ユーザーガイド

AWS Entity Resolution

AWS Entity Resolution: ユーザーガイド

Table of Contents

とは AWS Entity Resolution

AWS Entity Resolution は、複数のアプリケーション、チャネル、データストアに保存されている関連レコードを照合、リンク、強化するのに役立つサービスです。柔軟でスケーラブルで、既存のアプリケーションやデータサービスプロバイダーに接続できるエンティティ解決ワークフローの使用を開始できます。

AWS Entity Resolution は、ルールベースのマッチング、機械学習ベースのマッチング (ML マッチング) 、データサービスプロバイダー主導のマッチングなどの高度なマッチング手法を提供します。これらの手法は、顧客情報、製品コード、またはビジネスデータコードの関連レコードをより正確にリンクおよび強化するのに役立ちます。

を使用して AWS Entity Resolution 、最近のイベント (広告クリック、カートの中止、購入など) をデータサービスプロバイダーからの仮名化されたシグナルと一意のエンティティ ID にリンクすることで、カスタマーインタラクションの統合ビューを作成できます。ストア間で異なるコード (SKU、UPC など) を使用する製品をより適切に追跡することもできます。を使用すると AWS Entity Resolution 、データの移動を最小限に抑えながら、マッチングの精度を制御し、データセキュリティをより適切に保護できます。

トピック

初めての AWS Entity Resolution ユーザーですか?

を初めて使用する場合は AWS Entity Resolution、まず以下のセクションを読むことをお勧めします。

の機能 AWS Entity Resolution

AWS Entity Resolution には次の機能が含まれています。

- 柔軟でカスタマイズ可能なデータ準備

  AWS Entity Resolution は からデータを読み取り AWS Glue 、一致処理の入力として使用します。最大 20 個のデータ入力を指定できます。 は、データ入力テーブルの各行をレコードとして AWS Entity Resolution 処理し、一意のエンティティをプライマリキーとして使用します。 は暗号化されたデータセットで動作 AWS Entity Resolution できます。まず、 のスキーママッピングを定義 AWS Entity Resolution して、一致するワークフローで使用する入力フィールドを理解します。既存の AWS Glue データ入力から独自のデータスキーマまたはブループリントを取り込むことができます。または、インタラクティブユーザーインターフェイスまたは JSON エディタを使用してカスタムスキーマを構築することもできます。デフォルトでは、 は一致する前にデータ入力 AWS Entity Resolution を正規化し、特殊文字や余分なスペースの削除、テキストの小文字へのフォーマットなど、一致処理を改善します。データ入力がすでに正規化されている場合は、正規化をオフにできます。また、GitHub ライブラリも用意されています。これを使用して、ニーズに合わせてデータ正規化プロセスをさらにカスタマイズできます。

- 設定可能なエンティティマッチングワークフロー

  エンティティマッチングワークフローは、データ入力を照合 AWS Entity Resolution する方法と統合データ出力をどこに書き込むかを指定するように設定した一連のステップです。エンティティ解決や ML エクスペリエンスなしで、1 つ以上のマッチングワークフローを設定して、異なるデータ入力を比較し、ルールベースのマッチング、機械学習マッチング、データサービスプロバイダー主導マッチングなど、さまざまなマッチング手法を使用できます。リソース番号、処理されたレコード数、見つかった一致の数など、既存の一致ワークフローとメトリクスのジョブステータスを表示することもできます。

  - Ready-to-use ルールベースのマッチング

    このマッチング手法には、 または AWS Command Line Interface () ready-to-use一連のルールが含まれますAWS CLI。 AWS マネジメントコンソール これらのルールを使用して、入力フィールドに基づいて関連レコードを検索できます。各ルールの入力フィールドの追加または削除、ルールの削除、ルールの優先度の再配置、新しいルールの作成によって、ルールをカスタマイズすることもできます。ルールをリセットして元の構成に戻すこともできます。Amazon Simple Storage Service (Amazon S3) バケットのデータ出力には、ルールベースのマッチング手法を使用して が AWS Entity Resolution 生成する一致グループがあります。各一致グループには、一致を理解するのに役立つように、関連付けられた一致を生成するために使用されるルール番号が

あります。たとえば、ルール番号は、ルール 1 がルール 2 よりも正確になるように、各一致グループの精度を示すことができます。

- 事前設定された機械学習ベースのマッチング (ML マッチング)

  このマッチング手法には、すべてのデータ入力、特にコンシューマーベースのレコードの一致を見つけるための事前設定された ML モデルが含まれています。このモデルは、名前、E メールアドレス、電話番号、住所、生年月日のデータ型に関連付けられたすべての入力フィールドを使用します。モデルは、他の一致グループと比較した一致の品質を説明する各グループの信頼スコアを含む関連レコードの一致グループを生成します。このモデルは、欠落している入力フィールドを考慮し、レコード全体をまとめて分析してエンティティを表します。Amazon S3 バケットのデータ出力には、ML マッチングを使用して が AWS Entity Resolution 生成する一致グループがあります。これは、各一致グループの関連する信頼スコアが 0.0 ~ 1.0 の場合で、一致の精度を示します。

- レコードとデータサービスプロバイダーの照合

  AWS Entity Resolution を使用すると、主要なデータサービスベンダーやライセンスデータセットとレコードを照合、リンク、強化して、顧客を理解し、到達し、サービスを提供する能力を拡張できます。たとえば、データに属性を追加してレコードを強化したり、ビジネス目標を達成するために連携するシステムとプラットフォームの相互運用性を改善したりできます。この一致するワークフローを数回クリックするだけで使用できるため、複雑な独自の統合を構築して維持する必要がなくなります。このマッチング手法を利用するには、これらのデータサービスプロバイダーとのライセンス契約が必要です。

- 手動一括処理と自動増分処理

  データ処理を使用すると、データ入力を、エンティティマッチングワークフロー設定を使用して生成された共通の一致 ID を持つ同様のレコードを持つ統合データ出力テーブルに変換できます。API および AWS マネジメントコンソール または を使用すると AWS CLI、既存の抽出、変換、ロード (ETL) データパイプラインに基づいて、オンデマンドで手動一括処理を実行できます。ETL データパイプラインは、新しいマッチングと既存のマッチングの更新のためにすべてのデータを再処理します。また、ルールベースのマッチングシナリオでは、自動増分処理を開始して、Amazon S3 バケットで新しいデータが利用可能になるとすぐに、サービスはそれらの新しいレコードを読み取り、既存のレコードと比較できます。これにより、Amazon S3 データの変更との一致が最新の状態になります。

- ほぼリアルタイムのルックアップ

  AWS Entity Resolution GetMatchId API オペレーションを使用してエンティティフィールドを検索すると、既存の一致 ID を同期的に取得できます。さまざまなソースとチャネルを通じて取得さ

れた個人を特定できる情報 (PII) 属性 AWS Entity Resolution を使用して を呼び出すことができます。 はデータ保護のためにこれらの属性を AWS Entity Resolution ハッシュし、対応する一致 ID を取得して、顧客をリンクして一致させます。たとえば、関連付けられた名前、E メール、および郵送先住所を含むウェブサインアップを取得できます。GetMatchId API オペレーションを使用して AWS Entity Resolution 、この顧客またはエンティティが S3 バケットに保存されている一致した結果に既に存在するかどうか、およびそれに関連付けられている対応するエンティティー致 ID を確認します。エンティティー致 ID を取得したら、顧客関係管理 (CRM) や顧客データプラットフォーム (CDP) システムなど、それに関連付けられたトランザクション情報をソースアプリケーションで見つけることができます。

- データ保護と設計によるリージョン化

  AWS Entity Resolution は、データを保護するのに役立つデフォルトの暗号化機能を提供し、サービスへのデータ入力ごとに暗号化キーを提供します。たとえば、 AWS Entity Resolution では、サーバー側の暗号化データとハッシュ化されたデータを使用してルールベースのマッチングワークフローを柔軟に実行できます。 はリージョン化 AWS Entity Resolution をサポートしています。つまり、一致するワークフローを実行して、サービスを使用している AWS リージョン のと同じ でデータを処理します。他のアプリケーションで解決済みのデータを使用する前に、Amazon S3 のデータ出力を暗号化してハッシュ化することもできます。

- マルチパーティートランスコーディング

  AWS Entity Resolution は、 などのデータコラボレーションを使用する複数の当事者間でデータソースと一致する設定を定義するのに役立ちます AWS Clean Rooms。

関連サービス

以下は AWS のサービス 、 に関連しています AWS Entity Resolution。

- Amazon S3

  Amazon S3 AWS Entity Resolution に取り込むデータを保存します。

  詳細については、「Amazon Simple Storage Service ユーザーガイド」の「Amazon S3 とは」を参照してください。

- AWS Glue

  で使用するテーブルを Amazon S3 のデータ AWS Glue から作成します AWS Entity Resolution。

詳細については、「 AWS Glue デベロッパーガイド」の「What is AWS Glue?」を参照してください。

- AWS CloudTrail

  CloudTrail ログ AWS Entity Resolution で を使用して、アクティビティの分析 AWS のサービス を強化します。

  詳細については、「を使用した AWS Entity Resolution API コールのログ記録 AWS CloudTrail」を参照してください。

- CloudFormation

  次のリソースを作成します CloudFormation。AWS::EntityResolution::MatchingWorkflow、AWS::EntityResolution::SchemaMapping、および AWS::EntityResolution::PolicyStatement

  詳細については、「を使用して AWS エンティティ解決リソースを作成する AWS CloudFormation」を参照してください。

アクセス AWS Entity Resolution

には、次のオプション AWS Entity Resolution を使用してアクセスできます。

- https://console.aws.amazon.com/entityresolution/ の AWS Entity Resolution コンソールから直接。
- AWS Entity Resolution API を介してプログラムで。詳細については、「AWS Entity Resolution APIリファレンス」を参照してください。
  - Runtime で AWS Entity Resolution API AWS Lambda を呼び出す場合は、独自のデプロイパッケージを作成し、目的のバージョンの AWS SDK ライブラリを含めます。詳細については、AWS Lambda デベロッパーガイドの以下の例を参照してください。
    - .zip または JAR ファイルアーカイブを使用して Java Lambda 関数をデプロイする
    - Python Lambda 関数の .zip ファイルアーカイブの使用

の料金 AWS Entity Resolution

料金については、「AWS Entity Resolution の料金」を参照してください。

セットアップ AWS Entity Resolution

AWS Entity Resolution を初めて使用する前に、 にサインアップ AWS し、ロールを作成する管理者ユーザーを作成します。

にサインアップする AWS

が既にある場合は AWS アカウント、このステップをスキップします。

がない場合は AWS アカウント、次の手順を実行して作成します。

にサインアップするには AWS アカウント

1. https://portal.aws.amazon.com/billing/signup を開きます。
2. オンラインの手順に従います。

   サインアップ手順の一環として、電話またはテキストメッセージを受け取り、電話キーパッドで検証コードを入力します。

   にサインアップすると AWS アカウント、 AWS アカウントのルートユーザー が作成されます。ルートユーザーには、アカウントのすべての AWS のサービス とリソースへのアクセス権があります。セキュリティベストプラクティスとして、ユーザーに管理アクセス権を割り当て、ルートユーザーアクセスが必要なタスクの実行にはルートユーザーのみを使用するようにしてください。

管理者ユーザーの作成

管理者ユーザーを作成するには、以下のいずれかのオプションを選択します。

管理者を管理する方法を 1 つ選択します	目的	方法	以下の操作も可能
IAM アイデンティティセンター内 (推奨)	短期の認証情報を使用して AWSにアクセスします。 これはセキュリティのベストプラクティスと一致しています。ベストプラクティスの詳細については、「IAM ユーザーガイド」の「IAM でのセキュリティのベストプラクティス」を参照してください。	AWS IAM Identity Center ユーザーガイドの「開始方法」の手順に従います。	AWS Command Line Interface ユーザーガイドの を使用する AWS CLI ように を設定 AWS IAM Identity Centerして、プログラムによるアクセスを設定します。
IAM 内 (非推奨)	長期認証情報を使用して AWSにアクセスします。	IAM ユーザーガイドの「緊急アクセス用の IAM ユーザーを作成する」の手順に従います。	IAM ユーザーガイドの「IAM ユーザーのアクセスキーを管理する」の手順に従って、プログラムによるアクセスを設定します。

コンソールユーザーの IAM ロールの作成

AWS Entity Resolution コンソールを使用している場合は、次の手順を実行します。

IAM ロールを作成するには

1. 管理者アカウントを使用して、IAM コンソール (https：//console.aws.amazon.com/iam/) にサインインします。
2. [アクセス管理] で、[ロール] を選択します。

ロールを使用して短期認証情報を作成できます。これはセキュリティを強化するために推奨されます。[ユーザー] を選択して長期間の認証情報を作成することもできます。

3. [ロールの作成] を選択してください。
4. ロールの作成ウィザードで、信頼されたエンティティタイプで、 を選択しますAWS アカウント。
5. このアカウントを選択したまま、次へを選択します。
6. アクセス許可を追加する で、ポリシーの作成 を選択します。

   新しいタブが開きます。

   a. JSON タブを選択し、コンソールユーザーに付与された機能に応じてポリシーを追加します。 は、一般的なユースケースに基づいて次の管理ポリシー AWS Entity Resolution を提供します。

      - AWS マネージドポリシー: AWSEntityResolutionConsoleFullAccess
      - AWS マネージドポリシー: AWSEntityResolutionConsoleReadOnlyAccess

   b. [次へ: タグ] を選択し、タグを追加して (オプション)、[次へ: 確認] を選択します。

   c. [ポリシーの確認] で [名前] と [説明] を入力し、[概要] を確認します。

   d. [ポリシーを作成] を選択します。

      コラボレーションメンバー用のポリシーが作成されました。

   e. 元のタブに戻り、「アクセス許可の追加」の下に、先ほど作成したポリシーの名前を入力します。(ページを再度読み込む必要がある場合があります)。

   f. 作成したポリシーの名前の横にあるチェックボックスをオンにし、次へを選択します。
7. [名前、確認、および作成] で、[ロール名] と [説明] を入力します。

   a. [信頼されたエンティティを選択] を確認し、ロールを引き受ける人物 (複数可) の AWS アカウント を入力します (必要な場合)。

   b. [許可を追加] でアクセス許可を確認し、必要に応じて編集します。

   c. [タグ] を確認し、必要に応じてタグを追加します。

   d. [ロールの作成] を選択してください。

のワークフロージョブロールの作成 AWS Entity Resolution

AWS Entity Resolution はワークフロージョブロールを使用してワークフローを実行します。必要な IAM アクセス許可がある場合には、コンソールを使用してこのロールを作成できます。アクセスCreateRole許可がない場合は、管理者にロールの作成を依頼してください。

のワークフロージョブロールを作成するには AWS Entity Resolution

1. 管理者アカウントを使用して、https://console.aws.amazon.com/iam/ の IAM コンソールにサインインします。
2. [アクセス管理] で、[ロール] を選択します。

   ロールを使用して短期認証情報を作成できます。これはセキュリティを強化するために推奨されます。[ユーザー] を選択して長期間の認証情報を作成することもできます。
3. [ロールの作成] を選択してください。
4. [ロールの作成] ウィザードの [信頼されたエンティティタイプ] で [カスタム信頼ポリシー] を選択します。
5. 次のカスタム信頼ポリシーをコピーして JSON エディタに貼り付けます。

   JSON

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

6. [次へ] をクリックします。
7. アクセス許可を追加する で、ポリシーの作成 を選択します。

新しいタブが表示されます。

a. 次のポリシーをコピーして JSON エディタに貼り付けます。

> **Note**
>
> 次のポリシー例では、Amazon S3 や などの対応するデータリソースを読み取るために必要なアクセス許可をサポートしています AWS Glue。ただし、データソースの設定方法によっては、このポリシーの変更が必要になる場合があります。
> AWS Glue リソースと基盤となる Amazon S3 リソースは、 AWS Glue がサポートされている AWS 商用パーティション内の任意のリージョンから使用できます。 と同じリージョンに存在する必要はありません AWS Entity Resolution。
> データソースが暗号化または復号化されていない場合、アクセス AWS KMS 許可を付与する必要はありません。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:s3:::{{input-buckets}}",
                "arn:aws:s3:::{{input-buckets}}/*"
            ],
            "Condition": {
                "StringEquals": {
                    "s3:ResourceAccount": [
                        "444455556666"
                    ]
                }
            }
```

```
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:s3:::{{output-bucket}}",
                "arn:aws:s3:::{{output-bucket}}/*"
            ],
            "Condition": {
                "StringEquals": {
                    "s3:ResourceAccount": [
                        "444455556666"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "glue:GetDatabase",
                "glue:GetTable",
                "glue:GetPartition",
                "glue:GetPartitions",
                "glue:GetSchema",
                "glue:GetSchemaVersion",
                "glue:BatchGetPartition"
            ],
            "Resource": [
                "arn:aws:glue:us-east-1:444455556666:database/{{input-
databases}}",
                "arn:aws:glue:us-east-1:444455556666:table/{{input-
database}}/{{input-tables}}",
                "arn:aws:glue:us-east-1:444455556666:catalog"
            ]
        }
    ]
}
```

各 *{{user input placeholder}}* を独自の情報に置き換えます。

aws-region	AWS リージョン of your resources. You can use AWS Glue, Amazon S3, and AWS KMS resources from any commercial same AWS リージョン where these services are supported.
&ExampleAWSAccountNo1;	Your AWS アカウント ID.
######	Amazon S3 buckets which contains the underlying data objects of AWS Glue where `AWS Entity Resolution` will read from.
######	Amazon S3 buckets where `AWS Entity Resolution` will generate the output data.
########	AWS Glue databases where `AWS Entity Resolution` will read from.

b. (オプション) 入力 Amazon S3 バケットが顧客の KMS キーを使用して暗号化されている場合は、以下を追加します。

```
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": [
                "arn:aws:kms:{{aws-
region}}:{{&ExampleAWSAccountNo1;}}:key/{{inputKeys}}"
            ]
        }
```

各 *{{user input placeholder}}* を独自の情報に置き換えます。

aws-region	AWS リージョン of your resources. You can use AWS Glue, Amazon S3, and AWS KMS resources from any commercial same AWS リージョン where these services are supported.
&ExampleAWSAccountNo1;	Your AWS アカウント ID.
inputKeys	Managed keys in AWS Key Managemen t Service. If your input sources are encrypted, `AWS Entity Resolution` must decrypt your data using your key.

c. (オプション) 出力 Amazon S3 バケットに書き込まれるデータを暗号化する必要がある場合は、以下を追加します。

```
        {
            "Effect": "Allow",
            "Action": [
                "kms:GenerateDataKey",
                "kms:Encrypt"
            ],
            "Resource": [
                "arn:aws:kms:{{aws-
region}}:{{&ExampleAWSAccountNo1;}}:key/{{outputKeys}}"
            ]
        }
```

各 *{{user input placeholder}}* を独自の情報に置き換えます。

aws-region	AWS リージョン of your resources. You can use AWS Glue, Amazon S3, and AWS KMS resources from any commercial same AWS リージョン where these services are supported.
&ExampleAWSAccountNo1;	Your AWS アカウント ID.

outputKeys	Managed keys in AWS Key Management Service. If you need your output sources to be encrypted, AWS Entity Resolution must encrypt the output data using your key.

d. (オプション) を通じてプロバイダーサービスにサブスクリプションがあり AWS Data Exchange、プロバイダーのサービスベースのワークフローに既存のロールを使用する場合は、以下を追加します。

