aws

ユーザーガイド

AWS Clean Rooms

AWS Clean Rooms: ユーザーガイド

Table of Contents

とは AWS Clean Rooms

AWS Clean Rooms は、ユーザーとパートナーが集合データセットを分析して共同作業を行い、基盤となるデータを互いに開示することなく、新しいインサイトを得ることができます。安全なコラボレーションワークスペース AWS Clean Roomsである を使用すると、独自のクリーンルームを数分で作成し、わずか数ステップで集合データセットの分析を開始できます。コラボレーションを行うパートナーを選び、そのパートナーのデータセットを選択して、参加者に制限を設定することができます。

を使用すると AWS Clean Rooms、すでに を使用している何千もの企業とコラボレーションできます AWS。コラボレーションでは、データを から移動したり AWS 、別のプラットフォームにロードしたりする必要はありません。クエリを実行すると、 は元の場所からデータを AWS Clean Rooms 読み取り、組み込みの分析ルールを適用して、データの制御を維持します。

AWS Clean Rooms には、設定できる組み込みのデータアクセスコントロールと監査サポートコントロールが用意されています。これらの制御には以下が含まれます。

- 分析ルール: SQL クエリを制限し、出力に制約を設けます。
- Cryptographic Computing for Clean Rooms: 厳格なデータ処理ポリシーに準拠するために、クエリが処理されている間もデータを暗号化したまま維持します。
- クエリログ記録: クエリの確認や監査サポートに役立ちます。
- ユーザー識別の試みから保護するための差分プライバシー。 AWS Clean Rooms 差分プライバシーは、数学的にバックアップされた手法と直感的コントロールを使用して、数回のクリックでユーザーのプライバシーを保護するフルマネージド機能です。
- AWS Clean Rooms ML は、データを相互に共有しなくても、2 人の関係者がデータ内の類似ユーザーを識別できるようにします。1 番目の関係者はトレーニングデータから類似モデルを作成し、設定します。2 番目の関係者はシードデータをコラボレーションに持ち込み、トレーニングデータに似た類似セグメントを作成します。

次の動画では、 について詳しく説明します AWS Clean Rooms。

AWS Clean Rooms

初めての AWS Clean Rooms ユーザーですか？

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

AWS Clean Rooms の仕組み

以下に説明するワークフローは次の条件を前提としています。

- コラボレーションメンバーが、既にデータテーブルを Amazon S3 にアップロードしており、AWS Glue テーブルを作成済みであること。
- (オプション) 暗号化されたデータテーブルの場合は、コラボレーションメンバーが、C3R 暗号化クライアントを使用して既に暗号化されたデータテーブルを準備していること。

要約すると、 のワークフロー AWS Clean Rooms は次のとおりです。

1. コラボレーションクリエーターが、以下のタスクを行います。
   - コラボレーションを作成します。
   - 1 人以上のメンバーをコラボレーションに招待します。
   - クエリを行えるメンバーや結果を受け取れるメンバーなど、メンバーに能力を割り当てます。

     コラボレーションクリエーターが結果を受け取れるメンバーでもある場合は、クエリ結果の送信先と形式を指定します。また、クエリ結果の送信先に結果を書き込むための、サービスロールの Amazon リソースネーム (ARN) も指定します。
   - コラボレーションでクエリの計算コストを負担するメンバーを設定します。
2. 招待されたメンバーが、メンバーシップリソースを作成してコラボレーションに参加します。

   招待されたメンバーが結果を受け取れるメンバーである場合は、そのメンバーがクエリ結果の送信先と形式を指定します。また、クエリ結果の送信先に書き込むための、サービスロールの ARN も指定します。

   招待されたメンバーがクエリの計算コストを負担するメンバーである場合は、コラボレーションに参加する前に、そのメンバーが支払いの責任を承諾します。

3. メンバーは、 で使用する既存の AWS Glue テーブルを設定します AWS Clean Rooms。(このステップは、Cryptographic Computing for Clean Rooms を使用している場合を除き、コラボレーションに参加する前でも後でも実行できます)。

> **Note**
>
> AWS Clean Rooms は AWS Glue テーブルをサポートします。 AWS Glueのデータの取得の詳細については、「ステップ 3: データテーブルを Amazon S3 にアップロードする」を参照してください。

1. メンバーが、設定済みテーブルに名前を付け、コラボレーションに使用する列を選択します。
2. メンバーが、設定済みテーブルに以下の分析ルールのいずれかを設定します。
   - 集計分析ルールまたはリスト分析ルール – テーブルに対して実行できる分析の種類を制御します。
   - カスタム分析ルール – 事前に承認された特定のクエリを許可したり、自身のデータを使用するクエリの提供を特定のアカウントに許可したりします。メンバーは差分プライバシーをオンにして、ユーザー特定の試行から保護できます。

> **Note**
>
> メンバーは、設定済みテーブルをコラボレーションに関連付ける前に、いつでも分析ルールを設定できます。

4. メンバーは、設定したテーブルをコラボレーションに関連付け、 AWS Glue テーブルにアクセスするための AWS Clean Rooms サービスロールを付与します。

> **Note**
>
> このサービスロールにはテーブルに対するアクセス許可があります。サービスロールは、 がクエリを行えるメンバーに代わって許可されたクエリ AWS Clean Rooms を実行する場合にのみ引き受けることができます。コラボレーションメンバー (データの所有者以外) は誰も、コラボレーションの基になるテーブルにはアクセスできません。データの所有者は差分プライバシーをオンにして、他のメンバーがテーブルのクエリを利用できるようにすることが可能です。

5. クエリを行えるメンバーが、設定済みテーブルで SQL クエリを実行します。

   クエリを実行できるのは、クエリの計算コストを負担するメンバーがアクティブメンバーとしてコラボレーションに参加している場合のみです。

   分析ルールと出力制約は自動的に適用されます。 は、ステップ 3.b で定義された分析ルールに準拠している結果 AWS Clean Rooms のみを返します。

   暗号化されたデータに対するクエリの場合、結果を受け取ることができるメンバーは、暗号化された出力を から受け取り AWS Clean Rooms 、そこから復号する必要があります (ステップ 8 を参照) 。

6. 結果を受け取ることができるメンバーは、 AWS Clean Rooms コンソールまたは指定した Amazon S3 バケットで結果を確認します。
7. クエリの計算コストを負担するメンバーに、コラボレーションで実行されたクエリの料金が請求されます。
8. (オプション) 暗号化されたデータテーブルの場合は、結果を受け取れるメンバーが、C3R 暗号化クライアントを復号モードで実行してクエリ結果を復号化します。

関連サービス

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

- Amazon S3

  コラボレーションメンバーは、Amazon S3 AWS Clean Rooms に取り込むデータを保存できます。

  詳細については、次のトピックを参照してください。

  でのクエリ用のデータテーブルの準備 AWS Clean Rooms

  「Amazon Simple Storage Service ユーザーガイド」の「Amazon S3 とは?」

- AWS Glue

  コラボレーションメンバーは、Amazon S3 のデータから AWS Glue テーブルを作成し、 で使用できます AWS Clean Rooms。

  詳細については、次のトピックを参照してください。

でのクエリ用のデータテーブルの準備 AWS Clean Rooms

「AWS Glue デベロッパーガイド」の「AWS Glueの概要」

- AWS CloudFormation

  で次のリソースを作成します AWS CloudFormation: コラボレーション、設定済みテーブル、設定済みテーブルの関連付け、メンバーシップ

  詳細については、「AWS Clean Rooms によるリソースの作成 AWS CloudFormation」を参照してください。

- AWS CloudTrail

  CloudTrail を ログ AWS Clean Rooms とともに使用して、 AWS のサービス アクティビティの分析を強化します。

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

アクセス AWS Clean Rooms

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

- https://console.aws.amazon.com/cleanrooms/ の AWS Clean Rooms コンソールから直接。
- AWS Clean Rooms API を使用してプログラムで。詳細については、「AWS Clean Rooms APIリファレンス」を参照してください。

の料金 AWS Clean Rooms

料金に関する情報については、[AWS Clean Rooms の料金]を参照してください。

の請求 AWS Clean Rooms

AWS Clean Rooms では、コラボレーションクリエーターは、コラボレーションでクエリコンピューティングコストを支払うメンバーを設定できます。

ほとんどの場合、クエリを行えるメンバーとクエリの計算コストを負担するメンバーは同じです。ただし、クエリを行えるメンバーとクエリの計算コストを負担するメンバーが異なる場合は、クエリを

行えるメンバーが自分のメンバーシップリソースに対してクエリを実行しても、請求は、クエリの計算コストを負担するメンバーのメンバーシップリソースに行われます。

クエリの計算コストを負担するメンバーは、クエリを実行しているメンバーでもクエリが実行されているリソースの所有者でもないため、 CloudTrail イベント履歴にクエリが実行されているイベントは表示されません。しかし、支払いに関しては、クエリを行えるメンバーが実行したすべてのクエリについて、コストを負担するメンバーのメンバーシップリソースで請求書が生成され、表示されます。

コラボレーションを作成する方法と、クエリの計算コストを負担するメンバーを設定する方法の詳細については、「コラボレーションを作成する」を参照してください。

の分析ルール AWS Clean Rooms

テーブルをコラボレーション分析に使用できるようにする一環として、コラボレーションメンバーは分析ルールを設定する必要があります。 AWS Clean Rooms

分析ルールは、各データ所有者が設定済みテーブルに設定するプライバシー強化のためのコントロールです。分析ルールによって、設定済みテーブルの分析方法が決まります。

分析ルールは、設定済みテーブルをアカウントレベルで制御するもの (アカウントレベルのリソース) で、設定済みテーブルが関連付けられているすべてのコラボレーションに適用されます。分析ルールが設定されていない場合、設定済みテーブルをコラボレーションに関連付けることはできますが、クエリは実行できません。クエリは、分析ルールの種類が同一の設定済みテーブルのみを参照できます。

分析ルールを設定するには、まず分析の種類を選択し、次に分析ルールを指定します。どちらの手順でも、有効にするユースケースと、基になるデータをどのように保護するかを検討する必要があります。

AWS Clean Rooms クエリで参照されるすべての設定済みテーブルに対して、より制限の厳しい制御を適用します。

制限に関するコントロールの例を以下に示します。

Example 制限に関するコントロール: 出力の制約

- コラボレーター A の ID 列には 100 という出力の制約があります。
- コラボレーター B の ID 列には 150 という出力の制約があります。

  両方の設定済みテーブルを参照する集約クエリで、クエリ出力に行を表示するには、1 つの出力行に少なくとも 150 個の個別 ID 値が必要です。クエリ出力には、出力の制約により結果が削除されたことは示されません。

Example 制限に関するコントロール: 未承認の分析テンプレート

- コラボレーター A は、カスタム分析ルールで、コラボレーター A とコラボレーター B の設定済みテーブルを参照するクエリを含む分析テンプレートを許可しました。
- コラボレーター B は分析テンプレートを許可していません。

コラボレーター B は分析テンプレートを許可していないため、クエリを実行できるメンバーはその分析テンプレートを実行できません。

分析ルールの種類

分析ルールには、集計、リスト、カスタムの 3 種類があります。以下の表で、分析ルールの種類を比較しています。各種分析ルールの指定については、それぞれ別のセクションで説明しています。

以下の表は、各種分析ルールの比較をまとめたものです。

対応するユースケース

次の表は、各種分析ルールで対応しているユースケースの比較をまとめたものです。

ユースケース	集計	リスト	カスタム
対応する分析	COUNT、SU、および AVG 関数を使用して、任意のディメンションで統計を集約するクエリ	複数のテーブルが重複している箇所の行レベルのリストを出力するクエリ	分析テンプレートまたは分析作成者によって確認、許可されている、あらゆるカスタム分析
一般的なユースケース	セグメント分析、測定、アトリビューション	エンリッチメント、セグメント構築	ファーストタッチアトリビューション、増分分析、オーディエンス発掘
SQL 構文	• ジョインステート	• JOIN ステートメ	SELECT コマンドで使

ユースケース	集計	リスト	カスタム
	メント:内部ジョイン • 集計関数:カウント/カウントが異なる、合計/合計が異なる、および AVG • スカラー関数:限定サブセット	ント:内部結合 • スカラー関数: なし	用できる SQL 関数と SQL 構文の大部分
サブクエリと共通テーブル式 (CTE)	いいえ	いいえ	はい
分析テンプレート	いいえ	いいえ	はい

対応するコントロール

次の表は、各分析ルールタイプが参照元データをどのように保護しているかを比較した概要を示しています。

コントロール	集計	リスト	カスタム
コントロールのメカニズム	テーブル内のデータをクエリでどのように使用してよいかを制御します (hashed_email 列の COUNT と SUM を許可するなど)。	テーブル内のデータをクエリでどのように使用してよいかを制御します (hashed_email 列の使用を結合のみで許可するなど)。	テーブルでどのクエリを実行してよいかを制御します (分析テンプレート「Custom query 1」で定義されたクエリのみを許可するなど)。
組み込みのプライバシー強化機能	• ブラインドマッチ • 集約が必須 • 最小数の集約しきい値 >= • 事前定義されたクエリ構造	• ブラインドマッチ • 重複が必須 • 事前定義されたクエリ構造	差分プライバシー
実行前のクエリの確認	いいえ	いいえ	はい。分析テンプレートを使用

で使用できる分析ルールの詳細については AWS Clean Rooms、以下のトピックを参照してください。

- 集計分析ルール
- リスト分析ルール
- のカスタム分析ルール AWS Clean Rooms

集計分析ルール

AWS Clean Rooms の集計分析ルールでは、COUNT、SUM、または AVG 関数を使用して、任意のディメンションで統計を集約します。設定済みテーブルに集計分析ルールを追加すると、クエリを行えるメンバーが設定済みテーブルに対してクエリを実行できるようになります。

集計分析ルールは、キャンペーン計画、メディアリーチ、頻度測定、アトリビューションなどのユースケースに対応します。

サポートされているクエリ構造と構文は、「集約クエリの構造と構文」で定義されています。

「集計分析ルール - クエリコントロール」で定義されている分析ルールのパラメータには、クエリコントロールとクエリ結果コントロールがあります。クエリコントロールでは、直接または間接的にクエリを実行できるメンバーが所有する 1 つ以上の設定済みテーブルに、1 つの設定済みテーブルを結合することを要求できます。この要求により、クエリを自分のテーブルと相手のテーブルの交差部分 (INNERJOIN) で実行することが可能になります。

集約クエリの構造と構文

集計分析ルールが追加されたテーブルに対するクエリは、次の構文に従う必要があります。

```
    --select_aggregate_function_expression
SELECT
aggregation_function(column_name) [[AS] column_alias ] [, ...]

    --select_grouping_column_expression
  [, {column_name|scalar_function(arguments)} [[AS] column_alias ]][, ...]

--table_expression
FROM table_name [[AS] table_alias ]
  [[INNER] JOIN table_name [[AS] table_alias] ON join_condition] [...]

--where_expression
[WHERE where_condition]
```

```
--group_by_expression
[GROUP BY {column_name|scalar_function(arguments)}, ...]]

--having_expression
[HAVING having_condition]

--order_by_expression
[ORDER BY {column_name|scalar_function(arguments)} [{ASC|DESC}]] [,...]]
```

次の表は、前述の構文で示したそれぞれの式について説明しています。

式	定義	例
select_aggregate_function_expression	次の式を含むカンマ区切りリストです。 • select_aggregation_function_expression • select_aggregate_expression **Note** select_aggregate_expression には少なくとも 1 つの select_aggregation_function_expression が必要です。	SELECT SUM(PRICE), user_segment
select_aggregation_function_expression	1 つ以上の列に適用される、1 つ以上のサポートされている集約関数です。集約関数の引	AVG(PRICE) COUNT(DISTINCT user_id)

式	定義	例
	数として指定できるのは列だけです。 **Note** `select_aggregate_expression` には少なくとも 1 つの `select_aggregation_function_expression` が必要です。	

式	定義	例

select_grouping_column_expression	以下を使用する任意の式を含めることができる式です。
	• テーブルの列名
	• サポートされているスカラー関数
	• [String literals] (文字列リテラル)
	• 数値リテラル
	
	Note
	`select_aggregate_expression` では AS パラメータの有無にかかわらず、列にエイリアスを指定できます。詳細については、「AWS Clean Rooms SQL リファレンス」を参照してください。
		`TRUNC(timestampColumn)`
		
		`UPPER(campaignName)`

式	定義	例
table_expression	join_condition で結合条件式を連結するテーブル、またはテーブルの結合。 join_condition はブール値を返します。 table_expression では、以下がサポートされています。 • 特定の JOIN 型 (INNER JOIN) • join_condition 内の等価比較条件 (=) • 論理演算子 (AND、OR)	`FROM consumer_table INNER JOIN provider_table ON consumer_table.identifier1 = provider_table.identifier1 AND consumer_table.identifier2 = provider_table.identifier2`

式	定義	例
where_expression	ブール値を返す条件式です。次ような要素で構成されています。 • テーブルの列名 • サポートされているスカラー関数 • 算術演算子 • [String literals] (文字列リテラル) • 数値リテラル サポートされている比較条件は (=, >, <, <=, >=, <>, !=, NOT, IN, NOT IN, LIKE, IS NULL, IS NOT NULL) です。 サポートされている論理演算子は (AND, OR) です。 where_expression はオプションです。	WHERE where_condition WHERE price > 100 WHERE TRUNC(timestampColumn) = '1/1/2022' WHERE timestampColumn = timestampColumn2 - 14
group_by_expression	select_grouping_column_expression の要件に合致する式のカンマ区切りリストです。	GROUP BY TRUNC(timestampColumn), UPPER(campaignName), segment

式	定義	例
having_expression	ブール値を返す条件式です。サポートされている集約関数が 1 つの列に適用され (例えば SUM(price))、数値リテラルと比較されます。 サポートされている条件は (=, >, <, <=, >=, <>, !=) です。 サポートされている論理演算子は (AND, OR) です。 having_expression はオプションです。	HAVING SUM(SALES) > 500

式	定義	例
order_by_expression	前述の select_aggregate_expression で定義されているものと同じ要件と互換性のある式のカンマ区切りリストです。 order_by_expression はオプションです。 **Note** order_by_expression では ASC と DESC のパラメータを使用できます。詳細については、「AWS Clean Rooms SQL リファレンス」で ASC DESC パラメータを参照してください。	ORDER BY SUM(SALES), UPPER(campaignName)

集約クエリの構造と構文については、次の点に注意してください。

- SELECT 以外の SQL コマンドはサポートされていません。
- サブクエリと共通テーブル式 (WITH など) はサポートされていません。
- 複数のクエリを組み合わせる演算子 (UNION など) はサポートされていません。
- TOP、LIMIT、および OFFSET パラメータはサポートされていません。

集計分析ルール - クエリコントロール

集約クエリコントロールを使用すると、テーブル内の列を使用してテーブルにクエリを実行する方法を制御できます。例えば、どの列を結合に使用するか、どの列がカウントされるようにするか、WHERE ステートメントでどの列を使用できるようにするかを制御できます。

以下のセクションでは、それぞれのコントロールについて説明します。

トピック

集約コントロール

集約コントロールを使用すると、どの集約関数を許可し、どの列に適用するかを定義できます。集約関数は、SELECT、HAVING、および ORDER BY の式で使用できます。

コントロール	定義	使用方法
`aggregateColumns`	集約関数での使用を許可する設定済みテーブル列の列。	`aggregateColumns` は、SELECT、HAVING、および ORDER BY の式の集約関数で使用できます。 一部の `aggregateColumns` は `joinColumn` としても分類されることがあります (後述)。 特定の `aggregateColumn` は `dimensionColumn` としても分類されることはありません (後述)。

コントロール	定義	使用方法
function	aggregateColumns で使用できる COUNT 関数、SUM 関数、および AVG 関数。	function は関連付けられている aggregateColumns に適用できます。

結合コントロール

JOIN 句を使用して、2 つ以上のテーブルの行を、テーブル間の関連する列に基づいて結合します。

結合コントロールを使用することで、テーブルを table_expression 内の他のテーブルに結合する方法を制御できます。AWS Clean Rooms は、INNER JOIN のみをサポートしています。INNER JOIN ステートメントでは、定義するコントロールに従い、分析ルールで明示的に joinColumn として分類された列のみを使用できます。

INNER JOIN は、自身の設定済みテーブルの joinColumn と、コラボレーション内のもう 1 つの設定済みテーブルの joinColumn で動作する必要があります。テーブルのどの列を joinColumn として使用できるようにするかはテーブルの所有者が決定します。

ON 句内の一致条件ごとに、2 つの列間の等価比較条件 (=) を使用する必要があります。

ON 句では次のようにして複数の一致条件を使用できます。

- AND 論理演算子を使用して組み合わせる
- OR 論理演算子を使用して区切る

> **Note**
>
> JOIN の一致条件では、必ず JOIN の各側の 1 つの行が一致しなければなりません。OR または AND 論理演算子で接続されるすべての条件がこの要件を満たす必要があります。

AND 論理演算子を使用したクエリの例を以下に示します。

```
SELECT some_col, other_col
FROM table1
    JOIN table2
```

```
    ON table1.id = table2.id AND table1.name = table2.name
```

OR 論理演算子を使用したクエリの例を以下に示します。

```
SELECT some_col, other_col
FROM table1
    JOIN table2
    ON table1.id = table2.id OR table1.name = table2.name
```

コントロール	定義	使用方法
joinColumns	クエリを行えるメンバーに INNER JOIN ステートメントでの使用を許可する列 (ある場合)。	特定の joinColumn は aggregateColumn としても分類されることがあります (「集約コントロール」を参照)。 同じ列を joinColumn と dimensionColumns の両方として使用することはできません (後述)。 aggregateColumn としても分類されていない限り、INNER JOIN 以外のクエリの他のどの部分でも joinColumn を使用することはできません。
joinRequired	クエリを行えるメンバーの設定済みテーブルとの INNER JOIN を必須にするかどうかを制御します。	このパラメータを有効した場合、INNER JOIN は必須になります。このパラメータを有効にしない場合、INNER JOIN はオプションになります。 このパラメータを有効にすると、クエリを行えるメンバーは、自身が所有するテーブル

コントロール	定義	使用方法
		を INNER JOIN に必ず含めなければなりません。クエリを行えるメンバーは、自身のテーブルを相手のテーブルと直接または間接的 (つまり自分のテーブルを、相手のテーブルと結合されている別のテーブルに結合) に JOIN する必要があります。

以下は間接的な結合の例です。

```
ON
my_table.identifer = third_party_table.identifier
....
ON
third_party_table.identifier = member_who_can_query_table.id
```

> **Note**
>
> クエリを行えるメンバーが、joinRequired パラメータを使用することもできます。その場合、クエリで自分のテーブルを少なくとも 1 つの他のテーブルと結合する必要があります。

ディメンションコントロール

ディメンションコントロールは、集約列をフィルタリング、グループ化、または集計する際の基準となる列を制御します。

コントロール	定義	使用方法
dimensionColumns	クエリを行えるメンバーに SELECT、WHERE、GROUP、BY、および ORDER BY で	dimensionColumn は、SELECT (select_grouping_column_expression)、WHERE、G

コントロール	定義	使用方法
	の使用を許可する列 (ある場合)。	ROUP、BY、および ORDER BY で使用できます。 同じ列を `dimensionColumn` と `joinColumn` または `aggregateColumn` の両方として使用することはできません。

スカラー関数

スカラー関数は、どのスカラー関数をディメンション列に使用できるかを制御します。

コントロール	定義	使用方法
`scalarFunctions`	クエリの `dimensionColumns` で使用できるスカラー関数。	`dimensionColumns` での適用を許可するスカラー関数 (CAST など) を指定します (ある場合)。 スカラー関数を他の関数と重ねて使用したり、他の関数内で使用したりすることはできません。スカラー関数の引数には、列、文字列リテラル、または数値リテラルを使用できます。

以下のスカラー関数がサポートされています。

- 数学関数 – ABS、CEILING、FLOOR、LOG、LN、ROUND、SQRT
- データ型フォーマット関数 – CAST, CONVERT, TO_CHAR, TO_DATE, TO_NUMBER, TO_TIMESTAMP
- 文字列関数 – LOWER、UPPER、TRIM、RTRIM、SUBSTRING

  - RTRIM では、カスタム文字セットをトリミングすることはできません。
- 条件式 – COALESCE
- 日付関数 – EXTRACT、GETDATE、CURRENT_DATE、DATEADD
- その他の関数 – TRUNC

詳細については、「AWS Clean Rooms SQL リファレンス」を参照してください。

集計分析ルール - クエリ結果コントロール

集約クエリ結果コントロールでは、返される各出力行が満たす必要がある条件を 1 つ以上指定することで、どの結果を返すかを制御できます。AWS Clean Rooms では、`COUNT（DISTINCT column）>= X` という形式の集約制約をサポートしています。この形式では、設定済みテーブルから、選択した少なくとも X 個の個別値 (例えば、個別の `user_id` 値の最小数) を各行が集約することになります。送信されたクエリ自体が指定された列を使用しない場合でも、この最小数のしきい値は自動的に適用されます。これらは、コラボレーション内の各メンバーの設定済みテーブルから、クエリ内の各設定済みテーブルにまとめて適用されます。

各設定済みテーブルの分析ルールには、少なくとも 1 つの集約制約が必要です。設定済みテーブルの所有者が複数の `columnName` や関連付けられた `minimum` を追加すると、それらはまとめて適用されます。

集約制約

集約制約は、クエリ結果のどの行を返すかを制御します。行が返されるには、その行が、集約制約で指定された各列で、指定された個別値の最小数を満たす必要があります。この要件は、クエリや分析ルールの他の部分にその列が明示的に記述されていない場合でも適用されます。

コントロール	定義	使用方法
`columnName`	各出力行が満たす必要がある条件で使用される `aggregateColumn` です。	設定済みテーブル内のどの列でもかまいません。
`minimum`	クエリ結果で返される出力行に含まれていなければならない、関連付けられた	`minimum` の値は 2 以上にする必要があります。

コントロール	定義	使用方法
	aggregateColumn の個別値の最小数です (COUNT DISTINCT など)。	

集計分析ルールの構造

次の例は、集計分析ルールの事前定義された構造を示しています。

次の例では、*MyTable* がデータテーブルを表しています。各###############は、独自の情報に置き換えることができます。

```
{
  "aggregateColumns": [
    {
      "columnNames": [MyTable column names], "function": [Allowed Agg Functions]
    },
  ],
  "joinRequired": ["QUERY_RUNNER"],
  "joinColumns": [MyTable column names],
  "dimensionColumns": [MyTable column names],
  "scalarFunctions": [Allowed Scalar functions],
  "outputConstraints": [
    {
      "columnName": [MyTable column names], "minimum": [Numeric value]
    },
  ]
}
```

集計分析ルール - 例

次の例は、2 つの企業が AWS Clean Rooms で集計分析を使用してコラボレーションを行う方法を示しています。

A 社には顧客データと売上データがあります。A 社は製品の返品状況を把握したいと考えています。B 社は A 社の小売業者の一社で、返品データを保有しています。B 社には、A 社にとって有益な、顧客に関するセグメント属性 (関連製品を購入した、小売業者からカスタマーサービスを利用したなど) のデータもあります。B 社は、行レベルの顧客返品データや属性情報は提供したくありませ

ん。B 社の希望は、重複する顧客に関する統計情報を最小数の集約しきい値で A 社が取得できるよう、一連のクエリを有効にすることです。

A 社が製品の返品状況を把握し、B 社やその他のチャネルでより良い製品を提供できるように、A 社と B 社は協力することにしました。

コラボレーションを作成して集計分析を行うために、両社は次のことを行います。

1. A 社がコラボレーションを作成し、メンバーシップを作成します。このコラボレーションには、B 社がコラボレーションの相手方メンバーとして参加します。A 社はコラボレーションでのクエリログ記録を有効にし、自社アカウントでのクエリログの記録を有効にします。
2. B 社がコラボレーションでメンバーシップを作成し、そのアカウントでのクエリログの記録を有効にします。
3. A 社が、設定済み売上テーブルを作成します。
4. A 社が、設定済み売上テーブルに次の集計分析ルールを追加します。

```
{
  "aggregateColumns": [
    {
      "columnNames": [
        "identifier"
      ],
      "function": "COUNT_DISTINCT"
    },
    {
      "columnNames": [
        "purchases"
      ],
      "function": "AVG"
    },
    {
      "columnNames": [
        "purchases"
      ],
      "function": "SUM"
    }
  ],
  "joinColumns": [
    "hashedemail"
  ],
  "dimensionColumns": [
```

```
    "demoseg",
    "purchasedate",
    "productline"
  ],
  "scalarFunctions": [
    "CAST",
    "COALESCE",
    "TRUNC"
  ],
  "outputConstraints": [
    {
      "columnName": "hashedemail",
      "minimum": 2,
      "type": "COUNT_DISTINCT"
    },
  ]
}
```

aggregateColumns – A 社は、売上データと返品データで重複している一意の顧客の数をカウントしたいと考えています。また A 社は、purchases の数を合計して、returns の数と比較したいと考えています。

joinColumns – A 社は、identifier を使用して売上データの顧客と返品データの顧客を照合したいと考えています。これにより、A 社は返品を適切な購入と照合できるようになります。また、A 社が重複する顧客をセグメント化するのにも役立ちます。

dimensionColumns – A 社は、dimensionColumns を使用して、特定の製品での絞り込みを行い、一定期間にわたって購入と返品を比較して返品日が購入日付より後になるものを確認し、重複する顧客をセグメント化します。

scalarFunctions – A 社は、A 社がコラボレーションに関連付けた設定済みテーブルに基づいて、必要に応じてデータ型フォーマットを更新できるよう、CAST スカラー関数を選択します。また、必要に応じて列のフォーマットに役立つスカラー関数も追加します。

outputConstraints – A 社は最小数の出力制約を設定します。アナリストは売上テーブルから行レベルのデータを確認できるため、結果を制約する必要はありません。

> **Note**
>
> 企業 A は分析ルールに joinRequired を追加しません。これにより、アナリストは売上テーブルだけに対して柔軟にクエリを実行できます。

5. B 社が、設定済み返品テーブルを作成します。
6. B 社が、設定済み返品テーブルに次の集計分析ルールを追加します。

```
{
  "aggregateColumns": [
    {
      "columnNames": [
        "identifier"
      ],
      "function": "COUNT_DISTINCT"
    },
    {
      "columnNames": [
        "returns"
      ],
      "function": "AVG"
    },
    {
      "columnNames": [
        "returns"
      ],
      "function": "SUM"
    }
  ],
  "joinColumns": [
    "hashedemail"
  ],
  "joinRequired": [
    "QUERY_RUNNER"
  ],
  "dimensionColumns": [
    "state",
    "popularpurchases",
    "customerserviceuser",
    "productline",
    "returndate"
```

```
  ],
  "scalarFunctions": [
    "CAST",
    "LOWER",
    "UPPER",
    "TRUNC"
  ],
  "outputConstraints": [
    {
      "columnName": "hashedemail",
      "minimum": 100,
      "type": "COUNT_DISTINCT"
    },
    {
      "columnName": "producttype",
      "minimum": 2,
      "type": "COUNT_DISTINCT"
    }
  ]
}
```

aggregateColumns – B 社は、A 社が returns の数を合計して購入数と比較できるようにします。集約クエリを有効にしているため、少なくとも 1 つは集約列があります。

joinColumns – B 社は、A 社が返品データの顧客と売上データの顧客を照合できるよう、identifier での結合を有効にします。identifier データは特に機密性が高く、joinColumn として指定すれば、データがクエリで出力されることはありません。

joinRequired – B 社は、返品データと売上データの重複に関するクエリを必須にします。A 社が B 社のデータセット内のすべての個人を照会できるようにはしたくありません。その制限についてはコラボレーション契約でも合意しています。

dimensionColumns – B 社 は、A 社が state、popularpurchases、customerserviceuser の属性でフィルタとグループ化を実行できるようにします。これらは A 社の分析に役立つ固有の属性です。B 社 は、A 社 が returndate を使用して purchasedate の後に発生した returndate で出力をフィルタリングできるようにします。このフィルタリングにより、出力がより正確になり、製品変更の影響を評価できるようになります。

scalarFunctions – B 社は以下を有効にします。

- 日付の TRUNC

- producttype が異なる形式でデータに入力された場合の LOWER と UPPER
- A 社が売上のデータ型を返品のデータ型と同じものに変換する必要がある場合の CAST

A 社は、他のスカラー関数はクエリに必要ではないと考えているため、有効にしていません。

outputConstraints – B 社は、顧客の再識別が困難になるように、hashedemail に最小数の出力制約を設定します。また、返品された特定の製品の再識別が困難になるように、producttype にも最小数の出力制約を設定します。出力のディメンション (state など) によっては、特定の製品タイプがより優勢になる可能性があります 。A 社がデータに追加した出力の制約にかかわらず、B 社の出力の制約が常に適用されます。

7. A 社が、売上テーブルとコラボレーションとの関連付けを作成します。
8. B 社が、返品テーブルとコラボレーションとの関連付けを作成します。
9. A 社が、B 社の返品数を 2022 年の場所別の購入総数と比較して詳しく把握するために、次の例のようなクエリを実行します。

```
SELECT
  companyB.state,
  SUM(companyB.returns),
  COUNT(DISTINCT companyA.hashedemail)
FROM
  sales companyA
  INNER JOIN returns companyB ON companyA.identifier = companyB.identifier
WHERE
  companyA.purchasedate BETWEEN '2022-01-01' AND '2022-12-31' AND
  TRUNC(companyB.returndate) > companyA.purchasedate
GROUP BY
  companyB.state;
```

10. A 社と B 社がクエリログを確認します。B 社は、クエリがコラボレーション契約で合意された内容と一致していることを確認します。

集計分析ルールに関する問題のトラブルシューティング

集計分析ルール使用時の一般的な問題については、ここにある情報を使用して診断と修正を行ってください。

問題

- クエリから何も結果が返されない

クエリから何も結果が返されない

この問題は、一致する結果がない場合や、一致する結果が最小数の集約しきい値を 1 つ以上満たしていない場合に発生する可能性があります。

最小数の集約しきい値の詳細については、「集計分析ルール - 例」を参照してください。

リスト分析ルール

AWS Clean Rooms の リスト分析ルールは、追加先の設定済みテーブルと、クエリを行えるメンバーの設定済みテーブルの間の重複を示す、行レベルのリストを出力します。クエリを行えるメンバーが、リスト分析ルールを含むクエリを実行します。

リスト分析ルールは、エンリッチメントやオーディエンス構築などのユースケースに対応します。

この分析ルールの事前定義されたクエリ構造と構文の詳細については、「リスト分析ルールの事前定義された構造」を参照してください。

リスト分析ルール - クエリコントロール で定義されているリスト分析ルールのパラメータにはクエリコントロールがあります。クエリコントロールには、出力に表示できる列を選択する機能が含まれています。クエリには、直接または間接的にクエリを実行できるメンバーの設定済みテーブルとの結合が少なくとも 1 つ必要です。

集計分析ルールのようなクエリ結果コントロールはありません。

リストクエリでは算術演算子しか使用できません。他の関数 (集約やスカラーなど) は使用できません。

トピック

- リストクエリの構造と構文
- リスト分析ルール - クエリコントロール
- リスト分析ルールの事前定義された構造
- リスト分析ルール - 例

リストクエリの構造と構文

リスト分析ルールが追加されたテーブルに対するクエリは、次の構文に従う必要があります。

```
--select_list_expression
SELECT
[TOP number ] DISTINCT column_name [[AS] column_alias ] [, ...]

--table_expression
FROM table_name [[AS] table_alias ]
  [[INNER] JOIN table_name [[AS] table_alias] ON join_condition] [...]

--where_expression
[WHERE where_condition]

--limit_expression
[LIMIT number]
```

次の表は、前述の構文で示したそれぞれの式について説明しています。

式	定義	例
select_list_expression	テーブル列名を 1 つ以上含むカンマ区切りリストです。 DISTINCT パラメータが必須です。 **Note** select_list_expression では AS パラメータの有無にかかわらず、列にエイリアスを指定できます。TOP パラメータもサポートしています。詳細については、「AWS Clean Rooms SQL リファレンス」を参照してください。	SELECT DISTINCT segment

式	定義	例
table_expression	テーブルまたはテーブルの結合と、それを join_condition に連結するための join_condition 。 join_condition はブール値を返します。 table_expression では、以下がサポートされています。 • 特定の JOIN 型 (INNER JOIN) • join_condition 内の等価比較条件 (=) • 論理演算子 (AND、OR)	FROM consumer_table INNER JOIN provider_table ON consumer_table.identifier1 = provider_table.identifier1 AND consumer_table.identifier2 = provider_table.identifier2

式	定義	例
where_expression	ブール値を返す条件式です。次ような要素で構成されています。 • テーブルの列名 • 算術演算子 • [String literals] (文字列リテラル) • 数値リテラル サポートされている比較条件は (=, >, <, <=, >=, <>, !=, NOT, IN, NOT IN, LIKE, IS NULL, IS NOT NULL) です。 サポートされている論理演算子は (AND, OR) です。 where_expression はオプションです。	WHERE state + '_' + city = 'NY_NYC' WHERE timestampColumn = timestampColumn2 - 14
limit_expression	この式は正の整数にする必要があります。TOP パラメータと交換することもできます。 limit_expression はオプションです。	LIMIT 100

リストクエリの構造と構文については、次の点に注意してください。

- SELECT 以外の SQL コマンドはサポートされていません。
- サブクエリと共通テーブル式 (WITH など) はサポートされていません。
- HAVING、GROUP BY、および ORDER BY 句はサポートされていません。

