "partnercentral:GetProfileVisibility",
        "partnercentral:GetVerification",
        "partnercentral:ListConnectionInvitations",
        "partnercentral:ListConnections",
        "partnercentral:ListPartners",
        "partnercentral:PutAllianceLeadContact",
        "partnercentral:PutProfileVisibility",
        "partnercentral:RejectConnectionInvitation",
```

```

        "partnercentral:SendEmailVerificationCode",
        "partnercentral:StartProfileUpdateTask",
    "partnercentral:StartVerification",
        "partnercentral:UpdateConnectionPreferences"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "partnercentral:Catalog": [
                "Sandbox"
            ]
        }
    }
},
{
    "Sid": "TaggingAccess",
    "Effect": "Allow",
    "Action": [
        "partnercentral:TagResource",
        "partnercentral:UntagResource",
        "partnercentral:ListTagsForResource"
    ],
    "Resource": [
        "arn:aws:partnercentral:*:*:catalog/Sandbox/partner/*"
    ],
    "Condition": {
        "StringEquals": {
            "partnercentral:Catalog": [
                "Sandbox"
            ]
        }
    }
}
]
}

```

3. Gunakan kredensial peran IAM:

Gunakan kredensial (kunci rahasia dan kunci akses) peran IAM ini dalam solusi Anda untuk melakukan tindakan API.

Pengujian AWS tindakan

Selama fase pengujian, seringkali perlu untuk mensimulasikan AWS tindakan. Simulasi ini memungkinkan mitra untuk menguji secara menyeluruh aliran end-to-end lengkap integrasi mereka dengan layanan AWS. Setiap permintaan menyertakan parameter katalog, yang menentukan lingkungan data.

Mensimulasikan penciptaan Mitra

Untuk mensimulasikan penciptaan Mitra, dalam muatan `CreatePartner` tindakan, termasuk `"catalog": "Sandbox"` dalam muatan.

Contoh:

```
{
  "ClientToken": "client-generated-uuid",
  "Catalog": "Sandbox",
  "LegalName": "Your Name",
  "PrimarySolutionType": "SOLUTION_TYPE",
  "AllianceLeadContact": {
    "FirstName": "Example First Name",
    "LastName": "Example Last Name",
    "BusinessTitle": "Example Title",
    "Email": "example@domain.com"
  },
  "EmailVerificationCode": "123456"
}
```

Catatan: Di Sandbox verifikasi email dinonaktifkan. Anda dapat menggunakan enam digit apa pun `EmailVerificationCode` untuk pengujian. Juga, jangan gunakan “Tag” dalam permintaan.

Catatan pengujian tambahan

1. Untuk menguji tindakan lain, atur `"catalog": "Sandbox"` di payload.
2. Untuk menguji koneksi akun mitra di kotak pasir, Anda harus menjalankan `StartProfileUpdateTask` tindakan setelah membuat mitra di katalog kotak pasir.
3. Untuk pindah ke produksi, ubah nilai katalog dari `Sandbox` ke `AWS`.

Menguji peristiwa di lingkungan kotak pasir

Mitra dapat menggunakan peristiwa dari lingkungan kotak pasir untuk membantu menguji implementasi berbasis peristiwa. Siapkan EventBridge sama Akun AWS dengan aturan untuk mendengarkan acara kotak pasir dengan menentukan catalog: Sandbox detail acara. Untuk informasi selengkapnya, lihat [Pemberitahuan untuk AWS API Rekening Sentral Mitra](#).

Contoh aturan acara:

```
{
  "source": ["aws.partnercentral-account"],
  "detail": {
    "catalog": ["Sandbox"]
  }
}
```

Aturan peristiwa yang ditentukan catalog sebagai hanya Sandbox akan cocok dengan peristiwa yang berasal dari kotak pasir, yang dihasilkan oleh tindakan yang Anda lakukan di lingkungan kotak pasir.

Pengujian di kotak pasir untuk AWS API Penjualan Sentral Mitra

Cara menggunakan kotak pasir

1. Buat peran IAM:

Buat peran IAM di akun yang Akun AWS ditautkan dengan akun Pusat AWS Mitra Anda.

2. Tetapkan Kebijakan:

Tetapkan kebijakan berikut ke peran IAM. Untuk informasi selengkapnya, lihat [Menambahkan dan menghapus izin identitas IAM](#).

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "partnercentral:CreateOpportunity",

```

```

        "partnercentral:UpdateOpportunity",
        "partnercentral:ListOpportunities",
        "partnercentral:GetOpportunity",
        "partnercentral:GetAwsOpportunitySummary",
        "partnercentral:ListSolutions",
        "partnercentral:AssociateOpportunity",
        "partnercentral:DisassociateOpportunity",
        "partnercentral:AssignOpportunity",
        "partnercentral:SubmitOpportunity",
        "partnercentral:AcceptEngagementInvitation",
        "partnercentral:CreateEngagementInvitation",
        "partnercentral:RejectEngagementInvitation",
        "partnercentral:GetEngagementInvitation",
        "partnercentral:ListEngagementInvitations",
        "partnercentral:StartEngagementFromOpportunityTask",
        "partnercentral:StartEngagementByAcceptingInvitationTask",
        "partnercentral:CreateResourceSnapshotJob",
        "partnercentral:StartResourceSnapshotJob",
        "partnercentral:CreateEngagement"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "partnercentral:Catalog": "Sandbox"
        }
    }
},
{
    "Effect": "Allow",
    "Action": [
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeEntity"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "aws-marketplace:SearchAgreements",
        "aws-marketplace:DescribeAgreement"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {

```

```

    "aws-marketplace:PartyType": "Proposer"
  }
}
}
}

```

3. Gunakan kredensial peran IAM:

Gunakan kredensial (kunci rahasia dan kunci akses) peran IAM ini dalam solusi Anda untuk melakukan tindakan API.

Pengujian AWS tindakan

Selama fase pengujian, seringkali perlu untuk mensimulasikan AWS tindakan. Simulasi ini memungkinkan mitra untuk menguji secara menyeluruh aliran end-to-end lengkap integrasi mereka dengan layanan AWS.

Mensimulasikan penciptaan sebuah AWS Kesempatan Berasal

Untuk mensimulasikan penciptaan peluang yang AWS berasal, dalam muatan CreateOpportunity tindakan, termasuk dan. "Catalog": "Sandbox" "Origin": "AWS Referral"

Contoh:

```

{
  "Catalog": "Sandbox",
  "Origin": "AWS Referral",
  "OpportunityIdentifier": "0123456",
  "Title": "Test Opportunity",
  ...
}

```

Mensimulasikan pembaruan untuk peluang yang berasal dari mitra

Untuk mensimulasikan pembaruan atau tindakan lain pada peluang yang berasal dari mitra, gunakan UpdateOpportunity tindakan dengan "Catalog": "Sandbox" dan Lifecycle.ReviewStatus: "Approved" atau "Action Required" dalam muatan.

Jika Anda mengambil payload dari GetOpportunity tindakan untuk melakukan pembaruan, pastikan Anda mengubah ID menjadi Identifier, dan menghapus bidang berikut:

- CreatedDate
- OpportunityTeam
- RelatedEntityIdentifiers

Contoh:

```
{
  "Catalog": "Sandbox",
  "Identifier": "0123456",
  "Title": "Updated Test Opportunity",
  "Lifecycle": {
    "ReviewStatus": "Approved"
  }
  ...
}
```

Menguji peristiwa di lingkungan kotak pasir

Mitra dapat menggunakan peristiwa peluang dari lingkungan kotak pasir untuk membantu menguji implementasi berbasis peristiwa. Siapkan EventBridge sama Akun AWS dengan aturan untuk mendengarkan acara kotak pasir dengan menentukan catalog: sandbox detail acara. Untuk informasi selengkapnya, lihat [Pemberitahuan untuk AWS API Penjualan Sentral Mitra](#).

Contoh aturan acara:

```
{
  "source": ["aws.partnercentral-selling"],
  "detail": {
    "catalog": ["Sandbox"]
  }
}
```

Aturan peristiwa yang ditentukan catalog sebagai hanya Sandbox akan cocok dengan peristiwa yang berasal dari kotak pasir, yang dihasilkan oleh tindakan yang Anda lakukan di lingkungan kotak pasir.

Catatan pengujian tambahan

1. AssociateOpportunityTindakan pengujian:

- a. Gunakan solusi default "S-1234567" untuk menguji AssociateOpportunity tindakan.

- b. Untuk menguji penawaran Marketplace, gunakan ID penawaran nyata dari akun Anda.
2. Untuk pindah ke produksi, ubah catalog nilainya dari Sandbox keAWS.

Mendapatkan bantuan

Jika Anda menghadapi tantangan mengintegrasikan CRM Anda dengan AWS, atau jika Anda perlu menguji skenario tertentu yang tidak tercakup di sini, hubungi dukungan dengan mengajukan kasus melalui langkah-langkah berikut:

1. Masuk ke [Pusat AWS Mitra](#) dengan AWS kredensi Jaringan Mitra Anda.
2. Pada [Support Center for Partner Central](#), pilih Open New Case untuk mencatat kasus baru. Lengkapi bidang sebagai berikut:
 - a. Jenis kasus dukungan:AWS Partner Central.
 - b. Pertanyaan tentang:Partner Central Tools or ACE leads and opportunities.
 - c. Dapatkan spesifik: Pilih jenis kasus Integrasi CRM yang paling tepat.
 - d. Subjek: Sertakan deskripsi singkat tentang permintaan.
 - e. Deskripsi: Berikan deskripsi rinci tentang masalah, pertanyaan, kesalahan, dan langkah pemecahan masalah.
 - f. Lampiran: Sertakan log dan tangkapan layar, jika berlaku.

Pengujian di kotak pasir untuk AWS API Manfaat Sentral Mitra

Cara menggunakan kotak pasir

Untuk menggunakan kotak pasir, selesaikan langkah-langkah berikut:

1. Buat peran IAM:

Buat peran IAM dalam Akun AWS tautan dengan akun Pusat AWS Mitra Anda.

2. Tetapkan Kebijakan:

Tetapkan kebijakan berikut ke peran IAM. Untuk informasi selengkapnya, lihat [Menambahkan dan menghapus izin identitas IAM](#).

```
{  
  "Version": "2012-10-17",
```



```

"Statement": [{
  "Sid": "BenefitsManagement",
  "Effect": "Allow",
  "Action": [
    "partnercentral:ListBenefits",
    "partnercentral:GetBenefit",
    "partnercentral:CreateBenefitApplication",
    "partnercentral:AmendBenefitApplication",
    "partnercentral:UpdateBenefitApplication",
    "partnercentral:SubmitBenefitApplication",
    "partnercentral:GetBenefitApplication",
    "partnercentral:CancelBenefitApplication",
    "partnercentral:RecallBenefitApplication",
    "partnercentral:ListBenefitApplications",
    "partnercentral:AssociateBenefitApplicationResource",
    "partnercentral:DisasssociateBenefitApplicationResource",
    "partnercentral:ListBenefitAllocations",
    "partnercentral:GetBenefitAllocation",
    "partnercentral:TagResource",
    "partnercentral:UntagResource",
    "partnercentral:ListTagsForResource"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "partnercentral:Catalog": [
        "Sandbox"
      ]
    }
  }
},
{
  "Sid": "AWSPartnerOpportunityAccess",
  "Effect": "Allow",
  "Action": [
    "partnercentral:GetAwsOpportunitySummary",
    "partnercentral:GetOpportunity",
    "partnercentral:ListOpportunities"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "partnercentral:Catalog": [
        "Sandbox"
      ]
    }
  }
}
]

```

```

        ]
    }
}
},
{
    "Sid": "ListingAWSMarketplaceEntities",
    "Effect": "Allow",
    "Action": [
        "aws-marketplace:ListEntities"
    ],
    "Resource": "*"
},
{
    "Sid": "AWSMarketplaceOffersAccess",
    "Effect": "Allow",
    "Action": [
        "aws-marketplace:DescribeEntity"
    ],
    "Resource": [
        "arn:aws:aws-marketplace::AWSMarketplace/Offer/*"
    ]
},
{
    "Sid": "S3PutObjectAccess",
    "Effect": "Allow",
    "Action": [
        "s3:PutObject"
    ],
    "Resource": ["arn:aws:s3::aws-partner-central-marketplace-ephemeral-
writeonly-files/*"]
},
{
    "Sid": "TaggingAccess",
    "Effect": "Allow",
    "Action": [
        "partnercentral:TagResource",
        "partnercentral:UntagResource",
        "partnercentral:ListTagsForResource"
    ],
    "Resource": [
        "arn:aws:partnercentral:us-east-1:*:catalog/Sandbox/benefit-
application/*",

```

```

        "arn:aws:partnercentral:us-east-1:*:catalog/Sandbox/benefit-
allocation/*"
    ],
    "Condition": {
        "StringEquals": {
            "partnercentral:Catalog": [
                "Sandbox"
            ]
        }
    }
}
]
}

```

3. Gunakan kredensial peran IAM:

Gunakan kredensial (kunci rahasia dan kunci akses) peran IAM ini dalam solusi Anda untuk melakukan tindakan API.

Pengujian AWS tindakan

Selama fase pengujian, seringkali perlu untuk mensimulasikan AWS tindakan. Simulasi ini memungkinkan mitra untuk menguji secara menyeluruh aliran end-to-end lengkap integrasi mereka dengan layanan AWS. Setiap permintaan menyertakan parameter katalog, yang menentukan lingkungan data.

Mensimulasikan pembuatan Aplikasi Manfaat

Untuk mensimulasikan pembuatan aplikasi manfaat, dalam muatan `CreateBenefitApplication` tindakan, termasuk `"catalog": "Sandbox"` dalam muatan.

Contoh:

```

{
  "Catalog": "Sandbox",
  "ClientToken": "String",
  "BenefitIdentifier": "String",
  "FulfillmentTypes": ["String"],
  "BenefitApplicationDetails": JsonDocument,
  "Name": "String",
  "Description": "String",
  "Tags": [

```

```

    {
      "Key": "string",
      "Value": "string"
    }
  ],
  "PartnerContacts": [
    {
      "BusinessTitle": "string",
      "Email": "string",
      "FirstName": "string",
      "LastName": "string",
      "Phone": "string"
    }
  ],
  "FileDetails": [
    {
      "FileURI": "String",
      "BusinessUseCase": "String"
    }
  ],
  "AssociatedResources": [
    {
      "ResourceType": "BENEFIT_ALLOCATION | OPPORTUNITY",
      "ResourceArn": "ARN"
    }
  ]
}

```

Catatan pengujian tambahan

1. Untuk menguji tindakan lain, atur "catalog": "Sandbox" di payload.
2. ID Manfaat dalam katalog Sandbox dan katalog AWS berbeda. Lakukan panggilan GetBenefit API "catalog": "Sandbox" untuk mendapatkan Pengenal Manfaat di Kotak Pasir.
3. Alokasi Benefit dalam katalog Sandbox dan katalog AWS berbeda.
4. Untuk pindah ke produksi, ubah nilai katalog dari Sandbox keAWS.

Menguji peristiwa di lingkungan kotak pasir

Mitra dapat menggunakan peristiwa dari lingkungan kotak pasir untuk membantu menguji implementasi berbasis peristiwa. Siapkan EventBridge sama Akun AWS dengan aturan untuk

mendengarkan acara kotak pasir dengan menentukan catalog: Sandbox detail acara. Untuk informasi selengkapnya, lihat [Pemberitahuan untuk AWS API Manfaat Sentral Mitra](#).

Contoh aturan acara:

```
{
  "source": ["aws.partnercentral-benefits"],
  "detail": {
    "catalog": ["Sandbox"]
  }
}
```

Aturan peristiwa yang ditentukan catalog sebagai hanya Sandbox akan cocok dengan peristiwa yang berasal dari kotak pasir, yang dihasilkan oleh tindakan yang Anda lakukan di lingkungan kotak pasir.

Pengujian di kotak pasir untuk AWS API Saluran Pusat Mitra

Akun di kotak pasir tidak memenuhi syarat untuk mendapatkan manfaat saluran, dan tidak divalidasi terhadap semua persyaratan program.

Cara menggunakan kotak pasir

Untuk menggunakan kotak pasir, selesaikan langkah-langkah berikut:

1. Buat peran IAM:

Buat peran IAM dalam Akun AWS tautan dengan akun Pusat AWS Mitra Anda.