```
        {
            "Effect": "Allow",
            "Sid": "DataExchangePermissions",
            "Action": "dataexchange:SendApiAsset",
            "Resource": [
                "arn:aws:dataexchange:{{aws-region}}::data-sets/{{datasetId}}/
revisions/{{revisionId}}/assets/{{assetId}}"
            ]
        }
```

各 *{{user input placeholder}}* を独自の情報に置き換えます。

aws-region	The AWS リージョン where the provider resource is granted. You can find this value in the asset ARN on the AWS Data Exchange console. For example: arn:aws:dataexchange:us-east-2::data-sets/111122223333/revisions/339ffc64444examplef3bc15cf0b2346b/assets/546468b8dexamplea37bfc73b8f79fefa
datasetId	The ID of the dataset, found on the AWS Data Exchange console.
revisionId	The revision of the dataset, found on the AWS Data Exchange console.

assetId	The ID of the asset, found on the AWS Data Exchange console.

8. 元のタブに戻り、「アクセス許可の追加」の下に、先ほど作成したポリシーの名前を入力します。(ページを再度読み込む必要がある場合があります)。
9. 作成したポリシーの名前の横にあるチェックボックスをオンにし、次へを選択します。
10. [名前、確認、および作成] で、[ロール名] と [説明] を入力します。

> **Note**
>
> ロール名は、一致するワークフローを作成するために を渡すことができるメンバーworkflow job roleに付与されたpassRoleアクセス許可のパターンと一致する必要があります。
> たとえば、 AWSEntityResolutionConsoleFullAccess管理ポリシーを使用している場合は、ロール名entityresolutionに を含めることを忘れないでください。

a. [信頼されたエンティティを選択] を確認し、必要に応じて編集します。
b. [許可を追加] でアクセス許可を確認し、必要に応じて編集します。
c. [タグ] を確認し、必要に応じてタグを追加します。
d. [ロールの作成] を選択してください。

のワークフロージョブロール AWS Entity Resolution が作成されました。

入力データテーブルを準備する

では AWS Entity Resolution、各入力データテーブルにソースレコードが含まれています。これらのレコードには、姓、名、E メールアドレス、電話番号などのコンシューマー識別子が含まれます。これらのソースレコードは、同じまたは他の入力データテーブル内で指定した他のソースレコードと照合できます。各レコードには一意のレコード ID (一意の ID) が必要であり、スキーママッピングの作成時にプライマリキーとして定義する必要があります AWS Entity Resolution。

すべての入力データテーブルは、Amazon S3 にバックアップされた AWS Glue テーブルとして使用できます。既に Amazon S3 内にあるファーストパーティデータを使用することも、他のサードパーティー SaaS プロバイダーから Amazon S3 にデータテーブルをインポートすることもできます。Amazon S3 にデータをアップロードした後、 AWS Glue クローラを使用して にデータテーブルを作成できます AWS Glue Data Catalog。その後、データテーブルを入力として使用できます AWS Entity Resolution。

以下のセクションでは、ファーストパーティデータとサードパーティーデータを準備する方法について説明します。

トピック

ファーストパーティ入力データの準備

次の手順では、ルールベースのマッチングワークフロー、機械学習ベースのマッチングワークフロー、または ID マッピングワークフローで使用するファーストパーティデータを準備します。

ステップ 1: ファーストパーティーデータテーブルを準備する

一致するワークフロータイプごとに、成功を確実にするための推奨事項とガイドラインのセットが異なります。

ファーストパーティデータテーブルを準備するには、次の表を参照してください。

ファーストパーティデータテーブルのガイドライン

ワークフロータイプ	必須

高度なルールタイプを使用したルールベースのマッチングワークフロー	• 一意の ID が必要です。
	• 一意の ID は 38 文字以下です。
	• (オプション) ワークフローの処理が完了した AWS Entity Resolution 後に削除するレコードを指定する DELETE 列。列に値がない場合、デフォルト値は *false* です。DELETE 列が *true* に設定されているレコードは削除されます。DELETE 列が *false* または empty に設定されているレコードは、 によって処理されます AWS Entity Resolution。
	
	スキーマには、タイプが でString、 matchKeyと がない DELETE 列が必要ですgroupName 。
	
	Note
	手動処理ケイデンスのアドバンストルールタイプは取り込まれたデータを保存しないため、検索一致 ID (GetMatchID) はサポートされていません。
	
	次の例では、 S1が取り込まれ、S2削除されます。
	
	Example
	
	`sourceID, name, lastName, DELETE`
	`S1, name, lastname, false`
	`S2, name2, lastname2, true`

Simple ルールタイプのルールベースのマッチングワークフロー	• 一意の ID が必要です。 • 一意の ID は 38 文字以下です。
機械学習ベースのマッチングワークフロー	• 一意の ID が必要です。 • データセットには、次のいずれかのタイプが含まれます。 • **Full Name**

ワークフロータイプ	必須
	• **Full Address** • **Full phone** • **Email address** • **Date** – 一致キー名が 生年月日の場合 • どの列名も、MatchId「」、MatchRule 「」、「」、RecordIdSourceId「」、「」の予約名を使用しTargetIdません。

ID マッピングワークフロー	• 一意の ID が必要です。
	• 一意の ID は 257 文字以下です。
	• (オプション) ワークフローの処理が完了した AWS Entity Resolution 後に削除するレコードを指定する DELETE 列。列に値がない場合、デフォルト値は *false* です。DELETE 列が *true* に設定されているレコードは削除されます。DELETE 列が *false* または empty に設定されているレコードは、 によって処理されます AWS Entity Resolution。
	
	スキーマには、タイプが でString、 matchKeyと がない DELETE 列が必要ですgroupName 。
	
	次の例では、 S1が取り込まれ、S2削除されます。
	
	Example
	
	`sourceID, name, lastName, DELETE`
	`S1, name, lastname, false`
	`S2, name2, lastname2, true`

ステップ 2: 入力データテーブルをサポートされているデータ形式で保存する

ファーストパーティー入力データをサポートされているデータ形式で既に保存している場合は、このステップをスキップできます。

を使用するには AWS Entity Resolution、入力データが が AWS Entity Resolution サポートする形式である必要があります。

AWS Entity Resolution は、次のデータ形式をサポートしています。

- カンマ区切り値 (CSV)
- Parquet

ステップ 3: 入力データテーブルを Amazon S3 にアップロードする

Amazon S3 にファーストパーティデータテーブルがすでにある場合は、このステップをスキップできます。

> **Note**
>
> 入力データは、S3resourcesに保存できます。 AWS S3 このデータは、別のリージョンから、または一致するワークフローを実行する AWS アカウント ときにアクセスできます。

入力データテーブルを Amazon S3 にアップロードするには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/s3/ で Amazon S3 コンソールを開きます。
2. バケットを選択し、データテーブルを保存するバケットを選択します。
3. [アップロード] を選択し、プロンプトに従います。
4. [オブジェクト] タブを選択し、データが保存されているプレフィックスを表示します。フォルダの名前を書き留めます。

   データテーブルを表示するフォルダを選択できます。

ステップ 4: AWS Glue テーブルを作成する

> **Note**
>
> パーティション AWS Glue テーブルが必要な場合は、「」に進みますステップ 4: パーティション分割された AWS Glue テーブルを作成する。

Amazon S3 の入力データは、 でカタログ化 AWS Glue され、 AWS Glue テーブルとして表される必要があります。Amazon S3 を入力として AWS Glue テーブルを作成する方法の詳細については、「 AWS Glue デベロッパーガイド」の「コンソールでのクローラの使用AWS Glue 」を参照してください。

このステップでは、S3 バケット内のすべてのファイルをクロールし、 AWS Glue AWS Glue テーブルを作成するクローラを にセットアップします。

> **Note**
>
> AWS Entity Resolution は現在、 に登録されている Amazon S3 ロケーションをサポートしていません AWS Lake Formation。

AWS Glue テーブルを作成するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/glue/ で AWS Glue コンソールを開きます。
2. ナビゲーションバーから、[クローラ] を選択します。
3. リストから S3 バケットを選択し、クローラの作成を選択します。
4. クローラープロパティの設定ページで、crawlerName オプションの説明を入力し、次へを選択します。
5. 引き続き [クローラを追加] ページで、詳細を指定します。
6. [IAM ロールの選択] ページで [既存の IAM ロールを選択] を選択し [次へ] 選択します。

   [IAM ロールを作成する] を選択することも、必要に応じて管理者に IAM ロールを作成してもらうこともできます。
7. [このクローラのスケジュールを設定する] で、[頻度] をデフォルト ([オンデマンドで実行]) のままにして、[次へ] を選択します。
8. クローラの出力を設定する で、 AWS Glue データベースを入力し、次へ を選択します。
9. すべての詳細を確認し、完了を選択します。
10. [クローラ] ページで、S3 バケットの横にあるチェックボックスをオンにし、[クローラの実行] を選択します。
11. クローラの実行が完了したら、 AWS Glue ナビゲーションバーでデータベースを選択し、データベース名を選択します。
12. [データベース] ページで、[{データベース名} のテーブル] を選択します。

a. AWS Glue データベース内のテーブルを表示します。
b. テーブルのスキーマを表示するには、特定のテーブルを選択します。
c. AWS Glue データベース名と AWS Glue テーブル名を書き留めます。

これで、スキーママッピングを作成する準備ができました。詳細については、「スキーママッピングの作成」を参照してください。

ステップ 4: パーティション分割された AWS Glue テーブルを作成する

> **Note**
>
> の AWS Glue パーティショニング機能は AWS Entity Resolution 、ID マッピングワークフローでのみサポートされています。この AWS Glue パーティショニング機能を使用すると、で処理する特定のパーティションを選択できます AWS Entity Resolution。
> パーティション AWS Glue テーブルが必要ない場合は、このステップをスキップできます。

パーティション分割された AWS Glue テーブルは、データ構造に新しいフォルダ (1 か月未満の新しい日フォルダなど) を追加すると、 AWS Glue テーブル内の新しいパーティションを自動的に反映します。

でパーティション分割された AWS Glue テーブルを作成するときに AWS Entity Resolution、ID マッピングワークフローで処理するパーティションを指定できます。次に、ID マッピングワークフローを実行するたびに、 AWS Glue テーブル全体のすべてのデータを処理するのではなく、それらのパーティションのデータのみが処理されます。この機能を使用すると、 でより正確で効率的で費用対効果の高いデータ処理が可能になり AWS Entity Resolution、エンティティ解決タスクの管理の制御と柔軟性が向上します。

ID マッピングワークフローでソースアカウントのパーティション AWS Glue テーブルを作成できます。

まず、 で Amazon S3 の入力データをカタログ AWS Glue 化し、テーブルとして AWS Glue 表現する必要があります。Amazon S3 を入力として AWS Glue テーブルを作成する方法の詳細については、「 AWS Glue デベロッパーガイド」の「コンソールでのクローラの使用AWS Glue 」を参照してください。

このステップでは、S3 バケット内のすべてのファイルをクロール AWS Glue し、パーティションテーブルを作成するクローラを にセットアップします AWS Glue 。

> **Note**
>
> AWS Entity Resolution は現在、 に登録されている Amazon S3 ロケーションをサポートしていません AWS Lake Formation。

パーティション分割された AWS Glue テーブルを作成するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/glue/ で AWS Glue コンソールを開きます。
2. ナビゲーションバーから、[クローラ] を選択します。
3. リストから S3 バケットを選択し、クローラの作成を選択します。
4. クローラのプロパティの設定ページで、クローラ名、オプションの説明を入力し、次へを選択します。
5. 引き続き [クローラを追加] ページで、詳細を指定します。
6. [IAM ロールの選択] ページで [既存の IAM ロールを選択] を選択し [次へ] 選択します。

   [IAM ロールを作成する] を選択することも、必要に応じて管理者に IAM ロールを作成してもらうこともできます。
7. [このクローラのスケジュールを設定する] で、[頻度] をデフォルト ([オンデマンドで実行]) のままにして、[次へ] を選択します。
8. クローラの出力を設定する で、 AWS Glue データベースを入力し、次へ を選択します。
9. すべての詳細を確認し、完了を選択します。
10. [クローラ] ページで、S3 バケットの横にあるチェックボックスをオンにし、[クローラの実行] を選択します。
11. クローラの実行が完了したら、 AWS Glue ナビゲーションバーでデータベースを選択し、データベース名を選択します。
12. データベースページのテーブルで、パーティション化するテーブルを選択します。
13. テーブルの概要で、アクションドロップダウンを選択し、テーブルの編集を選択します。

    a. テーブルプロパティで、追加 を選択します。

    b. 新しいキーには、「」と入力します**aerPushDownPredicateString**。

    c. 新しい値には、 と入力します**'<PartitionKey>=<PartitionValue'**。

    d. AWS Glue データベース名と AWS Glue テーブル名を書き留めます。

これで次の作業に進むことができます。

- スキーママッピングを作成し、1 つの の ID マッピングワークフローを作成します AWS アカウント。
- ID 名前空間ソースを作成し、ID 名前空間ターゲットを作成し、2 つの にまたがる ID マッピングワークフローを作成します AWS アカウント。

サードパーティーの入力データの準備

サードパーティーのデータサービスは、既知の識別子と照合できる識別子を提供します。

AWS Entity Resolution は現在、以下のサードパーティーのデータプロバイダーサービスをサポートしています。

データプロバイダーサービス

会社名	使用可能 AWS リージョン	識別子
LiveRamp	米国東部 (バージニア北部) (us-east-1)、米国東部 (オハイオ) (us-east-2)、米国西部 (オレゴン) (us-west-2)	ランプ ID
TransUnion	米国東部 (バージニア北部) (us-east-1)、米国東部 (オハイオ) (us-east-2)、米国西部 (オレゴン) (us-west-2)	TransUnion 個人 ID と世帯 IDs
統合 ID 2.0	米国東部 (バージニア北部) (us-east-1)、米国東部 (オハイオ) (us-east-2)、米国西部 (オレゴン) (us-west-2)	raw UID 2

次の手順では、プロバイダーのサービスベースのマッチングワークフローまたはプロバイダーのサービスベースの ID マッピングワークフローを使用するようにサードパーティーデータを準備する方法について説明します。

トピック

ステップ 1: でプロバイダーサービスをサブスクライブする AWS Data Exchange

を通じてプロバイダーサービスにサブスクリプションがある場合は AWS Data Exchange、次のいずれかのプロバイダーサービスで一致するワークフローを実行して、既知の識別子を任意のプロバイダーと一致させることができます。データは、優先プロバイダーによって定義された一連の入力と照合されます。

でプロバイダーサービスをサブスクライブするには AWS Data Exchange

1. プロバイダーのリストを表示します AWS Data Exchange。次のプロバイダーリストを利用できます。

   - LiveRamp
     - LiveRamp ID の解決
     - LiveRamp のトランスコーディング
   - TransUnion
     - TruAudience アイデンティティ解決とエンリッチメント
   - 統合 ID 2.0
     - 統合 ID 2.0 ID 解決
2. オファータイプに応じて、次のいずれかの手順を実行します。

   - プライベートオファー – プロバイダーと既存の関係がある場合は、AWS Data Exchange 「 ユーザーガイド」の「プライベート製品とオファー」の手順に従って、プライベートオファーを受け入れます AWS Data Exchange。
   - 独自のサブスクリプションを使用する – プロバイダーに既存のデータサブスクリプションがある場合は、AWS Data Exchange 「 ユーザーガイド」の「Bring Your Own Subscription (BYOS) offers」手順に従って BYOS オファーを受け入れます AWS Data Exchange。

3. でプロバイダーサービスをサブスクライブしたら AWS Data Exchange、そのプロバイダーサービスと一致するワークフローまたは ID マッピングワークフローを作成できます。

APIsAWS Data Exchange 「 ユーザーガイド」の「 API 製品へのアクセス」を参照してください。

ステップ 2: サードパーティーのデータテーブルを準備する

各サードパーティーサービスには、マッチングワークフローを成功させるのに役立つ推奨事項とガイドラインのセットがあります。

サードパーティーのデータテーブルを準備するには、次の表を参照してください。

データプロバイダーサービスのガイドライン

プロバイダーサービス	一意の ID が必要ですか?	アクション
LiveRamp	はい	以下のことを確認してください。 • 一意の ID は、独自の仮名識別子または行 ID のいずれかです。 • データ入力ファイルの形式と正規化は、LiveRamp ガイドラインに準拠しています。 一致するワークフローの入力ファイルフォーマットガイドラインの詳細については、LiveRamp ドキュメントの「ADX によるアイデンティティ解決の実行」を参照してください。 ID マッピングワークフローの入力ファイルフォーマットガイドラインの詳細については、LiveRamp ドキュメントの「ADX によるトランスコーディングの実行」を参照してください。
TransUnion	はい	入力ビューの`string`型列が以下であることを確認します。

プロバイダーサービス	一意の ID が必要ですか?	アクション

		• 一意の ID は必須であり、CRM ID、連絡先 ID、ユーザー ID、または任意の一意の ID にすることができます。
		• **Name**
		  • **First Name** は小文字でも大文字でもかまいませんが、ニックネームはサポートされていますが、タイトルとサフィックスは除外する必要があります。
		  • **Last Name** は小文字または大文字で、ミドルネームを除外できます。
		• **Address**
		  • **Street address1** と **Street address1** は、存在する場合は 1 **Full address** 行にまとめられます。
		  • **City** は から分離されています**Full address**。
		  • **Zip** (または **zip plus4**)。スペース、ハイフン、空白などの特殊文字は使用できません。データがない場合は null を使用します。
		  • **State** は、大文字で 2 文字のコードとして指定されます。
		• • **Phone**
		  • **Phone number** は 10 桁で、スペースやハイフンなどの特殊文字は使用できません。
		• **Email addresses** は、プレーンテキストまたは SHA256-hashed小文字の文字列です。
		• **Date of Birth** は yyyy-mm-dd 形式です。

プロバイダーサービス	一意の ID が必要ですか?	アクション

		• **Digital identifiers** (デバイス IDs) には、ハイフン (36 文字長IDs raw デバイス IDs/MAIDs/IFAs) とハイフンなし (32 および 40 文字長のハッシュデバイス IDs/MAIDs/IFAs) の ID を含めることができます。
		  • **IPV4** は、点線の 10 進表記で表される 32 ビット IP アドレスです。例: `192.0.2.1`
		  • **IPV6** は、コロンで区切られた 16 進表記で表される 128 ビット IP アドレスです。例: `2001:db8:0000:0000:0000:0000:0000:0001`
		  • **MAID** (モバイル広告 ID) は、広告目的でモバイルデバイスに割り当てられる一意の英数字の文字列です。MAID は通常 36 文字です。例: `a1b2c3d4-5678-90ab-cdef-EXAMPLE11111`

プロバイダーサービス	一意の ID が必要ですか?	アクション
統合 ID 2.0	はい	以下のことを確認してください。 • 一意の ID をハッシュにすることはできません。 • スキーマでは、両方ではなく、 **Phone number** または のいずれか**Email addresses** が使用されます。 • UID2 は、UID2 生成用の E メールと電話番号の両方をサポートしています。ただし、両方の値がスキーママッピングに存在する場合、ワークフローは出力の各レコードを複製します。1 つのレコードは UID2 生成に E メールを使用し、2 番目のレコードは電話番号を使用します。データに E メールと電話番号が混在していて、出力にこのレコードの重複が必要ない場合は、それぞれに個別のスキーママッピングを使用して個別のワークフローを作成するのが最善の方法です。このシナリオでは、ステップを 2 回実行します。E メールの場合は 1 つのワークフローを作成し、電話番号の場合は別のワークフローを作成します。 **Note** 特定の E メールまたは電話番号は、リクエストを行ったユーザーに関係なく、任意の時点で同じ raw UID2 値になります。 Raw UID2sは、1 年に約 1 回ローテーションされるソルトバケットからソルトを追加することで作成され、それに伴って raw UID2 もローテーションさ

プロバイダーサービス	一意の ID が必要ですか?	アクション
		れます。異なるソルトバケットは年間を通じて異なる時間にローテーションします。 AWS Entity Resolution は現在、ローテーションするソルトバケットと未加工UID2s を追跡しないため、未加工の UID2s毎日再生成することをお勧めします。詳細については、UID2s「増分更新のために UID2 を更新する頻度」を参照してください。

ステップ 3: 入力データテーブルをサポートされているデータ形式で保存する

サポートされているデータ形式でサードパーティーの入力データを既に保存している場合は、このステップをスキップできます。

を使用するには AWS Entity Resolution、入力データが が AWS Entity Resolution サポートする形式である必要があります。

AWS Entity Resolution は、次のデータ形式をサポートしています。

- カンマ区切り値 (CSV)

> **Note**
>
> LiveRamp は CSV ファイルのみをサポートしています。

- Parquet

ステップ 4: 入力データテーブルを Amazon S3 にアップロードする

Amazon S3 にサードパーティーのデータテーブルがすでにある場合は、このステップをスキップできます。

> **Note**
>
> 入力データは、S3 がサポートされている商用パーティションの任意のリージョンの Amazon S3 リソースに保存できます。 AWS このデータは、別のリージョンから、または一致するワークフローを実行する AWS アカウント ときにアクセスできます。

入力データテーブルを Amazon S3 にアップロードするには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/s3/ で Amazon S3 コンソールを開きます。
2. バケットを選択し、データテーブルを保存するバケットを選択します。
3. [アップロード] を選択し、プロンプトに従います。
4. [オブジェクト] タブを選択し、データが保存されているプレフィックスを表示します。フォルダの名前を書き留めます。

   フォルダを選択して、データテーブルを表示できます。

ステップ 5: AWS Glue テーブルを作成する

Amazon S3 の入力データは、 でカタログ化 AWS Glue され、 AWS Glue テーブルとして表される必要があります。Amazon S3 を入力として AWS Glue テーブルを作成する方法の詳細については、「 AWS Glue デベロッパーガイド」の「コンソールでのクローラの使用AWS Glue 」を参照してください。

> **Note**
>
> AWS Entity Resolution はパーティションテーブルをサポートしていません。

このステップでは、S3 バケット内のすべてのファイルをクロールして AWS Glue AWS Glue テーブルを作成するクローラを にセットアップします。

> **Note**
>
> AWS Entity Resolution は現在、 に登録されている Amazon S3 ロケーションをサポートしていません AWS Lake Formation。

AWS Glue テーブルを作成するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/glue/ で AWS Glue コンソールを開きます。
2. ナビゲーションバーから、[クローラ] を選択します。
3. リストから S3 バケットを選択し、[クローラを追加] を選択します。
4. [クローラを追加] ページで [クローラの名前] を入力し、[次へ] を選択します。
5. 引き続き [クローラを追加] ページで、詳細を指定します。
6. [IAM ロールの選択] ページで [既存の IAM ロールを選択] を選択し [次へ] 選択します。

   [IAM ロールを作成する] を選択することも、必要に応じて管理者に IAM ロールを作成してもらうこともできます。
7. [このクローラのスケジュールを設定する] で、[頻度] をデフォルト ([オンデマンドで実行]) のままにして、[次へ] を選択します。
8. クローラの出力を設定する で、 AWS Glue データベースを入力し、次へ を選択します。
9. 詳細を確認し、[完了] を選択します。
10. [クローラ] ページで、S3 バケットの横にあるチェックボックスをオンにし、[クローラの実行] を選択します。
11. クローラの実行が完了したら、 AWS Glue ナビゲーションバーでデータベースを選択し、データベース名を選択します。
12. [データベース] ページで、[{データベース名} のテーブル] を選択します。

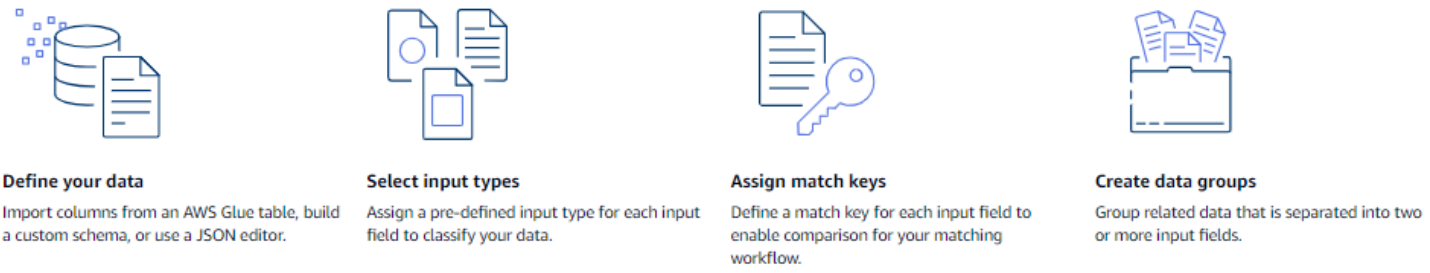
    a. AWS Glue データベース内のテーブルを表示します。
    b. テーブルのスキーマを表示するには、特定のテーブルを選択します。
    c. AWS Glue データベース名と AWS Glue テーブル名を書き留めます。

これで、スキーママッピングを作成する準備ができました。詳細については、「スキーママッピングの作成」を参照してください。

スキーママッピングを使用して入力データを定義する

スキーママッピングは、解決する入力データを定義します。また、列の属性タイプ (入力フィールド) や一致する列など、入力データに関するメタデータも提供します。

スキーママッピングを作成するときは、まず入力フィールドと属性タイプを定義し、次に一致キーとグループ関連データを定義します。次の図は、スキーママッピングを作成する方法をまとめたものです。



スキーママッピングを作成する前に、まずデータテーブルをセットアップ AWS Entity Resolution して準備する必要があります。詳細については、「セットアップ AWS Entity Resolution」および「入力データテーブルを準備する」を参照してください。

スキーママッピングを作成したら、次のいずれかを実行できます。

- 一致するワークフローを作成して、異なるデータ入力間の一致を検索します。
- ID マッピングワークフローで使用できる ID 名前空間ソースを作成し、ソースからターゲットにデータを変換します。
- スキーママッピングをソースとして使用して、同じ 内に ID マッピングワークフローを作成しますAWS アカウント。

トピック

- スキーママッピングの作成
- スキーママッピングのクローン作成
- スキーママッピングの編集
- スキーママッピングの削除

スキーママッピングの作成

この手順では、 AWS Entity Resolution コンソールを使用してスキーママッピングを作成するプロセスについて説明します。

スキーママッピングを作成するには、次の 3 つの方法があります。

- Import from AWS Glueオプションを使用して既存の入力データをインポートする – この作成方法を使用して、ガイド付きフローを使用して AWS Glue テーブルから事前入力された列で始まる入力フィールドを定義します。
- カスタムスキーマの構築オプションを使用して入力データを手動で定義する – この作成方法を使用して、ガイド付きフローを使用して入力フィールドを手動で定義します。
- JSON エディタの使用オプションを使用して手動で作成 – JSON エディタを使用して、既存の入力データを手動で作成、使用、またはインポートします。

> **Note**
>
> このオプションでは、一意の ID フィールドと入力フィールドは使用できません。

Import from AWS Glue

から既存の入力データをインポートしてスキーママッピングを作成するには AWS Glue

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのデータ準備で、スキーママッピングを選択します。
3. スキーママッピングページの右上隅で、スキーママッピングの作成を選択します。
4. ステップ 1: スキーマの詳細を指定するには、次の手順を実行します。
    a. 名前と作成方法には、スキーママッピング名とオプションの 説明を入力します。
    b. 作成方法 で、 からインポート AWS Glueを選択します。
    c. [AWS リージョン] を選択します。
    d. AWS Glue データベースを選択します。
    e. AWS Glue テーブルを選択します。

新しいテーブルを作成するには、 AWS Glue コンソール https://console.aws.amazon.com/glue/ に移動します。詳細については、 AWS Glue ユーザーガイドのAWS Glue 表を参照してください。

f. 一意の ID には、データの各行を個別に参照する列を指定します。

Example

たとえば、**`Primary_key`**、**`Row_ID`**、または **`Record_ID`** などです。

> **Note**
>
> 一意の ID 列は必須です。一意の ID は、単一のテーブル内の一意の識別子である必要があります。ただし、異なるテーブル間で、一意の ID に重複した値を含めることができます。一意の ID が指定されていない、同じソース内で一意ではない、またはソース間で属性名の点で重複している場合、 は一致するワークフローの実行時にレコード AWS Entity Resolution を拒否します。ルールベースのマッチングワークフローでこのスキーママッピングを使用している場合、一意の ID は 38 文字を超えることはできません。

g. 入力フィールドで、マッチングに使用する列とオプションのパススルーに使用する列を選択します。

マッチングとパススルーの両方で最大 34 列を選択できます。

i. 「マッチング」で、マッチングの入力フィールドとして使用する列を選択します。

マッチングには最大 24 列を選択できます。

ii. マッチングに使用されない列を指定する場合は、パススルーする列を追加するを選択します。

iii. (オプション) パススルーで、パススルー列として含める列を選択します。

> **Note**
>
> 機械学習ベースのマッチングワークフローを実行するときは、データ内の列名として、`MatchId「」`、`MatchRule「」`、`RecordId「」`、`SourceId「」`、`「」`、`TargetId「」`のいずれかの予約名を使用しないでください。これらの予

約名のいずれかを使用すると、命名が競合し、ML ベースのマッチングワークフローが失敗します。

h. (オプション) リソースのタグを有効にする場合は、新しいタグを追加を選択し、キーと値のペアを入力します。

i. [次へ] を選択します。

5. ステップ 2: 入力フィールドをマッピングするには、マッチングに使用する入力フィールドとオプションのパススルーに使用する入力フィールドを定義します。

a. 一致する入力フィールドについては、各入力フィールドについて、

- データを分類する属性タイプを指定します。
- 一致キー名を指定して、入力フィールドを一致するワークフローと比較できるようにします。特定の一致キー名は、デフォルトで特定の属性タイプに自動的に関連付けられます。
- その入力フィールドの列値がハッシュされている場合はハッシュチェックボックスをオンにし、値がクリアテキストの場合はチェックボックスを空白のままにします。

> **Note**
>
> LiveRamp プロバイダーのサービスベースのマッチング手法で使用するスキーママッピングを作成する場合は、次のことができます。
>
> - プロバイダー ID の属性タイプを LiveRamp ID として指定します。
> - 名前フィールドの属性タイプを複数のフィールド (名、姓など) または 1 つのフィールドで指定します。
> - 住所フィールドの属性タイプを複数のフィールド (住所 1、住所 2、 など) または 1 つのフィールド (住所全体) で指定します。
>
>   アドレスと照合する場合は、郵便番号 (郵便番号) が必要です。
> - 名前に E メール (E メールアドレス) または電話番号 (電話番号) を含めると、それらのフィールドは住所と照合できます。

> **Note**
>
> TransUnion プロバイダーのサービスベースのマッチング手法で使用するスキーママッピングを作成する場合は、次のいずれかの属性タイプを指定できます。
>
> - フルネーム、名、姓
> - 住所、住所 1、市区町村、都道府県、国、郵便番号
> - Phone number (電話番号)
> - [E メールアドレス]
> - 日付
> - デジタル識別子: IPV4、IPV6、または MAID

> **Note**
>
> 機械学習ベースのマッチングワークフローで使用するスキーママッピングを作成する場合、データセットには次の属性タイプの少なくとも 1 つが含まれている必要があります。
>
> - フルネーム
> - フルアドレス
> - フルフォン
> - [E メールアドレス]
> - 一致キー名が生年月日の日付
>
> これらの属性の属性タイプをカスタム文字列として指定しないでください。

b. (オプション) パススルーの入力フィールドには、一致しない入力フィールドと対応するハッシュステータスを追加します。

   ハッシュステータスは、その入力フィールドの列値がハッシュかクリアテキストかを示します。

c. [次へ] を選択します。

6. ステップ 3: データをグループ化するには、名前、住所、電話番号の入力フィールドを複数のフィールドに分割します。

   このステップでは、関連する入力フィールドを 1 つのフィールドに連結します。これにより、一致するワークフローの 1 つのフィールドとして比較できます。

   名前、住所、または電話番号の入力フィールドにデータがマッピングされていない場合、このセクションは空白になります。

   より多くのタイプのデータがある場合は、さらにグループを追加することもできます。

   a. 名前入力データをグループ化する場合:

      フルネームで、グループ化する入力フィールドを 2 つ以上選択します。

      グループ名と一致キーは、データ型に自動的に関連付けられます。

      グループ名を更新でき、カスタム一致キーで一致キーには、文字、数字、アンダースコア (_)、ハイフン (-) など、最大 255 文字を含めることができます。

      グループの追加 を選択して、別のグループを追加します。

      > **Note**
      >
      > 正規化はフルネームでのみサポートされています。
      > フルネームサブタイプを正規化する場合は、フルネームグループに名、ミドルネーム、姓のサブタイプを割り当てます。

   b. Address 入力データをグループ化する場合:

      フルアドレスで、グループ化する入力フィールドを 2 つ以上選択します。

      グループ名と一致キー。 は自動的にデータ型に関連付けられます。

      グループ名を更新でき、カスタム一致キーには、文字、数字、アンダースコア (_)、ハイフン (-) など、最大 255 文字を含めることができます。

      グループの追加を選択して、別のグループを追加します。

> **Note**
>
> 正規化はフルアドレスでのみサポートされます。
> フルアドレスサブタイプを正規化する場合は、フルアドレスグループに次のサブタイプを割り当てます。住所 1、住所 2: 住所 3 名、市名、州、国、郵便番号。

c. 電話入力データをグループ化する場合:

フルフォンの場合は、グループ化する入力フィールドを 2 つ以上選択します。

グループ名と一致キー。 は自動的にデータ型に関連付けられます。

グループ名を更新でき、カスタム一致キーには、文字、数字、アンダースコア (_)、ハイフン (-) など、最大 255 文字を含めることができます。

グループの追加 を選択して、別のグループを追加します。

> **Note**
>
> 正規化はフルフォンでのみサポートされています。
> 完全な電話サブタイプを正規化する場合は、完全な電話グループに電話番号と電話の国コードのサブタイプを割り当てます。

d. [次へ] を選択します。

7. ステップ 4: 確認して作成するには、次の手順を実行します。

a. 前のステップで行った選択内容を確認し、必要に応じて編集します。

b. スキーママッピングの作成 を選択します。

> **Note**
>
> ワークフローに関連付けた後でスキーママッピングを変更することはできません。既存の設定を使用して新しいスキーママッピングを作成する場合は、スキーママッピングのクローンを作成できます。

スキーママッピングを作成したら、一致するワークフローを作成するか、ID 名前空間を作成する準備が整います。

Build custom schema

カスタムスキーマのビルドオプションを使用してスキーママッピングを作成するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのデータ準備で、スキーママッピングを選択します。
3. スキーママッピングページの右上隅で、スキーママッピングの作成を選択します。
4. ステップ 1: スキーマの詳細を指定するには、次の手順を実行します。
   a. 名前と作成方法には、スキーママッピング名とオプションの 説明を入力します。
   b. 作成方法 で、カスタムスキーマの構築 を選択します。
   c. 一意の ID には、データの各行を識別する一意の ID を入力します。

      Example

      たとえば、**`Primary_key`**、**`Row_ID`**、または **`Record_ID`** などです。

      > **Note**
      >
      > Unique ID 列は必須です。一意の ID は、単一のテーブル内の一意の識別子である必要があります。ただし、異なるテーブル間で、一意の ID に重複した値を含めることができます。一意の ID が指定されていない、同じソース内で一意ではない、またはソース間で属性名の点で重複している場合、 は一致するワークフローの実行時にレコード AWS Entity Resolution を拒否します。ルールベースのマッチングワークフローでこのスキーママッピングを使用している場合、一意の ID は 38 文字を超えることはできません。

   d. (オプション) リソースのタグを有効にする場合は、新しいタグを追加を選択し、キーと値のペアを入力します。
   e. [次へ] を選択します。
5. ステップ 2: 入力フィールドをマッピングするには、マッチングに使用する入力フィールドとオプションのパススルーに使用する入力フィールドを定義します。

   マッチングとパススルーの両方について、合計で最大 34 列を定義できます。

   a. 一致する入力フィールドには、入力フィールドに入力します。

> **Note**
>
> 機械学習ベースのマッチングワークフローを実行するときは、データ内の列名として、MatchId「」、MatchRule「」、RecordId「」、SourceId「」、「」、TargetId「」のいずれかの予約名を使用しないでください。これらの予約名のいずれかを使用すると、命名が競合し、ML ベースのマッチングワークフローが失敗します。

b. 属性タイプを選択してデータを分類します。

> **Note**
>
> LiveRamp プロバイダーのサービスベースのマッチング手法で使用するスキーママッピングを作成する場合は、providerID 属性タイプを LiveRamp ID として指定できます。出力に PII データを含める場合は、属性タイプをカスタム文字列として指定する必要があります。

> **Note**
>
> TransUnion プロバイダーのサービスベースのマッチング手法で使用するスキーママッピングを作成する場合は、次のいずれかの属性タイプを指定できます。
>
> - フルネーム、名、姓
> - 住所、住所 1、市区町村、都道府県、国、郵便番号
> - Phone number (電話番号)
> - [E メールアドレス]
> - 日付
> - デジタル識別子: IPV4、IPV6、または MAID

> **Note**
>
> 機械学習ベースのマッチングワークフローで使用するスキーママッピングを作成する場合、データセットには次の属性タイプの少なくとも 1 つが含まれている必要があります。

- フルネーム
- フルアドレス
- フルフォン
- [E メールアドレス]
- 一致キー名が生年月日の日付

これらの属性の属性タイプをカスタム文字列として指定しないでください。

c. 一致キー名を選択して、入力フィールドを一致するワークフローと比較できるようにします。

特定の一致キー名は、デフォルトで特定の属性タイプに自動的に関連付けられます。

d. その入力フィールドの列値がハッシュされている場合はハッシュされたチェックボックスを選択し、値がクリアテキストの場合はチェックボックスを空白のままにします。

e. 入力フィールドの追加 を選択して、入力フィールドを追加します。

マッチングには、最大 24 個の入力フィールドを追加できます。

f. (オプション) パススルーの入力フィールドには、一致しない入力フィールドと対応するハッシュステータスを追加します。

g. [次へ] を選択します。

6. ステップ 3: データをグループ化するには、名前、住所、電話番号の入力フィールドを複数のフィールドに分割します。

このステップでは、関連する入力フィールドを 1 つのフィールドに連結します。これにより、一致するワークフローで 1 つのフィールドとして比較できます。

名前、住所、電話番号の入力フィールドにマッピングされたデータがない場合、このセクションは空白になります。

より多くのタイプのデータがある場合は、さらにグループを追加することもできます。

a. 名前入力データをグループ化する場合:

フルネームで、グループ化する入力フィールドを 2 つ以上選択します。

グループ名と一致キーは、データ型に自動的に関連付けられます。

グループ名を更新でき、カスタム一致キーには、文字、数字、アンダースコア (_)、ハイフン (-) など、最大 255 文字を含めることができます。

グループの追加を選択して、別のグループを追加します。

> **Note**
>
> 正規化はフルネームでのみサポートされています。
> フルネームサブタイプを正規化する場合は、フルネームグループに名、ミドルネーム、姓のサブタイプを割り当てます。

b. Address 入力データをグループ化する場合:

フルアドレスで、グループ化する入力フィールドを 2 つ以上選択します。

グループ名と一致キー。 は自動的にデータ型に関連付けられます。

グループ名を更新でき、カスタム一致キーには、文字、数字、アンダースコア (_)、ハイフン (-) を含む最大 255 文字を含めることができます。

グループの追加 を選択して、別のグループを追加します。

> **Note**
>
> 正規化はフルアドレスでのみサポートされています。
> フルアドレスサブタイプを正規化する場合は、フルアドレスグループに次のサブタイプを割り当てます。住所 1、住所 2: 住所 3 名、市名、州、国、郵便番号。

c. 電話入力データをグループ化する場合:

フルフォンの場合は、グループ化する入力フィールドを 2 つ以上選択します。

グループ名と一致キー。 は自動的にデータ型に関連付けられます。

グループ名を更新でき、カスタム一致キーで一致キーには、文字、数字、アンダースコア (_)、ハイフン (-) など、最大 255 文字を含めることができます。

グループの追加を選択して、別のグループを追加します。

> **Note**
>
> 正規化はフルフォンでのみサポートされています。
> 完全な電話サブタイプを正規化する場合は、完全な電話グループに電話番号と電話の国コードのサブタイプを割り当てます。

d. [次へ] を選択します。

7. ステップ 4: 確認して作成するには、以下を実行します。

a. 前のステップで行った選択内容を確認し、必要に応じて編集します。

b. スキーママッピングの作成 を選択します。

> **Note**
>
> ワークフローに関連付けた後でスキーママッピングを変更することはできません。既存の設定を使用して新しいスキーママッピングを作成する場合は、スキーママッピングのクローンを作成できます。

スキーママッピングを作成したら、一致するワークフローを作成するか、ID 名前空間を作成する準備が整います。

Use JSON editor

JSON エディタを使用してスキーママッピングを作成するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのデータ準備で、スキーママッピングを選択します。
3. スキーママッピングページの右上隅で、スキーママッピングの作成を選択します。
4. ステップ 1: スキーマの詳細を指定するには、次の手順を実行します。

a. 名前と作成方法には、スキーママッピング名とオプションの 説明を入力します。

b. 作成方法 で、JSON エディタを使用する を選択します。

c. (オプション) リソースのタグを有効にする場合は、新しいタグを追加を選択し、キーと値のペアを入力します。

d. [次へ] を選択します。

5. ステップ 2: マッピングを指定するには:

   a. JSON エディタでスキーマの構築を開始するか、目標に基づいて次のいずれかのオプションを選択します。

目標	推奨されるオプション
スキーママッピングの構築を開始する	サンプル JSON を挿入し、必要に応じて情報を編集します。
既存の JSON ファイルを使用する	ファイルからインポート

> **Note**
>
> 正規化は、、NAME、ADDRESS、PHONEおよび のタイプでのみサポートされますEMAIL_ADRESS。
> NAME サブタイプを正規化する場合は、NAMEgroupName に次のサブタイプを割り当てます: NAME_FIRST、NAME_MIDDLE、および NAME_LAST
> ADDRESS サブタイプを正規化する場合は、、、ADDRESS_STREET1、、ADDRESS_STREET2、ADDRESS_STREET3ADDRESS_CIよび のサブタイプを ADDRESS groupName に割り当てますADDRESS_POSTALCODE。
> PHONE サブタイプを正規化する場合は、PHONEgroupName に次のサブタイプを割り当てます: PHONE_NUMBERおよび PHONE_COUNTRYCODE。

   b. [次へ] を選択します。

6. ステップ 3: 確認して作成する:

   a. 前のステップで行った選択内容を確認し、必要に応じて編集します。

   b. スキーママッピングの作成 を選択します。

> **Note**
>
> ワークフローに関連付けた後でスキーママッピングを変更することはできません。既存の設定を使用して新しいスキーママッピングを作成する場合は、スキーママッピングのクローンを作成できます。

スキーママッピングを作成したら、一致するワークフローを作成するか、ID 名前空間を作成する準備が整います。

スキーママッピングのクローン作成

既存の設定を使用して新しいスキーママッピングを作成する場合は、スキーママッピングのクローンを作成できます。

スキーママッピングのクローンを作成するには:

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのデータ準備で、スキーママッピングを選択します。
3. スキーママッピングを選択します。
4. [クローンを作成] を選択します。
5. スキーマの詳細を指定ページで、必要な変更を加え、次へを選択します。
6. 一致する手法の選択ページで、必要な変更を加え、次へを選択します。
7. マップ入力フィールドページで、必要な変更を加え、次へを選択します。
8. グループデータページで、必要な変更を加え、次へを選択します。
9. 確認および保存ページで、必要な変更を加え、スキーママッピングのクローンを選択します。

スキーママッピングの編集

スキーママッピングは、ワークフローに関連付ける前にのみ編集できます。ワークフローにスキーママッピングを関連付けた後は、編集できません。既存の設定を使用して新しいスキーママッピングを作成する場合は、スキーママッピングのクローンを作成できます。

スキーママッピングを編集するには:

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのデータ準備で、スキーママッピングを選択します。
3. スキーママッピングを選択します。
4. [編集] を選択します。
5. スキーマの詳細を指定ページで、必要な変更を加え、次へを選択します。

6. 一致する手法の選択ページで、必要な変更を加え、次へを選択します。
7. Map input fields ページで、必要な変更を加え、Next を選択します。
8. グループデータページで、必要な変更を加え、次へを選択します。

> **Note**
>
> 正規化は、フルネーム、フルアドレス、フルフォン、E メールアドレスでのみサポートされています。
> フルネームサブタイプを正規化する場合は、フルネームグループに名、ミドルネーム、姓のサブタイプを割り当てます。
> フルアドレスサブタイプを正規化する場合は、フルアドレスグループに次のサブタイプを割り当てます。住所 1、住所 2: 住所 3 の名前、市名、州、国、郵便番号。
> 完全な電話サブタイプを正規化する場合は、完全な電話グループに電話番号と電話の国コードのサブタイプを割り当てます。

9. 確認および保存ページで、必要な変更を加え、スキーママッピングの編集を選択します。

スキーママッピングの削除

一致するワークフローに関連付けられているスキーママッピングは削除できません。スキーママッピングを削除する前に、関連するすべての一致するワークフローからスキーママッピングを削除する必要があります。

スキーママッピングを削除するには:

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのデータ準備で、スキーママッピングを選択します。
3. スキーママッピングを選択します。
4. [削除] を選択します。
5. 削除を確定し、[削除] を選択します。

ID 名前空間を使用して入力データを定義する

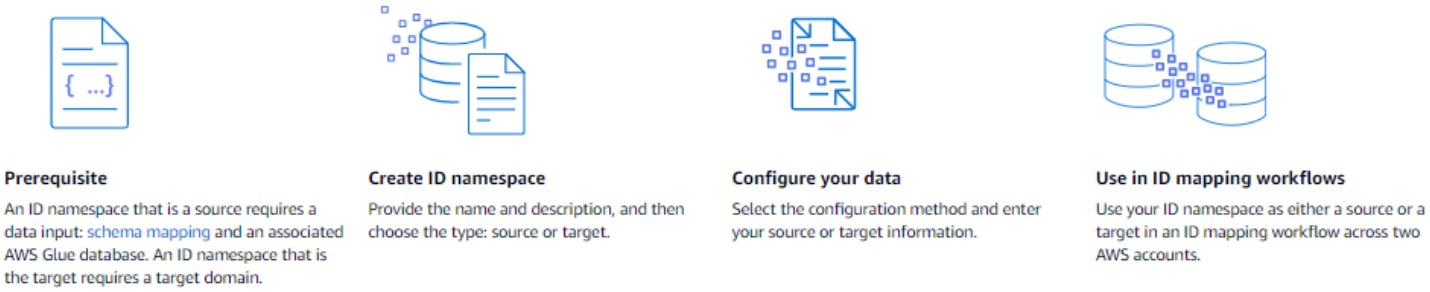
ID 名前空間は、入力データテーブルを囲むラッパーです。ID 名前空間を使用して、入力データとマッチング手法、および ID マッピングワークフローでそれらを使用する方法を説明するメタデータを提供します。

ID 名前空間には、[ソース] と [ターゲット] の 2 種類があります。

- ソースには、ID マッピングワークフローで が AWS Entity Resolution 処理するソースデータの設定が含まれています。
- ターゲットには、すべてのソースが解決するターゲットデータの設定が含まれます。

ID マッピングワークフロー AWS アカウント の 2 つの間で解決する入力データを定義できます。1 人の参加者が ID 名前空間ソースを作成し、別の参加者が ID 名前空間ターゲットを作成します。参加者がソースとターゲットを作成したら、ID マッピングワークフローを実行して、ソースからターゲットにデータを変換できます。

次の図は、ID マッピングワークフローで使用する ID 名前空間を作成する方法をまとめたものです。



以下のセクションでは、ID 名前空間ソースと ID 名前空間ターゲットを作成する方法について説明します。

トピック

- ID 名前空間ソース
- ID 名前空間ターゲット
- ID 名前空間の編集
- ID 名前空間の削除
- ID 名前空間のリソースポリシーの追加または更新

ID 名前空間ソース

ID 名前空間ソースは、ID マッピングワークフロー内のデータのソースです。

ID 名前空間ソースを作成する前に、ユースケースに応じて、まずスキーママッピングまたは一致するワークフローを作成する必要があります。詳細については、「スキーママッピングの作成」および「マッチングワークフローを使用して入力データを照合する」を参照してください。

ID 名前空間ソースを作成したら、ID マッピングワークフローで ID 名前空間ターゲットとともに使用できます。詳細については、「Map input data using an ID mapping workflow」を参照してください。

AWS Entity Resolution コンソールで ID 名前空間ソースを作成するには、ルールベースのメソッドとプロバイダーサービスメソッドの 2 つの方法があります。

トピック

- ID 名前空間ソースの作成 (ルールベース)
- ID 名前空間ソースの作成 (プロバイダーサービス)

ID 名前空間ソースの作成 (ルールベース)

このトピックでは、ルールベースの メソッドを使用して ID 名前空間ソースを作成するプロセスについて説明します。このメソッドは、一致するルールを使用して、ID マッピングワークフローのソースからターゲットにファーストパーティデータを変換します。

> **Note**
>
> 入力データがソースの場合、スキーママッピングと関連付けられた AWS Glue データベースが必要です。

ID 名前空間ソースを作成するには (ルールベース)

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのデータ準備で、ID 名前空間を選択します。
3. ID 名前空間ページの右上隅で、ID 名前空間の作成を選択します。
4. 詳細については、以下を実行します。

a. ID 名前空間名には、一意の名前を入力します。

b. (オプション) 説明 に、オプションの説明を入力します。

c. ID 名前空間タイプで、ソースを選択します。

5. ID 名前空間メソッドで、ルールベースを選択します。

6. データ入力で、使用する入力タイプを選択し、推奨されるアクションを実行します。

入力タイプ	推奨されるアクション
既存のスキーママッピング	1. スキーママッピングを選択します。 2. ドロップダウンリストからAWS リージョン、AWS Glue データベース、AWS Glue テーブル、スキーママッピングを選択します。 最大 19 個のデータ入力を追加できます。 **Note** データテーブルに DELETE 列がある場合、スキーママッピングのタイプは でStringなければならず、matchKeyと を持つことはできませんgroupName 。
既存の一致するワークフロー	1. マッチングワークフローを選択します。 2. ID 名前空間に関連付けられているアカウントを選択します。自分 AWS アカウントまたは別の AWS アカウントのいずれかです。 3. アカウントのタイプに応じて、一致するワークフロー名を選択するか、一致するワークフロー ARN を入力します。

7. ルールパラメータについては、以下を実行します。

a. 目標に基づいて次のいずれかのオプションを選択して、ルールコントロールを指定します。

目標	推奨されるオプション
ソースとターゲットの両方からルールを許可する	設定なし
ソース、ターゲット、またはその両方が ID マッピングワークフローでルールを提供できるかどうかを選択します。	制限されたルール

ルールコントロールは、ID マッピングワークフローで使用するソースとターゲットの間で互換性がある必要があります。例えば、ソース ID 名前空間がルールをターゲットに制限するが、ターゲット ID 名前空間がルールをソースに制限する場合、エラーが発生します。

b. データ入力タイプに基づいて次のいずれかのオプションを選択して、一致ルールを指定します。

データ入力タイプ	推奨されるアクション
スキーママッピング	別のルールを追加を選択して、一致するルールを追加します。 最大 25 個の一致ルールを適用して、一致基準を定義できます。
マッチングワークフロー	一致するワークフローからルールを使用するか、新しいルールを指定して一致するルールを定義します。

8. 比較パラメータとマッチングパラメータについては、以下を実行します。

a. 目標に基づいて次のいずれかのオプションを選択して、比較タイプを指定します。

目標	推奨されるオプション
ID マッピングワークフローを作成するときに、任意の比較タイプの使用を許可します。	設定なし
データが同じ入力フィールドにあるか異なる入力フィールドにあるかに関係なく、複数の入力フィールドに保存されているデータ間で一致の任意の組み合わせを見つけます。	複数の入力フィールド
複数の入力フィールドに保存されている類似データが一致しない場合、1 つの入力フィールド内の比較を制限します。	単一入力フィールド

b. 目標に基づいて次のいずれかのオプションを選択して、レコードマッチングタイプを指定します。

目標	推奨されるオプション
ID マッピングワークフローを作成するときに、任意の比較タイプの使用を許可します。	設定なし
ID マッピングワークフローを作成するときに、レコード一致タイプを制限して、ターゲット内の一致するレコードごとに、一致するレコードを 1 つだけソースに保存します。	レコードマッチングの制限 と 1 つのソースから 1 つのターゲットへ
ID マッピングワークフローを作成するときに、レコード一致タイプを制限して、ターゲット内の一致するレコードごとに、一致するすべてのレコードをソースに保存します。	レコードマッチングの制限 と 1 つのターゲットへの多くのソース

> **Note**
>
> ソース ID 名前空間とターゲット ID 名前空間に互換性のある制限を指定する必要があります。例えば、ソース ID 名前空間がルールをターゲットに制限するが、ターゲット ID 名前空間がルールをソースに制限する場合、エラーが発生します。

9. ドロップダウンリストから既存のサービスロール名を選択して、サービスアクセス許可を指定します。
10. (オプション) リソースのタグを有効にするには、新しいタグを追加を選択し、キーと値のペアを入力します。
11. [ID 名前空間の作成] を選択します。

ID 名前空間ソースが作成されます。これで、ID 名前空間ターゲットを作成する準備が整いました。

ID 名前空間ソースの作成 (プロバイダーサービス)

このトピックでは、プロバイダーサービスメソッドを使用して ID 名前空間ソースを作成するプロセスについて説明します。この方法では、LiveRamp というプロバイダーサービスを使用します。LiveRamp は、ID マッピングワークフロー中に、サードパーティーでエンコードされたデータをソースからターゲットに変換します。

> **Note**
>
> 入力データがソースの場合、スキーママッピングと関連付けられた AWS Glue データベースが必要です。

ID 名前空間ソースを作成するには (プロバイダーサービス)

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのデータ準備で、ID 名前空間を選択します。
3. ID 名前空間ページの右上隅で、ID 名前空間の作成を選択します。
4. 詳細については、以下を実行します。

    a. ID 名前空間名には、一意の名前を入力します。

b. (オプション) 説明 に、オプションの説明を入力します。

c. ID 名前空間タイプで、ソースを選択します。

5. ID 名前空間メソッドで、プロバイダーサービスを選択します。

> **Note**
>
> AWS Entity Resolution は現在、ID 名前空間メソッドとして LiveRamp プロバイダーサービスを提供しています。LiveRamp のサブスクリプションをお持ちの場合、ステータスは Subscribed と表示されます。LiveRamp をサブスクライブする方法の詳細については、「」を参照してくださいステップ 1: でプロバイダーサービスをサブスクライブする AWS Data Exchange。

6. データ入力で、ドロップダウンリストから AWS リージョン、AWS Glue データベース、AWS Glue テーブル、スキーママッピングを選択します。

   最大 20 個のデータ入力を追加できます。

7. サービスアクセス許可を指定するには、 オプションを選択し、推奨アクションを実行します。

オプション	推奨されるアクション
新しいサービスロールを作成して使用	• AWS Entity Resolution は、このテーブルに必要なポリシーを持つサービスロールを作成します。 • デフォルトのサービスロール名は ですentityresolution-id-mapping-workflow-<timestamp> 。 • ロールを作成してポリシーをアタッチするアクセス許可が必要です。 • 入力データが暗号化されている場合は、このデータは KMS キーオプションで暗号化されます。次に、データ入力の復号に使用される AWS KMS キーを入力します。
既存のサービスロールを使用	1. ドロップダウンリストから [既存のサービスロール名] を選択します。

オプション	推奨されるアクション
	ロールを一覧表示するアクセス許可がある場合は、ロールのリストが表示されます。 ロールを一覧表示するアクセス許可がない場合は、使用するロールの Amazon リソースネーム (ARN) を入力できます。 既存のサービスロールがない場合、[既存のサービスロールを使用] オプションは使用できません。 2. [IAM で表示] 外部リンクを選択してサービスロールを表示します。 デフォルトでは、 AWS Entity Resolution は既存のロールポリシーを更新して必要なアクセス許可を追加しようとしません。

8. (オプション) リソースのタグを有効にするには、新しいタグを追加を選択し、キーと値のペアを入力します。
9. [ID 名前空間の作成] を選択します。

ID 名前空間ソースが作成されます。これで、ID 名前空間ターゲットを作成する準備が整いました。

ID 名前空間ターゲット

ID 名前空間ターゲットは、ID マッピングワークフロー内のデータのターゲットです。すべてのソースはターゲットに解決されます。

ID 名前空間ターゲットを作成する前に、ユースケースに応じて、まず一致するワークフローを作成するか、プロバイダーサービス (LiveRamp) へのサブスクリプションが必要です。詳細については、「マッチングワークフローを使用して入力データを照合する」および「ステップ 1: でプロバイダーサービスをサブスクライブする AWS Data Exchange」を参照してください。

ID 名前空間ターゲットを作成したら、ID マッピングワークフローで ID 名前空間ソースとともに使用できます。詳細については、「Map input data using an ID mapping workflow」を参照してください。

AWS Entity Resolution コンソールで ID 名前空間ターゲットを作成するには、ルールベースのメソッドとプロバイダーサービスメソッドの 2 つの方法があります。

トピック

ID 名前空間ターゲットの作成 (ルールベースのメソッド)

このトピックでは、ルールベースの メソッドを使用して ID 名前空間ターゲットを作成するプロセスについて説明します。このメソッドは、一致するルールを使用して、ID マッピングワークフロー中にファーストパーティデータをソースからターゲットに変換します。

ID 名前空間ターゲットを作成するには (ルールベース)

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのデータ準備で、ID 名前空間を選択します。
3. ID 名前空間ページの右上隅で、ID 名前空間の作成を選択します。
4. 詳細については、以下を実行します。
    a. ID 名前空間名には、一意の名前を入力します。
    b. (オプション) 説明 に、オプションの説明を入力します。
    c. ID 名前空間タイプで、Target を選択します。
5. ID 名前空間メソッドで、ルールベースを選択します。
6. データ入力の場合、一致ワークフローで以下を実行します。
    a. ID 名前空間に関連付けられているアカウントを選択します。自分 AWS アカウントまたは別の AWS アカウントのいずれかです。
    b. アカウントのタイプに応じて、一致するワークフロー名を選択するか、一致するワークフロー ARN を入力します。
7. ルールパラメータの場合は、次の操作を行います。
    a. 目標に基づいて次のいずれかのオプションを選択して、ルールコントロールを指定します。

目標	推奨されるオプション
ソースとターゲットの両方からルールを許可する	設定なし
ソース、ターゲット、またはその両方が ID マッピングワークフローでルールを提供できるかどうかを選択する	制限されたルール

ルールコントロールは、ID マッピングワークフローで使用するソースとターゲットの間で互換性がある必要があります。例えば、ソース ID 名前空間がルールをターゲットに制限するが、ターゲット ID 名前空間がルールをソースに制限する場合、エラーが発生します。

b. 一致するルールの場合、 は一致するワークフローからルール AWS Entity Resolution を自動的に追加します。

8. 比較パラメータとマッチングパラメータについては、以下を実行します。

a. 目標に基づいて次のいずれかのオプションを選択して、比較タイプを指定します。

目標	推奨されるオプション
ID マッピングワークフローを作成するときに、任意の比較タイプの使用を許可します。	設定なし
データが同じ入力フィールドにあるか異なる入力フィールドにあるかに関係なく、複数の入力フィールドに保存されているデータ間で一致の任意の組み合わせを見つけます。	複数の入力フィールド
複数の入力フィールドに保存されている類似データが一致しない場合、1 つの入力フィールド内で比較を制限します。	単一入力フィールド

b. 目標に基づいて次のいずれかのオプションを選択して、レコードマッチングタイプを指定します。

目標	推奨されるオプション
ID マッピングワークフローを作成するときに、任意の比較タイプの使用を許可します。	設定なし
ID マッピングワークフローを作成するときに、レコード一致タイプを制限して、ターゲット内の一致するレコードごとに、一致するレコードを 1 つだけソースに保存します。	レコードマッチングの制限 と 1 つのソースから 1 つのターゲットへ
ID マッピングワークフローを作成するときに、レコード一致タイプを制限して、ターゲット内の一致するレコードごとに、一致するすべてのレコードをソースに保存します。	レコードマッチングの制限 と 1 つのターゲットへの多くのソース

> **Note**
>
> ソース ID 名前空間とターゲット ID 名前空間に互換性のある制限を指定する必要があります。例えば、ソース ID 名前空間がルールをターゲットに制限するが、ターゲット ID 名前空間がルールをソースに制限する場合、エラーが発生します。

9. ドロップダウンリストから既存のサービスロール名を選択して、サービスアクセス許可を指定します。
10. (オプション) リソースのタグを有効にするには、新しいタグを追加を選択し、キーと値のペアを入力します。
11. [ID 名前空間の作成] を選択します。

ID 名前空間ターゲットが作成されます。ID マッピングワークフローに必要な ID 名前空間 (ソースとターゲット) を作成したら、ID マッピングワークフローを作成する準備が整います。

ID 名前空間ターゲットの作成 (プロバイダーサービスメソッド)

このトピックでは、プロバイダーサービスメソッドを使用して ID 名前空間ターゲットを作成するプロセスについて説明します。この方法では、LiveRamp というプロバイダーサービスを使用します。LiveRamp は、ID マッピングワークフロー中に、サードパーティーでエンコードされたデータをソースからターゲットに変換します。

ID 名前空間ターゲットを作成するには (プロバイダーサービス)

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのデータ準備で、ID 名前空間を選択します。
3. ID 名前空間ページの右上隅で、ID 名前空間の作成を選択します。
4. 詳細については、以下を実行します。
   a. ID 名前空間名には、一意の名前を入力します。
   b. (オプション) 説明 に、オプションの説明を入力します。
   c. ID 名前空間タイプで、Target を選択します。
5. ID 名前空間メソッドで、プロバイダーサービスを選択します。

> **Note**
>
> AWS Entity Resolution 現在、 は ID 名前空間メソッドとして LiveRamp プロバイダーサービスを提供しています。
> LiveRamp のサブスクリプションをお持ちの場合、ステータスは Subscribed と表示されます。
> LiveRamp をサブスクライブする方法の詳細については、「」を参照してくださいステップ 1: でプロバイダーサービスをサブスクライブする AWS Data Exchange。

6. ターゲットドメインには、LiveRamp が提供するトランスコードの対象となる LiveRamp クライアントドメイン識別子を入力します。
7. (オプション) リソースのタグを有効にするには、新しいタグを追加を選択し、キーと値のペアを入力します。
8. [ID 名前空間の作成] を選択します。

ID 名前空間ターゲットが作成されます。ID マッピングワークフローに必要な ID 名前空間 (ソースとターゲット) を作成したら、ID マッピングワークフローを作成する準備が整います。

ID 名前空間の編集

ID 名前空間は、ID マッピングワークフローに関連付ける前にのみ編集できます。ID 名前空間を ID マッピングワークフローに関連付けると、編集できなくなります。

ID 名前空間を編集するには:

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのデータ準備で、ID 名前空間を選択します。
3. ID 名前空間を選択します。
4. [編集] を選択します。
5. ID 名前空間の編集ページで、必要な変更を加え、保存を選択します。

ID 名前空間の削除

ID マッピングワークフローに関連付けられている ID 名前空間は削除できません。スキーママッピングを削除する前に、まず関連するすべての ID マッピングワークフローからスキーママッピングを削除する必要があります。

ID 名前空間を削除するには:

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのデータ準備で、ID 名前空間を選択します。
3. ID 名前空間を選択します。
4. [削除] を選択します。
5. 削除を確定し、[削除] を選択します。

ID 名前空間のリソースポリシーの追加または更新

リソースポリシーは、ID マッピングリソースの作成者が ID 名前空間リソースにアクセスすることを許可します。

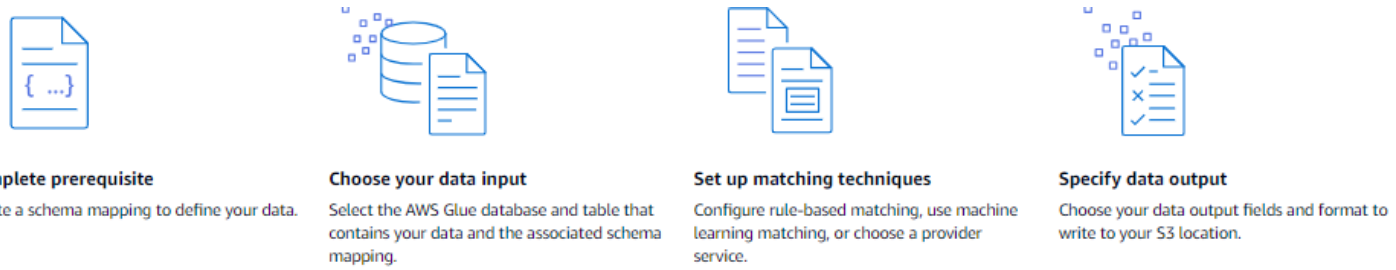
リソースポリシーを追加または更新するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、ID 名前空間を選択します。
3. ID 名前空間を選択します。
4. ID 名前空間の詳細ページで、アクセス許可タブを選択します。
5. リソースポリシーセクションで、編集を選択します。
6. JSON エディタでポリシーを追加または更新します。
7. [Save changes] (変更の保存) をクリックします。

マッチングワークフローを使用して入力データを照合する

マッチングワークフローは、さまざまな入力ソースのデータを組み合わせて比較し、さまざまなマッチング手法に基づいて一致するレコードを決定するデータ処理ジョブです。 は、指定された場所からデータを AWS Entity Resolution 読み取り、レコード間の一致を見つけ、一致する各データセットに一致 ID を割り当てます。

次の図は、一致するワークフローを作成する方法をまとめたものです。



トピック

一致するワークフロータイプ

AWS Entity Resolution は、次の 3 種類のマッチングワークフローをサポートしています。

ルールベースのマッチング

設定可能なルールを使用して、指定されたフィールドの完全一致またはあいまい一致に基づいて一致するレコードを識別します。同様にスペルされた名前の一致や、形式が異なるアドレスなど、一致する条件を定義します。

機械学習ベースマッチング

機械学習モデルを使用して、データにバリエーション、エラー、欠落しているフィールドがある場合でも、同様のレコードを識別します。このアプローチでは、ルールベースのマッチングよりも複雑なマッチングを検出できます。

プロバイダーのサービスベースのマッチング

サードパーティーのデータプロバイダーを使用して、マッチング前にデータを強化および検証します。このタイプのマッチングは、Amazon Connect Customer Profiles 出力と互換性がありません。

データ出力オプション

AWS Entity Resolution は、データ出力ファイルを次の宛先に書き込むことができます。

- 指定した Amazon S3 の場所
- Amazon Connect Customer Profiles (顧客データの重複排除用)

> **Important**
>
> Amazon Connect Customer Profiles へのエクスポートは、プロバイダーベースのマッチングと互換性がありません。Amazon Connect Customer Profiles にエクスポートするには、ルールベースのマッチングまたは機械学習ベースのマッチングを使用する必要があります。

必要に応じて AWS Entity Resolution を使用して出力データをハッシュできるため、データの制御を維持できます。

次の表は、3 種類の一致するワークフローと、サポートされている出力先を示しています。

マッチングタイプ	S3 出力	Customer Profiles 出力
ルールベース	はい	はい
機械学習ベース	はい	はい
プロバイダーのサービスベース	はい	な し

ワークフロー結果の一致

一致するワークフローを作成して実行すると、指定した S3 の場所または Amazon Connect Customer Profiles で結果を表示できます。一致するワークフローはIDs を生成します。

一致するワークフローには複数の実行を含めることができ、結果 (成功またはエラー) は を名前`jobId`とするフォルダに書き込まれます。

S3 出力先の実行ごとに:

- データ出力には、一致するファイルとエラーのファイルの両方が含まれます。
- 成功した結果は、複数のファイルを含む`success`フォルダに書き込まれます。
- エラーは複数のフィールドを持つ `error`フォルダに書き込まれます

Amazon Connect Customer Profiles 出力先の実行ごとに:

- 重複排除された顧客レコードは Amazon Connect インスタンスに直接送信されます。
- AWS Entity Resolution コンソールで最近のジョブ履歴を表示できます。
- Amazon Connect の既存のプロファイルは重複排除プロセスに含まれません

一致するワークフローを作成して実行したら、ルールベースのマッチングまたは機械学習 (ML) マッチングの出力を、プロバイダーのサービスベースのマッチングへの入力として、またはビジネスニーズを満たすための逆の方法として使用できます。

例えば、プロバイダーのサブスクリプションコストを節約するために、まずルールベースのマッチングを実行してデータに対する一致を見つけることができます。その後、一致しないレコードのサ

ブセットをプロバイダーのサービスベースのマッチングに送信できます。Customer Profiles にエクスポートする場合は、ルールベースまたは機械学習ベースのマッチングのみを使用する必要があります。

エラーのトラブルシューティングの詳細については、「」を参照してくださいマッチングワークフローのトラブルシューティング。

ルールベースのマッチングワークフローの作成

ルールベースのマッチングは、入力したデータに基づいて によって提案され AWS Entity Resolution、ユーザーが完全に設定可能なウォーターフォールマッチングルールの階層セットです。ルールベースのマッチングワークフローを使用すると、クリアテキストデータまたはハッシュデータを比較して、カスタマイズした基準に基づいて完全一致を見つけることができます。

がデータ内の 2 つ以上のレコード間の一致 AWS Entity Resolution を検出すると、以下が割り当てられます。

- 一致したデータセット内のレコードへの一致 ID
- 一致を生成した一致ルール。

でルールベースのマッチングワークフローを作成するときは AWS Entity Resolution、Simple ルールタイプまたは Advanced ルールタイプを選択する必要があります。ルールタイプによって、作成できるルール条件の複雑さが決まります。ワークフローの作成後にルールタイプを変更することはできません。

次のグラフを使用して、2 つのルールタイプを比較し、ユースケースに適したルールタイプを特定できます。

ルールタイプ比較グラフ

ユースケース	高度なルールタイプ	シンプルなルールタイプ
入力タイプで one-to-oneでマッピングされたスキーママッピング	Yes	No
複数のデータ列が同じ入力タイプにマッピングされたスキーママッピング	No	Yes

ユースケース	高度なルールタイプ	シンプルなルールタイプ
完全一致とあいまい一致をサポート	Yes	No (Exact matching only)
AND、OR、および括弧演算子をサポート	Yes	No (AND operator only)
バッチワークフローをサポート	Yes	Yes
増分ワークフローをサポート	Yes	Yes
リアルタイムワークフローをサポート	No	Yes
ID マッピングワークフローをサポート	No	Yes

使用するルールタイプを決定したら、次のトピックを使用して、アドバンストルールタイプまたはシンプルルールタイプでルールベースのマッチングワークフローを作成します。

トピック

- 高度なルールタイプを使用したルールベースのマッチングワークフローの作成
- Simple ルールタイプを使用したルールベースのマッチングワークフローの作成

高度なルールタイプを使用したルールベースのマッチングワークフローの作成

前提条件

ルールベースのマッチングワークフローを作成する前に、以下を行う必要があります。

1. スキーママッピングを作成します。詳細については、「スキーママッピングの作成」を参照してください。

2. Amazon Connect Customer Profiles を出力先として使用する場合は、適切なアクセス許可が設定されていることを確認してください。

次の手順は、 AWS Entity Resolution コンソールまたは `CreateMatchingWorkflow` API を使用して、アドバンストルールタイプでルールベースのマッチングワークフローを作成する方法を示しています。

Console

コンソールを使用してアドバンストルールタイプでルールベースのマッチングワークフローを作成するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、一致を選択します。
3. 一致するワークフローページの右上隅で、一致するワークフローの作成を選択します。
4. ステップ 1: 一致するワークフローの詳細を指定するには、以下を実行します。
   a. 一致するワークフロー名とオプションの 説明を入力します。
   b. データ入力で、、AWS リージョンAWS Glue データベース、AWS Glue テーブルを選択し、対応するスキーママッピングを選択します。

      最大 19 個のデータ入力を追加できます。

      > **Note**
      >
      > アドバンストルールを使用するには、スキーママッピングが次の要件を満たしている必要があります。
      >
      > 1. 各入力フィールドは、フィールドがグループ化されていない限り、一意の一致キーにマッピングする必要があります。
      > 2. 入力フィールドがグループ化されている場合、同じ一致キーを共有できます。
      >
      > たとえば、次のスキーママッピングはアドバンストルールで有効です。
      >
      > ```
      > firstName: { matchKey: 'name', groupName: 'name' }
      > ```

```
lastName: { matchKey: 'name', groupName: 'name' }
```

この場合、 `firstName`フィールドと `lastName`フィールドはグループ化され、同じ名前一致キーを共有します。これは許可されます。

高度なルールを使用するには、フィールドが適切にグループ化されていない限り、スキーママッピングを確認して更新し、この one-to-oneの一致ルールに従います。

3. データテーブルに DELETE 列がある場合、スキーママッピングのタイプは でなければならず`String`、 `matchKey`と を持つことはできませんgroupName。

c. データ正規化オプションはデフォルトで選択され、一致する前にデータ入力が正規化されます。データを正規化しない場合は、データの正規化オプションの選択を解除します。

> **Note**
>
> 正規化は、スキーママッピングの作成で以下のシナリオでのみサポートされています。
>
> - 次の名前サブタイプがグループ化されている場合: 名、ミドルネーム、姓。
> - 住所サブタイプがグループ化されている場合: 住所 1、住所 2、住所 3、市区町村、州、国、郵便番号。
> - 電話番号サブタイプがグループ化されている場合: 電話番号、電話番号の国コード。

d. サービスアクセス許可を指定するには、 オプションを選択し、推奨アクションを実行します。

オプション	推奨されるアクション
新しいサービスロールを作成して使用	• AWS Entity Resolution は、このテーブルに必要なポリシーを持つサービスロールを作成します。 • デフォルトの [サービスロール名] は `entityresolution-matching-workflow-<timestamp>` です。 • ロールを作成してポリシーをアタッチするアクセス許可が必要です。 • 入力データが暗号化されている場合、このデータは KMS キーオプションで暗号化され、データ入力の復号に使用される AWS KMS キーを入力できます。

オプション	推奨されるアクション
既存のサービスロールを使用	1. ドロップダウンリストから [既存のサービスロール名] を選択します。 ロールを一覧表示するアクセス許可がある場合は、ロールのリストが表示されます。 ロールを一覧表示するアクセス許可がない場合は、使用するロールの Amazon リソースネーム (ARN) を入力できます。 既存のサービスロールがない場合、[既存のサービスロールを使用] オプションは使用できません。 2. [IAM で表示] 外部リンクを選択してサービスロールを表示します。 デフォルトでは、 AWS Entity Resolution は既存のロールポリシーを更新して必要なアクセス許可を追加しようとしません。

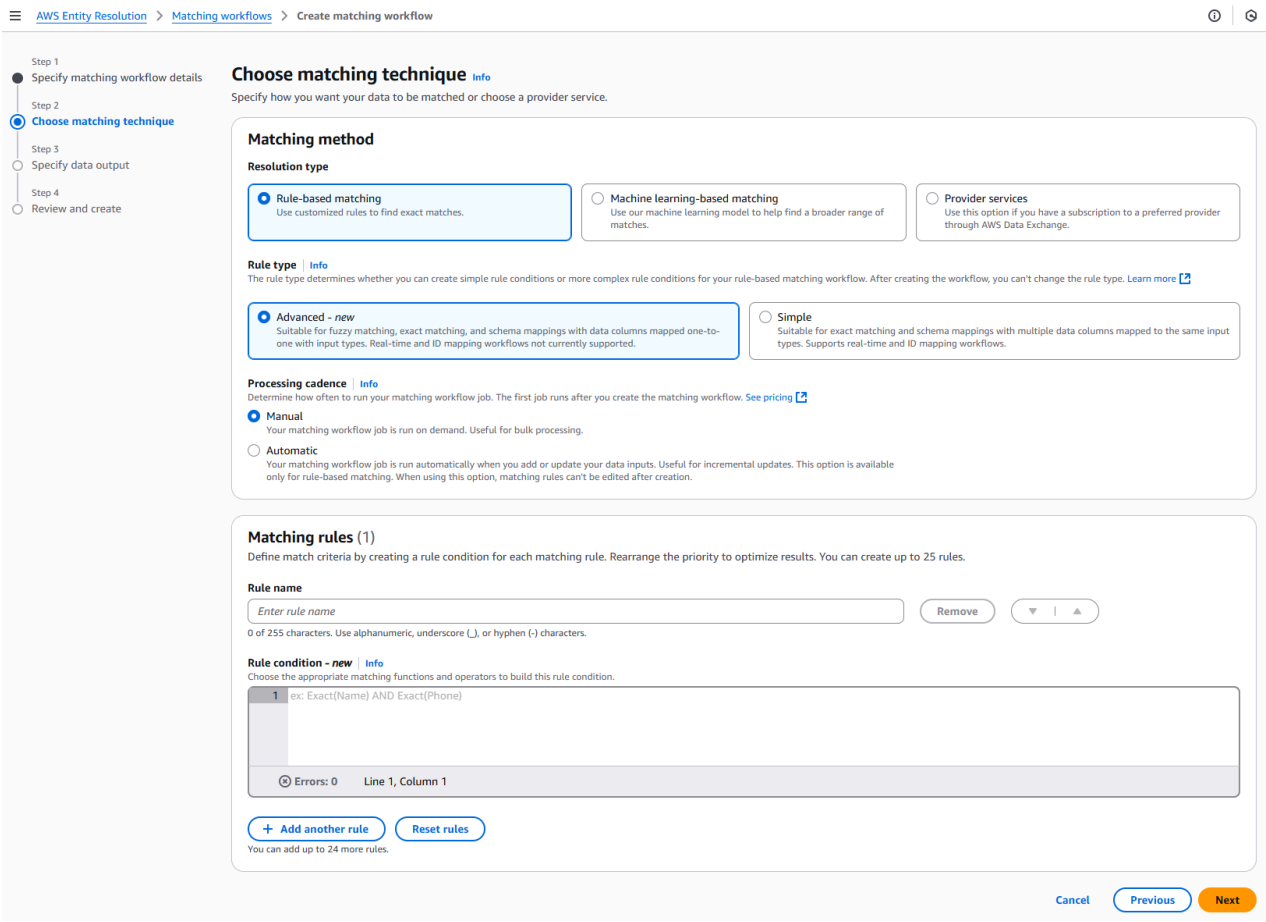
e. (オプション) リソースのタグを有効にするには、新しいタグを追加を選択し、キーと値のペアを入力します。

f. [次へ] を選択します。

5. ステップ 2: 一致する手法を選択するには:

a. マッチングメソッドで、ルールベースのマッチングを選択します。

b. ルールタイプで、詳細 を選択します。



c. Processing cadence で、次のいずれかのオプションを選択します。

- 手動を選択して、一括更新のワークフローをオンデマンドで実行する
- 自動 を選択して、新しいデータが S3 バケットに保存されたらすぐにワークフローを実行します。

> **Note**
>
> 自動 を選択した場合は、S3 バケットに対して Amazon EventBridge 通知が有効になっていることを確認します。S3 コンソールを使用して Amazon EventBridge を有効にする手順については、「Amazon S3 ユーザーガイド」の「Amazon EventBridge の有効化」を参照してください。 Amazon S3

d. 一致するルールには、ルール名を入力し、目標に基づいてドロップダウンリストから適切な一致する関数と演算子を選択してルール条件を構築します。

最大 25 個のルールを作成できます。

AND 演算子を使用して、あいまい一致関数 (Cosine、Levenshtein、または Soundex) と完全に一致する関数 (Exact、ExactManyToMany) を組み合わせる必要があります。

以下の表を使用して、目標に応じて使用する関数または演算子のタイプを決定できます。

目標	推奨される関数または演算子	推奨されるオプションの修飾子	メリット
正確なデータでは同じ文字列を一致させますが、空の値では一致しません。	完全一致	EmptyValues=Process	
正確なデータで同一の文字列を一致させ、空の値を無視します。	Exact(*matchKey*)	EmptyValues=無視	
一致キー間で複数のレコードを一致させます。柔軟なペアリングに適しています。制限: 15 個の一致キー	ExactManyToMany(*matchKey*、	該当なし	

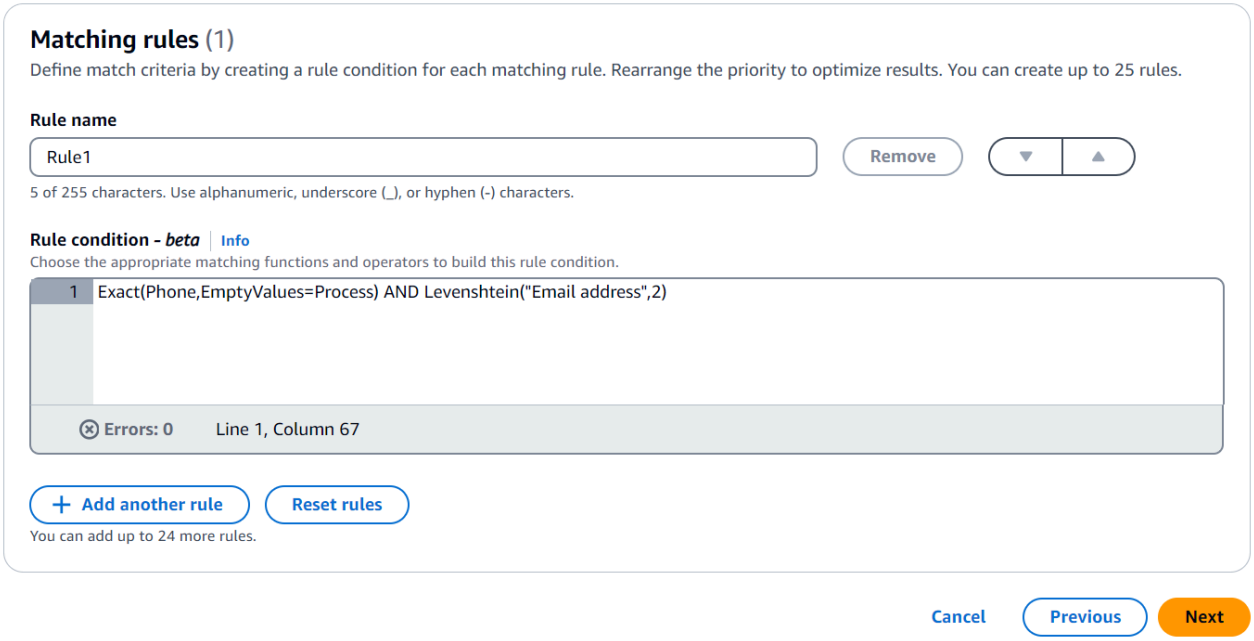
目標	推奨される関数または演算子	推奨されるオプションの修飾子	メリット
データの数値表現間の類似性を測定しますが、空の値では一致しません。テキスト、数字、またはその両方の組み合わせに適しています。	コサイン	EmptyValues=Process	シンプルで効率的です。 TF-IDF 重み付けと組み合わせると、ロングテキストに適しています。 単語ベースの完全一致に適しています。
データの数値表現間の類似性を測定し、空の値を無視します。	Cosine(***matchKey***、***###***など)	EmptyValues=無視	誤字、スペルエラー、転置を適切に処理します。 幅広い PII タイプに有効です。 短い文字列 (名前や電話番号など) に適しています。
ある単語を別の単語に変更するために必要な変更の最小数をカウントしますが、空の値では一致しません。スペルにわずかな違いがあるテキストに適しています。	レベンシュテイン	EmptyValues=Process	
ある単語を別の単語に変更するために必要な変更の最小数をカウントし、空の値を無視します。	Levenshtein(***matchKey***、***###***など)	EmptyValues=無視	

目標	推奨される関数または演算子	推奨されるオプションの修飾子	メリット
テキスト文字列は、どの程度類似しているが、空の値では一致しないかに基づいて比較して一致させます。スペルや発音のバリエーションがあるテキストに適しています。	Soundex	EmptyValues=Process	音声マッチングに有効で、類似する単語を識別します。 高速で計算コストが低くなります。 発音は似ていますが、スペルが異なる名前を一致させるのに適しています。
テキスト文字列をどの程度類似しているかに基づいて比較して一致させ、空の値を無視します。	Soundex(*matchKey*)	EmptyValues=無視	
関数を結合します。	AND	該当なし	
関数を区切ります。	または	該当なし	
条件をグループ化してネストされた条件を作成します。	(...)	該当なし	

Example電話番号と E メールに一致するルール条件

以下は、電話番号 (電話一致キー) と E メールアドレス (E メールアドレス一致キー) のレコードに一致するルール条件の例です。