- OFFSET パラメータはサポートされていません。

リスト分析ルール - クエリコントロール

リストクエリコントロールを使用すると、テーブル内の列を使用してテーブルにクエリを実行する方法を制御できます。例えば、どの列を結合に使用するか、SELECT ステートメントや WHERE 句でどの列を使用できるようにするかを制御できます。

以下のセクションでは、それぞれのコントロールについて説明します。

トピック

結合コントロール

結合コントロールを使用することで、テーブルを table_expression で他のテーブルに結合する方法を制御できます。AWS Clean Rooms は、INNER JOIN のみをサポートしています。リスト分析ルールでは、少なくとも 1 つの INNER JOIN が必要であり、クエリを行えるメンバーは、自身が所有するテーブルを INNER JOIN に必ず含めなければなりません。つまり、クエリを行えるメンバーは、自身のテーブルを相手のテーブルと直接または間接的に結合する必要があります。

以下は間接的な結合の例です。

```
ON
my_table.identifer = third_party_table.identifier
....
ON
third_party_table.identifier = member_who_can_query_table.id
```

INNER JOIN ステートメントでは、分析ルールで明示的に `joinColumn` として分類された列のみを使用できます。

INNER JOIN は、自身の設定済みテーブルの `joinColumn` と、コラボレーション内のもう 1 つの設定済みテーブルの `joinColumn` で動作する必要があります。テーブルのどの列を `joinColumn` として使用できるようにするかはテーブルの所有者が決定します。

ON 句内の一致条件ごとに、2 つの列間の等価比較条件 (=) を使用する必要があります。

ON 句では次のようにして複数の一致条件を使用できます。

- AND 論理演算子を使用して組み合わせる
- OR 論理演算子を使用して区切る

> **Note**
>
> JOIN の一致条件では、必ず JOIN の各側の 1 つの行が一致しなければなりません。OR または AND 論理演算子で接続されるすべての条件がこの要件を満たす必要があります。

AND 論理演算子を使用したクエリの例を以下に示します。

```
SELECT some_col, other_col
FROM table1
    JOIN table2
    ON table1.id = table2.id AND table1.name = table2.name
```

OR 論理演算子を使用したクエリの例を以下に示します。

```
SELECT some_col, other_col
FROM table1
    JOIN table2
    ON table1.id = table2.id OR table1.name = table2.name
```

コントロール	定義	使用方法
joinColumns	クエリを行えるメンバーに INNER JOIN ステートメントでの使用を許可する列。	同じ列を joinColumn と listColumn の両方に分類することはできません (「リストコントロール」を参照)。 joinColumn は INNER JOIN 以外のクエリのどの部分でも使用できません。

リストコントロール

リストコントロールは、クエリ出力 (SELECT ステートメントで使用) または結果のフィルタリング (WHEREステートメントで使用) に使用できる列を制御します。

コントロール	定義	使用方法
`listColumns`	クエリを行えるメンバーに SELECT および WHERE での使用を許可する列。	`listColumn` は SELECT と WHERE で使用できます。 同じ列を `listColumn` と `joinColumn` の両方として使用することはできません。

リスト分析ルールの事前定義された構造

次の例には、リスト分析ルールを完成させる方法を示す事前定義された構造が含まれています。

次の例では、*MyTable* がデータテーブルを表しています。各###############は、独自の情報に置き換えることができます。

```
{
  "joinColumns": [MyTable column name(s)],
  "listColumns": [MyTable column name(s)],
}
```

リスト分析ルール - 例

次の例は、2 つの企業が AWS Clean Rooms でリスト分析を使用してコラボレーションを行う方法を示しています。

A 社には顧客関係管理 (CRM) データがあります。A 社は、顧客についてさらに詳しく知るため、また属性を他の分析のインプットとして使用するために、顧客に関する追加のセグメントデータを取得したいと考えています。B 社には、自社で入手したデータに基づいて作成した独自のセグメント属性で構成されるセグメントデータがあります。B 社は、自社のデータと A 社のデータの間で重複している顧客についてのみ、固有のセグメント属性を A 社に提供したいと考えています。

両社は、A 社が重複するデータのエンリッチメントを行えるよう協力することにしました。A 社はクエリを行えるメンバーで、B 社はデータを寄稿するメンバーです。

コラボレーションを作成し、コラボレーションでリスト分析を実行するために、各社は次のことを行います。

1. A 社がコラボレーションを作成し、メンバーシップを作成します。このコラボレーションには、B 社がコラボレーションの相手方のメンバーとして参加します。A 社はコラボレーションでのクエリログ記録を有効にし、自社アカウントでのクエリログの記録を有効にします。
2. B 社がコラボレーションでメンバーシップを作成し、そのアカウントでのクエリログの記録を有効にします。
3. A 社が、設定済み CRM テーブルを作成します。
4. A 社が、次の例のような分析ルールを設定済み顧客テーブルに追加します。

```
{
  "joinColumns": [
    "identifier1",
    "identifier2"
  ],
  "listColumns": [
    "internalid",
    "segment1",
    "segment2",
    "customercategory"
  ]
}
```

joinColumns – A 社は、hashedemail と thirdpartyid (ID ベンダーから取得) を使用して CRM データの顧客とセグメントデータの顧客を照合したいと考えています。これにより、A 社は適切な顧客のエンリッチメントデータと照合できるようになります。JoinColumns が 2 つあることで、分析の一致率が向上する可能性があります。

listColumns – A 社は、listColumns を使用して、自社のシステム内で使用している internalid に加えてエンリッチメントのための列を取得します。segment1、segment2、customercategory を追加し、フィルタで使用することでエンリッチメントを特定のセグメントに限定できるようにします。

5. B 社が、設定済みセグメントテーブルを作成します。
6. B 社が、分析ルールを設定済みセグメントテーブルに追加します。

```
{
  "joinColumns": [
```

```
    "identifier2"
  ],
  "listColumns": [
    "segment3",
    "segment4"
  ]
}
```

joinColumns – B 社は、A 社が identifier2 で結合を実行して、セグメントデータの顧客を CRM データの顧客と照合できるようにします。A 社と B 社は ID ベンダーと協力して、今回のコラボレーションに適していそうな identifier2 を取得しています。identifier2 は最も一致率が高く、最も正確で、これ以外の ID はクエリに必要ないと考えられたため、他の joinColumns は追加しませんでした。

listColumns – B 社は、自社で作成、収集し、データエンリッチメントに含めるために (A 社とともに) 調整した属性である segment3 および segment4 の属性で、A 社がエンリッチメントを行えるようにします。こうした重複するセグメントを A 社に行レベルで取得してもらうのは、これがデータエンリッチメントのコラボレーションであるためです。

7. A 社が、CRM テーブルとコラボレーションとの関連付けを作成します。
8. B 社が、セグメントテーブルとコラボレーションとの関連付けを作成します。
9. A 社が、次のようなクエリを実行して、重複する顧客データのエンリッチメントを行います。

```
SELECT companyA.internalid, companyB.segment3, companyB.segment4
INNER JOIN returns companyB
 ON companyA.identifier2 = companyB.identifier2
WHERE companyA.customercategory > 'xxx'
```

10. A 社と B 社がクエリログを確認します。B 社は、クエリがコラボレーション契約で合意された内容と一致していることを確認します。

のカスタム分析ルール AWS Clean Rooms

では AWS Clean Rooms、カスタム分析ルールは、設定されたテーブルでカスタムクエリを実行できるようにする新しいタイプの分析ルールです。カスタム SQL クエリは、現在も SELECT コマンドの使用のみに制限されていますが、集約クエリやリストクエリよりも多くの SQL 構文を使用できます (例えば、ウィンドウ関数、OUTER JOIN、CTE、サブクエリなど。詳細なリストについては、「AWS Clean Rooms SQL リファレンス」を参照してください)。カスタム SQL クエリでは、集約クエリやリストクエリのようなクエリ構造に従う必要はありません。

カスタム分析ルールは、カスタムアトリビューション分析、ベンチマーキング、増分解析、オーディエンス発掘など、集計分析ルールやリスト分析ルールよりも高度なユースケースに対応します。これは、集計分析ルールおよびリスト分析ルールでサポートされるユースケースのスーパーセットに追加されるものです。

カスタム分析ルールは差分プライバシーもサポートします。差分プライバシーは、データプライバシー保護のための数学的に厳格なフレームワークです。詳細については、「AWS Clean Rooms 差分プライバシー」を参照してください。分析テンプレートを作成すると、 AWS Clean Rooms 差分プライバシーはテンプレートをチェックして、 AWS Clean Rooms 差分プライバシーの汎用クエリ構造と互換性があるかどうかを確認します。この検証により、差分プライバシー保護対象テーブルでは許可されない分析テンプレートが作成されなくなります。

カスタム分析ルールを設定する場合、データ所有者は、分析テンプレートに保存されている特定のカスタムクエリを自身の設定済みテーブルで実行することを許可できます。データ所有者は、分析テンプレートを確認してから、カスタム分析ルールの許可された分析コントロールに追加します。分析テンプレートは、(テーブルが他のコラボレーションに関連付けられている場合でも) 作成されたコラボレーションでのみ使用および表示され、そのコラボレーションでクエリを行えるメンバーのみが実行できます。

または、他のメンバー (クエリプロバイダー) が確認なしでクエリを作成できるよう許可することもできます。その場合は、許可対象のクエリプロバイダーが管理するクエリプロバイダーのアカウントをカスタム分析ルールに追加します。クエリプロバイダーがクエリを行えるメンバーであれば、設定済みテーブルで任意のクエリを直接実行できます。クエリプロバイダーは、分析テンプレートを作成することによってクエリを作成することもできます。クエリプロバイダーによって作成されたクエリは、 が存在し、テーブル AWS アカウント が関連付けられているすべてのコラボレーションで、テーブルで自動的に実行できるようになります。

データ所有者がクエリの作成を許可できるのは分析テンプレートまたはアカウントのみで、両方には許可できません。データ所有者がこれを空欄のままにすると、クエリを行えるメンバーは設定済みテーブルでクエリを実行できません。

トピック

- カスタム分析ルールの事前定義された構造
- カスタム分析ルールの例
- 差分プライバシーによるカスタム分析ルール

カスタム分析ルールの事前定義された構造

次の例には、差分プライバシーが有効なカスタム分析ルールを完成させる方法を示す、事前定義された構造が含まれています。userIdentifier 値は user_id のような、ユーザーを一意に識別する列です。コラボレーションで差分プライバシーが有効になっているテーブルが 2 つ以上ある場合、AWS Clean Rooms では、テーブル間でユーザーの一貫した定義を維持するために、両方の分析ルールでユーザー識別子列と同じ列を設定する必要があります。

```
{
  "allowedAnalyses": ["ANY_QUERY"] | string[],
  "allowedAnalysisProviders": [],
  "differentialPrivacy": {
    "columns": [
      {
        "name": "userIdentifier"
      }
    ]
  }
}
```

次のいずれかを行うことができます。

- 分析テンプレート ARN を許可された分析コントロールに追加します。この場合、allowedAnalysisProviders コントロールは含まれません。

  ```
  {
    allowedAnalyses: string[]
  }
  ```

- allowedAnalysisProviders コントロール AWS アカウント IDs を追加します。この場合は、ANY_QUERY を allowedAnalyses コントロールに追加します。

  ```
  {
    allowedAnalyses: ["ANY_QUERY"],
    allowedAnalysisProviders: string[]
  }
  ```

カスタム分析ルールの例

次の例は、2 つの企業がカスタム分析ルール AWS Clean Rooms を使用して でコラボレーションする方法を示しています。

A 社には顧客データと売上データがあります。A 社は、B 社のサイトで実施した広告キャンペーンによる売上の増分を把握したいと考えています。B 社には、A 社にとって有用な閲覧データとセグメント属性 (広告を閲覧する際に使用したデバイスなど) があります。

企業 A には、コラボレーションで実行したい特定の増分クエリがあります。

コラボレーションを作成し、コラボレーションでカスタム分析を実行するために、各社は次のことを行います。

1. A 社がコラボレーションを作成し、メンバーシップを作成します。このコラボレーションには、B 社がコラボレーションの相手方のメンバーとして参加します。A 社はコラボレーションでのクエリログ記録を有効にし、自社アカウントでのクエリログの記録を有効にします。
2. B 社がコラボレーションでメンバーシップを作成し、そのアカウントでのクエリログの記録を有効にします。
3. A 社が、設定済み CRM テーブルを作成します。
4. A 社が、設定済み売上テーブルに空のカスタム分析ルールを追加します。
5. A 社が、設定済み売上テーブルをコラボレーションに関連付けます。
6. B 社が、設定済み閲覧者数テーブルを作成します。
7. B 社が、設定済み閲覧者数テーブルに空のカスタム分析ルールを追加します。
8. B 社が、設定済み閲覧者数テーブルをコラボレーションに関連付けます。
9. A 社が、コラボレーションに関連付けられている売上テーブルと閲覧者数テーブルを表示し、キャンペーン月の増分クエリとパラメータを追加して分析テンプレートを作成します。

```
{
    "analysisParameters": [
    {
        "defaultValue": ""
        "type": "DATE"
        "name": "campaign_month"
    }
    ],
    "description": "Monthly incrementality query using sales and viewership data"
    "format": "SQL"
    "name": "Incrementality analysis"
```

```
    "source":
        "WITH labeleddata AS
        (
        SELECT hashedemail, deviceid, purchases, unitprice, purchasedate,
        CASE
            WHEN testvalue IN ('value1', 'value2', 'value3') THEN 0
            ELSE 1
        END AS testgroup
        FROM viewershipdata
        )
        SELECT labeleddata.purchases, provider.impressions
        FROM labeleddata
        INNER JOIN salesdata
          ON labeleddata.hashedemail = provider.hashedemail
        WHERE MONTH(labeleddata.purchasedate) > :campaignmonth
        AND testgroup = :group
       "
}
```

10A 社は、カスタム分析ルールで許可された分析プロバイダーコントロールにアカウント (例: 444455556666) を追加します。A 社は、作成したすべてのクエリを設定済み販売テーブルで実行できるようにしたいため、許可された分析プロバイダーコントロールを使用しています。

```
{
  "allowedAnalyses": [
    "ANY_QUERY"
  ],
  "allowedAnalysisProviders": [
    "444455556666"
  ]
}
```

11B 社が、作成された分析テンプレートをコラボレーション内で確認し、クエリ文字列やパラメータなどの内容を確認します。

12B 社が、分析テンプレートが増分のユースケースに対応しており、設定済み視聴者数テーブルのクエリの実行方法がプライバシー要件を満たしていることを判断します。

13B 社が、視聴者数テーブルのカスタム分析ルールの許可された分析コントロールに分析テンプレート ARN を追加します。B 社は、設定済み視聴者数テーブルでのみ増分クエリを実行できるようにしたいため、許可された分析コントロールを使用しています。

```
{
```

```
  "allowedAnalyses": [
    "arn:aws:cleanrooms:us-east-1:111122223333:membership/41327cc4-bbf0-43f1-b70c-
a160dddceb08/analysistemplate/1ff1bf9d-781c-418d-a6ac-2b80c09d6292"
  ]
}
```

14A 社が分析テンプレートを実行し、パラメータ値 `05-01-2023` を使用します。

差分プライバシーによるカスタム分析ルール

では AWS Clean Rooms、カスタム分析ルールは差分プライバシーをサポートしています。差分プライバシーは、データプライバシー保護のための数学的に厳格なフレームワークであり、データを再識別されないように保護するのに役立ちます。

差分プライバシーは、広告キャンペーンの計画、 post-ad-campaign 測定、金融機関コンソーシアムでのベンチマーク、医療研究のための A/B テストなどの集計分析をサポートします。

サポートされているクエリ構造と構文は、「クエリの構造と構文」で定義されています。

差分プライバシーを使用したカスタム分析ルールの例

前のセクションで説明したカスタム分析ルールの例を考えてみます。この例は、差分プライバシーを使用して再識別の試みからデータを保護すると同時に、パートナーがデータからビジネスクリティカルなインサイトを学べるようにする方法を示しています。閲覧者データを保有している B 社が、差分プライバシーを使用してデータを保護したいと想定します。差分プライバシーの設定を完了するために、B 社は次のステップを実行します。

1. B 社は差分プライバシーを有効にし、設定済み閲覧者数テーブルにカスタム分析ルールを追加します。B 社は `viewershipdata.hashedemail` をユーザー識別子列として選択します。
2. B 社はコラボレーションに差分プライバシーポリシーを追加し、閲覧者データテーブルをクエリに使用できるようにします。B 社はデフォルトポリシーを選択し、設定を迅速に完了します。

A社 は、B 社のサイトでの広告キャンペーンの売上増分を把握したいと考えており、分析テンプレートを実行します。このクエリは AWS Clean Rooms 差分プライバシーの汎用クエリ構造と互換性があるため、クエリは正常に実行されます。

クエリの構造と構文

差分プライバシーが有効になっているテーブルが 1 つ以上含まれているクエリは、次の構文に従う必要があります。

```
query_statement:
    [cte, ...] final_select

 cte:
    WITH sub_query AS (
       inner_select
       [ UNION | INTERSECT | UNION_ALL | EXCEPT/MINUS ]
       [ inner_select ]
    )

 inner_select:
     SELECT [user_id_column, ] expression [, ...]
     FROM table_reference [, ...]
     [ WHERE condition ]
     [ GROUP BY user_id_column[, expression] [, ...] ]
     [ HAVING condition ]

 final_select:
     SELECT [expression, ...] | COUNT | COUNT_DISTINCT | SUM | AVG | STDDEV
     FROM table_reference [, ...]
     [ WHERE condition ]
     [ GROUP BY expression [, ...] ]
     [ HAVING COUNT | COUNT_DISTINCT | SUM | AVG | STDDEV | condition ]
     [ ORDER BY column_list ASC | DESC ]
     [ OFFSET literal ]
     [ LIMIT literal ]

 expression:
    column_name [, ...] | expression AS alias | aggregation_functions |
window_functions_on_user_id | scalar_function | CASE | column_name math_expression [,
expression]

 window_functions_on_user_id:
    function () OVER (PARTITION BY user_id_column, [column_name] [ORDER BY column_list
 ASC|DESC])
```

> **Note**
>
> 差分プライバシークエリの構造と構文については、次の点に注意してください。
>
> - サブクエリはサポートされていません。

- テーブルや CTE に差分プライバシーで保護されたデータが含まれている場合は、テーブル共通式 (CTE) がユーザー識別子列を出力する必要があります。フィルター、グループ化、集計はユーザーレベルで行う必要があります。
- Final_select では、COUNT DISTINCT、COUNT、SUM、AVG、STDDEV 集約関数が使用できます。

差分プライバシーでサポートされる SQL キーワードの詳細については、「差分プライバシーの SQL AWS Clean Rooms 機能」を参照してください。

AWS Clean Rooms 差分プライバシー

AWS Clean Rooms 差分プライバシーは、数回のクリックで直感的なコントロールで実装される数学に基づいた手法でユーザーのプライバシーを保護するのに役立ちます。フルマネージド機能として、ユーザーの再識別を防ぐために、差分プライバシーエクスペリエンスは必要ありません。 は、個々のレベルのデータを保護するために、実行時にクエリ結果に慎重に調整された量のノイズ AWS Clean Rooms を自動的に追加します。

AWS Clean Rooms 差分プライバシーは、幅広い分析クエリをサポートし、クエリ結果のわずかなエラーが分析の有用性を損なうことのない、さまざまなユースケースに適しています。これにより、パートナーは広告キャンペーン、投資決定、臨床研究などについて、ビジネスに不可欠なインサイトを生成できます。しかも、パートナーによる追加設定は必要ありません。

AWS Clean Rooms 差分プライバシーは、スカラー関数や数学演算子記号を悪意のある方法で使用するオーバーフローや無効なキャストエラーから保護します。

AWS Clean Rooms 差分プライバシーの詳細については、以下のトピックを参照してください。

トピック

差分プライバシー

差分プライバシーでは、集約されたインサイトのみが許可され、それらのインサイトにおける個人のデータの関与をわかりにくくします。差分プライバシーは、特定の個人について学習した結果を受け取ることができるメンバーからコラボレーションデータを保護します。差分プライバシーがなければ、結果を受け取ることができるメンバーは、個人に関するレコードを追加または削除したり、クエリ結果の違いを確認したりして、個々のユーザーデータを推測しようとする可能性があります。

差分プライバシーをオンにすると、特定の量のノイズがクエリ結果に追加され、個々のユーザーの関与がわかりにくくなります。結果を受け取ることができるメンバーが、データセットから個人に関す

るレコードを削除した後にクエリ結果の違いを観察しようとすると、クエリ結果の変動性により、個人のデータの特定を防ぐことができます。 AWS Clean Rooms 差分プライバシーは、 によって開発された実証済みの正しいサンプラー実装である SampCert サンプラーを使用します AWS。

での差分プライバシーの AWS Clean Rooms 仕組み

で差分プライバシーを有効にするワークフロー AWS Clean Rooms では、 のワークフローを完了するときに、以下の追加ステップが必要です AWS Clean Rooms。

1. カスタム分析ルールを追加するときに、差分プライバシーを有効にします。
2. コラボレーションの差分プライバシーポリシーを設定して、差分プライバシーで保護されているデータテーブルをクエリに使用できるようにします。

これらのステップを完了すると、クエリを行えるメンバーは、差分プライバシー保護データに対するクエリの実行を開始できます。 は、差分プライバシーポリシーに準拠した結果を AWS Clean Rooms 返します。 AWS Clean Rooms 差分プライバシーは、車の現在の燃料レベルを示す車内のガスゲージと同様に、実行できる残りのクエリの推定数を追跡します。クエリを実行できるメンバーが実行できるクエリの数は、差分プライバシーポリシー で設定されている [プライバシー予算] と [クエリごとに追加されるノイズ] パラメータによって制限されます。

考慮事項

で差分プライバシーを使用する場合は AWS Clean Rooms、次の点を考慮してください。

- 結果を受け取ることができるメンバーは、差分プライバシーを使用できません。これらのメンバーは、設定したテーブルの差分プライバシーをオフにしたカスタム分析ルールを設定します。
- クエリを実行できるメンバーは、2 つ以上のデータプロバイダーの差分プライバシーがオンになっている場合、テーブルを結合できません。

差分プライバシーポリシー

差分プライバシーポリシーは、クエリを実行できるメンバーがコラボレーション内で実行できる集計関数の数を制御します。プライバシー予算は、コラボレーションのすべてのテーブルに適用される共通の有限リソースを定義します。クエリごとに追加されるノイズによって、プライバシー予算が枯渇する速度が決まります。

差分プライバシーで保護されたテーブルをクエリに使用できるようにするには、差分プライバシーポリシーが必要です。これはコラボレーションの 1 回限りのステップで、次の 2 つの入力が含まれます。

- プライバシー予算 – プライバシー予算はイプシロン単位で定量化され、プライバシー保護のレベルを制御します。これは、コラボレーションにおいて差分プライバシーで保護されるすべてのテーブルに適用される共通かつ有限のリソースです。その目的は、情報が複数のテーブルに存在する可能性があるユーザーのプライバシーを保護することです。

  プライバシー予算は、テーブルに対してクエリが実行されるたびに消費されます。プライバシー予算が使い果たされると、クエリを実行できるコラボレーションメンバーは、プライバシー予算が増えるか更新されるまで追加のクエリを実行できなくなります。プライバシー予算を大きく設定することで、結果を受け取ることができるメンバーは、データ内の個人に関する不確実性を減らすことができます。ビジネス上の意思決定者と相談した上で、コラボレーション要件とプライバシーニーズのバランスを考慮して、プライバシー予算を選択してください。

  コラボレーションに定期的に新しいデータを取り込む予定がある場合は、[プライバシー予算を毎月更新] を選択すると、1 か月ごとに新しいプライバシー予算が自動的に作成されます。このオプションを選択すると、更新を繰り返してクエリを繰り返し実行したときに、データ行に関する任意の量の情報を公開できます。プライバシー予算が更新されるたびに同じ行が繰り返しクエリされる場合は、このオプションを選択しないでください。
- クエリごとに追加されるノイズは、関与をわかりにくくするユーザーの数で測定されます。この値によって、プライバシー予算がどの程度枯渇するかが決まります。ノイズ値を大きくすると、プライバシー予算が枯渇する速度が下がるため、データに対して実行できるクエリの数が増えます。ただし、これは正確性の低いデータインサイトの公開とのバランスを取る必要があります。この値を設定するときは、コラボレーションに関するインサイトに必要な精度を考慮してください。

デフォルトの差分プライバシーポリシーを使用して、セットアップを迅速に完了したり、ユースケースに応じて差分プライバシーポリシーをカスタマイズしたりできます。 AWS Clean Rooms 差分プライバシーは、ポリシーを設定する直感的な制御を提供します。 AWS Clean Rooms 差分プライバシーを使用すると、データに対するすべてのクエリで可能な集約数の観点からユーティリティをプレビューし、データコラボレーションで実行できるクエリの数を推定できます。

インタラクティブな例を見れば、[プライバシー予算] と [クエリごとに追加されるノイズ] の値が異なると、さまざまな種類の SQL クエリの結果にどのような影響があるかを理解できます。一般的には、プライバシーに関するニーズと、許可するクエリの数やそれらのクエリの精度とのバランスを取る必要があります。[プライバシー予算] を小さくしたり、[クエリごとに追加されるノイズ] を大きく

したりすると、ユーザーのプライバシー保護は強化されますが、コラボレーションパートナーにとって意味のあるインサイトは得られません。

[クエリごとに追加されるノイズ] のパラメータをそのままにして、[プライバシー予算] を増やすと、クエリを実行できるメンバーは、コラボレーション内のテーブルに対してより多くの集計を実行できます。コラボレーション中はいつでも [プライバシー予算] を増やすことができます。[クエリごとに追加されるノイズ] パラメータをそのままにして、[プライバシー予算] を減らすと、クエリを実行できるメンバーが実行できる集計の数が減ります。クエリを実行できるメンバーがデータの分析を開始した後は、[プライバシー予算] を減らすことはできません。

[プライバシー予算] の入力をそのままにして、[クエリごとに追加されるノイズ] を増やすと、クエリを実行できるメンバーは、コラボレーション内のテーブルに対してより多くの集計を実行できます。[プライバシー予算] の入力をそのままにして、[クエリごとに追加されるノイズ] を減らすと、クエリを実行できるメンバーが実行できる集計の数が減ります。[クエリごとに追加されるノイズ] は、コラボレーション中いつでも増減できます。

差分プライバシーポリシーは、プライバシー予算テンプレート API アクションによって管理されます。

差分プライバシーの SQL AWS Clean Rooms 機能

AWS Clean Rooms 差分プライバシーは、汎用クエリ構造を使用して複雑な SQL クエリをサポートします。カスタム分析テンプレートはこの構造に対して検証され、差分プライバシーで保護されたテーブルで実行できるようにします。次の表は、どの関数がサポートされているかを示しています。詳細については、「クエリの構造と構文」を参照してください。

短縮名	SQL コンストラクト	テーブル共通式 (CTE)	最終 SELECT 句
集計関数	• ANY_VALUE 関数 • APPROXIMATE PERCENTILE_DISC 関数 • AVG 関数 • COUNT および COUNT DISTINCT 関数 • LISTAGG 関数	差分プライバシー保護テーブルを使用する CTEsがユーザーレベルのレコードを持つデータを生成する必要があるという条件でサポートされています。SELECT 式は、`SELECT userIdent	サポートされている集計: AVG、COUNT、COUNT DISTINCT、STDDEV、SUM。

短縮名	SQL コンストラクト	テーブル共通式 (CTE)	最終 SELECT 句
	• MAX 関数 • MEDIAN 関数 • MIN 関数 • PERCENTIL E_CONT 関数 • STDDEV_SAMP および STDDEV_POP 関数 • SUM および SUM DISTINCT 関数 • VAR_SAMP および VAR_POP 関数	`ifierColu mn...'` 形式を使用してそれらの CTEsで記述する必要があります。	
CTEs	WITH 句、WITH 句サブクエリ	差分プライバシー保護テーブルを使用する CTEsがユーザーレベルのレコードを持つデータを生成する必要があるという条件でサポートされています。SELECT 式は、`` `SELECT userIdent ifierColu mn...' `` 形式を使用してそれらの CTEsで記述する必要があります。	該当なし
サブクエリ	SELECT リストサブクエリ、FROM 句サブクエリ、WHERE 句サブクエリ	サポート外。差分プライバシーが有効になっているテーブルを参照するクエリのサブクエリはサポートされていません。サブクエリを共通テーブル式 (CTEs。	

短縮名	SQL コンストラクト	テーブル共通式 (CTE)	最終 SELECT 句
Join 句	• INNER JOIN • LEFT JOIN • RIGHT JOIN • FULL JOIN • [JOIN] OR 演算子 • CROSS JOIN	ユーザー識別子列の等価結合である JOIN 関数のみがサポートされ、差分プライバシーが有効になっている複数のテーブルをクエリする場合は必須であるという条件でサポートされます。必須の等価結合条件が正しいことを確認してください。テーブル所有者がすべてのテーブルに同じユーザー ID 列を設定して、ユーザーの定義がテーブル間で一貫していることを確認します。 差分プライバシーを有効にして 2 つ以上のリレーションを組み合わせる場合、CROSS JOIN 関数はサポートされません。	
セット演算子	UNION、UNION ALL、INTERSECT、EXCEPT | MINUS (これらはシノニムです)	すべてサポートされています	サポートされていません

短縮名	SQL コンストラクト	テーブル共通式 (CTE)	最終 SELECT 句
Window 関数	集計関数 • AVG ウィンドウ関数 • COUNT ウィンドウ関数 • CUME_DIST ウィンドウ関数 • DENSE_RANK ウィンドウ関数 • FIRST_VALUE ウィンドウ関数 • LAG ウィンドウ関数 • LAST_VALUE ウィンドウ関数 • LEAD ウィンドウ関数 • MAX ウィンドウ関数 • MEDIAN ウィンドウ関数 • MIN ウィンドウ関数 • NTH_VALUE ウィンドウ関数 • RATIO_TO_REPORT ウィンドウ関数 • STDDEV_SAMP および STDDEV_POP ウィンドウ関数	差分プライバシーがオンになっている関係をクエリする場合、ウィンドウ関数のパーティション句のユーザー識別子列が必要であるという条件で、すべてのがサポートされます。	サポートされていません

短縮名	SQL コンストラクト	テーブル共通式 (CTE)	最終 SELECT 句
	(STDDEV_SAMP および STDDEV はシノニムです) • SUM ウィンドウ関数 • VAR_SAMP および VAR_POP ウィンドウ関数 (VAR_SAMP および VARIANCE はシノニムです) ランク付け関数 • DENSE_RANK ウィンドウ関数 • NTILE ウィンドウ関数 • PERCENT_RANK ウィンドウ関数 • RANK ウィンドウ関数 • ROW_NUMBER ウィンドウ関数		
条件式	• CASE 条件式 • COALESCE 式 • GREATEST および LEAST 関数 • NVL および COALESCE 関数 • NVL2 関数 • NULLIF 関数	すべてサポートされています	すべてサポートされています

短縮名	SQL コンストラクト	テーブル共通式 (CTE)	最終 SELECT 句
条件	• 比較条件 • 論理条件 • パターンマッチング条件 • BETWEEN 範囲条件 • Null 条件	EXISTS および IN はサブクエリが必要なため使用できません。その他はすべてサポートされています。	すべてサポートされています
日時関数	• トランザクションにおける日付および時刻関数 • 連結演算子 • ADD_MONTHS 関数 • CONVERT_TIMEZONE 関数 • CURRENT_DATE 関数 • DATEADD 関数 • DATEDIFF 関数 • DATE_PART 関数 • DATE_TRUNC 関数 • EXTRACT 関数 • GETDATE 関数 • TIMEOFDAY 関数 • TO_TIMESTAMP 関数 • 日付関数またはタイムスタンプ関数の日付部分	すべてサポートされています	すべてサポートされています

短縮名	SQL コンストラクト	テーブル共通式 (CTE)	最終 SELECT 句
文字列関数	• || (連結) 演算子 • BTRIM 関数 • CHAR_LENGTH 関数 • CHARACTER _LENGTH 関数 • CHARINDEX 関数 • CONCAT 関数 • LEFT 関数および RIGHT 関数 • LEN 関数 • LENGTH 関数 • LOWER 関数 • LPAD 関数および RPAD 関数 • LTRIM 関数 • POSITION 関数 • REGEXP_COUNT 関数 • REGEXP_INSTR 関数 • REGEXP_RE PLACE 関数 • REGEXP_SUBSTR 関数 • REPEAT 関数 • REPLACE 関数 • REPLICATE 関数 • REVERSE 関数 • RTRIM 関数	すべてサポートされています	すべてサポートされています

短縮名	SQL コンストラクト	テーブル共通式 (CTE)	最終 SELECT 句
	• SOUNDEX 関数 • SPLIT_PART 関数 • STRPOS 関数 • SUBSTRING 関数 • TEXTLEN 関数 • TRANSLATE 関数 • TRIM 関数 • UPPER 関数		
データ型フォーマット関数	• CAST 関数 • TO_CHAR • TO_DATE 関数 • TO_NUMBER • 日時形式の文字列 • 数値形式の文字列	すべてサポートされています	すべてサポートされています
ハッシュ関数	• MD5 関数 • SHA 関数 • SHA1 関数 • SHA2 関数 • MURMUR3_32_HASH	すべてサポートされています	すべてサポートされています
数学演算子の記号	+、-、*、/、%、@	すべてサポートされています	すべてサポートされています

短縮名	SQL コンストラクト	テーブル共通式 (CTE)	最終 SELECT 句
数学関数	• ABS 関数 • ACOS 関数 • ASIN 関数 • ATAN 関数 • ATAN2 関数 • CBRT 関数 • CEILING (または CEIL) 関数 • COS 関数 • COT 関数 • DEGREES 関数 • DEXP 関数 • LTRIM 関数 • DLOG1 関数 • DLOG10 関数 • EXP 関数 • FLOOR 関数 • LN 関数 • LOG 関数 • MOD 関数 • PI 関数 • POWER 関数 • RADIANS 関数 • RANDOM 関数 • ROUND 関数 • SIGN 関数 • SIN 関数 • SQRT 関数 • TRUNC 関数	すべてサポートされています	すべてサポートされています

短縮名	SQL コンストラクト	テーブル共通式 (CTE)	最終 SELECT 句
SUPER 型の情報関数	• DECIMAL_PRECISION 関数 • DECIMAL_SCALE 関数 • IS_ARRAY 関数 • IS_BIGINT 関数 • IS_CHAR 関数 • IS_DECIMAL 関数 • IS_FLOAT 関数 • IS_INTEGER 関数 • IS_OBJECT 関数 • IS_SCALAR 関数 • IS_SMALLINT 関数 • IS_VARCHAR 関数 • JSON_TYPEOF 関数	すべてサポートされています	すべてサポートされています
VARBYTE 関数	• FROM_HEX 関数 • FROM_VARBYTE 関数 • TO_HEX 関数 • TO_VARBYTE 関数	すべてサポートされています	すべてサポートされています

短縮名	SQL コンストラクト	テーブル共通式 (CTE)	最終 SELECT 句
JSON	• CAN_JSON_PARSE 関数 • JSON_EXTRACT_ARRAY_ELEMENT_TEXT 関数 • JSON_EXTRACT_PATH_TEXT 関数 • JSON_PARSE 関数 • JSON_SERIALIZE 関数 • JSON_SERALIZE_TO_VARBYTE 関数	すべてサポートされています	すべてサポートされています
配列関数	• array 関数 • array_concat 関数 • array_flatten 関数 • get_array_length 関数 • split_to_array 関数 • subarray 関数	サポートされません	サポートされません
拡張 GROUP BY	グループ化セット、ロールアップ、立方体	サポートされません	サポートされません

短縮名	SQL コンストラクト	テーブル共通式 (CTE)	最終 SELECT 句
ソートオペレーション	ORDER BY	ORDER BY 句は、差分プライバシーがオンになっているテーブルをクエリするときに、ウィンドウ関数のパーティション句でのみサポートされるという条件でサポートされます。	サポート
行の制限	制限、オフセット	差分プライバシー保護テーブルを使用する CTEsではサポートされていません	すべてサポートされています
テーブルと列のエイリアス		サポート	サポート
集計関数の数学関数		サポート	サポート
集計関数内のスカラー関数		サポート	サポート

サポートされていない SQL コンストラクトの一般的な代替方法

カテゴリ	SQL コンストラクト	代替
Window 関数	• LISTAGG • PERCENTILE_CONT • PERCENTILE_DISC	GROUP BY では同等の集計関数を使用できます。
数学演算子の記号	• $column ||/ 2 • $column |/ 2 • $column ^ 2	• CBRT • SQRT • POWER($column, 2)

カテゴリ	SQL コンストラクト	代替
スカラー関数	• SYSDATE • $column::integer • convert(type, $column)	• CURRENT_DATE • CAST $column AS integer • CAST $column AS type
リテラル	INTERVAL '1 SECOND'	間隔「1」秒
行の制限	トップ n	制限 n
Join	• USING • NATURAL	ON 句には結合条件を明示的に含める必要があります。

差分プライバシークエリのヒントと例

AWS Clean Rooms 差分プライバシーは、汎用クエリ構造を使用して、データ準備のための共通テーブル式 (CTEs) や、 や などの一般的に使用される集計関数などCOUNT、さまざまな SQL コンストラクトをサポートしますSUM。実行時にクエリ結果を集計するためのノイズを追加して、データ内の可能なユーザーの寄与を難読化するために、 AWS Clean Rooms 差分プライバシーでは、最終的な集計関数SELECT statementをユーザーレベルのデータで実行する必要があります。