2. Tetapkan Kebijakan:

Tetapkan kebijakan berikut ke peran IAM. Untuk informasi selengkapnya, lihat [Menambahkan dan menghapus izin identitas IAM](#).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ChannelManagement",
      "Effect": "Allow",
      "Action": [
        "partnercentral:CreateProgramManagementAccount",
        "partnercentral:UpdateProgramManagementAccount",

```

```

    "partnercentral:DeleteProgramManagementAccount",
    "partnercentral:ListProgramManagementAccounts",
    "partnercentral:GetProgramManagementAccount",
    "partnercentral:CreateRelationship",
    "partnercentral:UpdateRelationship",
    "partnercentral:DeleteRelationship",
    "partnercentral:GetRelationship",
    "partnercentral:ListRelationships",
    "partnercentral:CreateChannelHandshake",
    "partnercentral:AcceptChannelHandshake",
    "partnercentral:RejectChannelHandshake",
    "partnercentral:CancelChannelHandshake",
    "partnercentral:ListChannelHandshakes"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "partnercentral:Catalog": [
        "Sandbox"
      ]
    }
  }
},
{
  "Sid": "TaggingAccess",
  "Effect": "Allow",
  "Action": [
    "partnercentral:TagResource",
    "partnercentral:UntagResource",
    "partnercentral:ListTagsForResource"
  ],
  "Resource": [
    "arn:aws:partnercentral:*:*:catalog/Sandbox/program-management-account/*",
    "arn:aws:partnercentral:*:*:catalog/Sandbox/channel-handshake/*"
  ],
  "Condition": {
    "StringEquals": {
      "partnercentral:Catalog": [
        "Sandbox"
      ]
    }
  }
}
]

```

```
}
```

3. Gunakan kredensial peran IAM:

Gunakan kredensial (kunci rahasia dan kunci akses) peran IAM ini dalam solusi Anda untuk melakukan tindakan API.

Pengujian AWS tindakan

Selama fase pengujian, seringkali perlu untuk mensimulasikan AWS tindakan. Simulasi ini memungkinkan mitra untuk menguji secara menyeluruh aliran end-to-end lengkap integrasi mereka dengan layanan AWS. Setiap permintaan menyertakan parameter katalog, yang menentukan lingkungan data.

Mensimulasikan pembuatan Akun Manajemen Program

Untuk mensimulasikan pembuatan Akun Manajemen Program, dalam muatan `CreateProgramManagementAccount` tindakan, sertakan `"catalog": "Sandbox"` dalam muatan.

Contoh:

```
{
  "catalog": "Sandbox",
  "accountId": "123456789012",
  "displayName": "Test PMA Name",
  "clientToken": "123abc456def789ghi",
  "program": "SOLUTION_PROVIDER",
  "tags": [
    {
      "key": "testkey",
      "value": "testvalue"
    }
  ]
}
```

Mensimulasikan pembaruan untuk Hubungan

Untuk mensimulasikan pembaruan pada suatu hubungan, gunakan `UpdateRelationship` tindakan dengan `"catalog": "Sandbox"` di payload.

Contoh:

```
{
  "catalog": "Sandbox",
  "displayName": "Updated Test PMA",
  "identifier": "rs-abc123def456g",
  "programManagementAccountIdentifier": "pma-123abc456def7"
}
```

Catatan pengujian tambahan

1. Untuk menguji tindakan lain, atur "catalog": "Sandbox" di payload.
2. Untuk pindah ke produksi, ubah nilai katalog dari Sandbox keAWS.

Menguji peristiwa di lingkungan kotak pasir

Mitra dapat menggunakan peristiwa dari lingkungan kotak pasir untuk membantu menguji implementasi berbasis peristiwa. Siapkan EventBridge sama Akun AWS dengan aturan untuk mendengarkan acara kotak pasir dengan menentukan catalog: Sandbox detail acara. Untuk informasi selengkapnya, lihat [Pemberitahuan untuk AWS API Saluran Pusat Mitra](#).

Contoh aturan acara:

```
{
  "source": ["aws.partnercentral-channel"],
  "detail": {
    "catalog": ["Sandbox"]
  }
}
```

Aturan peristiwa yang ditentukan catalog sebagai hanya Sandbox akan cocok dengan peristiwa yang berasal dari kotak pasir, yang dihasilkan oleh tindakan yang Anda lakukan di lingkungan kotak pasir.

Mendapatkan bantuan

Jika Anda menghadapi tantangan dalam pengujian, atau jika Anda perlu menguji skenario tertentu yang tidak tercakup di sini, hubungi dukungan dengan mengajukan kasus melalui langkah-langkah berikut:

1. Masuk ke [Pusat AWS Mitra](#) dengan AWS kredensi Jaringan Mitra Anda.

2. Pada [Support Center for Partner Central](#), pilih Open New Case untuk mencatat kasus baru. Lengkapi bidang sebagai berikut:
 - a. Jenis kasus dukungan: Channel Program
 - b. Pertanyaan tentang: General Program Inquiry
 - c. Dapatkan spesifik: Pilih jenis kasus Saluran yang paling sesuai.
 - d. Subjek: Sertakan deskripsi singkat tentang permintaan.
 - e. Deskripsi: Berikan deskripsi rinci tentang masalah, pertanyaan, kesalahan, dan langkah pemecahan masalah.
 - f. Lampiran: Sertakan log dan tangkapan layar, jika berlaku.

Pengujian di kotak pasir untuk AWS API Pengukuran Pendapatan Pusat Mitra

Cara menggunakan kotak pasir

Untuk menggunakan kotak pasir, selesaikan langkah-langkah berikut:

1. Buat peran IAM:

Buat peran IAM di akun yang Akun AWS ditautkan dengan akun Pusat AWS Mitra Anda.

2. Tetapkan Kebijakan:

Tetapkan kebijakan berikut ke peran IAM. Untuk informasi selengkapnya, lihat [Menambahkan dan menghapus izin identitas IAM](#).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "RevenueMeasurementCreation",
      "Effect": "Allow",
      "Action": [
        "partnercentral:CreateRevenueAttribution",
        "partnercentral:CreateMarketplaceRevenueShare",
        "partnercentral:CreateMarketplaceRevenueShareAllocation"
      ],
      "Resource": "*",
      "Condition": {
```

```

        "StringEquals": {
            "partnercentral:Catalog": [
                "Sandbox"
            ]
        }
    },
    {
        "Sid": "RevenueMeasurementListing",
        "Effect": "Allow",
        "Action": [
            "partnercentral:ListRevenueAttributions",
            "partnercentral:ListRevenueAttributionAllocations",
            "partnercentral:ListMarketplaceRevenueShares",
            "partnercentral:ListMarketplaceRevenueShareAllocations"
        ],
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "partnercentral:Catalog": [
                    "Sandbox"
                ]
            }
        }
    },
    {
        "Sid": "RevenueAttributionManagement",
        "Effect": "Allow",
        "Action": [
            "partnercentral:GetRevenueAttribution",
            "partnercentral:UpdateRevenueAttribution",
            "partnercentral:GetRevenueAttributionAllocation",
            "partnercentral:StartRevenueAttributionAllocationsTask",
            "partnercentral:GetRevenueAttributionAllocationsTask"
        ],
        "Resource": "arn:aws:partnercentral:*:*:catalog/Sandbox/revenue-
attribution/*",
        "Condition": {
            "StringEquals": {
                "partnercentral:Catalog": [
                    "Sandbox"
                ]
            }
        }
    }
}

```

```

    },
    {
      "Sid": "MarketplaceRevenueShareManagement",
      "Effect": "Allow",
      "Action": [
        "partnercentral:GetMarketplaceRevenueShare",
        "partnercentral:GetMarketplaceRevenueShareAllocation",
        "partnercentral:UpdateMarketplaceRevenueShareAllocation"
      ],
      "Resource": "arn:aws:partnercentral:*:*:catalog/Sandbox/marketplace-
revenue-share/*",
      "Condition": {
        "StringEquals": {
          "partnercentral:Catalog": [
            "Sandbox"
          ]
        }
      }
    },
    {
      "Sid": "TaggingAccess",
      "Effect": "Allow",
      "Action": [
        "partnercentral:TagResource",
        "partnercentral:UntagResource",
        "partnercentral:ListTagsForResource"
      ],
      "Resource": [
        "arn:aws:partnercentral:*:*:catalog/Sandbox/revenue-attribution/*",
        "arn:aws:partnercentral:*:*:catalog/Sandbox/marketplace-revenue-
share/*"
      ],
      "Condition": {
        "StringEquals": {
          "partnercentral:Catalog": [
            "Sandbox"
          ]
        }
      }
    },
    {
      "Sid": "OpportunityAccess",
      "Effect": "Allow",
      "Action": [

```

```

        "partnercentral:ListOpportunities",
        "partnercentral:GetOpportunity"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "partnercentral:Catalog": [
                "Sandbox"
            ]
        }
    }
},
{
    "Sid": "ListingAWSMarketplaceEntities",
    "Effect": "Allow",
    "Action": [
        "aws-marketplace:ListEntities"
    ],
    "Resource": "*"
},
{
    "Sid": "AWSMarketplaceEntityAccess",
    "Effect": "Allow",
    "Action": [
        "aws-marketplace:DescribeEntity"
    ],
    "Resource": [
        "arn:aws:aws-marketplace:*:*:AWSMarketplace*/Offer/*"
    ]
}
]
}

```

3. Gunakan kredensial peran IAM:

Gunakan kredensial (kunci rahasia dan kunci akses) peran IAM ini dalam solusi Anda untuk melakukan tindakan API.

Pengujian AWS tindakan

Selama fase pengujian, seringkali perlu untuk mensimulasikan AWS tindakan. Simulasi ini memungkinkan mitra untuk menguji secara menyeluruh aliran end-to-end lengkap integrasi mereka

dengan layanan.AWS Setiap permintaan menyertakan parameter katalog, yang menentukan lingkungan data.

Mensimulasikan pembuatan ID Atribusi Pendapatan

Untuk mensimulasikan pembuatan ID Atribusi Pendapatan, dalam muatan `CreateRevenueAttribution` tindakan, sertakan "Catalog": "Sandbox" dalam muatan.

Contoh:

```
{
  "Catalog": "Sandbox",
  "ClientToken": "String",
  "Name": "String",
  "Description": "String",
  "TenancyModel": "MULTI_TENANT | SINGLE_TENANT",
  "ProductIdentifier": "String",
  "Tags": [
    {
      "Key": "String",
      "Value": "String"
    }
  ]
}
```

Catatan pengujian tambahan

1. Untuk menguji tindakan lain, atur "Catalog": "Sandbox" di payload.
2. Untuk pindah ke produksi, ubah nilai katalog dari Sandbox keAWS.

Contoh kode untuk Partner Central menggunakan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan Partner Central dengan AWS perangkat pengembangan perangkat lunak (SDK).

Tindakan merupakan kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Sementara tindakan menunjukkan cara memanggil fungsi layanan individual, Anda dapat melihat tindakan dalam konteks dalam skenario terkait.

Skenario adalah contoh kode yang menunjukkan kepada Anda bagaimana menyelesaikan tugas tertentu dengan memanggil beberapa fungsi dalam layanan atau dikombinasikan dengan yang lain Layanan AWS.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Contoh kode

- [Contoh dasar untuk Partner Central menggunakan AWS SDK](#)
 - [Tindakan untuk Partner Central menggunakan AWS SDK](#)
 - [Gunakan AssignOpportunitydengan AWS SDK](#)
 - [Gunakan AssociateOpportunitydengan AWS SDK](#)
 - [Gunakan CreateOpportunitydengan AWS SDK](#)
 - [Gunakan DisassociateOpportunitydengan AWS SDK](#)
 - [Gunakan GetAwsOpportunitySummarydengan AWS SDK](#)
 - [Gunakan GetEngagementInvitationdengan AWS SDK](#)
 - [Gunakan GetOpportunitydengan AWS SDK](#)
 - [Gunakan ListEngagementInvitationsdengan AWS SDK](#)
 - [Gunakan ListOpportunitiesdengan AWS SDK](#)
 - [Gunakan ListSolutionsdengan AWS SDK](#)
 - [Gunakan RejectEngagementInvitationdengan AWS SDK](#)
 - [Gunakan StartEngagementByAcceptingInvitationTaskdengan AWS SDK](#)
 - [Gunakan StartEngagementFromOpportunityTaskdengan AWS SDK](#)
 - [Gunakan UpdateOpportunitydengan AWS SDK](#)

- [Skenario untuk Partner Central menggunakan AWS SDK](#)
 - [Perbarui entitas terkait dari suatu peluang](#)

Contoh dasar untuk Partner Central menggunakan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan dasar-dasar AWS Partner Central dengan AWS SDK.

Contoh

- [Tindakan untuk Partner Central menggunakan AWS SDK](#)
 - [Gunakan AssignOpportunitydengan AWS SDK](#)
 - [Gunakan AssociateOpportunitydengan AWS SDK](#)
 - [Gunakan CreateOpportunitydengan AWS SDK](#)
 - [Gunakan DisassociateOpportunitydengan AWS SDK](#)
 - [Gunakan GetAwsOpportunitySummarydengan AWS SDK](#)
 - [Gunakan GetEngagementInvitationdengan AWS SDK](#)
 - [Gunakan GetOpportunitydengan AWS SDK](#)
 - [Gunakan ListEngagementInvitationsdengan AWS SDK](#)
 - [Gunakan ListOpportunitiesdengan AWS SDK](#)
 - [Gunakan ListSolutionsdengan AWS SDK](#)
 - [Gunakan RejectEngagementInvitationdengan AWS SDK](#)
 - [Gunakan StartEngagementByAcceptingInvitationTaskdengan AWS SDK](#)
 - [Gunakan StartEngagementFromOpportunityTaskdengan AWS SDK](#)
 - [Gunakan UpdateOpportunitydengan AWS SDK](#)

Tindakan untuk Partner Central menggunakan AWS SDK

Contoh kode berikut menunjukkan cara melakukan tindakan Sentral Mitra individual dengan AWS SDK. Setiap contoh menyertakan tautan ke GitHub, di mana Anda dapat menemukan instruksi untuk mengatur dan menjalankan kode.

Kutipan ini memanggil API Pusat Mitra dan merupakan kutipan kode dari program yang lebih besar yang harus dijalankan dalam konteks. Anda dapat melihat tindakan dalam konteks di [Skenario untuk Partner Central menggunakan AWS SDK](#).

Contoh berikut hanya mencakup tindakan yang paling umum digunakan. Untuk daftar lengkapnya, lihat [Referensi API Pusat AWS Mitra](#).

Contoh

- [Gunakan AssignOpportunitydengan AWS SDK](#)
- [Gunakan AssociateOpportunitydengan AWS SDK](#)
- [Gunakan CreateOpportunitydengan AWS SDK](#)
- [Gunakan DisassociateOpportunitydengan AWS SDK](#)
- [Gunakan GetAwsOpportunitySummarydengan AWS SDK](#)
- [Gunakan GetEngagementInvitationdengan AWS SDK](#)
- [Gunakan GetOpportunitydengan AWS SDK](#)
- [Gunakan ListEngagementInvitationsdengan AWS SDK](#)
- [Gunakan ListOpportunitiesdengan AWS SDK](#)
- [Gunakan ListSolutionsdengan AWS SDK](#)
- [Gunakan RejectEngagementInvitationdengan AWS SDK](#)
- [Gunakan StartEngagementByAcceptingInvitationTaskdengan AWS SDK](#)
- [Gunakan StartEngagementFromOpportunityTaskdengan AWS SDK](#)
- [Gunakan UpdateOpportunitydengan AWS SDK](#)

Gunakan **AssignOpportunity**dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakanAssignOpportunity.

Java

SDK untuk Java 2.x

Tetapkan kembali Peluang yang ada ke pengguna lain.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
```



```
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
    software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
    software.amazon.awssdk.services.partnercentralselling.model.AssignOpportunityRequest;
import
    software.amazon.awssdk.services.partnercentralselling.model.AssignOpportunityResponse;
import
    software.amazon.awssdk.services.partnercentralselling.model.AssigneeContact;

/*
Purpose
PC-API-07 Assigning a new owner
*/

public class AssignOpportunity {

    static PartnerCentralSellingClient client =
        PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {

        String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

        String assigneeFirstName = "John";

        String assigneeLastName = "Doe";

        String assigneeEmail = "test@test.com";

        String businessTitle = "PartnerAccountManager";

        AssignOpportunityResponse response = getResponse(opportunityId,
            assigneeFirstName, assigneeLastName, assigneeEmail, businessTitle);

        ReferenceCodesUtils.formatOutput(response);
    }
}
```

```

static AssignOpportunityResponse getResponse(String opportunityId, String
assigneeFirstName, String assigneeLastName, String assigneeEmail, String
businessTitle) {

    AssignOpportunityRequest assignOpportunityRequest =
AssignOpportunityRequest.builder()
        .catalog(Constants.CATALOG_TO_USE)
        .identifier(opportunityId)
        .assignee(AssigneeContact.builder()
            .firstName(assigneeFirstName)
            .lastName(assigneeLastName)
            .email(assigneeEmail)
            .businessTitle(businessTitle)
            .build())
        .build();

    AssignOpportunityResponse response =
client.assignOpportunity(assignOpportunityRequest);

    return response;
}
}

```

- Untuk detail API, lihat [AssignOpportunity](#) di Referensi AWS SDK for Java 2.x API.

Python

SDK untuk Python (Boto3)

Tetapkan kembali Peluang yang ada ke pengguna lain.