```
Exact(Phone,EmptyValues=Process) AND Levenshtein("Email
address",2)
```



電話一致キーは、完全一致関数を使用して同一の文字列を一致させます。電話一致キーは、EmptyValues=Process 修飾子を使用して、マッチングで空の値を処理します。

E メールアドレス一致キーは、Levenshtein マッチング関数を使用して、デフォルトの Levenshtein Distance アルゴリズムしきい値 2 を使用してデータをスペルミスと照合します。E メール一致キーは、オプションの修飾子を使用しません。

AND 演算子は、完全一致関数と Levenshtein 一致関数を組み合わせます。

Example ExactManyToMany を使用してマッチキーマッチングを実行するルール条件

以下は、3 つのアドレスフィールド (HomeAddress 一致キー、BillingAddress 一致キー、および ShippingAddress 一致キー) のレコードに一致するルール条件の例です。いずれかのレコードに同じ値があるかどうかを確認します。

ExactManyToMany 演算子は、指定されたアドレスフィールドの可能なすべての組み合わせを評価して、2 つ以上のアドレス間の完全一致を特定します。たとえば、 が BillingAddressまたは HomeAddressに一致するかどうかShippingAddress、または 3 つのアドレスすべてが正確に一致するかどうかを検出します。

```
ExactManyToMany(HomeAddress, BillingAddress, ShippingAddress)
```

Exampleクラスタリングを使用するルール条件

あいまいな条件での高度なルールベースのマッチングでは、システムは完全一致に基づいて最初にレコードをクラスターにグループ化します。これらの初期クラスターが作成されると、システムはあいまい一致フィルターを適用して、各クラスター内の追加の一致を特定します。最適なパフォーマンスを得るには、データパターンに基づいて完全一致条件を選択して、明確に定義された初期クラスターを作成する必要があります。

以下は、複数の完全一致とあいまい一致要件を組み合わせたルール条件の例です。AND 演算子を使用して、、生年月日 (DOB)FullName、および Address の 3 つのフィールドがレコード間で正確に一致することをチェックします。また、 の Levenshtein 距離を使用して、 InternalIDフィールドのわずかなバリエーションも可能です1。Levenshtein の距離は、ある文字列を別の文字列に変更するために必要な 1 文字の編集の最小数を測定します。距離が 1 の場合、一致するInternalIDs文字は 1 文字だけ異なります (1 つのタイプミス、削除、挿入など) 。この条件の組み合わせは、識別子にわずかな不一致があっても、同じエンティティを表す可能性が非常に高いレコードを識別するのに役立ちます。

```
Exact(FullName) AND Exact(DOB) AND Exact(Address) and
 Levenshtein(InternalID, 1)
```

e. [次へ] を選択します。

6. ステップ 3: データ出力と形式を指定するには:

   a. データ出力の送信先と形式については、データ出力の Amazon S3 の場所と、データ形式を正規化データまたは元のデータのどちらにするかを選択します。

   b. 暗号化 で、暗号化設定をカスタマイズする場合は、AWS KMS キー ARN を入力します。

   c. システム生成の出力を表示します。

   d. データ出力では、含める、非表示にする、またはマスクするフィールドを決定し、目標に基づいて推奨アクションを実行します。

目標	推奨されるアクション
フィールドを含める	出力状態をインクルードのままにします。
フィールドを非表示 (出力から除外)	Output フィールドを選択し、Hide を選択します。
マスクフィールド	出力フィールドを選択し、ハッシュ出力を選択します。
以前の設定をリセットする	[リセット] を選択します。

e. [次へ] を選択します。

7. ステップ 4: 確認して作成する:

   a. 前のステップで行った選択内容を確認し、必要に応じて編集します。

   b. Create and run を選択します。

      一致するワークフローが作成され、ジョブが開始されたことを示すメッセージが表示されます。

8. 一致するワークフローの詳細ページで、メトリクスタブで、「最後のジョブメトリクス」の下に以下を表示します。

   - ジョブ ID。
   - 一致するワークフロージョブのステータス: Queued、In progress、Completed、Failed
   - ワークフロージョブの完了時刻。
   - 処理されたレコードの数。
   - 処理されていないレコードの数。
   - 生成された一意の一致 IDs。
   - 入力レコードの数。

   ジョブ履歴で以前に実行された一致するワークフロージョブのジョブメトリクスを表示することもできます。

9. 一致するワークフロージョブが完了したら (ステータスが完了)、データ出力タブに移動し、Amazon S3 の場所を選択して結果を表示できます。
10. (手動処理タイプのみ) 手動処理タイプを使用してルールベースのマッチングワークフローを作成した場合は、一致するワークフローの詳細ページでワークフローの実行を選択して、一致するワークフローをいつでも実行できます。
11. (自動処理タイプのみ) データテーブルに DELETE 列がある場合:

    - DELETE 列で *true* に設定されたレコードは削除されます。
    - DELETE 列で *false* に設定されたレコードは S3 に取り込まれます。

    詳細については、「ステップ 1: ファーストパーティーデータテーブルを準備する」を参照してください。

API

API を使用してアドバンストルールタイプでルールベースのマッチングワークフローを作成するには

> **Note**
>
> デフォルトでは、ワークフローは標準 (バッチ) 処理を使用します。増分 (自動処理) を使用するには、明示的に設定する必要があります。

1. ターミナルまたはコマンドプロンプトを開いて API リクエストを行います。
2. 次のエンドポイントへの POST リクエストを作成します。

```
/matchingworkflows
```

3. リクエストヘッダーで、Content-type を application/json に設定します。

> **Note**
>
> サポートされているプログラミング言語の完全なリストについては、 AWS Entity Resolution API リファレンスを参照してください。

4. リクエスト本文には、次の必須 JSON パラメータを指定します。

```
{
   "description": "string",
   "incrementalRunConfig": {
      "incrementalRunType": "string"
   },
   "inputSourceConfig": [
      {
         "applyNormalization": boolean,
         "inputSourceARN": "string",
         "schemaName": "string"
      }
   ],
   "outputSourceConfig": [
      {
         "applyNormalization": boolean,
         "KMSArn": "string",
         "output": [
            {
               "hashed": boolean,
               "name": "string"
            }
         ],
         "outputS3Path": "string"
      }
   ],
   "resolutionTechniques": {
      "providerProperties": {
         "intermediateSourceConfiguration": {
            "intermediateS3Path": "string"
         },
         "providerConfiguration": JSON value,
         "providerServiceArn": "string"
      },
      "resolutionType": "RULE_MATCHING",
      "ruleBasedProperties": {
         "attributeMatchingModel": "string",
         "matchPurpose": "string",
         "rules": [
            {
               "matchingKeys": [ "string" ],
               "ruleName": "string"
            }
         ]
```

```
      },
      "ruleConditionProperties": {
         "rules": [
            {
               "condition": "string",
               "ruleName": "string"
            }
         ]
      }
   },
   "roleArn": "string",
   "tags": {
      "string" : "string"
   },
   "workflowName": "string"
}
```

コードの説明は以下のとおりです。

- workflowName (必須) – 一意で、パターン [a-zA-Z_0-9-] に一致する 1～255 文字である必要があります*
- inputSourceConfig (必須) – 1～20 個の入力ソース設定のリスト
- outputSourceConfig (必須) — 正確に 1 つの出力ソース設定
- resolutionTechniques (必須) – ルールベースのマッチングの resolutionType として「RULE_MATCHING」に設定する
- roleArn (必須) – ワークフロー実行用の IAM ロール ARN
- ruleConditionProperties (必須) – ルール条件のリストと一致するルールの名前。

オプションパラメータは次のとおりです。

- description – 最大 255 文字
- incrementalRunConfig – 増分実行タイプ設定
- tags — 最大 200 個のキーと値のペア

5. (オプション) デフォルトの標準 (バッチ) 処理の代わりに増分処理を使用するには、リクエスト本文に次のパラメータを追加します。

```
"incrementalRunConfig": {
   "incrementalRunType": "AUTOMATIC"
```

```
}
```

6. リクエストを送信します。
7. 成功すると、ステータスコード 200 と以下を含む JSON 本文を含むレスポンスを受け取ります。

```
{
   "workflowArn": "string",
   "workflowName": "string",
   // Plus all configured workflow details
}
```

8. 呼び出しが失敗すると、次のいずれかのエラーが表示されることがあります。
   - 400 – ワークフロー名が既に存在する場合の ConflictException
   - 400 – 入力が検証に失敗した場合の ValidationException
   - 402 – アカウント制限を超えた場合の ExceedsLimitException
   - 403 – 十分なアクセスがない場合の AccessDeniedException
   - 429 – リクエストがスロットリングされた場合の ThrottlingException
   - 500 – 内部サービスに障害が発生した場合の InternalServerException

Simple ルールタイプを使用したルールベースのマッチングワークフローの作成

前提条件

ルールベースのマッチングワークフローを作成する前に、以下を行う必要があります。

1. スキーママッピングを作成します。詳細については、「スキーママッピングの作成」を参照してください。
2. Amazon Connect Customer Profiles を出力先として使用する場合は、適切なアクセス許可が設定されていることを確認してください。

次の手順は、 AWS Entity Resolution コンソールまたは `CreateMatchingWorkflow` API を使用して Simple ルールタイプでルールベースのマッチングワークフローを作成する方法を示しています。

Console

コンソールを使用して Simple ルールタイプでルールベースのマッチングワークフローを作成するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、一致を選択します。
3. 一致するワークフローページの右上隅で、一致するワークフローの作成を選択します。
4. ステップ 1: 一致するワークフローの詳細を指定するには、以下を実行します。

   a. 一致するワークフロー名とオプションの 説明を入力します。

   b. データ入力で、、AWS リージョンAWS Glue データベース、AWS Glue テーブルを選択し、対応するスキーママッピングを選択します。

      最大 19 個のデータ入力を追加できます。

   c. データ正規化オプションはデフォルトで選択され、一致する前にデータ入力が正規化されます。データを正規化しない場合は、データの正規化オプションの選択を解除します。

      > **Note**
      >
      > 正規化は、スキーママッピングの作成で以下のシナリオでのみサポートされています。
      >
      > - 名前サブタイプがグループ化されている場合: 名、ミドルネーム、姓。
      > - 住所サブタイプがグループ化されている場合: 住所 1、住所 2、住所 3、市区町村、州、国、郵便番号。
      > - 電話番号サブタイプがグループ化されている場合: 電話番号、電話番号の国コード。

   d. サービスアクセス許可を指定するには、 オプションを選択し、推奨アクションを実行します。

オプション	推奨されるアクション
新しいサービスロールを作成して使用	• AWS Entity Resolution は、このテーブルに必要なポリシーを持つサービスロールを作成します。 • デフォルトの [サービスロール名] は `entityresolution-matching-workflow-<timestamp>` です。 • ロールを作成してポリシーをアタッチするアクセス許可が必要です。 • 入力データが暗号化されている場合、このデータは KMS キーオプションで暗号化され、データ入力の復号に使用される AWS KMS キーを入力できます。

オプション	推奨されるアクション
既存のサービスロールを使用	1. ドロップダウンリストから [既存のサービスロール名] を選択します。 ロールを一覧表示するアクセス許可がある場合は、ロールのリストが表示されます。 ロールを一覧表示するアクセス許可がない場合は、使用するロールの Amazon リソースネーム (ARN) を入力できます。 既存のサービスロールがない場合、[既存のサービスロールを使用] オプションは使用できません。 2. [IAM で表示] 外部リンクを選択してサービスロールを表示します。 デフォルトでは、 AWS Entity Resolution は既存のロールポリシーを更新して必要なアクセス許可を追加しようとしません。

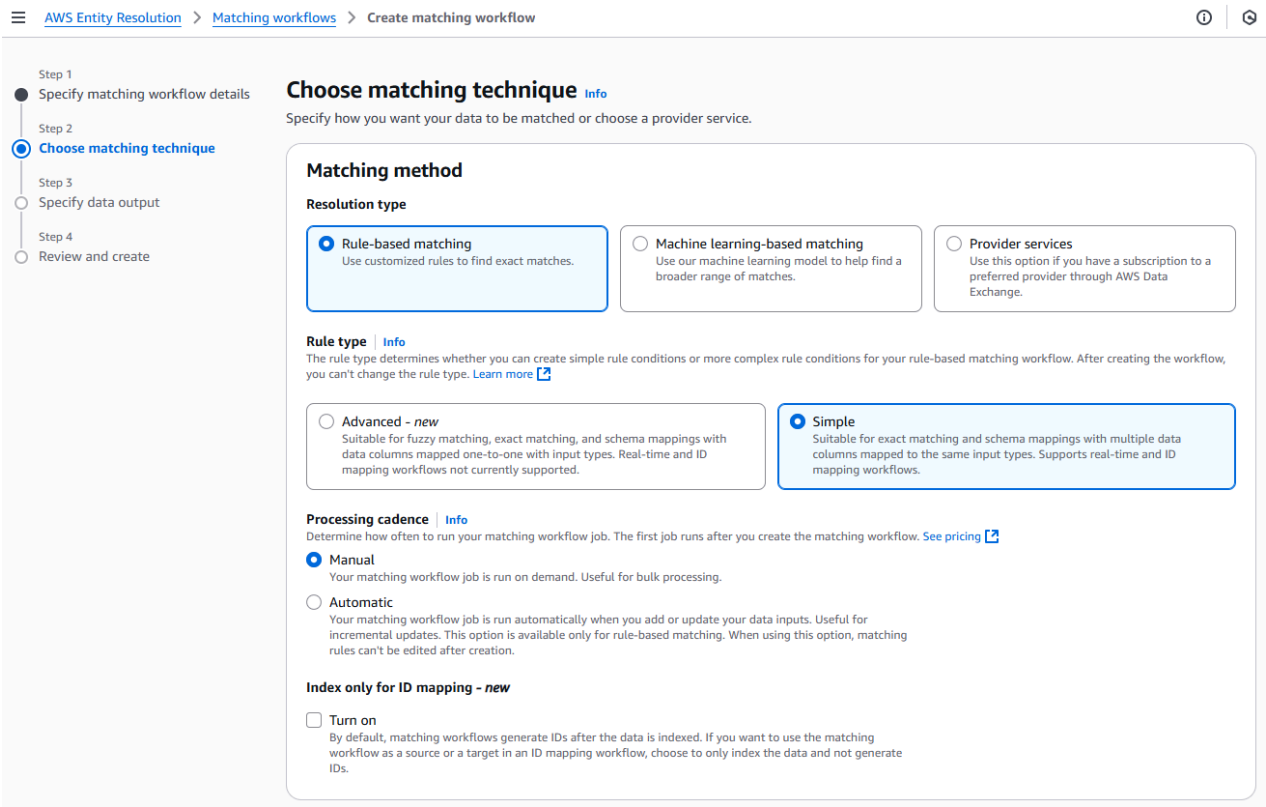
e. (オプション) リソースのタグを有効にするには、新しいタグを追加を選択し、キーと値のペアを入力します。

f. [次へ] を選択します。

5. ステップ 2: 一致する手法を選択するには:

a. マッチングメソッドで、ルールベースのマッチングを選択します。

b. ルールタイプで、シンプル を選択します。



c. Processing cadence で、次のいずれかのオプションを選択します。

- 手動を選択して、一括更新のワークフローをオンデマンドで実行する
- 自動 を選択して、新しいデータが S3 バケットに保存されたらすぐにワークフローを実行します。

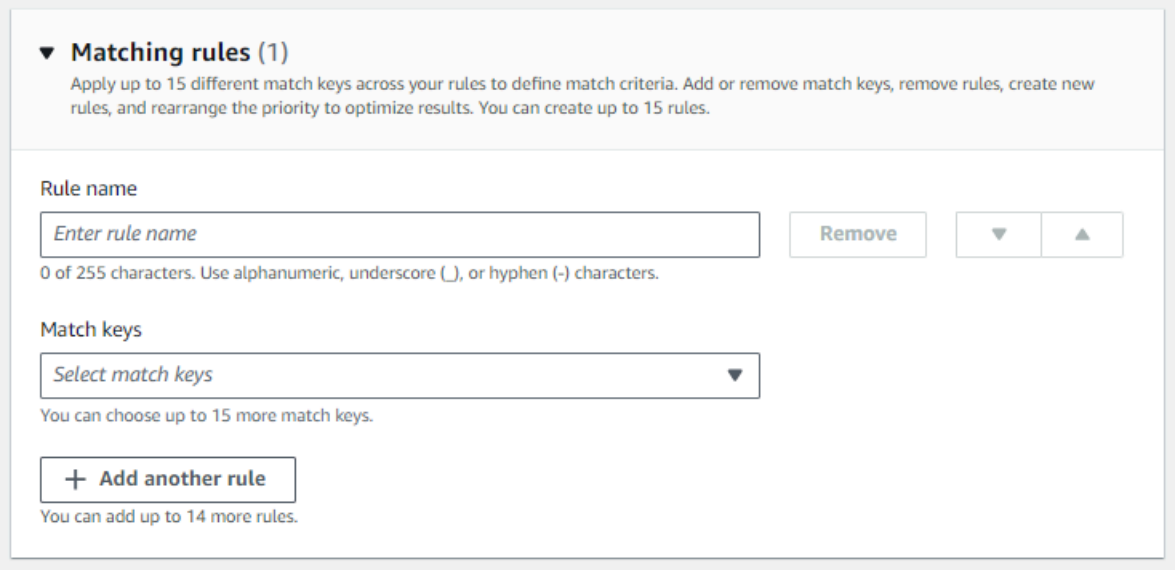
> **Note**
>
> 自動 を選択した場合は、S3 バケットに対して Amazon EventBridge 通知が有効になっていることを確認します。S3 コンソールを使用して Amazon EventBridge を有効にする手順については、「Amazon S3 ユーザーガイド」の「Amazon EventBridge の有効化」を参照してください。 Amazon S3

d. (オプション) ID マッピングのインデックスのみの場合、データのインデックス作成のみを有効にし、IDsを生成しないことを選択できます。

デフォルトでは、一致するワークフローは、データのインデックス作成後に IDs を生成します。

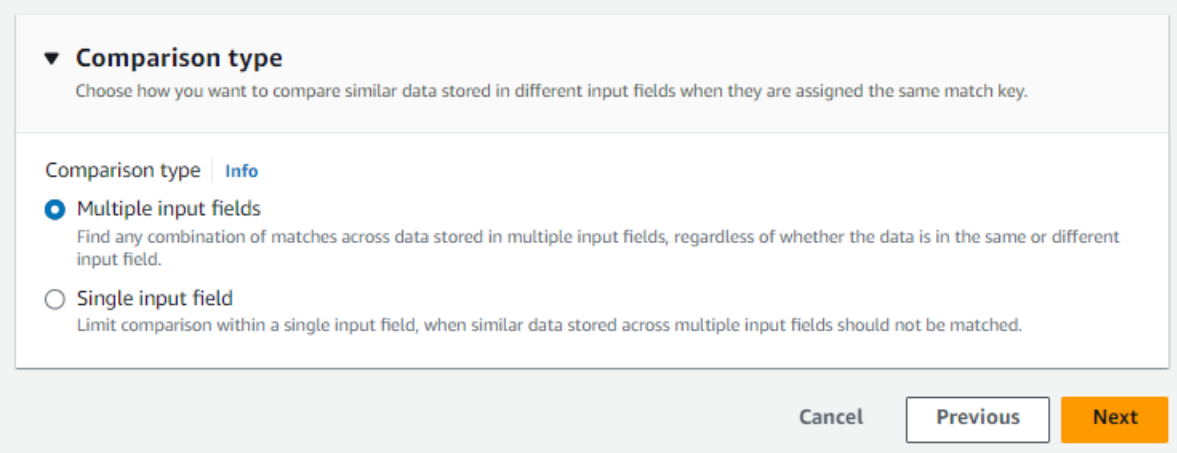
e. 一致ルールには、ルール名を入力し、そのルールの一致キーを選択します。

最大 15 個のルールを作成し、ルール全体に最大 15 個の異なる一致キーを適用して、一致基準を定義できます。



f. 比較タイプでは、目標に基づいて次のいずれかのオプションを選択します。

目標	推奨されるオプション
複数の入力フィールドに保存されているデータ間の一致の任意の組み合わせを検索する	複数の入力フィールド
比較を単一の入力フィールドに制限する	単一入力フィールド



g. [次へ] を選択します。

6. ステップ 3: データ出力と形式を指定するには:

   a. データ出力の送信先と形式については、データ出力の Amazon S3 の場所と、データ形式を正規化データまたは元のデータのどちらにするかを選択します。

   b. 暗号化 で、暗号化設定をカスタマイズする場合は、AWS KMS キー ARN を入力します。

   c. システム生成出力を表示します。

   d. データ出力では、含める、非表示にする、またはマスクするフィールドを決定し、目標に基づいて推奨アクションを実行します。

目標	推奨されるアクション
フィールドを含める	出力状態をインクルードのままにします。
フィールドを非表示 (出力から除外)	Output フィールドを選択し、Hide を選択します。
マスクフィールド	出力フィールドを選択し、ハッシュ出力を選択します。
以前の設定をリセットする	[リセット] を選択します。

   e. [次へ] を選択します。

7. ステップ 4: 確認して作成する:

   a. 前のステップで行った選択内容を確認し、必要に応じて編集します。

   b. Create and run を選択します。

      一致するワークフローが作成され、ジョブが開始されたことを示すメッセージが表示されます。

8. 一致するワークフローの詳細ページのメトリクスタブで、「最後のジョブメトリクス」で以下を表示します。

   - ジョブ ID。
   - 一致するワークフロージョブのステータス: Queued、In progress、Completed、Failed

- ワークフロージョブの完了時刻。
- 処理されたレコードの数。
- 処理されていないレコードの数。
- 生成された一意の一致 IDs。
- 入力レコードの数。

ジョブ履歴で以前に実行された一致するワークフロージョブのジョブメトリクスを表示することもできます。

9. 一致するワークフロージョブが完了したら (ステータスが完了)、データ出力タブに移動し、Amazon S3 の場所を選択して結果を表示できます。
10. (手動処理タイプのみ) 手動処理タイプを使用してルールベースのマッチングワークフローを作成した場合は、一致するワークフローの詳細ページでワークフローの実行を選択して、一致するワークフローをいつでも実行できます。

API

API を使用して Simple ルールタイプでルールベースのマッチングワークフローを作成するには

> **Note**
>
> デフォルトでは、ワークフローは標準 (バッチ) 処理を使用します。増分 (自動処理) を使用するには、明示的に設定する必要があります。

1. ターミナルまたはコマンドプロンプトを開いて API リクエストを行います。
2. 次のエンドポイントへの POST リクエストを作成します。

```
/matchingworkflows
```

3. リクエストヘッダーで、Content-type を application/json に設定します。

> **Note**
>
> サポートされているプログラミング言語の完全なリストについては、 AWS Entity Resolution API リファレンスを参照してください。

4. リクエスト本文には、次の必須 JSON パラメータを指定します。

```
{
   "description": "string",
   "incrementalRunConfig": {
      "incrementalRunType": "string"
   },
   "inputSourceConfig": [
      {
         "applyNormalization": boolean,
         "inputSourceARN": "string",
         "schemaName": "string"
      }
   ],
   "outputSourceConfig": [
      {
         "applyNormalization": boolean,
         "KMSArn": "string",
         "output": [
            {
               "hashed": boolean,
               "name": "string"
            }
         ],
         "outputS3Path": "string"
      }
   ],
   "resolutionTechniques": {
      "providerProperties": {
         "intermediateSourceConfiguration": {
            "intermediateS3Path": "string"
         },
         "providerConfiguration": JSON value,
         "providerServiceArn": "string"
      },
      "resolutionType": "RULE_MATCHING",
      "ruleBasedProperties": {
         "attributeMatchingModel": "string",
         "matchPurpose": "string",
         "rules": [
            {
               "matchingKeys": [ "string" ],
               "ruleName": "string"
```

```
            }
         ]
      },
      "ruleConditionProperties": {
         "rules": [
            {
               "condition": "string",
               "ruleName": "string"
            }
         ]
      }
   },
   "roleArn": "string",
   "tags": {
      "string" : "string"
   },
   "workflowName": "string"
}
```

コードの説明は以下のとおりです。

- workflowName (必須) – 一意で、パターン [a-zA-Z_0-9-] に一致する 1～255 文字である必要があります*
- inputSourceConfig (必須) – 1～20 個の入力ソース設定のリスト
- outputSourceConfig (必須) — 正確に 1 つの出力ソース設定
- resolutionTechniques (必須) – ルールベースのマッチングでは「RULE_MATCHING」に設定します
- roleArn (必須) – ワークフロー実行用の IAM ロール ARN
- ruleConditionProperties (必須) – ルール条件のリストと一致するルールの名前。

オプションパラメータは次のとおりです。

- description – 最大 255 文字
- incrementalRunConfig – 増分実行タイプ設定
- tags — 最大 200 個のキーと値のペア

5. (オプション) デフォルトの標準 (バッチ) 処理の代わりに増分処理を使用するには、リクエスト本文に次のパラメータを追加します。

```
"incrementalRunConfig": {
   "incrementalRunType": "AUTOMATIC"
}
```

6. リクエストを送信します。
7. 成功すると、ステータスコード 200 と以下を含む JSON 本文を含むレスポンスを受け取ります。

```
{
   "workflowArn": "string",
   "workflowName": "string",
   // Plus all configured workflow details
}
```

8. 呼び出しが失敗すると、次のいずれかのエラーが表示されることがあります。

- 400 – ワークフロー名が既に存在する場合の ConflictException
- 400 – 入力が検証に失敗した場合の ValidationException
- 402 – アカウント制限を超えた場合の ExceedsLimitException
- 403 – 十分なアクセスがない場合の AccessDeniedException
- 429 – リクエストがスロットリングされた場合の ThrottlingException
- 500 – 内部サービスに障害が発生した場合の InternalServerException

機械学習ベースのマッチングワークフローの作成

機械学習ベースのマッチングは、入力したすべてのデータのレコードを照合しようとするプリセットプロセスです。機械学習ベースのマッチングワークフローを使用すると、クリアテキストデータを比較して、機械学習モデルを使用して幅広いマッチングを見つけることができます。

> **Note**
>
> 機械学習モデルは、ハッシュ化されたデータの比較をサポートしていません。

がデータ内の 2 つ以上のレコード間の一致 AWS Entity Resolution を検出すると、以下が割り当てられます。

- 一致したデータセット内のレコードへの一致 ID
- 一致信頼レベルの割合。

ML ベースのマッチングワークフローの出力をデータサービスプロバイダーマッチングの入力として使用することも、その逆を使用して特定の目標を達成することもできます。たとえば、ML ベースのマッチングを実行して、最初に独自のレコードでデータソース間の一致を検索できます。サブセットが一致しなかった場合は、プロバイダーのサービスベースのマッチングを実行して、追加のマッチングを見つけることができます。

前提条件

ML ベースのマッチングワークフローを作成する前に、以下を行う必要があります。

1. スキーママッピングを作成します。詳細については、「スキーママッピングの作成」を参照してください。
2. Amazon Connect Customer Profiles を出力先として使用する場合は、適切なアクセス許可が設定されていることを確認してください。

ML ベースのマッチングワークフローを作成するには:

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、一致を選択します。
3. 一致するワークフローページの右上隅で、一致するワークフローの作成を選択します。
4. ステップ 1: 一致するワークフローの詳細を指定するには、以下を実行します。

    a. 一致するワークフロー名とオプションの 説明を入力します。

    b. データ入力で、AWS リージョン、AWS Glue データベース、AWS Glue テーブルを選択し、対応するスキーママッピングを選択します。

    最大 20 個のデータ入力を追加できます。

    c. データの正規化オプションはデフォルトで選択され、一致する前にデータ入力が正規化されます。データを正規化しない場合は、データの正規化オプションの選択を解除します。

    機械学習ベースのマッチングでは名前、、電話、 のみが正規化されますE メール。

    d. サービスアクセス許可を指定するには、 オプションを選択し、推奨アクションを実行します。

オプション	推奨されるアクション
新しいサービスロールを作成して使用	• AWS Entity Resolution は、このテーブルに必要なポリシーを持つサービスロールを作成します。 • デフォルトの [サービスロール名] は `entityresolution-matching-workflow-<timestamp>` です。 • ロールを作成してポリシーをアタッチするアクセス許可が必要です。 • 入力データが暗号化されている場合は、このデータは KMS キーオプションで暗号化されます。次に、データ入力の復号に使用される AWS KMS キーを入力します。

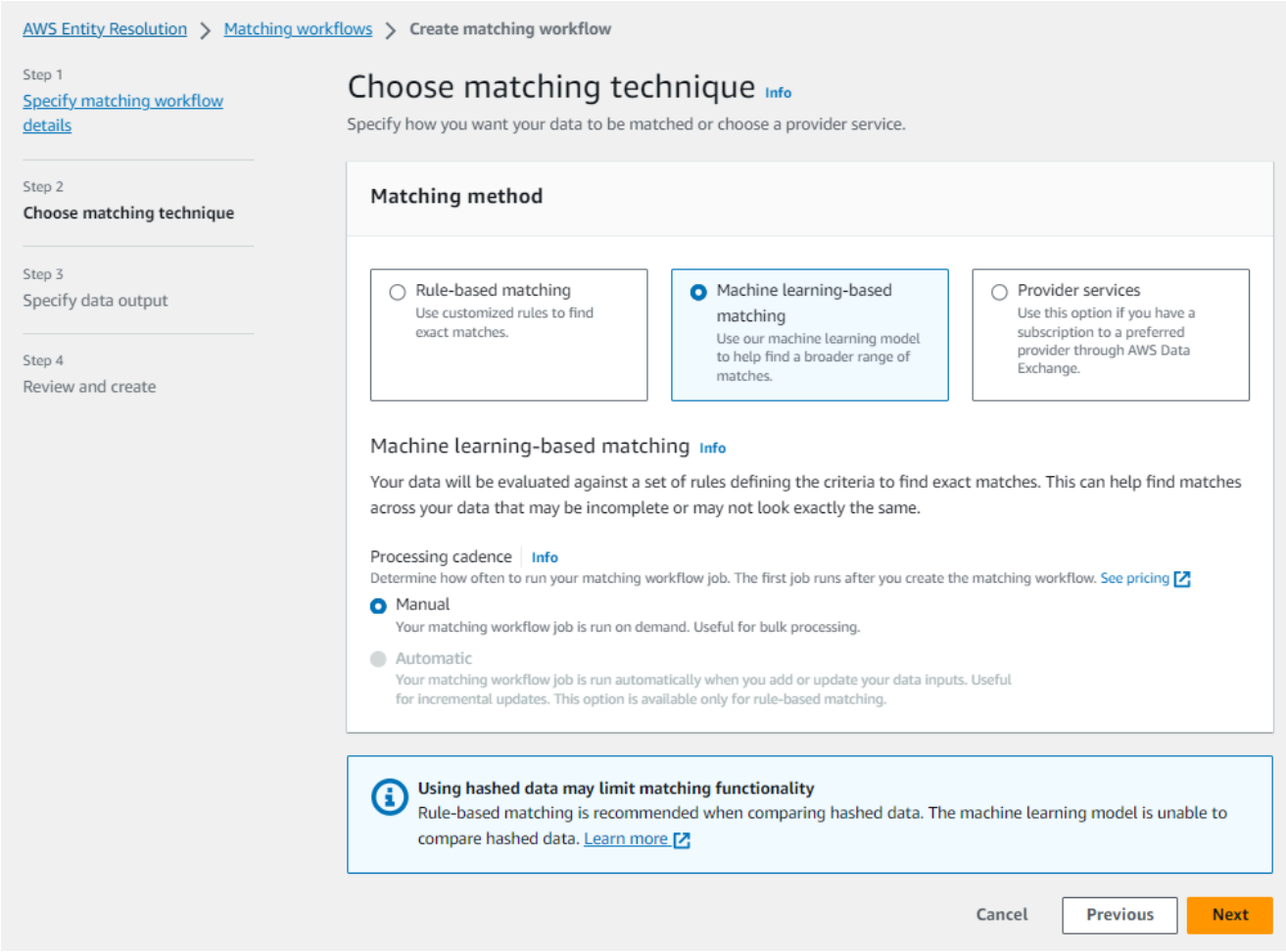
オプション	推奨されるアクション
既存のサービスロールを使用	1. ドロップダウンリストから [既存のサービスロール名] を選択します。 ロールを一覧表示するアクセス許可がある場合は、ロールのリストが表示されます。 ロールを一覧表示するアクセス許可がない場合は、使用するロールの Amazon リソースネーム (ARN) を入力できます。 既存のサービスロールがない場合、[既存のサービスロールを使用] オプションは使用できません。 2. [IAM で表示] 外部リンクを選択してサービスロールを表示します。 デフォルトでは、 AWS Entity Resolution は既存のロールポリシーを更新して必要なアクセス許可を追加しようとしません。

e. (オプション) リソースのタグを有効にするには、新しいタグを追加を選択し、キーと値のペアを入力します。

f. [次へ] を選択します。

5. ステップ 2: 一致する手法を選択するには:

a. マッチング方法 で、機械学習ベースのマッチングを選択します。



b. Processing ケイデンスでは、手動オプションが選択されます。

このオプションを使用すると、一括更新のワークフローをオンデマンドで実行できます。

> **Note**
>
> 自動 (増分) 処理は、機械学習ベースのマッチングワークフローではサポートされていません。

c. [次へ] を選択します。

6. ステップ 3: データ出力と形式を指定するには:

a. データ出力の送信先と形式については、データ出力の Amazon S3 の場所と、データ形式を正規化データまたは元のデータのどちらにするかを選択します。

b. 暗号化 で、暗号化設定をカスタマイズする場合は、AWS KMS キー ARN を入力します。

c. システム生成出力を表示します。

d. データ出力では、含める、非表示にする、またはマスクするフィールドを決定し、目標に基づいて推奨アクションを実行します。

目標	推奨されるオプション
フィールドを含める	出力状態をインクルードのままにします。
フィールドを非表示 (出力から除外)	Output フィールドを選択し、Hide を選択します。
マスクフィールド	出力フィールドを選択し、ハッシュ出力を選択します。
以前の設定をリセットする	[リセット] を選択します。

e. [次へ] を選択します。

7. ステップ 4: 確認して作成する:

a. 前のステップで行った選択内容を確認し、必要に応じて編集します。

b. Create and run を選択します。

一致するワークフローが作成され、ジョブが開始されたことを示すメッセージが表示されます。

8. 一致するワークフローの詳細ページのメトリクスタブで、「最後のジョブメトリクス」で以下を表示します。

- ジョブ ID。
- 一致するワークフロージョブのステータス: Queued、In progress、Completed、Failed
- ワークフロージョブの完了時刻。
- 処理されたレコードの数。
- 処理されていないレコードの数。
- 生成された一意の一致 IDs。
- 入力レコードの数。

ジョブ履歴で以前に実行された一致するワークフロージョブのジョブメトリクスを表示することもできます。

9. 一致するワークフロージョブが完了したら (ステータスが完了)、データ出力タブに移動し、Amazon S3 の場所を選択して結果を表示できます。
10. (手動処理タイプのみ) 手動処理タイプを使用して機械学習ベースのマッチングワークフローを作成した場合は、一致するワークフローの詳細ページでワークフローの実行を選択して、一致するワークフローをいつでも実行できます。

プロバイダーのサービスベースのマッチングワークフローの作成

プロバイダーのサービスベースのマッチングを使用すると、既知の識別子を任意のデータサービスプロバイダーと照合できます。

AWS Entity Resolution は現在、次のデータプロバイダーサービスをサポートしています。

- LiveRamp
- TransUnion
- 統合 ID 2.0

サポートされているプロバイダーサービスの詳細については、「」を参照してくださいサードパーティーの入力データの準備。

これらのプロバイダーのパブリックサブスクリプションを で使用すること AWS Data Exchange も、プライベートオファーをデータプロバイダーと直接交渉することもできます。新しいサブスクリプションの作成またはプロバイダーサービスへの既存のサブスクリプションの再利用の詳細については、「」を参照してくださいステップ 1: でプロバイダーサービスをサブスクライブする AWS Data Exchange。

以下のセクションでは、プロバイダーベースのマッチングワークフローを作成する方法について説明します。

トピック

- LiveRamp を使用した一致するワークフローの作成
- TransUnion を使用した一致するワークフローの作成
- UID 2.0 を使用した一致するワークフローの作成

LiveRamp を使用した一致するワークフローの作成

LiveRamp サービスは、RampID と呼ばれる識別子を提供します。RampID は、広告キャンペーンのオーディエンスを作成するために需要側のプラットフォームで最も一般的に使用される IDs の 1 つです。LiveRamp で一致するワークフローを使用すると、ハッシュ化された E メールアドレスを RAMPIDs に解決できます。

> **Note**
>
> AWS Entity Resolution は PII ベースの RampID 割り当てをサポートしています。

前提条件

LiveRamp で一致するワークフローを作成する前に、以下を行う必要があります。

1. スキーママッピングを作成します。詳細については、「スキーママッピングの作成」を参照してください。
2. LiveRamp サービスのサブスクリプションを取得する
3. 一致するワークフロー出力を一時的に書き込む Amazon S3 データステージングバケットに適切なアクセス許可を設定する

LiveRamp で ID マッピングワークフローを作成する前に、S3 データステージングバケットに次のアクセス許可を追加します。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::715724997226:root"

            },
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
```

```
                "s3:GetObjectVersion",
                "s3:DeleteObject"
            ],
            "Resource": [
                "arn:aws:s3:::<staging-bucket>",
                "arn:aws:s3:::<staging-bucket>/*"
            ]
        },
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::715724997226:root"
            },
            "Action": [
                "s3:ListBucket",
                "s3:GetBucketLocation",
                "s3:GetBucketPolicy",
                "s3:ListBucketVersions",
                "s3:GetBucketAcl"
            ],
            "Resource": [
                "arn:aws:s3:::<staging-bucket>",
                "arn:aws:s3:::<staging-bucket>/*"
            ]
        }
    ]
}
```

各 *<##############>* を独自の情報に置き換えます。

##########	Amazon S3 bucket that temporarily stores your data while running a provider service-based workflow.

LiveRamp で一致するワークフローを作成するには:

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、一致を選択します。

3. 一致するワークフローページの右上隅で、一致するワークフローの作成を選択します。
4. ステップ 1: 一致するワークフローの詳細を指定するには、以下を実行します。

   a. 一致するワークフロー名とオプションの 説明を入力します。

   b. データ入力で、AWS リージョン、AWS Glue データベース、AWS Glue テーブルを選択し、対応するスキーママッピングを選択します。

      最大 20 個のデータ入力を追加できます。

   c. データ正規化オプションはデフォルトで選択され、一致する前にデータ入力が正規化されます。

      > **Note**
      >
      > 正規化は、スキーママッピングの作成で以下のシナリオでのみサポートされています。
      >
      > - 名前サブタイプがグループ化されている場合: 名、ミドルネーム、姓。
      > - 住所サブタイプがグループ化されている場合: 住所 1、住所 2: 住所 3 名、市名、州、国、郵便番号。
      > - 電話番号サブタイプがグループ化されている場合: 電話番号、電話番号の国コード。

      E メールのみの解決プロセスを使用している場合は、データの正規化オプションの選択を解除します。これは、ハッシュ化された E メールのみが入力データに使用されるためです。

   d. サービスアクセス許可を指定するには、 オプションを選択し、推奨アクションを実行します。

オプション	推奨されるアクション
新しいサービスロールを作成して使用	• AWS Entity Resolution は、このテーブルに必要なポリシーを持つサービスロールを作成します。 • デフォルトの [サービスロール名] は `entityresolution-matching-workflow-<timestamp>` です。

オプション	推奨されるアクション
	• ロールを作成してポリシーをアタッチするアクセス許可が必要です。 • 入力データが暗号化されている場合は、このデータは KMS キーオプションで暗号化されます。次に、データ入力の復号に使用される AWS KMS キーを入力します。
既存のサービスロールを使用	1. ドロップダウンリストから [既存のサービスロール名] を選択します。 ロールを一覧表示するアクセス許可がある場合は、ロールのリストが表示されます。 ロールを一覧表示するアクセス許可がない場合は、使用するロールの Amazon リソースネーム (ARN) を入力できます。 既存のサービスロールがない場合、[既存のサービスロールを使用] オプションは使用できません。 2. [IAM で表示] 外部リンクを選択してサービスロールを表示します。 デフォルトでは、 AWS Entity Resolution は既存のロールポリシーを更新して必要なアクセス許可を追加しようとしません。

e. (オプション) リソースのタグを有効にするには、新しいタグを追加を選択し、キーと値のペアを入力します。

f. [次へ] を選択します。

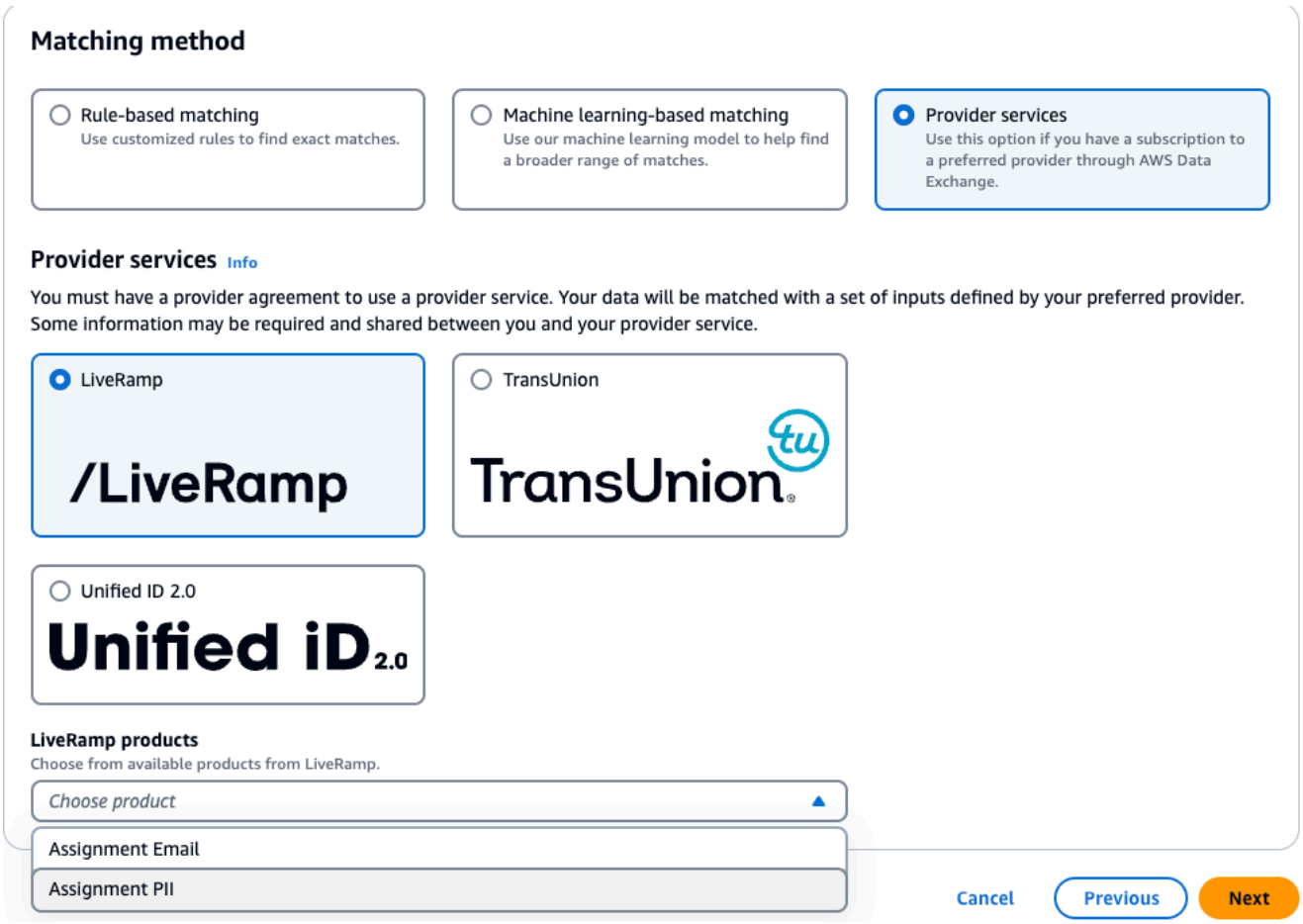
5. ステップ 2: 一致する手法を選択するには:

a. マッチング方法 で、プロバイダーサービスを選択します。

b. プロバイダーサービスで、LiveRamp を選択します。

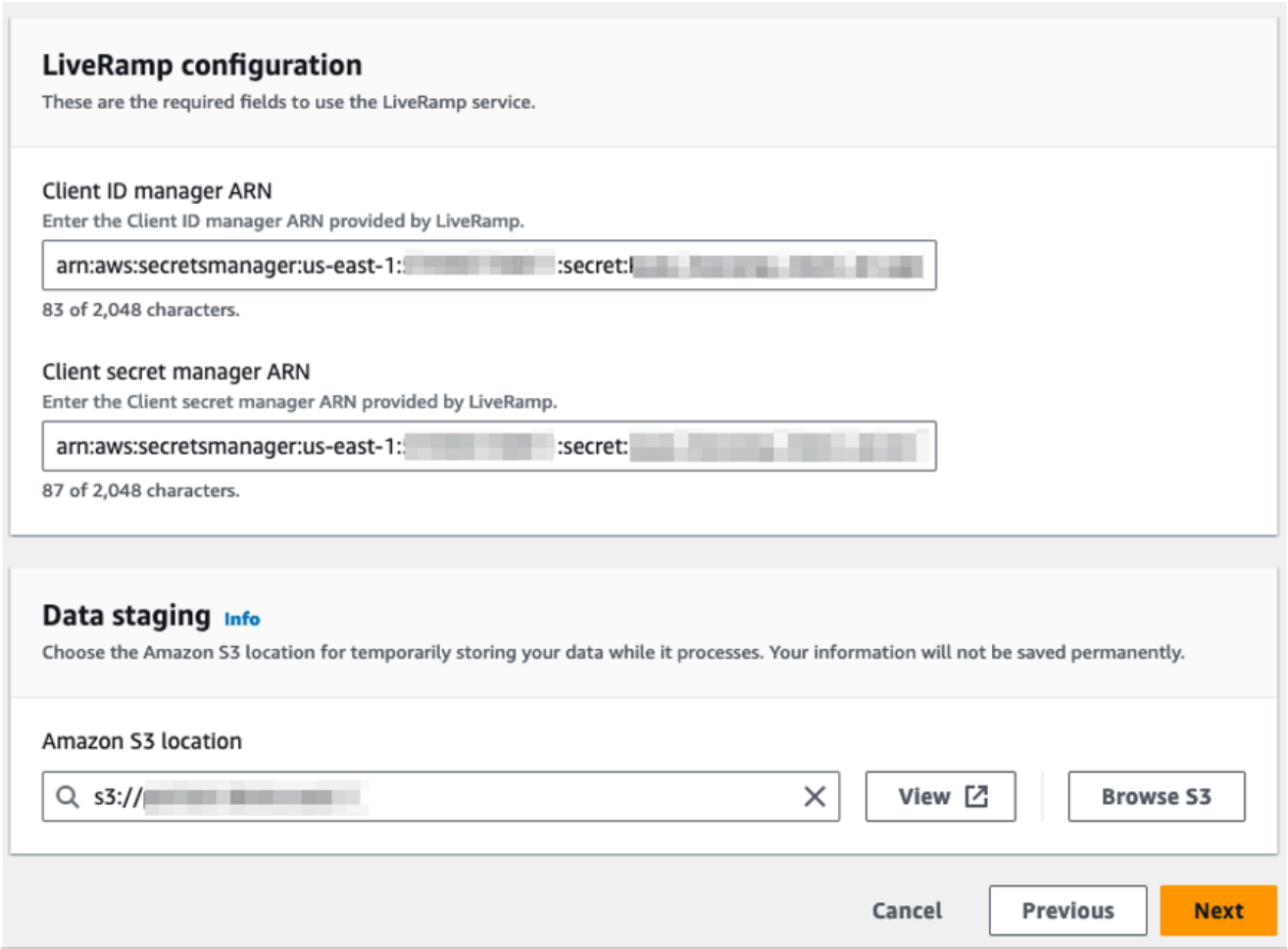
> **Note**
>
> データ入力ファイルの形式と正規化がプロバイダーサービスのガイドラインに沿っていることを確認します。
> 一致するワークフローの入力ファイルフォーマットガイドラインの詳細については、LiveRamp ドキュメントの「ADX によるアイデンティティ解決の実行」を参照してください。

c. LiveRamp 製品の場合は、ドロップダウンリストから製品を選択します。



> **Note**
>
> PII の割り当てを選択した場合は、エンティティ解決を実行するときに、少なくとも 1 つの非識別子列を指定する必要があります。例えば、GENDER などです。

d. LiveRamp 設定には、クライアント ID マネージャー ARN とクライアントシークレットマネージャー ARN を入力します。



e. データステージングでは、処理中のデータの一時ストレージの Amazon S3 の場所を選択します。

データステージング Amazon S3 の場所に対するアクセス許可が必要です。詳細については、「のワークフロージョブロールの作成 AWS Entity Resolution」を参照してください。

f. [Next] (次へ) を選択します。

6. ステップ 3: データ出力を指定するには:

a. データ出力の送信先と形式については、データ出力の Amazon S3 の場所と、データ形式を正規化データまたは元のデータのどちらにするかを選択します。

b. 暗号化 で、暗号化設定をカスタマイズする場合は、AWS KMS キー ARN を入力します。

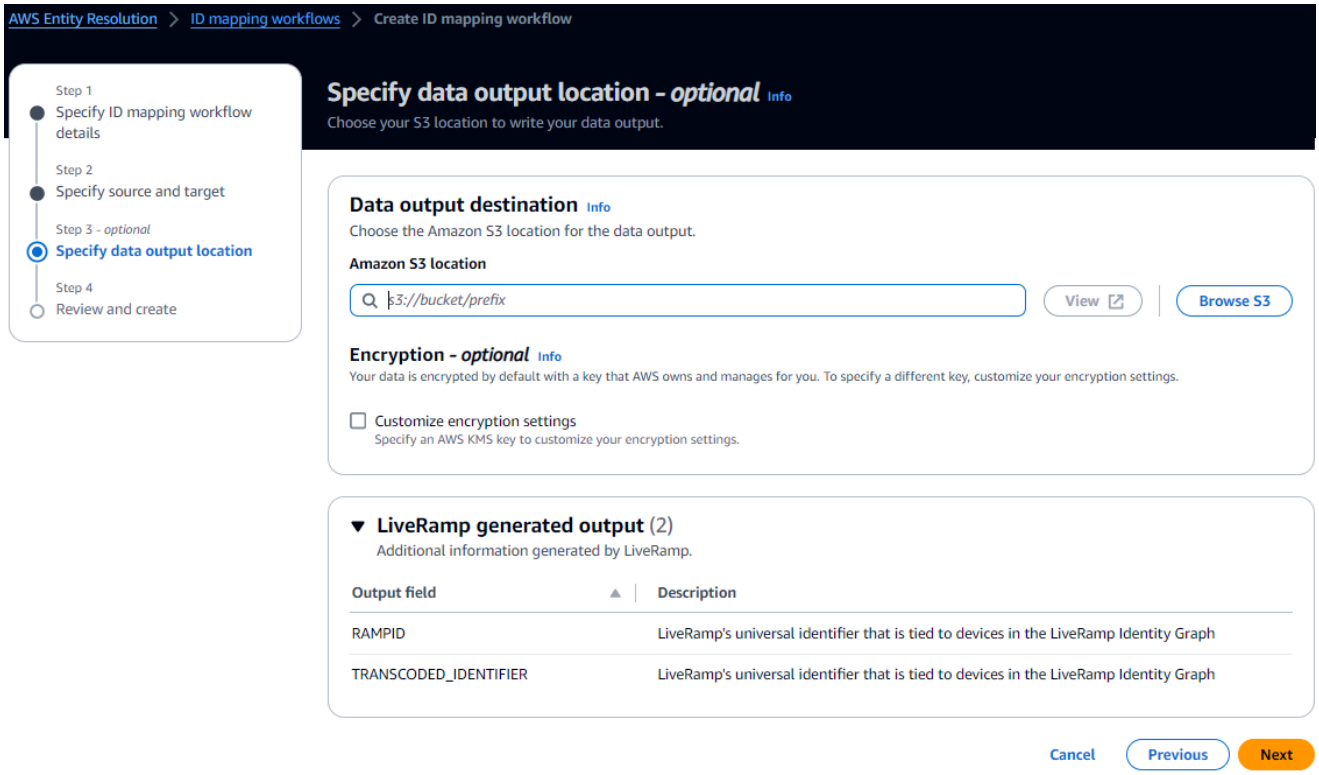
c. LiveRamp が生成した出力を表示します。

これは LiveRamp によって生成された追加情報です。

d. データ出力では、含める、非表示にする、またはマスクするフィールドを決定し、目標に基づいて推奨アクションを実行します。

> **Note**
>
> LiveRamp を選択した場合、個人を特定できる情報 (PII) を削除する LiveRamp プライバシーフィルターにより、一部のフィールドには出力状態が使用不可と表示されます。