次の例では、athletic_brand_sales データを持つスポーツブランドとのコラボレーションで差分プライバシーを使用してデータを保護したいと考えているメディアパブリッシャーの socialco_impressions と socialco_users という名前の 2 つのテーブルを使用しています。メディアパブリッシャーは、 AWS Clean Roomsで差分プライバシーを有効にして、user_id 列をユーザー識別子列として設定しています。広告主は差分プライバシー保護を必要としないため、組み合わせたデータに対して CTE を使用してクエリを実行したいと考えています。CTE では差分プライバシーで保護されたテーブルを使用しているため、広告主は保護されているテーブルのユーザー ID 列を CTE 列のリストに含め、保護対象のテーブルをユーザー ID 列に結合します。

```
WITH matches_table AS(
     SELECT si.user_id, si.campaign_id, s.sale_id, s.sale_price
     FROM socialco_impressions si
     JOIN socialco_users su
         ON su.user_id = si.user_id
     JOIN athletic_brand_sales s
         ON s.emailsha256 = su.emailsha256
     WHERE s.timestamp > si.timestamp
```

```
UNION ALL

     SELECT si.user_id, si.campaign_id, s.sale_id, s.sale_price
     FROM socialco_impressions si
     JOIN socialco_users su
         ON su.user_id = si.user_id
     JOIN athletic_brand_sales s
         ON s.phonesha256 = su.phonesha256
     WHERE s.timestamp > si.timestamp
)

SELECT COUNT (DISTINCT user_id) as unique_users
FROM matches_table
GROUP BY campaign_id
ORDER BY COUNT (DISTINCT user_id) DESC
LIMIT 5
```

同様に、差分プライバシーで保護されたデータテーブルでウィンドウ関数を実行する場合は、PARTITION BY 句にユーザー ID 列を含める必要があります。

```
ROW_NUMBER() OVER (PARTITION BY conversion_id, user_id ORDER BY match_type, match_age)
 AS row
```

AWS Clean Rooms 差分プライバシーの制限事項

AWS Clean Rooms 差分プライバシーは、以下の状況には対応していません。

1. AWS Clean Rooms 差分プライバシーはタイミング攻撃に対処しません。例えば、これらの攻撃は、個々のユーザーが大量の行を入力し、このユーザーを追加または削除するとクエリの計算時間が大幅に変化するようなシナリオで発生する可能性があります。
2. AWS Clean Rooms 差分プライバシーは、特定の SQL コンストラクトの使用により、SQL クエリの実行時にオーバーフローまたは無効なキャストエラーが発生する可能性がある場合、差分プライバシーを保証しません。次の表は、ランタイムエラーが発生する可能性があるため、分析テンプレートで検証する必要がある SQL コンストラクトの一部のリストです。ただし、すべてではありません。このようなランタイムエラーの可能性を最小限に抑える分析テンプレートを承認し、クエリログを定期的に確認して、クエリがコラボレーション契約と整合しているかどうかを確認することをお勧めします。

   次の SQL コンストラクトは、オーバーフローエラーに対して脆弱です。

- 集計関数 - AVG、LISTAVG、PERCENTILE_COUNT、PERCENTILE_DISC、SUM/SUM_DISTINCT
- データ型の書式設定関数 - TO_TIMESTAMP、TO_DATE
- 日付と時刻の関数 - ADD_MONTHS、DATEADD、DATEDIFF
- 数学関数 - +、-、*、/、POWER
- 文字列関数 - ||、CONCAT、REPEAT、REPLICATE
- ウィンドウ関数 - AVG、LISTAGG、PERCENTILE_COUNT、PERCENTILE_DISC、RATIO_TO_REPORT、SUM

CAST データ型フォーマット関数は、無効なキャストエラーに対して脆弱です。

AWS Clean Rooms ML

AWS Clean Rooms ML

AWS Clean Rooms ML は、データを相互に共有しなくても、データの類似ユーザーを識別するためのプライバシー保護方法を提供します。ファーストパーティはトレーニングデータを に持ち込んで、類似モデルを作成して設定し、コラボレーションに関連付ける AWS Clean Rooms ことができます。次に、セカンドパーティはシードデータを に取り込み、トレーニングデータに似た類似セグメント AWS Clean Rooms を生成します。

この動作の詳細な説明については、「クロスアカウントジョブ」を参照してください。

- トレーニングデータプロバイダー – トレーニングデータを提供し、類似モデルを作成および設定し、その類似モデルをコラボレーションに関連付ける関係者。
- シードデータプロバイダー – シードデータを提供し、類似セグメントを生成し、その類似セグメントをエクスポートする関係者。
- トレーニングデータ – 類似モデルの生成に使用されるトレーニングデータプロバイダーのデータ。トレーニングデータは、ユーザーの行動の類似性を測定するために使用されます。

  トレーニングデータには、ユーザー ID、項目 ID、タイムスタンプ列が含まれている必要があります。オプションで、トレーニングデータには数値特徴量またはカテゴリ別特徴量として他のインタラクションを含めることができます。インタラクションの例としては、視聴した動画のリスト、購入したアイテム、読んだ記事などがあります。
- シードデータ – 類似セグメントの作成に使用されるシードデータプロバイダーのデータ。類似セグメントの出力は、トレーニングデータに含まれるシードユーザーに最も近いユーザーの集合です。
- 類似モデル – 他のデータセット内の類似ユーザーを見つけるために使用されるトレーニングデータの機械学習モデル。

  API を使用する場合、オーディエンスモデルという用語は類似モデルと同じ意味で使用されます。例えば、CreateAudienceModel API を使用して類似モデルを作成します。
- 類似セグメント — シードデータに最も近いトレーニングデータのサブセット。

  API を使用する場合は、 StartAudienceGenerationJob API を使用して類似セグメントを作成します。

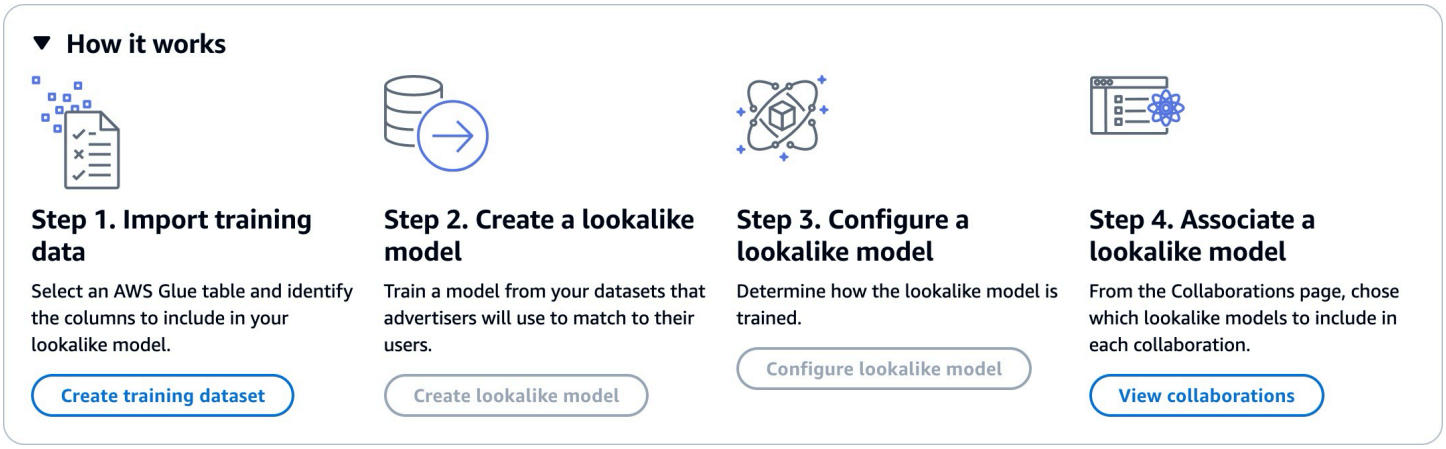
トレーニングデータプロバイダーのデータがシードデータプロバイダーと共有されることはなく、シードデータプロバイダーのデータがトレーニングデータプロバイダーと共有されることもありません。類似セグメントの出力はトレーニングデータプロバイダーと共有されますが、シードデータプロバイダーと共有されることはありません。

類似モデルの使用の詳細については、次のトピックを参照してください。

トピック

- AWS Clean Rooms ML の仕組み

AWS Clean Rooms ML の仕組み



Clean Rooms ML では、トレーニングデータプロバイダーとシードデータプロバイダーの 2 つの関係者が順番に で作業 AWS Clean Rooms して、データをコラボレーションに取り込む必要があります。トレーニングデータプロバイダーが最初に完了しなければならないワークフローは次のとおりです。

1. トレーニングデータプロバイダーのデータは、ユーザーアイテムインタラクション AWS Glue のデータカタログテーブルに保存する必要があります。少なくとも、トレーニングデータにはユーザー ID 列、インタラクション ID 列、タイムスタンプ列が含まれている必要があります。
2. トレーニングデータプロバイダーは、トレーニングデータを に登録します AWS Clean Rooms。
3. トレーニングデータプロバイダーは、複数のシードデータプロバイダーと共有できる類似モデルを作成します。類似モデルはディープニューラルネットワークであり、トレーニングに最大 24 時間かかることがあります。このモデルは自動的に再トレーニングされないため、毎週再トレーニングすることをお勧めします。

4. トレーニングデータプロバイダーは、関連メトリクスを共有するかどうかや、出力セグメントの Amazon S3 ロケーションなど、類似モデルの設定を行います。トレーニングデータプロバイダーは、1 つの類似モデルから複数の設定済み類似モデルを作成できます。
5. トレーニングデータプロバイダーは、設定したオーディエンスモデルをシードデータプロバイダーと共有されているコラボレーションに関連付けます。

シードデータプロバイダーが次に完了しなければならないワークフローは次のとおりです。

1. シードデータプロバイダーのデータは Amazon S3 バケットに保存されている必要があります。
2. シードデータプロバイダーは、トレーニングデータプロバイダーと共有するコラボレーションを開きます。
3. シードデータプロバイダーは、コラボレーションページの Clean Rooms ML タブから類似セグメントを作成します。
4. シードデータプロバイダーは、関連性メトリクスが共有されていれば、それを評価し、 AWS Clean Roomsの外部で使用するために類似セグメントをエクスポートできます。

AWS Clean Rooms ML のプライバシー保護

Clean Rooms ML は、トレーニングデータプロバイダーがシードデータ内の誰であるかを学習でき、シードデータプロバイダーがトレーニングデータ内の誰であるかを学習できる、メンバーシップ推論攻撃のリスクを軽減するように設計されています。この攻撃を防ぐためにいくつかの対策が講じられています。

まず、シードデータプロバイダーは Clean Rooms ML 出力を直接監視せず、トレーニングデータプロバイダーはシードデータを監視できません。シードデータプロバイダーは、シードデータを出力セグメントに含めることもできます。

次に、トレーニングデータのランダムなサンプルから類似モデルを作成します。このサンプルには、シードオーディエンスと一致しない多数のユーザーが含まれています。このプロセスにより、ユーザーがデータに含まれていないかどうかを判断することが難しくなります。これは、メンバーシップの推論のもう 1 つの手段です。

さらに、シード固有の類似モデルトレーニングの各パラメータに複数のシードカスタマーを使用できます。これにより、モデルがどれだけオーバーフィットできるかが制限され、ユーザーについてどれだけ推測できるかが制限されます。この結果、シードデータの最小サイズは 500 ユーザーとすることをお勧めします。

最後に、ユーザーレベルのメトリクスはトレーニングデータプロバイダーには提供されないため、メンバーシップ推論攻撃を受ける別の手段がなくなります。

AWS Clean Rooms ML モデル評価メトリクス

Clean Rooms ML はリコールと関連性のスコアを計算して、モデルのパフォーマンスを判断します。再現率は、類似データとトレーニングデータの類似性を比較します。関連性スコアは、モデルのパフォーマンスが良いかどうかではなく、オーディエンスの大きさを決定するために使用されます。

再現率は、類似セグメントがトレーニングデータとどの程度類似しているかを示すバイアスのない尺度です。リコールは、オーディエンス生成ジョブによってシードオーディエンスに含まれるトレーニングデータのサンプルからの最も類似したユーザーの割合 (デフォルトでは最も類似した 20%) です。値の範囲は 0～1 で、値が大きいほど対象者が高いことを示します。リコール値が最大ビンパーセンテージとほぼ等しい場合は、オーディエンスモデルがランダム選択と同等であることを示します。

Clean Rooms ML はモデルの構築時に真陰性ユーザーを正確にラベル付けしていないため、これは精度、精度、F1 スコアよりも評価メトリクスが良いと考えています。

セグメントレベルの関連性スコアは、-1 (最も類似しない) から 1 (最も類似する) までの値を持つ類似性の尺度です。Clean Rooms ML は、さまざまなセグメントサイズの関連性スコアのセットを計算し、データに最適なセグメントサイズを決定するのに役立ちます。関連性スコアは、セグメントサイズが大きくなるにつれて単調に減少するため、セグメントサイズが大きくなるにつれてシードデータと類似しなくなる可能性があります。セグメントレベルの関連性スコアが 0 に達すると、モデルは類似セグメントのすべてのユーザーがシードデータと同じディストリビューションに属すると予測します。出力サイズを大きくすると、類似セグメントに、シードデータと同じディストリビューションに属さないユーザーが含まれる可能性が高くなります。

関連性スコアは 1 つのキャンペーン内で標準化されるため、キャンペーン間の比較には使用しないでください。関連性スコアは、関連性に加えて複数の複雑な要因 (インベントリの品質、インベントリタイプ、広告のタイミングなど) の影響を受けるため、ビジネス成果の単一の情報源として使用すべきではありません。

関連性スコアはシードの品質を判断するためではなく、増減できるかどうかを判断するために使うべきです。次の例を考えます。

- すべて正のスコア – 類似していると予測される出力ユーザーの方が、類似セグメントに含まれるユーザーよりも多いことを示しています。これは、過去 1 か月間に歯磨き粉を購入したユーザー

など、大規模な市場に属するシードデータによく見られます。過去 1 か月に歯磨き粉を複数回購入したユーザーなど、比較的小さなシードデータを確認することをお勧めします。

- すべての負のスコアまたは希望する類似セグメントサイズに対する負のスコア — これは、クリーンルーム ML が、希望する類似セグメントサイズに十分な類似ユーザーがないと予測していることを示します。これは、シードデータが具体的すぎるか、市場が小さすぎることが原因と考えられます。シードデータに適用するフィルターの数を減らすか、市場を拡大することをお勧めします。例えば、元のシードデータがベビーカーとチャイルドシートを購入した顧客だった場合、ベビー用品を複数購入した顧客に市場を拡大できます。

トレーニングデータプロバイダーは、関連性スコアを公開するかどうか、および関連性スコアを計算するバケットビンを決定します。

AWS Clean Rooms ML の使用

類似モデルとは、トレーニングデータプロバイダーのデータのモデルです。これにより、シードデータプロバイダーは、トレーニングデータプロバイダーのデータのうち、シードデータに最も近い類似セグメントを作成できます。コラボレーションで使用できる類似モデルを作成するには、トレーニングデータをインポートし、類似モデルを作成し、その類似モデルを設定して、コラボレーションに関連付ける必要があります。

トレーニングデータプロバイダーが ML モデルの作成を完了すると、シードデータプロバイダーはシードセグメントを作成してエクスポートできます。

トピック

類似モデルの操作 (トレーニングデータプロバイダー)

トレーニングデータをインポートする

類似モデルを作成する前に、トレーニングデータを含む AWS Glue テーブルを指定する必要があります。Clean Rooms ML は、このデータのコピーを保存せず、データへのアクセスを許可するメタデータのみを保存します。

でトレーニングデータをインポートするには AWS Clean Rooms

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. 左のナビゲーションペインで [ML モデリング] を選択します。
3. [トレーニングデータセット] タブで [トレーニングデータセットを作成] を選択します。
4. [名前] と [説明] (オプション) を入力します。
5. データソース で、 AWS Glue テーブルを選択します。

   a. 設定する [データベース] をドロップダウンリストから選択します。

   b. ドロップダウンリストから設定するデータベースとテーブルを選択して、トレーニングデータソースを選択します。

   > **Note**
   >
   > テーブルが正しいことを確認するには、次のいずれかの操作を行います。
   >
   > - で表示を選択します AWS Glue。
   > - [スキーマを表示] をオンにして、スキーマを表示します。

6. トレーニングの詳細 で、データからユーザー識別子列 、項目識別子列 、タイムスタンプ列を選択します。トレーニングデータには、これら 3 つの列が含まれている必要があります。また、トレーニングデータに含める他の列を選択できます。

   Timestamp 列のデータは、Unix エポック時間を秒単位で指定する必要があります。

7. サービスアクセス では、データにアクセスできるサービスロールを指定し、データが暗号化されている場合は KMS キーを指定する必要があります。新しいサービスロールを作成して使用するを選択すると、Clean Rooms ML は自動的にサービスロールを作成し、必要なアクセス許可ポリシーを追加します。使用する特定のサービスロールがある場合は、「既存のサービスロールを使用する」を選択し、「サービスロール名」フィールドに入力します。

   データが暗号化されている場合は、 AWS KMS keyフィールドに KMS キーを入力するか、 の作成 AWS KMS key をクリックして新しい KMS キーを生成します。

8. トレーニングデータセットで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。
9. [トレーニングデータセットを作成] を選択します。

対応する API アクションについては、CreateTraining「データセット」を参照してください。

類似モデルの作成

トレーニングデータセットを作成したら、類似モデルを作成する準備が整います。1 つのトレーニングデータセットから多数の類似モデルを作成できます。

でデフォルトのデータベースを作成する AWS Glue Data Catalog か、指定されたロールに アクセス`glue:createDatabase`許可を含める必要があります。

で類似モデルを作成するには AWS Clean Rooms

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. 左のナビゲーションペインで [ML モデリング] を選択します。
3. [そっくりなモデル] タブで、[類似モデルの作成] を選択します。
4. [類似モデルの作成] の、[類似モデルの詳細] で以下を行います。
   a. [名前] と [説明] (オプション) を入力します。
   b. モデル化する [トレーニングデータセット] をドロップダウンリストから選択します。
   c. [トレーニングウィンドウ] (オプション) を入力します。
5. 類似モデルのカスタム暗号化設定を有効にする場合は、[暗号化の設定をカスタマイズ] を選択し、KMS キーを入力します。
6. 類似モデルで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。
7. [類似モデルの作成] を選択します。

対応する API アクションについては、CreateAudience「モデル」を参照してください。

類似モデルの設定

類似モデルを作成したら、コラボレーションで使用するように設定する準備が整います。1 つの類似モデルから、複数の設定済みの類似モデルを作成できます。

で類似モデルを設定するには AWS Clean Rooms

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。

2. 左のナビゲーションペインで [ML モデリング] を選択します。
3. [設定済みの類似モデル] タブで、[類似モデルの設定] を選択します。
4. [類似モデルの設定] の、[設定済みの類似モデルの詳細] で以下を行います。

   a. [名前] と [説明] (オプション) を入力します。

   b. 設定する [類似モデル] をドロップダウンリストから選択します。

   c. 希望する [マッチングシードサイズの最小値] を選択します。これは、シードデータプロバイダーのデータに含まれるユーザーのうち、トレーニングデータ内のユーザーと重複するユーザーの最小数です。この値は 0 より大きい必要があります。

5. [他のメンバーと共有するメトリクス] では、コラボレーションのシードデータプロバイダーに関連性スコアを含むモデルメトリクスを受信させるかどうかを選択します。
6. [類似セグメントの送信先ロケーション] には、類似セグメントがエクスポートされる Amazon S3 バケットを入力します。このバケットは、他のリソースと同じリージョンに配置する必要があります。
7. [サービスアクセス] では、このテーブルへのアクセスに使用する [既存のサービスロール名] を選択します。
8. 類似モデルの設定 を選択します。
9. 設定済みテーブルのリソースで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。

対応する API アクションについては、「」を参照してくださいCreateConfiguredAudienceModel。

設定済みの類似モデルを関連付ける

類似モデルを設定したら、それをコラボレーションに関連付けることができます。

で設定された類似モデルを関連付けるには AWS Clean Rooms

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きますAWS アカウント （ まだ開いていない場合 ）。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. [アクティブメンバーシップあり] タブでコラボレーションを選択します。
4. [ML モデリング] タブで、[アソシエイトそっくりモデル] を選択します。
5. [設定済みの類似モデルを関連付ける] の、[アソシエイトそっくりモデルの詳細] で以下を行います。

a. 関連する設定済みオーディエンスモデルの [名前] を入力します。

b. テーブルの [説明] を入力します。

この説明は、似たような名前を持つ他の関連する設定済みオーディエンスモデルと区別するのに役立ちます。

6. [設定済みの類似モデル] では、ドロップダウンリストから設定済みの類似モデルを選択します。
7. [関連付ける] を選択します。

対応する API アクションについては、CreateConfiguredAudienceModel「関連付け」を参照してください。

設定された類似モデルを更新する

類似モデルを関連付けた後、そのモデルを更新して、名前、共有するメトリクス、Amazon S3 の場所の出力などの情報を変更できます。

で関連する設定済み類似モデルを更新するには AWS Clean Rooms

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. 左側のナビゲーションペインで、ML モデリング を選択します。
3. 「類似モデルの設定」タブで、設定された類似モデルを選択し、「 の編集」を選択します。
4. [類似モデルの設定] の、[設定済みの類似モデルの詳細] で以下を行います。

a. ドロップダウンリストから、設定する類似モデルを選択します。

b. 希望する [マッチングシードサイズの最小値] を選択します。これは、シードデータプロバイダーのデータに含まれるユーザーのうち、トレーニングデータ内のユーザーと重複するユーザーの最小数です。この値は 0 より大きい必要があります。

5. [他のメンバーと共有するメトリクス] では、コラボレーションのシードデータプロバイダーに関連性スコアを含むモデルメトリクスを受信させるかどうかを選択します。
6. [類似セグメントの送信先ロケーション] には、類似セグメントがエクスポートされる Amazon S3 バケットを入力します。このバケットは、他のリソースと同じリージョンに配置する必要があります。
7. [サービスアクセス] では、このテーブルへのアクセスに使用する [既存のサービスロール名] を選択します。

8. 高度なビンサイズ設定 で、オーディエンスのビンサイズを設定する方法を選択します。
9. [変更の保存] を選択します。

対応する API アクションについては、「」を参照してくださいUpdateConfiguredAudienceModel。

類似セグメントの操作 (シードデータプロバイダー)

類似セグメントの作成

類似セグメントは、シードデータに最も近いトレーニングデータのサブセットです。

で類似セグメントを作成するには AWS Clean Rooms

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. [アクティブメンバーシップあり] タブでコラボレーションを選択します。
4. [ML モデリング] タブで [類似セグメントの作成] を選択します。
5. [類似セグメントの作成] で、[類似セグメントの詳細] に [名前] と [説明] (オプション) を入力します。
6. [シードプロファイル] では、シードデータが保存されている [Amazon S3 入力ソース] を選択します。
7. [サービスアクセス] では、このテーブルへのアクセスに使用する [既存のサービスロール名] を選択します。
8. トレーニングデータセットで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。
9. [類似セグメントの作成] を選択します。

対応する API アクションについては、「」を参照してくださいStartAudienceGenerationJob。

類似セグメントをエクスポートする

類似セグメントを作成したら、そのデータを Amazon S3 バケットにエクスポートできます。

で類似セグメントをエクスポートするには AWS Clean Rooms

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. [アクティブメンバーシップあり] タブでコラボレーションを選択します。
4. [ML モデリング] タブで類似セグメントを選択し、[エクスポート] を選択します。
5. [類似モデルのエクスポート] で、[類似モデルの詳細のエクスポート] に [名前] と [説明] (オプション) を入力します。
6. [セグメントサイズ] では、エクスポートするセグメントのサイズを選択します。
7. [エクスポート] をクリックします。

対応する API アクションについては、「」を参照してくださいStartAudienceExportJob。

次のステップ

類似モデルを作成し、シードセグメントをエクスポートする準備が整いました。

- 管理 AWS Clean Rooms

Cryptographic Computing for Clean Rooms

Cryptographic Computing for Clean Rooms (C3R) は、分析ルール に加えて AWS Clean Rooms 使用できる の機能です。 ???C3R を使うと、組織は機密データをまとめて、データ分析から新しいインサイトを引き出せると同時に、プロセスの中で関係者が知ることのできる事項を暗号化によって制限できます。C3R は、コラボレーションで機密データを扱いたいが、クラウド内の暗号化されたデータのみを使用する必要がある複数の関係者が使用できます。

C3R 暗号化クライアントは、 で使用するデータを暗号化するために使用できるクライアント側の暗号化ツールです AWS Clean Rooms。C3R 暗号化クライアントを使用すると、データは、 AWS Clean Rooms のコラボレーションで使用している間、暗号によって保護されたままになります。通常の AWS Clean Rooms コラボレーションと同様に、入力データはリレーショナルデータベーステーブルであり、計算は SQL クエリとして表されます。ただし、暗号化されたデータに対する SQL クエリについて、C3R は一部の限られたクエリしかサポートしていません。

具体的には、暗号で保護されたデータに対する SQL JOIN および SELECT のステートメントをサポートしています。入力テーブルの各列は、次の SQL ステートメントタイプのいずれか 1 つだけで使用できます。

- JOIN ステートメントで使用される、暗号化によって保護された列はfingerprint列と呼ばれます。
- SELECT ステートメントで使用される、暗号化によって保護された列はsealed列と呼ばれます。
- JOIN または SELECT ステートメントで使用される、暗号化によって保護されていない列はcleartext列と呼ばれます。

場合によっては、GROUP BY ステートメントがfingerprint列でサポートされることもあります。詳細については、「Fingerprint列」を参照してください。現在 C3R では、関連する分析ルールで許可されている場合でも、WHERE 句、または SUM や AVERAGE といった集約関数など、他の SQL 構文の暗号化済みデータへの使用はサポートしていません。

C3R はテーブルの個々のセル内のデータを保護するように設計されています。C3R のデフォルト設定を使用すると、お客様がコラボレーションを通じてサードパーティに提供する基データは、コンテンツが AWS Clean Rooms内部で使用されている間は暗号化されたままになります。C3R では、すべてのsealed列に業界標準の AES-GCM 暗号化を使用し、fingerprint列の保護には Hash-based Message Authentication Code (HMAC) と呼ばれる業界標準の疑似ランダム機能を使用します。

C3R はテーブル内のデータを暗号化しますが、以下の情報は推測できる場合があります。

- テーブルの列数、列名、行数など、テーブル自体に関する情報。
- ほとんどの標準的な暗号化形式と同様、C3R では、暗号化された値の長さについては隠蔽を試みません。C3R には、暗号化された値をパディングしてクリアテキストの正確な長さを隠蔽する機能がありますが、各列のクリアテキストの長さの上限は、第三者に把握される可能性があります。
- 暗号化された C3R テーブルに特定の行がいつ追加されたかなど、ログレベルの情報。

C3R の詳細については、以下のトピックを参照してください。

トピック

- Cryptographic Computing for Clean Rooms を使用する際の考慮事項
- Cryptographic Computing for Clean Rooms でサポートされているファイルとデータの種類
- Cryptographic Computing for Clean Rooms での列名
- Cryptographic Computing for Clean Rooms での列タイプ
- 暗号コンピューティングパラメータ
- Cryptographic Computing for Clean Rooms のオプションフラグ
- クエリと Cryptographic Computing for Clean Rooms
- C3R 暗号化クライアントのガイドライン

Cryptographic Computing for Clean Rooms を使用する際の考慮事項

Cryptographic Computing for Clean Rooms (C3R) は、データ保護を最大限に強化することを目的としています。ただし、一部のユースケースでは、追加機能と引き換えにデータの保護レベルを下げることでメリットが得られる場合があります。こうした特定のトレードオフは、C3R を最も安全な設定から変更することで実現できます。ユーザーは、これらのトレードオフを認識し、それが自身のユースケースに適しているかどうかを判断する必要があります。考慮すべきトレードオフは次のとおりです。

トピック

- テーブル内でcleartextデータと暗号化データの混在を許可する
- fingerprint列で値の繰り返しを許可する
- fingerprint列の命名方法に関する制限を緩和する

- NULL 値の表現方法を決定する

これらのシナリオでのパラメータの使用方法の詳細については、「暗号コンピューティングパラメータ」を参照してください。

テーブル内でcleartextデータと暗号化データの混在を許可する

すべてのデータをクライアント側で暗号化することで、最大限のデータ保護が可能になります。ただし、これにより特定の種類のクエリ (SUM 集約関数など) が制限されます。cleartextデータを許可することのリスクは、暗号化されたテーブルにアクセスできる人なら誰でも暗号化された値に関する情報を推測できるようになることです。これは、cleartextおよび関連するデータを統計的に分析することで可能になります。

例えば、`City` と `State` の列があるとします。`City` 列はcleartextで、`State` 列は暗号化されています。`City` 列の値 `Chicago` を見れば、`State` の値が`Illinois` であることを高い確率で特定できます。逆に、一方の列が `City` で、もう一方の列が `EmailAddress` の場合、暗号化されている `EmailAddress` の情報がcleartext `City` によって明らかになることはまずありません。

このシナリオのパラメータの詳細については、「[cleartext 列を許可] パラメータ」を参照してください。

fingerprint列で値の繰り返しを許可する

最も安全な方法では、どのfingerprint列にも変数のインスタンスが 1 つだけ含まれていると想定されてます。1 つのfingerprint列で項目を繰り返すことはできません。C3R 暗号化クライアントは、これらのcleartext値をランダム値と見分けがつかない一意の値にマッピングします。したがって、これらのランダム値からcleartextに関する情報を推測することはできません。

fingerprint列で値が繰り返される場合のリスクは、値が繰り返されるとランダムに見える値も繰り返されることです。したがって理論的には、暗号化されたテーブルにアクセスできる人なら誰でもfingerprint列の統計分析を行って、cleartext値に関する情報を明らかにできる可能性があります。

例えば、fingerprint列が `State` で、テーブルのすべての行が米国の世帯に対応しているとします。頻度分析を行うことで、どの州が `California` でどの州が `Wyoming` であるかを高い確率で推測できます。この推測が可能なのは、`California` の方が `Wyoming` よりも住民の数がはるかに多いからです。逆に、fingerprint列が世帯識別子に関するもので、数百万件のエントリからなるデータベースの中で各世帯が 1 ~ 4 回出現したとします。この場合、頻度分析によって有用な情報が明らかになることはまずありません。

このシナリオのパラメータの詳細については、「[複製を許可] パラメータ」を参照してください。

fingerprint列の命名方法に関する制限を緩和する

デフォルトでは、暗号化されたfingerprint列を使用して 2 つのテーブルが結合される場合、各テーブルのそれらの列は名前が同じであると想定されます。この結果の技術的な理由は、デフォルトでは、各fingerprint列を暗号化するために異なる暗号キーを派生させるからです。このキーは、コラボレーションの共有シークレットキーと列名の組み合わせから派生します。列名の異なる 2 つの列を結合しようとすると、異なるキーが派生し、有効な結合を処理できません。

この問題に対処するには、各列名からキーを派生させる機能をオフにします。すると、C3R 暗号化クライアントはすべてのfingerprint列に 1 つの派生キーを使用します。リスクは、別の種類の頻度分析を行うと、情報が明らかになる可能性があることです。

もう一度 `City` と `State` の例を使ってみましょう。各fingerprint列で同じランダム値を派生させる (列名を組み込まない) 場合、`New York` では `City` 列と `State` 列のランダム値が同じになります。ニューヨークは、米国でも数少ない、`City` と `State` の名前が同じ都市の 1 つです。逆に、データセットの各列の値がまったく異なれば、情報が漏れることはありません。

このシナリオのパラメータの詳細については、「[名前の異なる列の JOIN を許可] パラメータ」を参照してください。

NULL 値の表現方法を決定する

選択肢は、NULL 値を他の値と同様に暗号化処理 (暗号化と HMAC) するかどうかです。NULL 値を他の値と同様に処理しないと、情報が漏洩する可能性があります。

例えば、`Middle Name` 列のcleartextの NULL が、ミドルネームのない人を示しているとします。これらの値を暗号化しないと、暗号化されたテーブルのどの行がミドルネームを持たない人に使われているかが漏れてしまいます。そうした情報は、一部の集団の一部の人々にとっては、ある種の識別信号となる可能性があります。しかし、NULL 値を暗号化処理すると、特定の SQL クエリが異なる動作になります。例えば GROUP BY 句では、fingerprint列内のfingerprint NULL 値がまとめてグループ化されなくなります。

このシナリオのパラメータの詳細については、「[NULL 値を保存] パラメータ」を参照してください。

Cryptographic Computing for Clean Rooms でサポートされているファイルとデータの種類

C3R 暗号化クライアントは、以下の種類のファイルを認識します。

- CSV ファイル
- Parquet ファイル

C3R 暗号化クライアントの `--fileFormat` フラグを使用して、ファイル形式を明示的に指定できます。明示的に指定した場合、ファイル形式はファイル拡張子によって判断されません。

トピック

CSV ファイル

.csv 拡張子の付いたファイルは CSV 形式で、UTF-8 でエンコードされたテキストを含むものとみなされます。C3R 暗号化クライアントはすべての値を文字列として扱います。

.csv ファイルでサポートされるプロパティ

C3R 暗号化クライアントでは、.csv ファイルに次のプロパティが必要です。

- 各列に一意の名前を付ける最初のヘッダー行 (含まれる場合と含まれない場合があります)。
- カンマ区切り (現在、カスタム区切り文字はサポートされていません)。
- UTF-8 でエンコードされたテキスト。

.csv エントリからの空白の削除

.csv エントリから先頭および末尾の空白が両方とも削除されます。

.csv ファイルのカスタム NULL エンコーディング

.csv ファイルではカスタム NULL エンコーディングを使用できます。

C3R 暗号化クライアントでは、`--csvInputNULLValue=<csv-input-null>` フラグを使用して入力データの NULL エントリにカスタムエンコーディングを指定できます。C3R 暗号化クライアントは、`--csvOutputNULLValue=<csv-output-null>` フラグを使用することで、生成された出力ファイル内の NULL エントリに対してカスタムエンコーディングを使用できます。

> **Note**
>
> 特に SQL テーブルのようなよりリッチな表形式のコンテキストでは、NULL エントリは欠落したコンテンツとみなされます。.csv は歴史的な理由からこの特性を明示的にサポートしていませんが、空白だけを含む空のエントリは NULL と見なすのが一般的な慣習です。したがって、これは C3R 暗号化クライアントのデフォルト動作であり、必要に応じてカスタマイズできます。

C3R による .csv エントリの解釈

次の表は、`--csvInputNULLValue=<csv-input-null>` および `--csvOutputNULLValue=<csv-output-null>` のフラグに指定された値 (存在する場合) に基づいて、.csv エントリを (わかりやすくするために cleartext から cleartext に) 整列化する方法の例を示しています。引用符の外側にある先頭と末尾の空白は、C3R が値の意味を解釈する前に削除されます。

<csv-input-null>	**<csv-output-null>**	入力エントリ	出力エントリ
なし	なし	`,AnyProduct,`	`,AnyProduct,`
なし	なし	`, AnyProduct ,`	`,AnyProduct,`
なし	なし	`,"AnyProduct",`	`,AnyProduct,`
なし	なし	`, "AnyProduct" ,`	`,AnyProduct,`
なし	なし	`,,`	`,,`
なし	なし	`, ,`	`,,`
なし	なし	`,"",`	`,,`
なし	なし	`," ",`	`," ",`
なし	なし	`, " " ,`	`," ",`
`"AnyProduct"`	`"NULL"`	`,AnyProduct,`	`,NULL,`

<csv-input-null>	**<csv-output-null>**	入力エントリ	出力エントリ
"AnyProduct"	"NULL"	, AnyProduct ,	,NULL,
"AnyProduct"	"NULL"	,"AnyProduct",	,NULL,
"AnyProduct"	"NULL"	, "AnyProduct" ,	,NULL,
なし	"NULL"	,,	,NULL,
なし	"NULL"	, ,	,NULL,
なし	"NULL"	,"",	,NULL,
なし	"NULL"	," ",	," ",
なし	"NULL"	, " " ,	," ",
""	"NULL"	,,	,NULL,
""	"NULL"	, ,	,NULL,
""	"NULL"	,"",	,"",
""	"NULL"	," ",	," ",
""	"NULL"	, " " ,	," ",
"\"\""	"NULL"	,,	,,
"\"\""	"NULL"	, ,	,,
"\"\""	"NULL"	,"",	,NULL,
"\"\""	"NULL"	," ",	," ",
"\"\""	"NULL"	, " " ,	," ",

ヘッダーのない CSV ファイル

ソースの .csv ファイルでは、各列に一意の名前を付けるヘッダーが最初の行になくてもかまいません。ただし、ヘッダー行のない .csv ファイルには位置暗号化スキーマが必要です。ヘッダー行のある .csv ファイルと Parquet ファイルの両方に使用される一般的なマッピングスキーマの代わりに、位置暗号化スキーマが必要になります。

位置暗号化スキーマは、出力列を名前ではなく位置で指定します。マッピング暗号化スキーマは、ソースの列名をターゲットの列名にマッピングします。両方のスキーマ形式の詳細な説明や例については、「マッピングテーブルスキーマと位置テーブルスキーマ」を参照してください。

Parquet ファイル

.parquet 拡張子の付いたファイルは、Apache Parquet 形式であるとみなされます。

サポートされている Parquet データ型

C3R 暗号化クライアントでは、 AWS Clean Roomsでサポートされているデータ型を表す Parquet ファイル内の複雑でない (つまり、プリミティブ型の) データを処理できます。