```

#!/usr/bin/env python

"""
Purpose
PC-API-07 Assigning a new owner
"""

import logging
import boto3
import utils.helpers as helper
from botocore.client import ClientError

```

```
from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
    service_name=serviceName,
    region_name='us-east-1'
)

def assign_opportunity(identifier):
    assign_opportunity_request = {
        "Catalog": CATALOG_TO_USE,
        "Identifier": identifier,
        "Assignee": {
            "BusinessTitle": "OpportunityOwner",
            "Email": "test@test.com",
            "FirstName": "John",
            "LastName": "Doe"
        }
    }
    try:
        # Perform an API call
        response =
partner_central_client.assign_opportunity(**assign_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def usage_demo():
    identifier = "04236468"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Assigning a new owner to an opportunity.")
    print("-" * 88)

    helper.pretty_print_datetime(assign_opportunity(identifier))

if __name__ == "__main__":
    usage_demo()
```

- Untuk detail API, lihat [AssignOpportunity](#) di AWS SDK for Python (Boto3) Referensi API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **AssociateOpportunity** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan `AssociateOpportunity`.

Contoh tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Anda dapat melihat tindakan ini dalam konteks dalam contoh kode berikut:

- [Perbarui entitas terkait dari suatu peluang](#)

Java

SDK untuk Java 2.x

Buat hubungan formal antara Peluang dan berbagai entitas terkait.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
    software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
    software.amazon.awssdk.services.partnercentralselling.model.AssociateOpportunityRequest;
import
    software.amazon.awssdk.services.partnercentralselling.model.AssociateOpportunityResponse;

/*
Purpose
```

```
PC-API -11 Associating a product
PC-API -12 Associating a solution
PC-API -13 Associating an offer
entity_type = Solutions | AWSProducts | AWSMarketplaceOffers
*/

public class AssociateOpportunity {

    static PartnerCentralSellingClient client =
        PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {

        String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

        String entityType = "Solutions";

        String entityIdIdentifier = "S-00000000";

        AssociateOpportunityResponse response = getResponse(opportunityId,
            entityType, entityIdIdentifier );

        ReferenceCodesUtils.formatOutput(response);
    }

    static AssociateOpportunityResponse getResponse(String opportunityId, String
        entityType, String entityIdIdentifier) {

        AssociateOpportunityRequest associateOpportunityRequest =
            AssociateOpportunityRequest.builder()
                .catalog(Constants.CATALOG_TO_USE)
                .opportunityIdentifier(opportunityId)
                .relatedEntityType(entityType)
                .relatedEntityIdentifier(entityIdIdentifier)
                .build();

        AssociateOpportunityResponse response =
            client.associateOpportunity(associateOpportunityRequest);

        return response;
    }
}
```

```
}
}
```

- Untuk detail API, lihat [AssociateOpportunity](#) di Referensi AWS SDK for Java 2.x API.

Python

SDK untuk Python (Boto3)

Buat hubungan formal antara Peluang dan berbagai entitas terkait.

```
#!/usr/bin/env python

"""
Purpose
PC-API -11 Associating a product
PC-API -12 Associating a solution
PC-API -13 Associating an offer
"""

import logging
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
    service_name=serviceName,
    region_name='us-east-1'
)

def associate_opportunity(entity_type, entity_identifier, opportunityIdentifier):
    associate_opportunity_request = {
        "Catalog": CATALOG_TO_USE,
        "OpportunityIdentifier" : opportunityIdentifier,
        "RelatedEntityType" : entity_type,
        "RelatedEntityIdentifier" : entity_identifier
    }
    try:
```

```

        # Perform an API call
        response =
partner_central_client.associate_opportunity(**associate_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def usage_demo():
    #entity_type = Solutions | AWSProducts | AWSMarketplaceOffers
    entity_type = "Solutions"
    entity_identifier = "S-0059717"
    opportunityIdentifier = "05465588"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Associate Opportunity.")
    print("-" * 88)

    helper.pretty_print_datetime(associate_opportunity(entity_type,
    entity_identifier, opportunityIdentifier))

if __name__ == "__main__":
    usage_demo()

```

- Untuk detail API, lihat [AssociateOpportunity](#) di AWS SDK for Python (Boto3) Referensi API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **CreateOpportunity** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan `CreateOpportunity`.

.NET

SDK untuk .NET

Ciptakan peluang.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// PDX-License-Identifier: Apache-2.0

using System;
using Newtonsoft.Json;
using Amazon;
using Amazon.Runtime;
using Amazon.PartnerCentralSelling;
using Amazon.PartnerCentralSelling.Model;

namespace AWSExample
{
    class Program
    {
        static readonly string catalogToUse = "AWS";
        static async Task Main(string[] args)
        {
            // Initialize credentials from .aws/credentials file
            var credentials = new
Amazon.Runtime.CredentialManagement.SharedCredentialsFile();
            if (credentials.TryGetProfile("default", out var profile))
            {
                AWSCredentials awsCredentials =
profile.GetAWSCredentials(credentials);

                var client = new
AmazonPartnerCentralSellingClient(awsCredentials);

                var request = new CreateOpportunityRequest
                {
                    Catalog = catalogToUse,
                    Origin = "Partner Referral",
                    Customer = new Customer
                    {
                        Account = new Account
                        {
                            Address = new Address
                            {
                                CountryCode = "US",
                                PostalCode = "99502",
                                StateOrRegion = "Alaska"
                            },
                            CompanyName = "TestCompanyName",
```



```

        Duns = "123456789",
        WebsiteUrl = "www.test.io",
        Industry = "Automotive"
    },
    Contacts = new List<Contact>
    {
        new Contact
        {
            Email = "test@test.io",
            FirstName = "John ",
            LastName = "Doe",
            Phone = "+14444444444",
            BusinessTitle = "test title"
        }
    },
    Lifecycle = new Lifecycle
    {
        ReviewStatus = "Submitted",
        TargetCloseDate = "2024-12-30"
    },
    Marketing = new Marketing
    {
        Source = "None"
    },
    OpportunityType = "Net New Business",
    PrimaryNeedsFromAws = new List<string> { "Co-Sell -
Architectural Validation" },
    Project = new Project
    {
        Title = "Moin Test UUID",
        CustomerBusinessProblem = "Sandbox is not working as
expected",

        CustomerUseCase = "AI Machine Learning and Analytics",
        DeliveryModels = new List<string> { "SaaS or PaaS" },
        ExpectedCustomerSpend = new List<ExpectedCustomerSpend>
        {
            new ExpectedCustomerSpend
            {
                Amount = "2000.0",
                CurrencyCode = "USD",
                Frequency = "Monthly",
                TargetCompany = "Ibexlabs"
            }
        }
    }

```

```
        },  
        SalesActivities = new List<string> { "Initialized  
discussions with customer" }  
    }  
};  
  
try  
{  
    var response = await client.CreateOpportunityAsync(request);  
    Console.WriteLine(response.HttpStatusCode);  
    string formattedJson = JsonConvert.SerializeObject(response,  
Formatting.Indented);  
    Console.WriteLine(formattedJson);  
}  
catch (ValidationException ex)  
{  
    Console.WriteLine("Validation error: " + ex.Message);  
}  
catch (AmazonPartnerCentralSellingException e)  
{  
    Console.WriteLine("Failed:");  
    Console.WriteLine(e.RequestId);  
    Console.WriteLine(e.ErrorCode);  
    Console.WriteLine(e.Message);  
}  
}  
else  
{  
    Console.WriteLine("Profile not found.");  
}  
}  
}
```

- Untuk detail API, lihat [CreateOpportunity](#) di Referensi AWS SDK untuk .NET API.

Java

SDK untuk Java 2.x

Ciptakan peluang.

```
package org.example;
```

```
import java.time.Instant;
import java.util.ArrayList;
import java.util.List;

import static org.example.utils.Constants.*;

import org.example.entity.Root;
import org.example.utils.ReferenceCodesUtils;
import org.example.utils.StringSerializer;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
    software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import software.amazon.awssdk.services.partnercentralselling.model.Account;
import software.amazon.awssdk.services.partnercentralselling.model.Address;
import software.amazon.awssdk.services.partnercentralselling.model.Contact;
import
    software.amazon.awssdk.services.partnercentralselling.model.CreateOpportunityRequest;
import
    software.amazon.awssdk.services.partnercentralselling.model.CreateOpportunityResponse;
import software.amazon.awssdk.services.partnercentralselling.model.Customer;
import
    software.amazon.awssdk.services.partnercentralselling.model.ExpectedCustomerSpend;
import software.amazon.awssdk.services.partnercentralselling.model.LifeCycle;
import software.amazon.awssdk.services.partnercentralselling.model.Marketing;
import software.amazon.awssdk.services.partnercentralselling.model.MonetaryValue;
import
    software.amazon.awssdk.services.partnercentralselling.model.NextStepsHistory;
import software.amazon.awssdk.services.partnercentralselling.model.Project;
import
    software.amazon.awssdk.services.partnercentralselling.model.SoftwareRevenue;

import com.google.gson.Gson;
import com.google.gson.GsonBuilder;
import com.google.gson.ToNumberPolicy;

public class CreateOpportunity {

    static final Gson GSON = new GsonBuilder()
        .setObjectToNumberStrategy(ToNumberPolicy.LAZILY_PARSED_NUMBER)
        .registerTypeAdapter(String.class, new StringSerializer())
```

```
.create();

static PartnerCentralSellingClient client =
PartnerCentralSellingClient.builder()
    .region(Region.US_EAST_1)
    .credentialsProvider(DefaultCredentialsProvider.create())
    .httpClient(ApacheHttpClient.builder().build())
    .build();

public static void main(String[] args) {

    String inputFile = "CreateOpportunity2.json";

    if (args.length > 0)
        inputFile = args[0];

    CreateOpportunityResponse response = createOpportunity(inputFile);

    client.close();
}

static CreateOpportunityResponse createOpportunity(String inputFile) {

    String inputString = ReferenceCodesUtils.readInputFileToString(inputFile);

    Root root = GSON.fromJson(inputString, Root.class);

    List<NextStepsHistory> nextStepsHistories = new ArrayList<NextStepsHistory>();
    if ( root.lifeCycle != null && root.lifeCycle.nextStepsHistories != null) {
        for (org.example.entity.NextStepsHistory nextStepsHistoryJson :
root.lifeCycle.nextStepsHistories) {
            NextStepsHistory nextStepsHistory = NextStepsHistory.builder()
                .time(Instant.parse(nextStepsHistoryJson.time))
                .value(nextStepsHistoryJson.value)
                .build();
            nextStepsHistories.add(nextStepsHistory);
        }
    }

    LifeCycle lifeCycle = null;
    if ( root.lifeCycle != null ) {
        lifeCycle = LifeCycle.builder()
            .closedLostReason(root.lifeCycle.closedLostReason)
            .nextSteps(root.lifeCycle.nextSteps)
```

```
.nextStepsHistory(nextStepsHistories)
.reviewComments(root.lifeCycle.reviewComments)
.reviewStatus(root.lifeCycle.reviewStatus)
.reviewStatusReason(root.lifeCycle.reviewStatusReason)
.stage(root.lifeCycle.stage)
.targetCloseDate(root.lifeCycle.targetCloseDate)
.build();
}

Marketing marketing = null;
if ( root.marketing != null ) {
    marketing = Marketing.builder()
        .awsFundingUsed(root.marketing.awsFundingUsed)
        .campaignName(root.marketing.campaignName)
        .channels(root.marketing.channels)
        .source(root.marketing.source)
        .useCases(root.marketing.useCases)
        .build();
}

Address address = null;
if ( root.customer != null && root.customer.account != null &&
root.customer.account.address != null ) {
    address = Address.builder()
        .city(root.customer.account.address.city)
        .postalCode(root.customer.account.address.postalCode)
        .stateOrRegion(root.customer.account.address.stateOrRegion)
        .countryCode(root.customer.account.address.countryCode)
        .streetAddress(root.customer.account.address.streetAddress)
        .build();
}

Account account = null;
if ( root.customer != null && root.customer.account!= null) {
    account = Account.builder()
        .address(address)
        .awsAccountId(root.customer.account.awsAccountId)
        .duns(root.customer.account.duns)
        .industry(root.customer.account.industry)
        .otherIndustry(root.customer.account.otherIndustry)
        .companyName(root.customer.account.companyName)
        .websiteUrl(root.customer.account.websiteUrl)
        .build();
}
```

```
}

List<Contact> contacts = new ArrayList<Contact>();
if ( root.customer != null && root.customer.contacts != null) {
    for (org.example.entity.Contact jsonContact : root.customer.contacts) {
        Contact contact = Contact.builder()
            .email(jsonContact.email)
            .firstName(jsonContact.firstName)
            .lastName(jsonContact.lastName)
            .phone(jsonContact.phone)
            .businessTitle(jsonContact.businessTitle)
            .build();
        contacts.add(contact);
    }
}

Customer customer = Customer.builder()
    .account(account)
    .contacts(contacts)
    .build();

Contact oportunityTeamContact = null;
if (root.opportunityTeam != null && root.opportunityTeam.get(0) != null ) {
    oportunityTeamContact = Contact.builder()
        .firstName(root.opportunityTeam.get(0).firstName)
        .lastName(root.opportunityTeam.get(0).lastName)
        .email(root.opportunityTeam.get(0).email)
        .phone(root.opportunityTeam.get(0).phone)
        .businessTitle(root.opportunityTeam.get(0).businessTitle)
        .build();
}

List<ExpectedCustomerSpend> expectedCustomerSpend = new
ArrayList<ExpectedCustomerSpend>();
if ( root.project != null && root.project.expectedCustomerSpend != null) {
    for (org.example.entity.ExpectedCustomerSpend expectedCustomerSpendJson :
root.project.expectedCustomerSpend) {
        ExpectedCustomerSpend expectedCustomerSpend = null;
        expectedCustomerSpend = ExpectedCustomerSpend.builder()
            .amount(expectedCustomerSpendJson.amount)
            .currencyCode(expectedCustomerSpendJson.currencyCode)
            .frequency(expectedCustomerSpendJson.frequency)
            .targetCompany(expectedCustomerSpendJson.targetCompany)
            .build();
    }
}
```

```

        expectedCustomerSpend.add(expectedCustomerSpend);
    }

    }

    Project project = null;
    if ( root.project != null) {
        project = Project.builder()
            .title(root.project.title)
            .customerBusinessProblem(root.project.customerBusinessProblem)
            .customerUseCase(root.project.customerUseCase)
            .deliveryModels(root.project.deliveryModels)
            .expectedCustomerSpend(expectedCustomerSpend)
            .salesActivities(root.project.salesActivities)
            .competitorName(root.project.competitorName)
            .otherSolutionDescription(root.project.otherSolutionDescription)
            .build();
    }

    SoftwareRevenue softwareRevenue = null;
    if ( root.softwareRevenue != null) {
        MonetaryValue monetaryValue = null;
        if ( root.softwareRevenue.value != null) {
            monetaryValue = MonetaryValue.builder()
                .amount(root.softwareRevenue.value.amount)
                .currencyCode(root.softwareRevenue.value.currencyCode)
                .build();
        }
        softwareRevenue = SoftwareRevenue.builder()
            .deliveryModel(root.softwareRevenue.deliveryModel)
            .effectiveDate(root.softwareRevenue.effectiveDate)
            .expirationDate(root.softwareRevenue.expirationDate)
            .value(monetaryValue)
            .build();
    }

    // Building the Actual CreateOpportunity Request
    CreateOpportunityRequest createOpportunityRequest =
    CreateOpportunityRequest.builder()
        .catalog(CATALOG_TO_USE)
        .clientToken(root.clientToken)
        .primaryNeedsFromAwsWithStrings(root.primaryNeedsFromAws)
        .opportunityType(root.opportunityType)
        .lifeCycle(lifeCycle)
        .marketing(marketing)

```

```

        .nationalSecurity(root.nationalSecurity)
        .origin(root.origin)
        .customer(customer)
        .project(project)
        .partnerOpportunityIdentifier(root.partnerOpportunityIdentifier)
        .opportunityTeam(opportunityTeamContact)
        .softwareRevenue(softwareRevenue)
        .build();

    CreateOpportunityResponse response =
client.createOpportunity(createOpportunityRequest);
    System.out.println("Successfully created: " + response);

    return response;
}
}

```

- Untuk detail API, lihat [CreateOpportunity](#) di Referensi AWS SDK for Java 2.x API.

Python

SDK untuk Python (Boto3)

Ciptakan peluang.