目標	推奨されるオプション
フィールドを含める	出力状態をインクルードのままにします。
フィールドを非表示 (出力から除外)	Output フィールドを選択し、Hide を選択します。
マスクフィールド	出力フィールドを選択し、ハッシュ出力を選択します。
以前の設定をリセットする	[リセット] を選択します。



e. [次へ] を選択します。

7. ステップ 4: 確認して作成する:

   a. 前のステップで行った選択内容を確認し、必要に応じて編集します。

   b. Create and run を選択します。

      一致するワークフローが作成され、ジョブが開始されたことを示すメッセージが表示されます。

8. 一致するワークフローの詳細ページのメトリクスタブで、「最後のジョブメトリクス」で以下を表示します。

   - ジョブ ID。
   - 一致するワークフロージョブのステータス: Queued、In progress、Completed、Failed
   - ワークフロージョブの完了時刻。
   - 処理されたレコードの数。
   - 処理されていないレコードの数。
   - 生成された一意の一致 IDs。
   - 入力レコードの数。

ジョブ履歴で以前に実行された一致するワークフロージョブのジョブメトリクスを表示することもできます。

9. 一致するワークフロージョブが完了したら (ステータスが完了)、データ出力タブに移動し、Amazon S3 の場所を選択して結果を表示できます。

TransUnion を使用した一致するワークフローの作成

TransUnion サービスのサブスクリプションをお持ちの場合は、TransUnion Person および Household E Keys と 200 を超えるデータ属性を使用して、さまざまなチャネルに保存された顧客関連のレコードをリンク、照合、強化することで、顧客の理解を向上させることができます。

TransUnion サービスは、TransUnion 個人 ID と世帯 IDsと呼ばれる識別子を提供します。TransUnion は、名前、住所、電話番号、E メールアドレスなどの既知の識別子の ID 割り当て (エンコードとも呼ばれます) を提供します。

前提条件

LiveRamp で一致するワークフローを作成する前に、以下を行う必要があります。

1. スキーママッピングを作成します。詳細については、「スキーママッピングの作成」を参照してください。
2. TransUnion サービスのサブスクリプションを持っている
3. 一致するワークフロー出力を一時的に書き込む Amazon S3 データステージングバケットに適切なアクセス許可を設定する

TransUnion で一致するワークフローを作成する前に、S3 データステージングバケットに次のアクセス許可を追加します。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
```

```
                "AWS": "arn:aws:iam::381491956555:root"

            },
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:GetObjectVersion",
                "s3:DeleteObject"
            ],
            "Resource": [
                "arn:aws:s3:::<staging-bucket>",
                "arn:aws:s3:::<staging-bucket>/*"
            ]
        },
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::381491956555:root"
            },
            "Action": [
                "s3:ListBucket",
                "s3:GetBucketLocation",
                "s3:GetBucketPolicy",
                "s3:ListBucketVersions",
                "s3:GetBucketAcl"
            ],
            "Resource": [
                "arn:aws:s3:::<staging-bucket>",
                "arn:aws:s3:::<staging-bucket>/*"
            ]
        }
    ]
}
```

各 *<###############>* を独自の情報に置き換えます。

##########	Amazon S3 bucket that temporarily stores your data while running a provider service-based workflow.

TransUnion を使用して一致するワークフローを作成するには:

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、一致を選択します。
3. 一致するワークフローページの右上隅で、一致するワークフローの作成を選択します。
4. ステップ 1: 一致するワークフローの詳細を指定するには、以下を実行します。

   a. 一致するワークフロー名とオプションの 説明を入力します。

   b. データ入力で、、AWS リージョンAWS Glue データベース、AWS Glue テーブルを選択し、対応するスキーママッピングを選択します。

      最大 20 個のデータ入力を追加できます。

   c. データ正規化オプションはデフォルトで選択され、一致する前にデータ入力が正規化されます。データを正規化しない場合は、データの正規化オプションの選択を解除します。

      > **Note**
      >
      > 正規化は、スキーママッピングの作成で以下のシナリオでのみサポートされています。
      >
      > - 名前サブタイプがグループ化されている場合: 名、ミドルネーム、姓。
      > - 次のアドレスサブタイプがグループ化されている場合: 住所 1、住所 2: 住所 3 名、市名、州、国、郵便番号。
      > - 電話番号サブタイプがグループ化されている場合: 電話番号、電話番号の国コード。

   d. サービスアクセス許可を指定するには、 オプションを選択し、推奨アクションを実行します。

オプション	推奨されるアクション
新しいサービスロールを作成して使用	• AWS Entity Resolution は、このテーブルに必要なポリシーを持つサービスロールを作成します。

オプション	推奨されるアクション
	• デフォルトの [サービスロール名] は `entityresolution-matching-workflow-<timestamp>` です。 • ロールを作成してポリシーをアタッチするアクセス許可が必要です。 • 入力データが暗号化されている場合は、このデータは KMS キーオプションで暗号化されます。次に、データ入力の復号に使用される AWS KMS キーを入力します。
既存のサービスロールを使用	1. ドロップダウンリストから [既存のサービスロール名] を選択します。 ロールを一覧表示するアクセス許可がある場合は、ロールのリストが表示されます。 ロールを一覧表示するアクセス許可がない場合は、使用するロールの Amazon リソースネーム (ARN) を入力できます。 既存のサービスロールがない場合、[既存のサービスロールを使用] オプションは使用できません。 2. [IAM で表示] 外部リンクを選択してサービスロールを表示します。 デフォルトでは、 AWS Entity Resolution は既存のロールポリシーを更新して必要なアクセス許可を追加しようとしません。

e. (オプション) リソースのタグを有効にするには、新しいタグを追加を選択し、キーと値のペアを入力します。

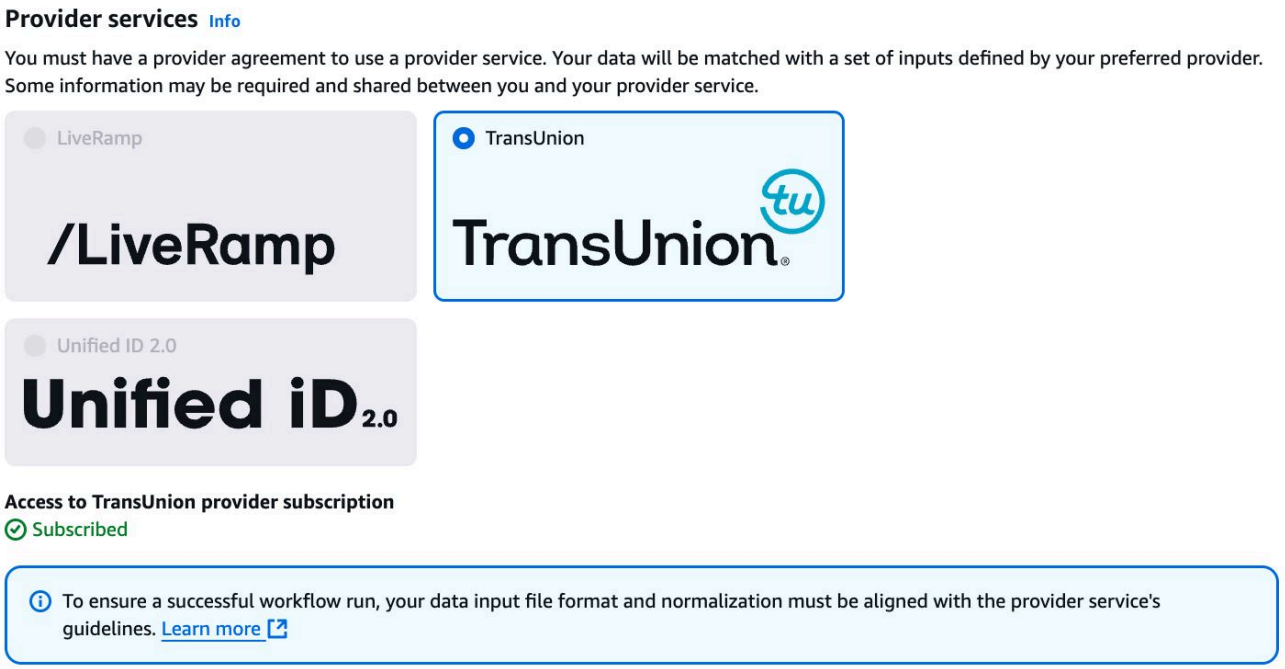
f. [次へ] を選択します。

5. ステップ 2: 一致する手法を選択するには:

a. マッチング方法 で、プロバイダーサービスを選択します。

b. プロバイダーサービスで、TransUnion を選択します。

> **Note**
>
> データ入力ファイルの形式と正規化がプロバイダーサービスのガイドラインに沿っていることを確認します。



c. データステージングでは、処理中のデータの一時ストレージの Amazon S3 の場所を選択します。

データステージング Amazon S3 の場所へのアクセス許可が必要です。詳細については、「the section called “ワークフロージョブロールの作成”」を参照してください。

6. [Next] (次へ) を選択します。

7. ステップ 3: データ出力を指定するには:

a. データ出力の送信先と形式については、データ出力の Amazon S3 の場所と、データ形式を正規化データまたは元のデータのどちらにするかを選択します。

b. 暗号化 で、暗号化設定をカスタマイズする場合は、AWS KMS キー ARN を入力します。

c. TransUnion が生成した出力を表示します。

これは、TransUnion によって生成された追加情報です。

d. データ出力では、含める、非表示にする、またはマスクするフィールドを決定し、目標に基づいて推奨アクションを実行します。

目標	推奨されるオプション
フィールドを含める	出力状態をインクルードのままにします。
フィールドを非表示 (出力から除外)	Output フィールドを選択し、Hide を選択します。
マスクフィールド	出力フィールドを選択し、ハッシュ出力を選択します。
以前の設定をリセットする	[リセット] を選択します。

e. システム生成出力の場合、含まれているすべてのフィールドを表示します。

f. [次へ] を選択します。

8. ステップ 4: 確認して作成する:

a. 前のステップで行った選択内容を確認し、必要に応じて編集します。

b. Create and run を選択します。

一致するワークフローが作成され、ジョブが開始されたことを示すメッセージが表示されます。

9. 一致するワークフローの詳細ページのメトリクスタブで、「最後のジョブメトリクス」で以下を表示します。

- ジョブ ID。
- 一致するワークフロージョブのステータス: Queued、In progress、Completed、Failed
- ワークフロージョブの完了時刻。
- 処理されたレコードの数。
- 処理されていないレコードの数。
- 生成された一意の一致 IDs。

- 入力レコードの数。

ジョブ履歴で以前に実行された一致するワークフロージョブのジョブメトリクスを表示することもできます。

10. 一致するワークフロージョブが完了したら (ステータスが完了)、データ出力タブに移動し、Amazon S3 の場所を選択して結果を表示できます。

UID 2.0 を使用した一致するワークフローの作成

Unified ID 2.0 サービスのサブスクリプションをお持ちの場合は、決定論的アイデンティティを持つ広告キャンペーンをアクティブ化し、広告エコシステム全体の多くの UID2-enabled参加者との相互運用性に頼ることができます。詳細については、「Unified ID 2.0 Overview」を参照してください。

Unified ID 2.0 サービスは raw UID 2 を提供します。これは、Trade Desk プラットフォームでの広告キャンペーンの構築に使用されます。UID 2.0 は、オープンソースフレームワークを使用して生成されます。

1 つのワークフローでは、未加工の UID2 生成**Phone number**に **Email Address**または のいずれかを使用できますが、両方を使用することはできません。両方がスキーママッピングに存在する場合、ワークフローは を選択し**Email Address**、 はパススルーフィールド**Phone number**になります。両方をサポートするには、 がマッピング**Phone number**されているが、 **Email Address**がマッピングされていない新しいスキーママッピングを作成します。次に、この新しいスキーママッピングを使用して 2 番目のワークフローを作成します。

> **Note**
>
> Raw UID2sは、1 年に約 1 回ローテーションされるソルトバケットからソルトを追加することで作成され、それに伴って raw UID2 もローテーションされます。したがって、未加工の UID2sを毎日更新することをお勧めします。詳細については、https://unifiedid.com/docs/getting-started/gs-faqs#how-often-should-uid2s-be-refreshed-for-incremental-updates を参照してください。

前提条件

UID 2.0 で一致するワークフローを作成する前に、以下を行う必要があります。

1. スキーママッピングを作成します。詳細については、「スキーママッピングの作成」を参照してください。
2. UID 2.0 サービスのサブスクリプションを持っている

UID 2.0 を使用して一致するワークフローを作成するには:

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、一致を選択します。
3. 一致するワークフローページの右上隅で、一致するワークフローの作成を選択します。
4. ステップ 1: 一致するワークフローの詳細を指定するには、以下を実行します。

   a. 一致するワークフロー名とオプションの 説明を入力します。

   b. データ入力で、、AWS リージョンAWS Glue データベース、AWS Glue テーブルを選択し、対応するスキーママッピングを選択します。

      最大 20 個のデータ入力を追加できます。

   c. データ正規化オプションを選択したままにして、一致する前にデータ入力 (**Email Address** または **Phone number**) を正規化します。

      Email Address 正規化の詳細については、UID 2.0 ドキュメントの「E メールアドレスの正規化」を参照してください。

      Phone number 正規化の詳細については、UID 2.0 ドキュメントの「電話番号の正規化」を参照してください。

   d. サービスアクセス許可を指定するには、 オプションを選択し、推奨アクションを実行します。

オプション	推奨されるアクション
新しいサービスロールを作成して使用	• AWS Entity Resolution は、このテーブルに必要なポリシーを持つサービスロールを作成します。 • デフォルトの [サービスロール名] は `entityresolution-matching-workflow-<timestamp>` です。

オプション	推奨されるアクション
	• ロールを作成してポリシーをアタッチするアクセス許可が必要です。 • 入力データが暗号化されている場合は、このデータは KMS キーオプションで暗号化されます。次に、データ入力の復号に使用される AWS KMS キーを入力します。
既存のサービスロールを使用	1. ドロップダウンリストから [既存のサービスロール名] を選択します。 ロールを一覧表示するアクセス許可がある場合は、ロールのリストが表示されます。 ロールを一覧表示するアクセス許可がない場合は、使用するロールの Amazon リソースネーム (ARN) を入力できます。 既存のサービスロールがない場合、[既存のサービスロールを使用] オプションは使用できません。 2. [IAM で表示] 外部リンクを選択してサービスロールを表示します。 デフォルトでは、 AWS Entity Resolution は既存のロールポリシーを更新して必要なアクセス許可を追加しようとしません。

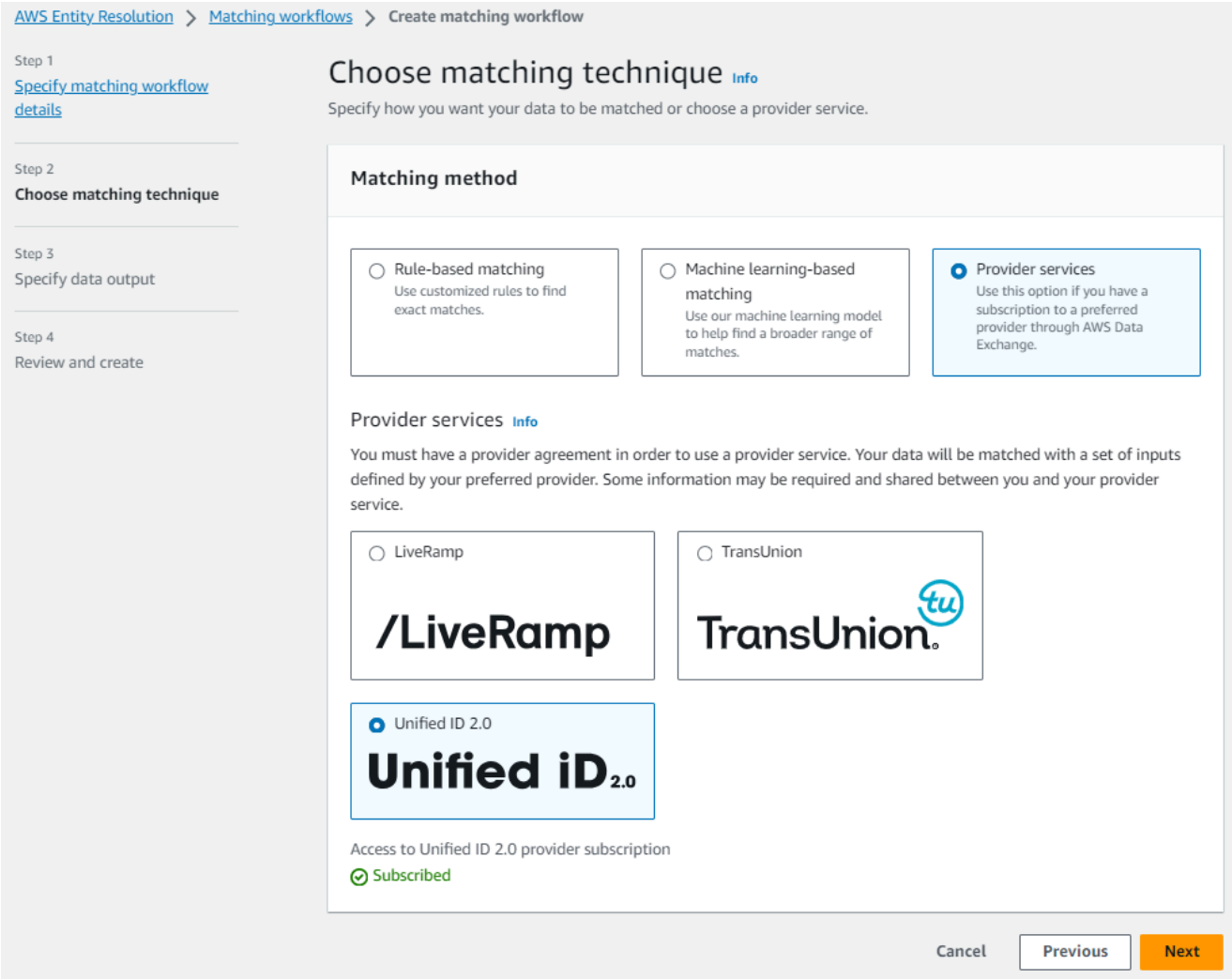
e. (オプション) リソースのタグを有効にするには、新しいタグを追加を選択し、キーと値のペアを入力します。

f. [次へ] を選択します。

5. ステップ 2: 一致する手法を選択するには:

a. マッチング方法 で、プロバイダーサービスを選択します。

b. プロバイダーサービスで、Unified ID 2.0 を選択します。



c. [次へ] を選択します。

6. ステップ 3: データ出力を指定するには:

a. データ出力の送信先と形式については、データ出力の Amazon S3 の場所と、データ形式を正規化データまたは元のデータのどちらにするかを選択します。

b. 暗号化 で、暗号化設定をカスタマイズする場合は、AWS KMS キー ARN を入力します。

c. Unified ID 2.0 で生成された出力を表示します。

これは、UID 2.0 によって生成されたすべての追加情報のリストです。

d. データ出力では、含める、非表示にする、またはマスクするフィールドを決定し、目標に基づいて推奨アクションを実行します。

目標	推奨されるオプション
フィールドを含める	出力状態をインクルードのままにします。
フィールドを非表示 (出力から除外)	Output フィールドを選択し、Hide を選択します。
マスクフィールド	出力フィールドを選択し、ハッシュ出力を選択します。
以前の設定をリセットする	[リセット] を選択します。

e. システム生成出力の場合、含まれているすべてのフィールドを表示します。

f. [次へ] を選択します。

7. ステップ 4: 確認して作成する:

a. 前のステップで行った選択内容を確認し、必要に応じて編集します。

b. Create and run を選択します。

一致するワークフローが作成され、ジョブが開始されたことを示すメッセージが表示されます。

8. 一致するワークフローの詳細ページのメトリクスタブで、「最後のジョブメトリクス」で以下を表示します。

- ジョブ ID。
- 一致するワークフロージョブのステータス: Queued、In progress、Completed、Failed
- ワークフロージョブの完了時刻。
- 処理されたレコードの数。
- 処理されていないレコードの数。
- 生成された一意の一致 IDs。
- 入力レコードの数。

ジョブ履歴で以前に実行された一致するワークフロージョブのジョブメトリクスを表示することもできます。

9. 一致するワークフロージョブが完了したら (ステータスが完了)、データ出力タブに移動し、Amazon S3 の場所を選択して結果を表示できます。

一致するワークフローの編集

一致するワークフローを編集すると、エンティティ解決プロセスをup-to-date状態に保ち、時間の経過とともに変化する組織の要件に対応できます。エンティティ解決プロセスの精度と効率を向上させるために、一致する基準、手法、またはデータ出力を調整することができます。現在のワークフローの結果で問題やエラーを特定した場合、編集すると、それらの問題を診断して解決するのに役立ちます。

一致するワークフローを編集するには:

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、一致を選択します。
3. 一致するワークフローを選択します。
4. 一致するワークフローの詳細ページの右上隅で、ワークフローの編集を選択します。
5. 一致するワークフローの詳細を指定ページで、必要な変更を加え、次へを選択します。
6. 一致する手法の選択ページで、必要な変更を加え、次へを選択します。

> **Important**
>
> 処理頻度は手動から自動に変更できますが、自動に変更した後は手動に戻すことはできません。
> 処理ケイデンスがすでに自動に設定されている場合、手動に変更することはできません。

7. データ出力を指定ページで、必要な変更を加え、次へを選択します。
8. 確認と保存ページで、必要な変更を加え、保存を選択します。

一致するワークフローの削除

一致するワークフローが使用されなくなったり、古くなったりした場合、それを削除すると、ワークスペースを整理して整頓しておくのに役立ちます。古いワークフローを置き換える改善された新しい

ワークフローを開発した場合、古いワークフローを削除すると、up-to-dateプロセスのみを使用できるようになります。

一致するワークフローを削除するには:

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、一致を選択します。
3. 一致するワークフローを選択します。
4. 一致するワークフローの詳細ページで、右上隅にある「削除」を選択します。
5. 削除を確定し、[削除] を選択します。

ルールベースの一致ワークフローの一致 ID の変更または生成

一致 ID は、一致するワークフローの実行後に によって生成 AWS Entity Resolution され、一致する各レコードセットに適用される識別子です。これは、出力に含まれる一致するワークフローメタデータの一部です。

既存の顧客のレコードを更新したり、データセットに新しい顧客を追加したりする必要がある場合は、 AWS Entity Resolution コンソールまたは `GenerateMatchID` API を使用できます。既存の一致 ID を変更すると、顧客情報を更新する際の一貫性を維持できますが、以前に識別されていない顧客をシステムに追加する場合は新しい一致 ID を生成する必要があります。

> **Note**
>
> コンソールと API のどちらを使用する場合でも、追加料金が適用されます。選択した処理タイプは、オペレーションの精度と応答時間の両方に影響します。

> **Important**
>
> ジョブの進行中に S3 バケットへのアクセス AWS Entity Resolution 許可を取り消すと、AWS Entity Resolution は引き続き結果を S3 に出力するための処理と課金を行いますが、結果をバケットに配信することはできません。この問題を回避するには、ジョブを開始する前に、 に S3 バケットに書き込むための正しいアクセス許可 AWS Entity Resolution があることを確認してください。処理中にアクセス許可が取り消された場合、 AWS Entity Resolution

正しいバケットアクセス許可を復元すると、 はジョブの完了から最大 30 日間、結果の再配信を試みます。

次の手順では、一致 ID を検索または生成し、処理タイプを選択し、結果を表示するプロセスについて説明します。

Console

コンソールを使用して一致 ID を変更または生成するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、一致を選択します。
3. 処理されたルールベースのマッチングワークフローを選択します (ジョブのステータスは完了です)。
4. 一致するワークフローの詳細ページで、一致 IDsタブを選択します。
5. 一致 ID の変更または生成を選択します。

> **Note**
>
> 一致 ID の変更または生成オプションは、自動処理頻度を使用するワークフローのマッチングでのみ使用できます。手動処理頻度を選択した場合、このオプションは非アクティブになります。このオプションを使用するには、ワークフローを編集して自動処理の頻度を使用します。ワークフローの編集の詳細については、「」を参照してください一致するワークフローの編集。

6. ドロップダウンリストからAWS Glue テーブルを選択します。

   ワークフローに AWS Glue テーブルが 1 つしかない場合は、デフォルトで選択されます。

7. 処理タイプを選択します。

   - 一貫性 – 既存の一致 ID を検索するか、新しい一致 ID をすぐに生成して保存できます。このオプションは、最も精度が高く、応答時間が遅くなります。
   - 背景 (API EVENTUALで と表示) – 既存の一致 ID を検索したり、新しい一致 ID をすぐに生成したりできます。更新されたレコードはバックグラウンドで保存されます。このオプションの初期応答は高速で、後で S3 で完全な結果を利用できます。

- クイック ID 生成 (API EVENTUAL_NO_LOOKUPで と表示) – 既存の一致 ID を検索せずに新しい一致 ID を作成できます。更新されたレコードはバックグラウンドで保存されます。このオプションは最速の応答です。一意のレコードにのみ推奨されます。

8. レコード属性の場合、

   a. 一意の ID の値を入力します。

   b. ワークフローで設定されたルールに基づいて、既存のレコードと一致する各一致キーの値を入力します。

9. 一致 ID を検索 を選択し、レコードを保存します。

   一致 ID が見つかったか、新しい一致 ID が生成され、レコードが保存されたことを示す成功メッセージが表示されます。

10. 対応する一致 ID と、一致するワークフローに保存された関連するルールを成功メッセージで表示します。

11. (オプション) 一致 ID をコピーするには、コピーを選択します。

API

API を使用して一致 ID を変更または生成するには

> **Note**
>
> この API を正常に呼び出すには、まず StartMatchingJob API を使用してルールベースのマッチングワークフローを正常に実行する必要があります。
> サポートされているプログラミング言語の完全なリストについては、GenerateMatchID のhttps://docs.aws.amazon.com/entityresolution/latest/apireference/API_GenerateMatchId.html#API_GenerateMatchId_SeeAlso「」セクションを参照してください。

1. ターミナルまたはコマンドプロンプトを開いて API リクエストを行います。

2. 次のエンドポイントへの POST リクエストを作成します。

```
/matchingworkflows/workflowName/generateMatches
```

3. リクエストヘッダーで、Content-type を application/json に設定します。

4. リクエスト URI で、 を指定しますworkflowName。

   は以下workflowNameを行う必要があります。

   - 1～255 文字の長さ
   - パターンに一致 [a-zA-Z_0-9-]*

5. リクエスト本文には、次の JSON を指定します。

```
{
   "processingType": "string",
   "records": [
      {
         "inputSourceARN": "string",
         "recordAttributeMap": {
            "string" : "string"
         },
         "uniqueId": "string"
      }
   ]
}
```

   コードの説明は以下のとおりです。

   - processingType (オプション) - デフォルトは ですCONSISTENT。次のいずれかの値を選択します。
     - CONSISTENT - 応答時間が遅く、最高の精度を実現
     - EVENTUAL - バックグラウンド処理による初期レスポンスの高速化
     - EVENTUAL_NO_LOOKUP - レコードが一意であることがわかっている場合の迅速な対応
   - records (必須) - 1 つのレコードオブジェクトのみを含む配列

6. リクエストを送信します。

   成功すると、ステータスコード 200 と以下を含む JSON 本文を含むレスポンスを受け取ります。

```
{
   "failedRecords": [
      {
         "errorMessage": "string",
         "inputSourceARN": "string",
```

```
            "uniqueId": "string"
         }
      ],
      "matchGroups": [
         {
            "matchId": "string",
            "matchRule": "string",
            "records": [
               {
                  "inputSourceARN": "string",
                  "recordId": "string"
               }
            ]
         }
      ]
}
```

呼び出しが失敗すると、次のいずれかのエラーが表示されることがあります。

- 403 - 十分なアクセスがない場合の AccessDeniedException
- 404 - リソースが見つからない場合の ResourceNotFoundException
- 429 - リクエストがスロットリングされた場合の ThrottlingException
- 400 - 入力が検証に失敗した場合の ValidationException
- 500 - 内部サービスに障害が発生した場合の InternalServerException

ルールベースの一致ワークフローの一致 ID を検索する

ルールベースのマッチングワークフローが完了したら、処理された各レコードの一致 ID と関連するルールを取得できます。この情報は、レコードがどのように照合され、どのルールが適用されたかを理解するのに役立ちます。次の手順は、 AWS Entity Resolution コンソールまたは `GetMatchID` API を使用してこのデータにアクセスする方法を示しています。

Console

コンソールを使用して一致 ID を検索するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、一致を選択します。

3. 処理されたルールベースのマッチングワークフローを選択します (ジョブのステータスは完了です)。
4. 一致するワークフローの詳細ページで、一致 IDsタブを選択します。
5. 一致 ID の検索 を選択します。

> **Note**
>
> Look up match ID オプションは、自動処理頻度を使用するマッチングワークフローでのみ使用できます。手動処理頻度を選択した場合、このオプションは非アクティブになります。このオプションを使用するには、ワークフローを編集して自動処理の頻度を使用します。ワークフローの編集の詳細については、「」を参照してください一致するワークフローの編集。

6. 次のいずれかを行います。

状況	結果
このワークフローに関連付けられているスキーママッピングは 1 つだけです。	デフォルトでは選択されているスキーママッピングを表示します。
このワークフローには複数のスキーママッピングが関連付けられています。	ドロップダウンリストからスキーママッピングを選択します。

7. レコード属性には、既存の各レコードを検索する既存の一致キーの値を入力します。

> **Tip**
>
> 一致 ID を見つけるためにできるだけ多くの値を入力します。

8. データの正規化オプションはデフォルトで選択され、一致する前にデータ入力が正規化されます。データを正規化しない場合は、データの正規化オプションの選択を解除します。
9. 一致するルールを表示する場合は、一致するルールの表示を展開します。
10. [検索] を選択します。

    一致 ID が見つかったことを示す成功メッセージが表示されます。
11. 対応する一致 ID と、見つかった関連するルールを表示します。

API

API を使用して一致 ID を検索するには

> **Note**
>
> この API を正常に呼び出すには、まず StartMatchingJob API を使用してルールベースのマッチングワークフローを正常に実行する必要があります。
> サポートされているプログラミング言語の完全なリストについては、GetMatchID API のhttps://docs.aws.amazon.com/entityresolution/latest/apireference/API_GetMatchId.html#API_GetMatchId_SeeAlso「」セクションを参照してください。

1. ターミナルまたはコマンドプロンプトを開いて API リクエストを行います。
2. 次のエンドポイントへの POST リクエストを作成します。

```
/matchingworkflows/workflowName/matches
```

3. リクエストヘッダーで、Content-type を application/json に設定します。
4. リクエスト URI で、 を指定しますworkflowName。

   は以下workflowNameを行う必要があります。

   - 1～255 文字の長さ
   - パターンに一致 [a-zA-Z_0-9-]*
5. リクエスト本文には、次の JSON を指定します。

```
{
   "applyNormalization": boolean,
   "record": {
      "string" : "string"
   }
}
```

   コードの説明は以下のとおりです。

   applyNormalization (オプション) - を に設定trueして、スキーマで定義された属性を正規化します。

record (必須) - の一致 ID を取得するレコード

6. リクエストを送信します。

   成功すると、ステータスコード 200 と以下を含む JSON 本文を含むレスポンスを受け取ります。

```
{
   "matchId": "string",
   "matchRule": "string"
}
```

   matchId は、一致したレコードのこのグループの一意の識別子であり、レコードが一致したルールmatchRuleを示します。

   呼び出しが失敗すると、次のいずれかのエラーが表示されることがあります。

   - 403 - 十分なアクセスがない場合の AccessDeniedException
   - 404 - リソースが見つからない場合の ResourceNotFoundException
   - 429 - リクエストがスロットリングされた場合の ThrottlingException
   - 400 - 入力が検証に失敗した場合の ValidationException
   - 500 - 内部サービスに障害が発生した場合の InternalServerException

ルールベースまたは ML ベースのマッチングワークフローからのレコードの削除

データ管理規制に準拠する必要がある場合は、ルールベースまたは ML ベースのマッチングワークフローからレコードを削除できます。

ルールベースまたは ML ベースのマッチングワークフローからレコードを削除するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、一致を選択します。
3. ルールベースまたは ML ベースのマッチングワークフローを選択します。
4. 一致するワークフローの詳細ページで、アクションドロップダウンリストから一意の IDs の削除 を選択します。

5. 削除する一意の ID を一意の IDsセクションに入力します。

   最大 10 IDs を入力できます。
6. 一意の IDs を削除する入力ソースを指定します。

   ワークフローの入力ソースが 1 つしかない場合、入力ソースはデフォルトで一覧表示されます。

   1 つの入力ソースのみを指定した場合、他の入力ソースIDs は影響を受けません。
7. 一意の IDs の削除 を選択します。

マッチングワークフローのトラブルシューティング

次の情報は、一致するワークフローの実行時に発生する可能性がある一般的な問題の診断と修正に役立ちます。

一致するワークフローを実行した後にエラーファイルを受け取った

一般的な原因

一致するワークフローには複数の実行を含めることができ、結果 (成功またはエラー) は を名前`jobId`とするフォルダに書き込まれます。

一致するワークフローの成功結果は、複数のファイルを含む`success`フォルダに書き込まれ、各ファイルには成功したレコードのサブセットが含まれます。

一致するワークフローのエラーは、複数のフィールドを持つ`error`フォルダに書き込まれ、それぞれにエラーレコードのサブセットが含まれます。

エラーファイルは、次の理由で作成できます。

- 一意の ID は次のとおりです。
  - null
  - データの行に がない
  - データテーブルのレコードに がない
  - データテーブル内の別の行のデータで繰り返される
  - 指定されていません
  - 同じソース内で一意ではない

- 複数のソース間で一意ではない
- ソース間で重複する
- が 38 文字を超えている (ルールベースのマッチングワークフローのみ)

- スキーママッピングのフィールドの 1 つに予約名が含まれています。
  - EmailAddress
  - InputSourceARN
  - MatchRule
  - MatchID
  - HashingProtocol
  - ConfidenceLevel
  - ソース

> **Note**
>
> 前述の理由でエラーファイルのレコードが作成された場合、サービスの処理コストが発生するため、料金が発生します。エラーファイルのレコードが内部サーバーエラーによるものである場合、料金は発生しません。

解決策

この問題を解決するには

1. 一意の ID が有効かどうかを確認します。

   一意の ID が有効でない場合は、データテーブルの一意の ID を更新し、新しいデータテーブルを保存して新しいスキーママッピングを作成し、一致するワークフローを再度実行します。
2. スキーママッピングのフィールドの 1 つに予約名が含まれているかどうかを確認します。

   いずれかのフィールドに予約名が含まれている場合は、新しい名前で新しいスキーママッピングを作成し、一致するワークフローを再度実行します。

Map input data using an ID mapping workflow

ID マッピングワークフローは、指定された ID マッピング方法に基づいて、入力データソースから入力データターゲットにデータをマッピングするデータ処理ジョブです。これにより、ID マッピングテーブルが生成されます。

ID マッピングワークフローには、入力データソースと入力データターゲットが必要です。データ入力ソースとターゲットは、実行する ID マッピングのタイプによって異なります。ID マッピングを実行するには、ルールベースまたはプロバイダーサービスの 2 つの方法があります。

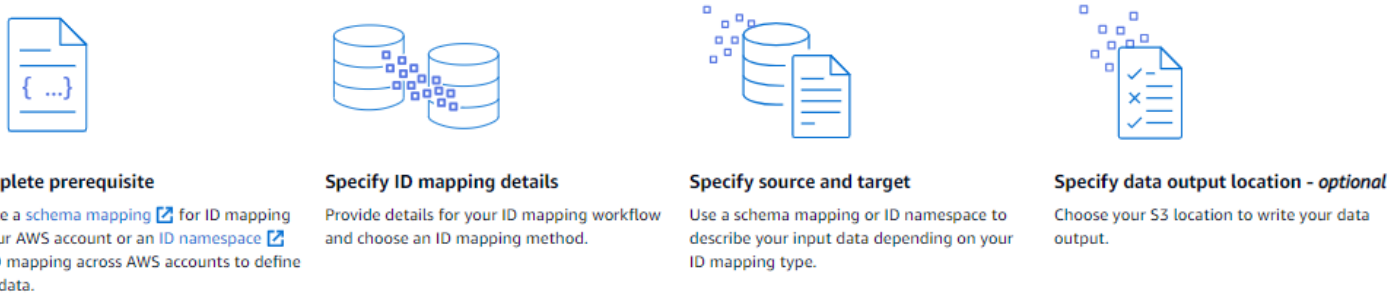
- ルールベースの ID マッピング – 一致するルールを使用して、ソースからターゲットにファーストパーティーデータを変換します。
- プロバイダーサービス ID マッピング – LiveRamp プロバイダーサービスを使用して、ソースからターゲットにサードパーティーデータを変換します。

> **Note**
>
> のプロバイダーサービス ID マッピングワークフロー AWS Entity Resolution は現在 LiveRamp と統合されています。LiveRamp サービスのサブスクリプションをお持ちの場合は、LiveRamp を使用して ID マッピングワークフローを作成して、トランスコードを実行できます。LiveRamp トランスコーディングを使用すると、ソース RampIDs のセットを任意のターゲット先 RampID に変換できます。RampID をトークンとして使用して顧客を表すことで、顧客データを広告プラットフォームと直接共有することを回避できます。詳細については、LiveRamp ドキュメントウェブサイトの「ADX による翻訳の実行」を参照してください。

次のシナリオのいずれかで、2 つのデータセット間で ID マッピングを実行できます。

- 独自の 内 AWS アカウント
- 2 つの異なる AWS アカウント

次の図は、ID マッピングワークフローを設定する方法をまとめたものです。



トピック

- 1 つの の ID マッピングワークフロー AWS アカウント
- 2 つの にわたる ID マッピングワークフロー AWS アカウント
- ID マッピングワークフローの実行
- カスタム ID マッピングワークフローの実行
- ID マッピングワークフローの編集
- ID マッピングワークフローの削除
- ID マッピングワークフローのリソースポリシーの追加または更新

1 つの の ID マッピングワークフロー AWS アカウント

1 つの ID マッピングワークフロー AWS アカウントを使用すると、2 つのデータセット間で独自の ID マッピングを実行できます AWS アカウント。

ID マッピングワークフローを自分で作成する前に AWS アカウント、まず前提条件を完了する必要があります。

ID マッピングワークフローを作成して実行したら、出力 (ID マッピングテーブル) を表示し、分析に使用できます。

以下のトピックでは、同じ で ID マッピングワークフローを作成する一連のステップについて説明します AWS アカウント。

トピック

- 前提条件
- ID マッピングワークフローの作成 (ルールベース)
- ID マッピングワークフローの作成 (プロバイダーサービス)

前提条件

ルールベースまたはプロバイダーサービス ID マッピング方法 AWS アカウント を使用して ID マッピングワークフローを作成する前に、まず以下を実行する必要があります。

- 「セットアップ AWS Entity Resolution」のタスクを完了します。
- 使用している入力データのタイプに応じて入力データテーブルを準備する、 のタスクを完了します。
- スキーママッピングを作成するか、一致するワークフローを作成します。
- (プロバイダーサービス ID マッピングのみ) LiveRamp で ID マッピングワークフローを作成する前に、ID マッピングワークフロー出力を一時的に書き込む Amazon Simple Storage Service (Amazon S3) データステージングバケットを選択する必要があります。

  LiveRamp プロバイダーサービスを使用してサードパーティーデータを翻訳する場合は、次のアクセス許可ポリシーを追加します。これにより、データステージングバケットにアクセスできます。

  JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::715724997226:root"

            },
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:GetObjectVersion",
                "s3:DeleteObject"
            ],
            "Resource": [
                "arn:aws:s3:::<staging-bucket>",
                "arn:aws:s3:::<staging-bucket>/*"
            ]
        },
        {
            "Effect": "Allow",
            "Principal": {
```

```
                "AWS": "arn:aws:iam::715724997226:root"
            },
            "Action": [
                "s3:ListBucket",
                "s3:GetBucketLocation",
                "s3:GetBucketPolicy",
                "s3:ListBucketVersions",
                "s3:GetBucketAcl"
            ],
            "Resource": [
                "arn:aws:s3:::<staging-bucket>",
                "arn:aws:s3:::<staging-bucket>/*"
            ]
        }
    ]
}
```

上記のアクセス許可ポリシーで、各 *<##############>* を独自の情報に置き換えます。

##########	The Amazon S3 bucket that temporarily stores your data while running a provider service-based workflow.