ただし、sealed列には文字列の列しか使用できません。

以下の Parquet データ型 がサポートされています。

- 以下の論理アノテーション付きの `Binary` プリミティブ型
  - `--parquetBinaryAsString` が設定されている場合は不要 (`STRING` データ型)
  - `Decimal(scale, precision)` (`DECIMAL` データ型)
  - `String` (`STRING` データ型)
- 論理アノテーションのない `Boolean` プリミティブデータ型 (`BOOLEAN` データ型)
- 論理アノテーションのない `Double` プリミティブデータ型 (`DOUBLE` データ型)
- `Decimal(scale, precision)` 論理アノテーション付きの `Fixed_Len_Binary_Array` プリミティブ型 (`DECIMAL` データ型)
- 論理アノテーションのない `Float` プリミティブデータ型 (`FLOAT` データ型)
- 以下の論理アノテーション付きの `Int32` プリミティブ型
  - 不要 (`INT` データ型)
  - `Date` (`DATE` データ型)

  - Decimal(scale, precision) (DECIMAL データ型)
  - Int(16, true) (SMALLINT データ型)
  - Int(32, true) (INT データ型)
- 以下の論理アノテーション付きの Int64 プリミティブデータ型
  - 不要 (BIGINT データ型)
  - Decimal(scale, precision) (DECIMAL データ型)
  - Int(64, true) (BIGINT データ型)
  - Timestamp(isUTCAdjusted, TimeUnit.MILLIS) (TIMESTAMP データ型)
  - Timestamp(isUTCAdjusted, TimeUnit.MICROS) (TIMESTAMP データ型)
  - Timestamp(isUTCAdjusted, TimeUnit.NANOS) (TIMESTAMP データ型)

文字列以外の値の暗号化

現在、sealed列では文字列値のみがサポートされています。

.csv ファイルの場合、C3R 暗号化クライアントはすべての値を UTF-8 でエンコードされたテキストとして扱い、暗号化前にそれらを異なる方法で解釈しようとはしません。

フィンガープリント列では、データ型は等価クラスにグループ化されます。等価クラスは、代表的なデータ型を使用して同等かどうかを明確に比較できるデータ型のセットです。

等価クラスを使用すると、元の表現に関係なく、同一のフィンガープリントを同じセマンティック値に割り当てることができます。ただし、2 つの等価クラスで同じ値があっても、同じフィンガープリント列にはなりません。

例えば、42 の INTEGRAL 値には、元の値が SMALLINT、INT、または BIGINT であったかどうかにかかわらず、同じフィンガープリントが割り当てられます。また、0 の INTEGRAL 値が FALSE の BOOLEAN 値 (値 0 で表される) と一致することはありません。

フィンガープリント列では、次の等価クラスと対応する AWS Clean Rooms データ型がサポートされています。

等価クラス	サポートされている AWS Clean Rooms データ型
BOOLEAN	BOOLEAN

等価クラス	サポートされている AWS Clean Rooms データ型
DATE	DATE
INTEGRAL	BIGINT, INT, SMALLINT
STRING	CHAR, STRING, VARCHAR

Cryptographic Computing for Clean Rooms での列名

デフォルトでは、Cryptographic Computing for Clean Rooms においては列の名前が重要になります。

[名前の異なる列の JOIN を許可] パラメータの値が [false] の場合は、fingerprint列の暗号化時に列名が使用されます。このためデフォルトでは、コラボレーターが事前に調整して、クエリで JOIN ステートメントを使用するデータで同じターゲット列名を使用する必要があります。デフォルトでは、JOIN のために暗号化された名前の異なる列は、どの値でも JOIN が正常に処理されません。

[名前の異なる列の JOIN を許可] パラメータの値が [true] の場合、fingerprint列として暗号化された複数の列にわたる JOIN ステートメントは成功します。このパラメータを使用してデータを暗号化すると、cleartext値をある程度推測できる可能性があります。例えば、ある行の `City` 列と `State` 列の両方に同じ Hash-based Message Authentication Code (HMAC) 値がある場合、その値は `New York` である可能性があります。

列ヘッダー名の正規化

列ヘッダー名は C3R 暗号化クライアントによって正規化されます。変換後の出力では、先頭と末尾のスペースはすべて削除され、列名は小文字になります。

正規化は、列名の影響を受ける可能性のある他のすべての計算や操作の前に適用されます。出力されるファイルには、正規化された名前のみが含まれます。

Cryptographic Computing for Clean Rooms での列タイプ

このトピックでは、Cryptographic Computing for Clean Rooms での列タイプに関する情報を提供します。

トピック

Fingerprint列

Fingerprint列は、JOIN ステートメントで使用される、暗号化によって保護された列です。

fingerprint列のデータを復号化することはできません。復号化できるのは、シール列のデータだけです。

Fingerprint列は次の SQL 句と関数でのみ使用する必要があります。

- 他のfingerprint列に対する JOIN (INNER, OUTER, LEFT, RIGHT, or FULL)
  - `allowJoinsOnColumnsWithDifferentNames` パラメータの値が `false` に設定されている場合、JOIN の両方のfingerprint列が同じ名前であることも必要になります。
- `SELECT COUNT()`
- `SELECT COUNT(DISTINCT )`
- `GROUP BY` (コラボレーションで `preserveNulls` パラメータの値が `true` に設定されている場合にのみ使用)

これらの制約に違反するクエリは、誤った結果をもたらす可能性があります。

シール列

シール列は、SELECT ステートメントで使用される、暗号化によって保護された列です。

シール列は、次の SQL 句と関数でのみ使用する必要があります。

- `SELECT`
- `SELECT ... AS`
- `SELECT COUNT()`

> **Note**
>
> `SELECT COUNT(DISTINCT )` はサポートされていません。

これらの制約に違反するクエリは、誤った結果をもたらす可能性があります。

暗号化前のsealed列のデータのパディング

列をsealed列にするように指定すると、C3R からどの種類のパディングを選択するかをたずねられます。暗号化前のデータのパディングは任意です。パディングを使用しない場合 (パディングタイプ none) は、暗号化されたデータの長さがcleartextのサイズを示します。状況によっては、cleartextのサイズによってプレーンテキストが明らかになる場合もあります。パディングを使用する場合 (パディングタイプ `fixed` またはmax) は、まずすべての値が共通のサイズにパディングされ、次に暗号化されます。パディングを使用すると、暗号化されたデータの長さによって、サイズの上限は示されるものの、元のcleartextの長さに関するそれ以外の情報は得られません。

特定の列のパディングが必要で、その列のデータの最大バイト長がわかっている場合は、`fixed` パディングを使用し、少なくともその列の最大バイト長と同じ大きさの `length` 値を使用してください。

> **Note**
>
> 値が指定した `length` 値より長いとエラーが発生し、暗号化は失敗します。

列のパディングが必要で、その列のデータの最大バイト長が不明な場合は、パmaxディングを使用します。このパディングモードは、すべてのデータを、最長値に追加の `length` バイトを加えた長さにパディングします。

> **Note**
>
> データをまとめて暗号化したり、テーブルを新しいデータで定期的に更新したりする場合、`max` パディングを行うと、指定したバッチ内の最も長いプレーンテキストエントリの長さ (プラス `length` バイト) までエントリがパディングされることに注意してください。つまり、暗号文の長さはバッチごとに異なる可能性があります。したがって、列の最大バイト長がわかっている場合は、`max` の代わりに `fixed` 使用してください。

Cleartext列

Cleartext 列は、 ステートメントJOINまたは SELECTステートメントでの使用のために暗号的に保護されていない列です。

Cleartext列は SQL クエリのどの部分でも使用できます。

暗号コンピューティングパラメータ

暗号コンピューティングパラメータは、コラボレーションの作成時に、Cryptographic Computing for Clean Rooms (C3R) を使用してコラボレーションに指定できます。 AWS Clean Rooms コンソールまたは `CreateCollaboration` API オペレーションを使用してコラボレーションを作成できます。コンソールでは、{暗号コンピューティングをサポート} オプションをオンにすると、[暗号コンピューティングパラメータ] のパラメータ値を設定できます。詳細については、以下のトピックを参照してください。

トピック

- [cleartext 列を許可] パラメータ
- [複製を許可] パラメータ
- [名前の異なる列の JOIN を許可] パラメータ
- [NULL 値を保存] パラメータ

[cleartext 列を許可] パラメータ

コンソールでは、コラボレーションの作成時に [cleartext 列を許可] パラメータを設定して、暗号化されたデータを含むテーブルで cleartext データを許可するかどうかを指定できます。

次の表で、[cleartext 列を許可] パラメータの値について説明します。

パラメータ値	説明
いいえ	暗号化されたテーブルでCleartext 列を使用できません。すべてのデータは暗号で保護されます。
はい	暗号化されたテーブルでCleartext 列を使用できます。 Cleartext 列は暗号化によって保護されず、cleartext として含まれます。行の cleartext データによって、テーブル内の他のデータについて明らかになるおそれがある情報をメモしておく必要があります。

パラメータ値	説明
	特定の列で SUM または AVG を実行するには、その列が cleartext である必要があります。

`CreateCollaboration` API オペレーションを使用して、`dataEncryptionMetadata` パラメータの `allowCleartext` 値を `true` または `false` に設定できます。API オペレーションの詳細については、「AWS Clean Rooms API リファレンス」を参照してください。

Cleartext 列は、テーブル固有のスキーマで cleartext に分類される列に対応します。これらの列のデータは暗号化されておらず、どのような方法でも使用できます。Cleartext 列は、データの機密性が低い場合や、暗号化された sealed 列や fingerprint 列で許容される以上の柔軟性が必要な場合に役立ちます。

[複製を許可] パラメータ

コンソールでは、コラボレーションの作成時に [複製を許可] パラメータを設定して、暗号化された列の JOIN クエリで NULL 値以外の重複する値を含めるかどうかを指定できます。

> **Important**
>
> [複製を許可]、[名前の異なる列の JOIN を許可]、[NULL 値を保存] の各パラメータには、別々でありながら関連する効果があります。

次の表で、[複製を許可] パラメータの値について説明します。

パラメータ値	説明
いいえ	1 つの fingerprint 列で値の繰り返しは許可されません。1 つの fingerprint 列の値は、すべて一意でなければなりません。
はい	1 つの fingerprint 列で値の繰り返しが許可されます。 値が繰り返される列を結合する必要がある場合は、この値を [はい] に設定します。[はい] に設定すると、C3R テーブルまたは結果の fingerprint 列に示される頻度パターンによって、c

パラメータ値	説明
	leartext データの構造に関するその他の情報が推察可能になるおそれがあります。

`CreateCollaboration` API オペレーションを使用して、`dataEncryptionMetadata` パラメータの `allowDuplicates` 値を `true` または `false` に設定できます。API オペレーションの詳細については、「AWS Clean Rooms API リファレンス」を参照してください。

デフォルトでは、暗号化されたデータを JOIN クエリで使用する必要がある場合、C3R 暗号化クライアントでは、それらの列に重複する値がないことが求められます。この要件は、データ保護を強化するためのものです。この動作によって、データ内で繰り返されるパターンが観測可能になることを防止できます。ただし、JOIN クエリで暗号化されたデータを処理するにあたって、値の重複を気にしない場合は、[複製を許可] パラメータでこの保守的なチェックを無効にできます。

[名前の異なる列の JOIN を許可] パラメータ

コンソールでは、コラボレーションの作成時に [名前の異なる列の JOIN を許可] パラメータを設定して、異なる名前の列間の JOIN ステートメントをサポートするかどうかを指定できます。

詳細については、「列ヘッダー名の正規化 」を参照してください。

次の表で、[名前の異なる列の JOIN を許可] パラメータの値について説明します。

パラメータ値	説明
いいえ	名前の異なる fingerprint 列の結合はサポートされません。JOIN ステートメントでは、同じ名前の列でのみ正確な結果が得られます。 **Important** [いいえ] を指定すると情報セキュリティは強化されますが、列名について事前にコラボレーション参加者の合意が必要になります。fingerprint列として暗号化したときに 2 つの列の名前が異なり、[名前の異なる列の JOIN を許可] が [いいえ] に設定されていると、それらの列の JOIN ステートメントは結果を生成しません。

パラメータ値	説明
	これは、暗号化後の値が 2 つの列の間で共有されないためです。
はい	名前の異なるfingerprint列の結合がサポートされます。柔軟性を高めるために、ユーザーはこの値を [はい] に設定できます。これにより、名前に関係なく列に対して JOIN ステートメントを実行できます。 [はい] に設定すると、C3R 暗号化クライアントは fingerprint 列を保護する際に列名を考慮しません。その結果、C3R テーブルの異なる fingerprint 列で共通する値が観測可能になります。 例えば、ある行の `City` 列と `State` 列の両方に暗号化された同一の JOIN 値が存在する場合、その値は `New York` であると合理的に推測できます。

`CreateCollaboration` API オペレーションを使用して、`dataEncryptionMetadata` パラメータの `allowJoinsOnColumnsWithDifferentNames` 値を `true` または `false` に設定できます。API オペレーションの詳細については、「AWS Clean Rooms API リファレンス」を参照してください。

デフォルトでは、fingerprint 列の暗号化は、「ステップ 4: 表形式ファイルの暗号化スキーマを生成する 」で設定されたその列の `targetHeader` 設定の影響を受けます。そのため、同じ cleartext 値でも、暗号化対象の fingerprint 列ごとに暗号化表現は異なります。

このパラメータは、cleartext 値が推測されるのを防ぐのに役立つ場合があります。例えば、fingerprint 列 `City` と `State` に暗号化された同一の値が表示されている場合、その値は `New York` であると合理的に推測できます。ただし、このパラメータを使用するには、クエリで結合されるすべての列が共通の名前になるように、事前に追加の調整が必要になります。

[名前の異なる列の JOIN を許可] パラメータを使用すると、この制限を緩和できます。パラメータ値を `Yes` に設定すると、名前に関係なく、暗号化されたすべての列を JOIN で一緒に使用できます。

[NULL 値を保存] パラメータ

コンソールでは、コラボレーションの作成時に [NULL 値を保存] パラメータを設定して、その列に値がないことを示すことができます。

次の表で、[NULL 値を保存] パラメータの値について説明します。

パラメータ値	説明
いいえ	NULL 値は保持されません。NULL 値は暗号化されたテーブルで NULL として表示されません。NULL 値は C3R テーブルに一意のランダム値として表示されます。
はい	NULL値は保持されます。NULL 値は暗号化されたテーブルで NULL として表示されます。NULL 値の SQL セマンティックが必要な場合は、この値を [はい] に設定できます。その結果、列が暗号化されているかどうか、および [複製を許可] パラメータの設定に関係なく、NULL エントリは C3R テーブルで NULL として表示されます。

`CreateCollaboration` API オペレーションを使用して、`dataEncryptionMetadata` パラメータの `preserveNulls` 値を `true` または `false` に設定できます。API オペレーションの詳細については、「AWS Clean Rooms API リファレンス」を参照してください。

コラボレーションの [NULL 値を保存] パラメータが [いいえ] に設定されている場合の動作は以下の通りです。

1. `cleartext` 列の NULL エントリは変更されません。
2. 暗号化された`fingerprint` 列の NULL エントリは、内容を隠すためにランダムな値として暗号化されます。暗号化された列をそのcleartext列の NULL エントリと結合しても、どの NULL エントリとも一致しません。それぞれが一意のランダムコンテンツを受け取るため、一致は生じません。
3. 暗号化された`sealed` 列の NULL エントリは暗号化されます。

コラボレーションの [NULL 値を保存] パラメータ値が [はい] に設定されている場合、列が暗号化されているかどうかに関係なく、すべての列の NULL エントリは NULL のままとなります。

[NULL 値を保存] パラメータは、データエンリッチメントなどのシナリオにおいて、情報が欠落し、NULL で表されているエントリを共有する場合に便利です。また、[NULL 値を保存] パラメータは、JOIN または GROUP BY を実行する列に NULL がある場合に、fingerprint または HMAC 形式でも役立ちます。

[複製を許可] パラメータと [NULL 値を保存] パラメータの値が [いいえ] に設定されている場合、1 つのfingerprint 列に複数の NULL エントリがあると、エラーが発生して暗号化が停止します。いずれかのパラメータ値が [はい] に設定されていれば、そのようなエラーは発生しません。

Cryptographic Computing for Clean Rooms のオプションフラグ

以下のセクションでは、表形式のファイルのカスタマイズとテストにおいて、C3R 暗号化クライアントを使ってデータを暗号化するときに設定できるオプションフラグについて説明します。

トピック

`--csvInputNULLValue` フラグ

C3R 暗号化クライアントを使用してデータを暗号化するときに、`--csvInputNULLValue` フラグを使用して入力データの NULL エントリにカスタムエンコーディングを指定できます。

次の表は、このフラグの使用方法とパラメータをまとめたものです。

使用方法	パラメータ
オプション。ユーザーは入力データの NULL エントリにカスタムエンコーディングを指定できます。	入力 CSV ファイルの NULL 値に対するユーザー指定のエンコーディング

NULL エントリとは、特に SQL テーブルのようなよりリッチな表形式のコンテキストでは、欠落したコンテンツとみなされるエントリです。.csv は歴史的な理由からこの特性を明示的にサポート

していませんが、空白だけを含む空のエントリは NULL と見なすのが一般的な慣習です。したがって、これは C3R 暗号化クライアントのデフォルト動作であり、必要に応じてカスタマイズできます。

--csvOutputNULLValue フラグ

C3R 暗号化クライアントを使用してデータを暗号化するときに、--csvOutputNULLValue フラグを使用して出力データの NULL エントリにカスタムエンコーディングを指定できます。

次の表は、このフラグの使用方法とパラメータをまとめたものです。

使用方法	パラメータ
オプション。ユーザーは、NULL エントリの生成された出力ファイルにカスタムエンコーディングを指定できます。	出力 CSV ファイルの NULL 値に対するユーザー指定のエンコーディング

NULL エントリとは、特に SQL テーブルのようなよりリッチな表形式のコンテキストでは、欠落したコンテンツとみなされるエントリです。.csv は歴史的な理由からこの特性を明示的にサポートしていませんが、空白だけを含む空のエントリは NULL と見なすのが一般的な慣習です。したがって、これは C3R 暗号化クライアントのデフォルト動作であり、必要に応じてカスタマイズできます。

--enableStackTraces フラグ

C3R 暗号化クライアントを使用してデータを暗号化するときに --enableStackTraces フラグを使用すると、C3R でエラーが発生したときに、エラー報告用の追加のコンテキスト情報が提供されます。

AWS はエラーを収集しません。エラーが発生した場合は、スタックトレースを使用してエラーを自分でトラブルシューティングするか、スタックトレースを に送信してサポート AWS Support を依頼してください。

次の表は、このフラグの使用方法とパラメータをまとめたものです。

使用方法	パラメータ
オプション。C3R 暗号化クライアントでエラーが発生したときに、エラー報告用の追加	なし

使用方法	パラメータ
のコンテキスト情報を提供するために使用します。	

--dryRun フラグ

C3R 暗号化クライアントの暗号化コマンドと復号化コマンドにはオプションの --dryRun フラグがあります。このフラグは、ユーザーが指定した引数をすべて受け取り、その有効性と一貫性をチェックします。

この --dryRun フラグを使用して、スキーマファイルが有効で対応する入力ファイルと一致しているかどうかを確認できます。

次の表は、このフラグの使用方法とパラメータをまとめたものです。

使用方法	パラメータ
オプション。C3R 暗号化クライアントはパラメータの解析とファイルのチェックを行いますが、暗号化や復号化は行いません。	なし

--tempDir フラグ

設定によっては、暗号化されたファイルは暗号化されていないファイルよりもサイズが大きくなることがあるため、一時ディレクトリを使用することができます。また、データセットを正常に機能させるには、コラボレーションごとに暗号化する必要があります。

C3R を使用してデータを暗号化する場合は、--tempDir フラグを使用して、入力の処理中に一時ファイルを作成する場所を指定できます。

次の表は、このフラグの使用方法とパラメータをまとめたものです。

使用方法	パラメータ
ユーザーは、入力の処理中に一時ファイルを作成する場所を指定できます。	デフォルトはシステム一時ディレクトリです。

クエリと Cryptographic Computing for Clean Rooms

このトピックでは、Cryptographic Computing for Clean Rooms で暗号化されたデータテーブルを使用するクエリの作成について情報を提供します。

トピック

NULL で分岐するクエリ

NULL ステートメントでクエリを分岐するということは、`IF x IS NULL THEN 0 ELSE 1` のような構文を使用することを意味します。

cleartext列では、NULL ステートメントで常にクエリを分岐させることができます。

sealed列やfingerprint列の場合は、[NULL 値を保存] パラメータ (`preserveNulls`) の値が `true` に設定されている場合のみ、NULL ステートメントでクエリを分岐させることができます。

これらの制約に違反するクエリは、誤った結果をもたらす可能性があります。

1 つのソース列を複数のターゲット列にマッピングする

1 つのソース列を複数のターゲット列にマッピングできます。例えば、1 つの列で JOIN と SELECT の両方を実行することができます。

詳細については、「JOIN クエリと SELECT クエリの両方に同じデータを使用する」を参照してください。

JOIN クエリと SELECT クエリの両方に同じデータを使用する

列内のデータが機密情報ではない場合、そのデータはcleartextのターゲット列に表示され、あらゆる目的に使用できます。

列内のデータが機密情報で、JOIN クエリと SELECT クエリの両方に使用する必要がある場合は、そのソース列を出力ファイルの 2 つのターゲット列にマッピングします。1 つの列は `type` を

fingerprint列として暗号化し、もう 1 つの列は type をシール列として暗号化します。C3R 暗号化クライアントのインタラクティブなスキーマ生成では、ヘッダーサフィックスとして _fingerprint と _sealed が提示されます。これらのヘッダーサフィックスは、このような列をすばやく区別するのに便利な規則です。

C3R 暗号化クライアントのガイドライン

C3R 暗号化クライアントは、組織が機密データをまとめてデータ分析から新しいインサイトを引き出すために使用するツールです。このツールは、任意の当事者やプロセス AWS で学習できる内容を暗号的に制限します。これはきわめて重要ですが、データを暗号化によって保護するプロセスでは、コンピューティングリソースとストレージリソースの両面で大きなオーバーヘッドが生じる可能性があります。そのため、各設定を使用する際のトレードオフや、必要な暗号化保証を維持しながら最適な設定を行う方法を理解することが重要です。このトピックでは、C3R 暗号化クライアントとスキーマのさまざまな設定がパフォーマンスに与える影響に焦点を当てています。

C3R 暗号化クライアントの暗号化設定はすべて、異なる暗号化保証を提供します。コラボレーションレベルの設定が、デフォルトでは最も安全です。コラボレーションの作成中に追加機能を有効にすると、暗号文で頻度分析などのアクティビティを実行できるようになり、プライバシーの保証が弱まります。これらの設定がどのように使用され、どのような影響を及ぼすかの詳細については、「暗号コンピューティング」を参照してください。

トピック

- 列タイプがパフォーマンスに与える影響
- 暗号文のサイズが予期せず大きくなった場合のトラブルシューティング

列タイプがパフォーマンスに与える影響

C3R ではcleartext、fingerprint、sealedの 3 つの列タイプを使用します。これらの列タイプはそれぞれ異なる暗号化保証を提供し、使用目的も異なります。以下のセクションでは、列タイプがパフォーマンスに与える影響と、各設定がパフォーマンスに与える影響について説明します。

トピック

- Cleartext列
- Fingerprint列
- Sealed列

Cleartext列

Cleartext列は元の形式から変更されておらず、暗号化処理もされていません。この列タイプを設定することはできず、ストレージやコンピューティングのパフォーマンスにも影響はありません。

Fingerprint列

Fingerprint列は複数のテーブルにまたがるデータを結合するために使用されます。そのためには、生成される暗号文のサイズが常に同じでなければなりません。ただし、これらの列はコラボレーションレベルの設定による影響を受けます。Fingerprint列は、入力に含まれるcleartextに応じて、出力ファイルのサイズにさまざまな程度の影響を与える可能性があります。

トピック

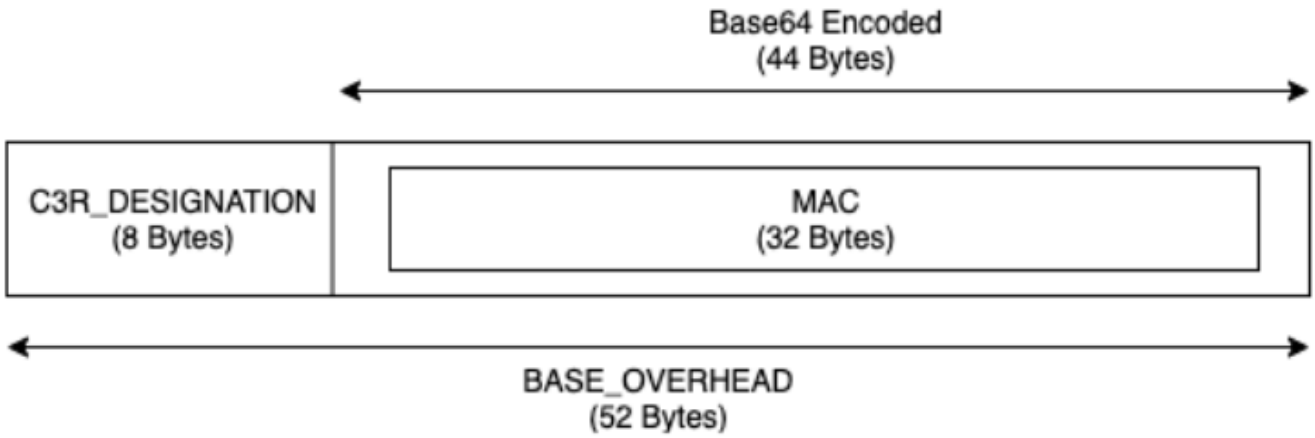
- fingerprint列の基本オーバーヘッド
- fingerprint列のコラボレーション設定
- fingerprint列のデータ例
- fingerprint列のトラブルシューティング

fingerprint列の基本オーバーヘッド

fingerprint列には基本オーバーヘッドがあります。このオーバーヘッドは一定であり、cleartextのバイトサイズに代わるものです。

fingerprint列内のデータは、Hash-based Message Authentication Code (HMAC)関数によって暗号化処理され、データが 32 バイトのメッセージ認証コード (MAC) に変換されます。その後、このデータは base64 エンコーダで処理され、バイトサイズが約 33% 増加します。先頭には、データが属する列の種類とそれを生成したクライアントバージョンを示す 8 バイトの C3R 指定子が付加されます。最終結果は 52 バイトです。次に、この結果に行数を掛けて、基本オーバーヘッドの合計を求めます (`preserveNulls` が true に設定されている場合は、`null` 値以外の合計数を使用します)。

以下の図は、*`BASE_OVERHEAD = C3R_DESIGNATION + (MAC * 1.33)`* を表しています。



fingerprint列の出力暗号文は常に 52 バイトです。cleartextの入力データの平均が 52 バイトを超える場合 (完全な住所など) は、これによってストレージサイズが大幅に減少する可能性があります。cleartextの入力データの平均が 52 バイト未満の場合 (顧客の年齢など) は、ストレージサイズが大幅に増える可能性があります。

fingerprint列のコラボレーション設定

`preserveNulls` の設定

コラボレーションレベルの `preserveNulls` 設定が `false` (デフォルト) の場合、各 `null` 値は固有のランダムな 32 バイトに置き換えられ、`null` ではないかのように処理されます。結果として、各 `null` 値は 52 バイトになります。これにより、データが非常に少ないテーブルでは、この設定が `true` で `null` 値が `null` として渡される場合と比べて、ストレージ要件が大幅に増加する可能性があります。

この設定によるプライバシー保証が不要で、`null` 値をデータセット内に保持する場合は、コラボレーションの作成時に `preserveNulls` 設定を有効にしてください。コラボレーションの作成後に `preserveNulls` 設定を変更することはできません。

fingerprint列のデータ例

以下は、再現可能な設定を含むfingerprint列の入出力データのサンプルセットです。`allowCleartext` や `allowDuplicates` などの他のコラボレーションレベルの設定は結果に影響せず、ローカルで再現を試みる場合は `true` または `false` に設定できます。

共有シークレットの例: `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`

コラボレーション ID の例: `a1b2c3d4-5678-90ab-cdef-EXAMPLE11111`

allowJoinsOnColumnsWithDifferentNames: True。この設定はパフォーマンスやストレージ要件には影響しません。ただし、この設定を行うと、次の表に示す値を再現するときに、列名の選択は重要ではなくなります。

例 1

入力	null
preserveNulls	TRUE
出力	null
確定的	Yes
入力バイト数	0
出力バイト数	0

例 2

入力	null
preserveNulls	FALSE
出力	01:hmac:3lkFjthvV3IUu6mMvFc1a+XAHwgw/ElmOq4p3Yg25kk=
確定的	No
入力バイト数	0
出力バイト数	52

例 3

入力	empty string
preserveNulls	-

出力	01:hmac:oKTgi3Gba+eUb3JteSz 2EMgXUkF1WgM77UP0Ydw5kPQ=
確定的	Yes
入力バイト数	0
出力バイト数	52

例 4

入力	abcdefghijklmnopqrstuvwxyz
preserveNulls	-
出力	01:hmac:kU/IqwG7FMmzzshr0B9 scomE0UJUEE7j9keTctplGww=
確定的	Yes
入力バイト数	26
出力バイト数	52

例 5

入力	abcdefghijklmnopqrstuvwxyzA BCDEFGHIJKLMNOPQRSTUVWXYZ01 23456789
preserveNulls	-
出力	01:hmac:ks3htnQbw2vdhCRFF6J NzW5LMndJaHG57uvE26mBtSs=
確定的	Yes
入力バイト数	62

出力バイト数	52

fingerprint列のトラブルシューティング

fingerprint列の暗号文が、入力されたcleartextのサイズより数倍大きいのはなぜですか?

fingerprint列の暗号文の長さは常に 52 バイトです。入力データが小さい場合 (顧客の年齢など) は、サイズが大幅に増加します。この現象は、`preserveNulls` 設定が `false` に設定されている場合にも発生する可能性があります。

fingerprint列の暗号文が、入力されたcleartextのサイズより数倍小さいのはなぜですか?

fingerprint列の暗号文の長さは常に 52 バイトです。入力データが大きい場合 (顧客の完全な住所など) は、サイズが大幅に減少します。

`preserveNulls` による暗号保証が必要かどうかはどうすればわかりますか?

答えは状況により異なります。少なくとも、`preserveNulls` 設定によってデータがどのように保護されるかを「the section called “パラメータ”」で確認する必要があります。ただし、組織のデータ処理要件と、それぞれのコラボレーションに適用される契約を参照することをお勧めします。

base64 のオーバーヘッドが発生するのはなぜですか?

CSV などの表形式のファイル形式との互換性を保つには、base64 エンコーディングが必要です。Parquet などの一部のファイル形式はデータのバイナリ表現をサポートしていますが、適切なクエリ結果を得るためには、コラボレーションのすべての参加者が同じ方法でデータを表現することが重要です。

Sealed列

Sealed列は、コラボレーションのメンバー間でデータを転送するために使用します。これらの列の暗号文は非確定的であり、列の構成方法によってはパフォーマンスとストレージの両方に大きな影響を与えます。これらの列は個別に設定でき、多くの場合、C3R 暗号化クライアントのパフォーマンスとそれに伴う出力ファイルサイズに最も大きな影響を与えます。

トピック

- sealed列の基本オーバーヘッド
- sealed列のコラボレーション設定
- スキーマ設定sealed列: パディングタイプ

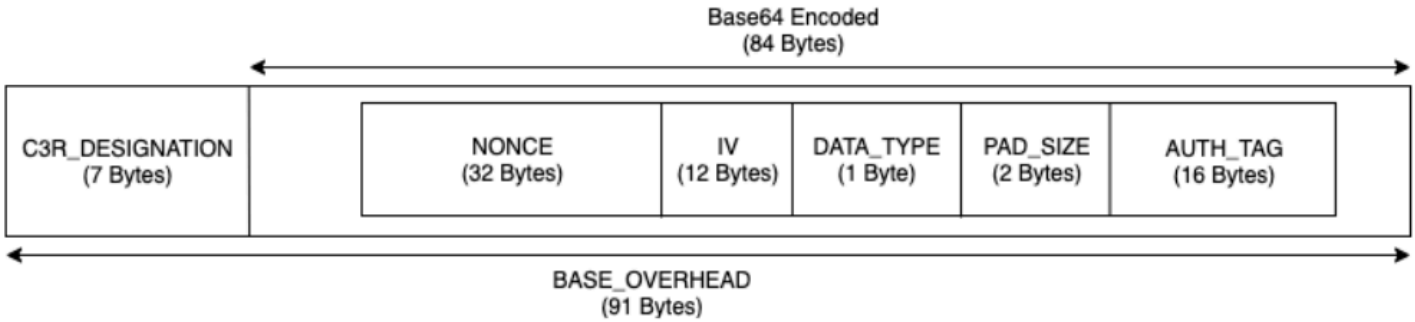
- sealed列のデータ例
- sealed列のトラブルシューティング

sealed列の基本オーバーヘッド

sealed列には基本オーバーヘッドがあります。このオーバーヘッドは一定であり、cleartextとパディング (ある場合) のバイトサイズに追加されるものです。

暗号化の前に、sealed列のデータの先頭には、含まれるデータのタイプを示す 1 バイトの文字が付加されます。パディングを選択すると、データがパディングされ、パッドサイズを示す 2 バイトが追加されます。これらのバイトが追加された後、データは AES-GCM を使用して暗号化処理され、IV (12 バイト)、nonce (32 バイト)、Auth Tag (16 バイト) と共に格納されます。その後、このデータは base64 エンコーダで処理され、バイトサイズが約 33% 増加します。データの先頭には、データが属する列の種類とその生成に使用されたクライアントバージョンを示す 7 バイトの C3R 指定子が付加されます。その結果、最終的な基本オーバーヘッドは 91 バイトになります。次に、この結果に行数を掛けて、基本オーバーヘッドの合計を求めます (`preserveNulls` が true に設定されている場合は、null 値以外の合計数を使用します)。

以下の図は、 *`BASE_OVERHEAD = C3R_DESIGNATION + ((NONCE + IV + DATA_TYPE + PAD_SIZE + AUTH_TAG) * 1.33)`* を表しています。



sealed列のコラボレーション設定

`preserveNulls` の設定

コラボレーションレベルの `preserveNulls` 設定が `false` (デフォルト) の場合、各 `null` 値は固有のランダムな 32 バイト値となり、`null` ではないかのように処理されます。結果として、各 `null` 値は 91 バイト (パディングされている場合はそれ以上) になります。これにより、データが非常に少ないテーブルでは、この設定が `true` で `null` 値が `null` として渡される場合と比べて、ストレージ要件が大幅に増加する可能性があります。

この設定によるプライバシー保証が不要で、null 値をデータセット内に保持する場合は、コラボレーションの作成時に preserveNulls 設定を有効にしてください。コラボレーションの作成後に preserveNulls 設定を変更することはできません。

スキーマ設定sealed列: パディングタイプ

トピック

パディングタイプ **none**

none のパディングタイプを選択すると、cleartextにパディングは追加されず、前述の基本オーバーヘッドにさらにオーバーヘッドが追加されることもありません。パディングがないと、最もスペース効率の良い出力サイズになります。ただし、fixed や max のパディングタイプと同じプライバシー保証は提供されません。これは、基になるcleartextのサイズが暗号文のサイズから識別できるためです。

パディングタイプ **fixed**

fixed のパディングタイプを選択すると、列に含まれるデータの長さが隠蔽され、プライバシーを保護する手段となります。これは、暗号化の前に、すべてのcleartextを指定された pad_length にパディングすることで行われます。このサイズを超えるデータがあると、C3R 暗号化クライアントは機能しなくなります。

暗号化の前にcleartextにパディングが追加される場合、AES-GCM で、cleartextと暗号文のバイトとの 1 対 1 のマッピングが行われます。base64 エンコーディングによってサイズが 33% 増加します。パディングによって増加するストレージのオーバーヘッドは、pad_length の値からcleartextの長さの平均値を引き、1.33 を掛けることで算出できます。その結果が、レコードごとのパディングの平均オーバーヘッドになります。次に、この結果に行の数を掛けて、パディングのオーバーヘッドの合計を求めます (preserveNulls が true に設定されている場合は、null 値以外の合計数を使用します)。

*PADDING_OVERHEAD = (PAD_LENGTH - AVG_CLEARTEXT_LENGTH) * 1.33 * ROW_COUNT*

pad_length の最小値には、列内の最大値が収まる値を選択することをお勧めします。例えば、最大値が 50 バイトの場合は、50 バイトの pad_length で十分です。これより大きい値では、ストレージのオーバーヘッドが増えるだけです。

固定長のパディングでは、コンピューティングのオーバーヘッドは大幅に増加しません。

パディングタイプ **max**

max のパディングタイプを選択すると、列に含まれるデータの長さが隠蔽され、プライバシーを保護する手段となります。これは、暗号化の前に、すべての cleartext を列内の最大値に追加の pad_length を加算した値までパディングすることで行われます。一般に、max のパディングは 1 つのデータセットに対して fixed のパディングと同様の保証を提供しますが、列内のcleartextの最大値を把握していなくてもかまいません。ただし、更新時には、個々のデータセットの最大値が変わる場合があるため、max のパディングで fixed のパディングと同様のプライバシー保証が提供されるとは限りません。

max のパディングを使用するときは、0 の pad_length を追加で選択することをお勧めします。この長さにより、すべての値が列の最大値と同じサイズにパディングされます。これより大きい値では、ストレージのオーバーヘッドが増えるだけです。