```

#!/usr/bin/env python
import boto3
import logging
import sys
import os
sys.path.append(os.path.dirname(os.path.dirname(os.path.abspath(__file__))))
import utils.helpers as helper
import utils.stringify_details as sd
from botocore.client import ClientError
from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

def create_opportunity(partner_central_client):

```



```

    create_opportunity_request = helper.remove_nulls(sd.stringify_json("src/
create_opportunity/createOpportunity.json"))
    try:
        # Perform an API call
        response =
partner_central_client.create_opportunity(**create_opportunity_request)

        helper.pretty_print_datetime(response)

        # Retrieve the opportunity details
        get_response = partner_central_client.get_opportunity(
            Identifier=response["Id"],
            Catalog=CATALOG_TO_USE
        )
        helper.pretty_print_datetime(get_response)
        return response
    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Create Opportunity.")
    print("-" * 88)

    partner_central_client = boto3.client(
        service_name=serviceName,
        region_name='us-east-1'
    )

    create_opportunity(partner_central_client)

if __name__ == "__main__":
    usage_demo()

```

- Untuk detail API, lihat [CreateOpportunity](#) di AWS SDK for Python (Boto3) Referensi API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **DisassociateOpportunity** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan `DisassociateOpportunity`.

Contoh tindakan adalah kutipan kode dari program yang lebih besar dan harus dijalankan dalam konteks. Anda dapat melihat tindakan ini dalam konteks dalam contoh kode berikut:

- [Perbarui entitas terkait dari suatu peluang](#)

Java

SDK untuk Java 2.x

Hapus asosiasi yang ada antara Peluang dan entitas terkait.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
    software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
    software.amazon.awssdk.services.partnercentralselling.model.DisassociateOpportunityRequest;
import
    software.amazon.awssdk.services.partnercentralselling.model.DisassociateOpportunityResponse;

/*
Purpose
PC-API -14 Removing a Solution
PC-API -15 Removing an offer
PC-API -16 Removing a product
entity_type = Solutions | AWSProducts | AWSMarketplaceOffers
*/

public class DisassociateOpportunity {
```

```

static PartnerCentralSellingClient client =
PartnerCentralSellingClient.builder()
    .region(Region.US_EAST_1)
    .credentialsProvider(DefaultCredentialsProvider.create())
    .httpClient(ApacheHttpClient.builder().build())
    .build();

public static void main(String[] args) {

    String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

    String entityType = "Solutions";

    String entityIdIdentifier = "S-00000000";

    DisassociateOpportunityResponse response = getResponse(opportunityId,
entityType, entityIdIdentifier );

    ReferenceCodesUtils.formatOutput(response);
}

static DisassociateOpportunityResponse getResponse(String opportunityId, String
entityType, String entityIdIdentifier) {

    DisassociateOpportunityRequest disassociateOpportunityRequest =
DisassociateOpportunityRequest.builder()
    .catalog(Constants.CATALOG_TO_USE)
    .opportunityIdentifier(opportunityId)
    .relatedEntityType(entityType)
    .relatedEntityIdentifier(entityIdIdentifier)
    .build();

    DisassociateOpportunityResponse response =
client.disassociateOpportunity(disassociateOpportunityRequest);

    return response;
}
}

```

- Untuk detail API, lihat [DisassociateOpportunity](#) di Referensi AWS SDK for Java 2.x API.

Python

SDK untuk Python (Boto3)

Hapus asosiasi yang ada antara Peluang dan entitas terkait.

```
#!/usr/bin/env python

"""
Purpose
PC-API -14 Removing a Solution
PC-API -15 Removing an offer
PC-API -16 Removing a product
"""

import logging
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
    service_name=serviceName,
    region_name='us-east-1'
)

def disassociate_opportunity(entity_type, entity_identifier,
    opportunityIdentifier):
    disassociate_opportunity_request = {
        "Catalog": CATALOG_TO_USE,
        "OpportunityIdentifier" : opportunityIdentifier,
        "RelatedEntityType" : entity_type,
        "RelatedEntityIdentifier" : entity_identifier
    }
    try:
        # Perform an API call
        response =
partner_central_client.disassociate_opportunity(**disassociate_opportunity_request)
        return response

    except ClientError as err:
```

```

        # Catch all client exceptions
        print(err.response)

def usage_demo():
    #entity_type = Solutions | AWSProducts | AWSMarketplaceOffers
    entity_type = "Solutions"
    entity_identifier = "S-0049999"
    opportunityIdentifier = "04397574"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Get updated Opportunity.")
    print("-" * 88)

    helper.pretty_print_datetime(disassociate_opportunity(entity_type,
    entity_identifier, opportunityIdentifier))

if __name__ == "__main__":
    usage_demo()

```

- Untuk detail API, lihat [DisassociateOpportunity](#) di AWS SDK for Python (Boto3) Referensi API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **GetAwsOpportunitySummary** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan `GetAwsOpportunitySummary`.

Java

SDK untuk Java 2.x

Mengambil ringkasan dari AWS Peluang.

```

package org.example;

import static org.example.utils.Constants.*;

```

```
import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
    software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
    software.amazon.awssdk.services.partnercentralselling.model.GetAwsOpportunitySummaryRequest;
import
    software.amazon.awssdk.services.partnercentralselling.model.GetAwsOpportunitySummaryResponse;

/*
 * Purpose
 * PC-API-25 Retrieves a summary of an AWS Opportunity.
 */

public class GetAwsOpportunitySummary {

    static PartnerCentralSellingClient client =
        PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {

        String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

        GetAwsOpportunitySummaryResponse response = getResponse(opportunityId);

        ReferenceCodesUtils.formatOutput(response);
    }

    public static GetAwsOpportunitySummaryResponse getResponse(String opportunityId)
    {

        GetAwsOpportunitySummaryRequest getOpportunityRequest =
            GetAwsOpportunitySummaryRequest.builder()
                .catalog(Constants.CATALOG_TO_USE)
                .relatedOpportunityIdentifier(opportunityId)
```

```

        .build();

        GetAwsOpportunitySummaryResponse response =
        client.getAwsOpportunitySummary(getOpportunityRequest);

        return response;
    }
}

```

- Untuk detail API, lihat [GetAwsOpportunitySummary](#) di Referensi AWS SDK for Java 2.x API.

Python

SDK untuk Python (Boto3)

Mengambil ringkasan dari AWS Peluang.

```

#!/usr/bin/env python

"""
Purpose
PC-API-25 Retrieves a summary of an AWS Opportunity.
    Lifecycle.ReviewStatus=Approved
"""

import logging
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
    service_name=serviceName,
    region_name='us-east-1'
)

def get_opportunity(identifier):
    get_opportunity_request = {
        "Catalog": CATALOG_TO_USE,

```

```

        "RelatedOpportunityIdentifier": identifier
    }
    try:
        # Perform an API call
        response =
partner_central_client.get_aws_opportunity_summary(**get_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def usage_demo():
    identifier = "05465588"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Get AWS Opportunity summary.")
    print("-" * 88)

    helper.pretty_print_datetime(get_opportunity(identifier))

if __name__ == "__main__":
    usage_demo()

```

- Untuk detail API, lihat [GetAwsOpportunitySummary](#) di AWS SDK for Python (Boto3) Referensi API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **GetEngagementInvitation** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan `GetEngagementInvitation`.

Java

SDK untuk Java 2.x

Mengambil detail undangan pertunangan yang dibagikan AWS dengan pasangan.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
    software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
    software.amazon.awssdk.services.partnercentralselling.model.GetEngagementInvitationRequest;
import
    software.amazon.awssdk.services.partnercentralselling.model.GetEngagementInvitationResponse;

/*
 * Purpose
 * PC-API-22 Get engagement invitation opportunity
 */

public class GetEngagementInvitation {

    static PartnerCentralSellingClient client =
        PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {

        String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

        GetEngagementInvitationResponse response = getResponse(opportunityId);

        ReferenceCodesUtils.formatOutput(response);
    }
}
```

```

    }

    static GetEngagementInvitationResponse getResponse(String opportunityId) {

        GetEngagementInvitationRequest getOpportunityRequest =
        GetEngagementInvitationRequest.builder()
            .catalog(Constants.CATALOG_T0_USE)
            .identifier(opportunityId)
            .build();

        GetEngagementInvitationResponse response =
        client.getEngagementInvitation(getOpportunityRequest);

        return response;
    }
}

```

- Untuk detail API, lihat [GetEngagementInvitation](#) di Referensi AWS SDK for Java 2.x API.

Python

SDK untuk Python (Boto3)

Mengambil detail undangan pertunangan yang dibagikan AWS dengan pasangan.

```

#!/usr/bin/env python

"""
Purpose
PC-API-22  GetOpportunityEngagementInvitation - Retrieves details of a specific
engagement invitation.
This operation allows partners to view the invitation and its associated
information,
such as the customer, project, and lifecycle details.
"""

import json
import logging
import boto3
import utils.helpers as helper

from utils.constants import CATALOG_T0_USE

```

```

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
    service_name=serviceName,
    region_name='us-east-1'
)

def get_opportunity_engagement_invitation(identifier):
    get_opportunity_engagement_invitation_request = {
        "Catalog": CATALOG_TO_USE,
        "Identifier": identifier
    }
    try:
        # Perform an API call
        response =
partner_central_client.get_engagement_invitation(**get_opportunity_engagement_invitation
        return response

    except Exception as err:
        # Catch all client exceptions
        print(json.dumps(err.response))

def usage_demo():
    identifier = "arn:aws:partnercentral-selling:us-east-1:aws:catalog/Sandbox/
engagement-invitation/engi-00000000IS0Qga"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Given the ARN identifier, retrieve details of Opportunity Engagement
Invitation.")
    print("-" * 88)

    helper.pretty_print_datetime(get_opportunity_engagement_invitation(identifier))

if __name__ == "__main__":
    usage_demo()

```

- Untuk detail API, lihat [GetEngagementInvitation](#) di AWS SDK for Python (Boto3) Referensi API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **GetOpportunity** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan `GetOpportunity`.

.NET

SDK untuk .NET

Dapatkan kesempatan.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// PDX-License-Identifier: Apache-2.0

using System;
using Newtonsoft.Json;
using Amazon;
using Amazon.Runtime;
using Amazon.PartnerCentralSelling;
using Amazon.PartnerCentralSelling.Model;

namespace AWSExample
{
    class Program
    {
        static readonly string catalogToUse = "AWS";
        static readonly string identifier = "01111111";
        static async Task Main(string[] args)
        {
            // Initialize credentials from .aws/credentials file
            var credentials = new
            Amazon.Runtime.CredentialManagement.SharedCredentialsFile();
            if (credentials.TryGetProfile("default", out var profile))
            {
                AWSCredentials awsCredentials =
                profile.GetAWSCredentials(credentials);

                var client = new
                AmazonPartnerCentralSellingClient(awsCredentials);
```

```
var request = new GetOpportunityRequest
{
    Catalog = catalogToUse,
    Identifier = identifier
};

try {
    var response = await client.GetOpportunityAsync(request);
    Console.WriteLine(response.HttpStatusCode);
    string formattedJson = JsonConvert.SerializeObject(response,
Formatting.Indented);
    Console.WriteLine(formattedJson);
} catch (ValidationException ex) {
    Console.WriteLine("Validation error: " + ex.Message);
} catch (AmazonPartnerCentralSellingException e) {
    Console.WriteLine("Failed:");
    Console.WriteLine(e.RequestId);
    Console.WriteLine(e.ErrorCode);
    Console.WriteLine(e.Message);
}
}
else
{
    Console.WriteLine("Profile not found.");
}
}
}
```

- Untuk detail API, lihat [GetOpportunity](#) di Referensi AWS SDK untuk .NET API.

Go

SDK untuk Go V2

Dapatkan kesempatan.

```
package main

import (
    "context"
```

```
"encoding/json"
"fmt"
"log"

"github.com/aws/aws-sdk-go-v2/aws"
"github.com/aws/aws-sdk-go-v2/config"
"github.com/aws/aws-sdk-go-v2/service/partnercentralselling"
)

func main() {
    config, err := config.LoadDefaultConfig(context.TODO())

    if err != nil {
        log.Fatal(err)
    }

    config.Region = "us-east-1"

    client := partnercentralselling.NewFromConfig(config)

    output, err := client.GetOpportunity(context.TODO(),
        &partnercentralselling.GetOpportunityInput{
            Identifier: aws.String("01111111"),
            Catalog:   aws.String("AWS"),
        })

    if err != nil {
        log.Fatal(err)
    }
    log.Println("printing opportuniy...\n")

    jsonOutput, err := json.MarshalIndent(output, "", "    ")

    fmt.Println(string(jsonOutput))
}
```

- Untuk detail API, lihat [GetOpportunity](#) di Referensi AWS SDK untuk Go API.

Java

SDK untuk Java 2.x

Dapatkan kesempatan.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
    software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
    software.amazon.awssdk.services.partnercentralselling.model.GetOpportunityRequest;
import
    software.amazon.awssdk.services.partnercentralselling.model.GetOpportunityResponse;

/*
 * Purpose
 * PC-API-08 Get updated Opportunity
 */

public class GetOpportunity {

    static PartnerCentralSellingClient client =
        PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {

        String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

        GetOpportunityResponse response = getResponse(opportunityId);

        ReferenceCodesUtils.formatOutput(response);
    }
}
```

```

    }

    public static GetOpportunityResponse getResponse(String opportunityId) {

        GetOpportunityRequest getOpportunityRequest =
        GetOpportunityRequest.builder()
            .catalog(Constants.CATALOG_T0_USE)
            .identifier(opportunityId)
            .build();

        GetOpportunityResponse response =
        client.getOpportunity(getOpportunityRequest);

        return response;
    }
}

```

- Untuk detail API, lihat [GetOpportunity](#) di Referensi AWS SDK for Java 2.x API.

Python

SDK untuk Python (Boto3)

Dapatkan kesempatan.

```

#!/usr/bin/env python

"""
Purpose
PC-API -08 Get updated Opportunity given opportunity id
"""

import logging
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_T0_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(

```



```
        service_name=serviceName,
        region_name='us-east-1'
    )

def get_opportunity(identifier):
    get_opportunity_request = {
        "Catalog": CATALOG_TO_USE,
        "Identifier": identifier
    }
    try:
        # Perform an API call
        response =
partner_central_client.get_opportunity(**get_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def usage_demo():
    identifier = "05465588"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Get updated Opportunity.")
    print("-" * 88)

    helper.pretty_print_datetime(get_opportunity(identifier))

if __name__ == "__main__":
    usage_demo()
```

- Untuk detail API, lihat [GetOpportunity](#) di AWS SDK for Python (Boto3) Referensi API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **ListEngagementInvitations** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan `ListEngagementInvitations`.

Java

SDK untuk Java 2.x

Mengambil daftar undangan keterlibatan yang dikirim ke mitra.

```
package org.example;

import java.util.ArrayList;
import java.util.List;

import org.example.utils.ReferenceCodesUtils;
import static org.example.utils.Constants.*;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
    software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
    software.amazon.awssdk.services.partnercentralselling.model.ListEngagementInvitationsReq
import
    software.amazon.awssdk.services.partnercentralselling.model.ListEngagementInvitationsRes
import
    software.amazon.awssdk.services.partnercentralselling.model.ParticipantType;
import
    software.amazon.awssdk.services.partnercentralselling.model.EngagementInvitationSummary;

public class ListEngagementInvitations {

    static PartnerCentralSellingClient client =
        PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {

        List<EngagementInvitationSummary> opportunitySummaries = getResponse();
```

```

        ReferenceCodesUtils.formatOutput(opportunitySummaries);
    }

    static List<EngagementInvitationSummary> getResponse() {

        List<EngagementInvitationSummary> opportunitySummaries = new
        ArrayList<EngagementInvitationSummary>();

        ListEngagementInvitationsRequest listOpportunityRequest =
        ListEngagementInvitationsRequest.builder()
            .catalog(CATALOG_TO_USE)
            .participantType(ParticipantType.RECEIVER)
            .maxResults(5)
            .build();

        ListEngagementInvitationsResponse response =
        client.listEngagementInvitations(listOpportunityRequest);

        opportunitySummaries.addAll(response.engagementInvitationSummaries());

        client.close();

        return opportunitySummaries;
    }
}

```

- Untuk detail API, lihat [ListEngagementInvitations](#) di Referensi AWS SDK for Java 2.x API.

Python

SDK untuk Python (Boto3)

Mengambil daftar undangan keterlibatan yang dikirim ke mitra.