ID マッピングワークフローの作成 (ルールベース)

このトピックでは、一致するルールを使用してファーストパーティデータをソースからターゲットに変換 AWS アカウント する ID マッピングワークフローを作成するプロセスについて説明します。

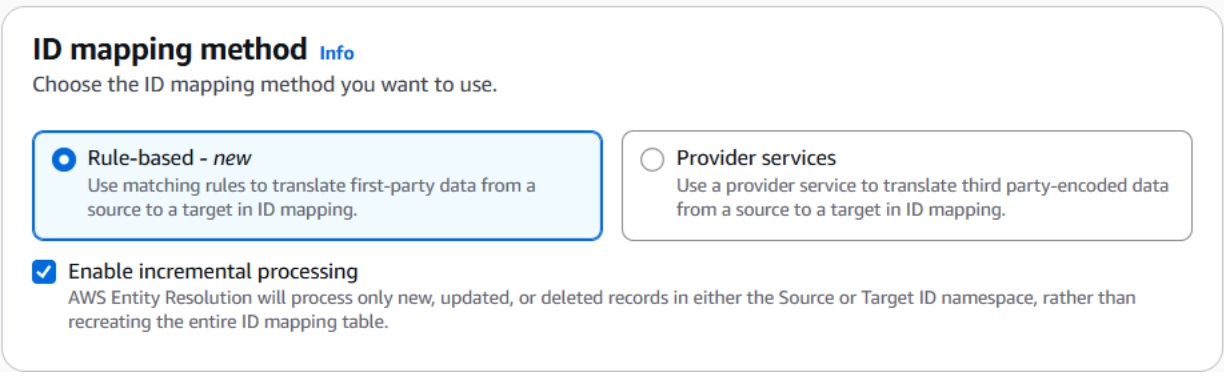
ルールベースの ID マッピングワークフローを作成するには AWS アカウント

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、ID マッピングを選択します。
3. ID マッピングワークフローページの右上隅で、ID マッピングワークフローの作成を選択します。
4. ステップ 1: ID マッピングワークフローの詳細を指定するには、次の手順を実行します。

    a. ID マッピングワークフロー名とオプションの Description を入力します。

b. ID マッピングメソッドで、ルールベースを選択します。

c. (オプション) ワークフローで新規、更新、または削除されたレコードのみを処理するには、増分処理を有効にするを選択します。



AWS Entity Resolution は、ID マッピングテーブル全体を再作成するのではなく、ソース ID またはターゲット ID 名前空間の新規、更新、または削除されたレコードのみを処理します。

増分処理を選択し、データテーブルに DELETE 列がある場合、 は DELETE 列の値に基づいてレコードを異なる方法で AWS Entity Resolution 処理します。

- DELETE 列`true`で とマークされたレコードは、ID マッピングテーブルから削除されます。
- DELETE 列`false`で とマークされたレコードは Amazon S3 に取り込まれます。

このオプションを選択したままにすると、 は ID マッピングテーブルでデフォルトのバッチ処理 ID マッピングワークフロー AWS Entity Resolution を実行します。

d. (オプション) リソースのタグを有効にするには、新しいタグを追加を選択し、キーと値のペアを入力します。

e. [次へ] を選択します。

5. ステップ 2: ソースとターゲットを指定するには、次の手順を実行します。

a. ソース で、該当するシナリオを選択し、推奨されるアクションを実行します。

シナリオ	推奨されるアクション
ID マッピングワークフローで独自の AWS Glue データベース、 AWS Glue テーブル、スキーママッピングを使用します。	1. スキーママッピングを選択します。 2. AWS リージョン、AWS Glue データベース、AWS Glue テーブルを選択し、対応するスキーママッピングを選択します。 最大 19 個のデータ入力を追加できます。
ID マッピングワークフローで使用するレコードデータを指す既存の一致ワークフローを使用します。	1. 一致ワークフローを選択します。 2. ドロップダウンリストから既存の一致ワークフローを選択します。

b. ターゲット で、ドロップダウンリストから既存の一致ワークフローを選択します。

c. ルールパラメータについては、以下を実行します。

i. ソースタイプに基づいて次のいずれかのオプションを選択して、ルールコントロールを指定します。

ソースタイプ	推奨されるアクション
マッチングワークフロー	ソース、ターゲット、またはその両方が ID マッピングワークフローでルールを提供できるかどうかを選択して、ルールコントロールを指定します。 ID マッピングワークフローで使用するルールコントロールは、ソースとターゲットの間で互換性がある必要があります。 例えば、ソース ID 名前空間がルールをターゲットに制限するが、ターゲット ID 名前空間がルールをソースに制限する場合、エラーが発生します。

ソースタイプ	推奨されるアクション
スキーママッピング	この手順をスキップしてください。

ii. 比較パラメータとマッチングパラメータの場合、比較タイプは自動的に複数の入力フィールドに設定されます。

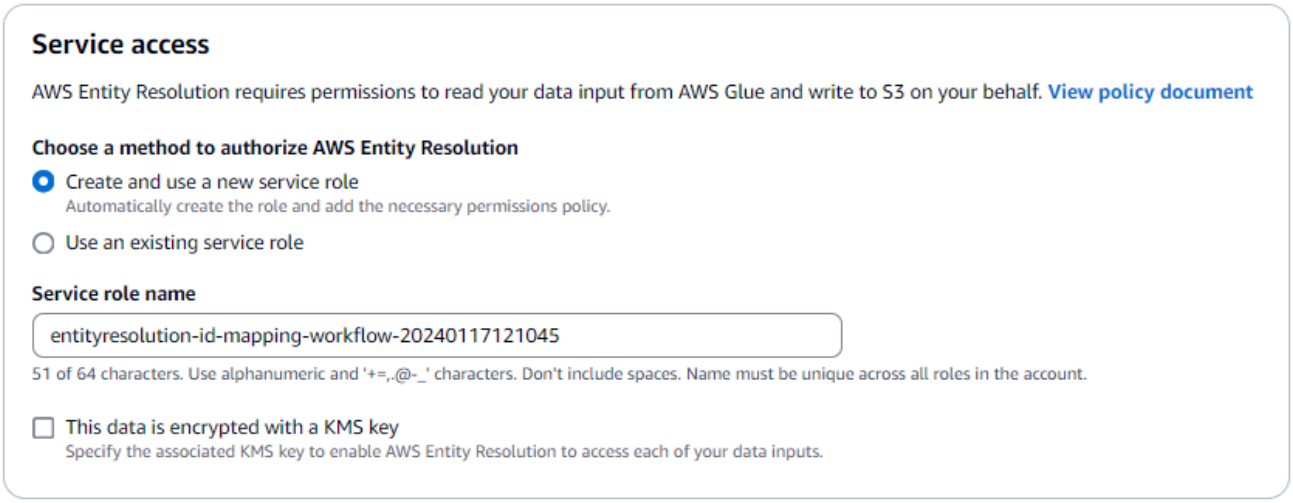
これは、両方の参加者が以前にこのオプションを選択したためです。

d. 目標に基づいて次のいずれかのオプションを選択して、レコードマッチングタイプを指定します。

目標	推奨されるオプション
ID マッピングワークフローを作成するときに、レコード一致タイプを制限して、ターゲット内の一致するレコードごとに、一致するレコードを 1 つだけソースに保存します。	1 つのソースから 1 つのターゲットへ
ID マッピングワークフローを作成するときに、レコード一致タイプを制限して、ターゲット内の一致するレコードごとに、一致するすべてのレコードをソースに保存します。	1 つのターゲットへの多くのソース

> **Note**
>
> ソース ID 名前空間とターゲット ID 名前空間に互換性のある制限を指定する必要があります。

e. サービスアクセス許可を指定するには、 オプションを選択し、推奨アクションを実行します。



オプション	推奨されるアクション
新しいサービスロールを作成して使用	• AWS Entity Resolution は、このテーブルに必要なポリシーを持つサービスロールを作成します。 • デフォルトの [サービスロール名] は `entityresolution-id-mapping-workflow-<timestamp>` です。 • ロールを作成してポリシーをアタッチするアクセス許可が必要です。 • 入力データが暗号化されている場合は、このデータは KMS キーオプションで暗号化されます。次に、データ入力の復号に使用される AWS KMS キーを入力します。

オプション	推奨されるアクション
既存のサービスロールを使用	1. ドロップダウンリストから [既存のサービスロール名] を選択します。 ロールを一覧表示するアクセス許可がある場合は、ロールのリストが表示されます。 ロールを一覧表示するアクセス許可がない場合は、使用するロールの Amazon リソースネーム (ARN) を入力できます。 既存のサービスロールがない場合、[既存のサービスロールを使用] オプションは使用できません。 2. [IAM で表示] 外部リンクを選択してサービスロールを表示します。 デフォルトでは、 AWS Entity Resolution は既存のロールポリシーを更新して必要なアクセス許可を追加しようとしません。

6. [次へ] を選択します。
7. ステップ 3: データ出力の場所を指定する – オプションで、次の手順を実行します。
   a. データ出力先については、次の操作を行います。
      i. データ出力の Amazon S3 の場所を選択します。
      ii. 暗号化で、暗号化設定をカスタマイズする場合は、AWS KMS キー ARN を入力するか、 AWS KMS キーの作成を選択します。
   b. [次へ] を選択します。
8. ステップ 4: 確認して作成するには、次の手順を実行します。
   a. 前のステップで選択した内容を確認し、必要に応じて編集します。
   b. [作成] を選択します。

ID マッピングワークフローが作成されたことを示すメッセージが表示されます。

ID マッピングワークフローを作成したら、ID マッピングワークフローを実行する準備が整います。

ID マッピングワークフローの作成 (プロバイダーサービス)

このトピックでは、LiveRamp というプロバイダーサービス AWS アカウント を使用して ID マッピングワークフローを作成するプロセスについて説明します。LiveRamp は、維持された RampIDsまたは派生した RampID を使用して、ソース RampIDs。

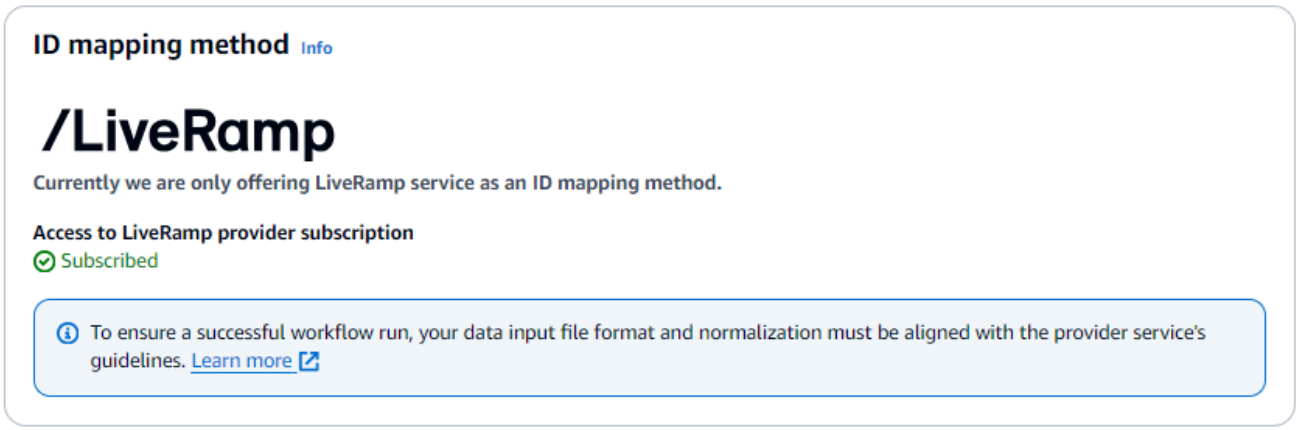
プロバイダーのサービスベースの ID マッピングワークフローを作成するには AWS アカウント

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、ID マッピングを選択します。
3. ID マッピングワークフローページの右上隅で、ID マッピングワークフローの作成を選択します。
4. ステップ 1: ID マッピングワークフローの詳細を指定するには、次の手順を実行します。

    a. ID マッピングワークフロー名とオプションの Description を入力します。

    b. ID マッピングメソッドで、プロバイダーサービスを選択します。

    AWS Entity Resolution は現在、ID マッピング方法として LiveRamp プロバイダーサービスを提供しています。LiveRamp のサブスクリプションをお持ちの場合、ステータスは Subscribed と表示されます。LiveRamp をサブスクライブする方法の詳細については、「」を参照してくださいステップ 1: でプロバイダーサービスをサブスクライブする AWS Data Exchange。



> **Note**
>
> データ入力ファイル形式がプロバイダーサービスのガイドラインと一致していることを確認します。LiveRamp の入力ファイルフォーマットガイドラインの詳細については、LiveRamp ドキュメントウェブサイトの「ADX による翻訳の実行」を参照してください。

c. LiveRamp 設定には、LiveRamp が提供する次の値を入力します。

- クライアント ID マネージャー ARN
- クライアントシークレットマネージャー ARN

d. (オプション) リソースのタグを有効にするには、新しいタグを追加を選択し、キーと値のペアを入力します。

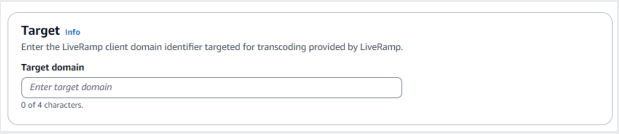
e. [次へ] を選択します。

5. ステップ 2: ソースとターゲットを指定するには、次の手順を実行します。

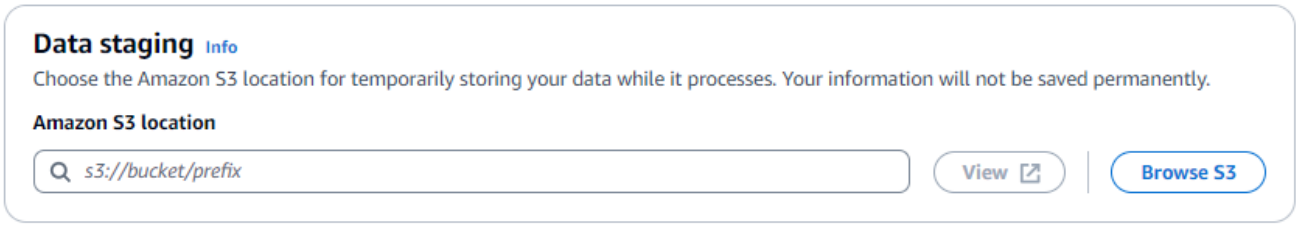
a. Source で、該当するシナリオを選択し、推奨されるアクションを実行します。

シナリオ	推奨されるアクション
ID マッピングワークフローで独自の AWS Glue データベース、 AWS Glue テーブル、スキーママッピングを使用します。	1. スキーママッピングを選択します。 2. AWS リージョン、AWS Glue データベース、AWS Glue テーブルを選択し、対応するスキーママッピングを選択します。 最大 19 個のデータ入力を追加できます。
ID マッピングワークフローで使用するレコードデータを指す既存の一致ワークフローを使用します。	1. 一致ワークフローを選択します。 2. ドロップダウンリストから既存の一致ワークフローを選択します。

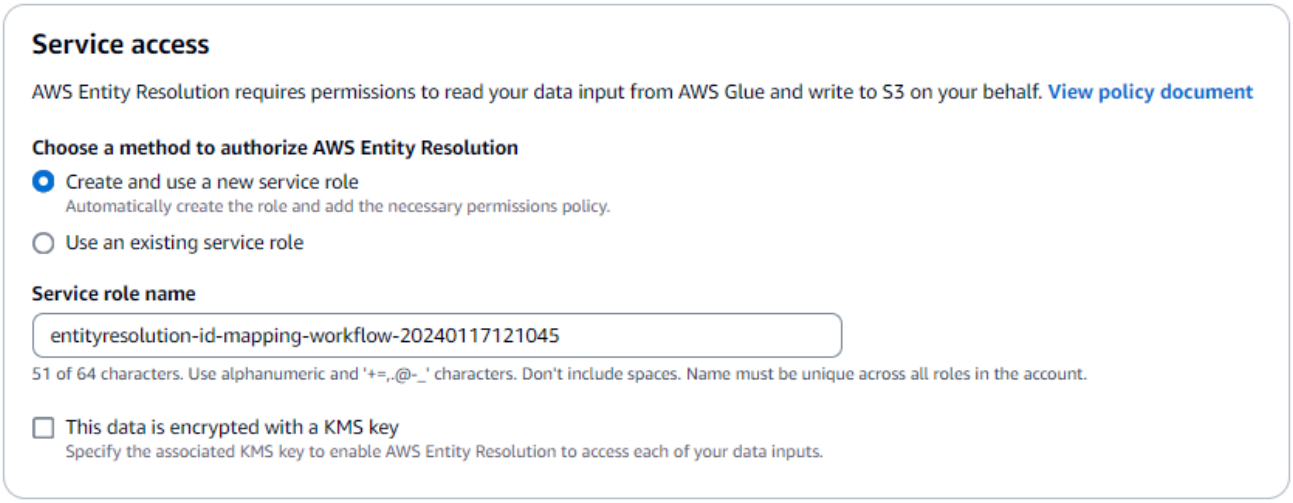
b. ターゲット では、選択した ID マッピング方法に基づいて、次のいずれかのアクションを実行します。

ID マッピング方法	推奨されるアクション
ルールベース	ドロップダウンリストから既存の一致ワークフローを選択します。
プロバイダーサービス	LiveRamp がターゲットドメインで提供するトランスコードの対象となる LiveRamp クライアントドメイン識別子を入力します。 

c. データステージングでは、ID マッピングワークフロー出力を一時的に書き込む Amazon S3 の場所を選択します。



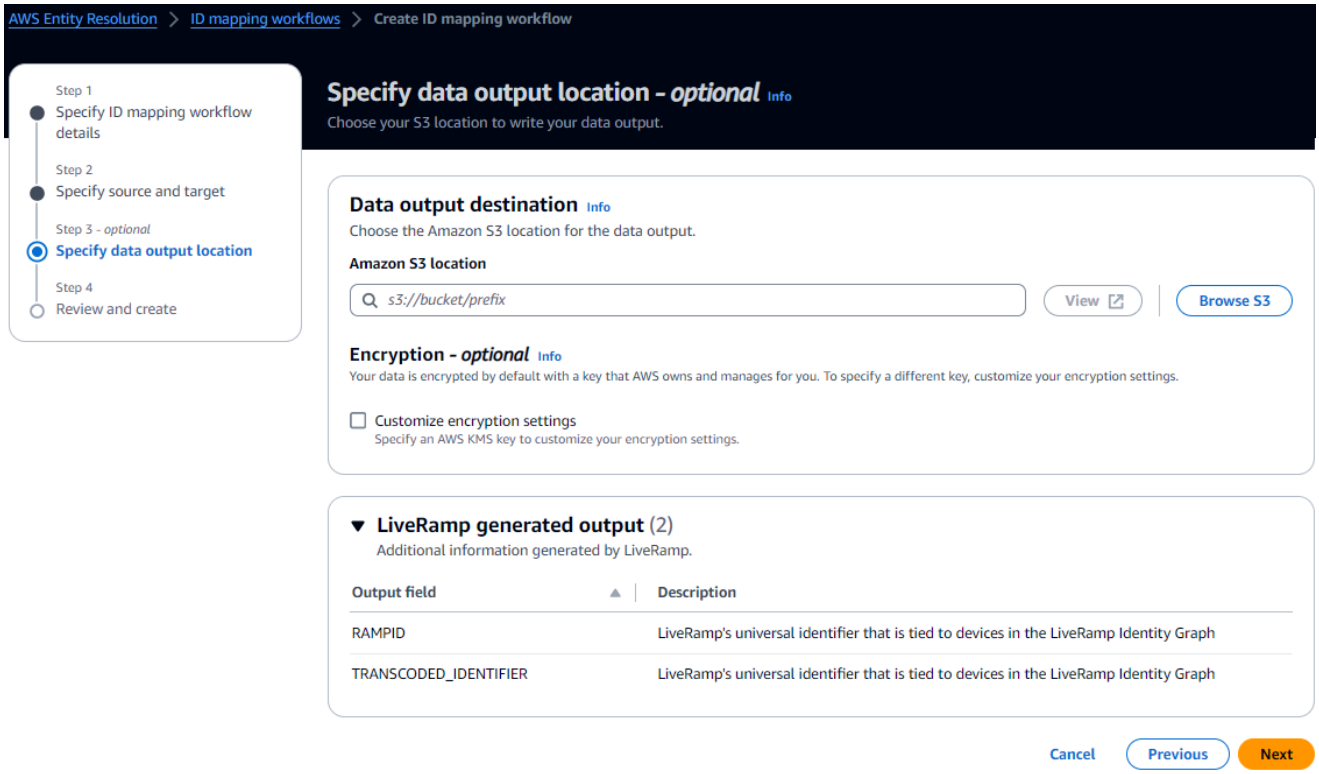
d. サービスアクセス許可を指定するには、 オプションを選択し、推奨アクションを実行します。



オプション	推奨されるアクション
新しいサービスロールを作成して使用	• AWS Entity Resolution は、このテーブルに必要なポリシーを持つサービスロールを作成します。 • デフォルトの [サービスロール名] は `entityresolution-id-mapping-workflow-<timestamp>` です。 • ロールを作成してポリシーをアタッチするアクセス許可が必要です。 • 入力データが暗号化されている場合は、このデータは KMS キーオプションで暗号化されます。次に、データ入力の復号に使用される AWS KMS キーを入力します。

オプション	推奨されるアクション
既存のサービスロールを使用	1. ドロップダウンリストから [既存のサービスロール名] を選択します。 ロールを一覧表示するアクセス許可がある場合は、ロールのリストが表示されます。 ロールを一覧表示するアクセス許可がない場合は、使用するロールの Amazon リソースネーム (ARN) を入力できます。 既存のサービスロールがない場合、[既存のサービスロールを使用] オプションは使用できません。 2. [IAM で表示] 外部リンクを選択してサービスロールを表示します。 デフォルトでは、 AWS Entity Resolution は既存のロールポリシーを更新して必要なアクセス許可を追加しようとしません。

6. [次へ] を選択します。
7. ステップ 3: データ出力場所を指定する – オプションで、次の手順を実行します。
   a. データ出力先については、次の操作を行います。
      i. データ出力の Amazon S3 の場所を選択します。
      ii. 暗号化で、暗号化設定をカスタマイズする場合は、AWS KMS キー ARN を入力するか、 AWS KMS キーの作成を選択します。
   b. LiveRamp が生成した出力を表示します。
   c. [次へ] を選択します。



8. ステップ 4: 確認して作成するには、次の手順を実行します。

   a. 前のステップで選択した内容を確認し、必要に応じて編集します。

   b. [作成] を選択します。

      ID マッピングワークフローが作成されたことを示すメッセージが表示されます。

9. ID マッピングワークフローを作成したら、ID マッピングワークフローを実行する準備が整います。

2 つの にわたる ID マッピングワークフロー AWS アカウント

2 つの にわたる ID マッピングワークフロー AWS アカウントを使用すると、2 つのデータセット間で ID マッピングを実行できます AWS アカウント。これは通常、独自の AWS アカウント と別の の間で行われます AWS アカウント。

たとえば、パブリッシャーは、独自のターゲット ID 名前空間 (独自の AWS アカウント) とアドバタイザーのソース ID 名前空間 (別の) を使用して ID マッピングワークフローを作成できます AWS アカウント。

2 つの ID マッピングワークフローを作成する前に AWS アカウント、まず前提条件を完了する必要があります。

ID マッピングワークフローを作成したら、出力 (ID マッピングテーブル) を表示し、分析に使用できます。

以下のトピックでは、2 つの ID マッピングワークフローを作成する一連のステップについて説明します AWS アカウント。

トピック

前提条件

2 つの ID マッピングワークフローを作成する前に AWS アカウント、まず以下を実行する必要があります。

- セットアップ AWS Entity Resolution の各タスクを完了する。
- ID 名前空間ソースを作成します。
- ID 名前空間ターゲットを作成します。
- 別の から ID 名前空間ソースを使用している場合は、ID 名前空間 ARN を取得します AWS アカウント。
- (プロバイダーサービスのみ) 2 つの間で ID マッピングワークフローを作成するには、LiveRamp が S3 バケットと AWS Key Management Service (AWS KMS) カスタマーマネージドキーにアクセスするためのアクセス許可 AWS アカウント が必要です。

  LiveRamp AWS アカウント を使用して 2 つの にまたがる ID マッピングワークフローを作成する前に、次のアクセス許可ポリシーを追加します。これにより、LiveRamp は S3 バケットとカスタマーマネージドキーにアクセスできます。

  JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
        "Effect": "Allow",
```

```
        "Principal": {
            "AWS": "arn:aws:iam::715724997226:root"
        },
        "Action": [
            "kms:Decrypt"
        ],
        "Resource": "arn:aws:kms:us-east-1:111122223333:key/key-id",
        "Condition": {
            "StringEquals": {
            "kms:ViaService": "s3.us-east-1.amazonaws.com"
            }
        }
    }]
}
```

前述のアクセス許可ポリシーで、各 *<###############>* を独自の情報に置き換えます。

<KMSKeyARN>	The ARN of an AWS KMS customer managed key.

ID マッピングワークフローの作成 (ルールベース)

前提条件を完了したら、1 つ以上の ID マッピングワークフローを作成して、一致するルールを使用してファーストパーティーデータをソースからターゲットに変換できます。

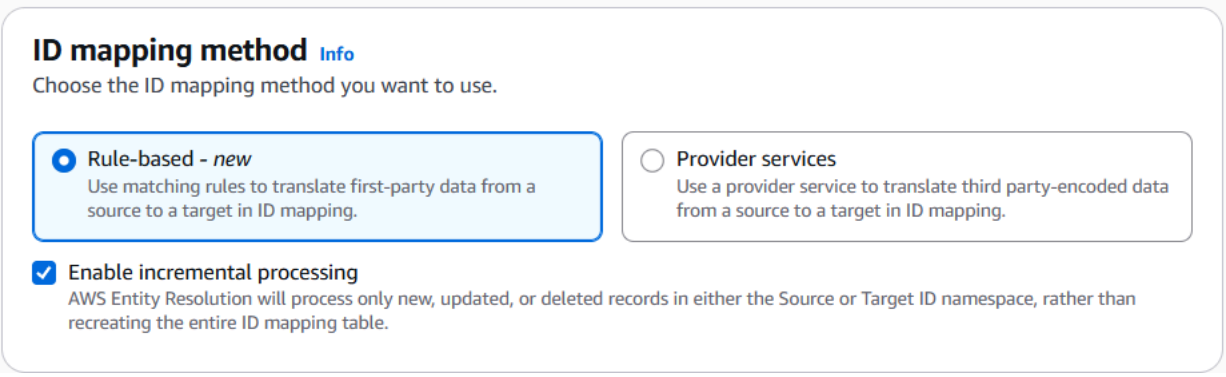
2 つの 間でルールベースの ID マッピングワークフローを作成するには AWS アカウント

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、ID マッピングを選択します。
3. ID マッピングワークフローページの右上隅で、ID マッピングワークフローの作成を選択します。
4. ステップ 1: ID マッピングワークフローの詳細を指定するには、次の手順を実行します。

    a. ID マッピングワークフロー名とオプションの Description を入力します。

    b. ID マッピングメソッドで、ルールベースを選択します。

c. (オプション) ワークフローで新規、更新、または削除されたレコードのみを処理するには、増分処理を有効にするを選択します。



AWS Entity Resolution は、ID マッピングテーブル全体を再作成するのではなく、ソース ID またはターゲット ID 名前空間の新規、更新、または削除されたレコードのみを処理します。

増分処理を選択し、データテーブルに DELETE 列がある場合、 は DELETE 列の値に基づいてレコードを異なる方法で AWS Entity Resolution 処理します。

- DELETE 列`true`で とマークされたレコードは、ID マッピングテーブルから削除されます。
- DELETE 列`false`で とマークされたレコードは Amazon S3 に取り込まれます。

このオプションを選択したままにすると、 は ID マッピングテーブルでデフォルトのバッチ処理 ID マッピングワークフロー AWS Entity Resolution を実行します。

d. (オプション) リソースのタグを有効にするには、新しいタグを追加を選択し、キーと値のペアを入力します。

e. [次へ] を選択します。

5. ステップ 2: ソースとターゲットを指定するには、次の手順を実行します。

a. 詳細オプションをオンにします。

b. ソース で、一致ワークフローを選択し、ドロップダウンリストから既存の一致ワークフローを選択します。

c. ターゲット で、一致ワークフローを選択し、ドロップダウンリストから既存の一致ワークフローを選択します。

d. ルールパラメータで、ソースまたはターゲットが ID マッピングワークフローでルールを提供できるかどうかを選択して、ルールコントロールを指定します。

ID マッピングワークフローで使用するルールコントロールは、ソースとターゲットの間で互換性がある必要があります。例えば、ソース ID 名前空間がルールをターゲットに制限するが、ターゲット ID 名前空間がルールをソースに制限する場合、エラーが発生します。

e. 比較パラメータとマッチングパラメータについては、以下を実行します。

i. 目標に基づいてオプションを選択して、比較タイプを指定します。

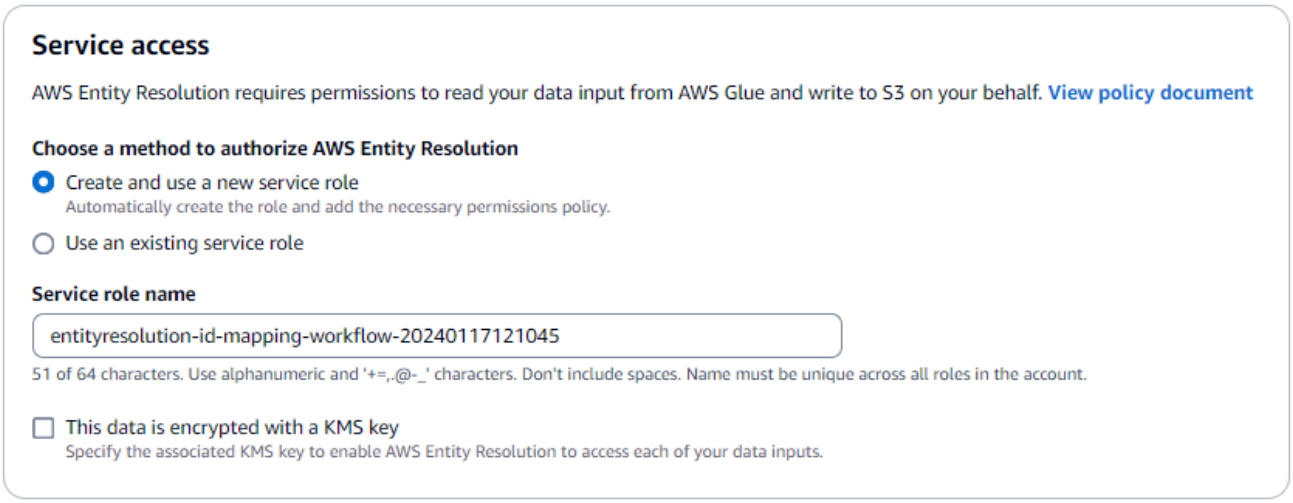
目標	推奨されるオプション
データが同じ入力フィールドにあるか異なる入力フィールドにあるかに関係なく、複数の入力フィールドに保存されているデータ間で一致の任意の組み合わせを見つけます。	複数の入力フィールド
複数の入力フィールドに保存されている類似データが一致しない場合、1 つの入力フィールド内で比較を制限します。	単一の入力フィールド

ii. 目標に基づいてオプションを選択して、レコードマッチングタイプを指定します。

目標	推奨されるオプション
ID マッピングワークフローを作成するときに、レコード一致タイプを制限して、ターゲット内の一致するレコードごとに、一致するレコードを 1 つだけソースに保存します。	1 つのソースから 1 つのターゲットへ
ID マッピングワークフローを作成するときに、レコード一致タイプを制限して、ターゲット内の一致するレコードごとに、一致するすべてのレコードをソースに保存します。	1 つのターゲットへの多くのソース

> **Note**
>
> ソース ID 名前空間とターゲット ID 名前空間に互換性のある制限を指定する必要があります。

f. サービスアクセス許可を指定するには、 オプションを選択し、推奨アクションを実行します。



オプション	推奨されるアクション
新しいサービスロールを作成して使用	• AWS Entity Resolution は、このテーブルに必要なポリシーを持つサービスロールを作成します。 • デフォルトの [サービスロール名] は `entityresolution-id-mapping-workflow-<timestamp>` です。 • ロールを作成してポリシーをアタッチするアクセス許可が必要です。 • 入力データが暗号化されている場合は、このデータは KMS キーオプションで暗号化されます。次に、データ入力の復号に使用される AWS KMS キーを入力します。

オプション	推奨されるアクション
既存のサービスロールを使用	1. ドロップダウンリストから [既存のサービスロール名] を選択します。 ロールを一覧表示するアクセス許可がある場合は、ロールのリストが表示されます。 ロールを一覧表示するアクセス許可がない場合は、使用するロールの Amazon リソースネーム (ARN) を入力できます。 既存のサービスロールがない場合、[既存のサービスロールを使用] オプションは使用できません。 2. [IAM で表示] 外部リンクを選択してサービスロールを表示します。 デフォルトでは、 AWS Entity Resolution は既存のロールポリシーを更新して必要なアクセス許可を追加しようとしません。

6. [次へ] を選択します。
7. ステップ 3: データ出力場所を指定する – オプションで、次の手順を実行します。
   a. データ出力先については、次の操作を行います。
      i. データ出力の Amazon S3 の場所を選択します。
      ii. 暗号化では、暗号化設定をカスタマイズする場合は、AWS KMS キー ARN を入力するか、 AWS KMS キーの作成を選択します。
   b. LiveRamp が生成した出力を表示します。
   c. [次へ] を選択します。
8. ステップ 4: 確認して作成するには、次の手順を実行します。
   a. 前のステップで選択した内容を確認し、必要に応じて編集します。

b. [作成] を選択します。

ID マッピングワークフローが作成されたことを示すメッセージが表示されます。

ID マッピングワークフローを作成したら、ID マッピングワークフローを実行する準備が整います。

ID マッピングワークフローの作成 (プロバイダーサービス)

前提条件を完了したら、LiveRamp プロバイダーサービスを使用して 1 つ以上の ID マッピングワークフローを作成できます。LiveRamp は、維持された RampIDsまたは派生した RampID を使用して、ソース RampIDs。

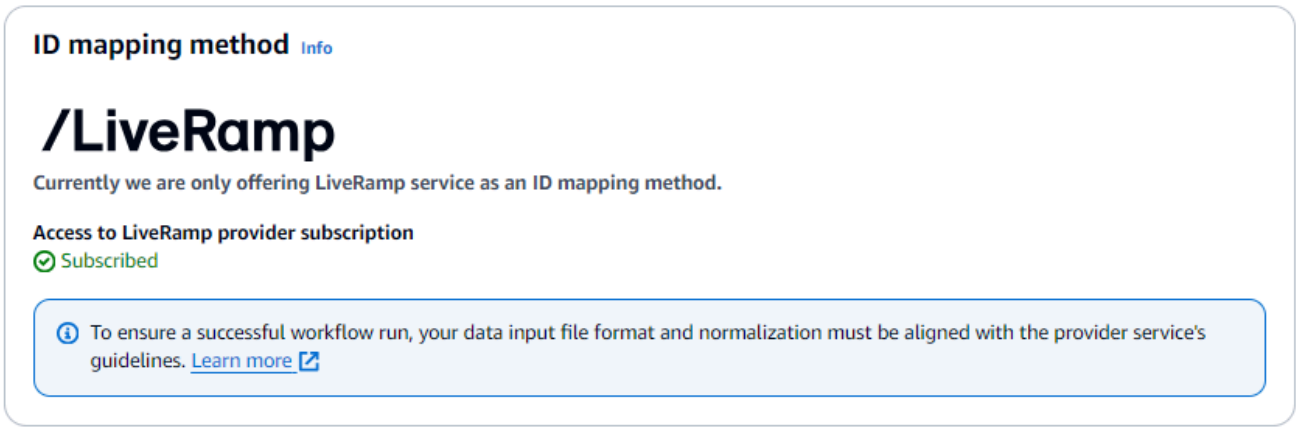
プロバイダーサービスを使用して ID マッピングワークフローを作成するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、ID マッピングを選択します。
3. ID マッピングワークフローページの右上隅で、ID マッピングワークフローの作成を選択します。
4. ステップ 1: ID マッピングワークフローの詳細を指定するには、次の手順を実行します。

   a. ID マッピングワークフロー名とオプションの Description を入力します。

   b. ID マッピングメソッドで、プロバイダーサービスを選択します。

      AWS Entity Resolution は現在、ID マッピング方法として LiveRamp プロバイダーサービスを提供しています。LiveRamp にサブスクリプションがある場合、ステータスは Subscribed と表示されます。LiveRamp をサブスクライブする方法の詳細については、「」を参照してくださいステップ 1: でプロバイダーサービスをサブスクライブする AWS Data Exchange。



> **Note**
>
> データ入力ファイル形式がプロバイダーサービスのガイドラインと一致していることを確認します。LiveRamp の入力ファイルフォーマットガイドラインの詳細については、LiveRamp ドキュメントウェブサイトの「ADX による翻訳の実行」を参照してください。

c. LiveRamp 設定には、LiveRamp が提供する次の値を入力します。

- クライアント ID マネージャー ARN
- クライアントシークレットマネージャー ARN

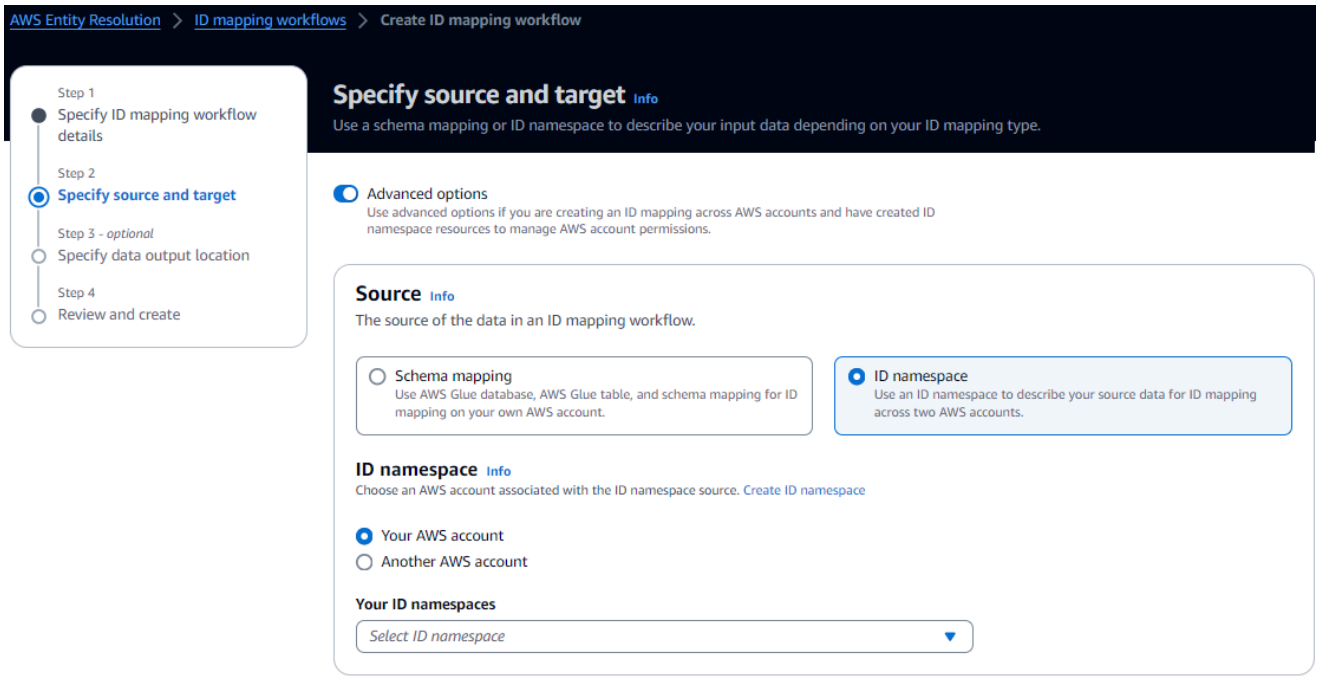
d. (オプション) リソースのタグを有効にするには、新しいタグを追加を選択し、キーと値のペアを入力します。

e. [次へ] を選択します。

5. ステップ 2: ソースとターゲットを指定するには、次の手順を実行します。

a. 詳細オプションをオンにします。

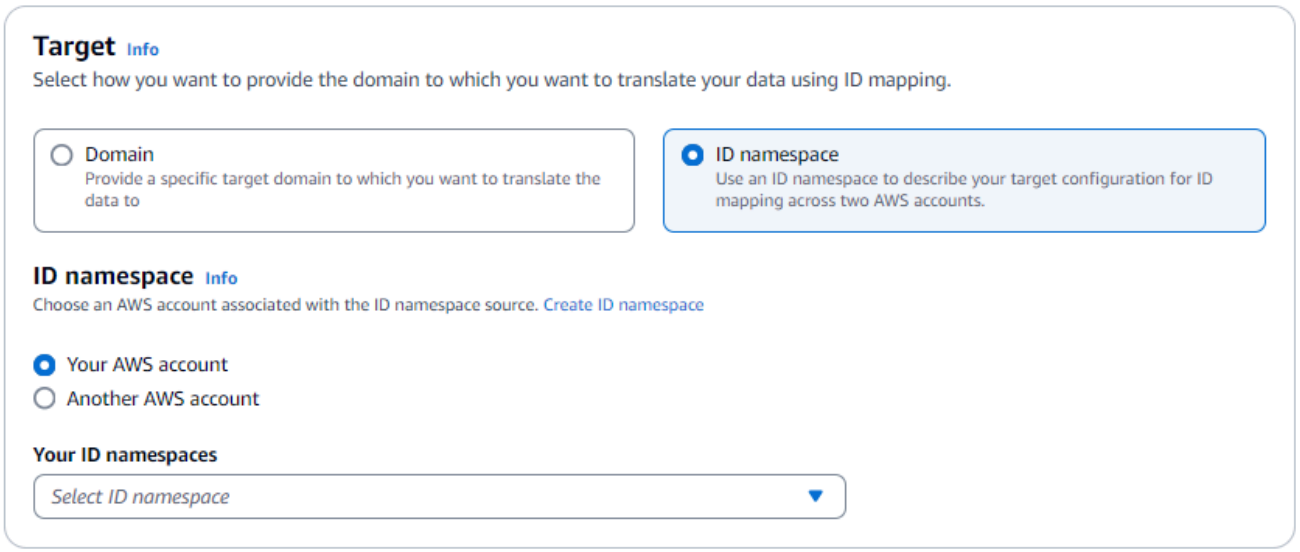
b. Source で、ID 名前空間を選択します。



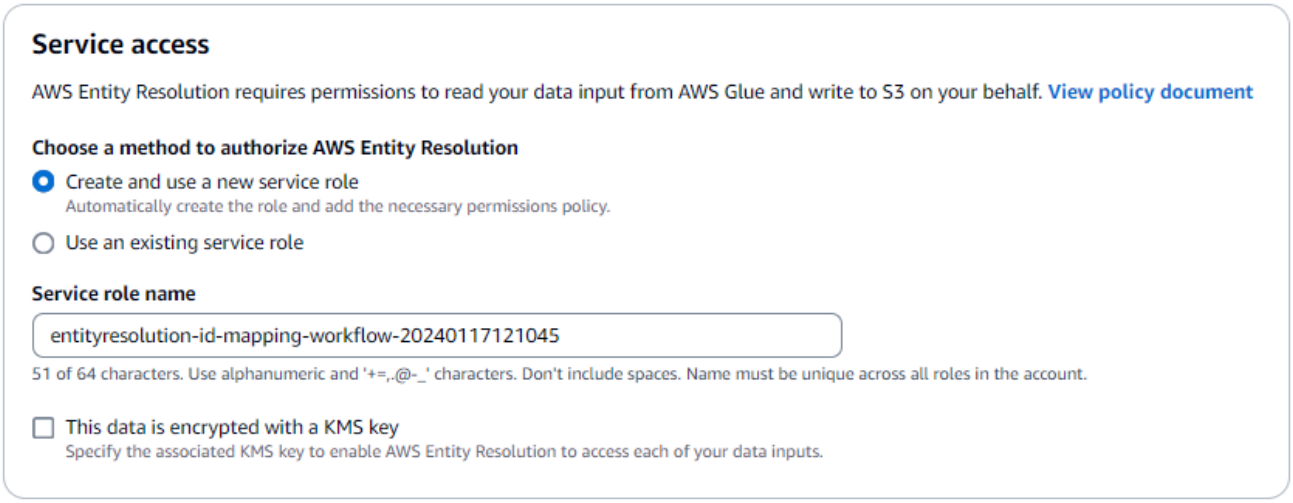
c. ID 名前空間の場合は、ID 名前空間の場所を特定し、推奨されるアクションを実行します。

ID 名前空間の場所	推奨されるアクション
独自の AWS アカウント	1. 自分の AWS アカウント を選択します。 2. ID 名前空間ドロップダウンリストから ID 名前空間を選択します。
他のユーザーの AWS アカウント	1. 別の AWS アカウント を選択します。 2. ID 名前空間 ARN を入力します。

d. Target で、ID 名前空間を選択します。



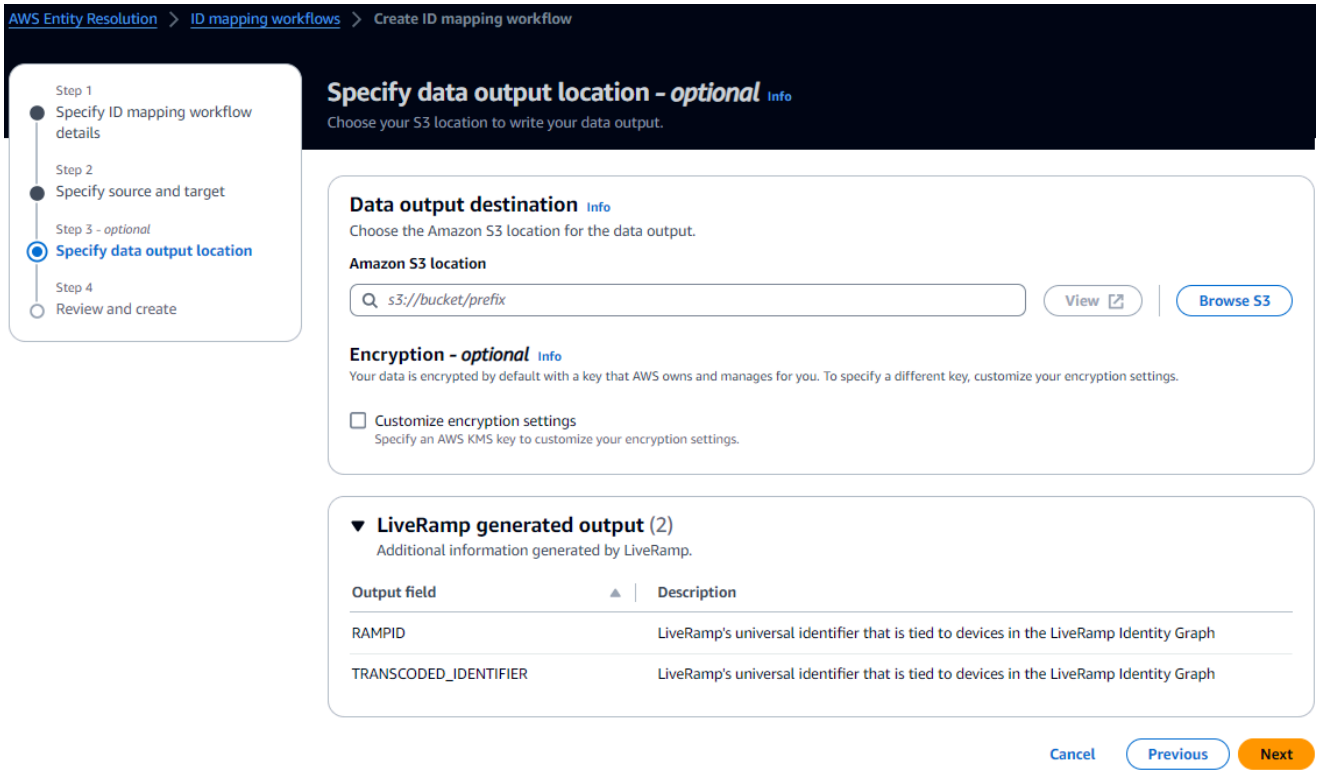
e. サービスアクセス許可を指定するには、 オプションを選択し、推奨アクションを実行します。



オプション	推奨されるアクション
新しいサービスロールを作成して使用	• AWS Entity Resolution は、このテーブルに必要なポリシーを持つサービスロールを作成します。 • デフォルトの [サービスロール名] は `entityresolution-id-mapping-workflow-<timestamp>` です。 • ロールを作成してポリシーをアタッチするアクセス許可が必要です。 • 入力データが暗号化されている場合は、このデータは KMS キーオプションで暗号化されます。次に、データ入力の復号に使用される AWS KMS キーを入力します。

オプション	推奨されるアクション
既存のサービスロールを使用	1. ドロップダウンリストから [既存のサービスロール名] を選択します。 ロールを一覧表示するアクセス許可がある場合は、ロールのリストが表示されます。 ロールを一覧表示するアクセス許可がない場合は、使用するロールの Amazon リソースネーム (ARN) を入力できます。 既存のサービスロールがない場合、[既存のサービスロールを使用] オプションは使用できません。 2. [IAM で表示] 外部リンクを選択してサービスロールを表示します。 デフォルトでは、 AWS Entity Resolution は既存のロールポリシーを更新して必要なアクセス許可を追加しようとしません。

6. [次へ] を選択します。
7. ステップ 3: データ出力場所を指定する – オプションで、次の手順を実行します。
   a. データ出力先については、次の操作を行います。
      i. データ出力の Amazon S3 の場所を選択します。
      ii. 暗号化では、暗号化設定をカスタマイズする場合は、AWS KMS キー ARN を入力するか、 AWS KMS キーの作成を選択します。
   b. LiveRamp が生成した出力を表示します。
   c. [次へ] を選択します。



8. ステップ 4: 確認して作成するには、次の手順を実行します。

   a. 前のステップで選択した内容を確認し、必要に応じて編集します。

   b. [作成] を選択します。

      ID マッピングワークフローが作成されたことを示すメッセージが表示されます。

ID マッピングワークフローを作成したら、ID マッピングワークフローを実行する準備が整います。

ID マッピングワークフローの実行

1 つの ID マッピングワークフローを作成する AWS アカウントか、2 つの ID マッピングワークフローを作成 AWS アカウントしたら、ID マッピングワークフローを実行できます。ID マッピングワークフローは CSV ファイルを出力します。

ID マッピングワークフローを実行するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、ID マッピングを選択します。

3. ID マッピングワークフローを選択します。
4. ID マッピングワークフローの詳細ページの右上隅にある「実行」を選択します。
5. 一致するワークフローの詳細ページのメトリクスタブで、「最後のジョブメトリクス」で以下を表示します。

   - ジョブ ID
   - 一致するワークフロージョブのステータス: Queued、In progress、Completed、Failed
   - 実行タイプ
   - ワークフロージョブの開始時刻
   - ワークフロージョブの完了時刻
   - ワークフロージョブの期間
   - 出力先
   - AWS KMS key
   - サービスロール
   - 入力レコードの数
   - 一意のレコードの数
   - ロードされた新しい一意のレコードの数
   - マッピングされたレコードの数
   - 削除されたマッピングされたレコードの数
   - マッピングされた新しいレコードの数
   - マッピングされたソースレコードの数
   - マッピングされた新しいソースレコードの数
   - 削除されたマッピングされたソースレコードの数
   - マッピングされたターゲットレコードの数
   - マッピングされた新しいターゲットレコードの数
   - 削除されたマッピングされたターゲットレコードの数
   - 処理された削除レコードの数
   - 処理されたレコードの数
   - 処理されていないレコードの数

ジョブ履歴では、以前に実行した ID マッピングワークフロージョブのジョブメトリクスを表示することもできます。

6. ID マッピングワークフロージョブが完了したら (ステータスが完了)、データ出力を選択し、Amazon S3 の場所を選択して結果を表示します。

   CSV ファイルを取得したら、 `RAMPID`と を結合できます`TRANSCODED_ID`。

カスタム ID マッピングワークフローの実行

> **Note**
>
> この手順は、1 つのワークフロー、 AWS アカウントまたは増分処理を有効にした 2 つのワークフロー AWS アカウントで使用できます。

ID マッピングワークフローを実行するときは、出力データに元の設定とは異なる Amazon S3 の場所を指定できます。また、バッチ (すべてのデータを処理する) 、増分 (新規または変更されたデータのみを処理する) 、削除のみ (削除リクエストのみを処理する) の 3 つの実行タイプのいずれかを選択して、データを処理する方法を選択することもできます。

新しい出力先で ID マッピングワークフローを実行するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、ID マッピングを選択します。
3. 実行する ID マッピングワークフローを選択します。
4. ID マッピングワークフローの詳細ページで、ワークフローの実行を選択し、新しい出力先で実行を選択します。
5. データ出力先として、以下を設定します。

   a. 実行タイプで、次のいずれかのオプションを選択します。

      - バッチ – ID マッピングテーブル全体を処理します。

        初期設定、定期的なフル更新、またはソース ID 名前空間とターゲット ID 名前空間の両方で大幅な変更が発生した場合に推奨されます。

- 増分 – ソース ID またはターゲット ID 名前空間のいずれかで、新規、更新、または削除されたレコードのみを処理します。

  頻繁な更新、日次実行、またはリアルタイムのデータ同期に推奨されます。

- 削除のみ – ターゲット ID 名前空間から削除されたレコードのみを処理します。

  削除をすばやく同期するために推奨されます。

b. データ出力の Amazon S3 の場所を選択します。

c. 暗号化の場合は、次のいずれかを実行します。

- デフォルトの暗号化設定を保持する
- 暗号化設定をカスタマイズを選択し、AWS KMS キー ARN を入力するか、 AWS KMS キーの作成を選択します。

6. サービスアクセス許可を指定するには、 オプションを選択し、推奨アクションを実行します。

オプション	推奨されるアクション
新しいサービスロールを作成して使用	• AWS Entity Resolution は、このテーブルに必要なポリシーを持つサービスロールを作成します。 • デフォルトの [サービスロール名] は `entityresolution-id-mapping-workflow-<timestamp>` です。 • ロールを作成してポリシーをアタッチするアクセス許可が必要です。 • 入力データが暗号化されている場合は、このデータは KMS キーオプションで暗号化されます。次に、データ入力の復号に使用される AWS KMS キーを入力します。
既存のサービスロールを使用	1. ドロップダウンリストから [既存のサービスロール名] を選択します。 ロールを一覧表示するアクセス許可がある場合は、ロールのリストが表示されます。

オプション	推奨されるアクション
	ロールを一覧表示するアクセス許可がない場合は、使用するロールの Amazon リソースネーム (ARN) を入力できます。 既存のサービスロールがない場合、[既存のサービスロールを使用] オプションは使用できません。 2. [IAM で表示] 外部リンクを選択してサービスロールを表示します。 デフォルトでは、 AWS Entity Resolution は既存のロールポリシーを更新して必要なアクセス許可を追加しようとしません。

7. [Run] (実行) を選択します。
8. 一致するワークフローの詳細ページのメトリクスタブで、「最後のジョブメトリクス」で以下を表示します。

   - ジョブ ID
   - ワークフロージョブの完了時刻
   - 一致するワークフロージョブのステータス: Queued、In progress、Completed、Failed
   - 処理されたレコードの数
   - 処理されていないレコードの数
   - 入力レコードの数
   - 生成された一意の一致 IDsの数。
   - マッピングされた新しいレコードの数。
   - マッピングされた新しいターゲットレコードの数。
   - マッピングされた新しいソースレコードの数。
   - 削除された新しいマッピングされたソースレコードの数。
   - 削除された新しいマッピングされたターゲットレコードの数。
   - 削除された新しいマッピングされたレコードの数。

ジョブ履歴では、以前に実行した ID マッピングワークフロージョブのジョブメトリクスを表示することもできます。

9. ID マッピングワークフロージョブが完了したら (ステータスが完了)、データ出力を選択し、Amazon S3 の場所を選択して結果を表示します。

   CSV ファイルを取得したら、 `RAMPID`と を結合できます`TRANSCODED_ID`。

ID マッピングワークフローの編集

ID マッピングワークフローを編集すると、エンティティ解決機能をup-to-date状態に保ち、時間の経過とともに変化するビジネスニーズに合わせることができます。マッピングルール、手法、パラメータを調整すると、ワークフローを最適化して、より正確で信頼性の高い ID マッチング結果を提供できます。また、新しいデータソースの追加、マッピングされる IDs のタイプの拡大、ワークフローへの追加の一致基準の組み込みを行うこともできます。ID マッピング結果の問題やエラーを特定した場合、ワークフローで編集すると、それらの問題の診断と解決に役立ちます。

ID マッピングワークフローを編集するには:

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、ID マッピングを選択します。
3. ID マッピングワークフローを選択します。
4. ID マッピングワークフローの詳細ページの右上隅で、編集を選択します。
5. ID マッピングワークフローの詳細の指定ページで、必要な変更を加え、次へを選択します。
6. データ出力を指定ページで、必要な変更を加え、次へを選択します。
7. 確認と保存ページで、必要な変更を加え、保存を選択します。

ID マッピングワークフローの削除

ID マッピングワークフローを使用しなくなった場合は、削除することでワークフロー管理を効率化できます。さらに、同様の目的を果たす冗長な ID マッピングワークフローや効率の低い ID マッピングワークフローを削除すると、プロセスを統合するのに役立ちます。

ID マッピングワークフローを削除するには:

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、ID マッピングを選択します。
3. ID マッピングワークフローを選択します。
4. ID マッピングワークフローの詳細ページの右上隅にある「削除」を選択します。
5. 削除を確定し、[削除] を選択します。

ID マッピングワークフローのリソースポリシーの追加または更新

リソースポリシーは、ID マッピングリソースの作成者が ID マッピングワークフローリソースにアクセスすることを許可します。

リソースポリシーを追加または更新するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/entityresolution/ で AWS Entity Resolution コンソールを開きます。
2. 左側のナビゲーションペインのワークフローで、ID マッピングを選択します。
3. ID マッピングワークフローを選択します。
4. ID マッピングワークフローの詳細ページで、アクセス許可タブを選択します。
5. リソースポリシーで、「編集」を選択します。
6. JSON エディタでポリシーを追加または更新します。
7. [Save changes] (変更の保存) をクリックします。

プロバイダー AWS Entity Resolution として と統合する

AWS Entity Resolution サードパーティープロバイダーの統合は、顧客が消費者のプライバシーを保護し、データ主権法への準拠を維持するのに役立ちます。LiveRamp や TransUnion などのサードパーティープロバイダーは、コンシューマー識別子を Ramp IDsや Fabrick IDs などの広告 ID に変換 IDs。これらの広告識別子は、コンシューマーデータが非AWS マネージドシステムにエクスポートされないようにするために、広告およびマーケティングツールで一般的に使用されます。このセクションでは、プロバイダーが と統合 AWS Entity Resolution して、プロバイダーのサービスベースのマッチングワークフローで使用できるように、コンシューマー識別子を広告 IDs にエンコードまたはトランスコードするためのガイダンスを提供します。

現在 と統合されているプロバイダーサービスの詳細については AWS Entity Resolution、「」を参照してくださいプロバイダーのサービスベースのマッチングワークフローの作成。

トピック

- 要件
- AWS Entity Resolution OpenAPI 仕様の使用
- プロバイダー統合のテスト

要件

プロバイダーサービスとして と統合する前に AWS Entity Resolution、次の要件を満たす必要があります。

トピック

- でプロバイダーサービスを一覧表示する AWS Data Exchange
- 属性を特定する
- AWS Entity Resolution OpenAPI 仕様をリクエストする

でプロバイダーサービスを一覧表示する AWS Data Exchange

サードパーティープロバイダーとして、AWS Data Exchange (ADX) 製品カタログに製品を一覧表示する必要があります。製品が AWS Data Exchange Product Catalog にリストされると、サブスクライバーはパブリックオファーまたはプライベートオファーを通じて製品をサブスクライブできます。

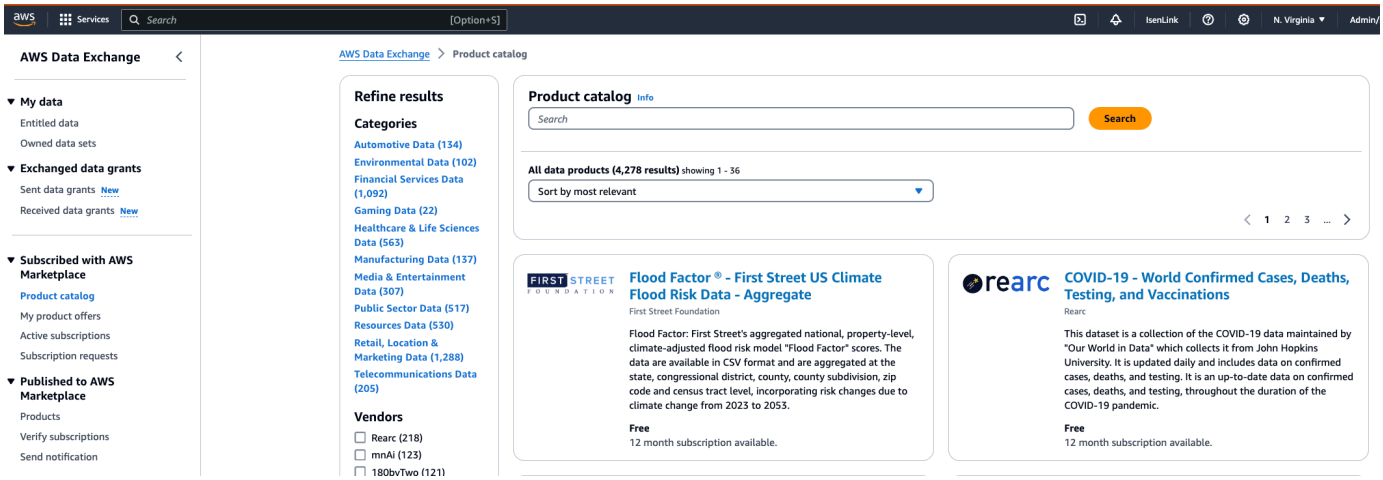
でプロバイダーサービスを一覧表示するには AWS Data Exchange

1. 新しいデータ製品プロバイダーの場合は AWS Data Exchange、AWS Data Exchange 「 ユーザーガイド」の「プロバイダーとしての開始方法」セクションのステップを完了します。
2. REST API データセットを作成し、「 AWS Data Exchange ユーザーガイドAPIs AWS Data Exchange 」の「API を含む製品の公開方法APIs を含む新しい製品を公開します。このプロセスは、 AWS Data Exchange コンソールまたは を使用して完了できます AWS Command Line Interface。

   製品の可視性パブリックを設定した場合、パブリックオファーはすべてのサブスクライバーが利用できます。

   製品の可視性プライベートを設定した場合は、ユースケースに応じて、AWS Data Exchange 「 ユーザーガイド」の「カスタムオファーの作成」セクションのステップを完了します。

   次の図は、Product Catalog で使用可能な AWS Data Exchange 製品の例を示しています。



3. 製品が AWS Data Exchange Product Catalog で利用可能になると、サブスクライバーは次の方法で製品をサブスクライブできます。

   - パブリック製品をサブスクライブします。
   - プロバイダーサービスによって発行されたプライベートオファー (カスタムオファー) を使用します。
   - Bring Your Own Subscription (BYOS) オファーを使用します。

   詳細については、「 AWS Data Exchange ユーザーガイド」のAPIs」を参照してください。

属性を特定する

入力データの属性は、ワークフローで解決されるエンティティのタイプ定義です。属性の例には、FirstName、LastName、Email、または がありますCustom String。

属性を特定するときは、要件やガイドラインに注意してください。

Example例

プロバイダー属性を識別するための検証の例を次に示します。

- FirstName または LastName 属性のいずれかは必須です。
- Email 属性が存在する場合は、ハッシュする必要があります。

プロバイダーとして、プロバイダーサービス製品の属性を特定し、これらの属性を AWS Entity Resolution <aws-entity-resolution-bd@amazon.com> のビジネス開発チームに伝達して、先に進む前に追加の検証を行う必要があります。

AWS Entity Resolution OpenAPI 仕様をリクエストする

AWS Entity Resolution には、プロバイダーとして、統合 APIs に関連する API を含むハンドシェイクとして使用できる OpenAPI 仕様があります。詳細については、「AWS Entity Resolution OpenAPI 仕様の使用」を参照してください。

OpenAPI 定義をリクエストするには、 AWS Entity Resolution ビジネス開発チーム <aws-entity-resolution-bd@amazon.com> にお問い合わせください。

AWS Entity Resolution OpenAPI 仕様の使用

OpenAPI 仕様は、関連付けられたすべてのプロトコルを定義します AWS Entity Resolution。この仕様は、統合を実装するために必要です。

OpenAPI 定義には、次の API オペレーションが含まれています。

- POST AssignIdentities
- POST CreateJob
- GET GetJob
- POST StartJob
- POST MapIdentities

- `GET Schema`

OpenAPI 仕様をリクエストするには、 AWS Entity Resolution ビジネス開発チーム <aws-entity-resolution-bd@amazon.com> にお問い合わせください。

OpenAPI 仕様は、コンシューマー識別子のバッチ処理と同期処理の両方について、2 種類の統合をサポートしています。OpenAPI 仕様を取得したら、ユースケースの処理統合のタイプを実装します。

トピック

バッチ処理の統合

バッチ処理統合は、非同期設計パターンに従います。ワークフローが開始されると AWS Data Exchange、プロバイダー統合エンドポイントを介してジョブが送信され、ワークフローはジョブのステータスを定期的にポーリングしてこのジョブの完了を待機します。このソリューションは、時間がかかり、プロバイダーのスループットが低いジョブ実行に適しています。プロバイダーは、データセットの場所を Amazon S3 リンクとして取り込みます。このリンクは、プロバイダー側で処理し、結果を所定の出力 S3 の場所に書き込むことができます。

バッチ処理統合は、3 つの API 定義を使用して有効 AWS Entity Resolution になります。 は次の順序 AWS Data Exchange で を通じて利用可能なプロバイダーエンドポイントを呼び出します。

1. `POST CreateJob`: この API オペレーションは、処理するジョブ情報をプロバイダーに送信します。これらの情報は、エンコードまたはトランスコード、S3 の場所、顧客が提供するスキーマ、必要な追加のジョブプロパティなど、ジョブのタイプに関するものです。

   この API は を返し`JobId`、ジョブのステータスは `PENDING`、`READY`、、`IN_PROGRESS`、`COMPLETE`、または のいずれかになります`FAILED`。

   エンコードのサンプルリクエスト