特定の列の cleartext の最大値がわかっている場合は、代わりに fixed のパディングタイプを使用することをお勧めします。fixed のパディングを使用すると、データセットの更新前後で一貫性が保たれます。max のパディングを使用すると、データの各サブセットが、そのサブセットにあった最大値までパディングされます。

sealed列のデータ例

以下は、再現可能な設定を含む sealed 列の入出力データのサンプルセットです。allowCleartext、allowJoinsOnColumnsWithDifferentNames、allowDuplicates などの他のコラボレーションレベルの設定は結果に影響せず、ローカルで再現を試みる場合は true または false に設定できます。これらは再現するための基本設定ですが、sealed 列は非確定的であり、値は毎回変動します。目標は、入力されたバイトと出力されたバイトを比較することです。サンプルの pad_length 値は意図的に選択されています。ここでは、fixed のパディングが、推奨される最小 pad_length 設定を使用した max のパディングと同じ値になること、または追加のパディングが望ましいケースが示されています。

共有シークレットの例: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY

コラボレーション ID の例: a1b2c3d4-5678-90ab-cdef-EXAMPLE11111

トピック

- パディングタイプ none
- パディングタイプ fixed (例 1)
- パディングタイプ fixed (例 2)

- パディングタイプ max (例 1)
- パディングタイプ max (例 2)

パディングタイプ **none**

例 1

入力	null
preserveNulls	TRUE
出力	null
確定的	Yes
入力バイト数	0
出力バイト数	0

例 2

入力	null
preserveNulls	FALSE
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5MG5vbmNlMDEyMzQ1Njc4OTBqfRYZ98t5KU6aWfssGSPbNIJfG3iXmu6cbCUrizuV
確定的	No
入力バイト数	0
出力バイト数	91

例 3

入力	empty string

preserveNulls	-
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5MG5vbmNlMDEyMzQ1Njc4OTBqfRYZ98t5KU6aWfstGSPEM6qR8DWC2PB2GMlX41YK
確定的	No
入力バイト数	0
出力バイト数	91

例 4

入力	abcdefghijklmnopqrstuvwxyz
preserveNulls	-
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5MG5vbmNlMDEyMzQ1Njc4OTBqfRYZ98t5KU6aWfsteEE1GKEPiRzyh0h7t6OmWMLTWCvO2ckr6pkx9sGL5VLDQeHzh6DmPpyWNuI=
確定的	No
入力バイト数	26
出力バイト数	127

例 5

入力	abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
preserveNulls	-

出力	01:enc:bm9uY2UwMTIzNDU2Nzg5MG5vbmNlMDEyMzQ1Njc4OTBqfRYZ98t5KU6aWfsteEE1GKEPiRzyh0h7t6OmWMLTWCvO2ckr6plwtH/8tRFnn2rF91bcB9G4+n8GiRfJNmqdP4/QOQ3cXb/pbvPcnnohrHIGSX54ua+1/JfcVjc=
確定的	No
入力バイト数	62
出力バイト数	175

パディングタイプ **fixed** (例 1)

この例では、pad_length は 62 で最大入力値は 62 バイトです。

例 1

入力	null
preserveNulls	TRUE
出力	null
確定的	Yes
入力バイト数	0
出力バイト数	0

例 2

入力	null
preserveNulls	FALSE

出力	01:enc:bm9uY2UwMTIzNDU2Nzg5 MG5vbmNlMDEyMzQ1Njc4OTBqfRY Z98t5KU6aWfssGSNWfMRp7nSb7S MX2s3JKLOhK1+7r75Tk+Mx9jy48 Fcg1yOPvBqRSZ7oqy1V3UKfYTLEZb/ hCz7oaIneVsrcoNpATs0GzbnLkor4L+/ aSuA=
確定的	No
入力バイト数	0
出力バイト数	175

例 3

入力	empty string
preserveNulls	-
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5 MG5vbmNlMDEyMzQ1Njc4OTBqfRY Z98t5KU6aWfstGSNWfMRp7nSb7S MX2s3JKLOhK1+7r75Tk+Mx9jy48 Fcg1yOPvBqRSZ7oqy1V3UKfYTLE Zb/hCz7oaIneVsrcoLB53l07VZp A6OwkuXu29CA=
確定的	No
入力バイト数	0
出力バイト数	175

例 4

入力	abcdefghijklmnopqrstuvwxyz

preserveNulls	-
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5MG5vbmNlMDEyMzQ1Njc4OTBqfRYZ98t5KU6aWfsteEE1GKEPiRzyh0h7t6OmWMLTWCvO2ckr6pkx9jy48Fcg1yOPvBqRSZ7oqy1V3UKfYTLEZb/hCz7oaIneVsrcutBAcO+Mb9tuU2KIHH31AWg=
確定的	No
入力バイト数	26
出力バイト数	175

例 5

入力	abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
preserveNulls	-
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5MG5vbmNlMDEyMzQ1Njc4OTBqfRYZ98t5KU6aWfsteEE1GKEPiRzyh0h7t6OmWMLTWCvO2ckr6plwtH/8tRFnn2rF91bcB9G4+n8GiRfJNmqdP4/QOQ3cXb/pbvPcnnohrHIGSX54ua+1/JfcVjc=
確定的	No
入力バイト数	62
出力バイト数	175

パディングタイプ **fixed** (例 2)

この例では、pad_length は 162 で最大入力値は 62 バイトです。

例 1

入力	null
preserveNulls	TRUE
出力	null
確定的	Yes
入力バイト数	0
出力バイト数	0

例 2

入力	null
preserveNulls	FALSE
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5 MG5vbmNlMDEyMzQ1Njc4OTBqfRY Z98t5KU6aWfssGSNWfMRp7nSb7S MX2s3JKLOhK1+7r75Tk+Mx9jy48 Fcg1yOPvBqRSZ7oqy1V3UKfYTLE Zb/hCz7oaIneVsrcnkB0xbLWD7z NdAqQGR0rXoSESdW0I0vpNoGcBf v4cJbG0A3h1DvtkSSVc2B80OOGp pzdDqhrUVN5wFNyn8vgfPMqDaeJk5bn +8o4WtG/ClipNcjDXvXVtK4vfCohcCA6 uwrmwv/xAySX+xcntotL703aBTBb
確定的	No
入力バイト数	0

出力バイト数	307

例 3

入力	empty string
preserveNulls	-
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5 MG5vbmNlMDEyMzQ1Njc4OTBqfRY Z98t5KU6aWfstGSNWfMRp7nSb7S MX2s3JKLOhK1+7r75Tk+Mx9jy48 Fcg1yOPvBqRSZ7oqy1V3UKfYTLE Zb/hCz7oaIneVsrcnkB0xbLWD7z NdAqQGR0rXoSESdW0I0vpNoGcBf v4cJbG0A3h1DvtkSSVc2B80OOGp pzdDqhrUVN5wFNyn8vgfPMqDaeJk5bn +8o4WtG/ClipNcjDXvXVtK4vfCohcCA6 uwrmwv84lVaT9Yd+6oQx65/+gdVT
確定的	No
入力バイト数	0
出力バイト数	307

例 4

入力	abcdefghijklmnopqrstuvwxyz
preserveNulls	-
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5 MG5vbmNlMDEyMzQ1Njc4OTBqfRY Z98t5KU6aWfsteEE1GKEPiRzyh0 h7t6OmWMLTWCvO2ckr6pkx9jy48 Fcg1yOPvBqRSZ7oqy1V3UKfYTLE

	Zb/hCz7oaIneVsrcnkB0xbLWD7z NdAqQGR0rXoSESdW0I0vpNoGcBf v4cJbG0A3h1DvtkSSVc2B80OOGp pzdDqhrUVN5wFNyn8vgfPMqDaeJk5bn +8o4WtG/ClipNcjDXvXVtK4vfCohcCA6 uwrmwtX5Hnl+WyfO6ks3QMaRDGSf
確定的	No
入力バイト数	26
出力バイト数	307

例 5

入力	abcdefghijklmnopqrstuvwxyzA BCDEFGHIJKLMNOPQRSTUVWXYZ01 23456789
preserveNulls	-
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5 MG5vbmNlMDEyMzQ1Njc4OTBqfRY Z98t5KU6aWfsteEE1GKEPiRzyh0 h7t6OmWMLTWCvO2ckr6plwtH/8t RFnn2rF91bcB9G4+n8GiRfJNmqd P4/QOQ3cXb/pbvPcnkB0xbLWD7z NdAqQGR0rXoSESdW0I0vpNoGcBf v4cJbG0A3h1DvtkSSVc2B80OOGp pzdDqhrUVN5wFNyn8vgfPMqDaeJk5bn +8o4WtG/ClipNcjDXvXVtK4vfCohcCA6 uwrmwjkJXQZOgPdeFX9Yr/8alV5i
確定的	No
入力バイト数	62
出力バイト数	307

パディングタイプ **max** (例 1)

この例では、pad_length は 0 で最大入力値は 62 バイトです。

例 1

入力	null
preserveNulls	TRUE
出力	null
確定的	Yes
入力バイト数	0
出力バイト数	0

例 2

入力	null
preserveNulls	FALSE
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5 MG5vbmNlMDEyMzQ1Njc4OTBqfRY Z98t5KU6aWfssGSNWfMRp7nSb7S MX2s3JKLOhK1+7r75Tk+Mx9jy48 Fcg1yOPvBqRSZ7oqy1V3UKfYTLEZb/ hCz7oaIneVsrcoNpATs0GzbnLkor4L+/ aSuA=
確定的	No
入力バイト数	0
出力バイト数	175

例 3

入力	empty string
preserveNulls	-
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5MG5vbmNlMDEyMzQ1Njc4OTBqfRYZ98t5KU6aWfstGSNWfMRp7nSb7SMX2s3JKLOhK1+7r75Tk+Mx9jy48Fcg1yOPvBqRSZ7oqy1V3UKfYTLEZb/hCz7oaIneVsrcoLB53l07VZpA6OwkuXu29CA=
確定的	No
入力バイト数	0
出力バイト数	175

例 4

入力	abcdefghijklmnopqrstuvwxyz
preserveNulls	-
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5MG5vbmNlMDEyMzQ1Njc4OTBqfRYZ98t5KU6aWfsteEE1GKEPiRzyh0h7t6OmWMLTWCvO2ckr6pkx9jy48Fcg1yOPvBqRSZ7oqy1V3UKfYTLEZb/hCz7oaIneVsrcutBAcO+Mb9tuU2KIHH31AWg=
確定的	No
入力バイト数	26
出力バイト数	175

例 5

入力	abcdefghijklmnopqrstuvwxyzA BCDEFGHIJKLMNOPQRSTUVWXYZ01 23456789
preserveNulls	-
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5 MG5vbmNlMDEyMzQ1Njc4OTBqfRY Z98t5KU6aWfsteEE1GKEPiRzyh0 h7t6OmWMLTWCvO2ckr6plwtH/8t RFnn2rF91bcB9G4+n8GiRfJNmqdP4/ QOQ3cXb/pbvPcnnohrHIGSX54ua+1/ JfcVjc=
確定的	No
入力バイト数	62
出力バイト数	175

パディングタイプ **max** (例 2)

この例では、pad_length は 100 で最大入力値は 62 バイトです。

例 1

入力	null
preserveNulls	TRUE
出力	null
確定的	Yes
入力バイト数	0
出力バイト数	0

例 2

入力	null
preserveNulls	FALSE
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5 MG5vbmNlMDEyMzQ1Njc4OTBqfRY Z98t5KU6aWfssGSNWfMRp7nSb7S MX2s3JKLOhK1+7r75Tk+Mx9jy48 Fcg1yOPvBqRSZ7oqy1V3UKfYTLE Zb/hCz7oaIneVsrcnkB0xbLWD7z NdAqQGR0rXoSESdW0I0vpNoGcBf v4cJbG0A3h1DvtkSSVc2B80OOGp pzdDqhrUVN5wFNyn8vgfPMqDaeJk5bn +8o4WtG/ClipNcjDXvXVtK4vfCohcCA6 uwrmwv/xAySX+xcntotL703aBTBb
確定的	No
入力バイト数	0
出力バイト数	307

例 3

入力	empty string
preserveNulls	-
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5 MG5vbmNlMDEyMzQ1Njc4OTBqfRY Z98t5KU6aWfstGSNWfMRp7nSb7S MX2s3JKLOhK1+7r75Tk+Mx9jy48 Fcg1yOPvBqRSZ7oqy1V3UKfYTLE Zb/hCz7oaIneVsrcnkB0xbLWD7z NdAqQGR0rXoSESdW0I0vpNoGcBf v4cJbG0A3h1DvtkSSVc2B80OOGp

	pzdDqhrUVN5wFNyn8vgfPMqDaeJk5bn +8o4WtG/ClipNcjDXvXVtK4vfCohcCA6 uwrmwv84lVaT9Yd+6oQx65/+gdVT
確定的	No
入力バイト数	0
出力バイト数	307

例 4

入力	abcdefghijklmnopqrstuvwxyz
preserveNulls	-
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5 MG5vbmNlMDEyMzQ1Njc4OTBqfRY Z98t5KU6aWfsteEE1GKEPiRzyh0 h7t6OmWMLTWCvO2ckr6pkx9jy48 Fcg1yOPvBqRSZ7oqy1V3UKfYTLE Zb/hCz7oaIneVsrcnkB0xbLWD7z NdAqQGR0rXoSESdW0I0vpNoGcBf v4cJbG0A3h1DvtkSSVc2B80OOGp pzdDqhrUVN5wFNyn8vgfPMqDaeJk5bn +8o4WtG/ClipNcjDXvXVtK4vfCohcCA6 uwrmwtX5Hnl+WyfO6ks3QMaRDGSf
確定的	No
入力バイト数	26
出力バイト数	307

例 5

入力	abcdefghijklmnopqrstuvwxyzA BCDEFGHIJKLMNOPQRSTUVWXYZ01 23456789
preserveNulls	-
出力	01:enc:bm9uY2UwMTIzNDU2Nzg5 MG5vbmNlMDEyMzQ1Njc4OTBqfRY Z98t5KU6aWfsteEE1GKEPiRzyh0 h7t6OmWMLTWCvO2ckr6plwtH/8t RFnn2rF91bcB9G4+n8GiRfJNmqd P4/QOQ3cXb/pbvPcnkB0xbLWD7z NdAqQGR0rXoSESdW0I0vpNoGcBf v4cJbG0A3h1DvtkSSVc2B80OOGp pzdDqhrUVN5wFNyn8vgfPMqDaeJk5bn +8o4WtG/ClipNcjDXvXVtK4vfCohcCA6 uwrmwjkJXQZOgPdeFX9Yr/8alV5i
確定的	No
入力バイト数	62
出力バイト数	307

sealed列のトラブルシューティング

sealed列の暗号文が、入力されたcleartextのサイズより数倍大きいのはなぜですか?

これはいくつかの要因によって異なります。第一に、Cleartext 列の暗号文の長さは必ず 91 バイト以上になります。入力データが小さい場合 (顧客の年齢など) は、サイズが大幅に増加します。第二に、preserveNulls が false に設定されており、入力データに多数の null 値が含まれていた場合、それらの null 値がそれぞれ 91 バイトの暗号文に変換されます。そして最後に、パディングを使用すると、当然のことながら、暗号化の前に cleartext データにバイトが追加されます。

sealed 列のほとんどのデータは非常に小さいのですが、パディングを使う必要があります。スペースを節約するために、大きな値を除外して別途処理することはできますか?

大きな値を除外して別途処理することはお勧めしません。そうすることで、C3R 暗号化クライアントで提供されるプライバシー保証が変わります。脅威の一例として、観察者が暗号化された両方のデータセットを見ることができると仮定します。あるデータサブセットの列のパディングが別のサブセットよりも大幅に大きかったり小さかったりすることを観察者に知られると、各サブセット内のデータのサイズが推測されるおそれがあります。例えば `fullName` 列が、あるファイルでは合計 40 バイトにパディングされ、別のファイルでは 800 バイトにパディングされているとします。この場合観察者は、一方のデータセットに世界で最も長い名前 (747 バイト) が含まれていると仮定する可能性があります。

`max` のパディングタイプを使用する場合、追加のパディングは必要ですか?

いいえ。`max` のパディングを使用するときは、`pad_length` (列の最大値を超える追加パディングとも呼ばれます) を 0 に設定することをお勧めします。

`fixed` のパディングを使用するときに、最大値が確実に収まるよう **`pad_length`** に大きい値を選択してもよいですか ?

はい。ただし、パディングの長さが長いと効率が低下し、必要以上に多くのストレージを消費します。最大値の大きさを確認し、`pad_length` にその値を設定することをおすすめします。

`preserveNulls` による暗号保証が必要かどうかはどうすればわかりますか?

答えは状況により異なります。少なくとも、`preserveNulls` 設定によってデータがどのように保護されるかを「Cryptographic Computing for Clean Rooms」で確認する必要があります。ただし、組織のデータ処理要件と、それぞれのコラボレーションに適用される契約を参照することをお勧めします。

base64 のオーバーヘッドが発生するのはなぜですか?

CSV などの表形式のファイル形式との互換性を保つには、base64 エンコーディングが必要です。Parquet などの一部のファイル形式はデータのバイナリ表現をサポートしていますが、適切なクエリ結果を得るためには、コラボレーションのすべての参加者が同じ方法でデータを表現することが重要です。

暗号文のサイズが予期せず大きくなった場合のトラブルシューティング

データを暗号化し、生成されたデータのサイズが驚くほど大きかったとしましょう。次の手順は、サイズが増加している場所と、実行できるアクション (ある場合) を特定するのに役立ちます。

サイズの増加が発生した場所の特定

暗号化されたデータが cleartext データよりも大幅に大きくなった理由をトラブルシューティングする前に、まずサイズが増加している場所を特定する必要があります。Cleartext 列は変更されていないため、無視しても問題ありません。残りの fingerprint 列と sealed 列を見て、大幅に増えている列を 1 つ選びます。

サイズが増加した理由の特定

fingerprint 列または sealed 列がサイズ増加の原因となっている可能性があります。

トピック

- fingerprint列が原因でサイズの増加が生じている場合
- sealed 列が原因でサイズの増加が生じている場合

fingerprint列が原因でサイズの増加が生じている場合

ストレージ増加の最も大きな原因となっているのがfingerprint列である場合は、cleartextデータが小さいこと (顧客の年齢など) が要因と考えられます。生成される各fingerprint暗号文の長さは 52 バイトになります。残念ながら、この問題については何もできません column-by-column 。ストレージ要件への影響など、この列の詳細については「fingerprint列の基本オーバーヘッド」を参照してください。

fingerprint列のサイズが大きくなるもう 1 つの原因として、`preserveNulls` のコラボレーション設定が考えられます。`preserveNulls` のコラボレーション設定が無効 (デフォルト設定) になっていると、fingerprint 列の `null` 値はすべて 52 バイトの暗号文になります。現在のコラボレーションでは、これに対してできることは何もありません。`preserveNulls` 設定はコラボレーションの作成時に設定され、正しいクエリ結果を得るためにすべてのコラボレーターが同じ設定を使用する必要があります。`preserveNulls` 設定の詳細と、この設定を有効にした場合にデータのプライバシー保護にどのような影響が及ぶかについては、「暗号コンピューティング」を参照してください。

sealed 列が原因でサイズの増加が生じている場合

ストレージ増加の最も大きな原因となっているのが sealed 列である場合、サイズの増加を引き起こしている可能性のある要因がいくつかあります。

cleartext データが小さい場合 (顧客の年齢など)、生成される各 sealed 暗号文の長さは 91 バイト以上になります。残念ながら、この問題についてできることは何もありません。ストレージ要件への影響など、この列の詳細については「sealed列の基本オーバーヘッド」を参照してください。

sealed列でストレージが増加する 2 つ目の主な要因はパディングです。パディングとは、データセット内の個々の値のサイズを隠蔽するために、暗号化の前にcleartextに余分なバイトを追加することです。データセットにとって最小限の値でパディングを設定することをお勧めします。少なくとも、fixed パディングの pad_length は、列内で考えられる最大の値が収まるように設定する必要があります。これより大きい値を設定しても、プライバシー保証は追加されません。例えば、列内で考えられる最大の値が 50 バイトであることがわかっている場合は、pad_length を 50 バイトに設定することをお勧めします。ただし、sealed 列で max のパディングを使用している場合は、pad_length を 0 バイトに設定することをお勧めします。これは、max のパディングが、列内の最大値を超える追加のパディングを指すためです。

sealed列のサイズが大きくなる原因として考えられる最後の要因は、preserveNulls のコラボレーション設定です。preserveNulls のコラボレーション設定が無効 (デフォルト設定) になっていると、sealed 列の null 値はすべて 91 バイトの暗号文になります。現在のコラボレーションでは、これに対してできることは何もありません。preserveNulls 設定はコラボレーションの作成時に設定され、正しいクエリ結果を得るためにすべてのコラボレーターが同じ設定を使用する必要があります。この設定の詳細と、設定を有効にした場合にデータのプライバシー保護にどのような影響が及ぶかについては、「暗号コンピューティング」を参照してください。

クエリログイン AWS Clean Rooms

クエリログ記録は の機能です AWS Clean Rooms。コラボレーションを作成してクエリログ記録を有効にすると、メンバーは自分に関連するクエリログを Amazon CloudWatch Logs に保存できます。

クエリのログを確認することで、クエリが分析ルールに準拠しているかどうか、コラボレーション契約に準拠しているかどうかを判断できます。さらに、クエリログは監査にも役立ちます。

AWS Clean Rooms コンソールでクエリログ記録オプションが有効になっている場合、クエリログには次のものが含まれます。

- `analysisRule` – 設定済みテーブルの分析ルール。
- `analysisTemplateArn` – 実行された分析テンプレート (分析ルールに応じて表示されます)。
- `collaborationId` – クエリが実行されたコラボレーションの一意の ID。
- `configuredTableID` – クエリで参照された設定済みテーブルの一意の ID。
- `directQueryAnalysisRulePolicy.custom.allowedAnalysis` – 設定済みテーブルで実行が許可された分析テンプレート (分析ルールに応じて表示されます)。
- `directQueryAnalysisRulePolicy.v1.custom.allowedAnalysisProviders` – クエリの作成を許可されたクエリプロバイダー (分析ルールに応じて表示されます)。
- `eventID` – 実行されたクエリの一意の ID。2023 年 8 月 31 日以降、この一意の ID は `protectedQueryID` と同じになります。
- `eventTimestamp` – クエリ実行時間。
- `parameters.parametervalue` – パラメータ値 (クエリテキストに応じて表示されます)。
- `queryText` – 実行されたクエリの SQL 定義。パラメータがある場合は、`:parametervalue` というラベルが付けられます。
- `queryValidationErrors` – クエリ検証時のクエリエラー。
- `schemaName` – クエリで参照された設定済みテーブルの関連付けの名前。

クエリログ記録の受信

クエリログを AWS Clean Rooms セットアップするために、 の外部でアクションを実行する必要はありません。 AWS Clean Rooms は、各コラボレーションメンバーがメンバーシップ を作成した後に、コラボレーションのロググループを作成します。

クエリを行えるメンバー、結果を受け取れるメンバー、およびクエリで参照される設定済みテーブルを所有するメンバーに、クエリログが送信されます。

クエリを行えるメンバーと結果を受け取れるメンバーは、クエリで参照される各設定済みテーブルのクエリログを受け取ります。双方が設定済みテーブルの所有者でない場合、設定済みテーブルの ID (`configuredTableID`) は表示されません。

1 人のメンバーが所有する複数の設定済みテーブルの関連付けがクエリで参照されている場合、そのメンバーは設定済みテーブルごとにクエリログを受け取ります。

AWS Clean Roomsでサポートされていない、またはサポートされている SQL を含むクエリログが作成されます。詳細については、「AWS Clean Rooms SQL リファレンス」を参照してください。

コラボレーションに関連付けられていない設定済みテーブルをクエリで参照する場合にもログが作成されます。

の SQL が正しくない場合、ログは作成されません AWS Clean Rooms。

クエリログには、クエリが成功したことやクエリ出力が送信されたことは記録されません。クエリログでは、クエリを行えるメンバーによってクエリが送信されたことが確認されます。クエリログは、クエリに でサポートされている SQL が含まれ AWS Clean Rooms 、コラボレーションに関連付けられた設定済みテーブルを参照していることも確認します。

Example

例えば、分析ルールへの準拠 AWS Clean Rooms を検証した後、クエリの処理中にクエリがキャンセルされた場合、ログは生成されません。

ロググループを削除した場合は、同じロググループ名 (コラボレーションのコラボレーション ID) を使用してロググループを手動で再作成する必要があります。または、メンバーシップのログの有効/無効を切り替えることもできます。

クエリログ記録の設定の詳細については、「AWS Clean Rooms でのコラボレーションの作成」を参照してください。

Amazon CloudWatch Logs の詳細については、「Amazon CloudWatch Logs ユーザーガイド」を参照してください。

クエリログの使用

メンバーが次のアクションを定期的に実行することを推奨します。

- クエリがコラボレーションに際して合意されたユースケースまたはクエリと一致していることを確認するために、コラボレーションで実行されるクエリを確認してください。

  最近のクエリを表示する方法の詳細については、「最近のクエリの表示」を参照してください。
- 設定済みテーブルの列がコラボレーションに際して合意されたものと一致していることを確認するために、コラボレーションメンバーの分析ルールとクエリで使用される設定済みテーブルの列を確認してください。

  設定した列を表示する方法の詳細については、「テーブルと分析ルールの表示」を参照してください。

セットアップ AWS Clean Rooms

以下のトピックでは、 の設定方法について説明します AWS Clean Rooms。

トピック

にサインアップする AWS

を含む を使用する前に AWS のサービス、 にサインアップ AWS Clean Roomsする必要があります AWS。

がない場合は AWS アカウント、次のステップ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、通話呼び出しを受け取り、電話キーパッドで検証コードを入力するように求められます。
3. にサインアップすると AWS アカウント、AWS アカウント 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。セキュリティのベストプラクティスとして、管理ユーザーに管理アクセスを割り当て、ルートユーザーアクセスが必要なタスクを実行する場合にのみ、ルートユーザーを使用してください。

のサービスロールの設定 AWS Clean Rooms

トピック

- 結果を受け取るサービスロールを作成する

管理者ユーザーの作成

を使用するには AWS Clean Rooms、管理者ユーザーを自分で作成し、管理者ユーザーを管理者グループに追加する必要があります。

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 Identity Center 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。	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。 AWS Command Line Interface
IAM 内 (非推奨)	長期認証情報を使用して AWSにアクセスする。	IAM ユーザーガイドの「最初の IAM 管理者のユーザーおよびグループの作成」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の管理」に従って、プログラムによるアクセスを設定します。

コラボレーションメンバー用の IAM ロールの作成

メンバーは、コラボレーションに参加している AWS 顧客です。

コラボレーションメンバー用の IAM ロールを作成するには

1. 「 AWS Identity and Access Management ユーザーガイド」の「ロールの作成」に従って、IAM ユーザーにアクセス許可を委任します。
2. ポリシーの作成ステップでは、ポリシーエディタ で JSON タブを選択し、コラボレーションメンバーに付与された機能に応じてポリシーを追加します。

   AWS Clean Rooms は、一般的なユースケースに基づいて以下の マネージドポリシーを提供します。

目的	使用
リソースとメタデータを表示する	AWS 管理ポリシー: AWSCleanRoomsReadOnlyAccess
Query	AWS 管理ポリシー: AWSCleanRoomsFullAccess
クエリを実行して結果を受け取る	AWS 管理ポリシー: AWSCleanRoomsFullAccess
コラボレーションリソースを管理するが、クエリはしない	AWS 管理ポリシー: AWSCleanRoomsFullAccessNoQuerying

   が提供するさまざまな 管理ポリシーの詳細については AWS Clean Rooms、「」を参照してください。AWS の マネージドポリシー AWS Clean Rooms

データを読み取るサービスロールの作成

AWS Clean Rooms は、サービスロールを使用してデータを読み込みます。

このサービスロールを作成するには、次の 2 つの方法があります。

... の場合	THEN
サービスロールを作成するために必要な IAM アクセス許可がある	AWS Clean Rooms コンソールを使用してサービスロールを作成します。
`iam:CreateRole` 、、`iam:CreatePolicy` および の`iam:AttachRolePolicy` アクセス許可がない または IAM ロールを手動で作成したい	次のいずれかを行います。 • サービスロールを作成するには、次の手順に従います。 • 次の手順を使用してサービスロールを作成するように管理者に依頼します。

データを読み取るサービスロールを作成するには

> **Note**
>
> ユーザーまたは IAM 管理者は、 AWS Clean Rooms コンソールを使用してサービスロールを作成するために必要なアクセス許可がない場合にのみ、この手順に従う必要があります。

1. 「 AWS Identity and Access Management ユーザーガイド」の「カスタム信頼ポリシーを使用したロールの作成 (コンソール) 」の手順に従います。
2. カスタム信頼ポリシー (コンソール) を使用したロールの作成の手順に従って、次のカスタム信頼ポリシーを使用します。

   > **Note**
   >
   > ロールを特定のコラボレーションメンバーシップのコンテキストでのみ使用できるようにするには、信頼ポリシーをさらに絞り込みます。詳細については、「サービス間の混乱した代理の防止」を参照してください。

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

```
            "Sid": "RoleTrustPolicyForCleanRoomsService",
            "Effect": "Allow",
            "Principal": {
                "Service": "cleanrooms.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

3. カスタム信頼ポリシー (コンソール) を使用したロールの作成手順に従って、次のアクセス許可ポリシーを使用します。

> **Note**
>
> 以下のポリシー例は、 AWS Glue メタデータとそれに対応する Amazon S3 データを読み取るのに必要なアクセス許可をサポートしています。ただし、S3 データの設定方法によっては、このポリシーを変更する必要が生じる場合があります。例えば、S3 データのカスタム KMS キーを設定している場合は、追加の AWS KMS アクセス許可でこのポリシーを修正する必要がある場合があります。
> AWS Glue リソースと基盤となる Amazon S3 リソースは、 AWS Clean Rooms コラボレーション AWS リージョン と同じ にある必要があります。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "NecessaryGluePermissions",
            "Effect": "Allow",
            "Action": [
                "glue:GetDatabase",
                "glue:GetDatabases",
                "glue:GetTable",
                "glue:GetTables",
                "glue:GetPartition",
                "glue:GetPartitions",
                "glue:BatchGetPartition"
            ],
            "Resource": [
                "arn:aws:glue:aws-region:accountId:database/database",
```

```
                "arn:aws:glue:aws-region:accountId:table/table",
                "arn:aws:glue:aws-region:accountId:catalog"
            ]
        },
{
            "Effect": "Allow",
            "Action": [
                "glue:GetSchema",
                "glue:GetSchemaVersion"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Sid": "NecessaryS3BucketPermissions",
            "Effect": "Allow",
            "Action": [
                "s3:GetBucketLocation",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::bucket"
            ],
            "Condition":{
                "StringEquals":{
                    "s3:ResourceAccount":[
                        "s3BucketOwnerAccountId"
                    ]
                }
            }
        },
        {
            "Sid": "NecessaryS3ObjectPermissions",
            "Effect": "Allow",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3::bucket/prefix/*"
            ],
            "Condition":{
                "StringEquals":{
                    "s3:ResourceAccount":[
```

```
                    "s3BucketOwnerAccountId"
                ]
            }
        }
    }

  ]
}
```

4. 各########を独自の情報に置き換えます。
5. カスタム信頼ポリシー (コンソール) を使用してロールを作成するの手順に引き続き従ってロールを作成します。

結果を受け取るサービスロールを作成する

> **Note**
>
> 結果のみを受信できるメンバー (コンソールでは、メンバーの機能は結果の受信のみ) の場合は、次の手順に従います。
> クエリと結果の受信の両方が可能なメンバーの場合 (コンソールでは、メンバーの機能はクエリと結果の受信の両方です) 、この手順をスキップできます。

結果のみを受信できるコラボレーションメンバーの場合、 はサービスロール AWS Clean Rooms を使用して、コラボレーション内のクエリされたデータの結果を指定された Amazon S3 バケットに書き込みます。

このサービスロールを作成するには、次の 2 つの方法があります。

... の場合	THEN
サービスロールを作成するために必要な IAM アクセス許可がある	AWS Clean Rooms コンソールを使用してサービスロールを作成します。
`iam:CreateRole` 、、`iam:CreatePolicy` および の`iam:AttachRolePolicy` アクセス許可がない または	次のいずれかを行います。 • サービスロールを作成するには、次の手順に従います。

... の場合	THEN
IAM ロールを手動で作成したい	• 次の手順を使用してサービスロールを作成するように管理者に依頼します。

結果を受け取るサービスロールを作成するには

> **Note**
>
> ユーザーまたは IAM 管理者は、 AWS Clean Rooms コンソールを使用してサービスロールを作成するために必要なアクセス許可がない場合にのみ、この手順に従う必要があります。

1. 「 AWS Identity and Access Management ユーザーガイド」の「カスタム信頼ポリシーを使用したロールの作成 (コンソール) 」の手順に従います。
2. カスタム信頼ポリシー (コンソール) を使用したロールの作成の手順に従って、次のカスタム信頼ポリシーを使用します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowIfExternalIdMatches",
            "Effect": "Allow",
            "Principal": {
                "Service": "cleanrooms.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "ArnLike": {
                    "sts:ExternalId":
 "arn:aws:*:region:*:dbuser:*/a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa*"
                }
            }
        },
        {
            "Sid": "AllowIfSourceArnMatches",
            "Effect": "Allow",
            "Principal": {
```

```
                "Service": "cleanrooms.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "ForAnyValue:ArnEquals": {
                    "aws:SourceArn": [
                        "arn:aws:cleanrooms:us-east-1:555555555555:membership/
a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa"
                    ]

                }
            }
        }
    ]
}
```

3. カスタム信頼ポリシーを使用したロールの作成 (コンソール) の手順に従って、次のアクセス許可ポリシーを使用します。

> **Note**
>
> 以下のポリシー例は、 AWS Glue メタデータとそれに対応する Amazon S3 データを読み取るのに必要なアクセス許可をサポートしています。ただし、S3 データの設定方法によっては、このポリシーを変更する必要が生じる場合があります。
> AWS Glue リソースと基盤となる Amazon S3 リソースは、 AWS Clean Rooms コラボレーション AWS リージョン と同じ にある必要があります。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetBucketLocation",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::bucket_name"
            ],
            "Condition": {
                "StringEquals": {
```

```
                    "aws:ResourceAccount":"accountId"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucket_name/optional_key_prefix/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:ResourceAccount":"accountId"
                }
            }
        }
    ]
}
```

4. 各########を自分の情報に置き換えます。

   - *region* – AWS リージョンの名前。例えば **us-east-1** です。
   - *a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa* – クエリが行えるメンバーのメンバーシップ ID。[メンバーシップ ID] はコラボレーションの [詳細] タブにあります。これにより、このメンバーがこのコラボレーションで分析を実行する場合にのみ、 AWS Clean Rooms がロールを引き受けるようになります。
   - *arn:aws:cleanrooms:us-east-1:555555555555:membership/a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa* – クエリを行えるメンバーの単一のメンバーシップ ARN。[メンバーシップ ARN] はコラボレーションの [詳細] タブにあります。これにより、このメンバーがこのコラボレーションで分析を実行した場合にのみ、 AWS Clean Rooms がロールを引き受けるようになります。
   - *bucket_name* – S3 バケットの Amazon リソースネーム (ARN)。[Amazon リソースネーム (ARN)] は Amazon S3 のバケットの [プロパティ] タブにあります。
   - *accountId* – S3 バケットが配置されている AWS アカウント ID。

     bucket_name/optional_key_prefix – S3 内の結果の送信先の Amazon リソースネーム (ARN)。[Amazon リソースネーム (ARN)] は Amazon S3 のバケットの [プロパティ] タブにあります。

5. カスタム信頼ポリシー (コンソール) を使用してロールを作成するの手順に引き続き従ってロールを作成します。

AWS Clean Rooms ML のサービスロールを設定する

トピック

- トレーニングデータを読み取るサービスロールの作成
- サービスロールを作成して類似セグメントを書き込む
- シードデータを読み取るサービスロールの作成

トレーニングデータを読み取るサービスロールの作成

AWS Clean Rooms は、サービスロールを使用してトレーニングデータを読み込みます。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、管理者にサービスロールの作成を依頼してください。

データセットをトレーニングするサービスロールの作成

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、[ポリシー] を選択します。
3. [ポリシーの作成] を選択します。
4. [ポリシーエディタ] で [JSON] タブを選択し、次のポリシーをコピーして貼り付けます。

> **Note**
>
> 以下のポリシー例は、 AWS Glue メタデータとそれに対応する Amazon S3 データを読み取るのに必要なアクセス許可をサポートしています。ただし、S3 データの設定方法によっては、このポリシーを変更する必要が生じる場合があります。このポリシーには、データを復号するための KMS キーは含まれていません。
> AWS Glue リソースと基盤となる Amazon S3 リソースは、 AWS Clean Rooms コラボレーション AWS リージョン と同じ にある必要があります。

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "glue:GetDatabase",
                "glue:GetDatabases",
                "glue:GetTable",
                "glue:GetTables",
                "glue:GetPartitions",
                "glue:GetPartition",
                "glue:BatchGetPartition",
                "glue:GetUserDefinedFunctions"
            ],
            "Resource": [
                "arn:aws:glue:region:accountId:database/databases",
                "arn:aws:glue:region:accountId:table/databases/tables",
                "arn:aws:glue:region:accountId:catalog",
                "arn:aws:glue:region:accountId:database/default"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "glue:CreateDatabase"
            ],
            "Resource": [
                "arn:aws:glue:region:accountId:database/default"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:s3:::bucket"
            ],
            "Condition":{
                "StringEquals":{
                    "s3:ResourceAccount":[
                        "accountId"
                    ]
```

```
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucketFolders/*"
            ],
            "Condition":{
                "StringEquals":{
                    "s3:ResourceAccount":[
                        "accountId"
                    ]
                }
            }
        }
    ]
}
```

KMS キーを使用してデータを復号する必要がある場合は、前のテンプレートに次の AWS KMS ステートメントを追加します。