```

#!/usr/bin/env python

"""
Purpose
PC-API-21 ListEngagementInvitations - Retrieves a list of engagement invitations
based on specified criteria.
This operation allows partners to view all invitations to engagement.

```

```
"""
import json
import logging
import boto3
import utils.helpers as helper

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
    service_name=serviceName,
    region_name='us-east-1'
)

def list_engagement_invitations():
    list_engagement_invitations_request = {
        "Catalog": CATALOG_TO_USE,
        "MaxResults": 20
    }
    try:
        # Perform an API call
        response =
partner_central_client.list_engagement_invitations(**list_engagement_invitations_request)
        return response

    except Exception as err:
        # Catch all client exceptions
        print(json.dumps(err.response))

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Retrieve list of Engagement Invitations.")
    print("-" * 88)

    helper.pretty_print_datetime(list_engagement_invitations())

if __name__ == "__main__":
    usage_demo()
```

- Untuk detail API, lihat [ListEngagementInvitations](#) di AWS SDK for Python (Boto3) Referensi API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **ListOpportunities** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan `ListOpportunities`.

.NET

SDK untuk .NET

Daftar peluang.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// PDX-License-Identifier: Apache-2.0

using System;
using Newtonsoft.Json;
using Amazon;
using Amazon.Runtime;
using Amazon.PartnerCentralSelling;
using Amazon.PartnerCentralSelling.Model;

namespace AWSExample
{
    class Program
    {
        static readonly string catalogToUse = "Sandbox";
        static async Task Main(string[] args)
        {
            // Initialize credentials from .aws/credentials file
            var credentials = new
            Amazon.Runtime.CredentialManagement.SharedCredentialsFile();
            if (credentials.TryGetProfile("default", out var profile))
            {
                AWSCredentials awsCredentials =
                profile.GetAWSCredentials(credentials);
            }
        }
    }
}
```

```

        //var config = new AmazonPartnerCentralSellingConfig()
        //{
        //    ServiceURL = "https://partnercentral-selling.us-
east-1.api.aws",
        //};
        //var client = new
AmazonPartnerCentralSellingClient(awsCredentials, config);
        var client = new
AmazonPartnerCentralSellingClient(awsCredentials);
        var request = new ListOpportunitiesRequest
        {
            Catalog = catalogToUse,
            MaxResults = 2
        };

        try {
            var response = await client.ListOpportunitiesAsync(request);
            Console.WriteLine(response.HttpStatusCode);
            foreach (var opportunity in response.OpportunitySummaries)
            {
                Console.WriteLine("Opportunity id: " + opportunity.Id);
            }
            string formattedJson =
JsonConvert.SerializeObject(response.OpportunitySummaries, Formatting.Indented);
            Console.WriteLine(formattedJson);
        } catch (ValidationException ex) {
            Console.WriteLine("Validation error: " + ex.Message);
        } catch (AmazonPartnerCentralSellingException e) {
            Console.WriteLine("Failed:");
            Console.WriteLine(e.RequestId);
            Console.WriteLine(e.ErrorCode);
            Console.WriteLine(e.Message);
        }
    }
    else
    {
        Console.WriteLine("Profile not found.");
    }
}
}
}
}

```

- Untuk detail API, lihat [ListOpportunities](#) di Referensi AWS SDK untuk .NET API.

Go

SDK untuk Go V2

Daftar peluang.

```
package main

import (
    "context"
    "encoding/json"
    "fmt"
    "log"

    "github.com/aws/aws-sdk-go-v2/aws"
    "github.com/aws/aws-sdk-go-v2/config"
    "github.com/aws/aws-sdk-go-v2/service/partnercentral-selling"
)

func main() {
    config, err := config.LoadDefaultConfig(context.TODO())

    if err != nil {
        log.Fatal(err)
    }

    config.Region = "us-east-1"

    client := partnercentral-selling.NewFromConfig(config)

    output, err := client.ListOpportunities(context.TODO(),
        &partnercentral-selling.ListOpportunitiesInput{
            MaxResults: aws.Int32(2),
            Catalog:   aws.String("AWS"),
        })

    if err != nil {
        log.Fatal(err)
    }

    jsonOutput, err := json.MarshalIndent(output, "", "    ")
    fmt.Println(string(jsonOutput))
}
```

- Untuk detail API, lihat [ListOpportunities](#) di Referensi AWS SDK untuk Go API.

Java

SDK untuk Java 2.x

Daftar peluang.

```
package org.example;

import java.util.ArrayList;
import java.util.List;

import org.example.utils.ReferenceCodesUtils;
import static org.example.utils.Constants.*;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
    software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
    software.amazon.awssdk.services.partnercentralselling.model.ListOpportunitiesRequest;
import
    software.amazon.awssdk.services.partnercentralselling.model.ListOpportunitiesResponse;
import
    software.amazon.awssdk.services.partnercentralselling.model.OpportunitySummary;

/*
 * Purpose
 * PC-API-18 Getting list of Opportunities
 */

public class ListOpportunititesPaging {

    static PartnerCentralSellingClient client =
        PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();
```



```
public static void main(String[] args) {
    List<OpportunitySummary> opportunitySummaries = getResponse();
    ReferenceCodesUtils.formatOutput(opportunitySummaries);
}

private static List<OpportunitySummary> getResponse() {
    List<OpportunitySummary> opportunitySummaries = new
    ArrayList<OpportunitySummary>();

    ListOpportunitiesRequest listOpportunityRequest =
    ListOpportunitiesRequest.builder()
        .catalog(CATALOG_T0_USE)
        .maxResults(5)
        .build();

    ListOpportunitiesResponse response =
    client.listOpportunities(listOpportunityRequest);

    opportunitySummaries.addAll(response.opportunitySummaries());

    while (response.nextToken() != null && response.nextToken().length() > 0) {
        listOpportunityRequest = ListOpportunitiesRequest.builder()
            .catalog(CATALOG_T0_USE)
            .maxResults(5)
            .nextToken(response.nextToken())
            .build();
        response = client.listOpportunities(listOpportunityRequest);
        opportunitySummaries.addAll(response.opportunitySummaries());
    }

    client.close();

    return opportunitySummaries;
}
}
```

- Untuk detail API, lihat [ListOpportunities](#) di Referensi AWS SDK for Java 2.x API.

Python

SDK untuk Python (Boto3)

Daftar peluang.

```
#!/usr/bin/env python

"""
Purpose
PC-API -18 Getting list of Opportunities
"""

import json
import logging
import boto3
import utils.helpers as helper

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
    service_name=serviceName,
    region_name='us-east-1'
)

def get_list_of_opportunities():

    opportunity_list = []

    list_opportunities_request = {
        "Catalog": CATALOG_TO_USE,
        "MaxResults": 20
    }
    try:
        # Perform an API call
        response =
partner_central_client.list_opportunities(**list_opportunities_request)
        opportunity_list.extend(response["OpportunitySummaries"])

        while "NextToken" in response and response["NextToken"] is not None:
            list_opportunities_request["NextToken"] = response["NextToken"]
```

```

        response =
partner_central_client.list_opportunities(**list_opportunities_request)
        opportunity_list.extend(response["OpportunitySummaries"])

    return opportunity_list

except Exception as err:
    # Catch all client exceptions
    print(json.dumps(err.response))

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Getting list of Opportunities.")
    print("-" * 88)

    helper.pretty_print_datetime(get_list_of_opportunities())

if __name__ == "__main__":
    usage_demo()

```

- Untuk detail API, lihat [ListOpportunities](#) di AWS SDK for Python (Boto3) Referensi API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **ListSolutions** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan `ListSolutions`.

Java

SDK untuk Java 2.x

Mengambil daftar Solusi Mitra yang didaftarkan mitra di Partner Central.

```

package org.example;

import java.util.ArrayList;
import java.util.List;

```

```
import static org.example.utils.Constants.*;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
    software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
    software.amazon.awssdk.services.partnercentralselling.model.ListSolutionsRequest;
import
    software.amazon.awssdk.services.partnercentralselling.model.ListSolutionsResponse;
import software.amazon.awssdk.services.partnercentralselling.model.SolutionBase;

/*
 * Purpose
 * PC-API-10 Getting list of solutions
 */

public class ListSolutions {

    static PartnerCentralSellingClient client =
        PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {
        List<SolutionBase> solutionSummaries = getResponse();
        ReferenceCodesUtils.formatOutput(solutionSummaries);
    }

    static List<SolutionBase> getResponse() {
        List<SolutionBase> solutionSummaries = new ArrayList<SolutionBase>();

        ListSolutionsRequest listSolutionsRequest = ListSolutionsRequest.builder()
            .catalog(CATALOG_TO_USE)
            .maxResults(5)
            .build();

        ListSolutionsResponse response = client.listSolutions(listSolutionsRequest);
    }
}
```

```

        solutionSummaries.addAll(response.solutionSummaries());

        return solutionSummaries;
    }
}

```

- Untuk detail API, lihat [ListSolutions](#) di Referensi AWS SDK for Java 2.x API.

Python

SDK untuk Python (Boto3)

Mengambil daftar Solusi Mitra yang didaftarkan mitra di Partner Central.

```

#!/usr/bin/env python

"""
Purpose
PC-API-10 Getting list of solutions
"""

import logging
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
    service_name=serviceName,
    region_name='us-east-1'
)

def get_list_of_solutions():
    list_solutions_request = {
        "Catalog": CATALOG_TO_USE,
        "MaxResults": 20
    }
    try:
        # Perform an API call

```

```

        response =
partner_central_client.list_solutions(**list_solutions_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Getting list of solutions.")
    print("-" * 88)

    helper.pretty_print_datetime(get_list_of_solutions())

if __name__ == "__main__":
    usage_demo()

```

- Untuk detail API, lihat [ListSolutions](#) di AWS SDK for Python (Boto3) Referensi API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **RejectEngagementInvitation** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan `RejectEngagementInvitation`.

Java

SDK untuk Java 2.x

Menolak `EngagementInvitation` yang AWS dibagikan.

```

package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

```

```
import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
    software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
    software.amazon.awssdk.services.partnercentralselling.model.RejectEngagementInvitationResponse;
import
    software.amazon.awssdk.services.partnercentralselling.model.RejectEngagementInvitationRequest;

/*
 * Purpose
 * PC-API-05 AWS Originated(A0) rejection
 */

public class RejectEngagementInvitation {

    static PartnerCentralSellingClient client =
        PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {

        String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

        RejectEngagementInvitationResponse response = getResponse(opportunityId);

        ReferenceCodesUtils.formatOutput(response);
    }

    static RejectEngagementInvitationResponse getResponse(String invitationId) {

        RejectEngagementInvitationRequest rejectOpportunityRequest =
            RejectEngagementInvitationRequest.builder()
                .catalog(Constants.CATALOG_TO_USE)
                .identifier(invitationId)
                .rejectionReason("Unable to support")
                .build();
    }
}
```

```

    RejectEngagementInvitationResponse response =
    client.rejectEngagementInvitation(rejectOpportunityRequest);

        return response;
    }
}

```

- Untuk detail API, lihat [RejectEngagementInvitation](#) di Referensi AWS SDK for Java 2.x API.

Python

SDK untuk Python (Boto3)

Menolak EngagementInvitation yang AWS dibagikan.

```

#!/usr/bin/env python

"""
Purpose
PC-API-05 AWS Originated AO rejection - RejectOpportunityEngagementInvitation -
Rejects a engagement invitation.
This action indicates that the partner does not wish to participate in the
engagement and
provides a reason for the rejection.
Upon rejection, a OpportunityEngagementInvitationRejected event is triggered.
Subsequently, the invitation will no longer be available for the partner to act
on.
"""

import json
import logging
import boto3
import utils.helpers as helper

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
    service_name=serviceName,
    region_name='us-east-1'
)

```



```

def reject_opportunity_engagement_invitation(identifier, reject_reason):
    reject_opportunity_engagement_invitation_request = {
        "Catalog": CATALOG_TO_USE,
        "Identifier": identifier,
        "RejectionReason": reject_reason
    }
    try:
        # Perform an API call
        response =
partner_central_client.reject_engagement_invitation(**reject_opportunity_engagement_invi
        return response

    except Exception as err:
        # Catch all client exceptions
        print(json.dumps(err.response))

def usage_demo():
    identifier = "arn:aws:partnercentral:us-east-1::catalog/Sandbox/engagement-
invitation/engi-0000002isviga"
    reject_reason = "Customer problem unclear"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Given the ARN identifier and reject reason, reject the Opportunity
Engagement Invitation.")
    print("-" * 88)

    helper.pretty_print_datetime(reject_opportunity_engagement_invitation(identifier,
reject_reason))

if __name__ == "__main__":
    usage_demo()

```

- Untuk detail API, lihat [RejectEngagementInvitation](#) di AWS SDK for Python (Boto3) Referensi API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **StartEngagementByAcceptingInvitationTask** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan **StartEngagementByAcceptingInvitationTask**.

Java

SDK untuk Java 2.x

Memulai pertunangan dengan menerima EngagementInvitation.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
    software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
    software.amazon.awssdk.services.partnercentralselling.model.StartEngagementByAcceptingInvitationTask;
import
    software.amazon.awssdk.services.partnercentralselling.model.StartEngagementByAcceptingInvitationResponse;
import
    software.amazon.awssdk.services.partnercentralselling.model.GetEngagementInvitationRequest;
import
    software.amazon.awssdk.services.partnercentralselling.model.GetEngagementInvitationResponse;
import
    software.amazon.awssdk.services.partnercentralselling.model.InvitationStatus;

/*
Purpose
PC-API-04: Start Engagement By Accepting InvitationTask for AWS Originated(A0)
opportunity
*/
```

```
public class StartEngagementByAcceptingInvitationTask {

    static PartnerCentralSellingClient client =
        PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    static String clientToken = "test-a30d161";

    public static void main(String[] args) {

        String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

        StartEngagementByAcceptingInvitationTaskResponse response =
            getResponse(opportunityId);

        if ( response == null) {
            System.out.println("Opportunity is not AWS Originated.");
        } else {
            ReferenceCodesUtils.formatOutput(response);
        }
    }

    private static GetEngagementInvitationResponse getInvitation(String
invitationId) {

        GetEngagementInvitationRequest getRequest =
            GetEngagementInvitationRequest.builder()
                .catalog(Constants.CATALOG_TO_USE)
                .identifier(invitationId)
                .build();

        GetEngagementInvitationResponse response =
            client.getEngagementInvitation(getRequest);

        return response;
    }

    static StartEngagementByAcceptingInvitationTaskResponse getResponse(String
invitationId) {
```

```

    if ( getInvitation(invitationId).status().equals(InvitationStatus.PENDING)) {
        StartEngagementByAcceptingInvitationTaskRequest acceptOpportunityRequest =
            StartEngagementByAcceptingInvitationTaskRequest.builder()
                .catalog(Constants.CATALOG_TO_USE)
                .identifier(invitationId)
                .clientToken(clientToken)
                .build();

        StartEngagementByAcceptingInvitationTaskResponse response =
            client.startEngagementByAcceptingInvitationTask(acceptOpportunityRequest);
        return response;
    }
    return null;
}
}

```

- Untuk detail API, lihat [StartEngagementByAcceptingInvitationTask](#) di Referensi AWS SDK for Java 2.x API.

Python

SDK untuk Python (Boto3)

Memulai pertunangan dengan menerima EngagementInvitation.