```
POST /jobs
{
  "actionType": "ID_ASSIGNMENT",
  "s3SourceLocation": "string",
```

```
  "s3TargetLocation": "string",
  "jobProperties": {
    "assignmentJobProperties": {
      "fieldMappings": [
        {
          "name": "string",
          "type": "NAME"
        }
      ]
    }
  },
  "customerSpecifiedJobProperties": {
    "property1": "string",
    "property2": "string"
  },
  "outputSourceConfiguration": {
    "KMSArn": "string"
  }
}
```

レスポンス例

```
{
  "jobId": "string",
  "status": "PENDING"
}
```

2. POST StartJob: この API は、プロバイダーに、JobId提供された に基づいてジョブを開始することを知らせます。これにより、プロバイダーは から CreateJobまで必要な検証を実行できますStartJob。

   この API はJobId、、ジョブStatusの 、、statusMessageおよび を返しますstatusCode。

   エンコードのサンプルリクエスト

```
POST/jobs/{jobId}
{
  "customerSpecifiedJobProperties": {
    "property1": "string",
    "property2": "string"
  }
}
```

レスポンス例

```
{
  "jobId": "string",
  "status": "PENDING",
  "statusMessage": "string",
  "statusCode": 200
}
```

3. GET GetJob: この API は、ジョブが完了した AWS Entity Resolution か、その他のステータスになったかを通知します。

   この API はJobId、、ジョブStatusの 、、statusMessageおよび を返しますstatusCode。

   エンコードのサンプルリクエスト

```
GET /jobs/{jobId}
```

   レスポンス例

```
{
  "jobId": "string",
  "status": "PENDING",
  "statusMessage": "string",
  "statusCode": 200
}
```

これらの APIs の完全な定義は、 AWS Entity Resolution OpenAPI 仕様に記載されています。

同期処理の統合

同期処理ソリューションは、スループットが高く TPS が高いリアルタイム応答時間を持つほぼリアルタイムの応答時間を持つプロバイダーに適しています。この AWS Entity Resolution ワークフローはデータセットをパーティション化し、複数の API リクエストを並行して実行します。その後、AWS Entity Resolution ワークフローは結果を目的の出力場所に書き込む処理を行います。

このプロセスは、API 定義のいずれかを使用して有効になります。 は、以下を通じて利用可能なプロバイダーエンドポイントを AWS Entity Resolution 呼び出します AWS Data Exchange。

POST AssignIdentities: この API は、source_id識別子を使用して、そのレコードrecordFieldsに関連付けられたデータをプロバイダーに送信します。

この API は を返しますassignedRecords。

エンコードのサンプルリクエスト

```
POST /assignment
{
  "sourceRecords": [
    {
      "sourceId": "string",
      "recordFields": [
        {
          "name": "string",
          "type": "NAME",
          "value": "string"
        }
      ]
    }
  ]
}
```

レスポンス例

```
{
  "assignedRecords": [
    {
      "sourceRecord": {
        "sourceId": "string",
        "recordFields": [
          {
            "name": "string",
            "type": "NAME",
            "value": "string"
          }
        ]
      },
      "identity": any
    }
  ]
}
```

これらの APIs の完全な定義は、 AWS Entity Resolution OpenAPI 仕様に記載されています。

プロバイダーが選択するアプローチに応じて、 AWS Entity Resolution はエンコードまたはトランスコードの開始に使用されるプロバイダーの設定を作成します。さらに、これらの設定は、 が提供する APIs を使用してお客様が利用できます AWS Entity Resolution。

この設定には、Amazon リソースネーム (ARN) を使用してアクセスできます。ARN AWS Data Exchange は、 のプロバイダーサービスがホストされている場所とプロバイダーサービスのタイプから取得されます。 はこの ARN を と AWS Entity Resolution 呼びますproviderServiceARN。

プロバイダー統合のテスト

はデータマッチングサービスを AWS Entity Resolution ホストしますが、プロバイダー統合はend-to-endのマッチングワークフローにとって重要なサードパーティーコンポーネントです。この統合が失敗したときに保護を追加するプロバイダーに対して AWS Entity Resolution が定義したテストがいくつかあります。このアプローチは、プロバイダーがこれらのend-to-endのテストケースに従ってサービスの状態をモニタリングする機会を提供します。

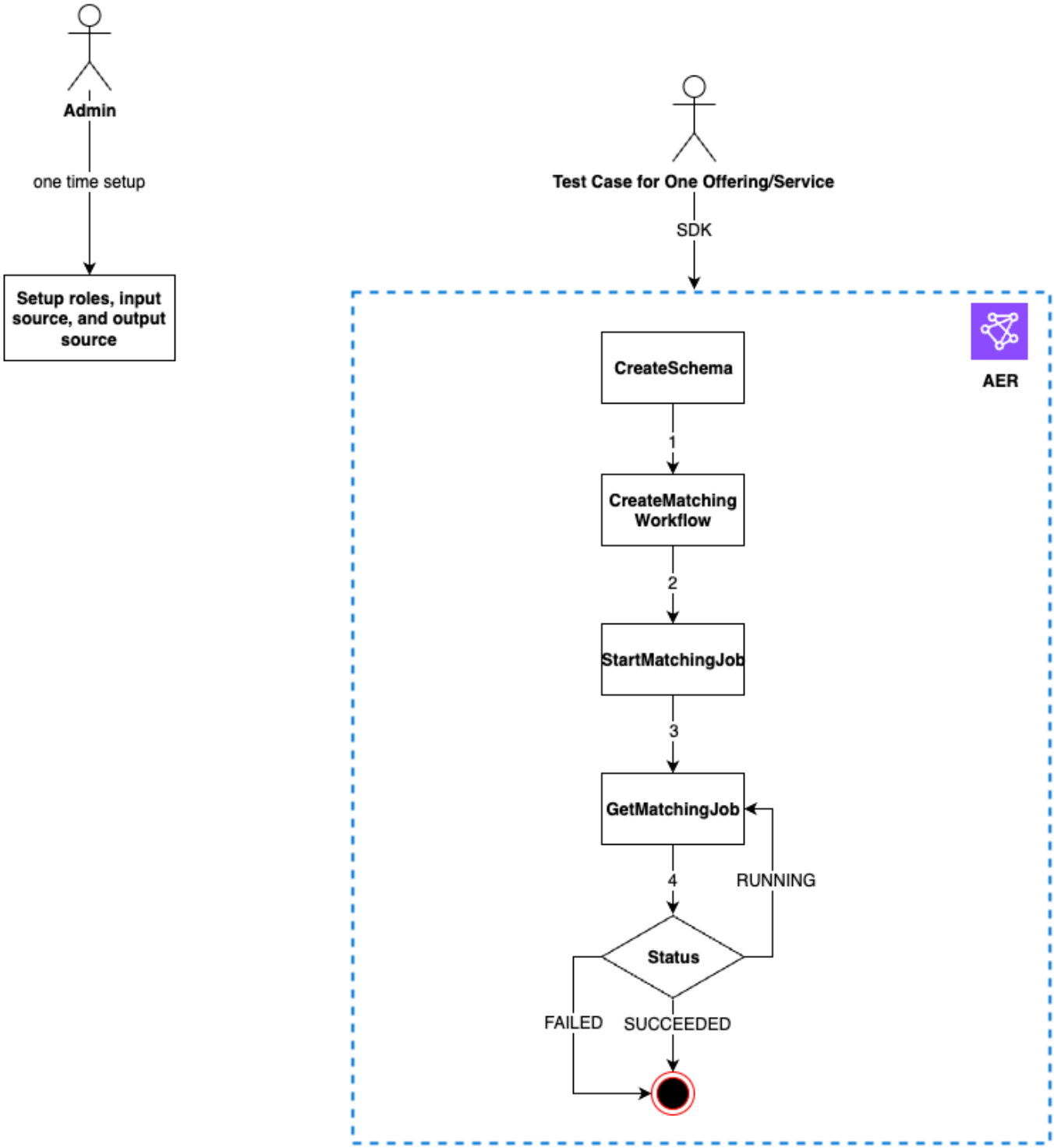
プロバイダーは、テストアカウントと独自のデータを使用して、 AWS Entity Resolution Software Development Kit (SDK) を使用してこれらのend-to-endのテストケースを実行できます。プロバイダーから問題が発生した場合、 は優先エスカレーションパス AWS Entity Resolution を使用して問題をエスカレーションします。さらに、プロバイダーはテスト結果に独自のモニタリングを実装する必要があります。プロバイダーは、これらのテストの実行に使用される AWS アカウント IDsを共有する必要があります AWS Entity Resolution。

実行が成功すると、プロバイダーはデータをセットアップし、 を通じて独自のサービスを使用し AWS Entity Resolution、ジョブステータスはエラーなしで完了を返します。これは、 が提供する APIs を使用してプログラムで実現できます AWS Entity Resolution。

たとえば、プロバイダーはサービスに応じて S3 バケット、入力ソース、ロール、スキーマ、ワークフローを設定できます。これらのセットアップが完了すると、プロバイダーは 200 レコードで 1 日 1 回これらのワークフローを実行してサービスをテストできます。このアプローチでは、プロバイダーは選択した SDK を使用し、テストアカウント AWS Data Exchange を使用して を通じて提供されるサービスに対してend-to-endのテストを実行します。プロバイダーは、サービスごとにこれらのテストを実行することが期待されます。

> **Note**
>
> プロバイダーは AWS Entity Resolution 、テストのためにこれらのワークフローを実行するために使用する AWS アカウント ID (`accountId`)) を指定する必要があります。さらに、プロバイダーはこれらのテストをモニタリングし、合格であることを確認する必要があります。つまり、プロバイダーは障害が発生した場合に通知を有効にして、それに応じて問題に対処する必要があります。

次の図は、一般的なend-to-endのワークフローテストケースを示しています。



プロバイダー統合をテストするには

1. (ワンタイムセットアップ) AWS Entity Resolution の手順に従って のリソースを設定しますセットアップ AWS Entity Resolution。

ワンタイムセットアップ手順を完了すると、ロール、データ、データソースの準備が整います。これで、 AWS Entity Resolution コンソールまたは APIs。

2. AWS Entity Resolution APIsまたはコンソールを使用してプロバイダー統合をテストします。

API

AWS Entity Resolution APIs

1. CreateSchemaMapping API を使用してスキーママッピングを作成します。サポートされているプログラミング言語の完全なリストについては、CreateSchemaMapping API のhttps://docs.aws.amazon.com/entityresolution/latest/apireference/API_CreateSchemaMapping.html#API_CreateSchemaMapping_SeeAlso「」セクションを参照してください。

   スキーママッピングは、マッチングのためにデータを解釈 AWS Entity Resolution する方法を に指示するプロセスです。AWS Entity Resolution が一致するワークフローに読み込む入力データテーブルのスキーマを定義します。

   スキーママッピングを作成するときは、一意の識別子を指定し、AWS Entity Resolution が読み取る入力データの各行に割り当てる必要があります。例えば、`Primary_key`、`Row_ID`、`Record_ID` などが挙げられます。

   Example **`id`** と を含むデータソースのスキーママッピングの作成 **`email`**

   以下は、 `id`と を含むデータソースのスキーママッピングの例です`email`。

```
[
  {
    "fieldName": "id",
    "type": "UNIQUE_ID"
  },
  {
    "fieldName": "email",
    "type": "EMAIL_ADDRESS"
  }
]
```

Example Java SDK を含むデータソース**id**と Java SDK **email**を使用するデータソースのスキーママッピングの作成

Java SDK を含むデータソースidのスキーママッピングの例emailを次に示します。

```
EntityResolutionClient.createSchemaMapping(
            CreateSchemaMappingRequest.builder()
                                      .schemaName(<schema-name>)
                                      .mappedInputFields([

 SchemaInputAttribute.builder().fieldName("id").type("UNIQUE_ID").build(),

 SchemaInputAttribute.builder().fieldName("email").type("EMAIL_ADDRESS").build()
                                       ])
                                      .build()

)
```

2. CreateMatchingWorkflow API を使用して一致するワークフローを作成します。サポートされているプログラミング言語の完全なリストについては、CreateMatchingWorkflow API のhttps://docs.aws.amazon.com/entityresolution/latest/apireference/API_CreateMatchingWorkflow.html#API_CreateMatchingWorkflow_SeeAlso「」セクションを参照してください。

   Example Java SDK を使用した一致するワークフローの作成

   Java SDK を使用した一致するワークフローの例を次に示します。

```
EntityResolutionClient.createMatchingWorkflow(
            CreateMatchingWorkflowRequest.builder()
                                         .workflowName(<workflow-name>)
                                         .inputSourceConfig(

 InputSource.builder().inputSourceARN(<glue-inputsource-from-
step1>).schemaName(<schema-name-from-step2>).build()
                                         )

 .outputSourceConfig(OutputSource.builder().outputS3Path(<output-s3-
path>).output(<output-1>, <output-2>, <output-3>).build())

 .resolutionTechniques(ResolutionTechniques.builder()
```

```
   .resolutionType(PROVIDER)

   .providerProperties(ProviderProperties.builder()

                                          .providerServiceArn(<provider-arn>)

                                           .providerConfiguration(<configuration-
depending-on-service>)


 .intermediateSourceConfiguration(<intermedaite-s3-path>)

                                          .build())

   .build()
                                          .roleArn(<role-from-step1>)
                                          .build()

)
```

一致するワークフローを設定したら、ワークフローを実行できます。

3. StartMatchingJob API を使用して一致するワークフローを実行します。一致するワークフローを実行するには、 `CreateMatchingWorkflow`エンドポイントを使用して一致するワークフローを作成しておく必要があります。

   サポートされているプログラミング言語の完全なリストについては、StartMatchingJob API のhttps://docs.aws.amazon.com/entityresolution/latest/apireference/API_StartMatchingJob.html#API_StartMatchingJob_SeeAlso「」セクションを参照してください。

   Example Java SDK を使用して一致するワークフローを実行する

   Java SDK を使用して一致するワークフローを実行する例を次に示します。

   ```
   EntityResolutionClient.startMatchingJob(StartMatchingJobRequest.builder()
                   .workflowName(<name-of-workflow-from-step3)
                   .build()
   )
   ```

4. GetMatchingJob API を使用してワークフローのステータスをモニタリングします。

この API は、ジョブに関連付けられているステータス、メトリクス、エラー (存在する場合) を返します。

Example Java SDK を使用した一致するワークフローのモニタリング

以下は、Java SDK を使用して一致するワークフロージョブをモニタリングする例です。

```
EntityResolutionClient.getMatchingJob(GetMatchingJobRequest.builder()
                .workflowName(<name-of-workflow-from-step3)
                .jobId(jobId-from-startMatchingJob)
                .build()
)
```

ワークフローが正常に完了するとend-to-endテストは完了します。

Console

AWS Entity Resolution コンソールを使用してプロバイダー統合をテストするには

1. 「」の手順に従ってスキーママッピングを作成しますスキーママッピングの作成。

   スキーママッピングは、マッチングのためにデータを解釈 AWS Entity Resolution する方法を に指示するプロセスです。一致するワークフローに AWS Entity Resolution 読み込む入力データテーブルのスキーマを定義します。

   スキーママッピングを作成するときは、一意の識別子を指定し、 が AWS Entity Resolution 読み取る入力データの各行に割り当てる必要があります。例えば、`Primary_key`、`Row_ID`、`Record_ID` などが挙げられます。

   Example **`id`** と を含むデータソースのスキーママッピング **`email`**

   以下は、 `id`と を含むデータソースのスキーママッピングの例です`email`。

```
[
  {
    "fieldName": "id",
    "type": "UNIQUE_ID"
  },
  {
    "fieldName": "email",
```

```
    "type": "EMAIL_ADDRESS"
  }
]
```

2. 「」の手順に従って、一致するワークフローを作成して実行しますプロバイダーのサービスベースのマッチングワークフローの作成。

   一致するワークフローの作成は、一致する入力データとマッチングの実行方法を指定するようにセットアップしたプロセスです。プロバイダーベースのワークフローでは、アカウントが を通じてプロバイダーサービスとサブスクリプションを持っている場合 AWS Data Exchange、既知の識別子を任意のプロバイダーと照合できます。エンドツーエンドのテストを実行するために使用しているプロバイダーとサービスに応じて、一致するワークフローを設定できます。

   AWS Entity Resolution コンソールは、作成と実行のアクションを 1 つのボタンで組み合わせます。作成と実行を選択すると、一致するワークフローが作成され、ジョブが開始されたことを示すメッセージが表示されます。

3. 一致するワークフローページでワークフローのステータスをモニタリングします。

   ワークフローが正常に完了した場合 (ジョブステータスは完了)、end-to-endテストは完了します。

   一致するワークフローの詳細ページのメトリクスタブで、Last ジョブメトリクスで以下を表示できます。

   - ジョブ ID。
   - 一致するワークフロージョブのステータス: Queued、In progress、Completed、Failed
   - ワークフロージョブの完了時刻。
   - 処理されたレコードの数。
   - 処理されていないレコードの数。
   - 生成された一意の一致 IDs。
   - 入力レコードの数。

   ジョブ履歴で以前に実行された一致するワークフロージョブのジョブメトリクスを表示することもできます。

のセキュリティ AWS Entity Resolution

のクラウドセキュリティが最優先事項 AWS です。お客様は AWS 、セキュリティを最も重視する組織の要件を満たすように構築されたデータセンターとネットワークアーキテクチャを活用できます。

セキュリティは、 AWS お客様とお客様の間の責任共有です。責任共有モデルではこれをクラウドのセキュリティおよびクラウド内のセキュリティと説明しています。

- クラウドのセキュリティ – AWS は、 AWS のサービス で実行されるインフラストラクチャを保護する責任を担います AWS クラウド。 は、お客様が安全に使用できるサービス AWS も提供します。サードパーティーの監査者は、AWS コンプライアンスプログラムコンプライアンスプログラムの一環として、当社のセキュリティの有効性を定期的にテストおよび検証。が適用されるコンプライアンスプログラムの詳細については AWS Entity Resolution、「コンプライアンスプログラムAWS による対象範囲内のサービスコンプライアンスプログラム」を参照してください。
- クラウドのセキュリティ – お客様の責任は、使用する によって決まり AWS のサービス ます。また、ユーザーは、データの機密性、会社の要件、適用される法律や規制など、その他の要因についても責任を負います。

このドキュメントは、 を使用する際の責任共有モデルの適用方法を理解するのに役立ちます AWS Entity Resolution。以下のトピックでは、セキュリティおよびコンプライアンスの目的を達成する AWS Entity Resolution ように を設定する方法を示します。また、 AWS Entity Resolution リソースのモニタリングと保護 AWS のサービス に役立つ他の の使用方法についても説明します。

トピック

- でのデータ保護 AWS Entity Resolution
- の ID とアクセスの管理 AWS Entity Resolution
- のコンプライアンス検証 AWS Entity Resolution
- の耐障害性 AWS Entity Resolution

でのデータ保護 AWS Entity Resolution

責任 AWS 共有モデル、 でのデータ保護に適用されます AWS Entity Resolution。このモデルで説明されているように、 AWS はすべての を実行するグローバルインフラストラクチャを保護する責任があります AWS クラウド。ユーザーは、このインフラストラクチャでホストされるコンテンツに対する管理を維持する責任があります。また、使用する「 AWS のサービス 」のセキュリティ設定

と管理タスクもユーザーの責任となります。データプライバシーの詳細については、データプライバシーに関するよくある質問を参照してください。欧州でのデータ保護の詳細については、AWS セキュリティブログに投稿された「AWS 責任共有モデルおよび GDPR」のブログ記事を参照してください。

データ保護の目的で、認証情報を保護し AWS アカウント 、 AWS IAM Identity Center または AWS Identity and Access Management (IAM) を使用して個々のユーザーを設定することをお勧めします。この方法により、それぞれのジョブを遂行するために必要な権限のみが各ユーザーに付与されます。また、次の方法でデータを保護することもお勧めします:

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。TLS 1.2 は必須ですが、TLS 1.3 を推奨します。
- で API とユーザーアクティビティのログ記録を設定します AWS CloudTrail。CloudTrail 証跡を使用して AWS アクティビティをキャプチャする方法については、「 AWS CloudTrail ユーザーガイド」のCloudTrail 証跡の使用」を参照してください。
- AWS 暗号化ソリューションと、 内のすべてのデフォルトのセキュリティコントロールを使用します AWS のサービス。
- Amazon Macie などの高度な管理されたセキュリティサービスを使用します。これらは、Amazon S3 に保存されている機密データの検出と保護を支援します。
- コマンドラインインターフェイスまたは API AWS を介して にアクセスするときに FIPS 140-3 検証済み暗号化モジュールが必要な場合は、FIPS エンドポイントを使用します。利用可能な FIPS エンドポイントの詳細については、「連邦情報処理規格 (FIPS) 140-3」を参照してください。

お客様の E メールアドレスなどの極秘または機密情報を、タグ、または [名前] フィールドなどの自由形式のテキストフィールドに含めないことを強くお勧めします。これは、コンソール AWS Entity Resolution 、API、または SDK を使用して AWS CLIまたは他の AWS のサービス を操作する場合も同様です。 AWS SDKs タグ、または名前に使用される自由記述のテキストフィールドに入力したデータは、請求または診断ログに使用される場合があります。外部サーバーに URL を提供する場合、そのサーバーへのリクエストを検証できるように、認証情報を URL に含めないことを強くお勧めします。

の保管中のデータ暗号化 AWS Entity Resolution

AWS Entity Resolution はデフォルトで暗号化を提供し、 AWS 所有の暗号化キーを使用して保管中の機密データを保護します。

AWS 所有のキー – デフォルトでこれらのキー AWS Entity Resolution を使用して、個人を特定できるデータを自動的に暗号化します。 AWS が所有するキーを表示、管理、使用したり、その使用を監査したりすることはできません。ただし、データを暗号化するキーを保護するためにアクションを実行する必要はありません。詳細については、AWS Key Management Service デベロッパーガイド の「AWS 所有キー」を参照してください。

デフォルトでは、保管中のデータを暗号化することで、機密データの保護に伴う運用のオーバーヘッドと複雑な作業を軽減できます。同時に、これを使用して、厳格な暗号化コンプライアンスと規制要件を満たす安全なアプリケーションを構築できます。

または、一致するワークフローリソースを作成するときに、暗号化用のカスタマーマネージド KMS キーを指定することもできます。

カスタマーマネージドキー – 機密データの暗号化を許可するために作成、所有、管理する対称カスタマーマネージド KMS キーの使用 AWS Entity Resolution をサポートします。この暗号化層はユーザーが完全に制御できるため、次のようなタスクを実行できます。

- キーポリシーの策定と維持
- IAM ポリシーとグラントの策定と維持
- キーポリシーの有効化と無効化
- キー暗号化マテリアルのローテーション
- タグを追加する
- キーエイリアスの作成
- キー削除のスケジュール設定

詳細については、「 AWS Key Management Service デベロッパーガイド」の「カスタマーマネージドキー」を参照してください。

詳細については AWS KMS、AWS Key Management Service とは」を参照してください。

キー管理

で が許可 AWS Entity Resolution を使用する方法 AWS KMS

AWS Entity Resolution には、カスタマーマネージドキーを使用するための許可が必要です。カスタマーマネージドキーで暗号化された一致するワークフローを作成すると、 は CreateGrant リクエストを送信してユーザーに代わって許可 AWS Entity Resolution を作成します AWS KMS。の権限 AWS KMS は、カスタマーアカウントの KMS キー AWS Entity Resolution へのアクセスを許可する

ために使用されます。 では、次の内部オペレーションでカスタマーマネージドキーを使用するには権限 AWS Entity Resolution が必要です。

- GenerateDataKey リクエストを に送信 AWS KMS して、カスタマーマネージドキーによって暗号化されたデータキーを生成します。
- に Decrypt リクエストを送信 AWS KMS して、暗号化されたデータキーを復号し、データの暗号化に使用できるようにします。

グラントへのアクセスの取り消しや、カスタマーマネージドキーに対するサービスのアクセスの取り消しは、いつでもできます。その場合、カスタマーマネージドキーによって暗号化されたデータにはアクセス AWS Entity Resolution できず、そのデータに依存するオペレーションに影響します。たとえば、グラントを通じてキーへのサービスアクセスを削除し、カスタマーキーで暗号化された一致するワークフローのジョブを開始しようとすると、オペレーションは`AccessDeniedException`エラーを返します。

カスタマーマネージドキーの作成

対称カスタマーマネージドキーは AWS マネジメントコンソール、 または AWS KMS APIs を使用して作成できます。

対称カスタマーマネージドキーを作成するには

AWS Entity Resolution は、対称暗号化 KMS キーを使用した暗号化をサポートしています。「AWS Key Management Service デベロッパーガイド」にある「対称カスタマーマネージドキーの作成」ステップに従います。

キーポリシーステートメント

キーポリシーは、カスタマーマネージドキーへのアクセスを制御します。すべてのカスタマーマネージドキーには、キーポリシーが 1 つだけ必要です。このポリシーには、そのキーを使用できるユーザーとその使用方法を決定するステートメントが含まれています。カスタマーマネージドキーを作成する際に、キーポリシーを指定することができます。詳細については、「 AWS Key Management Service デベロッパーガイド」の「カスタマーマネージドキーへのアクセスの管理」を参照してください。

AWS Entity Resolution リソースでカスタマーマネージドキーを使用するには、キーポリシーで次の API オペレーションを許可する必要があります。

- `kms:DescribeKey` – キー ARN、作成日 (該当する場合は削除日) 、キーの状態、キーマテリアルのオリジンと有効期限 (存在する場合) などの情報を提供します。これには、さまざまなタイプ

の KMS キーを区別するのに役立つ KeySpecなどのフィールドが含まれています。また、キーの使用状況 (暗号化、署名、または MACs の生成と検証) と、KMS キーがサポートするアルゴリズムも表示されます。 KeySpec は SYMMETRIC_DEFAULTで、 KeyUsage は であることを AWS Entity Resolution 検証しますENCRYPT_DECRYPT。

- kms:CreateGrant - カスタマーマネージドキーに許可を追加します。指定された KMS キーへのアクセスを制御する権限を付与します。これにより、必要な権限付与オペレーション AWS Entity Resolution へのアクセスが可能になります。詳細については、「AWS Key Management Service デベロッパーガイド」の「AWS KMS でのグラント」を参照してください。

これにより、 AWS Entity Resolution は以下を実行できます。

- GenerateDataKey を呼び出して、暗号化されたデータキーを生成して保存します。データキーは暗号化にすぐには使用されないからです。
- Decrypt を呼び出して、保存された暗号化データキーを使用して暗号化データにアクセスします。
- RetireGrant へのサービスを許可するために、削除プリンシパルを設定します。

追加できるポリシーステートメントの例を次に示します AWS Entity Resolution。

```
{
      "Sid" : "Allow access to principals authorized to use AWS Entity Resolution",
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "*"
      },
      "Action" : ["kms:DescribeKey","kms:CreateGrant"],
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "kms:ViaService" : "entityresolution.region.amazonaws.com",
          "kms:CallerAccount" : "111122223333"
        }
      }
}
```

ユーザーのアクセス許可

暗号化のデフォルトキーとして KMS キーを設定すると、デフォルトの KMS キーポリシーにより、必要な KMS アクションにアクセスできるすべてのユーザーがこの KMS キーを使用してリソースを

暗号化または復号できます。カスタマーマネージド KMS キー暗号化を使用するには、次のアクションを呼び出すアクセス許可をユーザーに付与する必要があります。

- kms:CreateGrant
- kms:Decrypt
- kms:DescribeKey
- kms:GenerateDataKey

CreateMatchingWorkflow リクエスト中、 AWS Entity Resolution は AWS KMS ユーザーに代わって DescribeKey と CreateGrant リクエストを に送信します。これには、カスタマーマネージド KMS キーを使用してCreateMatchingWorkflowリクエストを行う IAM エンティティが KMS キーポリシーに対するkms:DescribeKeyアクセス許可を持っている必要があります。

CreateIdMappingWorkflow および StartIdMappingJobリクエスト中、 AWS Entity Resolution は AWS KMS ユーザーに代わって DescribeKey および CreateGrant リクエストを に送信します。これには、 を行う IAM エンティティCreateIdMappingWorkflowと、カスタマーマネージド KMS キーを使用して KMS キーポリシーに対するkms:DescribeKeyアクセス許可をStartIdMappingJobリクエストする必要があります。プロバイダーはカスタマーマネージドキーにアクセスして、 AWS Entity Resolution Amazon S3 バケット内のデータを復号化できます。

以下は、Amazon AWS Entity Resolution Amazon S3 バケット内のデータを復号するためにプロバイダーに追加できるポリシーステートメントの例です。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Principal": {
            "AWS": "arn:aws:iam::715724997226:root"
        },
        "Action": [
            "kms:Decrypt"
        ],
        "Resource": "arn:aws:kms:us-east-1:111122223333:key/key-id",
        "Condition": {
            "StringEquals": {
            "kms:ViaService": "s3.us-east-1.amazonaws.com"
```

```
            }
        }
    }]
}
```

各 *<##############>* を独自の情報に置き換えます。

<KMSKeyARN>	AWS KMS Amazon Resource Name.

同様に、StartMatchingJobAPI を呼び出す IAM エンティティには、一致するワークフローで提供されるカスタマーマネージド KMS キーに対する `kms:Decrypt`および アクセス`kms:GenerateDataKey`許可が必要です。

ポリシーでアクセス許可を指定する方法の詳細については、「 AWS Key Management Service デベロッパーガイド」を参照してください。

キーアクセスのトラブルシューティングの詳細については、 AWS Key Management Service デベロッパーガイドを参照してください。

のカスタマーマネージドキーの指定 AWS Entity Resolution

カスタマーマネージドキーは、以下のリソースの第 2 レイヤー暗号化として指定できます。

ワークフローの一致 – 一致するワークフローリソースを作成するときに、KMSArn を入力してデータキーを指定できます。KMSArn は、 AWS Entity Resolution を使用して、リソースに保存されている識別可能な個人データを暗号化します。

KMSArn – AWS KMS カスタマーマネージドキーのキー識別子であるキー ARN を入力します。

2 つの で ID マッピングワークフローを作成または実行している場合、カスタマーマネージドキーを次のリソースの 2 番目のレイヤー暗号化として指定できます AWS アカウント。

ID マッピングワークフローまたは ID マッピングワークフローの開始 – ID マッピングワークフローリソースを作成するとき、または ID マッピングワークフロージョブを開始するときに、KMSArn を入力してデータキーを指定できます。KMSArn は、 AWS Entity Resolution を使用して、リソースに保存されている識別可能な個人データを暗号化します。

KMSArn – AWS KMS カスタマーマネージドキーのキー識別子であるキー ARN を入力します。

Service の暗号化キーのモニタリング AWS Entity Resolution

AWS Entity Resolution サービスリソースで AWS KMS カスタマーマネージドキーを使用する場合、AWS CloudTrail または Amazon CloudWatch Logs を使用して、 が AWS Entity Resolution に送信するリクエストを追跡できます AWS KMS。

次の例はCreateGrant、カスタマーマネージドキーによって暗号化されたデータにアクセスDescribeKeyするために によって呼び出される AWS KMS オペレーションをモニタリング AWS Entity Resolution するための GenerateDataKey、Decrypt、、および の AWS CloudTrail イベントです。

トピック

- CreateGrant
- DescribeKey
- GenerateDataKey
- Decrypt

CreateGrant

AWS KMS カスタマーマネージドキーを使用して一致するワークフローリソースを暗号化すると、はユーザーに代わって の KMS キーにアクセスするCreateGrantリクエスト AWS Entity Resolution を送信します AWS アカウント。が AWS Entity Resolution 作成する権限は、 AWS KMS カスタマーマネージドキーに関連付けられたリソースに固有です。さらに、 RetireGrantオペレーション AWS Entity Resolution を使用して、リソースを削除するときに許可を削除します。

次に、CreateGrant オペレーションを記録するイベントの例を示します。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAIGDTESTANDEXAMPLE:Sampleuser01",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE3",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAIGDTESTANDEXAMPLE:Sampleuser01",
```

```
                "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2021-04-22T17:02:00Z"
            }
        },
        "invokedBy": "entityresolution.amazonaws.com"
    },
    "eventTime": "2021-04-22T17:07:02Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateGrant",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "172.12.34.56",
    "userAgent": "ExampleDesktop/1.0 (V1; OS)",
    "requestParameters": {
        "retiringPrincipal": "entityresolution.region.amazonaws.com",
        "operations": [
            "GenerateDataKey",
            "Decrypt",
        ],
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE",
        "granteePrincipal": "entityresolution.region.amazonaws.com"
    },
    "responseElements": {
        "grantId":
 "0ab0ac0d0b000f00ea00cc0a0e00fc00bce000c000f0000000c0bc0a0000aaafSAMPLE",
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE",
    },
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
```

```
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "111122223333"
}
```

DescribeKey

AWS Entity Resolution は DescribeKeyオペレーションを使用して、一致するリソースに関連付けられた AWS KMS カスタマーマネージドキーがアカウントとリージョンに存在するかどうかを確認します。

以下のイベント例では DescribeKey オペレーションを記録しています。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAIGDTESTANDEXAMPLE:Sampleuser01",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE3",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAIGDTESTANDEXAMPLE:Sampleuser01",
                "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2021-04-22T17:02:00Z"
            }
        },
        "invokedBy": "entityresolution.amazonaws.com"
    },
    "eventTime": "2021-04-22T17:07:02Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DescribeKey",
    "awsRegion": "us-west-2",
```

```
    "sourceIPAddress": "172.12.34.56",
    "userAgent": "ExampleDesktop/1.0 (V1; OS)",
    "requestParameters": {
        "keyId": "00dd0db0-0000-0000-ac00-b0c000SAMPLE"
    },
    "responseElements": null,
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "111122223333"
}
```

GenerateDataKey

一致するワークフローリソースの AWS KMS カスタマーマネージドキーを有効にすると、 は Amazon Simple Storage Service (Amazon S3) を介して、 AWS KMS リソースの AWS KMS カスタマーマネージドキーを指定するGenerateDataKeyリクエストを AWS Entity Resolution に送信します。

以下のイベント例では GenerateDataKey オペレーションを記録しています。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "s3.amazonaws.com"
    },
    "eventTime": "2021-04-22T17:07:02Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateDataKey",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "172.12.34.56",
```

```
    "userAgent": "ExampleDesktop/1.0 (V1; OS)",
    "requestParameters": {
        "keySpec": "AES_256",
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
    },
    "responseElements": null,
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "111122223333",
    "sharedEventID": "57f5dbee-16da-413e-979f-2c4c6663475e"
}
```

Decrypt

一致するワークフローリソースの AWS KMS カスタマーマネージドキーを有効にすると、 は Amazon Simple Storage Service (Amazon S3) を介して、リソースの AWS KMS カスタマーマネージドキー AWS KMS を指定するDecryptリクエストを AWS Entity Resolution に送信します。

以下のイベント例では Decrypt オペレーションを記録しています。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "s3.amazonaws.com"
    },
    "eventTime": "2021-04-22T17:10:51Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "us-west-2",
```

```
    "sourceIPAddress": "172.12.34.56",
    "userAgent": "ExampleDesktop/1.0 (V1; OS)",
    "requestParameters": {
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE",
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT"
    },
    "responseElements": null,
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "111122223333",
    "sharedEventID": "dc129381-1d94-49bd-b522-f56a3482d088"
}
```


考慮事項

AWS Entity Resolution は、新しいカスタマーマネージド KMS キーを使用したマッチングワークフローの更新をサポートしていません。このような場合は、カスタマーマネージド KMS キーを使用して新しいワークフローを作成できます。

詳細情報

次のリソースは、保管時のデータ暗号化についての詳細を説明しています。

AWS Key Management Service の基本概念の詳細については、 AWS Key Management Service デベロッパーガイドを参照してください。

AWS Key Management Service のセキュリティのベストプラクティスの詳細については、 AWS Key Management Service デベロッパーガイドを参照してください。

インターフェイスエンドポイント (AWS PrivateLink) AWS Entity Resolution を使用した へのアクセス

を使用して AWS PrivateLink 、VPC と の間にプライベート接続を作成できます AWS Entity Resolution。インターネットゲートウェイ、NAT デバイス、VPN 接続、または Direct Connect 接続を使用せずに、VPC 内にある AWS Entity Resolution かのように にアクセスできます。VPC 内のインスタンスは AWS Entity Resolutionにアクセスするためにパブリック IP アドレスを必要としません。

このプライベート接続を確立するには、 AWS PrivateLinkを利用したインターフェイスエンドポイントを作成します。インターフェイスエンドポイントに対して有効にする各サブネットにエンドポイントネットワークインターフェイスを作成します。これらは、 AWS Entity Resolution宛てのトラフィックのエントリポイントとして機能するリクエスタ管理型ネットワークインターフェイスです。

詳細については、「 AWS PrivateLink ガイド」の「Access AWS のサービス through AWS PrivateLink」を参照してください。

に関する考慮事項 AWS Entity Resolution

のインターフェイスエンドポイントを設定する前に AWS Entity Resolution、「 AWS PrivateLink ガイド」の「考慮事項」を参照してください。

AWS Entity Resolution は、インターフェイスエンドポイントを介したすべての API アクションの呼び出しをサポートしています。

VPC エンドポイントポリシーがサポートされています AWS Entity Resolution。デフォルトでは、インターフェイスエンドポイント経由での AWS Entity Resolution への完全なアクセスが許可されます。または、セキュリティグループをエンドポイントのネットワークインターフェイスに関連付けて、インターフェイスエンドポイント経由での AWS Entity Resolution へのトラフィックを制御することもできます。