```
{
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
            ],
            "Resource": [
                "arn:aws:kms:region:accountId:key/keyId"
            ],
            "Condition": {
                "ArnLike": {
                        "kms:EncryptionContext:aws:s3:arn":
 "arn:aws:s3:::bucketFolders*"
                }
            }
        }
    ]
}
```

5. [次へ] をクリックします。
6. [確認して作成] で [ポリシー名] と [説明] を入力し、[概要] を確認します。
7. [ポリシーの作成] を選択します。

   のポリシーを作成しました AWS Clean Rooms。
8. [アクセス管理] で、[ロール] を選択します。

   [ロール] を使用すると、短期間の認証情報を作成できるため、セキュリティ強化のためにお勧めです。[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。
9. [ロールの作成] を選択します。
10. [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。
11. 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。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowAssumeRole",
            "Effect": "Allow",
            "Principal": {
                "Service": "cleanrooms-ml.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEqualsIfExists": {
                    "aws:SourceAccount": ["accountId"]
                },
                "StringLikeIfExists": {
                    "aws:SourceArn": "arn:aws:cleanrooms-
ml:region:account:training-dataset/*"
                }
            }
        }
    ]
}
```

は常にお客様のアカウントSourceAccountです AWS 。SourceArn は特定のトレーニングデータセットに制限できますが、そのデータセットが作成された後に限られます。トレーニングデータセットの ARN を事前に知ることはできないため、ここではワイルドカードを指定します。

12. [次へ] を選択し、[アクセス許可を追加] で、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)。
13. 作成したポリシー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。
14. [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、クエリを実行して結果を受け取ることができるメンバーとメンバーロールに付与された passRole アクセス許可のパターンと一致している必要があります。

    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] を選択します。
15. のサービスロール AWS Clean Rooms が作成されました。

サービスロールを作成して類似セグメントを書き込む

AWS Clean Rooms は、サービスロールを使用して類似セグメントをバケットに書き込みます。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、管理者にサービスロールの作成を依頼してください。

サービスロールを作成して類似セグメントを書き込むには

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、[ポリシー] を選択します。
3. [ポリシーの作成] を選択します。
4. [ポリシーエディタ] で [JSON] タブを選択し、次のポリシーをコピーして貼り付けます。

> **Note**
>
> 以下のポリシー例は、 AWS Glue メタデータとそれに対応する Amazon S3 データを読み取るのに必要なアクセス許可をサポートしています。ただし、S3 データの設定方法によっては、このポリシーを変更する必要が生じる場合があります。このポリシーには、データを復号するための KMS キーは含まれていません。
> AWS Glue リソースと基盤となる Amazon S3 リソースは、 AWS Clean Rooms コラボレーション AWS リージョン と同じ にある必要があります。

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

```
                        "accountId"
                    ]
                }
            }
        }
    ]
}
```

KMS キーを使用してデータを暗号化する必要がある場合は、次の AWS KMS ステートメントをテンプレートに追加します。

```
{
            "Effect": "Allow",
            "Action": [
                "kms:Encrypt",
                "kms:GenerateDataKey*",
                "kms:ReEncrypt*",
            ],
            "Resource": [
                "arn:aws:kms:region:accountId:key/keyId"
            ],
            "Condition": {
                "ArnLike": {
                        "kms:EncryptionContext:aws:s3:arn":
 "arn:aws:s3:::bucketFolders*"
                }
            }
        }
    ]
}
```

KMS キーを使用してデータを復号する必要がある場合は、テンプレートに次の AWS KMS ステートメントを追加します。

```
{
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "kms:DescribeKey"
            ],
            "Resource": [
                "arn:aws:kms:region:accountId:key/keyId"
```

```
            ],
            "Condition": {
                "ArnLike": {
                        "kms:EncryptionContext:aws:s3:arn":
 "arn:aws:s3:::bucketFolders*"
                }
            }
        }
  ]
}
```

5. [次へ] をクリックします。
6. [確認して作成] で [ポリシー名] と [説明] を入力し、[概要] を確認します。
7. [ポリシーの作成] を選択します。

   のポリシーを作成しました AWS Clean Rooms。
8. [アクセス管理] で、[ロール] を選択します。

   [ロール] を使用すると、短期間の認証情報を作成できるため、セキュリティ強化のためにお勧めです。[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。
9. [ロールの作成] を選択します。
10. [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。
11. 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。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowAssumeRole",
            "Effect": "Allow",
            "Principal": {
                "Service": "cleanrooms-ml.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEqualsIfExists": {
                    "aws:SourceAccount": ["accountId"]

                },
                "StringLikeIfExists": {
```

```
                    "aws:SourceArn": "arn:aws:cleanrooms-
ml:region:account:configured-audience-model/*"
                }
            }
        }
    ]
}
```

は常にお客様のアカウントSourceAccountです AWS 。SourceArn は特定のトレーニングデータセットに制限できますが、そのデータセットが作成された後に限られます。トレーニングデータセットの ARN を事前に知ることはできないため、ここではワイルドカードを指定します。

12. [次へ] をクリックします。
13. 作成したポリシー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。
14. [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、クエリを実行して結果を受け取ることができるメンバーとメンバーロールに付与された passRole アクセス許可のパターンと一致している必要があります。

    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] を選択します。
15. のサービスロール AWS Clean Rooms が作成されました。

シードデータを読み取るサービスロールの作成

AWS Clean Rooms はサービスロールを使用してシードデータを読み込みます。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、管理者にサービスロールの作成を依頼してください。

シードデータを読み取るサービス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、[ポリシー] を選択します。
3. [ポリシーの作成] を選択します。
4. [ポリシーエディタ] で [JSON] タブを選択し、次のポリシーをコピーして貼り付けます。

> **Note**
>
> 以下のポリシー例は、 AWS Glue メタデータとそれに対応する Amazon S3 データを読み取るのに必要なアクセス許可をサポートしています。ただし、S3 データの設定方法によっては、このポリシーを変更する必要が生じる場合があります。このポリシーには、データを復号するための KMS キーは含まれていません。
> AWS Glue リソースと基盤となる Amazon S3 リソースは、 AWS Clean Rooms コラボレーション AWS リージョン と同じ にある必要があります。

```
{
    "Version": "2012-10-17",
    "Statement": [
    {
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket",
            ],
            "Resource": [
                "arn:aws:s3:::buckets"
            ],
            "Condition":{
                "StringEquals":{
                    "s3:ResourceAccount":[
                        "accountId"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
```

```
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::bucketFolders/*"
            ],
            "Condition":{
                "StringEquals":{
                    "s3:ResourceAccount":[
                        "accountId"
                    ]
                }
            }
        }
  ]
}
```

KMS キーを使用してデータを復号する必要がある場合は、テンプレートに次の AWS KMS ステートメントを追加します。

```
{
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "kms:DescribeKey"
            ],
            "Resource": [
                "arn:aws:kms:region:accountId:key/keyId"
            ],
            "Condition": {
                "ArnLike": {
                        "kms:EncryptionContext:aws:s3:arn":
 "arn:aws:s3:::bucketFolders*"
                }
            }
        }
  ]
}
```

5. [次へ] をクリックします。
6. [確認して作成] で [ポリシー名] と [説明] を入力し、[概要] を確認します。
7. [ポリシーの作成] を選択します。

のポリシーを作成しました AWS Clean Rooms。

8. [アクセス管理] で、[ロール] を選択します。

   [ロール] を使用すると、短期間の認証情報を作成できるため、セキュリティ強化のためにお勧めです。[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。

9. [ロールの作成] を選択します。

10. [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。

11. 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。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowAssumeRole",
            "Effect": "Allow",
            "Principal": {
                "Service": "cleanrooms-ml.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEqualsIfExists": {
                    "aws:SourceAccount": ["accountId"]

                },
                "StringLikeIfExists": {
                    "aws:SourceArn": "arn:aws:cleanrooms-
ml:region:account:audience-generation-job/*"
                }
            }
        }
    ]
}
```

   は常にお客様のアカウントSourceAccountです AWS 。SourceArn は特定のトレーニングデータセットに制限できますが、そのデータセットが作成された後に限られます。トレーニングデータセットの ARN を事前に知ることはできないため、ここではワイルドカードを指定します。

12. [次へ] をクリックします。

13. 作成したポリシー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。
14. [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、クエリを実行して結果を受け取ることができるメンバーとメンバーロールに付与された `passRole` アクセス許可のパターンと一致している必要があります。

    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] を選択します。
15. のサービスロール AWS Clean Rooms が作成されました。

AWS Clean Rooms でのコラボレーションの作成

コラボレーションは、設定済みのテーブルに対してメンバーが SQL クエリを実行できる AWS Clean Rooms 内の安全な論理的境界です。

AWS Clean Rooms のメンバーであれば誰でもコラボレーションを作成できます。

コラボレーションクリエーターは、クエリを実行して結果を受け取るメンバーを 1 人だけ指定できます。ただし、クエリを行えるメンバーがクエリ結果にアクセスすることを、コラボレーションクリエーターが望まない場合もあります。その場合、コラボレーションクリエーターは、クエリを行えるメンバー を 1 人と、結果を受け取れるメンバーをもう 1 人指定できます。

ほとんどの場合、クエリを行えるメンバーは、クエリの計算コストを負担するメンバーでもあります。ただし、コラボレーションクリエーターは、別のメンバーがクエリの計算コストを負担するよう設定することもできます。

AWS SDK を使用してコラボレーションを作成する方法については、「AWS Clean Rooms API リファレンス」を参照してください。

トピック

コラボレーションを作成する

開始する前に、以下の前提条件を満たしていることを確認してください。

- コラボレーションに招待する各メンバーの名前と AWS アカウント ID を把握していること。
- 各メンバーの名前と AWS アカウント ID をコラボレーションのすべてのメンバーと共有する権限があること。

> **Note**
>
> コラボレーションの作成後にメンバーを追加することはできません。

AWS Clean Rooms コンソールを使用してコラボレーションを作成するには

1. コラボレーションクリエーターとして機能する AWS アカウントで AWS Management Console にサインインし、AWS Clean Rooms コンソールを開きます。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. 右上隅にある [コラボレーションを作成] を選択します。
4. [ステップ 1: コラボレーションの定義] で以下の操作を行います。

   a. [詳細] で、コラボレーションの [名前] と [説明] を入力します。

      この情報は、コラボレーションに参加するよう招待されたコラボレーションメンバーに表示されます。メンバーは、[名前] と [説明] でコラボレーションが何を指しているのかを把握できます。

   b. [メンバー] で以下の操作を行います。

      i. [メンバー 1: 自分] の [メンバー表示名] に、コラボレーションで表示する自身のメンバー表示名を入力します。

      > **Note**
      >
      > [メンバー AWS アカウント ID] には、コラボレーションクリエーターの AWS アカウント ID が自動的に挿入されます。

      ii. [メンバー 2] で、コラボレーションに招待するメンバーの [メンバー表示名] と [メンバー AWS アカウント ID] を入力します。

      メンバー表示名とメンバー AWS アカウント ID は、コラボレーションに招待されたすべてのユーザーに表示されます。これらのフィールドに入力して保存した値を、後から編集することはできません。

      > **Note**
      >
      > コラボレーションメンバーのメンバー AWS アカウント ID とメンバー表示名は、コラボレーションに参加している招待されたアクティブなコラボレーター全員に表示されるため、この点について本人に知らせておく必要があります。

iii. 別のメンバーを追加する場合は、[別のメンバーを追加] を選択します。データを寄稿できる各招待メンバーの [メンバー表示名] と [メンバー AWS アカウント ID] を入力します。

c. [メンバー能力] で以下の操作を行います。

目的	操作
コラボレーション内のデータにクエリを実行し、結果を受け取る	1. クエリを実行するメンバーとして自分自身を選択します。 2. 結果を受け取るメンバーは、デフォルト設定 ([クエリの実行者と同じ]) のままにしておきます。
自分がコラボレーション内のデータにクエリを実行し、結果の受け取りは別のメンバーに割り当てる	1. クエリを実行するメンバーとして自分自身を選択します。 2. ドロップダウンリストから、結果を受け取るメンバーを選択します。
自分がコラボレーションでのクエリ結果を受け取り、データに対するクエリの実行は別のメンバーに割り当てる	1. ドロップダウンリストから、クエリを実行するメンバーを選択します。 2. ドロップダウンリストから、結果を受け取るメンバーとして自分自身を選択します。
自分がコラボレーションを作成して管理し、データに対するクエリの実行と結果の受け取りはそれぞれ別のメンバーに割り当てる	1. ドロップダウンリストから、クエリを実行するメンバーを選択します。 2. ドロップダウンリストから、結果を受け取るメンバーを選択します。

d. [支払い設定] で、以下のいずれかを行います。

目的	操作
クエリを実行するメンバーを、クエリの計算コストを負担するメンバーとして割り当てる	クエリのコストを負担するメンバーは、デフォルト設定 ([クエリの実行者と同じ]) のままにしておきます。
クエリの計算コストの負担を別のメンバーを割り当てる	ドロップダウンリストから、クエリのコストを負担するメンバーを選択します。

e. [クエリログ記録] を有効にする場合は、[このコラボレーションでクエリのログ記録をサポートする] チェックボックスをオンにします。

f. [暗号コンピューティング] 機能を有効にする場合は、[このコラボレーションで暗号コンピューティングをサポートする] チェックボックスを選択し、次の [暗号コンピューティングパラメータ] を選択します。

- cleartext列を許可

  暗号化されたテーブルでcleartext列を許可しない場合は、[いいえ] を選択します。

  暗号化されたテーブルでcleartext列を許可する場合は、[はい] を選択します。

  特定の列で SUM または AVG を実行するには、その列がcleartextである必要があります。

- 重複を許可

  1 つのfingerprint列で重複するエントリを許可しない場合は、[いいえ] を選択します。

  1 つのfingerprint列で重複するエントリを許可する場合は、[はい] を選択します。

- 名前の異なる列の JOIN を許可

  名前の異なるfingerprint列を結合しない場合は、[いいえ] を選択します。

  名前の異なるfingerprint列を結合する場合は、[はい] を選択します。

- NULL 値を保存

  NULL 値を保持しない場合は [いいえ] を選択します。NULL 値は暗号化されたテーブルで NULL として表示されません。

NULL 値を保持する場合は [はい] を選択します。NULL 値は暗号化されたテーブルで NULL として表示されます。

暗号コンピューティングパラメータの詳細については、「暗号コンピューティングパラメータ」を参照してください。

AWS Clean Rooms で使用するデータを暗号化する方法については、「Cryptographic Computing for Clean Rooms による暗号化データテーブルの準備」を参照してください。

> **Note**
>
> 次のステップを完了する前に、これらの設定を注意深く確認してください。コラボレーションの作成後に編集できるのは、コラボレーションの名前、説明、およびクエリログを Amazon CloudWatch Logs に保存するかどうかのみです。

g. コラボレーションリソースで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。

h. [Next] (次へ) を選択します。

5. [ステップ 2: メンバーシップを設定] で、以下の操作を行います。

a. いずれかのオプションを選択します。

選択内容	結果
はい、今すぐメンバーシップを作成してご入会ください	コラボレーションとメンバーシップの両方が作成されます。 コラボレーションのステータスはアクティブになります。
いいえ、後でメンバーシップを作成します	コラボレーションのみが作成されます。 コラボレーションのステータスは非アクティブになります。

b. 自分が結果を受け取るメンバーの場合は、[クエリ結果のデフォルト設定] で次のいずれかのオプションを選択します。

選択内容	操作
[今すぐデフォルトの設定を設定] チェックボックスをオン (デフォルトで選択されています)	1. [Amazon S3 内の結果の送信先] に、Amazon S3 の送信先を入力します。 2. [結果フォーマット] で、[CSV] または [Parquet] を選択します。
[今すぐデフォルトの設定を設定] チェックボックスをオフ	コラボレーションのみが作成されます。 コラボレーションのステータスは非アクティブになります。

c. ステップ 4.e でクエリログ記録を有効にすることを選択した場合は、[Amazon CloudWatch Logs のログストレージ] で以下のいずれかのオプションを選択します。

選択内容	結果
オンにする	関連するクエリログが Amazon CloudWatch Logs に保存されます。 各メンバーは、自分が開始したクエリ、または自分のデータを含むクエリの口グのみを受け取ることができます。 また、結果を受け取れるメンバーは、自分のデータがクエリで利用されていない場合でも、コラボレーションで実行されたすべてのクエリの口グを受け取ります。
オフにする	関連するクエリログは、Amazon CloudWatch Logs アカウントに保存されません。

> **Note**
>
> [クエリログ記録] を有効にした後、ログストレージが設定されて Amazon CloudWatch Logs でログの受信が開始されるまでに数分かかることがあります。この短い間、クエリを行えるメンバーがクエリを実行しても、実際にはログが送信されない場合があります。

d. メンバーシップリソースで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. 自分がクエリのコストを負担するメンバーである場合は、[このコラボレーションのクエリの計算コストを負担することに同意する] チェックボックスを選択して同意します。

> **Note**
>
> 続行するには、このチェックボックスをオンにする必要があります。
> 料金の計算方法については、「の料金 AWS Clean Rooms」を参照してください。

クエリの計算コストを負担するメンバーではあるが、クエリを行えるメンバーではない場合は、AWS Budgets を使用して AWS Clean Rooms の予算を設定し、最大予算に達した時点で通知が送られるようにしておくことをお勧めします。詳細については、「AWS Cost Management ユーザーガイド」の「AWS Budgets を用いてコストを管理する」を参照してください。通知の設定については、「AWS Cost Management ユーザーガイド」の「予算の通知に関する Amazon SNS トピックを作成する」を参照してください。予算の上限に達した場合は、クエリを行えるメンバーに連絡するか、コラボレーションから退出できます。コラボレーションから退出すると、それ以上クエリの実行は許可されなくなるため、クエリの計算コストも請求されなくなります。

f. [Next] (次へ) を選択します。

6. [ステップ 3: 確認と作成] で、以下の操作を行います。

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. 次のいずれかを選択します。

前のステップでの選択内容	選択する項目
コラボレーションと共にメンバーシップを作成 ([はい、今すぐメンバーシップを作成してご入会ください])	コラボレーションとメンバーシップを作成
コラボレーションは作成するが、現時点ではメンバーシップを作成しない ([いいえ、後でメンバーシップを作成します])	コラボレーションを作成

コラボレーションが正常に作成されると、[コラボレーション] の下にコラボレーションの詳細ページが表示されます。

次のステップ

これで次の作業に進むことができます。

- AWS Clean Rooms でクエリ対象となるデータテーブルを準備する (自身のデータにクエリを実行する場合は省略可能)
- 設定済みテーブルをコラボレーションに関連付ける (自身のデータにクエリを実行する場合は省略可能)
- 設定済みテーブルに分析ルールを設定する (自身のデータにクエリを実行する場合は省略可能)
- メンバーシップを作成してコラボレーションに参加する
- コラボレーションを管理する

メンバーシップの作成とコラボレーションへの参加

メンバーシップは、 AWS Clean Roomsでメンバーがコラボレーションに参加するときに作成されるリソースです。

コラボレーションには、データに対してクエリを行えるメンバー、クエリの結果を受け取れるメンバー、またはその両方として参加できます。クエリの計算コストを負担するメンバーとしてコラボレーションに参加することもできます。メンバー全員がデータを寄稿できます。

AWS SDK を使用してメンバーシップを作成し、コラボレーションに参加する方法については、「AWS Clean Rooms API リファレンス」を参照してください。

トピック

メンバーシップを作成してコラボレーションに参加する

メンバーシップを作成してコラボレーションに参加するには

1. にサインイン AWS Management Console し、メンバー でAWS Clean Rooms コンソールを開きます AWS アカウント。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. [参加可能] タブの [参加可能なコラボレーション] で、コラボレーションの [名前] を選択します。
4. コラボレーションの詳細ページに、自分のメンバー情報や他のメンバーのリストなど、コラボレーションの詳細情報が表示されます。

   コラボレーションの各メンバーの AWS アカウント IDs が、コラボレーションに入力する予定の ID であることを確認します。
5. [メンバーシップを作成] を選択します。
6. メンバーシップの作成ページの概要で、コラボレーション名 、コラボレーションの説明 、コラボレーション作成者の AWS アカウント ID 、メンバーの能力 、およびクエリの支払いを行うメンバーの AWS アカウント ID を表示します。
7. コラボレーションクリエーターがクエリログ記録 を有効にすることを選択した場合は、Amazon CloudWatch Logs のログストレージに次のいずれかのオプションを選択します。

選択内容	結果
オンにする	関連するクエリログは Amazon CloudWatch Logs に保存されます。 各メンバーは、自分が開始したクエリ、または自分のデータを含むクエリのログのみを受け取ることができます。 結果を受け取ることができるメンバーは、データがクエリでアクセスされていない場合でも、コラボレーションで実行されるすべてのクエリのログも受け取ります。
オフにする	関連するクエリログは Amazon CloudWatch Logs アカウントに保存されません。

> **Note**
>
> クエリログを有効にした後、ログストレージがセットアップされ、Amazon CloudWatch Logs でログの受信が開始されるまでに数分かかることがあります。この短い間、クエリを行えるメンバーがクエリを実行しても、実際にはログが送信されない場合があります。

8. [自身のメンバー能力] に [結果を受け取る] が含まれる場合は、以下の操作を行います。

   a. [クエリ結果設定] で以下を行います。

      i. Amazon S3 の送信先を入力して [Amazon S3 内の結果の送信先] を指定するか、[S3 を参照] を選択して利用可能な S3 バケットのリストから選択します。

         Example

         例：**`s3://bucket/prefix`**

      ii. [結果フォーマット] ([CSV] または [Parquet]) を選択します。

   b. [サービスアクセス] で [[新しいサービスロールを作成して使用] または [既存のサービスロールを使用] を選択します。

> **Note**
>
> 既存のサービスロールを選択するか、新しいサービスロールを作成するためのアクセス許可が必要です。詳細については、「結果を受け取るサービスロールを作成する」を参照してください。

9. メンバーシップリソースで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。
10. コラボレーションクリエーターからクエリのコストを負担するメンバーとして指定されている場合は、[このコラボレーションのクエリの計算コストを負担することに同意する]、チェックボックスを選択して同意します。

> **Note**
>
> 続行するには、このチェックボックスをオンにする必要があります。
> 料金の計算方法の詳細については、「の料金 AWS Clean Rooms」を参照してください。

クエリコンピューティングコストを負担するメンバーで、 をクエリできるメンバーではない場合は、 AWS Budgets を使用して の予算を設定し、最大予算に達したら通知 AWS Clean Rooms を受け取ることをお勧めします。詳細については、「AWS Cost Management ユーザーガイド」の「AWS Budgetsを用いてコストを管理する」を参照してください。通知の設定の詳細については、「AWS Cost Management ユーザーガイド」の「予算の通知に関する Amazon SNS トピックを作成する」を参照してください。予算の上限に達した場合は、クエリを行えるメンバーに連絡するか、コラボレーションから退出できます。コラボレーションから退出すると、それ以上クエリの実行は許可されなくなるため、クエリの計算コストも請求されなくなります。

11. メンバーシップを作成してコラボレーションに参加してもよい場合は、[メンバーシップを作成] を選択します。

コラボレーションメタデータへの読み取りアクセスが付与されます。これには、他のメンバーのすべての名前と AWS アカウント ID に加え、コラボレーションの表示名や説明などの情報が含まれます。

コラボレーションから退出する方法については、「コラボレーションからの退出」を参照してください。

次のステップ

これで次の作業に進むことができます。

- でクエリするデータテーブルを準備します AWS Clean Rooms。 (自身のデータにクエリを実行する場合は省略可能)
- 設定済みテーブルをコラボレーションに関連付ける
- 設定済みテーブルに分析ルールを設定する

でのクエリ用のデータテーブルの準備 AWS Clean Rooms

> **Note**
>
> データテーブルの準備は、コラボレーションに参加する前でも後でも行うことができます。テーブルの準備が完了したら、そのテーブルのプライバシーニーズが同じである限り、複数のコラボレーションでテーブルを再利用できます。

コラボレーションのメンバーとして、 をクエリできる AWS Clean Rooms コラボレーションメンバーがデータテーブルをクエリする前に、データテーブルを準備する必要があります。

ユースケースで独自のデータを持ち込む必要がない場合は、この手順をスキップできます。

データテーブルが既に でカタログ化されている場合は AWS Glue、「」に進みますAWS Clean Roomsでの設定済みテーブルの作成。

データテーブルを準備するには、以下のステップに従います。

- ステップ 1: 前提条件を満たす
- ステップ 2: (オプション) 暗号化コンピューティング用のデータを準備する
- ステップ 3: データテーブルを Amazon S3 にアップロードする
- ステップ 4: AWS Glue テーブルを作成する
- 次のステップ

クエリに使用できるデータ形式の詳細については、「のデータ形式 AWS Clean Rooms」を参照してください。

ステップ 1: 前提条件を満たす

で使用するデータテーブルを準備するには AWS Clean Rooms、次の前提条件を満たす必要があります。

- データセットが、AWS Clean Roomsでサポートされているデータ形式のいずれかで保存されていること。
- データテーブルは でカタログ化 AWS Glue され、 でサポートされているデータ型 AWS Clean Roomsを使用する必要があります。

- すべてのデータテーブルは、コラボレーションが作成されたのと同じ AWS リージョン の Amazon Simple Storage Service (Amazon S3) に保存する必要があります。
- は、コラボレーションが作成されたリージョンと同じリージョンに存在する AWS Glue Data Catalog 必要があります。
- はメンバーシップ AWS アカウント と同じ にある AWS Glue Data Catalog 必要があります。
- Amazon S3 バケットを に登録することはできません AWS Lake Formation。
- コラボレーションクリエーターが AWS Clean Roomsでコラボレーションを設定ししていること。詳細については、「AWS Clean Rooms でのコラボレーションの作成」を参照してください。
- コラボレーションクリエーターから、コラボレーションの参加者としてコラボレーション ID が送信されていること。

ステップ 2: (オプション) 暗号化コンピューティング用のデータを準備する

(オプション) 暗号化コンピューティングを使用していて、データテーブルに暗号化が必要な機密情報が含まれている場合は、C3R 暗号化クライアントを使用してデータテーブルを暗号化する必要があります。

暗号化コンピューティング用のデータを準備するには、「Cryptographic Computing for Clean Rooms による暗号化データテーブルの準備」の手順に従ってください。

ステップ 3: データテーブルを Amazon S3 にアップロードする

> **Note**
>
> 暗号化されたデータテーブルをコラボレーションで使用する場合は、データテーブルを Amazon S3 にアップロードする前に、まずデータを暗号化コンピューティング用に暗号化する必要があります。詳細については、「Cryptographic Computing for Clean Rooms による暗号化データテーブルの準備」を参照してください。

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 Management Console 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。

   フォルダを選択すると、データが表示されます。

ステップ 4: AWS Glue テーブルを作成する

データ AWS Glue テーブルが既にある場合は、このステップをスキップできます。

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ーを にセットアップします。詳細については、「 ユーザーガイド」の「 でのクローラーの定義 AWS Glue」を参照してください。 AWS Glue

サポートされている AWS Glue Data Catalog データ型の詳細については、「」を参照してくださいサポートされているデータ型。

> **Note**
>
> AWS Clean Rooms は現在、 に登録されている S3 バケットをサポートしていません AWS Lake Formation。

次の手順では、 AWS Glue テーブルを作成する方法について説明します。 AWS Key Management Service （ AWS KMS) キーで暗号化された AWS Glue Data Catalog オブジェクトを使用する場合は、その暗号化されたテーブルへのアクセスを許可するように KMS キーのアクセス許可ポリシーを設定する必要があります。詳細については、「AWS Glue デベロッパーガイド」の「AWS Glue での暗号化のセットアップ」を参照してください。

AWS Glue テーブルを作成するには

1. 「 ユーザーガイド」の「 AWS Glue コンソールでのクローラーの操作」の手順に従います。 AWS Glue
2. AWS Glue データベース名と AWS Glue テーブル名を書き留めます。

次のステップ

これで、データテーブルを準備できたので、次の準備が整いました。

- 設定済みテーブルを作成する
- ML モデルを作成する

のデータ形式 AWS Clean Rooms

でクエリに使用するデータセット AWS Clean Rooms は、通常、他のアプリケーションで使用するのと同じタイプのデータセットです。例えば、Amazon Athena 、Amazon EMR、Amazon Redshift Spectrum、および Amazon では、同じタイプのデータセットが使用されます QuickSight。Amazon Simple Storage Service (Amazon S3) から直接、元の形式のデータにクエリを実行できます。

データをクエリするには、データセットが が AWS Clean Rooms サポートする形式である必要があります。データセットと AWS Clean Rooms クラスターを含む Amazon S3 バケットは、同じ にある必要があります AWS リージョン。

サポートされているデータ形式

AWS Clean Rooms では、次の構造化形式がサポートされています。

- Apache Iceberg テーブル
- Parquet
- RCFile
- TextFile
- SequenceFile
- RegexSerde
- OpenCSV
- AVRO
- JSON

> **Note**
>
> テキストファイル内の `timestamp` 値は、`yyyy-MM-dd HH:mm:ss.SSSSSS` の形式である必要があります。例えば、`2017-05-01 11:30:59.000000` です。

Apache Parquet など、列指向ストレージファイル形式を使用することをお勧めします。列指向ストレージファイル形式により、必要な列のみを選択し、Amazon S3 からのデータ転送を最小限に抑えることができます。最適なパフォーマンスを得るには、大きなオブジェクトを 100 MB ~ 1 GB のオブジェクトに分割する必要があります。

サポートされているデータ型

で最適なエクスペリエンスを得るには AWS Clean Rooms、すべてのデータを でカタログ化する必要があります AWS Glue。詳細については、「AWS Glue デベロッパーガイド」の「AWS Glue Data Catalogの開始方法」を参照してください。

AWS Clean Rooms は、次の AWS Glue Data Catalog データ型をサポートしています。

- bigint
- boolean
- char
- date
- decimal
- double
- float
- int
- 以下のようなネストされたデータ型
  - array
  - map
  - struct
- smallint
- string
- timestamp
- varchar

AWS Clean Rooms は以下をサポートしていません。

- バイナリ
- interval

のファイル圧縮タイプ AWS Clean Rooms

ストレージスペースの縮小、パフォーマンスの向上、コストの最小化を行うため、データセットを圧縮することを強くお勧めします。

AWS Clean Rooms は、ファイル拡張子に基づいてファイル圧縮タイプを認識し、次の表に示す圧縮タイプと拡張子をサポートします。

圧縮アルゴリズム	ファイル拡張子
GZIP	.gz
Bzip2	.bz2
Snappy	.snappy

さまざまなレベルで圧縮を適用できます。通常、ファイル全体を圧縮するか、ファイル内の個々のブロックを圧縮します。ファイルレベルで列形式を圧縮しても、パフォーマンス上の利点はありません。

のサーバー側の暗号化 AWS Clean Rooms

> **Note**
>
> 暗号化コンピューティングを必要とするユースケースで、サーバー側の暗号化が代わりの役割を果たすことはありません。

AWS Clean Rooms は、次の暗号化オプションを使用して暗号化されたデータセットを透過的に復号します。

- SSE-S3 – Amazon S3 によって管理される AES-256 暗号化キーを使用したサーバー側暗号化。
- SSE-KMS – によって管理されるキーによるサーバー側の暗号化 AWS Key Management Service

SSE-S3 を使用するには、設定済みテーブルをコラボレーションに関連付けるために使用される AWS Clean Rooms サービスロールに KMS 復号アクセス許可が必要です。SSE-KMS を使用するには、KMS キーポリシーで AWS Clean Rooms サービスロールの復号も許可する必要があります。

AWS Clean Rooms は Amazon S3 クライアント側の暗号化をサポートしていません。サーバー側の暗号化の詳細については、「Amazon Simple Storage Service ユーザーガイド」の「サーバー側の暗号化によるデータの保護」を参照してください。

でのApache Icebergテーブルの使用 AWS Clean Rooms

Apache Iceberg は、データレイクのオープンソーステーブル形式です。 は、Apache Icebergメタデータに保存されている統計を使用してクエリプランを最適化し、クリーンルームのクエリ処理中のファイルスキャンを減らす AWS Clean Rooms ことができます。詳細については、Apache Iceberg のドキュメントを参照してください。

Iceberg テーブル AWS Clean Rooms で を使用する場合は、次の点を考慮してください。

- 内のテーブル AWS Glue Data Catalog のみ – Apache Icebergテーブルは、オープンソースのグルーカタログ実装 AWS Glue Data Catalog に基づいて で定義する必要があります。 https://iceberg.apache.org/docs/latest/aws/#glue-catalog
- Parquet ファイル形式 – Parquet データファイル形式の Iceberg テーブル AWS Clean Rooms のみをサポートします。
- GZIP および Snappy 圧縮 – GZIP とSnappy圧縮を備えた Parquet AWS Clean Rooms をサポートします。
- Iceberg バージョン – バージョン 1 およびバージョン 2 の Iceberg テーブルに対するクエリの実行 AWS Clean Rooms をサポートします。
- パーティション – でApache Icebergテーブルのパーティションを手動で追加する必要はありません AWS Glue。 はApache Icebergテーブル内の新しいパーティションを自動的に AWS Clean Rooms 検出し、テーブル定義のパーティションを更新するための手動操作は必要ありません。Iceberg パーティションは、 AWS Clean Rooms テーブルスキーマでは通常の列として表示され、設定済みテーブルスキーマではパーティションキーとして個別に表示されません。
- 制限
  - 新しい Iceberg テーブルのみ

    Apache Iceberg テーブルから変換された Apache Parquet テーブルはサポートされていません。
  - タイムトラベルクエリ

    AWS Clean Rooms は、Apache Icebergテーブルを使用したタイムトラベルクエリをサポートしていません。
  - Athena エンジンバージョン 2

Athena エンジンバージョン 2 で作成されたIceberg テーブルはサポートされていません。

- ファイル形式

  Avro および最適化された行列 (ORC) のファイル形式はサポートされていません。

- 圧縮

  Parquet の Zstandard (Zstd) 圧縮はサポートされていません。

Iceberg テーブルでサポートされているデータ型

AWS Clean Rooms は、次のデータ型を含むIcebergテーブルをクエリできます。

- boolean
- date
- decimal
- double
- float
- int
- list
- long
- map
- string
- struct
- timestamp without time zone

Iceberg のデータ型の詳細については、Apache Iceberg ドキュメントで Iceberg のスキーマを参照してください。

Cryptographic Computing for Clean Rooms による暗号化データテーブルの準備

Cryptographic Computing for Clean Rooms (C3R) は の機能です AWS Clean Rooms。C3R を使用して、任意の当事者やコラボレーション AWS Clean Rooms で学習できる内容を暗号 AWS 的に制限できます。

データテーブルを Amazon Simple Storage Service (Amazon S3) にアップロードする前に、クライアント側の暗号化ツールである C3R 暗号化クライアントを使用してデータテーブルを暗号化できます。

詳細については、「Cryptographic Computing for Clean Rooms」を参照してください。

C3R で暗号化されたデータテーブルを準備するには、以下のステップに従います。

ステップ

ステップ 1: 前提条件を満たす

C3R で使用するデータテーブルを準備するには、以下の前提条件を満たす必要があります。

- の Cryptographic Computing for Clean Roomsリポジトリにアクセスできます GitHub。

  https://github.com/aws/c3r
- C3R 暗号化クライアントを使用するための AWS 認証情報を設定しました。これらの認証情報は、C3R 暗号化クライアントがコラボレーションメタデータを取得するための AWS Clean

Rooms への読み取り専用 API コールに使用されます。詳細については、「AWS Command Line Interface バージョン 2 用ユーザーガイド」の「AWS CLIを設定する」を参照してください。

- 使用中のマシンに Java Runtime Environment (JRE) 11 以降がインストールされていること。
  - 推奨される Java Runtime Environment、Amazon Corretto 11 以降は、https://aws.amazon.com/corretto からダウンロードできます。
  - Java Development Kit (JDK) には、同じバージョンの対応する JRE が含まれています。ただし、Cryptographic Computing for Clean Rooms (C3R) 暗号化クライアントを実行するために、JDK の追加機能は必要ありません。
- 表形式のデータファイル (.csv) または Parquet ファイル (.parquet) がローカルに保存されていること。
- 自分またはコラボレーションに参加している他のメンバーが、共有シークレットキーを作成できること。詳細については、「ステップ 5: 共有シークレットキーを作成する」を参照してください。
- コラボレーションクリエーターは、コラボレーションのために暗号化コンピューティングを有効に AWS Clean Rooms したコラボレーションを で作成しました。詳細については、「AWS Clean Rooms でのコラボレーションの作成」を参照してください。
- コラボレーションクリエーターから、コラボレーションの参加者としてコラボレーション ID が送信されていること。送信された招待状にはコラボレーションの Amazon リソースネーム (ARN) が含まれ、コラボレーション ID が示されています。

ステップ 2: C3R 暗号化クライアントをダウンロードする

から C3R 暗号化クライアントをダウンロードするには GitHub

1. Cryptographic Computing for Clean Rooms AWS GitHubリポジトリに移動します。https://github.com/aws/c3r
2. ファイルを選択してダウンロードします。

   ソースコード、ライセンス、および関連資料は、GitHub リポジトリのランディングページから複製、または zip ファイルとしてダウンロードできます (リポジトリのコンテンツリストの右上にある [Code] ボタンを参照してください)。

   最新の署名済み C3R 暗号化クライアントの Java Executable File (つまり、コマンドラインインターフェースアプリケーション) は、GitHub リポジトリの [Releases] ページにあります。