```

#!/usr/bin/env python

"""
Purpose
PC-API -11 Associating a product
PC-API -12 Associating a solution
PC-API -13 Associating an offer
"""

import logging
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

```

```
partner_central_client = boto3.client(
    service_name=serviceName,
    region_name='us-east-1'
)

def get_opportunity(identifier):
    get_opportunity_request = {
        "Identifier": identifier,
        "Catalog": CATALOG_TO_USE
    }
    try:
        # Perform an API call
        response =
partner_central_client.get_engagement_invitation(**get_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def start_engagement_by_accepting_invitation_task(identifier):

    response = get_opportunity(identifier)

    if ( response['Status'] == 'PENDING') :
        accept_opportunity_engagement_invitation_request = {
            "Catalog": CATALOG_TO_USE,
            "Identifier" : identifier,
            "ClientToken": "test-123456"
        }
        try:
            # Perform an API call
            response =
partner_central_client.start_engagement_by_accepting_invitation_task(**accept_opportunity_engagement_invitation_request)
            return response

        except ClientError as err:
            # Catch all client exceptions
            print(err.response)
            return None
    else:
        return None
```

```
def usage_demo():
    identifier = "arn:aws:partnercentral:us-east-1::catalog/Sandbox/engagement-
invitation/engi-0000002isusga"
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Get updated Opportunity.")
    print("-" * 88)

    helper.pretty_print_datetime(start_engagement_by_accepting_invitation_task(identifier))

if __name__ == "__main__":
    usage_demo()
```

- Untuk detail API, lihat [StartEngagementByAcceptingInvitationTask](#) di AWS SDK for Python (Boto3) Referensi API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **StartEngagementFromOpportunityTask** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan **StartEngagementFromOpportunityTask**.

Java

SDK untuk Java 2.x

Memulai proses keterlibatan dari peluang yang ada dengan menerima undangan keterlibatan dan menciptakan peluang yang sesuai dalam sistem mitra.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
```

```

import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
    software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import software.amazon.awssdk.services.partnercentralselling.model.AwsSubmission;
import
    software.amazon.awssdk.services.partnercentralselling.model.SalesInvolvementType;
import
    software.amazon.awssdk.services.partnercentralselling.model.StartEngagementFromOpportunityTask;
import
    software.amazon.awssdk.services.partnercentralselling.model.StartEngagementFromOpportunityTaskResponse;
import software.amazon.awssdk.services.partnercentralselling.model.Visibility;

/*
 * Purpose
 * PC-API-01 Partner Originated (PO) opp submission(Start Engagement From
 * Opportunity Task for AO Originated Opportunity)
 */

public class StartEngagementFromOpportunityTask {

    static PartnerCentralSellingClient client =
        PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {

        String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

        StartEngagementFromOpportunityTaskResponse response =
            getResponse(opportunityId);

        ReferenceCodesUtils.formatOutput(response);
    }

    static StartEngagementFromOpportunityTaskResponse getResponse(String
        opportunityId) {

        StartEngagementFromOpportunityTaskRequest submitOpportunityRequest =
            StartEngagementFromOpportunityTaskRequest.builder()
                .catalog(Constants.CATALOG_T0_USE)

```

```

        .identifier(opportunityId)
        .clientToken("test-annjqwesdsd99")

        .awsSubmission(AwsSubmission.builder().involvementType(SalesInvolvementType.CO_SELL).vis
            .build());

        StartEngagementFromOpportunityTaskResponse response =
        client.startEngagementFromOpportunityTask(submitOpportunityRequest);

        return response;
    }
}

```

- Untuk detail API, lihat [StartEngagementFromOpportunityTask](#) di Referensi AWS SDK for Java 2.x API.

Python

SDK untuk Python (Boto3)

Memulai proses keterlibatan dari peluang yang ada dengan menerima undangan keterlibatan dan menciptakan peluang yang sesuai dalam sistem mitra.

```

#!/usr/bin/env python

"""
Purpose
PC-API -11 Associating a product
PC-API -12 Associating a solution
PC-API -13 Associating an offer
"""

import logging
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

```



```
partner_central_client = boto3.client(
    service_name=serviceName,
    region_name='us-east-1'
)

def start_engagement_from_opportunity_task(identifier):

    start_engagement_from_opportunity_task_request = {
        "AwsSubmission": {
            "InvolvementType": "Co-Sell",
            "Visibility": "Full"
        },
        "Catalog": CATALOG_TO_USE,
        "Identifier" : identifier,
        "ClientToken": "test-annjqwesdsd99"
    }
    try:
        # Perform an API call
        response =
partner_central_client.start_engagement_from_opportunity_task(**start_engagement_from_op
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)
        return None

def usage_demo():
    identifier = "05465588"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Start Engagement from Opportunity Task.")
    print("-" * 88)

    helper.pretty_print_datetime(start_engagement_from_opportunity_task(identifier))

if __name__ == "__main__":
    usage_demo()
```

- Untuk detail API, lihat [StartEngagementFromOpportunityTask](#) di AWS SDK for Python (Boto3) Referensi API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Gunakan **UpdateOpportunity** dengan AWS SDK

Contoh kode berikut menunjukkan cara menggunakan `UpdateOpportunity`.

Java

SDK untuk Java 2.x

Perbarui peluang.

```
package org.example;

import java.time.Instant;
import java.util.ArrayList;
import java.util.List;

import static org.example.utils.Constants.*;

import org.example.entity.Root;
import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;
import org.example.utils.StringSerializer;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
    software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import software.amazon.awssdk.services.partnercentralselling.model.Account;
import software.amazon.awssdk.services.partnercentralselling.model.Address;
import software.amazon.awssdk.services.partnercentralselling.model.Contact;
import software.amazon.awssdk.services.partnercentralselling.model.Customer;
import
    software.amazon.awssdk.services.partnercentralselling.model.ExpectedCustomerSpend;
```

```
import
    software.amazon.awssdk.services.partnercentralselling.model.GetOpportunityRequest;
import
    software.amazon.awssdk.services.partnercentralselling.model.GetOpportunityResponse;
import software.amazon.awssdk.services.partnercentralselling.model.LifeCycle;
import software.amazon.awssdk.services.partnercentralselling.model.Marketing;
import
    software.amazon.awssdk.services.partnercentralselling.model.NextStepsHistory;
import software.amazon.awssdk.services.partnercentralselling.model.Project;
import software.amazon.awssdk.services.partnercentralselling.model.ReviewStatus;
import
    software.amazon.awssdk.services.partnercentralselling.model.UpdateOpportunityRequest;
import
    software.amazon.awssdk.services.partnercentralselling.model.UpdateOpportunityResponse;

import com.google.gson.Gson;
import com.google.gson.GsonBuilder;
import com.google.gson.ToNumberPolicy;

/*
 * Purpose
 * PC-API-02/06 Update opportunity when LifeCycle.ReviewStatus is not Submitted
 or In-Review
 */

public class UpdateOpportunity {

    static final Gson GSON = new GsonBuilder()
        .setObjectToNumberStrategy(ToNumberPolicy.LAZILY_PARSED_NUMBER)
        .registerTypeAdapter(String.class, new StringSerializer())
        .create();

    static PartnerCentralSellingClient client =
        PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    static String OPPORTUNITY_ORIGIN = ORIGIN_PARTNER_ORIGINATED;

    public static void main(String[] args) {

        String inputFile = "updateOpportunity.json";
```

```
if (args.length > 0)
    inputFile = args[0];

UpdateOpportunityResponse response = updateOpportunity(inputFile);

client.close();
}

public static GetOpportunityResponse getResponse(String opportunityId) {

    GetOpportunityRequest getOpportunityRequest =
GetOpportunityRequest.builder()
    .catalog(Constants.CATALOG_TO_USE)
    .identifier(opportunityId)
    .build();

    GetOpportunityResponse response =
client.getOpportunity(getOpportunityRequest);
    System.out.println(opportunityId + ":" + response);
    return response;
}

public static UpdateOpportunityResponse updateOpportunity(String inputFile) {

    String inputString = ReferenceCodesUtils.readInputFileToString(inputFile);

    Root root = GSON.fromJson(inputString, Root.class);
    GetOpportunityResponse response = getResponse(root.identifier);

    if (response != null
        && response.lifeCycle() != null
        && response.lifeCycle().reviewStatus() != null
        && response.lifeCycle().reviewStatus() != ReviewStatus.SUBMITTED
        && response.lifeCycle().reviewStatus() != ReviewStatus.IN_REVIEW) {

        List<NextStepsHistory> nextStepsHistories = new ArrayList<NextStepsHistory>();
        if ( root.lifeCycle != null && root.lifeCycle.nextStepsHistories != null) {
            for (org.example.entity.NextStepsHistory nextStepsHistoryJson :
root.lifeCycle.nextStepsHistories) {
                NextStepsHistory nextStepsHistory = NextStepsHistory.builder()
                    .time(Instant.parse(nextStepsHistoryJson.time))
                    .value(nextStepsHistoryJson.value)
                    .build();
```

```
        nextStepsHistories.add(nextStepsHistory);
    }
}

Lifecycle lifeCycle = null;
if ( root.lifeCycle != null ) {
    lifeCycle = Lifecycle.builder()
        .closedLostReason(root.lifeCycle.closedLostReason)
        .nextSteps(root.lifeCycle.nextSteps)
        .nextStepsHistory(nextStepsHistories)
        .reviewComments(root.lifeCycle.reviewComments)
        .reviewStatus(root.lifeCycle.reviewStatus)
        .reviewStatusReason(root.lifeCycle.reviewStatusReason)
        .stage(root.lifeCycle.stage)
        .targetCloseDate(root.lifeCycle.targetCloseDate)
        .build();
}

Marketing marketing = null;
if ( root.marketing != null ) {
    marketing = Marketing.builder()
        .awsFundingUsed(root.marketing.awsFundingUsed)
        .campaignName(root.marketing.campaignName)
        .channels(root.marketing.channels)
        .source(root.marketing.source)
        .useCases(root.marketing.useCases)
        .build();
}

Address address = null;
if (root.customer != null && root.customer.account != null &&
root.customer.account.address != null) {
    address =
Address.builder().postalCode(root.customer.account.address.postalCode)
    .stateOrRegion(root.customer.account.address.stateOrRegion)
    .countryCode(root.customer.account.address.countryCode).build();
}

Account account = null;
if (root.customer != null && root.customer.account != null) {
    account = Account.builder().address(address).duns(root.customer.account.duns)

.industry(root.customer.account.industry).companyName(root.customer.account.companyName)
```

```

        .websiteUrl(root.customer.account.websiteUrl).build();
    }

    List<Contact> contacts = new ArrayList<Contact>();
    if ( root.customer != null && root.customer.contacts != null) {
        for (org.example.entity.Contact jsonContact : root.customer.contacts) {
            Contact contact = Contact.builder()
                .email(jsonContact.email)
                .firstName(jsonContact.firstName)
                .lastName(jsonContact.lastName)
                .phone(jsonContact.phone)
                .businessTitle(jsonContact.businessTitle)
                .build();
            contacts.add(contact);
        }
    }

    Customer customer =
    Customer.builder().account(account).contacts(contacts).build();

    List<ExpectedCustomerSpend> expectedCustomerSpends = new
    ArrayList<ExpectedCustomerSpend>();
    if ( root.project != null && root.project.expectedCustomerSpend != null) {
        for (org.example.entity.ExpectedCustomerSpend expectedCustomerSpendJson :
    root.project.expectedCustomerSpend) {
            ExpectedCustomerSpend expectedCustomerSpend = null;
            expectedCustomerSpend = ExpectedCustomerSpend.builder()
                .amount(expectedCustomerSpendJson.amount)
                .currencyCode(expectedCustomerSpendJson.currencyCode)
                .frequency(expectedCustomerSpendJson.frequency)
                .targetCompany(expectedCustomerSpendJson.targetCompany)
                .build();
            expectedCustomerSpends.add(expectedCustomerSpend);
        }
    }

    Project project = null;
    if (root.project != null) {
        project = Project.builder().title(root.project.title)
            .customerBusinessProblem(root.project.customerBusinessProblem)

        .customerUseCase(root.project.customerUseCase).deliveryModels(root.project.deliveryModel
            .expectedCustomerSpend(expectedCustomerSpends)

```

```

        .salesActivities(root.project.salesActivities).competitorName(root.project.competitorName)
        .otherSolutionDescription(root.project.otherSolutionDescription).build();
    }

    // Building the Actual CreateOpportunity Request
    UpdateOpportunityRequest updateOpportunityRequest =
    UpdateOpportunityRequest.builder().catalog(root.catalog)

    .identifier(root.identifier).lastModifiedDate(Instant.parse(root.lastModifiedDate))

    .primaryNeedsFromAwsWithStrings(root.primaryNeedsFromAws).opportunityType(root.opportunityType)
    .lifeCycle(lifeCycle)
    .customer(customer)
    .project(project)
    .partnerOpportunityIdentifier(root.partnerOpportunityIdentifier)
    .marketing(marketing)
    .nationalSecurity(root.nationalSecurity)
    .opportunityType(root.opportunityType)
    .build();

    UpdateOpportunityResponse updateResponse =
    client.updateOpportunity(updateOpportunityRequest);
    System.out.println("Successfully updated opportunity: " + updateResponse);

    return updateResponse;
} else {
    System.out.println("Opportunity cannot be updated.");
    return null;
}
}
}

```

- Untuk detail API, lihat [UpdateOpportunity](#) di Referensi AWS SDK for Java 2.x API.

Python

SDK untuk Python (Boto3)

Perbarui peluang.

```
#!/usr/bin/env python

"""
Purpose
PC-API-2 Updating Partner Originated Opportunity
"""
import logging
import boto3
import sys
import os
sys.path.append(os.path.dirname(os.path.dirname(os.path.abspath(__file__))))
import utils.helpers as helper
from botocore.client import ClientError
import utils.stringify_details as sd
from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
    service_name=serviceName,
    region_name='us-east-1'
)

def get_opportunity(identifier):
    get_opportunity_request = {
        "Identifier": identifier,
        "Catalog": CATALOG_TO_USE
    }
    try:
        # Perform an API call
        response =
partner_central_client.get_opportunity(**get_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def update_opportunity():
    update_opportunity_request_orig = sd.stringify_json("src/update_opportunity/
update_opportunity_technical_validation.json")
    update_opportunity_request =
    helper.remove_nulls(update_opportunity_request_orig)
```



```

try:
    # Perform an API call
    response =
partner_central_client.update_opportunity(**update_opportunity_request)
    return response

except ClientError as err:
    # Catch all client exceptions
    print(err.response)

def update_opportunity_if_eligible(identifier):
    response = get_opportunity(identifier)
    if response is not None:
        return update_opportunity()
    else:
        print("Failed to retrieve opportunity details")

def usage_demo():
    identifier = "05465588"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Updating opportunity.")
    print("-" * 88)

    helper.pretty_print_datetime(update_opportunity_if_eligible(identifier))

if __name__ == "__main__":
    usage_demo()

```

- Untuk detail API, lihat [UpdateOpportunity](#) di AWS SDK for Python (Boto3) Referensi API.

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Skenario untuk Partner Central menggunakan AWS SDK

Contoh kode berikut menunjukkan cara menerapkan skenario umum di Partner Central dengan AWS SDK. Skenario ini menunjukkan kepada Anda cara menyelesaikan tugas tertentu dengan memanggil

beberapa fungsi di dalam Partner Central atau digabungkan dengan yang lain Layanan AWS. Setiap skenario menyertakan tautan ke kode sumber lengkap, di mana Anda dapat menemukan instruksi tentang cara mengatur dan menjalankan kode.

Skenario menargetkan tingkat pengalaman menengah untuk membantu Anda memahami tindakan layanan dalam konteks.

Contoh

- [Perbarui entitas terkait dari suatu peluang](#)

Perbarui entitas terkait dari suatu peluang

Contoh-contoh kode berikut menunjukkan cara:

- Putus entitas lama.
- Mengasosiasikan entitas baru.

Java

SDK untuk Java 2.x

Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankan di repositori [Skenario](#).