のインターフェイスエンドポイントを作成する AWS Entity Resolution

Amazon VPC コンソールまたは AWS Command Line Interface () AWS Entity Resolution を使用して、 のインターフェイスエンドポイントを作成できますAWS CLI。詳細については、「AWS PrivateLink ガイド」の「インターフェイスエンドポイントを作成」を参照してください。

次のサービス名 AWS Entity Resolution を使用して のインターフェイスエンドポイントを作成します。

```
com.amazonaws.region.entityresolution
```

AWS Entity Resolution は、FIPS (Federal Information Processing Standard) 準拠のエンドポイントもサポートしています。FIPS エンドポイントを使用するには、次のサービス名を使用します。

```
com.amazonaws.region.entityresolution-fips
```

インターフェイスエンドポイントのプライベート DNS を有効にすると、リージョンのデフォルト DNS 名を使用して、 AWS Entity Resolution への API リクエストを実行できます。例えば、`entityresolution.us-east-1.amazonaws.com`。

インターフェイスエンドポイントのエンドポイントポリシーを作成する

エンドポイントポリシーは、インターフェイスエンドポイントにアタッチできる IAM リソースです。デフォルトのエンドポイントポリシーでは、インターフェイスエンドポイント AWS Entity Resolution を介して へのフルアクセスを許可します。VPC AWS Entity Resolution から に許可されるアクセスを制御するには、カスタムエンドポイントポリシーをインターフェイスエンドポイントにアタッチします。

エンドポイントポリシーは以下の情報を指定します。

- アクションを実行できるプリンシパル (AWS アカウント、IAM ユーザー、IAM ロール)。
- 実行可能なアクション。
- このアクションを実行できるリソース。

詳細については、AWS PrivateLink ガイドのControl access to services using endpoint policies (エンドポイントポリシーを使用してサービスへのアクセスをコントロールする)を参照してください。

例: AWS Entity Resolution アクションの VPC エンドポイントポリシー

以下は、カスタムエンドポイントポリシーの例です。このポリシーをインターフェイスエンドポイントにアタッチすると、すべてのリソースのすべてのプリンシパルに対して、リストされた AWS Entity Resolution アクションへのアクセスが許可されます。

```
{
   "Statement": [
      {
         "Principal": "*",
         "Effect": "Allow",
```

```
            "Action": [
               "entityresolution:CreateMatchingWorkflow",
               "entityresolution:StartMatchingJob",
               "entityresolution:GetMatchingJob"
            ],
            "Resource":"*"
         }
      ]
}
```

の ID とアクセスの管理 AWS Entity Resolution

AWS Identity and Access Management (IAM) は、管理者が AWS リソースへのアクセスを安全に制御 AWS のサービス するのに役立つ です。IAM 管理者は、誰を認証 (サインイン) し、誰に AWS Entity Resolution リソースの使用を許可する (アクセス許可を付与する) かを制御します。IAM は、追加料金なしで使用できる AWS のサービス です。

> **Note**
>
> AWS Entity Resolution はクロスアカウントポリシーをサポートしています。詳細については、IAM ユーザーガイドのIAM でのクロスアカウントリソースアクセスを参照してください。

トピック

- オーディエンス
- アイデンティティを使用した認証
- ポリシーを使用したアクセスの管理
- が IAM と AWS Entity Resolution 連携する方法
- のアイデンティティベースのポリシーの例 AWS Entity Resolution
- AWS の 管理ポリシー AWS Entity Resolution
- AWS Entity Resolution ID とアクセスのトラブルシューティング

オーディエンス

AWS Identity and Access Management (IAM) の使用方法は、ロールによって異なります。

- サービスユーザー - 機能にアクセスできない場合は、管理者にアクセス許可をリクエストします (「AWS Entity Resolution ID とアクセスのトラブルシューティング」を参照)。
- サービス管理者 - ユーザーアクセスを決定し、アクセス許可リクエストを送信します (「が IAM と AWS Entity Resolution 連携する方法」を参照)
- IAM 管理者 - アクセスを管理するためのポリシーを作成します (「のアイデンティティベースのポリシーの例 AWS Entity Resolution」を参照)

アイデンティティを使用した認証

認証とは、ID 認証情報 AWS を使用して にサインインする方法です。、IAM ユーザー AWS アカウントのルートユーザー、または IAM ロールを引き受けることで認証される必要があります。

(AWS IAM Identity Center IAM Identity Center)、シングルサインオン認証、Google/Facebook 認証情報などの ID ソースからの認証情報を使用して、フェデレーティッド ID としてサインインできます。サインインの詳細については、「AWS サインイン ユーザーガイド」の「AWS アカウントにサインインする方法」を参照してください。

プログラムによるアクセスの場合、 は SDK と CLI AWS を提供してリクエストを暗号化して署名します。詳細については、「IAM ユーザーガイド」の「API リクエストに対するAWS 署名バージョン 4」を参照してください。

AWS アカウント ルートユーザー

を作成するときは AWS アカウント、すべての AWS のサービス および リソースへの完全なアクセス権を持つ AWS アカウント ルートユーザーと呼ばれる 1 つのサインインアイデンティティから始めます。日常的なタスクには、ルートユーザーを使用しないことを強くお勧めします。ルートユーザー認証情報を必要とするタスクについては、「IAM ユーザーガイド」の「ルートユーザー認証情報が必要なタスク」を参照してください。

フェデレーテッドアイデンティティ

ベストプラクティスとして、人間のユーザーが一時的な認証情報 AWS のサービス を使用して にアクセスするには、ID プロバイダーとのフェデレーションを使用する必要があります。

フェデレーティッド ID は、エンタープライズディレクトリ、ウェブ ID プロバイダー、または ID Directory Service ソースの認証情報 AWS のサービス を使用して にアクセスするユーザーです。フェデレーテッドアイデンティティは、一時的な認証情報を提供するロールを引き受けます。

アクセスを一元管理する場合は、 AWS IAM Identity Centerをお勧めします。詳細については、「AWS IAM Identity Center ユーザーガイド」の「IAM アイデンティティセンターとは」を参照してください。

IAM ユーザーとグループ

IAM ユーザーは、特定の個人やアプリケーションに対する特定のアクセス許可を持つアイデンティティです。長期認証情報を持つ IAM ユーザーの代わりに一時的な認証情報を使用することをお勧めします。詳細については、IAM ユーザーガイドの「ID プロバイダーとのフェデレーションを使用して にアクセスする必要がある AWS」を参照してください。

IAM グループは、IAM ユーザーの集合を指定し、大量のユーザーに対するアクセス許可の管理を容易にします。詳細については、「IAM ユーザーガイド」の「IAM ユーザーに関するユースケース」を参照してください。

IAM ロール

IAM ロールは、特定のアクセス許可を持つアイデンティであり、一時的な認証情報を提供します。ユーザーから IAM ロール (コンソール) に切り替えるか、 または API オペレーションを呼び出すことで、ロールを引き受けることができます。 AWS CLI AWS 詳細については、「IAM ユーザーガイド」の「ロールを引き受けるための各種方法」を参照してください。

IAM ロールは、フェデレーションユーザーアクセス、一時的な IAM ユーザーのアクセス許可、クロスアカウントアクセス、クロスサービスアクセス、および Amazon EC2 で実行するアプリケーションに役立ちます。詳細については、IAM ユーザーガイド の IAM でのクロスアカウントリソースアクセス を参照してください。

ポリシーを使用したアクセスの管理

でアクセスを制御する AWS には、ポリシーを作成し、ID AWS またはリソースにアタッチします。ポリシーは、アイデンティティまたはリソースに関連付けられたときにアクセス許可を定義します。は、プリンシパルがリクエストを行うときにこれらのポリシー AWS を評価します。ほとんどのポリシーは JSON ドキュメント AWS として に保存されます。JSON ポリシードキュメントの詳細については、「IAM ユーザーガイド」の「JSON ポリシー概要」を参照してください。

管理者は、ポリシーを使用して、どのプリンシパルがどのリソースに対して、どのような条件でアクションを実行できるかを定義することで、誰が何にアクセスできるかを指定します。

デフォルトでは、ユーザーやロールにアクセス許可はありません。IAM 管理者は IAM ポリシーを作成してロールに追加し、このロールをユーザーが引き受けられるようにします。IAM ポリシーは、オペレーションの実行方法を問わず、アクセス許可を定義します。

アイデンティティベースのポリシー

アイデンティティベースのポリシーは、アイデンティティ (ユーザー、グループ、またはロール) にアタッチできる JSON アクセス許可ポリシードキュメントです。これらのポリシーは、アイデンティティがどのリソースに対してどのような条件下でどのようなアクションを実行できるかを制御します。アイデンティティベースポリシーの作成方法については、IAM ユーザーガイド の カスタマー管理ポリシーでカスタム IAM アクセス許可を定義する を参照してください。

アイデンティティベースのポリシーは、インラインポリシー (単一の ID に直接埋め込む) または管理ポリシー (複数の ID にアタッチされたスタンドアロンポリシー) にすることができます。管理ポリシーとインラインポリシーのいずれかを選択する方法については、「IAM ユーザーガイド」の「管理ポリシーとインラインポリシーのいずれかを選択する」を参照してください。

リソースベースのポリシー

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。例としては、IAM ロール信頼ポリシーや Amazon S3 バケットポリシーなどがあります。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。リソースベースのポリシーでは、プリンシパルを指定する必要があります。

リソースベースのポリシーは、そのサービス内にあるインラインポリシーです。リソースベースのポリシーでは、IAM の AWS マネージドポリシーを使用できません。

その他のポリシータイプ

AWS は、より一般的なポリシータイプによって付与されるアクセス許可の上限を設定できる追加のポリシータイプをサポートしています。

- アクセス許可の境界 – アイデンティティベースのポリシーで IAM エンティティに付与することのできるアクセス許可の数の上限を設定します。詳細については、「IAM ユーザーガイド」の「IAM エンティティのアクセス許可境界」を参照してください。
- サービスコントロールポリシー (SCP) - AWS Organizations内の組織または組織単位の最大のアクセス許可を指定します。詳細については、「AWS Organizations ユーザーガイド」の「サービスコントロールポリシー」を参照してください。

- リソースコントロールポリシー (RCP) – は、アカウント内のリソースで利用できる最大数のアクセス許可を定義します。詳細については、「AWS Organizations ユーザーガイド」の「リソースコントロールポリシー (RCP)」を参照してください。
- セッションポリシー – ロールまたはフェデレーションユーザーの一時セッションを作成する際にパラメータとして渡される高度なポリシーです。詳細については、「IAM ユーザーガイド」の「セッションポリシー」を参照してください。

複数のポリシータイプ

1 つのリクエストに複数のタイプのポリシーが適用されると、結果として作成されるアクセス許可を理解するのがさらに難しくなります。が複数のポリシータイプが関与する場合にリクエストを許可するかどうか AWS を決定する方法については、「IAM ユーザーガイド」の「ポリシー評価ロジック」を参照してください。

が IAM と AWS Entity Resolution 連携する方法

IAM を使用して へのアクセスを管理する前に AWS Entity Resolution、 で使用できる IAM 機能を確認してください AWS Entity Resolution。

で使用できる IAM 機能 AWS Entity Resolution

IAM 機能	AWS Entity Resolution サポート
アイデンティティベースのポリシー	あり
リソースベースのポリシー	はい
ポリシーアクション	あり
ポリシーリソース	あり
ポリシー条件キー	あり
ACL	なし
ABAC (ポリシー内のタグ)	部分的
一時認証情報	あり

IAM 機能	AWS Entity Resolution サポート
転送アクセスセッション (FAS)	あり
サービスロール	あり
サービスリンクロール	いいえ

AWS Entity Resolution および他の AWS のサービスがほとんどの IAM 機能と連携する方法の概要については、「IAM ユーザーガイド」のAWS 「IAM と連携する のサービス」を参照してください。

のアイデンティティベースのポリシー AWS Entity Resolution

アイデンティティベースのポリシーのサポート: あり

アイデンティティベースポリシーは、IAM ユーザー、ユーザーグループ、ロールなど、アイデンティティにアタッチできる JSON 許可ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件をコントロールします。アイデンティティベースポリシーの作成方法については、「IAM ユーザーガイド」の「カスタマー管理ポリシーでカスタム IAM アクセス許可を定義する」を参照してください。

IAM アイデンティティベースのポリシーでは、許可または拒否するアクションとリソース、およびアクションを許可または拒否する条件を指定できます。JSON ポリシーで使用できるすべての要素について学ぶには、「IAM ユーザーガイド」の「IAM JSON ポリシーの要素のリファレンス」を参照してください。

のアイデンティティベースのポリシーの例 AWS Entity Resolution

AWS Entity Resolution アイデンティティベースのポリシーの例を表示するには、「」を参照してくださいのアイデンティティベースのポリシーの例 AWS Entity Resolution。

内のリソースベースのポリシー AWS Entity Resolution

リソースベースのポリシーのサポート: あり

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあげられます。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使

用して特定のリソースへのアクセスをコントロールできます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーで、プリンシパルを指定する必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーティッドユーザー、または を含めることができます AWS のサービス。

クロスアカウントアクセスを有効にするには、全体のアカウント、または別のアカウントの IAM エンティティを、リソースベースのポリシーのプリンシパルとして指定します。詳細については、IAM ユーザーガイドのIAM でのクロスアカウントリソースアクセスを参照してください。

のポリシーアクション AWS Entity Resolution

ポリシーアクションのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

JSON ポリシーの `Action` 要素にはポリシー内のアクセスを許可または拒否するために使用できるアクションが記述されます。このアクションは関連付けられたオペレーションを実行するためのアクセス許可を付与するポリシーで使用されます。

AWS Entity Resolution アクションのリストを確認するには、「サービス認可リファレンス」の「で定義されるアクション AWS Entity Resolution」を参照してください。

のポリシーアクションは、アクションの前に次のプレフィックス AWS Entity Resolution を使用します。

```
entityresolution
```

単一のステートメントで複数のアクションを指定するには、アクションをカンマで区切ります。

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

AWS Entity Resolution アイデンティティベースのポリシーの例を表示するには、「」を参照してくださいのアイデンティティベースのポリシーの例 AWS Entity Resolution。

のポリシーリソース AWS Entity Resolution

ポリシーリソースのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

`Resource` JSON ポリシー要素はアクションが適用されるオブジェクトを指定します。ベストプラクティスとして、Amazon リソースネーム (ARN) を使用してリソースを指定します。リソースレベルのアクセス許可をサポートしないアクションの場合は、ステートメントがすべてのリソースに適用されることを示すために、ワイルドカード (*) を使用します。

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

AWS Entity Resolution リソースタイプとその ARNs「 で定義されるリソース AWS Entity Resolution」を参照してください。 どのアクションで各リソースの ARN を指定できるかについては、「AWS Entity Resolutionで定義されるアクション」を参照してください。

AWS Entity Resolution アイデンティティベースのポリシーの例を表示するには、「」を参照してくださいのアイデンティティベースのポリシーの例 AWS Entity Resolution。

のポリシー条件キー AWS Entity Resolution

サービス固有のポリシー条件キーのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

`Condition` 要素は、定義された基準に基づいてステートメントが実行される時期を指定します。イコールや未満などの条件演算子を使用して条件式を作成して、ポリシーの条件とリクエスト内の値を一致させることができます。すべての AWS グローバル条件キーを確認するには、「IAM ユーザーガイド」のAWS 「グローバル条件コンテキストキー」を参照してください。

AWS Entity Resolution 条件キーのリストを確認するには、「サービス認可リファレンス」の「 の条件キー AWS Entity Resolution」を参照してください。条件キーを使用できるアクションとリソースについては、「 で定義されるアクション AWS Entity Resolution」を参照してください。

AWS Entity Resolution アイデンティティベースのポリシーの例を表示するには、「」を参照してくださいのアイデンティティベースのポリシーの例 AWS Entity Resolution。

ACLs AWS Entity Resolution

ACL のサポート: なし

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするためのアクセス許可を持つかを制御します。ACL はリソースベースのポリシーに似ていますが、JSON ポリシードキュメント形式は使用しません。

を使用した ABAC AWS Entity Resolution

ABAC (ポリシー内のタグ) のサポート: 一部

属性ベースのアクセスコントロール (ABAC) は、タグと呼ばれる属性に基づいてアクセス許可を定義する認可戦略です。IAM エンティティと AWS リソースにタグをアタッチし、プリンシパルのタグがリソースのタグと一致するときにオペレーションを許可する ABAC ポリシーを設計できます。

タグに基づいてアクセスを管理するには、`aws:ResourceTag/key-name`、`aws:RequestTag/key-name`、または `aws:TagKeys` の条件キーを使用して、ポリシーの条件要素でタグ情報を提供します。

サービスがすべてのリソースタイプに対して 3 つの条件キーすべてをサポートする場合、そのサービスの値はありです。サービスが一部のリソースタイプに対してのみ 3 つの条件キーのすべてをサポートする場合、値は「部分的」になります。

ABAC の詳細については、「IAM ユーザーガイド」の「ABAC 認可でアクセス許可を定義する」を参照してください。ABAC をセットアップする手順を説明するチュートリアルについては、「IAM ユーザーガイド」の「属性ベースのアクセスコントロール (ABAC) を使用する」を参照してください。

での一時的な認証情報の使用 AWS Entity Resolution

一時的な認証情報のサポート: あり

一時的な認証情報は、 AWS リソースへの短期的なアクセスを提供し、フェデレーションまたはスイッチロールの使用時に自動的に作成されます。長期的なアクセスキーを使用する代わりに、一時的

な認証情報を動的に生成 AWS することをお勧めします。詳細については、「IAM ユーザーガイド」の「IAM の一時的な認証情報」および「AWS のサービス と IAM との連携」を参照してください。

の転送アクセスセッション AWS Entity Resolution

転送アクセスセッション (FAS) のサポート: あり

転送アクセスセッション (FAS) は、 を呼び出すプリンシパルのアクセス許可と AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストをリクエストする を使用します。FAS リクエストを行う際のポリシーの詳細については、「転送アクセスセッション」を参照してください。

のサービスロール AWS Entity Resolution

サービスロールのサポート: あり

サービスロールとは、サービスがユーザーに代わってアクションを実行するために引き受ける IAM ロールです。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、IAM ユーザーガイドの AWS のサービスに許可を委任するロールを作成するを参照してください。

> **Warning**
>
> サービスロールのアクセス許可を変更すると、 AWS Entity Resolution 機能が破損する可能性があります。 AWS Entity Resolution が指示する場合にのみ、サービスロールを編集します。

のサービスにリンクされたロール AWS Entity Resolution

サービスにリンクされたロールのサポート: なし

サービスにリンクされたロールは、 にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント 、サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールのアクセス許可を表示できますが、編集することはできません。

サービスにリンクされたロールの作成または管理の詳細については、「IAM と提携するAWS のサービス」を参照してください。表の「サービスリンクロール」列に Yes と記載されたサービスを見つ

けます。サービスにリンクされたロールに関するドキュメントをサービスで表示するには、[はい] リンクを選択します。

のアイデンティティベースのポリシーの例 AWS Entity Resolution

デフォルトでは、 ユーザーおよびロールには、 AWS Entity Resolution リソースを作成または変更する権限はありません。IAM 管理者は、リソースで必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。

これらのサンプルの JSON ポリシードキュメントを使用して IAM アイデンティティベースのポリシーを作成する方法については、「IAM ユーザーガイド」の「IAM ポリシーを作成する (コンソール)」を参照してください。

各リソースタイプの ARN の形式など AWS Entity Resolution、 で定義されるアクションとリソースタイプの詳細については、「サービス認可リファレンス」の「 のアクション、リソース、および条件キー AWS Entity Resolution」を参照してください。 ARNs

トピック

- ポリシーに関するベストプラクティス
- AWS Entity Resolution コンソールの使用
- 自分の権限の表示をユーザーに許可する

ポリシーに関するベストプラクティス

ID ベースのポリシーは、アカウント内の AWS Entity Resolution リソースを作成、アクセス、または削除できるかどうかを決定します。これらのアクションでは、 AWS アカウントに費用が発生する場合があります。アイデンティティベースポリシーを作成したり編集したりする際には、以下のガイドラインと推奨事項に従ってください:

- AWS 管理ポリシーを開始し、最小特権のアクセス許可に移行する – ユーザーとワークロードにアクセス許可の付与を開始するには、多くの一般的なユースケースにアクセス許可を付与するAWS 管理ポリシーを使用します。これらは で使用できます AWS アカウント。ユースケースに固有の AWS カスタマー管理ポリシーを定義することで、アクセス許可をさらに減らすことをお勧めします。詳細については、IAM ユーザーガイド の AWS マネージドポリシー または ジョブ機能のAWS マネージドポリシー を参照してください。
- 最小特権を適用する – IAM ポリシーでアクセス許可を設定する場合は、タスクの実行に必要な許可のみを付与します。これを行うには、特定の条件下で特定のリソースに対して実行できるアク

ションを定義します。これは、最小特権アクセス許可とも呼ばれています。IAM を使用して許可を適用する方法の詳細については、IAM ユーザーガイド の IAM でのポリシーとアクセス許可 を参照してください。

- IAM ポリシーで条件を使用してアクセスをさらに制限する - ポリシーに条件を追加して、アクションやリソースへのアクセスを制限できます。たとえば、ポリシー条件を記述して、すべてのリクエストを SSL を使用して送信するように指定できます。条件を使用して、サービスアクションが などの特定の を通じて使用されている場合に AWS のサービス、サービスアクションへのアクセスを許可することもできます CloudFormation。詳細については、IAM ユーザーガイド の IAM JSON ポリシー要素:条件 を参照してください。
- IAM アクセスアナライザー を使用して IAM ポリシーを検証し、安全で機能的な権限を確保する - IAM アクセスアナライザー は、新規および既存のポリシーを検証して、ポリシーが IAM ポリシー言語 (JSON) および IAM のベストプラクティスに準拠するようにします。IAM アクセスアナライザーは 100 を超えるポリシーチェックと実用的な推奨事項を提供し、安全で機能的なポリシーの作成をサポートします。詳細については、IAM ユーザーガイド の IAM Access Analyzer でポリシーを検証する を参照してください。
- 多要素認証 (MFA) を要求する – で IAM ユーザーまたはルートユーザーを必要とするシナリオがある場合は AWS アカウント、セキュリティを強化するために MFA を有効にします。API オペレーションが呼び出されるときに MFA を必須にするには、ポリシーに MFA 条件を追加します。詳細については、IAM ユーザーガイド の MFA を使用した安全な API アクセス を参照してください。

IAM でのベストプラクティスの詳細については、IAM ユーザーガイド の IAM でのセキュリティのベストプラクティス を参照してください。

AWS Entity Resolution コンソールの使用

AWS Entity Resolution コンソールにアクセスするには、最小限のアクセス許可のセットが必要です。これらのアクセス許可により、 の AWS Entity Resolution リソースの詳細を一覧表示および表示できます AWS アカウント。最小限必要な許可よりも制限が厳しいアイデンティティベースのポリシーを作成すると、そのポリシーを持つエンティティ (ユーザーまたはロール) に対してコンソールが意図したとおりに機能しません。

AWS CLI または AWS API のみを呼び出すユーザーには、最小限のコンソールアクセス許可を付与する必要はありません。代わりに、実行しようとしている API オペレーションに一致するアクションのみへのアクセスが許可されます。

ユーザーとロールが AWS Entity Resolution 引き続きコンソールを使用できるようにするには、エンティティに AWS Entity Resolution *ConsoleAccess*または *ReadOnly* AWS 管理ポリシーもアタッ

チします。詳細については、「IAM ユーザーガイド」の「ユーザーへのアクセス許可の追加」を参照してください。

自分の権限の表示をユーザーに許可する

この例では、ユーザーアイデンティティにアタッチされたインラインおよびマネージドポリシーの表示を IAM ユーザーに許可するポリシーの作成方法を示します。このポリシーには、コンソールで、または AWS CLI または AWS API を使用してプログラムでこのアクションを実行するアクセス許可が含まれています。

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

AWS の 管理ポリシー AWS Entity Resolution

AWS 管理ポリシーは、 によって作成および管理されるスタンドアロンポリシーです AWS。 AWS 管理ポリシーは、ユーザー、グループ、ロールにアクセス許可の割り当てを開始できるように、多くの一般的なユースケースにアクセス許可を付与するように設計されています。

AWS 管理ポリシーは、すべての AWS お客様が使用できるため、特定のユースケースに対して最小特権のアクセス許可を付与しない場合があることに注意してください。ユースケースに固有のカスタマー管理ポリシーを定義して、アクセス許可を絞り込むことをお勧めします。

AWS 管理ポリシーで定義されているアクセス許可は変更できません。が AWS マネージドポリシーで定義されたアクセス許可 AWS を更新すると、ポリシーがアタッチされているすべてのプリンシパル ID (ユーザー、グループ、ロール) に影響します。 AWS は、新しい が起動されるか、新しい API オペレーション AWS のサービス が既存のサービスで使用できるようになったときに、 AWS マネージドポリシーを更新する可能性が最も高くなります。

詳細については、「IAM ユーザーガイド」の「AWS マネージドポリシー」を参照してください。

AWS マネージドポリシー: AWSEntityResolutionConsoleFullAccess

`AWSEntityResolutionConsoleFullAccess` ポリシーを IAM アイデンティティにアタッチできます。

このポリシーは、 AWS Entity Resolution エンドポイントとリソースへのフルアクセスを許可します。

このポリシーでは、S3、Tagging AWS Glue、Amazon EventBridge AWS のサービス などの関連への特定の読み取りアクセスも許可 AWS Data Exchange するため、コンソールは選択肢を表示し AWS KMS、選択したものを使用してエンティティ解決アクションを実行できます。さらに、このポリシーは Amazon Connect Customer Profiles APIsし、自動一致結果処理の統合を有効にします。一部のリソースは、サービス名 を含むように絞り込まれます`entityresolution`。

AWS Entity Resolution は、関連する AWS リソースに対してアクションを実行するために渡されたロールに依存するため、このポリシーは目的のロールを選択して渡すアクセス許可も付与します。

アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- `EntityResolutionAccess` – プリンシパルに AWS Entity Resolution エンドポイントとリソースへのフルアクセスを許可します。
- `GlueSourcesConsoleDisplay` – ユーザーエクスペリエンスのために、データソースオプションとして AWS Glue テーブルを一覧表示し、データソースのテーブルスキーマをインポートするアクセス許可を付与します。
- `S3BucketsConsoleDisplay` – すべての S3 バケットをデータソースオプションとして一覧表示するアクセス許可を付与します。
- `S3SourcesConsoleDisplay` – S3 バケットをデータソースオプションとして表示するためのアクセス許可を付与します。
- `TaggingConsoleDisplay` – タグ付けキーと値を読み取るアクセス許可を付与します。
- `KMSConsoleDisplay` – データソースを復号および暗号化 AWS Key Management Service するために、 でキーを記述し、エイリアスを一覧表示するアクセス許可を付与します。
- `ListRolesToPickForPassing` – ユーザーが渡すロールを選択できるように、すべてのロールを一覧表示するアクセス許可を付与します。
- `PassRoleToEntityResolutionService` – 絞り込まれたロールを AWS Entity Resolution サービスに渡すためのアクセス許可を付与します。
- `ManageEventBridgeRules` – S3 通知を取得するための Amazon EventBridge ルールを作成、更新、削除するアクセス許可を付与します。
- `ADXReadAccess` – 顧客が使用権限を持っているかサブスクリプションを持っているか AWS Data Exchange を確認するための へのアクセス権を付与します。
- `CustomerProfilesIntegrationAccess` – Amazon Connect と Amazon Connect Customer Profiles APIsし、 と Amazon Connect Customer Profiles の統合 AWS Entity Resolution を有効にして、一致結果の自動処理を行います。

このポリシーのアクセス許可を確認するには、「 AWS マネージドポリシーリファレンス」のAWSEntityResolutionConsoleFullAccess」を参照してください。

AWS マネージドポリシー: AWSEntityResolutionConsoleReadOnlyAccess

IAM エンティティに `AWSEntityResolutionConsoleReadOnlyAccess` をアタッチできます。

このポリシーは、 AWS Entity Resolution エンドポイントとリソースへの読み取り専用アクセスを許可します。

アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- `EntityResolutionRead` – プリンシパルに AWS Entity Resolution エンドポイントとリソースへの読み取り専用アクセスを許可します。

このポリシーのアクセス許可を確認するには、「 AWS マネージドポリシーリファレンス」の[AWSEntityResolutionConsoleReadOnlyAccess」を参照してください。

AWS Entity Resolution AWS 管理ポリシーの更新

このサービスがこれらの変更の追跡を開始 AWS Entity Resolution してからの の AWS 管理ポリシーの更新に関する詳細を表示します。このページの変更に関する自動アラートについては、 AWS Entity Resolution ドキュメント履歴ページの RSS フィードにサブスクライブしてください。

変更	説明	日付
`AWSEntityResolutionConsoleFullAccess` – 既存のポリシーの更新	一致結果の自動処理のために Amazon Connect Customer Profiles との統合を有効にする`CustomerProfilesIntegrationAccess` ために を追加しました。	2025 年 12 月 15 日
`AWSEntityResolutionConsoleFullAccess` – 既存のポリシーの更新	一致するワークフローでプロバイダーサービスオプションを有効にする `ManageEventBridgeRules` `ADXReadAccess` と を追加しました。	2023 年 10 月 16 日
AWS Entity Resolution が変更の追跡を開始しました	AWS Entity Resolution は、AWS 管理ポリシーの変更の追跡を開始しました。	2023 年 8 月 18 日

AWS Entity Resolution ID とアクセスのトラブルシューティング

次の情報は、 および IAM の使用時に発生する可能性がある一般的な問題の診断 AWS Entity Resolution と修正に役立ちます。

トピック

でアクションを実行する権限がありません AWS Entity Resolution

にアクションを実行する権限がないと AWS マネジメントコンソール 通知された場合は、管理者に連絡してサポートを依頼する必要があります。管理者とは、ユーザーにユーザー名とパスワードを提供した人です。

以下のエラー例は、mateojackson IAM ユーザーがコンソールを使用して架空の *my-example-widget* リソースに関する詳細情報を表示しようとしているが、架空の entityresolution:*GetWidget* アクセス許可がないという場合に発生します。

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

この場合、Mateo は、entityresolution:*GetWidget* アクションを使用して *my-example-widget* リソースにアクセスできるように、ポリシーの更新を管理者に依頼します。

iam:PassRole を実行する権限がない

iam:PassRole アクションを実行する権限がないというエラーが表示された場合は、ポリシーを更新して AWS Entity Resolutionにロールを渡すことができるようにする必要があります。

一部の AWS のサービス では、新しいサービスロールまたはサービスにリンクされたロールを作成する代わりに、既存のロールをそのサービスに渡すことができます。そのためには、サービスにロールを渡す権限が必要です。

以下の例のエラーは、marymajor という IAM ユーザーがコンソールを使用して AWS Entity Resolutionでアクションを実行しようとする場合に発生します。ただし、このアクションをサービス

が実行するには、サービスロールから付与された権限が必要です。Mary には、ロールをサービスに渡すアクセス許可がありません。

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

この場合、Mary のポリシーを更新してメアリーに `iam:PassRole` アクションの実行を許可する必要があります。

サポートが必要な場合は、 AWS 管理者にお問い合わせください。サインイン資格情報を提供した担当者が管理者です。

自分の 以外のユーザーに自分の AWS Entity Resolution リソース AWS アカウント へのアクセスを許可したい

他のアカウントのユーザーや組織外の人が、リソースにアクセスするために使用できるロールを作成できます。ロールの引き受けを委託するユーザーを指定できます。リソースベースのポリシーまたはアクセスコントロールリスト (ACL) をサポートするサービスの場合、それらのポリシーを使用して、リソースへのアクセスを付与できます。

詳細については、以下を参照してください:

- がこれらの機能 AWS Entity Resolution をサポートしているかどうかを確認するには、「」を参照してくださいが IAM と AWS Entity Resolution 連携する方法。
- 所有 AWS アカウント している のリソースへのアクセスを提供する方法については、IAM ユーザーガイドの「所有 AWS アカウント している別の の IAM ユーザーへのアクセスを提供する」を参照してください。
- リソースへのアクセスをサードパーティーに提供する方法については AWS アカウント、IAM ユーザーガイドの「サードパーティー AWS アカウント が所有する へのアクセスを提供する」を参照してください。
- ID フェデレーションを介してアクセスを提供する方法については、IAM ユーザーガイド の 外部で認証されたユーザー (ID フェデレーション) へのアクセスの許可 を参照してください。
- クロスアカウントアクセスにおけるロールとリソースベースのポリシーの使用方法の違いについては、「IAM ユーザーガイド」の「IAM でのクロスアカウントのリソースへのアクセス」を参照してください。

のコンプライアンス検証 AWS Entity Resolution

AWS のサービス が特定のコンプライアンスプログラムの範囲内にあるかどうかを確認するには、AWS のサービス「コンプライアンスプログラムによる対象範囲内のコンプライアンス範囲内の」を参照し、関心のあるコンプライアンスプログラムを選択します。一般的な情報については、AWS「コンプライアンスプログラム」を参照してください。

を使用して、サードパーティーの監査レポートをダウンロードできます AWS Artifact。詳細については、「Downloading Reports in AWS Artifact」を参照してください。

を使用する際のお客様のコンプライアンス責任 AWS のサービス は、お客様のデータの機密性、貴社のコンプライアンス目的、適用される法律および規制によって決まります。を使用する際のコンプライアンス責任の詳細については AWS のサービス、AWS「 セキュリティドキュメント」を参照してください。

AWS Entity Resolution コンプライアンスのベストプラクティス

このセクションでは、 を使用する際のコンプライアンスのベストプラクティスと推奨事項について説明します AWS Entity Resolution。

決済カード業界のデータセキュリティ基準 (PCI DSS)

AWS Entity Resolution は、マーチャントまたはサービスプロバイダーによるクレジットカードデータの処理、保存、および送信をサポートし、Payment Card Industry (PCI) Data Security Standard (DSS) に準拠していることが確認されています。PCI コンプライアンスパッケージのコピーをリクエストする方法など、 AWS PCI DSS の詳細については、「PCI DSS レベル 1」を参照してください。

System and Organization Controls (SOC)

AWS Entity Resolution は、SOC 1、SOC 2、SOC 3 など、System and Organization Controls (SOC) の対策に準拠しています。SOC レポートは、 が主要なコンプライアンスコントロールと目的 AWS を達成する方法を示す、独立したサードパーティーの検査レポートです。これらの監査によって、お客様のデータや企業データのセキュリティ、機密保持、アベイラビリティーに影響を及ぼす可能性のあるリスクから守るために、適切な安全策と手順を講じます。これらのサードパーティー監査の結果は AWS SOC Compliance ウェブサイトで入手できます。ここでは、公開されたレポートを表示して、 AWS 運用とコンプライアンスをサポートするコントロールに関する詳細情報を取得できます。

の耐障害性 AWS Entity Resolution

AWS グローバルインフラストラクチャは、 AWS リージョン およびアベイラビリティーゾーンを中心に構築されています。 は、低レイテンシー、高スループット、および高度に冗長なネットワークで接続された、物理的に分離および分離された複数のアベイラビリティーゾーン AWS リージョン を提供します。アベイラビリティーゾーンでは、ゾーン間で中断することなく自動的にフェイルオーバーするアプリケーションとデータベースを設計および運用することができます。アベイラビリティーゾーンは、従来の単一または複数のデータセンターインフラストラクチャよりも可用性が高く、フォールトトレラントで、スケーラブルです。

AWS リージョン およびアベイラビリティーゾーンの詳細については、AWS 「 グローバルインフラストラクチャ」を参照してください。

グローバル AWS インフラストラクチャに加えて、 AWS Entity Resolution には、データの耐障害性とバックアップのニーズをサポートするのに役立ついくつかの機能が用意されています。

モニタリング AWS Entity Resolution

モニタリングは、 およびその他の AWS Entity Resolution AWS ソリューションの信頼性、可用性、パフォーマンスを維持する上で重要な部分です。 AWS には、監視 AWS Entity Resolution、問題発生時の報告、必要に応じて自動アクションを実行するための以下のモニタリングツールが用意されています。

- AWS CloudTrail は、 によって、または に代わって行われた API コールおよび関連イベントをキャプチャ AWS アカウント し、指定した Amazon S3 バケットにログファイルを配信します。呼び出し元のユーザーとアカウント AWS、呼び出し元の IP、呼び出しの発生日時を確認できます。詳細については、「AWS CloudTrail ユーザーガイド」を参照してください。
- Amazon CloudWatch Logs を使用すると、Amazon EC2 インスタンス、CloudTrail、およびその他のソースからログを確認、保存、アクセスできます。CloudWatch Logs は、ログファイル内の情報をチェックし、特定のしきい値が満たされたときに通知できます。高い耐久性を備えたストレージにログデータをアーカイブすることもできます。詳細については、「Amazon CloudWatch Logs ユーザーガイド」を参照してください。

トピック

を使用した AWS Entity Resolution API コールのログ記録 AWS CloudTrail

AWS Entity Resolution は、ユーザー AWS CloudTrail、ロール、または のサービスによって実行されたアクションを記録する AWS サービスである と統合されています AWS Entity Resolution。CloudTrail は、 AWS Entity Resolution のすべての API コールをイベントとしてキャプチャします。キャプチャされた呼び出しには、 AWS Entity Resolution コンソールからの呼び出しと AWS Entity Resolution API オペレーションへのコード呼び出しが含まれます。証跡を作成する場合は、イベントを含む Amazon S3 バケットへの CloudTrail イベントの継続的な配信を有効にすることができます AWS Entity Resolution。証跡を設定しない場合でも、CloudTrail コンソールの [イベント履歴] で最新のイベントを表示できます。CloudTrail で収集された情報を使用して、リクエストの実行元の IP アドレス AWS Entity Resolution、リクエストの実行者、リクエストの実行日時などの詳細を確認できます。

CloudTrail の詳細については、「AWS CloudTrail ユーザーガイド」を参照してください。

AWS Entity Resolution CloudTrail の情報

CloudTrail は、アカウントの作成 AWS アカウント 時に で有効になります。でアクティビティが発生すると AWS Entity Resolution、そのアクティビティはイベント履歴の他の AWS サービスイベントとともに CloudTrail イベントに記録されます。最近のイベントは、 AWS アカウントで表示、検索、ダウンロードできます。詳細については、 CloudTrail イベント履歴でのイベントの表示を参照してください。

のイベントなど AWS アカウント、 のイベントの継続的な記録については AWS Entity Resolution、証跡を作成します。証跡により、CloudTrail はログファイルを Amazon S3 バケットに配信できます。デフォルトでは、コンソールで証跡を作成するときに、証跡がすべての AWS リージョンに適用されます。証跡は、 AWS パーティション内のすべてのリージョンからのイベントをログに記録し、指定した Amazon S3 バケットにログファイルを配信します。さらに、CloudTrail ログで収集したイベントデータをより詳細に分析し、それに基づいて対応するため、他の AWS サービスを構成できます。詳細については、次を参照してください:

- 追跡を作成するための概要
- CloudTrail がサポートされているサービスと統合
- 「CloudTrail の Amazon SNS 通知の設定」
- 複数のリージョンから CloudTrail ログファイルを受け取るおよび複数のアカウントから CloudTrail ログファイルを受け取る

すべての AWS Entity Resolution アクションは CloudTrail によってログに記録され、 AWS Entity Resolution API リファレンスに記載されています。

各イベントまたはログエントリには、誰がリクエストを生成したかという情報が含まれます。アイデンティティ情報は、以下を判別するのに役立ちます。

- リクエストがルートまたは AWS Identity and Access Management (IAM) ユーザー認証情報を使用して行われたかどうか。
- リクエストがロールまたはフェデレーションユーザーのテンポラリなセキュリティ認証情報を使用して行われたかどうか。
- リクエストが別の AWS サービスによって行われたかどうか。

詳細については、「CloudTrail userIdentity エレメント」を参照してください。

AWS Entity Resolution ログファイルエントリについて

「トレイル」は、指定した Amazon S3 バケットにイベントをログファイルとして配信するように設定できます。CloudTrail のログファイルは、単一か複数のログエントリを含みます。イベントは、任意の出典からの単一のリクエストを表し、リクエストされたアクション、アクションの日時、リクエストパラメータなどに関する情報が含まれます。CloudTrail ログファイルは、パブリック API コールの順序付けられたスタックトレースではないため、特定の順序では表示されません。

Amazon CloudWatch Logs を使用したワークフローのモニタリングとログ記録

AWS Entity Resolution は、マッチングワークフローと ID マッピングワークフローのチェックと分析に役立つ包括的なログ記録機能を提供します。Amazon CloudWatch Logs との統合により、イベントタイプ、タイムスタンプ、処理統計、エラー数など、ワークフロー実行に関する詳細情報をキャプチャできます。これらのログを CloudWatch Logs、Amazon S3、または Amazon Data Firehose の送信先に配信することを選択できます。これらのログを分析することで、サービスのパフォーマンスの評価、問題のトラブルシューティング、顧客ベースのインサイトの取得、使用状況と請求の理解を深める AWS Entity Resolution ことができます。ログ記録はデフォルトで無効になっていますが、コンソールまたは API を使用して、新規ワークフローと既存のワークフローの両方で有効にできます。

標準の Amazon CloudWatch 販売料金は、ログの取り込み、ストレージ、分析に関連するコストなど、 AWS Entity Resolution ワークフローのログ記録を有効にする場合に適用されます。料金の詳細については、CloudWatch の料金ページを参照してください。

トピック

- ログ配信の設定
- ログ記録の無効化 (コンソール)
- ログの読み取り

ログ配信の設定

このセクションでは、 AWS Entity Resolution ログ記録を使用するために必要なアクセス許可と、 コンソールと APIs を使用してログ配信を有効にする方法について説明します。

トピック

- アクセス許可
- 新しいワークフローのログ記録の有効化 (コンソール)
- 新しいワークフローのログ記録の有効化 (API)
- 既存のワークフローのログ記録の有効化 (コンソール)

アクセス許可

AWS Entity Resolution は CloudWatch 提供のログを使用してワークフローログを配信します。ワークフローログを配信するには、指定したログ記録先へのアクセス許可が必要です。

各ログ記録先に必要なアクセス許可を確認するには、Amazon CloudWatch Logs ユーザーガイド」の次の AWS サービスから選択します。

- Amazon CloudWatch Logs
- Amazon Simple Storage Service (Amazon S3)
- Amazon Data Firehose

でログ記録設定を作成、表示、または変更するには AWS Entity Resolution、必要なアクセス許可が必要です。IAM ロールには、 AWS Entity Resolution コンソールでワークフローログを管理するための以下の最小限のアクセス許可が含まれている必要があります。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowLogDeliveryActionsConsoleCWL",
            "Effect": "Allow",
            "Action": [
                "logs:DescribeLogGroups"
            ],
            "Resource": [
                "arn:aws:logs:us-east-1:111122223333:log-group:*"
            ]
        },
        {
            "Sid": "AllowLogDeliveryActionsConsoleS3",
```

```
            "Effect": "Allow",
            "Action": [
                "s3:ListAllMyBuckets",
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:s3:::*"
            ]
        },
        {
            "Sid": "AllowLogDeliveryActionsConsoleFH",
            "Effect": "Allow",
            "Action": [
                "firehose:ListDeliveryStreams",
                "firehose:DescribeDeliveryStream"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

ワークフローログ記録を管理するアクセス許可の詳細については、Amazon CloudWatch Logs ユーザーガイド」の AWS 「サービスからのログ記録を有効にする」を参照してください。

新しいワークフローのログ記録の有効化 (コンソール)

ログ記録先へのアクセス許可を設定したら、 コンソール AWS Entity Resolution を使用して で新しいワークフローのログ記録を有効にできます。

新しいワークフローのログ記録を有効にするには (コンソール)

1. https://console.aws.amazon.com/entityresolution/home で AWS Entity Resolution コンソールを開きます。
2. ワークフローで、一致するワークフローまたは ID マッピングワークフローを選択します。
3. ステップに従って、次のいずれかのワークフローを作成します。

   - ルールベースのマッチングワークフロー
   - 機械学習ベースのマッチングワークフロー

- プロバイダーのサービスベースのマッチングワークフロー
- 1 つのアカウントの ID マッピングワークフロー
- 2 つのアカウントにわたる ID マッピングワークフロー

4. ステップ 1 ー致ワークフローの詳細を指定するには、ログ配信 – EntityResolution ワークフローログで、追加を選択します。

   - 次のいずれかのログ記録先を選択します。

     - Amazon CloudWatch Logs へ
     - Amazon S3 へ
     - Amazon Data Firehose へ

   > **Tip**
   >
   > Amazon S3 または Firehose を選択した場合、クロスアカウントまたは現在のアカウントにログを配信できます。
   > クロスアカウント配信を有効にするには、両方に必要なアクセス許可 AWS アカウント が必要です。詳細については、Amazon CloudWatch Logs ユーザーガイド」の「クロスアカウント配信の例」を参照してください。

5. 送信先ロググループの場合、プレフィックスが '/aws/vendedlogs/' のロググループが自動的に作成されます。他のロググループを使用している場合は、ログ配信を設定する前にロググループを使用します。詳細については、「Amazon CloudWatch Logs ユーザーガイド」の「ロググループとログストリームの操作」を参照してください。
6. その他の設定 - オプションで、以下を選択します。

   a. フィールド選択で、各ログレコードに含めるログフィールドを選択します。
   b. (CloudWatch Logs) 出力形式で、ログの出力形式を選択します。
   c. フィールド区切り記号で、各ログフィールドを区切る方法を選択します。
   d. (Amazon S3) Suffix には、データをパーティション化するためのサフィックスパスを指定します。
   e. (Amazon S3) Hive 互換の場合は、Hive 互換 S3 パスを使用する場合は Enable を選択します。

7. 別のログ送信先を作成するには、「追加」を選択し、ステップ 4 ~ 6 を繰り返します。
8. 残りのステップを完了して、ワークフローをセットアップして実行します。

9. ワークフロージョブが完了したら、指定したログ配信先のワークフローログを確認します。

新しいワークフローのログ記録の有効化 (API)

ログ記録先へのアクセス許可を設定したら、Amazon CloudWatch Logs APIs AWS Entity Resolution を使用して で新しいワークフローのログ記録を有効にできます。

新しいワークフローのログ記録を有効にするには (API)

1. AWS Entity Resolution コンソールでワークフローを作成したら、ワークフローの Amazon リソースネーム (ARN) を取得します。

   AWS Entity Resolution コンソールのワークフローページから ARN を見つけるか、`GetMatchingWorkflow` または `GetIdMappingWorkflow` API オペレーションを呼び出します。

   ワークフロー ARN は次の形式に従います。

   ```
   arn:(aws|aws-us-gov|aws-cn):entityresolution:[a-z]{2}-[a-z]{1,10}-[0-9]:[0-9]{12}:(matchingworkflow/[a-zA-Z_0-9-]{1,255})
   ```

   ID マッピング ARN は次の形式に従います。

   ```
   arn:(aws|aws-us-gov|aws-cn):entityresolution:[a-z]{2}-[a-z]{1,10}-[0-9]:[0-9]{12}:(idmappingworkflow/[a-zA-Z_0-9-]{1,255})
   ```

   詳細については、 AWS Entity Resolution API リファレンスのGetMatchingWorkflow」またはGetIdMappingWorkflow」を参照してください。

2. CloudWatch Logs `PutDeliverySource` API オペレーションを使用して、ワークフローログの配信ソースを作成します。

   詳細については、Amazon CloudWatch Logs API リファレンス」の「PutDeliverySource」を参照してください。

   a. を渡します`resourceArn`。

   b. の場合`logType`、収集されるログのタイプは ですWORKFLOW_LOGS。

Example

`PutDeliverySource` API オペレーションの例

```
{
    "logType": "WORKFLOW_LOGS",
    "name": "my-delivery-source",
    "resourceArn": "arn:aws:entityresolution:region:accoungId:matchingworkflow/
XXXWorkflow"
}
```

3. `PutDeliveryDestination` API オペレーションを使用して、ログの保存先を設定します。

CloudWatch Logs、Amazon S3、または Firehose のいずれかを送信先として選択できます。ログの保存先オプションのいずれかの ARN を指定する必要があります。

詳細については、Amazon CloudWatch Logs API リファレンス」のPutDeliveryDestination」を参照してください。

Example

`PutDeliveryDestination` API オペレーションの例

```
{
   "delivery-destination-configuration": {
      "destinationResourceArn": "arn:aws:logs:region:accountId:log-group:my-log-
group"
   },
   "name": "my-delivery-destination",
   "outputFormat": "json",
   }
}
```

> **Note**
>
> ログをクロスアカウントで配信する場合は、PutDeliveryDestinationPolicy API を使用して、送信先アカウントに (IAM) ポリシーを割り当てる AWS Identity and Access Management 必要があります。IAM ポリシーは、あるアカウントから別のアカウントへの配信を許可します。

4. `CreateDelivery` API オペレーションを使用して、配信ソースを前のステップで作成した送信先にリンクします。この API オペレーションは、配信ソースを最終配信先と関連付けます。

   詳細については、Amazon CloudWatch Logs API リファレンス」の「PutDeliveryDestination」を参照してください。

   Example

   `CreateDelivery` API オペレーションの例

   ```
   {
      "delivery-destination-arn": "arn:aws:logs:region:accountId:log-group:my-log-group",
      "delivery-source-name": "my-delivery-source",
      "tags": {
         "string" : "string"
      }
   }
   ```

5. ワークフローを実行します。
6. ワークフロージョブが完了したら、指定したログ配信先のワークフローログを確認します。

既存のワークフローのログ記録の有効化 (コンソール)

ログ記録先へのアクセス許可を設定したら、 コンソールのログ配信タブ AWS Entity Resolution を使用して、 で既存のワークフローのログ記録を有効にできます。

ログ配信タブを使用して既存のワークフローのログ記録を有効にするには (コンソール)

1. https://console.aws.amazon.com/entityresolution/home で AWS Entity Resolution コンソールを開きます。
2. ワークフローで、一致するワークフローまたは ID マッピングワークフローを選択し、既存のワークフローを選択します。
3. 「ログ配信」タブの「ログ配信」で、「追加」を選択し、次のいずれかのログ記録先を選択します。