Apache Spark 用の C3R 暗号化クライアントパッケージ (`c3r-cli-spark`) は c3r-cli の 1 バージョンであり、実行中の Apache Spark サーバーにジョブとして送信する必要があります。詳細については、「Running C3R on Apache Spark」を参照してください。

(オプション) ステップ 3: C3R 暗号化クライアントで使用可能なコマンドを表示する

以下の手順に従って、C3R 暗号化クライアントで使用可能なコマンドを理解してください。

C3R 暗号化クライアントで使用可能なコマンドをすべて表示するには

1. コマンドラインインターフェイス (CLI) から、ダウンロードした c3r-cli.jar ファイルが含まれているフォルダに移動します。
2. 次のコマンドを実行します。`java -jar c3r-cli.jar`
3. 使用可能なコマンドとオプションが一覧表示されます。

ステップ 4: 表形式ファイルの暗号化スキーマを生成する

データを暗号化するには、データの使用方法を記述した暗号化スキーマが必要です。このセクションでは、ヘッダー行を含む CSV ファイルまたは Parquet ファイルの暗号化スキーマが、C3R 暗号化クライアントによってどのように生成されるかについて説明します。

この操作はファイルごとに 1 回のみ必要です。スキーマを作成したら、それを再利用して同じファイル (または同じ列名を持つ任意のファイル) を暗号化できます。列名または必要な暗号化スキーマが変更された場合は、スキーマファイルを更新する必要があります。詳細については、「(オプション) スキーマの作成 (上級ユーザー)」を参照してください。

> **Important**
>
> コラボレーションを行うすべての関係者が同じ共有シークレットキーを使用することが最も重要です。また、コラボレーションの関係者は、クエリで列を JOIN する場合や等価比較を行う場合に、列名が一致するように調整する必要があります。これを行わないと、SQL クエリが予期しない結果や誤った結果を生成する可能性があります。ただし、コラボレーションクリエーターがコラボレーションの作成中に `allowJoinsOnColumnsWithDifferentNames` 暗号化設定を有効にしている場合は、これ

は必要ありません。暗号化に関する設定の詳細については、「暗号コンピューティングパラメータ」を参照してください。

スキーマモードで実行すると、C3R 暗号化クライアントは入力ファイルを列ごとにチェックし、その列を処理すべきかどうか、またどのように処理すべきかのプロンプトを表示します。ファイルに暗号化後の出力では必要のない列が多数含まれていると、不要な列をそれぞれスキップしなければならないため、スキーマ生成のインタラクティブなプロセスが面倒になることがあります。これを回避するには、スキーマを手動で記述するか、必要な列のみを含む簡略版の入力ファイルを作成します。そうすることで、その縮小されたファイルに対してインタラクティブなスキーマ生成プロセスを実行できます。C3R 暗号化クライアントはスキーマファイルに関する情報を出力し、ソース列をターゲット出力にどのように含めるか、またはどのように暗号化するか (暗号化する場合) をたずねます。

入力ファイルのソース列ごとに、次の入力を求められます。

1. 生成するターゲット列の数
2. 各ターゲット列を暗号化する方法 (暗号化する場合)
3. 各ターゲット列の名前
4. 列をsealed列として暗号化する場合は、暗号化前にデータをどのようにパディングするか

> **Note**
>
> sealed列として暗号化されている列のデータを暗号化する場合は、どのデータをパディングする必要があるかを判断する必要があります。C3R 暗号化クライアントでは、スキーマ生成時に、列のすべてのエントリを同じ長さにするデフォルトのパディングを推奨します。`fixed` の長さを決定する際、パディングはビット単位ではなくバイト単位であることに注意してください。

以下はスキーマを作成するための決定表です。

スキーマの決定表

決定	ソース列 <'name-of-column'> のターゲット列の数。	Target column type: [c] cleartext, [f] fingerprint, or [s] sealed ?	ターゲット列ヘッダー名 <デフォルト 'name-of-column'>	Add suffix <suffix> to header to indicate how it was encrypted, [y] yes or [n] no <default 'yes'>	< 'name-of-column_詞あり'> パディングタイプ: [n] 1、[f] 固定、または [m] 最大 <デフォルト 'max'>
列を暗号化しない。	1	c	該当しない	該当しない	該当しない
列をfingerprint列として暗号化する。	1	f	デフォルトを選択するか、新しいヘッダー名を入力します。	「y」を入力してデフォルト (`_fingerprint`) を選択するか、「n」を入力します。	該当しない
列をsealed列として暗号化する。	1	s	デフォルトを選択するか、新しいヘッダー名を入力します。	「y」を入力してデフォルト (`_sealed`) を選択するか、「n」を入力します。	パディングタイプを選択します。 詳細については、「(オプション) スキーマの作成 (上級ユーザー)」を参照してください。

決定	ソース列 <'name-of-column'> のターゲット列の数。	Target column type: [c] cleartext, [f] fingerprint, or [s] sealed ?	ターゲット列ヘッダー名 <デフォルト 'name-of-column'>	Add suffix <suffix> to header to indicate how it was encrypted, [y] yes or [n] no <default 'yes'>	< 'name-of-column_詞あり'> パディングタイプ: [n] 1、[f] 固定、または [m] 最大 <デフォルト 'max'>
列をfingerprintとsealedの両方として暗号化する。	2	1 番目のターゲット列に「f」と入力します。 2 番目のターゲット列に「s」と入力します。	各ターゲット列のターゲットヘッダーを選択します。	「y」を入力してデフォルトを選択するか、「n.」を入力します。	パディングタイプを選択します (sealed 列のみ)。 詳細については、「(オプション) スキーマの作成 (上級ユーザー)」を参照してください。

次の 2 つの例は、暗号化スキーマの作成方法を示しています。インタラクションの正確な内容は、入力ファイルとユーザーが入力した回答によって異なります。

例

- 例: fingerprint列とcleartext列の暗号化スキーマの生成
- 例:sealed、fingerprint、およびcleartext列を含む暗号化スキーマの生成

例: fingerprint列とcleartext列の暗号化スキーマの生成

この例では、`ads.csv` に、`username` と `ad_variant` の 2 つの列しかありません。これらの列を次のようにします。

- username 列を fingerprint 列として暗号化する
- ad_variant 列を cleartext 列にする

fingerprint列とcleartext列の暗号化スキーマを生成するには

1. (オプション) c3r-cli.jar ファイルと暗号化するファイルが存在することを確認します。

   a. 目的のディレクトリに移動し、ls (Mac または Unix/Linux を使用している場合)、あるいは dir (Windows を使用している場合) を実行します。

   b. 表形式のデータファイル (.csv など) のリストを表示し、暗号化するファイルを選択します。

      この例では、ads.csv が暗号化するファイルです。

2. CLI から、次のコマンドを実行してスキーマをインタラクティブに作成します。

   ```
   java -jar c3r-cli.jar schema ads.csv --interactive --output=ads.json
   ```

   > **Note**
   >
   > - java --jar PATH/TO/c3r-cli.jar を実行することができます。または、CLASSPATH 環境変数に PATH/TO/c3r-cli.jar を追加した場合は、クラス名を実行することもできます。C3R 暗号化クライアントは CLASSPATH をチェックしてクラス名を検索します (java com.amazon.psion.cli.Main など)。
   > - --interactive フラグは、インタラクティブモードでのスキーマ作成を選択するものです。これにより、ユーザーはウィザードに従ってスキーマを作成することになります。高度なスキルを持つユーザーは、ウィザードを使用せずに独自のスキーマ JSON を作成できます。詳細については、「(オプション) スキーマの作成 (上級ユーザー)」を参照してください。
   > - --output フラグは出力名を設定するものです。--output フラグを含めないと、C3R 暗号化クライアントはデフォルトの出力名 (<input>.out.csv、またはスキーマには <input>.json など) を選択しようとします。

3. Number of target columns from source column 'username'? で「**1**」を入力して、Enter キーを押します。

4. Target column type: [c]leartext, [f]ingerprint, or [s]ealed? で「**f**」を入力して、Enter キーを押します。

5. `Target column headername <default 'username'>` で Enter キーを押します。

   デフォルトのユーザー名「`username`」が使用されます。

6. `Add suffix '_fingerprint' to header to indicate how it was encrypted, [y]es or [n]o <default 'yes'>` で「**y**」を入力して、Enter キーを押します。

   > **Note**
   >
   > インタラクティブモードでは、暗号化された列のヘッダーに追加するサフィックス (fingerprint列の場合は `_fingerprint`、sealed列の場合は `_sealed`) が提示されます。サフィックスは、 へのデータのアップロード AWS のサービス や AWS Clean Rooms コラボレーションの作成などのタスクを実行するときに役立ちます。これらのサフィックスによって、各列の暗号化されたデータで何ができるかを判別できます。例えば、sealed列 (`_sealed`) として暗号化した列に JOIN を実行しようとしても (またはその逆を試みても)、正常に処理されません。

7. `Number of target columns from source column 'ad_variant'?` で「**1**」を入力して、Enter キーを押します。

8. `Target column type: [c]leartext, [f]ingerprint, or [s]ealed?` で「**c**」を入力して、Enter キーを押します。

9. `Target column headername <default 'username'>` で Enter キーを押します。

   デフォルトのユーザー名「`ad_variant`」が使用されます。

   スキーマが `ads.json` という新しいファイルに書き込まれます。

   > **Note**
   >
   > スキーマは Windows の Notepad や macOS の TextEdit などの任意のテキストエディタで開いて表示できます。

10. これで、データを暗号化する準備ができました。

例:sealed、fingerprint、およびcleartext列を含む暗号化スキーマの生成

この例では、`sales.csv` に、`username`、`purchased`、および `product` の 3 つの列があります。これらの列を次のようにします。

- product 列を sealed 列にする
- username 列を fingerprint 列として暗号化する
- purchased 列を cleartext 列にする

sealed、fingerprint、およびcleartext列を含む暗号化スキーマを生成するには

1. (オプション) c3r-cli.jar ファイルと暗号化するファイルが存在することを確認します。

   a. 目的のディレクトリに移動し、ls (Mac または Unix/Linux を使用している場合)、あるいは dir (Windows を使用している場合) を実行します。

   b. 表形式のデータファイル (.csv) のリストを表示し、暗号化するファイルを選択します。

      この例では、sales.csv が暗号化するファイルです。

2. CLI から、次のコマンドを実行してスキーマをインタラクティブに作成します。

   ```
   java -jar c3r-cli.jar schema sales.csv --interactive --
   output=sales.json
   ```

   > **Note**
   >
   > - --interactive フラグは、インタラクティブモードでのスキーマ作成を選択するものです。これにより、ユーザーはガイド付きワークフローに従ってスキーマを作成することになります。
   > - 高度なスキルを持つユーザーは、ガイド付きのワークフローを使用せずに独自のスキーマ JSON を作成できます。詳細については、「(オプション) スキーマの作成 (上級ユーザー)」を参照してください。
   > - 列ヘッダーのない .csv ファイルについては、CLI で使用可能なスキーマコマンドの --noHeaders フラグを参照してください。
   > - --output フラグは出力名を設定するものです。--output フラグを含めないと、C3R 暗号化クライアントはデフォルトの出力名 (<input>.out、またはスキーマには <input>.json など) を選択しようとします。

3. Number of target columns from source column 'username'? で「**1**」を入力して、Enter キーを押します。

4. Target column type: [c]leartext, [f]ingerprint, or [s]ealed? で「**f**」を入力して、Enter キーを押します。

5. `Target column headername <default 'username'>` で Enter キーを押します。

   デフォルトのユーザー名「`username`」が使用されます。

6. `Add suffix '_fingerprint' to header to indicate how it was encrypted, [y]es or [n]o <default 'yes'>` で「**y**」を入力して、Enter キーを押します。

7. `Number of target columns from source column 'purchased'?` で「**1**」を入力して、Enter キーを押します。

8. `Target column type: [c]leartext, [f]ingerprint, or [s]ealed?` で「**c**」を入力して、Enter キーを押します。

9. `Target column headername <default 'purchased'>` で Enter キーを押します。

   デフォルトのユーザー名「`purchased`」が使用されます。

10. `Number of target columns from source column 'product'?` で「**1**」を入力して、Enter キーを押します。

11. `Target column type: [c]leartext, [f]ingerprint, or [s]ealed?` で「**s**」を入力して、Enter キーを押します。

12. `Target column headername <default 'product'>` で Enter キーを押します。

    デフォルトのユーザー名「`product`」が使用されます。

13. `'product_sealed' padding type: [n]one, [f]ixed, or [m]ax <default 'max'?>` で Enter キーを押してデフォルトを選択します。

14. `Byte-length beyond max length to pad cleartext to in 'product_sealed' <default '0'>?` で Enter キーを押してデフォルトを選択します。

    スキーマが `sales.json` という新しいファイルに書き込まれます。

15. これで、データを暗号化する準備ができました。

ステップ 5: 共有シークレットキーを作成する

データテーブルを暗号化するには、コラボレーション参加者が共有シークレットキーについて合意し、キーを安全に共有する必要があります。

共有シークレットキーは 256 ビット (32 バイト) 以上である必要があります。これより大きいキーを指定することもできますが、それによってセキュリティがさらに強化されることはありません。

> **Important**
>
> 暗号化と復号化に使用されるキーとコラボレーション ID は、すべてのコラボレーション参加者で同一でなければならないことに注意してください。

以下のセクションでは、各端末の現在の作業ディレクトリに secret.key として保存される共有シークレットキーを生成するコンソールコマンドの例を示します。

トピック

- 例: OpenSSL を使用したキーの生成
- 例: Windows の PowerShell を使用したキーの生成

例: OpenSSL を使用したキーの生成

一般的な汎用目的の暗号化ライブラリでは、以下のコマンドを実行して共有シークレットキーを作成します。

```
openssl rand 32 > secret.key
```

Windows を使用していてまだ OpenSSL をインストールしていない場合は、「例:Windows の PowerShell を使用したキーの生成」で説明されている例を使用してキーを生成できます。

例: Windows の PowerShell を使用したキーの生成

Windows で利用可能なターミナルアプリケーションの PowerShell では、以下のコマンドを実行して共有シークレットキーを作成します。

```
$bs = New-Object Byte[](32);
[Security.Cryptography.RandomNumberGenerator]::Create().GetBytes($bs); Set-Content 'secret.key' -Encoding Byte -Value $bs
```

ステップ 6: 共有シークレットキーを環境変数に保存する

環境変数は、ユーザーが などのさまざまなキーストアからシークレットキーを提供 AWS Secrets Manager して C3R 暗号化クライアントに渡すための便利で拡張可能な方法です。

を使用してそれらのキーを関連する環境変数に保存 AWS のサービス する場合 AWS CLI 、C3R 暗号化クライアントは に保存されているキーを使用できます。例えば、C3R 暗号化クライアントは の

キーを使用できます AWS Secrets Manager。詳細については、「AWS Secrets Manager ユーザーガイド」の「AWS Secrets Managerでのシークレットの作成と管理」を参照してください。

> **Note**
>
> ただし、 AWS のサービス などの を使用して C3R キー AWS Secrets Manager を保持する前に、ユースケースで許可されていることを確認してください。ユースケースによっては、キーを から保留する必要がある場合があります AWS。これは、暗号化されたデータとキーが同じサードパーティに保持されないようにするためです。

共有シークレットキーの唯一の要件は、共有シークレットキーを base64 でエンコードして環境変数 C3R_SHARED_SECRET に保存することです。

以下のセクションでは、secret.key ファイルを base64 に変換して環境変数として保存するためのコンソールコマンドについて説明します。secret.key ファイルは、「ステップ 5: 共有シークレットキーを作成する」で紹介されているコマンドのいずれかによって生成されたもので、ソースの一例にすぎません。

Windows で PowerShell を使用して環境変数にキーを保存

Windows で PowerShell を使用して base64 に変換し、環境変数を設定するには、次のコマンドを実行します。

```
$Bytes=[IO.File]::ReadAllBytes((Get-Location).ToString()+'\secret.key');
 $env:C3R_SHARED_SECRET=[Convert]::ToBase64String($Bytes)
```

Linux または macOS で環境変数にキーを保存

Linux または macOS で base64 に変換し、環境変数を設定するには、次のコマンドを実行します。

```
export C3R_SHARED_SECRET="$(cat secret.key | base64)"
```

ステップ 7: データを暗号化する

このステップを実行するには、 AWS Clean Rooms コラボレーション ID と共有シークレットキーを取得する必要があります。詳細については、「前提条件を満たす」を参照してください。

次の例では、作成した ads.json というスキーマを使用して ads.csv に暗号化を実行します。

データを暗号化するには

1. コラボレーション用の共有シークレットキーを保存します (「ステップ 6: 共有シークレットキーを環境変数に保存する」を参照)。
2. コマンドラインで、以下のコマンドを入力します。

   ```
   java -jar c3r-cli.jar encrypt <name of input .csv file> --schema=<name of schema .json file> --id=<collaboration id> --output=<name of output.csv file> <optional flags>
   ```

3. *<name of input .csv file>* には、入力 .csv ファイルの名前を入力します。
4. schema= には、.json 暗号化スキーマのファイル名を入力します。
5. id= には、コラボレーション ID を入力します。
6. output= には、出力ファイルの名前 (例えば ads-output.csv) を入力します。
7. 「暗号コンピューティングパラメータ」および「Cryptographic Computing for Clean Rooms のオプションフラグ」で説明されているコマンドラインフラグをすべて含めます。
8. コマンドを実行します。

ads.csv の例では、以下のコマンドを実行します。

java -jar c3r-cli.jar encrypt ***ads.csv*** --schema=***ads.json*** --id=***123e4567-e89b-42d3-a456-556642440000*** --output=***ads-output.csv***

sales.csv の例では、以下のコマンドを実行します。

java -jar c3r-cli.jar encrypt ***sales.csv*** --schema=***sales.json*** --id=***123e4567-e89b-42d3-a456-556642440000***

> **Note**
>
> この例では、出力ファイル名 (--output=*sales-output.csv*) は指定していません。その結果、デフォルトの出力ファイル名 name-of-file.out.csv が生成されました。

これで、暗号化されたデータを確認する準備ができました。

ステップ 8: データ暗号化を確認する

データが暗号化されていることを確認するには

1. 暗号化されたデータファイル (`sales-output.csv` など) を表示します。
2. 以下の列を確認します。

   a. 列 1 – 暗号化済み (`username_fingerprint` など)。

      fingerprint列 (HMAC) には、バージョンとタイプのプレフィックス (`01:hmac:` など) の後に、base64 でエンコードされた 44 文字のデータがあります。

   b. 列 2 – 暗号化されていません (`purchased` など)。

   c. 列 3 – 暗号化済み (`product_sealed` など)。

      暗号化された (SELECT) 列では、バージョンとタイプのプレフィックス (`01:enc:` など) の後に続く、cleartextに任意のパディングを加えた長さは、暗号化されたcleartextの長さに正比例します。つまりこの長さは、入力データのサイズにエンコーディングによる約 33% のオーバーヘッドを加えたものです。

これで次の作業に進むことができます。

1. 暗号化されたデータを S3 にアップロードする
2. AWS Glue テーブル を作成します。
3. AWS Clean Roomsで設定済みテーブルを作成する

C3R 暗号化クライアントは、暗号化されていないデータを含まない一時ファイルを作成します (最終出力でそのデータも暗号化されない場合を除く)。ただし、暗号化された値の中には、正しくパディングされていないものもあります。`allowRepeatedFingerprintValue` のコラボレーション設定が `false` であっても、フィンガープリント列に重複する値が含まれる場合があります。この問題は、適切なパディング長と重複削除のプロパティがチェックされる前に一時ファイルが書き込まれることが原因で発生します。

暗号化中に C3R 暗号化クライアントに障害や中断が発生すると、一時ファイルを書き込んだ後、これらのプロパティを確認して一時ファイルを削除する前に、C3R 暗号化クライアントが停止することがあります。そのため、こうした一時ファイルがまだディスク上に残っている可能性があります。このような場合、こうしたファイル内のコンテンツでは、プレーンテキストデータが出力と同じレベルで保護されることはありません。特に、このような一時ファイルに対して、最終出力では効果がな

い統計分析が実行されることで、プレーンテキストデータが明らかになる可能性があります。こうしたファイルが権限のない人の手に渡らないように、ユーザーはこれらのファイル (特に SQLite データベース) を削除する必要があります。

(オプション) スキーマの作成 (上級ユーザー)

スキーマの手動作成は、上級ユーザーを対象としています。

以下のセクションでは、列ヘッダーがある場合とない場合の、入力ファイルの JSON スキーマファイル形式について説明しています。上級ユーザーは、必要に応じてスキーマを直接記述または変更できます。

> **Note**
>
> C3R 暗号化クライアントでは、「例:sealed、fingerprint、およびcleartext列を含む暗号化スキーマの生成」で説明されているインタラクティブなプロセス、またはスタブテンプレートの作成のいずれかを通じてスキーマを作成できます。

マッピングテーブルスキーマと位置テーブルスキーマ

次のセクションでは、2 種類のテーブルスキーマについて説明します。

- マッピングテーブルスキーマ – このスキーマは、ヘッダー行を含む .csv ファイルと Apache Parquet ファイルの暗号化に使用されます。
- 位置テーブルスキーマ – このスキーマは、ヘッダー行のない .csv ファイルの暗号化に使用されます。

C3R 暗号化クライアントでは、コラボレーション用の表形式ファイルを暗号化できます。それには、入力から暗号化された出力をどのように導き出すかを指定するための、対応するスキーマファイルが必要です。

C3R 暗号化クライアントでは、コマンドラインで C3R 暗号化クライアントのスキーマコマンドを実行することで、INPUT ファイルのスキーマを生成できます。`java -jar c3r-cli.jar schema --interactive INPUT` はコマンドの一例です。

スキーマでは、以下の情報を指定します。

1. ヘッダー名 (マッピングスキーマ) または位置 (位置スキーマ) によって、どのソース列が出力ファイルのどの変換後の列にマッピングされるか
2. どのターゲット列をcleartextのまま残すか
3. SELECT クエリのためにどのターゲット列を暗号化するか
4. JOIN クエリのためにどのターゲット列を暗号化するか

この情報が、テーブル固有の JSON スキーマファイルにエンコードされます。スキーマファイルは headerRow フィールドがブール値の 1 つのオブジェクトで構成されます。この値は、Parquet ファイルとヘッダー行のある .csv ファイルでは `true`、それ以外の場合は `false` である必要があります。

マッピングテーブルスキーマ

マッピングスキーマは次のような形式です。

```
{
  "headerRow": true,
  "columns": [
    {
      "sourceHeader": STRING,
      "targetHeader": STRING,
      "type": TYPE,
      "pad": PAD
    },
    ...
  ]
}
```

headerRow が `true` の場合、オブジェクトの次のフィールドは `columns` になります。これには、ソースヘッダーをターゲットヘッダーにマッピングする列スキーマの配列 (つまり、出力列に含める内容を記述した JSON オブジェクト) が含まれます。

- `sourceHeader` – データの取得元となるソース列の `STRING` ヘッダー名。

> **Note**
>
> 同じソース列を複数のターゲット列に使用できます。

スキーマのどこにも sourceHeader としてリストされていない入力ファイルの列は、出力ファイルには表示されません。

- targetHeader – 出力ファイル内の対応する列の STRING ヘッダー名。

> **Note**
>
> このフィールドはマッピングスキーマでは省略可能です。このフィールドを省略すると、sourceHeader が出力ファイルのヘッダー名として再利用されます。出力列がfingerprint列またはsealed列の場合は、それぞれ _fingerprint または _sealed が追加されます。

- type – 出力ファイルのターゲット列の TYPE。コラボレーションでの列の使用方法に応じて、cleartext、sealed、fingerprint のいずれかになります。
- pad – TYPE が sealed の場合にのみ存在する列スキーマオブジェクトのフィールド。対応する値 PAD は、データを暗号化する前にどのようにパディングするかを記述するオブジェクトです。

```
{
  "type": PAD_TYPE,
  "length": INT
}
```

暗号化前のパディングを指定するには、type と length を次のように使用します。

- PAD_TYPE が none の場合 – 列のデータにはパディングが適用されず、length フィールドは無効になります (つまり、省略されます)。
- PAD_TYPE が fixed の場合 – 列のデータは指定された length のバイト数にパディングされます。
- PAD_TYPE が max の場合 –列のデータは、最も長い値のバイト長に length バイトを加えたサイズにパディングされます。

以下は、各タイプの列を含むマッピングスキーマの例です。

```
{
  "headerRow": true,
  "columns": [
    {
      "sourceHeader": "FullName",
```

```
      "targetHeader": "name",
      "type": "cleartext"
    },
    {
      "sourceHeader": "City",
      "targetHeader": "city_sealed",
      "type": "sealed",
      "pad": {
        "type": "max",
        "length": 16
      }
    },
    {
      "sourceHeader": "PhoneNumber",
      "targetHeader": "phone_number_fingerprint",
      "type": "fingerprint"
    },
    {
      "sourceHeader": "PhoneNumber",
      "targetHeader": "phone_number_sealed",
      "type": "sealed",
      "pad": {
        "type": "fixed",
        "length": 20
      }
    }
  ]
}
```

より複雑な例として、ヘッダー付きの .csv ファイルの例を以下に示します。

```
FirstName,LastName,Address,City,State,PhoneNumber,Title,Level,Notes
Jorge,Souza,12345 Mills Rd,Anytown,SC,703-555-1234,CEO,10,
Paulo,Santos,0 Street,Anytown,MD,404-555-111,CIO,9,This is a really long note that
 could really be a paragraph
Mateo,Jackson,1 Two St,Anytown,NY,304-555-1324,COO,9,""
Terry,Whitlock4 N St,Anytown,VA,407-555-8888,EA,7,Secret notes
Diego,Ramirez,9 Hollows Rd,Anytown,VA,407-555-1222,SDE I,4,null
John,Doe,8 Hollows Rd,Anytown,VA,407-555-4321,SDE I,4,Jane's younger brother
Jane,Doe,8 Hollows Rd,Anytown,VA,407-555-4322,SDE II,5,John's older sister
```

次のマッピングスキーマの例では、列 FirstName と LastName は cleartext 列です。State 列は fingerprint 列として暗号化され、また、パディングが none で sealed 列としても暗号化されます。残りの列は省略されます。

```
{
  "headerRow": true,
  "columns": [
    {
      "sourceHeader": "FirstName",
      "targetHeader": "GivenName",
      "type": "cleartext"
    },
    {
      "sourceHeader": "LastName",
      "targetHeader": "Surname",
      "type": "cleartext"
    },
    {
      "sourceHeader": "State",
      "targetHeader": "State_Join",
      "type": "fingerprint"
    },
    {
      "sourceHeader": "State",
      "targetHeader": "State",
      "type": "sealed",
      "pad": {
        "type": "none"
      }
    }
  ]
}
```

以下は、マッピングスキーマから生成された .csv ファイルです。

```
givenname,surname,state_fingerprint,state
John,Doe,01:hmac:UK8s8Cn/WR2JO/To2dTxWD73aDEe2ZUXeSHy3Tv
+1Mk=,01:enc:FQ3n3Ahv9BQQNWQGcugeHzHYzEZE1vapHa2Uu4SRgSAtZ3qObjPA4TcsHt
+BOkMKBcnHWI13BeGG/SBqmj7vKpI=
Paulo,Santos,01:hmac:CHF4eIrtTNgAooU9v4h9Qjc
+txBnMidQTjdjWuaDTTA=,01:enc:KZ5n5GtaXACco65AXk48BQO2durDNR2ULc4YxmMC8NaZZKKJiksU1IwFadAvV4iBQ1
Mateo,Jackson,01:hmac:iIRnjfNBzryusIJ1w35lgNzeY1RQ1bSfq6PDHW8Xrbk=,01:enc:mLKpS5HIOSgphdEsrzhEd
eN9nBO2gAbIygt4OFn4LalYn9Xyj/XUWXlmn8zFe2T4kyDTD8kGOvpQEUGxAUFk=
```

```
Diego,Ramirez,01:hmac:UK8s8Cn/WR2JO/To2dTxWD73aDEe2ZUXeSHy3Tv+1Mk=,01:enc:rmZhT98Zm
+IIGw1UTjMIJP4IrW/AAltBLMXcHvnYfRgmWP623VFQ6aUnhsb2MDqEw4G5Uwg5rKKZepUxx5uKbfk=
Jorge,Souza,01:hmac:3BxJdXiFFyZ8HBbYNqqEhBVqhNOd7s2ZiKUe7QiTyo8=,01:enc:vVaqWC1VRbhvkf8gnuR7q0z
Terry,Whitlock01:hmac:UK8s8Cn/WR2JO/To2dTxWD73aDEe2ZUXeSHy3Tv+1Mk=,01:enc:3c9VEWbODO/
xbQjdGuccLvI7oZTBdPU+SyrJIyr2kudfAxbuMQ2uRdU/q7rbgyJjxZS8M2U35ILJf/lDgTyg7cM=
Jane,Doe,01:hmac:UK8s8Cn/WR2JO/To2dTxWD73aDEe2ZUXeSHy3Tv+1Mk=,01:enc:9RWv46YLveykeNZ/
G0NdlYFg+AVdOnu05hHyAYTQkPLHnyX+0/jbzD/g9ZT8GCgVE9aB5bV4ooJIXHGBVMXcjrQ=
```

位置テーブルスキーマ

位置スキーマは次のような形式です。

```
{
  "headerRow": false,
  "columns": [
    [
      {
        "targetHeader": STRING,
        "type": TYPE,
        "pad": PAD
      },
      {
        "targetHeader": STRING,
        "type": TYPE,
        "pad": PAD
      }
    ],
    [],
    ...
  ]
}
```

headerRow が false の場合、オブジェクトの次のフィールドはエントリの配列を含む columns になります。各エントリは、それ自体が 0 個以上の位置列スキーマ (sourceHeader フィールドなし) の配列、つまり、出力に含める内容を記述した JSON オブジェクトです。

- sourceHeader – データの取得元となるソース列の STRING ヘッダー名。

> **Note**
>
> 位置スキーマでは、このフィールドは省略する必要があります。位置スキーマの場合、ソース列は、スキーマファイルの列の対応するインデックスによって推測されます。

- `targetHeader` – 出力ファイル内の対応する列の STRING ヘッダー名。

> **Note**
>
> このフィールドは、位置スキーマでは必須です。

- `type` – 出力ファイルのターゲット列の TYPE。コラボレーションでの列の使用方法に応じて、`cleartext`、`sealed`、`fingerprint` のいずれかになります。
- `pad` – TYPE が `sealed` の場合にのみ存在する列スキーマオブジェクトのフィールド。対応する値 PAD は、データを暗号化する前にどのようにパディングするかを記述するオブジェクトです。

```
{
  "type": PAD_TYPE,
  "length": INT
}
```

暗号化前のパディングを指定するには、`type` と `length` を次のように使用します。

- PAD_TYPE が none の場合 – 列のデータにはパディングが適用されず、`length` フィールドは無効になります (つまり、省略されます)。
- PAD_TYPE が `fixed` の場合 – 列のデータは指定された `length` のバイト数にパディングされます。
- PAD_TYPE が `max` の場合 –列のデータは、最も長い値のバイト長に `length` バイトを加えたサイズにパディングされます。

> **Note**
>
> `fixed` は、列のデータのバイトサイズの上限が事前にわかっている場合に便利です。その列に指定した `length` 値よりも長いデータが 1 つでもあると、エラーが発生します。`max` は、データのサイズに関係なく動作するため、入力データの正確なサイズが不明な場合に便利です。ただし `max` の場合は、データが 2 回暗号化されるため、処理時間が長

くなります。max では、データを一時ファイルに読み込むときに 1 回暗号化し、列内の最も長いデータエントリがわかった後に 1 回暗号化します。
また、最も長い値の長さは、クライアントの呼び出しの合間には保存されません。データをバッチで暗号化したり、新しいデータを定期的に暗号化したりする場合は、生成される暗号文の長さはバッチによって異なる可能性があることに注意してください。

以下は位置スキーマの例です。

```
{
  "headerRow": false,
  "columns": [
    [
      {
        "targetHeader": "name",
        "type": "cleartext"
      }
    ],
    [
      {
        "targetHeader": "city_sealed",
        "type": "sealed",
        "pad": {
          "type": "max",
          "length": 16
        }
      }
    ],
    [
      {
        "targetHeader": "phone_number_fingerprint",
        "type": "fingerprint"
      },
      {
        "targetHeader": "phone_number_sealed",
        "type": "sealed",
        "pad": {
          "type": "fixed",
          "length": 20
        }
      }
    ]
```

```
  ]
}
```

複雑な例として、先頭にヘッダー行がない .csv ファイルの例を以下に示します。

```
Jorge,Souza,12345 Mills Rd,Anytown,SC, 703 -555 -1234,CEO, 10,
Paulo,Santos, 0 Street,Anytown,MD, 404-555-111,CIO, 9,This is a really long note that
 could really be a paragraph
Mateo,Jackson, 1 Two St,Anytown,NY, 304-555-1324,COO, 9, ""
Terry,Whitlock, 4 N St,Anytown,VA, 407-555-8888,EA, 7,Secret notes
Diego,Ramirez, 9 Hollows Rd,Anytown,VA, 407-555-1222,SDE I, 4,null
John,Doe, 8 Hollows Rd,Anytown,VA, 407-555-4321,SDE I, 4,Jane's younger brother
Jane,Doe, 8 Hollows Rd,Anytown,VA, 407-555-4322,SDE II, 5,John's older sister
```

位置スキーマは次のような形式になります。

```
{
  "headerRow": false,
  "columns": [
    [
      {
        "targetHeader": "GivenName",
        "type": "cleartext"
      }
    ],
    [
      {
        "targetHeader": "Surname",
        "type": "cleartext"
      }
    ],
    [],
    [],
    [
      {
        "targetHeader": "State_Join",
        "type": "fingerprint"
      },
      {
        "targetHeader": "State",
        "type": "sealed",
        "pad": {
          "type": "none"
```

```
        }
      }
    ],
    [],
    [],
    [],
    []
  ]
}
```

前述のスキーマは、指定されたターゲットヘッダーがヘッダー行に含まれる出力ファイルを生成します。

```
givenname,surname,state_fingerprint,state
Mateo,Jackson,01:hmac:iIRnjfNBzryusIJ1w35lgNzeY1RQ1bSfq6PDHW8Xrbk=,01:enc:ENS6QD3cMVl9vQEGfe9MN
Q8m/Y5SA89dJwKpT5rGPp8e36h6klwDoslpFzGvU0=
Jorge,Souza,01:hmac:3BxJdXiFFyZ8HBbYNqqEhBVqhNOd7s2ZiKUe7QiTyo8=,01:enc:LKo0zirq2+
+XEIIIMNRjAsGMdyWUDwYaum0B+IFP+rUf1BNeZDJjtFe1Z+zbZfXQWwJy52Rt7HqvAb2WIK1oMmk=
Paulo,Santos,01:hmac:CHF4eIrtTNgAooU9v4h9Qjc
+txBnMidQTjdjWuaDTTA=,01:enc:MyQKyWxJ9kvK1xDQQtXlUNwv3F+yrBRr0xrUY/1BGg5KFgOn9pK+MZ7g
+ZNqZEPcPz4lht1u0t/wbTaqzOCLXFQ=
Jane,Doe,01:hmac:UK8s8Cn/WR2JO/To2dTxWD73aDEe2ZUXeSHy3Tv+1Mk=,01:enc:Pd8sbITBfb0/
ttUB4svVsgoYkDfnDvgkvxzeci0Yxq54rLSwccy1o3/B50C3cpkkn56dovCwzgmmPNwrmCmYtb4=
Terry,Whitlock01:hmac:UK8s8Cn/WR2JO/To2dTxWD73aDEe2ZUXeSHy3Tv
+1Mk=,01:enc:Qmtzu3B3GAXKh2KkRYTiEAaMopYedsSdF2e/
ADUiBQ9kv2CxKPzWyYTD3ztmKPMka19dHre5VhUHNpO3O+j1AQ8=
Diego,Ramirez,01:hmac:UK8s8Cn/WR2JO/To2dTxWD73aDEe2ZUXeSHy3Tv+1Mk=,01:enc:ysdg
+GHKdeZrS/geBIooOEPLHG68MsWpx1dh3xjb+fG5rmFmqUcJLNuuYBHhHAlxchM2WVeV1fmHkBX3mvZNvkc=
John,Doe,01:hmac:UK8s8Cn/WR2JO/To2dTxWD73aDEe2ZUXeSHy3Tv+1Mk=,01:enc:9uX0wZuO7kAPAx
+Hf6uvQownkWqFSKtWS7gQIJSe5aXFquKWCK6yZN0X5Ea2N3bn03Uj1kh0agDWoiP9FRZGJA4=
```

AWS Clean Roomsでの設定済みテーブルの作成

設定済みテーブルは、 AWS Glue Data Catalog内の既存のテーブルを参照するものです。設定済みテーブルには、 AWS Clean Roomsでどのようにデータにクエリを実行するかを決定する分析ルールが含まれています。設定済みテーブルは 1 つ以上のコラボレーションに関連付けることができます。の詳細については AWS Glue、AWS Glueデベロッパーガイド」を参照してください。

によって提供される統計生成を使用して AWS Glue 、 AWS Glue Data Catalog テーブルの列レベルの統計を計算します。がデータカタログ内のテーブルの統計 AWS Glue を生成すると、Amazon Redshift Spectrum は自動的にこれらの統計を使用してクエリプランを最適化します。を使用した列レベルの統計の計算の詳細については AWS Glue、「列統計の使用ガイド」を参照してください。