Perbarui entitas terkait dari suatu peluang

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
```

```

import
    software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
    software.amazon.awssdk.services.partnercentralselling.model.AssociateOpportunityRequest;
import
    software.amazon.awssdk.services.partnercentralselling.model.AssociateOpportunityResponse;
import
    software.amazon.awssdk.services.partnercentralselling.model.DisassociateOpportunityRequest;
import
    software.amazon.awssdk.services.partnercentralselling.model.DisassociateOpportunityResponse;

/*
Purpose
PC-API -17 Replacing a solution
*/

public class ReplaceSolution {

    static PartnerCentralSellingClient client =
        PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {

        String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

        String entityType = "Solutions";
        String originalEntityIdentifier = "S-00000000";
        String newEntityIdentifier = "S-0011111";

        disassociateOpportunityResponse(opportunityId, entityType,
originalEntityIdentifier );
        AssociateOpportunityResponse associateOpportunityResponse =
associateOpportunityResponse(opportunityId, entityType, newEntityIdentifier );

        ReferenceCodesUtils.formatOutput(associateOpportunityResponse);
    }

    private static AssociateOpportunityResponse associateOpportunityResponse(String
opportunityId, String entityType, String entityIdentifier) {

```

```
AssociateOpportunityRequest associateOpportunityRequest =
AssociateOpportunityRequest.builder()
    .catalog(Constants.CATALOG_TO_USE)
    .opportunityIdentifier(opportunityId)
    .relatedEntityType(entityType)
    .relatedEntityIdentifier(entityIdentifier)
    .build();

AssociateOpportunityResponse response =
client.associateOpportunity(associateOpportunityRequest);

return response;
}

private static DisassociateOpportunityResponse
disassociateOpportunityResponse(String opportunityId, String entityType, String
entityIdentifier) {
    PartnerCentralSellingClient client = PartnerCentralSellingClient.builder()
        .region(Region.US_EAST_1)
        .credentialsProvider(DefaultCredentialsProvider.create())
        .httpClient(ApacheHttpClient.builder().build())
        .build();

    DisassociateOpportunityRequest disassociateOpportunityRequest =
DisassociateOpportunityRequest.builder()
        .catalog(Constants.CATALOG_TO_USE)
        .opportunityIdentifier(opportunityId)
        .relatedEntityType(entityType)
        .relatedEntityIdentifier(entityIdentifier)
        .build();

    DisassociateOpportunityResponse response =
client.disassociateOpportunity(disassociateOpportunityRequest);

    return response;
}
}
```

- Untuk detail API, lihat topik berikut di Referensi API AWS SDK for Java 2.x.
 - [AssociateOpportunity](#)
 - [DisassociateOpportunity](#)

Python

SDK untuk Python (Boto3)

Note

Ada lebih banyak tentang GitHub. Temukan contoh lengkapnya dan pelajari cara mengatur dan menjalankannya di [Repositori Contoh Kode AWS](#).

Perbarui Entitas Terkait dari suatu peluang

```
#!/usr/bin/env python

"""
Purpose
PC-API -17 Replacing a solution
"""

import logging
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
    service_name=serviceName,
    region_name='us-east-1'
)

def replace_solution(original_entity_identifier, new_entity_identifier,
    opportunityIdentifier):
    disassociate_opportunity_request = {
        "Catalog": CATALOG_TO_USE,
        "OpportunityIdentifier" : opportunityIdentifier,
        "RelatedEntityType" : "Solutions",
        "RelatedEntityIdentifier" : original_entity_identifier
    }

    associate_opportunity_request = {
```

```

        "Catalog": CATALOG_TO_USE,
        "OpportunityIdentifier" : opportunityIdentifier,
        "RelatedEntityType" : "Solutions",
        "RelatedEntityIdentifier" : new_entity_identifier
    }
    try:
        # Perform an API call
        response =
partner_central_client.disassociate_opportunity(**disassociate_opportunity_request)
        response =
partner_central_client.associate_opportunity(**associate_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def usage_demo():
    original_entity_identifier = "S-0049999"
    new_entity_identifier = "S-0050014"
    opportunityIdentifier = "04397574"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Replacing a solution.")
    print("-" * 88)

    helper.pretty_print_datetime(replace_solution(original_entity_identifier,
new_entity_identifier, opportunityIdentifier))