   - Amazon CloudWatch Logs へ
   - Amazon S3 へ
     - クロスアカウント

    - 現行のアカウント
- Amazon Data Firehose へ
    - クロスアカウント
    - 現行のアカウント

> **Tip**
>
> Amazon S3 または Firehose を選択した場合、クロスアカウントまたは現在のアカウントにログを配信できます。
> クロスアカウント配信を有効にするには、両方に必要なアクセス許可 AWS アカウントが必要です。詳細については、Amazon CloudWatch Logs ユーザーガイド」の「クロスアカウント配信の例」を参照してください。

4. モーダルで、選択したログ配信のタイプに応じて、次の操作を行います。

    a. ログタイプを表示する: WORKFLOW_LOGS。

    ログタイプは変更できません。

    b. (CloudWatch Logs) 送信先ロググループの場合、'/aws/vendedlogs/' というプレフィックスが付いたロググループが自動的に作成されます。他のロググループを使用している場合は、ログ配信を設定する前にロググループを使用します。詳細については、「Amazon CloudWatch Logs ユーザーガイド」の「ロググループとログストリームの操作」を参照してください。

    (現在のアカウントの Amazon S3) 送信先 S3 バケットの場合は、バケットを選択するか、ARN を入力します。

    (Amazon S3 クロスアカウント) 配信先 ARN に配信先 ARN を入力します。

    (現在のアカウントのファイアウォール) 送信先配信ストリームには、別のアカウントで作成された配信先リソースの ARN を入力します。

    (Firehose クロスアカウント) 配信先 ARN に配信先 ARN を入力します。

5. その他の設定 - オプションで、以下を選択します。

    a. フィールド選択で、各ログレコードに含めるログフィールドを選択します。

    b. (CloudWatch Logs) 出力形式で、ログの出力形式を選択します。

c. フィールド区切り記号で、各ログフィールドを区切る方法を選択します。

d. (Amazon S3) Suffix には、データをパーティション化するためのサフィックスパスを指定します。

e. (Amazon S3) Hive 互換の場合は、Hive 互換 S3 パスを使用する場合は Enable を選択します。

6. [Add] (追加) を選択します。

7. ワークフローページで、実行 を選択します。

8. ワークフロージョブが完了したら、指定したログ配信先のワークフローログを確認します。

ログ記録の無効化 (コンソール)

ワークフローのログ記録は、コンソールで AWS Entity Resolution いつでも無効にできます。

ワークフローログ記録を無効にするには (コンソール)

1. https://console.aws.amazon.com/entityresolution/home で AWS Entity Resolution コンソールを開きます。

2. ワークフローで、一致するワークフローまたは ID マッピングワークフローを選択し、ワークフローを選択します。

3. 「ログ配信」タブの「ログ配信」で、送信先を選択し、「削除」を選択します。

4. 変更を確認し、次のステップに移動して変更を保存します。

ログの読み取り

Amazon CloudWatch Logs を読み取ると、効率的な AWS Entity Resolution ワークフローを維持できます。ログは、処理されたレコードの数や発生したエラーなどの重要なメトリクスなど、ワークフローの実行を詳細に可視化するため、データ処理がスムーズに実行されていることを確認できます。さらに、ログはタイムスタンプやイベントタイプを通じてワークフローの進行状況をリアルタイムで追跡できるため、データ処理パイプラインのボトルネックや問題をすばやく特定できます。包括的なエラー追跡とレコード数情報は、正常に処理されたレコードの数と、未処理のレコードがあるかどうかを正確に示すことで、データの品質と完全性を維持するのに役立ちます。

CloudWatch Logs を送信先として使用している場合は、CloudWatch Logs Insights を使用してワークフローログを読み取ることができます。CloudWatch Logs の一般料金が適用されます。詳細につ

いては、「Amazon CloudWatch Logs ユーザーガイド」の「CloudWatch Logs Insights でログデータを分析する」を参照してください。

> **Note**
>
> ワークフローログが送信先に表示されるまでに数分かかる場合があります。ログが表示されない場合は、数分待ってからページを更新します。

ワークフローログは、フォーマットされた一連のログレコードで構成され、各ログレコードは 1 つのワークフローを表します。ログ内のフィールドの順序は変わることがあります。

```
{
  "resource_arn": "arn:aws:ses:us-east-1:1234567890:mailmanager-ingress-point/inp-
xxxxx",
  "event_type": "JOB_START",
  "event_timestamp": 1728562395042,
  "job_id": "b01eea4678d4423a4b43eeada003f6",
  "workflow_name": "TestWorkflow",
  "workflow_start_time": "2025-03-11 10:19:56",
  "data_procesing_progression": "Matching Job Starts ...",
  "total_records_processed": 1500,
  "total_records_unprocessed": 0,
  "incremental_records_processed": 0,
  "error_message": "sample error that caused workflow failure"
}
```

次のリストで、ログレコードのフィールドを順番に従い説明します。

`resource_arn`

ワークフローで使用されているリソースを一意に識別する Amazon AWS リソースネーム (ARN)。

`event_type`

ワークフローの実行中に発生したイベントのタイプ。 AWS Entity Resolution 現在、 は以下をサポートしています。

`JOB_START`

`DATA_PROCESSING_STEP_START`

DATA_PROCESSING_STEP_END

JOB_SUCCESS

JOB_FAILURE

event_timestamp

ワークフロー中にイベントが発生した時刻を示す Unix タイムスタンプ。

job_id

特定のワークフロージョブ実行に割り当てられた一意の識別子。

workflow_name

実行中のワークフローに付けられた名前。

workflow_start_time

ワークフロー実行が開始された日時。

data_procesing_progression

データ処理ワークフローの現在のステージの説明。例: "Matching Job Starts"、"Loading Step Starts"、"ID_Mapping Job Ends Successfully"。

total_records_processed

ワークフロー中に正常に処理されたレコードの合計数。

total_records_unprocessed

ワークフローの実行中に処理されなかったレコードの数。

incremental_records_processed

増分ワークフロー更新で処理された新しいレコードの数。

error_message

ワークフロー障害の根本原因。

を使用して AWS エンティティ解決リソースを作成する AWS CloudFormation

AWS Entity Resolution は と統合されています。これは AWS CloudFormation、 AWS リソースとインフラストラクチャの作成と管理に費やす時間を短縮できるように、リソースのモデル化とセットアップに役立つサービスです。必要なすべての AWS リソース (AWS::EntityResolution::MatchingWorkflow、AWS::EntityResolution::SchemaMapping、AWS::EntityResoluti など) を記述するテンプレートを作成し、それらのリソースを CloudFormation プロビジョニングして設定します。

を使用すると CloudFormation、テンプレートを再利用して AWS エンティティ解決リソースを一貫して繰り返しセットアップできます。リソースを一度記述し、複数の AWS アカウント およびリージョンで同じリソースを何度もプロビジョニングします。

AWS エンティティ解決と CloudFormation テンプレート

AWS エンティティ解決および関連サービスのリソースをプロビジョニングおよび設定するには、CloudFormation テンプレートを理解する必要があります。テンプレートは、JSON や YAML でフォーマットされたテキストファイルです。これらのテンプレートは、 CloudFormation スタックにプロビジョニングするリソースを記述します。JSON または YAML に慣れていない場合は、CloudFormation デザイナー を使用して CloudFormation テンプレートの使用を開始できます。詳細については、「AWS CloudFormation ユーザーガイド」の「CloudFormation Designer とは」を参照してください。

AWS エンティティ解決では、AWS::EntityResolution::MatchingWorkflow、AWS::EntityResolution::SchemaMapping、AWS::EntityRes および AWS::EntityResolution::PolicyStatement の作成がサポートされています CloudFormation。AWS::EntityResolution::MatchingWorkflow、AWS::EntityResolution::SchemaMapping、AW および AWS::EntityResolution::PolicyStatement の JSON テンプレートと YAML テンプレートの例を含む詳細については、AWS CloudFormation 「 ユーザーガイド」の「AWS エンティティ解決リソースタイプのリファレンス」を参照してください。

次のテンプレートを使用できます。

- マッチングワークフロー

  実行するデータ処理ジョブの設定を保存する `MatchingWorkflow` オブジェクトを作成します。

詳細については、以下の各トピックを参照してください。

「CloudFormation ユーザーガイド」の「AWS::EntityResolution::MatchingWorkflow」

「CreateMatchingWorkflow API リファレンス」の「AWS Entity Resolution 」

- スキーママッピング

  入力カスタマーレコードテーブルのスキーマを定義するスキーママッピングを作成します。

  詳細については、以下の各トピックを参照してください。

  「CloudFormation ユーザーガイド」の「AWS::EntityResolution::SchemaMapping」

  「CreateSchemaMapping API リファレンス」の「AWS Entity Resolution 」

- ID マッピングワークフロー

  実行するデータ処理ジョブの設定を保存する `IdMappingWorkflow` オブジェクトを作成します。

  詳細については、以下の各トピックを参照してください。

  「CloudFormation ユーザーガイド」の「AWS::EntityResolution::IdMappingWorkflow」

  「CreateIdMappingWorkflow API リファレンス」の「AWS Entity Resolution 」

- ID 名前空間

  データセットとその使用方法を説明するメタデータを保存する `IdNamespace` オブジェクトを作成します。

  詳細については、以下の各トピックを参照してください。

  「CloudFormation ユーザーガイド」の「AWS::EntityResolution::IdNamespace」

  「CreateIdNamespace API リファレンス」の「AWS Entity Resolution 」

- PolicyStatement

  `PolicyStatement` オブジェクトを作成します。

  詳細については、以下の各トピックを参照してください。

  「CloudFormation ユーザーガイド」の「AWS::EntityResolution::PolicyStatement」

「AddPolicyStatement API リファレンス」の「AWS Entity Resolution 」

の詳細 CloudFormation

詳細については CloudFormation、次のリソースを参照してください。

- AWS CloudFormation
- AWS CloudFormation ユーザーガイド
- CloudFormation API リファレンス
- AWS CloudFormation コマンドラインインターフェイスユーザーガイド

のクォータ AWS Entity Resolution

AWS アカウント には、各 の制限と呼ばれるデフォルトのクォータがあります AWS のサービス。特に明記していない限り、クォータはリージョン固有です。一部のクォータの引き上げをリクエストすることはできますが、他のクォータは引き上げることができません。

のクォータを表示するには AWS Entity Resolution、Service Quotas コンソールを開きます。ナビゲーションペインで、[AWS のサービス] を選択し、[AWS Entity Resolution] を選択します。

クォータの引き上げをリクエストするには、Service Quotas ユーザーガイド の「クォータ引き上げリクエスト」を参照してください。Service Quotas でクォータがまだ利用できない場合は、制限の引き上げ フォームを使用します。

AWS アカウント には、次のクォータが関連しています AWS Entity Resolution。

名前	デフォルト	引き上げ可能	説明
同時 ID マッピングジョブ	サポートされている各リージョン: 1	[いいえ]	現在の AWS リージョンで同時に処理できる ID マッピングワークフローの最大数。
同時マッチングジョブ	サポートされている各リージョン: 1	[いいえ]	現在の AWS リージョンで同時に処理できる一致するワークフローの最大数。
同時プロバイダーサービスマッチングジョブ	サポートされている各リージョン: 1	[いいえ]	現在の AWS リージョンで同時に処理できるプロバイダーサービスマッチングワークフローの最大数。

名前	デフォルト	引き上げ可能	説明
ID マッピングワークフロー	サポートされている各リージョン: 10	あり	このアカウントで現在の AWS リージョンに作成できる ID マッピングワークフローの最大数。
ID 名前空間	サポートされている各リージョン: 10	あり	このアカウントで現在の AWS リージョンに作成できる ID 名前空間の最大数。
ワークフローの一致	サポートされている各リージョン: 10	あり	このアカウントで現在の AWS リージョンに作成できる一致するワークフローの最大数。
GenerateMatchId API リクエストのレート	サポートされている各リージョン: 10	あり	1 秒あたりの GenerateMatchId API リクエストの最大数
GetMatchId API リクエストのレート	サポートされている各リージョン: 50	可能	1 秒あたりの GetMatchId API リクエストの最大数。
機械学習ベースのマッチングワークフローあたりのレコード	サポートされている各リージョン: 150,000,000	あり	af-south-1、ap-northeast-2、eu-west-2 の AWS リージョンで、このアカウントの機械学習ベースのマッチングワークフローで処理できるレコードの最大数。

名前	デフォルト	引き上げ可能	説明
機械学習ベースのマッチングワークフローあたりのレコード	サポートされている各リージョン: 600,000,000	あり	ap-northeast-1、ap-southeast-1、ap-southeast-2、ca-central-1、eu-central-1、eu-west-1、us-east-1、us-east-2、us-west-2 の AWS リージョンで、このアカウントの機械学習ベースのマッチングワークフローで処理できるレコードの最大数。
プロバイダー ID マッピングワークフローあたりのレコード数	サポートされている各リージョン: 150,000,000	あり	af-south-1、ap-northeast-2、eu-west-2 の AWS リージョンで、このアカウントのプロバイダー ID マッピングのために処理できるレコードの最大数。

名前	デフォルト	引き上げ可能	説明
プロバイダー ID マッピングワークフローあたりのレコード数	サポートされている各リージョン: 250,000,000	あり	ap-northeast-1、ap-southeast-1、ap-southeast-2、ca-central-1、eu-central-1、eu-west-1、us-east-1、us-east-2、us-west-2 の AWS リージョンで、このアカウントのプロバイダー ID マッピング用に処理できるレコードの最大数。
プロバイダーのサービスベースのマッチングワークフローあたりのレコード	サポートされている各リージョン: 100,000,000	あり	現在の AWS リージョンで、このアカウントでプロバイダーのサービスベースのマッチングワークフローによって処理できるレコードの最大数。
ルールベースの ID マッピングワークフローあたりのレコード	サポートされている各リージョン: 1,000,000,000	あり	ap-northeast-1、ap-southeast-1、ap-southeast-2、ca-central-1、eu-central-1、eu-west-1、us-east-1、us-east-2、us-west-2 の AWS リージョンで、このアカウントでルールベースの ID マッピングのために処理できるレコードの最大数。

名前	デフォルト	引き上げ可能	説明
ルールベースの ID マッピングワークフローあたりのレコード	サポートされている各リージョン: 150,000,000	あり	af-south-1、ap-northeast-2、eu-west-2 の AWS リージョンで、このアカウントでルールベースの ID マッピングのために処理できるレコードの最大数。
ルールベースのマッチングワークフローあたりのレコード	サポートされている各リージョン: 100,000,000	あり	現在の AWS リージョンで、このアカウントでルールベースのマッチングワークフローで処理できるレコードの最大数。
スキーママッピング	サポートされている各リージョン: 50	可能	このアカウントで現在の AWS リージョンに作成できるスキーママッピングの最大数。

API スロットリングのクォータ

[リソース]	[Rate limit] (レート制限)	説明
`CreateMatchingWorkflow` リクエストのレート	5 TPS	1 秒あたりの `CreateMatchingWorkflow` API コールの最大数。

[リソース]	[Rate limit] (レート制限)	説明
DeleteMatchingWorkflow リクエストのレート	5 TPS	1 秒あたりの DeleteMatchingWorkflow API コールの最大数。
GetMatchingWorkflow リクエストのレート	5 TPS	1 秒あたりの GetMatchingWorkflow API コールの最大数。
ListMatchingWorkflows リクエストのレート	5 TPS	1 秒あたりの ListMatchingWorkflows API コールの最大数。
UpdateMatchingWorkflow リクエストのレート	5 TPS	1 秒あたりの UpdateMatchingWorkflow API コールの最大数。
CreateSchemaMapping リクエストのレート	5 TPS	1 秒あたりの CreateSchemaMapping API コールの最大数。
DeleteSchemaMapping リクエストのレート	5 TPS	1 秒あたりの DeleteSchemaMapping API コールの最大数。
GetSchemaMapping リクエストのレート	5 TPS	1 秒あたりの GetSchemaMapping API コールの最大数。
ListSchemaMappings リクエストのレート	5 TPS	1 秒あたりの ListSchemaMappings API コールの最大数。
UpdateSchemaMapping リクエストのレート	5 TPS	1 秒あたりの UpdateSchemaMapping API コールの最大数。

[リソース]	[Rate limit] (レート制限)	説明
GetPartnerComponent リクエストのレート	5 TPS	1 秒あたりの GetPartnerComponent API コールの最大数。
ListPartnerComponents リクエストのレート	5 TPS	1 秒あたりの ListPartnerComponents API コールの最大数。
TagResource リクエストのレート	5 TPS	1 秒あたりの TagResource API コールの最大数。
UntagResource リクエストのレート	5 TPS	1 秒あたりの UntagResource API コールの最大数。
ListTagsForResource リクエストのレート	5 TPS	1 秒あたりの ListTagsForResource API コールの最大数。
CreateIdMappingWorkflow リクエストのレート	5 TPS	1 秒あたりの CreateIdMappingWorkflow API コールの最大数。
DeleteIdMappingWorkflow リクエストのレート	5 TPS	1 秒あたりの DeleteIdMappingWorkflow API コールの最大数。
GetIdMappingWorkflow リクエストのレート	5 TPS	1 秒あたりの GetIdMappingWorkflow API コールの最大数。
ListIdMappingWorkflow リクエストのレート	5 TPS	1 秒あたりの ListIdMappingWorkflow API コールの最大数。
UpdateIdMappingWorkflow リクエストのレート	5 TPS	1 秒あたりの UpdateIdMappingWorkflow API コールの最大数。

[リソース]	[Rate limit] (レート制限)	説明
ListProviderServices リクエストのレート	5 TPS	1 秒あたりの ListProviderServices API コールの最大数。
GetProviderService リクエストのレート	5 TPS	1 秒あたりの GetProviderService API コールの最大数。
CreateIdNamespace リクエストのレート	5 TPS	1 秒あたりの CreateIdNamespace API コールの最大数。
DeleteIdNamespace リクエストのレート	5 TPS	1 秒あたりの DeleteIdNamespace API コールの最大数。
GetIdNamespace リクエストのレート	5 TPS	1 秒あたりの GetIdNamespace API コールの最大数。
ListIdNamespaces リクエストのレート	5 TPS	1 秒あたりの ListIdNamespaces API コールの最大数。
UpdateIdNamespace リクエストのレート	5 TPS	1 秒あたりの UpdateIdNamespace API コールの最大数。
AddPolicyStatement リクエストのレート	5 TPS	1 秒あたりの AddPolicyStatement API コールの最大数。
DeletePolicyStatement リクエストのレート	5 TPS	1 秒あたりの DeletePolicyStatement API コールの最大数。

[リソース]	[Rate limit] (レート制限)	説明
GetPolicy リクエストのレート	5 TPS	1 秒あたりの GetPolicy API コールの最大数。
PutPolicy リクエストのレート	5 TPS	1 秒あたりの PutPolicy API コールの最大数。
GetMatchingJob リクエストのレート	10 TPS	1 秒あたりの GetMatchingJob API コールの最大数。
ListMatchingJobs リクエストのレート	5 TPS	1 秒あたりの ListMatchingJobs API コールの最大数。
StartMatchingJob リクエストのレート	5 TPS	1 秒あたりの StartMatchingJob API コールの最大数。
GetMatchId リクエストのレート	50 TPS	1 秒あたりの GetMatchId API コールの最大数。
GetIdMappingJob リクエストのレート	10 TPS	1 秒あたりの GetIdMappingJob API コールの最大数。
ListIdMappingJobs リクエストのレート	5 TPS	1 秒あたりの ListIdMappingJobs API コールの最大数。
StartIdMappingJob リクエストのレート	5 TPS	1 秒あたりの StartIdMappingJob API コールの最大数。
BatchDeleteUniqueId リクエストのレート	5 TPS	1 秒あたりの BatchDeleteUniqueId API コールの最大数。

AWS Entity Resolution ユーザーガイドのドキュメント履歴

次の表に、 のドキュメントリリースを示します AWS Entity Resolution。

このドキュメントの更新に関する通知については、RSS フィードにサブスクライブできます。RSS の更新をサブスクリプションするには、使用しているブラウザで RSS プラグインを有効にする必要があります。

変更	説明	日付
既存のポリシーの更新	次の新しい権限が`AWSEntityResolutionConsoleFullAccess` マネージドポリシーに追加されました: `CustomerProfilesIntegrationAccess`	2025 年 12 月 15 日
Amazon Connect Customer Profiles のサポート	ルールベースまたは機械学習ベースのマッチングワークフローを使用する場合に、重複排除された顧客レコードを Amazon Connect Customer Profiles に直接エクスポートする機能を追加しました。	2025 年 12 月 15 日
FIPS のサポート	AWS Entity Resolution は、を通じて連邦情報処理規格 (FIPS) 140-2 準拠のエンドポイントをサポートするようになりました AWS PrivateLink。	2025 年 10 月 21 日
ID マッピングワークフロー – 更新	お客様は、ルールベースの ID マッピングワークフローで増分処理を使用して、大規模なデータセットをより効率的に処理できるようになりまし	2025 年 9 月 22 日

	た。お客様は、ID マッピングワークフローからレコードを削除して、データ管理規制に準拠することもできます。	
クロスリージョンのサポート	お客様は、ID 名前空間、マッピングワークフロー、または ID マッピングワークフローへの入力 AWS リージョン として別の のデータを使用できるようになりました。	2025 年 9 月 8 日
拡張ルール条件と増分削除のサポート	お客様は、ブール演算子と ExactManyToMany などの新しいマッチング関数でルール条件を使用できるようになりました。これにより、完全一致とあいまい一致の組み合わせによるより正確なマッチング基準が可能になります。さらに、お客様は Amazon S3 ファイルを使用して、高度なマッチングワークフローでレコードを段階的に削除できます。	2025 年 7 月 30 日
一致 ID 処理の明確化	一致 ID の変更または生成、および一致 ID の検索オプションでは、一致するワークフローで自動処理の頻度が必要であることを明確にしました。	2025 年 7 月 17 日
新しい一致 ID を生成する	ルールベースの一致ワークフローを使用するときに、既存の一致 ID を検索して変更したり、新しい一致 ID を生成したりできます。	2025 年 6 月 2 日

プロバイダーのサービスベースのマッチングワークフロー – 更新	TransUnion プロバイダーのサービスベースのマッチングワークフローを使用するときに、IPV4, IPV6、MAID などのデジタル識別子を使用できるようになりました。	2025 年 4 月 21 日
Amazon CloudWatch Logs	AWS Entity Resolution は CloudWatch Logs 統合をサポートするようになりました。これにより、CloudWatch Logs、Amazon S3、または Amazon Data Firehose の送信先に配信できるジョブ実行メトリクス、タイミング、処理統計をキャプチャする詳細なワークフローログ記録を有効にできます。	2025 年 4 月 14 日
ID マッピングワークフロー – 更新	ID マッピングワークフローを使用するときに AWS Glue パーティショニングを設定できるようになりました。	2025 年 3 月 25 日
クォータ – 更新	ドキュメントのみの更新。ルールベースのマッチングワークフローは最大 100Mレコードを処理でき、機械学習ベースのマッチングワークフローは最大 250Mレコードを処理できます。制限の引き上げが必要なお客様は、 サービスチームにお問い合わせください。	2025 年 2 月 7 日

スキーママッピング – 更新	フルネーム、フルアドレス、およびフルフォン属性タイプで正規化がサポートされていることを明確にするためのドキュメントのみの更新。	2025 年 1 月 17 日
プロバイダーの統合	ドキュメントのみの更新。お客様は、プロバイダーサービスとして と統合する方法を学習できます AWS Entity Resolution。	2024 年 8 月 8 日
ID マッピングワークフロー – 更新	お客様は、一致するルールを使用して、ID マッピングワークフローでファーストパーティデータを翻訳できるようになりました。	2024 年 7 月 23 日
一致するワークフロー – 更新	お客様は、データ管理規制に準拠するために、ルールベースまたは ML ベースのマッチングワークフローからレコードを削除できるようになりました。	2024 年 4 月 8 日
ID マッピングワークフロー – 更新	お客様は、複数の で ID マッピングワークフローを使用できるようになりました AWS アカウント。	2024 年 4 月 2 日

CloudFormation リソース - 新規および更新されたリソース	AWS Entity Resolution は、次のリソースを追加しました。 `AWS::EntityResolution::IdNamespace` `AWS::EntityResolution::PolicyStatement` および は、次のリソースを更新しました`AWS::EntityResolution::IdMappingWorkflow` 。	2024 年 4 月 2 日
一致 ID の検索	お客様は、処理されたルールベースのワークフローに対応する一致 ID と関連するルールを見つけることができるようになりました。	2024 年 3 月 25 日
一致するワークフロー – 更新	AWS Entity Resolution は、LiveRamp プロバイダーのサービスベースのマッチングワークフローで PII ベースの RAMPID 割り当てをサポートするようになりました。	2024 年 2 月 12 日
AWS PrivateLink	AWS Entity Resolution は、 で追加のデータセキュリティをサポートするようになりました。 AWS PrivateLink これにより、お客様は でホストされているサービスにプライベートにアクセスできます AWS。	2023 年 10 月 20 日

CloudFormation リソース – 新規および更新されたリソース	AWS Entity Resolution では、次のリソースが追加されました。 AWS::EntityResolution:IdMappingWorkflow および のリソースが更新されAWS::EntityResolution::MatchingWorkflow ましたAWS::EntityResolution::Schemamapping 。	2023 年 10 月 19 日
既存のポリシーの更新	AWSEntityResolutionConsoleFullAccess 管理ポリシーに次の新しいアクセス許可が追加されました: ADXReadAccess および ManageEventBridgeRules 。	2023 年 10 月 16 日
スキーママッピング – 更新	お客様は、既存のデータスキーマを編集および更新できるようになりました。	2023 年 10 月 16 日
一致するワークフロー – 更新	お客様は、データの照合とリンクに役立つ任意のデータプロバイダーサービスを選択できるようになりました。	2023 年 10 月 16 日
ID マッピングワークフロー	お客様はこの新しいワークフローを使用して、ID マッピングの詳細を指定し、目的の ID マッピング方法を選択し、データ入力フィールドと出力フィールドを指定できます。	2023 年 10 月 16 日

CloudFormation 統合	AWS Entity Resolution が と統合されるようになりましたCloudFormation。	2023 年 8 月 24 日
AWS マネージドポリシーの更新 - 新しいポリシー	AWS Entity Resolution に 2 つの新しい管理ポリシーが追加されました。	2023 年 8 月 18 日
初回リリース	AWS Entity Resolution ユーザーガイドの初回リリース	2023 年 7 月 26 日

AWS Entity Resolution 用語集

Amazon リソースネーム (ARN)

AWS リソースの一意の識別子。ARNs は、 AWS Entity Resolution ポリシー、Amazon Relational Database Service (Amazon RDS) タグ AWS Entity Resolution、API コールなど、すべての でリソースを明確に指定する必要がある場合に必要です。

属性タイプ

入力フィールドの属性のタイプ。スキーママッピングを作成するときは、名前、住所、電話番号、Eメールアドレスなどの事前設定された値のリストから属性タイプを選択します。属性タイプは、表示するデータ AWS Entity Resolution の種類を指定し、適切に分類および正規化できるようにします。

自動処理

データ入力が変更されたときに自動的に実行できるようにする、一致するワークフロージョブの処理頻度オプション。

このオプションは、ルールベースのマッチングでのみ使用できます。

デフォルトでは、一致するワークフロージョブの処理頻度は手動に設定されます。これにより、オンデマンドで実行できます。データ入力が変更されると、一致するワークフロージョブを自動的に実行するように自動処理を設定できます。これにより、一致するワークフロー出力up-to-date状態になります。

AWS KMS key ARN

これは、保管時の暗号化用の AWS KMS Amazon リソースネーム (ARN) です。指定しない場合、システムは AWS Entity Resolution マネージド KMS キーを使用します。

バッチワークフロー

スケジュールされた間隔で実行され、データセット全体のデータを照合して解決するプロセス。のバッチワークフロー AWS Entity Resolution は、初期設定、定期的なフル更新、ソースデータセットとターゲットデータセットの両方に大きな変更があるシナリオに最適です。

クリアテキスト

暗号化で保護されていないデータ。

信頼レベル (ConfidenceLevel)

ML マッチングの場合、ML が一致するレコードセットを識別する AWS Entity Resolution ときに によって適用される信頼レベルです。これは、出力に含まれる一致するワークフローメタデータの一部です。

復号

暗号化されたデータを元の形式に戻すプロセスです。復号化は、シークレットキーにアクセスできる場合にのみ実行できます。

暗号化

キーと呼ばれる秘密の値を使用して、データをランダムに見える形式にエンコードするプロセスです。キーにアクセスしない限り、元のプレーンテキストを特定することはできません。

グループ名

グループ名は入力フィールドのグループ全体を参照し、解析されたデータを一致する目的でグループ化するのに役立ちます。

例えば、、**first_name**、および の 3 つの入力フィールドがある場合**last_name**、一致と出力**full_name**のために としてグループ名を入力して**middle_name**、それらをグループ化できます。

ハッシュ

ハッシュとは、固定サイズの不可逆的で一意の文字列を生成する暗号化アルゴリズムを適用することを意味します。これは hash. AWS Entity Resolution uses Secure Hash Algorithm 256-bit (SHA256) ハッシュプロトコルと呼ばれ、32 バイトの文字列を出力します。では AWS Entity Resolution、出力でデータ値をハッシュするかどうかを選択できます。

ハッシュプロトコル (HashingProtocol)

AWS Entity Resolution は Secure Hash Algorithm 256 ビット (SHA256) ハッシュプロトコルを使用し、32 バイトの文字列を出力します。これは、出力に含まれる一致するワークフローメタデータの一部です。

ID マッピング方法

ID マッピングの実行方法。

ID マッピングには 2 つの方法があります。

- ルールベース – 一致するルールを使用して、ID マッピングワークフローのソースからターゲットにファーストパーティデータを変換する方法。
- プロバイダーサービス – プロバイダーサービスを使用して、ID マッピングワークフローでサードパーティーでエンコードされたデータをソースからターゲットに変換する方法。

  AWS Entity Resolution は現在、プロバイダーのサービスベースの ID マッピング方法として LiveRamp をサポートしています。この方法 AWS Data Exchange を使用するには、 を通じて LiveRamp へのサブスクリプションが必要です。詳細については、「ステップ 1: でプロバイダーサービスをサブスクライブする AWS Data Exchange」を参照してください。

ID マッピングワークフロー

指定された ID マッピング方法に基づいて、入力データソースから入力データターゲットにデータをマッピングするデータ処理ジョブ。これにより、ID マッピングテーブルが生成されます。このワークフローでは、ID マッピング方法と、ソースからターゲットに変換する入力データを指定する必要があります。

ID マッピングワークフローを設定して、独自の または 2 つの AWS アカウント で実行できます AWS アカウント。

ID 名前空間

複数の AWS アカウント データセットを説明するメタデータと、ID マッピングワークフローでこれらのデータセットを使用する方法 AWS Entity Resolution を含む のリソース。

ID 名前空間には、 SOURCEと の 2 種類がありますTARGET。には、ID マッピングワークフローで処理されるソースデータの設定SOURCEが含まれています。には、すべてのソースが解決されるターゲットデータの設定TARGETが含まれています。2 つの で解決する入力データを定義するには AWS アカウント、ID 名前空間ソースと ID 名前空間ターゲットを作成して、データを 1 つのセット (SOURCE) から別のセット () に変換しますTARGET。

自分と別のメンバーが ID 名前空間を作成し、ID マッピングワークフローを実行したら、 でコラボレーションに参加 AWS Clean Rooms して、ID マッピングテーブルでマルチテーブル結合を実行し、データを分析できます。

詳細については、「AWS Clean Rooms ユーザーガイド」を参照してください。

増分ワークフロー

データセット全体を処理するのではなく、前回の実行以降に新規または更新されたレコードのみを照合して解決するプロセス。の増分ワークフロー AWS Entity Resolution は、データセットのごく一部のみが変更されたときにデータの鮮度を維持するための頻繁な更新に最適です。

入力フィールド

入力フィールドは、 AWS Glue 入力データテーブルの列名に対応します。

入力ソース ARN (InputSourceARN)

AWS Glue テーブル入力用に生成された Amazon リソースネーム (ARN)。これは、出力に含まれる一致するワークフローメタデータの一部です。

機械学習ベースのマッチング

機械学習ベースのマッチング (ML マッチング) は、不完全なデータやまったく同じように見えないデータ間で一致を検索します。ML マッチングは、入力したすべてのデータのレコードを照合しようとするプリセットプロセスです。ML マッチングは、一致したデータセットごとに一致 ID と信頼レベルを返します。

手動処理

オンデマンドで実行できるようにする、一致するワークフロージョブの処理頻度オプション。

このオプションはデフォルトで設定され、ルールベースのマッチングと機械学習ベースのマッチングの両方で使用できます。

Many-to-Manyマッチング

Many-to-manyマッチングは、類似データの複数のインスタンスを比較します。同じ一致キーが割り当てられた入力フィールドの値は、同じ入力フィールドにあるか異なる入力フィールドにあるかに関係なく、互いに照合されます。

たとえば、 mobile_phoneや など、同じ一致キー「Phonehome_phone」を持つ複数の電話番号入力フィールドがあるとします。many-to-manyマッチングを使用して、mobile_phone入力フィールドのデータとmobile_phone入力フィールドのデータおよびhome_phone入力フィールドのデータを比較します。

一致ルールは、 (または) オペレーションで同じ一致キーを持つ複数の入力フィールドのデータを評価し、one-to-many一致は複数の入力フィールドの値を比較します。つまり、2 つのレコード間で mobile_phoneまたは の組み合わせhome_phoneが一致した場合、「電話」一致キーは一致を返します。一致を見つけるための一致キー「Phone」の場合は、Record One mobile_phone = Record Two mobile_phoneOR Record One mobile_phone = Record Two home_phone OR Record One home_phone = Record Two home_phone OR ですRecord One home_phone = Record Two mobile_phone。

一致 ID (MatchID)

ルールベースのマッチングと ML マッチングの場合、これは によって生成 AWS Entity Resolution され、一致する各レコードセットに適用される ID です。これは、出力に含まれる一致するワークフローメタデータの一部です。

一致キー (MatchKey)

一致キーは、 AWS Entity Resolution どの入力フィールドを類似データと見なし、どの入力フィールドを異なるデータと見なすかを指示します。これにより、ルールベースのマッチングルール AWS Entity Resolution を自動的に設定し、異なる入力フィールドに保存されている同様のデータを比較できます。

mobile_phone 入力フィールドやhome_phone入力フィールドなど、比較するデータに複数のタイプの電話番号情報がある場合は、両方の一致キー「Phone」を指定できます。次に、ルールベースの

一致を設定して、すべての入力フィールドの「または」ステートメントと「電話」一致キーを使用してデータを比較できます (「一致ワークフロー」セクションのOne-to-One の一致とMany-to-Many一致の定義」を参照してください)。

ルールベースのマッチングで異なるタイプの電話番号情報を個別に考慮する場合は、「Mobile_Phone」や「Home_Phone」などのより具体的なマッチングキーを作成できます。次に、一致するワークフローを設定するときに、各電話一致キーをルールベースのマッチングで使用する方法を指定できます。

特定の入力フィールドに MatchKey が指定されていない場合、マッチングには使用できませんが、マッチングワークフロープロセスを通じて実行でき、必要に応じて出力できます。

一致キー名

一致キーに割り当てられた名前。

一致ルール (MatchRule)

ルールベースのマッチングの場合、これは、一致したレコードセットを生成するために適用されたルール番号です。これは、出力に含まれる一致するワークフローメタデータの一部です。

一致

さまざまな入力フィールド、テーブル、またはデータベースのデータを組み合わせて比較し、特定の一致基準を満たす (例えば、一致するルールやモデルを通じて) ことに基づいて、どちらが類似しているか、または「一致する」かを判断するプロセス。

マッチングワークフロー

一致する入力データとマッチングの実行方法を指定するように設定したプロセス。

一致するワークフローの説明

入力することを選択できる、一致するワークフローのオプションの説明。説明は、複数のワークフローを作成する場合、一致するワークフローを区別するのに役立ちます。

一致するワークフロー名

指定した一致するワークフローの名前。

> **Note**
>
> 一致するワークフロー名は一意である必要があります。同じ名前にすることはできません。そうしないと、エラーが返されます。

ワークフローメタデータの一致

一致するワークフロージョブ AWS Entity Resolution 中に によって生成および出力される情報。この情報は出力時に必要です。

正規化 (ApplyNormalization)

スキーマで定義されているように入力データを正規化するかどうかを選択します。正規化は、余分なスペースや特殊文字を削除し、小文字の形式に標準化することで、データを標準化します。

たとえば、入力フィールドの属性タイプがフルフォンで、入力テーブルの値が の形式である場合(123) 456-7890、 AWS Entity Resolution は値を に正規化します1234567890。

> **Note**
>
> 正規化は、名前、住所、電話番号、E メールのグループタイプでのみサポートされます。

以下のセクションでは、標準の正規化ルールについて説明します。

ML ベースのマッチングについては、「」を参照してください正規化 (ApplyNormalization) – ML ベースのみ。

トピック

- 名前
- E メール
- 電話
- Address

- ハッシュ
- Source_ID

名前

> **Note**
>
> 正規化は名前グループタイプでのみサポートされています。
> 名前グループタイプは、コンソールではフルネームとして、API NAME では として表示されます。
> Name グループタイプのサブタイプを正規化する場合:
>
> - コンソールで、フルネームグループに名、ミドルネーム、姓のサブタイプを割り当てます。
> - CreateSchemaMapping API で、NAMEgroupName に次のタイプを割り当てます: NAME_FIRST、NAME_MIDDLE、NAME_LAST。

- TRIM = 先頭と末尾の空白をトリミングする
- LOWERCASE = すべてのアルファ文字を小文字にします
- CONVERT_ACCENT = Covert アクセント付き文字から通常の文字へ
- REMOVE_ALL_NON_ALPHA = 英数字以外の文字をすべて削除します [a-zA-Z]

E メール

> **Note**
>
> 正規化は E メールグループタイプでサポートされています。
> E メールグループタイプは、コンソールには E メールアドレスとして、API EMAIL_ADDRESS には として表示されます。

- TRIM = 先頭と末尾の空白をトリミングする
- LOWERCASE = すべてのアルファ文字を小文字にします
- CONVERT_ACCENT = Covert アクセント付き文字から通常の文字へ

- EMAIL_ADDRESS_UTIL_NORM = ユーザー名からドット (.) を削除し、ユーザー名のプラス記号 (+) の後にすべてを削除し、一般的なドメインバリエーションを標準化します。
- REMOVE_ALL_NON_EMAIL_CHARS = non-alpha-numeric文字 [a-zA-Z0-9] と [.@-] をすべて削除します

電話

> **Note**
>
> 正規化は、電話グループタイプでのみサポートされています。
> 電話グループタイプは、コンソールではフルフォンとして、API PHONE では として表示されます。
> 電話グループタイプのサブタイプを正規化する場合:
>
> - コンソールで、完全な電話グループに電話番号と電話の国コードのサブタイプを割り当てます。
> - CreateSchemaMapping API で、次のタイプを PHONE groupName PHONE_NUMBERと に割り当てますPHONE_COUNTRYCODE。

- TRIM = 先頭と末尾の空白をトリミングする
- REMOVE_ALL_NON_NUMERIC = 数値以外の文字をすべて削除します [0-9]
- REMOVE_ALL_LEADING_ZEROES = 先頭のゼロをすべて削除します
- EN"_PREFIX_WITH_MAP, "phonePrefixMap" = 各電話番号を調べ、phonePrefixMap のパターンと照合しようとします。一致が見つかった場合、ルールは電話番号のプレフィックスを追加または変更して、マップで指定された標準化された形式に準拠していることを確認します。

Address

> **Note**
>
> 正規化は、アドレスグループタイプでのみサポートされています。
> アドレスグループタイプは、コンソールにはフルアドレスとして、API ADDRESS　にはフルアドレスとして表示されます。
> Address グループタイプのサブタイプを正規化する場合:

- コンソールで、フルアドレスグループに次のサブタイプを割り当てます。住所 1、住所 2: 住所 3 名、市名、州、国、郵便番号 t
- CreateSchemaMapping API で、ADDRESSgroupName に次のタイプを割り当てます: ADDRESS_STREET1、ADDRESS_STREET2、ADDRESS_STREET3、、ADDRESS_CITYADDRESS_STA ADDRESS_POSTALCODE

- TRIM = 先頭と末尾の空白をトリミングする
- LOWERCASE = すべてのアルファ文字を小文字にします
- CONVERT_ACCENT = Covert アクセント付き文字から通常の文字へ
- REMOVE_ALL_NON_ALPHA = アルファベット以外の文字をすべて削除します [a-zA-Z]
- ADDRESS_RENAME_WORD_MAP を使用した RENAME_WORDS = Address 文字列の単語を ADDRESS_RENAME_WORD_MAP の単語に置き換える
- ADDRESS_RENAME_DELIMITER_MAP を使用する RENAME_DELIMITERS = Address 文字列の区切り文字を ADDRESS_RENAME_DELIMITER_MAP の文字列に置き換えます
- ADDRESS_RENAME_DIRECTION_MAP を使用する RENAME_DIRECTIONS= Address 文字列の区切り文字を ADDRESS_RENAME_DIRECTION_MAP の文字列に置き換えます
- ADDRESS_RENAME_NUMBER_MAP を使用する RENAME_NUMBERS = Address 文字列の数値を ADDRESS_RENAME_NUMBER_MAP の文字列に置き換えます
- ADDRESS_RENAME_SPECIAL_CHAR_MAP を使用する RENAME_SPECIAL_CHARS = Address 文字列の特殊文字を ADDRESS_RENAME_SPECIAL_CHAR_MAP の文字列に置き換えます

ADDRESS_RENAME_WORD_MAP

これらは、アドレス文字列を正規化するときに名前が変更される単語です。

```
"avenue": "ave",
 "bouled": "blvd",
 "circle": "cir",
 "circles": "cirs",
 "court": "ct",
 "centre": "ctr",
 "center": "ctr",
 "drive": "dr",
 "freeway": "fwy",
 "frwy": "fwy",
```

```
 "highway": "hwy",
 "lane": "ln",
 "parks": "park",
 "parkways": "pkwy",
 "pky": "pkwy",
 "pkway": "pkwy",
 "pkwys": "pkwy",
 "parkway": "pkwy",
 "parkwy": "pkwy",
 "place": "pl",
 "plaza": "plz",
 "plza": "plz",
 "road": "rd",
 "square": "sq",
 "squ": "sq",
 "sqr": "sq",
 "street": "st",
 "str": "st",
 "str.": "strasse"
```

ADDRESS_RENAME_DELIMITER_MAP

これらは、アドレス文字列を正規化するときに名前が変更される区切り文字です。

```
",": " ",
".": " ",
"[": " ",
"]": " ",
"/": " ",
"-": " ",
"#": " number "
```

ADDRESS_RENAME_DIRECTION_MAP

これらは、アドレス文字列を正規化するときに名前が変更される方向識別子です。

```
"east": "e",
"north": "n",
"south": "s",
"west": "w",
"northeast": "ne",
"northwest": "nw",
```

```
"southeast": "se",
"southwest": "sw"
```

ADDRESS_RENAME_NUMBER_MAP

これらは、アドレス文字列を正規化するときに名前が変更される数値文字列です。

```
"número": "number",
 "numero": "number",
 "no": "number",
 "núm": "number",
 "num": "number"
```

ADDRESS_RENAME_SPECIAL_CHAR_MAP

これらは、アドレス文字列を正規化するときに名前が変更される特殊文字文字列です。

```
"ß": "ss",
 "ä": "ae",
 "ö": "oe",
 "ü": "ue",
 "ø": "o",
 "æ": "ae"
```

ハッシュ

- TRIM = 先頭と末尾の空白をトリミングする

Source_ID

- TRIM = 先頭と末尾の空白をトリミングする

正規化 (ApplyNormalization) – ML ベースのみ

スキーマで定義されているように入力データを正規化するかどうかを選択します。正規化は、余分なスペースや特殊文字を削除し、小文字の形式に標準化することで、データを標準化します。

たとえば、入力フィールドの属性タイプが でNAME、入力テーブルの値が の形式である場合Johns Smith、 AWS Entity Resolution は値を に正規化しますjohn smith。

以下のセクションでは、機械学習ベースのマッチングワークフローの正規化ルールについて説明します。

トピック

- 名前
- E メール
- 電話

名前

- TRIM = 先頭と末尾の空白をトリミングする
- LOWERCASE = すべてのアルファ文字を小文字にします

E メール

- LOWERCASE = すべてのアルファ文字を小文字にします
- (at)(大文字と小文字を区別) のみを @ 記号に置き換えます
- 値内の任意の場所にあるすべての空白を削除します。
- 存在する"<>"場合、最初の の外部にあるものをすべて削除します

電話

- TRIM = 先頭と末尾の空白をトリミングする
- REMOVE_ALL_NON_NUMERIC = 数値以外の文字をすべて削除します [0 ~ 9]
- REMOVE_ALL_LEADING_ZEROES = 先頭のゼロをすべて削除します
- EN"_PREFIX_WITH_MAP, "phonePrefixMap" = 各電話番号を調べ、phonePrefixMap のパターンと照合しようとします。一致が見つかった場合、ルールは電話番号のプレフィックスを追加または変更して、マップで指定された標準化された形式に準拠していることを確認します。

One-to-Oneマッチング

One-to-one のマッチングは、類似データの単一インスタンスを比較します。同じ入力フィールド内の同じ一致キーと値を持つ入力フィールドは、互いに照合されます。

たとえば、 mobile_phoneや など、同じ一致キー「Phonehome_phone」を持つ複数の電話番号入力フィールドがあるとします。one-to-oneのマッチングを使用して、mobile_phone入力フィールド内のデータとmobile_phone入力フィールド内のデータを比較し、home_phone入力フィールド内のデータとhome_phone入力フィールド内のデータを比較します。mobile_phone 入力フィールドのデータは、home_phone入力フィールドのデータと比較されません。

一致ルールは、 (または) オペレーションで同じ一致キーを持つ複数の入力フィールドのデータを評価し、one-to-many一致は 1 つの入力フィールド内の値を比較します。つまり、2 つのレコード間で mobile_phoneまたは home_phoneが一致すると、「電話」一致キーは一致を返します。一致を見つけるための一致キー「Phone」の場合は、 Record One mobile_phone = Record Two mobile_phone または Record One home_phone = Record Two home_phone。

一致ルールは、 (および) オペレーションで異なる一致キーを持つ入力フィールドのデータを評価します。ルールベースのマッチングで異なるタイプの電話番号情報を個別に考慮する場合は、「mobile_phone」や「home_phone」などのより具体的なマッチングキーを作成できます。ルールで両方の一致キーを使用して一致を検索する場合は、 Record One mobile_phone = Record Two mobile_phone AND Record One home_phone = Record Two home_phone。

Output

OutputAttribute オブジェクトのリスト。各オブジェクトには名前とハッシュというフィールドがあります。これらの各オブジェクトは、 AWS Glue 出力テーブルに含める列と、列内の値をハッシュするかどうかを表します。

OutputS3Path

AWS Entity Resolution が出力テーブルを書き込む S3 送信先。

OutputSourceConfig

OutputSource オブジェクトのリスト。各オブジェクトには OutputS3PathApplyNormalization、および Output フィールドがあります。

プロバイダーのサービスベースのマッチング

プロバイダーのサービスベースのマッチングは、優先データサービスプロバイダーとライセンスデータセットを使用してレコードを照合、リンク、強化するプロセスです。このマッチング手法を使用するには、プロバイダーサービス AWS Data Exchange で のサブスクリプションが必要です。

AWS Entity Resolution は現在、次のデータサービスプロバイダーと統合されています。

- LiveRamp
- TransUnion
- UID 2.0

ルールベースのマッチング

ルールベースのマッチングは、完全一致を見つけるように設計されたプロセスです。ルールベースのマッチングは、ウォーターフォールマッチングルールの階層的なセットであり、入力したデータに基づいて提案され AWS Entity Resolution、ユーザーが完全に設定可能です。ルール条件内で提供されるすべての一致キーは、比較データを一致と宣言し、関連するメタデータを出力するために完全に一致する必要があります。ルールベースの一致は、一致したデータセットごとに一致 ID とルール番号を返します。

エンティティを一意に識別できるルールを定義することをお勧めします。ルールを順序付けして、より正確な一致を最初に見つけます。

たとえば、ルール 1 とルール 2 の 2 つのルールがあるとします。

これらのルールには、次の一致キーがあります。

- ルール 1 にはフルネームと住所が含まれます
- ルール 2 にはフルネーム、住所、電話番号が含まれます

ルール 1 が最初に実行されるため、すべてルール 1 で見つかったはずであるため、ルール 2 では一致は見つかりません。

電話によって区別される一致を検索するには、次のようにルールの順序を変更します。

- ルール 2 にはフルネーム、住所、電話番号が含まれます
- ルール 1 にはフルネームと住所が含まれます

Schema

一連のデータの編成と接続方法を定義する構造またはレイアウトに使用される用語。

スキーマの説明

入力できるスキーマのオプションの説明。説明は、複数のスキーマを作成する場合にスキーママッピングを区別するのに役立ちます。

スキーマ名

スキーマの名前。

> **Note**
>
> スキーマ名は一意である必要があります。同じ名前にすることはできません。そうしないと、エラーが返されます。

スキーママッピング

のスキーママッピング AWS Entity Resolution は、マッチングのためにデータを解釈 AWS Entity Resolution する方法を指示するプロセスです。一致するワークフローに AWS Entity Resolution 読み込む入力データテーブルのスキーマを定義します。

スキーママッピング ARN

スキーママッピング用に生成された Amazon リソースネーム (ARN)。

一意の ID

指定した一意の識別子で、 が AWS Entity Resolution 読み取る入力データの各行に割り当てる必要があります。

Example

たとえば、**`Primary_key`**、**`Row_ID`**、または **`Record_ID`** などです。

Unique ID 列は必須です。

一意の ID は、単一のテーブル内の一意の識別子である必要があります。

一意の ID はこのパターンを満たす必要があります。[a-zA-Z0-9_-]

異なるテーブル間で、一意の ID に重複した値を含めることができます。

一致するワークフローの一意の ID の最大長は 38 です

の一意の ID の最大長 257 文字 ID マッピングワークフロー

一致するワークフローが実行されると、一意の ID が次の場合、レコードは拒否されます。

- が指定されていない
- は同じテーブル内で一意ではありません
- ソース間で属性名の点で重複する
- が 38 文字を超えている (ルールベースのマッチングワークフローのみ)

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。