if __name__ == "__main__":
    usage_demo()

```

- Untuk detail API, lihat topik berikut di Referensi API AWS SDK untuk Python (Boto3).
 - [AssociateOpportunity](#)
 - [DisassociateOpportunity](#)

Untuk daftar lengkap panduan pengembang AWS SDK dan contoh kode, lihat [Penggunaan AWS Mitra Central API dengan AWS SDK](#). Topik ini juga mencakup informasi tentang memulai dan detail tentang versi SDK sebelumnya.

Catatan rilis

Halaman ini berisi ringkasan perubahan signifikan pada dokumentasi untuk API Pusat AWS Mitra, dimulai dengan pembaruan terbaru.

Riwayat revisi

Ubah	Deskripsi	Date
RC 6.0	AWS Partner Central menambahkan API Pengukuran Pendapatan Mitra baru.	2026-06-30
RC 5.3	Pembaruan ACE Resonate: GetAwsOpportunitySummary respons diperbarui.	2026-06-16
RC 5.2	- Dukungan mata uang EUR: CreateOpportunity, UpdateOpportunity, dan CreateEngagementInvitation sekarang terima EUR selain USD untuk kode mata uang MRR. Ketika AWS Partition adalah aws-eusc (AWS European Sovereign Cloud), kode mata uangnya harus EUR. - EUR dalam operasi baca: GetOpportunity, ListOpportunities, dan GetAwsOpportunitySummary sekarang kembali	2026-03-30

Ubah	Deskripsi	Date
	EUR di CurrencyCode lapangan untuk peluang di aws-eusc partisi.	
RC 5.1	• Dukungan mata uang EUR: CreateOppportunity UpdateOppportunity , dan CreateEngagementInvitation sekarang terima EUR selain USD untuk kode mata uang MRR. Ketika AWS Partition adalah aws-eusc (AWS European Sovereign Cloud), kode mata uangnya harus. EUR • EUR dalam operasi baca:GetOppportunity ,ListOpportunities , dan GetAwsOppportunitySummary sekarang kembali EUR di CurrencyCode lapangan untuk peluang di aws-eusc partisi.	2026-03-30

Ubah	Deskripsi	Date
RC 5	• API Akun yang Diperkenalkan: Menambahkan API Akun baru ke AWS Partner Central, memungkinkan mitra untuk mengelola informasi akun, pendaftaran, profil, manajemen domain, dan koneksi mitra mereka. • Dokumentasi referensi API Akun: Menambahkan dokumentasi referensi API komprehensif untuk API Akun, termasuk pendaftaran mitra, manajemen profil, verifikasi domain, dan alur kerja koneksi akun. • Pencatatan API akun: Menambahkan dukungan logging dan dokumentasi untuk API Akun untuk membantu mengaudit dan melacak penggunaan API. • Pemberitahuan API Akun: Menambahkan dukungan pemberitahuan acara untuk API Akun, memungkinkan mitra menerima pembaruan real-time tentang aktivitas terkait akun termasuk koneksi akun mitra. • Kuota API Akun: Menambahkan dokumentasi kuota layanan untuk API Akun.	2025-11-30

Ubah	Deskripsi	Date
	• Pengujian kotak pasir untuk API Akun: Menambahk an kemampuan pengujian kotak pasir untuk API Akun, memungkinkan mitra untuk menguji alur kerja manajemen akun di lingkungan non-produksi. • Memperkenalkan Benefits API: Menambahkan API Benefits baru ke AWS Partner Central, memungkinkan mitra untuk menemukan, mengajukan, dan mengelola manfaat di seluruh Program AWS Mitra melalui antarmuka terpusat. • Dokumentasi referensi API Benefits: Menambahkan dokumentasi referensi API komprehensif untuk Benefits API, yang mencakup aplikasi manfaat, alokasi, dan proses pemenuhan. • Benefits API logging: Menambahkan dukungan logging dan dokumentasi untuk Benefits API untuk membantu mengaudit dan melacak penggunaan API. • Pemberitahuan API Manfaat: Menambahkan dukungan pemberitahuan acara untuk API Manfaat,	

Ubah	Deskripsi	Date
	memungkinkan mitra menerima pembaruan waktu nyata tentang aplikasi dan alokasi manfaat. • Kuota API Manfaat: Menambahkan dokumentasi kuota layanan untuk API Manfaat. • Pengujian kotak pasir untuk API Manfaat: Menambahkan kemampuan pengujian kotak pasir untuk API Manfaat, memungkinkan mitra untuk menguji alur kerja manajemen manfaat di lingkungan non-produksi. • API Penjualan yang Diperbarui: Tindakan keterlibatan API Penjualan yang Ditingkatkan untuk mendukung alur kerja manajemen prospek. • Menjual pembaruan dokumentasi API: Dokumentasi API Penjualan yang disempurnakan dengan struktur yang ditingkatkan dan alur kerja yang diperbarui untuk manajemen prospek. • Menjual pembaruan pemberitahuan API: Dokumentasi pemberitahuan acara yang disempurnakan	

Ubah	Deskripsi	Date
	akan untuk acara Undangan Keterlibatan dan Keterlibatan. • Ekspansi struktur API: Struktur dokumentasi yang diperluas untuk mendukung empat API berbeda (Akun, Manfaat, Penjualan, dan Saluran) dengan bagian khusus untuk setiap izin API, logging, notifikasi, kuota, dan pengujian kotak pasir. • Dokumentasi penggunaan API yang disempurnakan: Menambahkan panduan penggunaan komprehensif untuk API Akun dan Manfaat dengan alur kerja terperinci dan pola integrasi. • Dokumentasi wilayah yang didukung: Dokumentasi tambahan untuk AWS wilayah yang didukung untuk API Akun dan Manfaat. • Filter baru untuk <code>ListOpportunities</code> : Menambahkan <code>TargetCloseDate</code> filter untuk mengaktifkan peluang pemfilteran berdasarkan tanggal penutupan yang diharapkan menggunakan <code>AfterTargetCloseDa</code>	

Ubah	Deskripsi	Date
	te dan BeforeTar getCloseDate bidang dalam YYYY-MM-DD format.	

Ubah	Deskripsi	Date
RC 4	• API Saluran yang Diperkenalkan: Menambahkan API Saluran baru ke AWS Partner Central, memungkinkan mitra mengelola hubungan saluran dan kolaborasi mitra-ke-mitra. • Dokumentasi referensi API Saluran: Menambahkan dokumentasi referensi API komprehensif untuk API Saluran, termasuk semua tindakan dan model data yang tersedia. • Izin API Saluran: Menambahkan kebijakan izin IAM dan dokumentasi kontrol akses khusus ke API Saluran. • Pencatatan API Saluran: Menambahkan dukungan logging dan dokumentasi untuk API Saluran untuk membantu mengaudit dan melacak penggunaan API. • Pemberitahuan API Saluran: Menambahkan dukungan pemberitahuan acara untuk API Saluran, memungkinkan mitra menerima pembaruan waktu nyata tentang aktivitas terkait saluran.	2025-11-19

Ubah	Deskripsi	Date
	• Kuota API Saluran: Menambahkan dokumentasi kuota layanan untuk API Saluran. • Pengujian kotak pasir untuk Channel API: Menambahkan kemampuan pengujian kotak pasir untuk API Saluran, memungkinkan mitra untuk menguji alur kerja saluran di lingkungan non-produksi. • Reorganisasi struktur API: Dokumentasi yang direorganisasi untuk membedakan antara API Penjualan dan API Saluran, dengan bagian khusus untuk setiap izin, logging, notifikasi, dan kuota API. • Dokumentasi wilayah yang didukung: Menambahkan dokumentasi untuk AWS wilayah yang didukung untuk API Penjualan dan Saluran.	

Ubah	Deskripsi	Date
RC 3	• Kebijakan izin API yang diperbarui: Kebijakan telah diperbarui untuk mencerminkan tindakan baru yang diperkenalkan dalam rilis ini. Pastikan peran dan izin IAM Anda disesuaikan. • Detail penggunaan header khusus: Menambahkan detail tentang cara menggunakan header khusus X-Amzn-User-Agent untuk membantu mengaudit dan mengukur penggunaan API. • Tindakan yang diganti dengan tindakan asinkron: Diganti <code>AcceptOpportunityEngagementInvitation</code> dengan tindakan asinkron <code>StartEngagementByAcceptingInvitationTask</code> dan format muatan telah berubah sehubungan dengan RC2. • Penggantian nama entitas dan tindakan: Diubah <code>OpportunityEngagementInvitation</code> menjadi <code>EngagementInvitation</code>. Atribut untuk tindakan ini juga telah berubah untuk menyelara	2024-10-17

Ubah	Deskripsi	Date
	skan dengan entitas baru. Tindakan berikut diganti: • <code>GetOpportunityEngagementInvitation</code> sekarang <code>GetEngagementInvitation</code> . Beberapa atribut dalam undangan telah diubah, ditambahkan, atau dihapus. • <code>ListOpportunityEngagementInvitations</code> sekarang <code>ListEngagementInvitations</code> . • <code>RejectOpportunityEngagementInvitation</code> sekarang <code>RejectEngagementInvitation</code> . • <code>AssignOpportunity</code> pembaruan parameter: <code>AssignOpportunity</code> sekarang mengambil <code>Assignee</code> alih-alih <code>AssigneeEmail</code> . Perbarui implementasi Anda sesuai dengan itu. • <code>SubmitOpportunity</code> perubahan fungsionalitas: sekarang <code>SubmitOpportunity</code> dilakukan oleh tindakan asinkron. <code>StartEnga</code>	

Ubah	Deskripsi	Date
	gementFromOpportunityTask SubmitOpportunity sekarang menerima InvolvementType danVisibility . • Penggantian nama dan ekspansi atribut: EstimatedAwsMonthlyRevenue diubah menjadiExpectedCustomerSpend . ExpectedCustomerSpend sekarang termasuk atribut baru, seperti Frequency danTargetCompany . • Tindakan baru untuk ringkasan AWS peluang: Visibilitas peluang sekarang disediakan melalui tindakan terpisah yang disebutGetAwsOpportunitySummary . • Pembaruan alur kerja untuk bekerja dengan peluang: Alur kerja untuk Bekerja dengan peluang Anda diperbarui untuk menggunakan tindakan baru yang diperkenalkan dalam rilis ini. • Pembaruan alur kerja untuk bekerja dengan peluang: Alur kerja untuk	

Ubah	Deskripsi	Date
	Bekerja dengan peluang diperbarui untuk digunakan EngagementInvitations , yang meningkatkan peluang penanganan yang disediakan oleh AWS. • Pembaruan alur kerja untuk melacak pembaruan : Alur kerja untuk Melacak Pembaruan sekarang Bekerja dengan pembaruan peluang, meningkatkan kejelasan dan efisiensi dalam memantau status peluang. • Filter baru untuk tindakan daftar: Filter baru tersedia untuk ListSolutions dan ListOpportunities tindakan. • UntukListSolutions , filter baru adalahFilterCategory ,FilterIdentifier , danFilterStatus . • UntukListOpportunities , filter baru adalahFilterCustomerCompany ,FilterIdentifier ,FilterLastModifiedDate ,FilterLif	

Ubah	Deskripsi	Date
	eCycleApp rovalStat us ,FilterLif eCycleStage , danFilterOrigin . • Penghapusan AccessControl subobject: AccessControl Subobject dalam model peluang telah dihapus. NationalSecurity sekarang atribut tingkat atas. • Atribut dipindahkan keGetAwsOpportunitySummary : Atribut sepertiAWSOpportunityTeam , insight (EngagementScore ,NextBestAction), origin, dan sekarang InvolvementType tersedia melalui GetAwsOpportunitySummary alih-alih. GetOpportunity • Pengenalan InvolvementType lapangan: Memperkenalkan InvolvementType bidang untuk membedakan antara peluang cosell dan visibilitas saja.	

Ubah	Deskripsi	Date
	• Pembaruan nama acara: Acara diperbarui untuk mencerminkan perubahan . Misalnya, undangan keterlibatan Peluang yang dibuat sekarang adalah undangan Keterlibatan dibuat. • Pembaruan payload respons: Semua muatan respons sekarang menyertakan ID and/or ARN. • Minta pembaruan payload: Semua muatan permintaan sekarang menggunakan pengidentifikasi sebagai input, yang menerima ID atau ARN. • Pembaruan entitas keterlibatan: Atribut judul dalam entitas keterlibatan diubah namanya EngagementTitle . • Penghapusan awalan filter: Semua filter tidak lagi menyertakan awalan filter, menyederhanakan mekanisme penyaringan. • StartEngagementFromOpportunity pembaruan tindakan: Atribut AwsInvolvement	

Ubah	Deskripsi	Date
	diubah menjadi <code>AwsSubmission</code> . • Pembaruan detail proyek undangan: <code>Invitation.ProjectDetails.TargetCompletionDate</code> diganti namanya <code>Invitation.ProjectDetails.EstimatedCompletionDate</code> . • Pembaruan estimasi pendapatan: <code>Invitation.PartnerRevenueEstimate</code> sekarang <code>ExpectedCustomerSpend</code> . Tipe data berubah dari satu objek ke array yang berisi satu objek. • Pembaruan akun penerima: Bidang <code>ReceiverAccount</code> diganti namanya <code>AccountReceiver</code> . • Atribut baru undangan keterlibatan: <code>SenderContact</code> atribut diperkenalkan untuk undangan keterlibatan. • <code>CustomerContact</code> sekarang array bukan objek. <code>OpportunityOwner</code> , <code>PartnerAc</code>	

Ubah	Deskripsi	Date
	<code>countManager</code> adalah nilai yang didukung sebagai <code>BusinessTitle</code> . • <code>PartnerOpportunityTeam</code> sekarang disebut <code>OpportunityTeam</code> . • <code>APNPrograms</code> sekarang <code>ApnPrograms</code> • Status undangan keterlibatan diperbarui. • <code>ListEngagementInvitations</code> sekarang membutuhkan <code>ParticipantType</code> sebagai <code>RECEIVER</code>.	

Ubah	Deskripsi	Date
RC 2	• Memperkenalkan entitas baru yang disebut OpportunityEngagementInvitation. • Memperkenalkan aksi baru: ListOpportunityEngagementInvitations. • Memperkenalkan aksi baru: GetOpportunityEngagementInvitation. • Memperkenalkan acara baru: “Undangan Keterlibatan Peluang Dibuat”. • Diganti AcceptOpportunity dengan AcceptOpportunityEngagementInvitation. • Diganti RejectOpportunity dengan RejectOpportunityEngagementInvitation. • Mengganti “Peluang Diterima” dengan “Undangan Keterlibatan Peluang Diterima”. • Mengganti “Peluang Ditolak” dengan “Undangan Keterlibatan Peluang Ditolak”. • PrimaryNeedsFromAWS berubah menjadi PrimaryNeedsFromAws. • AWSOpportunityTeam berubah menjadiAwsOpportunityTeam .	2024-08-07

Ubah	Deskripsi	Date
	• <code>AWSSalesLifeCycle</code> berubah menjadi <code>AwsSalesLifeCycle</code> . • <code>PrimaryNeedsFromAWSChangeReason</code> berubah menjadi <code>PrimaryNeedsFromAwsChangeReason</code> . • <code>AWSAccountId</code> berubah menjadi <code>AwsAccountId</code> . • <code>ExpectedMonthlyAWSRevenue</code> berubah menjadi <code>ExpectedMonthlyAwsRevenue</code> . • <code>AWSFundingUsed</code> berubah menjadi <code>AwsFundingUsed</code> . • <code>AWSMarketplaceOffers</code> berubah menjadi <code>AwsMarketplaceOffers</code> . • <code>Country</code> berubah menjadi <code>CountryCode</code> . • <code>PartnerOpportunityContact</code> berubah menjadi <code>PartnerOpportunityTeam</code> . • Bidang yang diperbarui <code>Contact.Title</code> ke <code>Contact.BusinessTitle</code> .	

Ubah	Deskripsi	Date
	• Bidang yang diperbarui <code>ContactInfo</code> ke <code>OwnerInfo</code> . • Mengubah AWS Tim dari peta ke daftar. • Menambahkan rincian tentang daftar diterima <code>RejectionReasons</code>. • Memberikan informasi tentang cara menggunakan Token Klien. • Termasuk detail tentang cara menggunakan <code>UpdateOpportunity</code>. • Menambahkan panduan tentang penggunaan AWS Produk dan Solusi Mitra. • Memberikan rincian tentang <code>OpportunityEngagementInvitation</code> entitas. • Diperbarui <code>StateOrRegion</code> untuk menyertakan daftar nilai yang diterima yang direvisi. • Menerapkan beberapa perbaikan bug untuk meningkatkan kinerja dan pesan kesalahan.	

Ubah	Deskripsi	Date
RC 1	• [Breaking] Mengganti nama menjadi ReviewStatus: Kami telah mengganti nama ApprovalStatus bidang ReviewStatus untuk lebih mencerminkan tujuannya. ApprovalStatus Selain itu, kami memperkenalkan status baru, Pending Submission, untuk menunjukkan peluang draf. Sekarang ReviewStatus akan menerima nilai-nilai berikut: Pengajuan Tertunda, Dikirim, Dalam Tinjauan, Disetujui, Ditolak, dan Tindakan yang Diperlukan. Status Draft sebelumnya akan diganti dengan Pending Submission. • [Breaking] Pengenalan SubmitOpportunity Aksi: Tindakan baru SubmitOpportunity,, telah diperkenalkan. Tidak seperti perilaku saat ini di mana CreateOpportunity juga mengirimkan kesempatan, Anda sekarang dapat membuat peluang dalam status Pending Submission tanpa segera menautkan Solusi atau AWS Produk.	2024-06-21

Ubah	Deskripsi	Date
	Untuk menyelesaikan pengiriman, gunakan AssociateOpportunity tindakan untuk menautkan solusi atau produk, diikuti oleh SubmitOpportunity. Pemisahan ini memberikan kinerja respons API yang lebih baik dan fase persiapan yang fleksibel sebelum pengiriman. • [Melanggar] Persyaratan Parameter Katalog dan Pengakhiran Titik Akhir Kotak Pasir: Titik akhir 'gamma' (partnercentral-selling-gamma.us-east-1.amazonaws.com) akan menjadi tidak tersedia setelah 7/30/2024. Semua tindakan API sekarang memerlukan parameter Catalog untuk menentukan apakah operasi dilakukan di Sandbox atau AWS katalog. Aturan notifikasi sekarang akan memfilter pada katalog, bukan EnvironmentName. • [Breaking] Mengganti nama OpportunityIdentifier menjadi Identifier: Actions AssignOpportunity, AcceptOpportunity,	

Ubah	Deskripsi	Date
	dan RejectOpportunity sekarang ambil Identifie r alih-alih tetap konsisten OpportunityIdentifier dengan. UpdateOpportunity • Perubahan dari tipe Enum ke String untuk APNProgram CustomerUseCase, dan UseCase: Kami akan mengonversi Project.APNPrograms, Marketing.ActivityUseCases, dan Project.CustomerUseCase dari jenis enumerasi ke tipe string untuk mengurangi risiko yang terkait dengan perubahan nilai. Bidang ini akan menerima nilai hanya dari daftar yang telah ditentukan tetapi tidak tersedia secara asli di SDK. • Penanganan Kesalahan yang Ditingkatkan: Penanganan kesalahan dalam API telah ditingkatkan secara signifikan untuk memfasilitasi proses debug yang lebih mudah dan penyelesaian masalah yang lebih cepat. Kesalahan baru menangani kesalahan struktur dalam dua tahap: 1. Permintaan Validasi - Gagal: Memeriksa tipe	

Ubah	Deskripsi	Date
	dan format data, atau 2. Validasi Bisnis Gagal: Semua masalah dalam muatan akan dicantumkan dalam satu respons, menentukan bidang yang memiliki kesalahan. Umpan balik terkonsolidasi ini memungkinkan Anda memecahkan masalah dan memperbaiki kesalahan dengan lebih efisien. Kode kesalahan dan format telah distandarisasi. • Client Source Identifier: Kemampuan untuk mitra termasuk X-Amzn-User-Agent header dalam permintaan mereka, yang membantu mengidentifikasi sumber lalu lintas. Gunakan format [Nama Solusi termasuk penyedia solusi] [Versi] Misalnya: Konektor CRM AWS Mitra v2.0.1 • Perbaiki bug, sudah diterapkan, tidak ada perubahan pada SDK] Pembuatan Peluang dan ID AWS Akun: Mitra sekarang dapat meluncurkan peluang sambil mengubah ID AWS Akun dari nol ke nilai yang	

Ubah	Deskripsi	Date
	valid tanpa mengalami kesalahan. Sebelumnya a, bug mencegah mitra melakukan tindakan ini secara terpisah. • Perbaikan bug, sudah diterapkan, tidak ada perubahan pada SDK] Asosiasi Solusi dengan Peluang Baru: Mitra dapat mengaitkan solusi dengan peluang yang baru dibuat. Sebelumnya, ketika mitra menciptakan peluang baru dan secara bersamaan mengaitkan solusi, informasi solusi hilang. Masalah ini telah diselesaikan. • Perbaikan bug, sudah diterapkan, tidak ada perubahan ke SDK] Tampilan Detail Pemilik Peluang: Masalah di mana beberapa peluang menampilkan detail pemilik dalam bidang gabungan "LastName" alih-alih bidang "Email", "FirstName", dan "yang diharapkan telah diperbaiki. LastName • Perbaikan bug, sudah diterapkan, tidak ada perubahan ke SDK] Customer.Account.W	

Ubah	Deskripsi	Date
	ebsiteUrl Bidang Opsional: Customer.Account.W ebsiteUrl Bidang telah dibuat opsional ketika AccessControls.Nat ionalSecurity bidang “Ya” sehingga menyelara skan perilaku API dengan perilaku UI yang ada. Jika AccessControls.Nat ionalSecurity bidang “Tidak” atau nol, WebsiteUrl bidang akan diperlukan sebagai validasi bisnis. • Perbaikan bug, sudah diterapkan, tidak ada perubahan pada SDK] Persyaratan ID AWS Akun untuk Mitra Konsultas i: Ketidakkonsistenan di UI yang menunjukk an bidang ID AWS Akun sebagai opsional untuk Mitra Konsultasi telah diselesaikan. Sekarang bidang wajib bersyarat, sesuai dokumentasi. • Perbaikan bug, Untuk diterapkan 6/19, tidak ada perubahan ke SDK] Tahap Co-sell Peluang untuk Menutup Hilang: Mitra sekarang dapat berhasil memperbarui Tahap Co-	

Ubah	Deskripsi	Date
	sell Peluang untuk Menutup Hilang, selama Alasan Hilang Tertutup disediakan. Ini menyelesaikan kesalahan sebelumnya yang menyatakan “Missing Required Field: Closed Lost Reason is required when close the opportunity.” • Dokumentasi] Perbaikan dokumentasi: Membuat beberapa perbaikan pada dokumentasi untuk mencerminkan perubahan. • Masalah yang diketahui dengan perbaikan yang tertunda dan daftar fitur yang akan datang: • Saat melakukan AssociateOpportunity tindakan dengan Solusi atau Produk, pesan kesalahan dari Penawaran ditampilkan secara tidak benar. • Kemampuan untuk menciptakan peluang dari jarak jauh dan mensimulasikan AWS proses validasi di Sandbox • Kode Pos mengembalikan null ketika itu dari format XXXXX-XXXX untuk AS.	

Ubah	Deskripsi	Date
	• Langkah Berikutnya dan Riwayat Langkah Berikutnya tidak tersedia untuk peluang yang ditandai sebagai “Tidak perlu dukungan dari AWS.” • Tidak dapat memperoleh peluang historis dengan deskripsi Masalah Bisnis Pelanggan lebih dari 255 karakter. • Kemampuan untuk menghapus peluang yang tertunda pengiriman. • LeadSource tidak tersedia dalam model data.	

Ubah	Deskripsi	Date
Beta 5	• Berubah LifeCycle.Approval Status menjadi LifeCycle.ReviewStatus • Berubah Insights.ReviewComments menjadi LifeCycle.ReviewComments • Ditambahkan LifeCycle.ReviewStatusReason (atribut baru) • Menambahkan Nilai Baru “Tertunda” ke PartnerAcceptance.Status • Dirilis SDK untuk dotnet, java, javascript, python, ruby. • Panduan Referensi API yang diperbarui untuk mencerminkan perubahan di atas.	2024-04-19

Ubah	Deskripsi	Date
Beta 4	• Menambahkan Pengujian Kotak Pasir: Kemampuan untuk menguji pemberitahuan Acara Peluang di Kotak Pasir. • Dokumentasi Ditambahkan: Memperkenalkan aturan contoh pemberitahuan Acara Peluang. • Rilis SDK yang Ditambahkan: Dirilis SDK untuk Java (V2), Javascript (V2), DotNet (V3), dan Python (V3). • Format Tanggal Diperbarui: Diperbarui untuk mengikuti standar ISO.	2024-03-04

Ubah	Deskripsi	Date
Beta 3	• Menambahkan File Log Perubahan: Memperkenalkan file log perubahan untuk melacak pembaruan dan perubahan yang lebih baik. • Panduan Referensi API yang Diperbarui (PDF): Merilis versi baru Panduan Referensi API dalam format PDF. • Panduan Referensi API yang Diperbarui (Versi Web): Memperbarui versi web Panduan Referensi API. • Menambahkan DotNet SDK: Merilis DotNet SDK, memperluas dukungan kami untuk lingkungan pemrograman yang berbeda. • Diperbarui Python Boto 3 SDK (V2): Dirilis Versi 2 dari Python Boto 3 SDK, memperkenalkan fitur dan peningkatan baru.	2024-01-24

Ubah	Deskripsi	Date
Beta 2	Panduan Referensi API yang Ditambahkan (Versi Web): Merilis versi baru Panduan Referensi API (Versi Web). Catatan: Versi ini memerlukan server web lokal untuk dijalankan setelah ekstraksi untuk fungsionalitas yang tepat.	2024-01-07
Beta 1	- Menambahkan Boto3 (Rilis Awal): Meluncurkan rilis awal Boto3, SDK Python untuk layanan kami. - Menambahkan Panduan Program Beta: Memperkenalkan panduan untuk peserta dalam program pengujian Beta kami. - Panduan Referensi API yang Ditambahkan (PDF): Mengunggah versi pertama Panduan Referensi API kami dalam format PDF, menyediakan dokumentasi komprehensif untuk API kami.	2023-12-15
Beta 0	Rilis awal ini menyediakan serangkaian fungsi untuk mengelola dan mengoptimalkan keterlibatan dan peluang mitra di AWS Partner Central.	2023-11-15

Riwayat dokumen

Berikut ini adalah daftar rilis dokumentasi untuk referensi API ini.

Perubahan	Deskripsi	Tanggal
Dokumentasi tambahan untuk API Pengukuran Pendapatan Mitra	AWS Pusat mitra menambahkan API baru untuk Pengukuran Pendapatan Mitra (PRM).	Juni 30, 2026
Dokumentasi yang diperbarui untuk ACE Resonate	AWS Respons API yang diperbarui pusat mitra untuk GetAwsOpportunitySummary tindakan guna mendukung Cosell Motion.	Juni 16, 2026
Ditambahkan dokumentasi untuk Mitra Central MCP Server	AWS Agen pusat mitra MCP server menyediakan alat sentral mitra melalui Model Context Protocol (MCP).	Maret 16, 2026
Rilis RC 5	Kandidat versi 5 dari API AWS Partner Central dirilis. Memperkenalkan API Akun baru ke AWS Partner Central, memungkinkan mitra untuk mengelola informasi akun, pendaftaran, profil, manajemen domain, dan koneksi mitra mereka. Memperkenalkan API Benefits baru ke AWS Partner Central, memungkinkan mitra untuk menemukan, mengajukan, dan mengelola manfaat di seluruh Program AWS Mitra melalui antarmuka terpusat.	November 30, 2025

Tindakan keterlibatan API
Penjualan yang disempurnakan untuk mendukung alur kerja manajemen prospek. Menambahkan panduan penggunaan komprehensif dan dokumentasi API untuk Account, Benefits API dengan alur kerja terperinci dan pola integrasi. Dokumentasi API Penjualan yang disempurnakan untuk manajemen Keterlibatan dengan konteks Lead.

Rilis RC 4

Kandidat versi 4 dari API AWS Partner Central dirilis. Memperkenalkan API Saluran untuk mengelola hubungan saluran dan kolaborasi mitra-ke-mitra, memungkinkan mitra memenuhi syarat untuk mendapatkan manfaat program saat bertransaksi dengan pelanggan akhir menggunakan Transfer Penagihan. Dokumentasi diatur ulang untuk mendukung beberapa API Pusat AWS Mitra.

November 19, 2025

Ditambahkan dokumentasi untuk peluang multi-mitra	Multi-partner Peluang (dalam pratinjau) adalah pengalaman terintegrasi di AWS Partner Central yang memungkinkan AWS Mitra untuk menemukan dan terhubung dengan Mitra lain dan menjual bersama pada penawaran pelanggan.	Desember 4, 2024
Rilis RC 3	Kandidat versi 3 dari API AWS Partner Central dirilis	Oktober 17, 2024
Rilis RC 2	Versi kandidat 2 dari API AWS Partner Central dirilis	Agustus 7, 2024
Rilis RC 1	Kandidat versi 1 dari API AWS Partner Central dirilis	Juni 21, 2024
Rilis beta 5	Beta 5 dari AWS Partner Central API dirilis	April 19, 2024
Rilis beta 4	Beta 4 dari AWS Partner Central API dirilis	Maret 4, 2024
Rilis beta 3	Beta 3 dari AWS Partner Central API dirilis	Januari 24, 2024
Rilis beta 2	Beta 2 dari AWS Partner Central API dirilis	Januari 7, 2024
Rilis beta 1	Beta 1 dari AWS Partner Central API dirilis	15 Desember 2023
Rilis beta 0	AWS Rilis awal API Partner Central	15 November 2023