設定済みテーブルを作成する

このステップでは、コラボレーション AWS Clean Rooms で使用する設定済みテーブルを に作成します。

で設定済みテーブルを作成するには AWS Clean Rooms

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. コンソールの左のナビゲーションペインで、[設定済みのテーブル] を選択します。
3. 右上隅にある [新しいテーブルを設定] を選択します。
4. [新しいテーブルを設定] の [AWS Glue テーブルを選択] で以下の操作を行います。

    a. 設定する [データベース] をドロップダウンリストから選択します。

    b. 設定する [テーブル] をドロップダウンリストから選択します。

    > **Note**
    >
    > テーブルが正しいことを確認するには、次のいずれかの操作を行います。
    >
    > - で表示を選択します AWS Glue。
    > - [スキーマを表示] をオンにして、スキーマを表示します。

5. [コラボレーションで許可された列] で、[すべての列] または [カスタムリスト] を選択します。

選択内容	結果
すべての列	すべての列は で使用できます AWS Clean Rooms（分析ルールに従います）。
カスタムリスト	[許可する列を指定] ドロップダウンリストから、許可する列を 1 つ以上選択します。

6. [設定済みのテーブルの詳細] で以下の操作を行います。

   a. 設定済みテーブルの [名前] を入力します。

      デフォルトの名前を使用することも、テーブルの名前を変更することもできます。

   b. テーブルの [説明] を入力します。

      この説明は、似たような名前を持つ他の設定済みテーブルと区別するのに役立ちます。

   c. 設定済みテーブルのリソースで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。

7. [新しいテーブルを設定] を選択します。

次のステップ

設定済みテーブルを作成したら、次の作業に進むことができます。

- 設定済みテーブルに分析ルールを設定する
- 設定済みテーブルをコラボレーションに関連付ける

設定済みテーブルへの分析ルールの設定

以下のセクションでは、設定済みテーブルに分析ルールを設定する方法を説明します。分析ルールを定義すると、クエリを行えるメンバーに対して、AWS Clean Rooms でサポートされる特定の分析ルールに一致するクエリを実行する権限を与えることができます。

AWS Clean Rooms では、集計、リスト、カスタムの各種分析ルールがサポートされています。

設定済みテーブルごとに 1 つの分析ルールのみを割り当てることができます。

> **Important**
>
> Cryptographic Computing for Clean Rooms を使用しており、コラボレーション内のデータテーブルを暗号化している場合、暗号化された設定済みテーブルに追加する分析ルールは、データの暗号化方法と一致している必要があります。例えば、SELECT のデータ (集計分析ルール) を暗号化した場合、JOIN の分析ルール (リスト分析ルール) は追加しないでください。

AWS Clean Rooms で使用できる分析ルールの種類を理解するには、「の分析ルール AWS Clean Rooms」を参照してください。

集計分析ルールの詳細については、「集計分析ルール」を参照してください。

リスト分析ルールの詳細については、「リスト分析ルール」を参照してください。

カスタム分析ルールの詳細については、「のカスタム分析ルール AWS Clean Rooms」を参照してください。

これらのセクションを確認して理解したら、次の手順を実行できます。

トピック

集計分析ルールをテーブルに設定する (ガイドフロー)

集計分析ルールでは、COUNT、SUM、および AVG の関数を使用して、行レベルの情報を明らかにすることなく、任意のディメンションで統計を集約するクエリが可能になります。

この手順では、AWS Clean Rooms コンソールの [ガイドフロー] オプションを使用して設定済みテーブルに集計分析ルールを追加するプロセスを説明します。

集計分析ルールをテーブルに追加するには (ガイドフロー)

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. コンソールの左のナビゲーションペインで、[設定済みのテーブル] を選択します。
3. 設定済みテーブルを選択します。
4. 設定済みテーブルの詳細ページで、[分析ルールを設定] を選択します。
5. [ステップ 1: タイプを選択] の [タイプ] で、デフォルトで選択されている[集約] オプションを選択したままにします。
6. [作成方法] で [ガイドフロー]] を選択し、[次へ] を選択します。
7. [ステップ 2: クエリコントロールを指定] の [集約関数] で以下の操作を行います。

   a. ドロップダウンから集約関数を選択します。

      - COUNT
      - COUNT DISTINCT
      - SUM
      - SUM DISTINCT
      - AVG

   b. 集約関数で使用可能にする列を [列] ドロップダウンから選択します。

   c. (オプション) [別の関数を追加] を選択して別の集約関数を追加し、その関数に 1 つ以上の列を関連付けます。

      > **Note**
      >
      > 少なくとも 1 つの集約関数が必要です。

   d. (オプション) [削除] を選択して集約関数を削除します。

8. [結合コントロール] で以下の操作を行います。

   a. [テーブルの単独クエリを許可] オプションを 1 つ選択します。

選択内容	結果
いいえ、クエリできるのは重複部分だけです	テーブルにクエリを実行できるのは、クエリを行えるメンバーが所有するテーブルに結合されている場合のみになります。
はい	テーブルは単独でクエリを実行することも、他のテーブルに結合している場合にクエリを実行することもできます。

   b. [結合列を指定] で、INNER JOIN ステートメントでの使用を許可する列を選択します。

      前のステップで [はい] を選択した場合、これは省略可能です。

   c. [照合に使用できる演算子を指定してください] で、複数の結合列の照合で使用可能にする演算子 (ある場合) を選択します。2 つ以上の JOIN 列を選択した場合、これらの演算子のいずれかが必要です。

選択内容	結果
AND	INNER JOIN 一致条件に AND を含めて、テーブル間で 1 つの列を別の列に結合できます。
または	INNER JOIN 一致条件に OR を含めて、テーブル間での複数の列一致を組み合わせることができます。この論理演算子は一致率を高めるのに便利です。

9. (オプション) [ディメンションコントロール] の [ディメンション列を指定] ドロップダウンで、SELECT ステートメントと、クエリの WHERE、GROUP BY、および ORDER BY 部分での使用を許可する列を選択します。

> **Note**
>
> 集約関数または結合列はディメンション列として使用できません。

10. [スカラー関数] の [許可するスカラー関数] でオプションを 1 つ選択します。

選択内容	結果
現在 AWS Clean Rooms でサポートされているすべての関数	AWS Clean Rooms で現在サポートされているすべてのスカラー関数を許可します。 • [リストを表示] を選択すると、[AWS Clean Rooms でサポートされているスカラー関数] の全リストが表示されます。
カスタムリスト	どのスカラー関数を許可するかをカスタマイズできます。 • [許可されるスカラー関数を指定] ドロップダウンから 1 つまたは複数のオプションを選択します。
[None] (なし)	スカラー関数を許可しません。

詳細については、「スカラー関数」を参照してください。

11. [次へ] を選択します。
12. [ステップ 3: クエリ結果コントロールを指定] の [集約制約] で以下の操作を行います。
    a. 各 [列名] のドロップダウンリストを選択します。
    b. COUNT DISTINCT 関数を適用した後に、返される各出力行が満たす必要がある [個別値の最小数] をドロップダウンリストから選択します。
    c. [制約を追加] を選択して集約制約を追加します。
    d. (オプション) [削除] を選択して集約制約を削除します。
13. [Next] (次へ) をクリックします。

14. [ステップ 4: 確認して設定] で、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て、[分析ルールを設定] を選択します。

集計分析ルールがテーブルに正常に設定されたことを示す確認メッセージが表示されます。

リスト分析ルールをテーブルに設定する (ガイドフロー)

リスト分析ルールでは、クエリを行えるメンバーのテーブルと関連するテーブルとの重複部分について行レベルのリストを出力するクエリが可能になります。

この手順では、AWS Clean Rooms コンソールの [ガイドフロー] オプションを使用して設定済みテーブルにリスト分析ルールを追加するプロセスを説明します。

リスト分析ルールをテーブルに追加するには (ガイドフロー)

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. コンソールの左のナビゲーションペインで、[設定済みのテーブル] を選択します。
3. 設定済みテーブルを選択します。
4. 設定済みテーブルの詳細ページで、[分析ルールを設定] を選択します。
5. [ステップ 1: タイプを選択] の [タイプ] で、[リスト] オプションを選択します。
6. [作成方法] で [ガイドフロー]] を選択し、[次へ] を選択します。
7. [ステップ 2: クエリコントロールを指定] の [結合コントロール] で以下の操作を行います。

    a. [結合列を指定] で、INNER JOIN ステートメントでの使用を許可する列を選択します。

    b. [照合に使用できる演算子を指定してください] で、複数の結合列の照合で使用可能にする演算子 (ある場合) を選択します。2 つ以上の JOIN 列を選択した場合、これらの演算子のいずれかが必要です。

選択内容	結果
AND	INNER JOIN 一致条件に AND を含めて、テーブル間で 1 つの列を別の列に結合できます。
または	INNER JOIN 一致条件に OR を含めて、テーブル間での複数の列一致を組み合わせ

選択内容	結果
	ることができます。この論理演算子は一致率を高めるのに便利です。

8. (オプション) [リストコントロール] の [リスト列を指定] ドロップダウンで、クエリ出力での使用 (つまり、SELECT ステートメントでの使用)、または結果のフィルター処理での使用 (つまり WHERE ステートメントでの使用) を許可する列を選択します。
9. [Next] (次へ) をクリックします。
10. [ステップ 3: 確認して設定] で、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て、[分析ルールを設定] を選択します。

リスト分析ルールがテーブルに正常に設定されたことを示す確認メッセージが表示されます。

カスタム分析ルールをテーブルに設定する (ガイドフロー)

カスタム分析ルールによって、設定済みテーブルでカスタム SQL クエリが有効になります。分析テンプレートまたは差分プライバシーを使用する場合は、カスタム分析ルールが必要です。

この手順では、AWS Clean Rooms コンソールの [ガイドフロー] オプションを使用して設定済みテーブルにカスタム分析ルールを追加するプロセスを説明します。

カスタム分析ルールをテーブルに追加するには (ガイドフロー)

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. コンソールの左のナビゲーションペインで、[設定済みのテーブル] を選択します。
3. 設定済みテーブルを選択します。
4. 設定済みテーブルの詳細ページで、[分析ルールを設定] を選択します。
5. [ステップ 1: タイプを選択] の [タイプ] で、[カスタム] オプションを選択します。
6. [作成方法] で [ガイドフロー]] を選択し、[次へ] を選択します。
7. [ステップ 2: 差分プライバシーの設定] で、差分プライバシーを有効にするか無効にするかを決定します。差分プライバシーは、再識別攻撃からデータを保護するための、数学的に証明された手法です。

    a. 差分プライバシーの場合:

目的	選択内容
ユーザーレベルのデータがあり、再識別攻撃から保護したい	[オンにする]
ユーザーレベルのデータがないか、再識別攻撃からの保護を必要としない	オフにする

b. 差分プライバシーを [オンにする] 場合は、プライバシーを保護したいユーザーの一意の識別子 (`user_id` 列など) を含む [ユーザー識別子列] を選択します。コラボレーション内の 2 つ以上のテーブルで差分プライバシーを有効にする場合は、両方の分析ルールで同じ列を [ユーザー識別子列] として設定して、テーブル間でユーザーの定義の一貫性を保つ必要があります。設定に誤りがある場合、クエリを実行できるメンバーには、クエリの実行中にユーザー投稿数 (ユーザーによる広告インプレッション数など) を計算するために選択できる列が 2 つあるというエラーメッセージが表示されます。

c. [Next] (次へ) をクリックします。

8. [ステップ 3: クエリコントロールを指定] で以下の操作を行います。

a. [コントロールタイプ] でいずれかを行います。

目的	選択内容
新しい各分析テンプレートを設定済みテーブルで実行する前に確認する	このテーブルでの実行を許可する前に新しい各分析テンプレートを確認する
設定済みテーブルで分析テンプレートやダイレクトクエリを実行できるようにする	特定のコラボレーターが作成したクエリはすべて、確認なしでこのテーブルでの実行を許可する

b. 以下のうちのひとつを選択します。

選択内容	操作
このテーブルでの実行を許可する前に新しい各分析テンプレートを確認する	[実行が許可された分析テンプレート] で [分析テンプレートを追加] を選択し、ドロップダウンリストから適切な [コラボ

選択内容	操作
	レーション] と [分析テンプレート] を選択します。
特定のコラボレーターが作成したクエリはすべて、確認なしでこのテーブルでの実行を許可する	[あらゆるクエリの作成が許可された AWS アカウント] で [AWS アカウントを追加] を選択し、適切な [AWS アカウント ID] を選択します。

9. [Next] (次へ) をクリックします。
10. [ステップ 4: 確認して設定] で、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て、[分析ルールを設定] を選択します。

カスタム分析ルールがテーブルに正常に設定されたことを示す確認メッセージが表示されます。

分析ルールをテーブルに設定する (JSON エディタ)

以下の手順は、AWS Clean Rooms コンソールの [JSON エディタ] オプションを使用して分析ルールをテーブルに追加する方法を示しています。

集計、リスト、またはカスタム分析ルールをテーブルに設定するには (JSON エディタ)

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. コンソールの左のナビゲーションペインで、[設定済みのテーブル] を選択します。
3. 設定済みテーブルを選択します。
4. 設定済みテーブルの詳細ページで、[分析ルールを設定] を選択します。
5. [ステップ 1: タイプを選択] の [タイプ] で [集約]、[リスト]、または [カスタム] オプションを選択します。
6. [作成方法] で [JSON エディタ] を選択し、[次へ] を選択します。
7. [ステップ 2: コントロールを指定] で、クエリ構造を挿入するか ([テンプレートを挿入])、ファイルを挿入するか ([ファイルからインポート]) を選択します。

選択内容	操作
テンプレートを挿入	1. [分析ルール定義] で、選択した分析ルールのパラメータを指定します。 2. Ctrl + スペースバー を押すとオートコンプリートが有効になります。 集計分析ルールパラメータの詳細については、「集計分析ルール - クエリコントロール」を参照してください。 リスト分析パラメータの詳細については、「リスト分析ルール - クエリコントロール」を参照してください。
ファイルからインポート	1. ローカルドライブから JSON ファイルを選択します。 2. 開く をクリックします。 アップロードされたファイルの分析ルールが [分析ルール定義] に表示されます。

8. [Next] (次へ) をクリックします。
9. [ステップ 3: 確認して設定] で、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て、[分析ルールを設定] を選択します。

分析ルールがテーブルに正常に設定されたことを示す確認メッセージが表示されます。

次のステップ

設定済みテーブルに分析ルールを設定したら、次の作業に進むことができます。

- 設定済みテーブルをコラボレーションに関連付ける
- (クエリを行えるメンバーとして) データテーブルにクエリを実行する

設定済みテーブルのコラボレーションへの関連付け

設定済みテーブルを作成して分析ルールを追加したら、そのテーブルをコラボレーションに関連付けることができます。

> **Important**
>
> 設定済み AWS Glue テーブルをコラボレーションに関連付ける前に、 AWS Glue テーブルの場所は 1 つのファイルではなく、Amazon Simple Storage Service (Amazon S3) フォルダを指す必要があります。この場所を確認するには、https://console.aws.amazon.com/glue/ の AWS Glue コンソールで テーブルを表示します。

> **Note**
>
> で暗号化を設定し AWS Glue 、サービスロールを作成した場合は、そのロールに AWS Glue テーブルの復号 AWS KMS keys 化に使用するアクセス権を付与する必要があります。
> で AWS KMS暗号化された Amazon S3 データセットにバックアップされた設定済みテーブルを関連付けた場合は、KMS キーを使用して Amazon S3 データを復号するためのアクセス許可をロールに付与する必要があります。
> 詳細については、「AWS Glue デベロッパーガイド」の「AWS Glueでの暗号化のセットアップ」を参照してください。

以下のトピックでは、 AWS Clean Rooms コンソールを使用して設定済みテーブルをコラボレーションに関連付ける方法について説明します。

トピック

- 設定済みテーブルの詳細ページから設定済みテーブルを関連付ける
- コラボレーションの詳細ページから設定済みテーブルを関連付ける
- 次のステップ

AWS SDK を使用して設定済みテーブルをコラボレーションに関連付ける方法については、「AWS Clean Rooms API リファレンス」を参照してください。

設定済みテーブルの詳細ページから設定済みテーブルを関連付ける

設定済み AWS Glue テーブルの詳細ページからテーブルをコラボレーションに関連付けるには

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. コンソールの左のナビゲーションペインで、[設定済みのテーブル] を選択します。
3. 設定済みテーブルを選択します。
4. 設定済みテーブルの詳細ページで、[コラボレーションに関連付ける] を選択します。
5. [テーブルをコラボレーションに関連付ける] ダイアログボックスで、ドロップダウンリストから[コラボレーション] を選択します。
6. [コラボレーションを選ぶ] を選択します。

   [テーブルを関連付ける] ページの [設定済みのテーブルを選択] セクションに、選択した設定済みテーブルの名前が表示されます。
7. [設定済みのテーブルを選択] で次の操作を行います。

目的が	操作
新しいテーブルを設定する	[テーブルを設定] を選択し、[テーブルを設定] ページのプロンプトに従います。
設定済みテーブルのスキーマと分析ルールを表示する	[スキーマと分析ルールを表示] をオンにします。

8. [新しいサービスロールを作成して使用] または [既存のサービスロールを使用] を選択して、[サービスアクセス] 許可を指定します。

選択内容	結果
新しいサービスロールを作成して使用	• AWS Clean Rooms 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 `cleanrooms-<timestamp>` です。

選択内容	結果
	• 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 キーで暗号化され、データ入力の復号 AWS KMS key に使用されるを入力できます。
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。 2. IAM 外部リンクで表示を選択して、サービスロールを表示します。 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。 デフォルトでは、 AWS Clean Rooms は、必要なアクセス許可を追加するために既存のロールポリシーの更新は試みません。 3. (オプション) [必要なアクセス許可を備えた事前設定済みポリシーをこのロールに追加] チェックボックスを選択して、必要なアクセス許可をロールにアタッチします。ロールを変更したりポリシーを作成したりするには、アクセス許可が必要です。

> **Note**
>
> - AWS Clean Rooms には、分析ルールに従ってクエリを実行するアクセス許可が必要です。のアクセス許可の詳細については、 AWS Clean Rooms「」を参照してくださいAWS の マネージドポリシー AWS Clean Rooms。
> - ロールに に対する十分なアクセス許可がない場合 AWS Clean Rooms、ロールに に対する十分なアクセス許可がないことを示すエラーメッセージが表示されます AWS Clean Rooms。続行する前に、ロールポリシーを追加する必要があります。
> - ロールポリシーを変更できない場合は、 AWS Clean Rooms でサービスロールのポリシーが見つからなかったという内容のエラーメッセージが表示されます。

9. 設定済みテーブルの関連付けリソースで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。
10. [テーブルを関連付ける] を選択します。

コラボレーションの詳細ページから設定済みテーブルを関連付ける

コラボレーションの詳細ページから AWS Glue テーブルをコラボレーションに関連付けるには

1. にサインイン AWS Management Console し、 で AWS アカウント AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. コラボレーションを選択します。
4. [テーブル] タブで [テーブルを関連付ける] を選択します。
5. [設定済みのテーブルを選択] で次の操作を行います。

目的が	操作
既存の設定済みテーブルを選択する	コラボレーションに関連付ける設定済みのテーブル名をドロップダウンリストから選択します。

目的が	操作
新しいテーブルを設定する	[テーブルを設定] を選択し、[テーブルを設定] ページのプロンプトに従います。
設定済みテーブルのスキーマと分析ルールを表示する	[スキーマと分析ルールを表示] をオンにします。

6. [テーブルの関連付けの詳細] で次の操作を行います。

   a. 関連付けるテーブルの[名前] を入力します。

      デフォルトの名前を使用することも、テーブルの名前を変更することもできます。

   b. (オプション) テーブルの [説明] を入力します。

      この説明はクエリの作成に役立ちます。

7. [新しいサービスロールを作成して使用] または [既存のサービスロールを使用] を選択して、[サービスアクセス] 許可を指定します。

選択内容	結果
新しいサービスロールを作成して使用	• AWS Clean Rooms 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 `cleanrooms-<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 キーで暗号化され、データ入力の復号 AWS KMS key に使用されるを入力できます。
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。

選択内容	結果
	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。 2. IAM 外部リンクで表示 を選択して、サービスロールを表示します。 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。 デフォルトでは、 AWS Clean Rooms 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。 3. (オプション) [必要なアクセス許可を備えた事前設定済みポリシーをこのロールに追加] チェックボックスを選択して、必要なアクセス許可をロールにアタッチします。ロールを変更したりポリシーを作成したりするには、アクセス許可が必要です。

> **Note**
>
> - AWS Clean Rooms には、分析ルールに従ってクエリを実行するアクセス許可が必要です。のアクセス許可の詳細については、 AWS Clean Rooms「」を参照してくださいAWS の マネージドポリシー AWS Clean Rooms。
> - ロールに に対する十分なアクセス許可がない場合 AWS Clean Rooms、ロールに に対する十分なアクセス許可がないことを示すエラーメッセージが表示されます AWS Clean Rooms。続行する前に、ロールポリシーを追加する必要があります。
> - ロールポリシーを変更できない場合は、 AWS Clean Rooms でサービスロールのポリシーが見つからなかったという内容のエラーメッセージが表示されます。

8. 設定済みテーブルの関連付けリソースで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。

9. [テーブルを関連付ける] を選択します。

次のステップ

設定済みデータテーブルをコラボレーションに関連付けたら、次の作業に進むことができます。

- コラボレーションクリエーターの場合はコラボレーションを編集する
- (クエリを行えるメンバーとして) データテーブルにクエリを実行する

差分プライバシーポリシーを設定する

この手順では、 AWS Clean Rooms コンソールのガイドフローオプションを使用して、コラボレーションで差分プライバシーポリシーを設定するプロセスについて説明します。これは、差分プライバシー保護のあるすべてのテーブルに対する 1 回限りの手順です。

差分プライバシー設定を設定するには (ガイドフロー)

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. コラボレーションを選択します。
4. コラボレーションページの [テーブル] タブで、[差分プライバシーポリシーを設定] を選択します。
5. [差分プライバシーポリシーを設定] ページで、次のプロパティの値を選択します。

   - プライバシー予算
   - プライバシー予算を毎月更新
   - クエリごとに追加されるノイズ

   デフォルト値を使用するか、特定のユースケースをサポートするカスタム値を入力できます。[プライバシー予算] と [クエリごとに追加されるノイズ] の値を選択した後、データに対するすべてのクエリ全体で使用可能な集計の数に関して、結果のユーティリティをプレビューできます。
6. [設定] を選択します。

コラボレーションの差分プライバシーポリシーが正常に設定されたことを示す確認メッセージが表示されます。

次のステップ

差分プライバシーが設定され、次の作業に進むことができます。

- (クエリを行えるメンバーとして) データテーブルにクエリを実行する
- (コラボレーションクリエーターの場合) コラボレーションを管理する

分析テンプレートの使用

分析テンプレートは のカスタム分析ルール AWS Clean Roomsで使用します。分析テンプレートを使用すると、同じクエリを再利用するのに役立つパラメータを定義できます。 は、リテラル値を含むパラメータ化のサブセット AWS Clean Rooms をサポートします。

分析テンプレートはコラボレーションごとに固有となります。各コラボレーションのメンバーは、そのコラボレーション内のクエリのみを表示できます。コラボレーションで差分プライバシーを使用する予定がある場合は、分析テンプレートが AWS Clean Rooms 差分プライバシーの汎用クエリ構造と互換性があることを確認する必要があります。

トピック

- 分析テンプレートの作成
- 分析テンプレートの確認
- 分析テンプレートを使用した設定済みテーブルのクエリ

分析テンプレートの作成

AWS SDKsAWS Clean Rooms リファレンス を参照してください。

AWS Clean Rooms コンソールを使用して分析テンプレートを作成するには

1. にサインイン AWS Management Console し、コラボレーションクリエーターとして機能する AWS アカウント でAWS Clean Rooms コンソールを開きます。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. コラボレーションを選択します。
4. [テンプレート] タブで、[自分で作成した分析テンプレート] セクションに移動します。
5. [分析テンプレートを作成] を選択します。
6. [分析テンプレートを作成] ページの [詳細] に、[名前] と [説明] (オプション) を入力します。
7. [テーブル] には、コラボレーションに関連する設定済みテーブルが表示されます。
8. [定義] で以下の操作を行います。

    a. 分析テンプレートの定義を入力します。

    b. [インポート元] を選択して定義をインポートします。

c. (オプション) SQL エディタで、パラメータ名の前にコロン (:) を入力してパラメータを指定します。

例:

```
WHERE table1.date + :date_period > table1.date
```

9. 以前にパラメータを追加したことがある場合は、[パラメータ – オプション] で、パラメータ名ごとに [タイプ] と [既定値] (オプション) を選択します。
10. 設定済みテーブルのリソースで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。
11. [作成] を選択します。

これで次の作業に進むことができます。

- コラボレーションメンバーに分析テンプレートを確認できることを伝える (自身のデータにクエリを実行する場合は省略可能)

分析テンプレートの確認

他のコラボレーションメンバーが分析テンプレートを作成したら、それを確認して承認できます。分析テンプレートが承認されると、 のクエリで分析テンプレートを使用できます AWS Clean Rooms。

AWS Clean Rooms コンソールを使用して分析テンプレートを確認するには

1. にサインイン AWS Management Console し、コラボレーションクリエーターとして機能する AWS アカウント でAWS Clean Rooms コンソールを開きます。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. コラボレーションを選択します。
4. [テンプレート] タブで、[他のメンバーが作成した分析テンプレート] セクションに移動します。
5. 実行可能ステータスが「いいえ レビューが必要」である分析テンプレートを選択します。
6. [Review] (レビュー) を選択します。
7. 分析ルールの [概要]、[定義]、および [パラメータ] (ある場合) を確認します。
8. [定義で参照されるテーブル] に表示されている設定済みのテーブルを確認します。

各テーブルの横の [ステータス] には、[テンプレートは許可されていません] と表示されています。

9. テーブルを選択します。

オプション:	選択内容
分析テンプレートを承認する	テーブル の テンプレート。を選択して、承認を確認します。
分析テンプレートを承認しない	[許可しない] を選択します。

これで、分析テンプレートを使用してデータテーブルをクエリする準備ができました (クエリを行えるメンバーとして)。

分析テンプレートを使用した設定済みテーブルのクエリ

この手順では、 AWS Clean Rooms コンソールで分析テンプレートを使用して、カスタム分析ルールで設定済みテーブルをクエリする方法を示します。

分析テンプレートを使用して、カスタム分析ルールが追加された設定済みテーブルにクエリを実行するには

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. [自身のメンバー能力] のステータスが [クエリ] になっているコラボレーションを選択します。
4. [クエリ] タブの [テーブル] に、テーブルとそれに関連する分析ルールの種類 ([カスタム分析ルール]) が表示されます。

> **Note**
>
> 想定したテーブルが表示されない場合は、次のいずれかの理由が考えられます。
>
> - テーブルが関連付けられていない。
> - テーブルに分析ルールが設定されていない。

5. [分析] セクションで、ドロップダウンリストから分析テンプレートを選択します。
6. クエリで使用する分析テンプレートのパラメータの値を入力します。値は、パラメータで指定されたデータ型である必要があります。分析テンプレートを実行するたびに異なる値を使用できます。パラメータの空の または NULL値はサポートされていません。LIMIT 句でのパラメータの使用はサポートされていません。
7. [実行] を選択します。

> **Note**
>
> 結果を受け取れるメンバーがクエリ結果の設定を行っていないと、クエリを実行できません。

8. 引き続きパラメータを調整してクエリを再度実行するか、[+] ボタンを選択して新しいタブで新しいクエリを開始します。

コラボレーション内のデータに対するクエリの実行

クエリを行えるメンバーは、次のいずれかを実行できます。

- SQL コードエディタを使用して SQL クエリを手動で作成する。
- 分析ビルダー UI を使用して、SQL コードを記述することなくクエリを作成する。
- 承認済みの分析テンプレートを使用する。

クエリを行えるメンバーがコラボレーション内のテーブルに対して SQL クエリを実行すると、AWS Clean Rooms は関連するロールを引き受けて、ユーザーに代わってテーブルにアクセスします。 は、必要に応じて分析ルールを入力クエリとその出力 AWS Clean Rooms に適用します。

AWS Clean Rooms は、他のクエリエンジンとは異なる SQL クエリをサポートします。仕様については、「AWS Clean Rooms SQL リファレンス」を参照してください。差分プライバシーで保護されているデータテーブルでクエリを実行する場合は、クエリが AWS Clean Rooms 差分プライバシーの汎用クエリ構造と互換性があることを確認する必要があります。

> **Note**
>
> Cryptographic Computing for Clean Rooms を使用する場合、すべての SQL オペレーションが有効な結果を生成するわけではありません。例えば、暗号化された列に対して COUNT を実行することはできますが、暗号化された数値に対して SUM を実行するとエラーが発生します。また、誤ったクエリ結果が得られることもあります。例えば、シール列に SUM を実行するクエリではエラーが発生します。一方、シール列に対する GROUP BY クエリは成功するように見えますが、クリアテキストに対する GROUP BY クエリで生成されるグループとは異なるグループが生成されます。

以下のトピックでは、 AWS Clean Rooms コンソールを使用してコラボレーション内のデータにクエリを実行する方法を説明します。

トピック

- SQL コードエディタの使用
- 分析ビルダーの使用
- 差分プライバシーによるデータクエリ
- 最近のクエリの表示

- クエリの詳細の表示

StartProtectedQuery API オペレーションを直接呼び出す AWS Clean Rooms か、 AWS SDKs、 AWS Clean Rooms API リファレンス を参照してください。

クエリログ記録の詳細については、「クエリログイン AWS Clean Rooms」を参照してください。

> **Note**
>
> 暗号化されたデータテーブルにクエリを実行すると、暗号化された列の結果は暗号化されます。

クエリ結果の受け取りについては、「クエリ結果の受信」を参照してください。

SQL コードエディタの使用

クエリを行えるメンバーは、SQL コードエディタで SQL コードを記述して手動でクエリを作成できます。SQL コードエディタは、 AWS Clean Rooms コンソールのクエリタブの分析セクションにあります。

SQL コードエディタはデフォルトで表示されます。分析ビルダーを使用してクエリを作成する場合は、「分析ビルダーの使用」を参照してください。

> **Important**
>
> コードエディタで SQL クエリの作成を開始してから [分析ビルダー UI] をオンにした場合、クエリは保存されません。

AWS Clean Rooms は、多くの SQL コマンド、関数、および条件をサポートしています。詳細については、「AWS Clean Rooms SQL リファレンス」を参照してください。

> **Tip**
>
> スケジュールされた保守管理がクエリの実行中に発生した場合、クエリは終了し、ロールバックされます。この場合は、クエリをやり直す必要があります。

SQL コードエディタを使用してクエリを手動で作成するには

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. [自身のメンバー能力] のステータスが [クエリ] になっているコラボレーションを選択します。
4. [クエリ] タブの [分析] セクションに移動します。

> **Note**
>
> [分析] セクションは、結果を受け取れるメンバーとクエリの計算コストを負担するメンバーが、アクティブなメンバーとしてコラボレーションに参加している場合にのみ表示されます。

5. [クエリ] タブの [テーブル] に、テーブルのリストとそれに関連する分析ルールの種類 (集計分析ルール、リスト分析ルール、またはカスタム分析ルール) が表示されます。

> **Note**
>
> 想定したテーブルが表示されない場合は、次のいずれかの理由が考えられます。
>
> - テーブルが関連付けられていない。
> - テーブルに分析ルールが設定されていない。

6. (オプション) テーブルのスキーマコントロールと分析ルールコントロールを表示するには、プラス記号アイコン (+) を選択してテーブルを展開します。
7. SQL コードエディタにクエリを入力してクエリを作成します。

(オプション) クエリの例を使用する場合	(オプション) 列名または関数を挿入する場合
1. テーブルの横にある 3 つの縦のドットを選択します。 2. [エディタに挿入] で [クエリの例] を選択します。	1. 列の横にある 3 つの縦のドットを選択します。 2. [エディタに挿入] で [列名] を選択します。 3. 列で許可されている関数を手動で挿入するには、列の横に

(オプション) クエリの例を使用する場合	(オプション) 列名または関数を挿入する場合
Note クエリの例を挿入すると、エディタに既に表示されているクエリに追加されます。 クエリの例が表示されます。[テーブル] の下に表示されているすべてのテーブルがクエリに含まれます。 3. クエリのプレースホルダー値を編集します。	ある 3 つの縦のドットを選択し、[エディタに挿入] を選択してから、許可されている関数の名前 (INNER JOIN、SUM、SUM DISTINCT、または COUNT) を選択します。 4. Ctrl + スペースキーを押すと、コードエディタにテーブルスキーマが表示されます。 **Note** クエリを行えるメンバーは、各設定済みテーブルの関連付けのパーティション列を表示して、使用できます。パーティション列が、設定済み AWS Glue テーブルの基盤となるテーブルのパーティション列としてラベル付けされていることを確認します。 5. クエリのプレースホルダー値を編集します。

8. [実行] を選択します。

Note

結果を受け取れるメンバーがクエリ結果の設定を行っていないと、クエリを実行できません。

9. 引き続きパラメータを調整してクエリを再度実行するか、[+] ボタンを選択して新しいタブで新しいクエリを開始します。

> **Note**
>
> AWS Clean Rooms は、明確なエラーメッセージを提供することを目的としています。トラブルシューティングに役立つ情報がエラーメッセージに不足している場合は、アカウントチームに連絡し、エラーが発生した経緯とエラーメッセージ (ID を含む) の詳細を伝えてください。詳細については、「トラブルシューティング AWS Clean Rooms」を参照してください。

分析ビルダーの使用

分析ビルダーを使用すると、SQL コードを記述しなくてもクエリを作成できます。分析ビルダーでは、次のようなテーブルが含まれるコラボレーションのクエリを作成できます。

- JOIN を必要とせず、集計分析ルールを使用する 1 つのテーブル
- 集計分析ルールを両方で使用する 2 つのテーブル (各メンバーから 1 つ)
- リスト分析ルールを両方で使用する 2 つのテーブル (各メンバーから 1 つ)
- 集計分析ルールを両方で使用する 2 つのテーブル (各メンバーから 1 つ) と、リスト分析ルールを両方で使用する 2 つのテーブル (各メンバーから 1 つ)

SQL クエリを手動で記述する場合は、「SQL コードエディタの使用」を参照してください。

分析ビルダーは、 AWS Clean Rooms コンソールの [クエリ] タブの [分析] セクションに [分析ビルダー UI] オプションとして表示されます。

> **Important**
>
> [分析ビルダー UI] をオンにし、分析ビルダーでクエリの作成を開始してから、[分析ビルダー UI] をオフにした場合、クエリは保存されません。

> **Tip**
>
> スケジュールされた保守管理がクエリの実行中に発生した場合、クエリは終了し、ロールバックされます。この場合は、クエリをやり直す必要があります。

以下のトピックでは、分析ビルダの使用方法について説明します。

トピック

分析ビルダーを使用して 1 つのテーブルにクエリを実行する (集計)

この手順では、 AWS Clean Rooms コンソールで Analysis Builder UI を使用してクエリを構築する方法を示します。このクエリは、JOIN を必要とせず、集計分析ルールを使用するテーブルが 1 つだけ含まれるコラボレーション用です。

分析ビルダーを使用して 1 つのテーブルにクエリを実行するには

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. [自身のメンバー能力] のステータスが [クエリ] になっているコラボレーションを選択します。
4. [クエリ] タブの [テーブル] に、テーブルとそれに関連する分析ルールの種類が表示されます (分析ルールの種類は集計分析ルールでなければなりません)。

   > **Note**
   >
   > 想定したテーブルが表示されない場合は、次のいずれかの理由が考えられます。
   >
   > - テーブルが関連付けられていない。
   > - テーブルに分析ルールが設定されていない。

5. [分析] セクションで、[分析ビルダー UI] をオンにします。
6. クエリを作成します。

集計メトリクスをすべて表示する場合は、ステップ 9 に進んでください。

a. [メトリクスを選択] で、デフォルトで事前に選択されている集計メトリクスを確認し、必要に応じてメトリクスを削除します。

b. (オプション) [セグメントを追加 – オプション] で、1 つ以上のパラメータを選択します。

> **Note**
>
> [セグメントを追加 – オプション] はテーブルにディメンションが指定されている場合にのみ表示されます。

c. (オプション) [フィルターを追加 – オプション] で [フィルターを追加] を選択し、パラメータ、演算子、および値を選択します。

さらにフィルターを追加するには、[別のフィルターを追加] を選択します。

フィルターを削除するには、[削除] を選択します。

> **Note**
>
> 集約クエリでは ORDER BY はサポートされていません。
> AND 演算子のみがフィルターでサポートされています。

d. (オプション) [説明を追加 – オプション] に、クエリのリストでクエリを識別しやすくするための説明を入力します。

7. [SQL コードをプレビュー] を展開します。

a. 分析ビルダーから生成された SQL コードを表示します。

b. SQL コードをコピーするには、[コピー] を選択します。

c. SQL コードを編集するには、[SQL コードエディタで編集] を選択します。

8. [実行] を選択します。

> **Note**
>
> 結果を受け取れるメンバーがクエリ結果の設定を行っていないと、クエリを実行できません。

9. 引き続きパラメータを調整してクエリを再度実行するか、[+] ボタンを選択して新しいタブで新しいクエリを開始します。

> **Note**
>
> AWS Clean Rooms は、明確なエラーメッセージを提供することを目的としています。トラブルシューティングに役立つ情報がエラーメッセージに不足している場合は、アカウントチームに連絡し、エラーが発生した経緯とエラーメッセージ (ID を含む) の詳細を伝えてください。詳細については、「トラブルシューティング AWS Clean Rooms」を参照してください。

分析ビルダーを使用して 2 つのテーブルにクエリを実行する (集計またはリスト)

この手順では、 AWS Clean Rooms コンソールで Analysis Builder を使用して、次の内容のコラボレーションのクエリを作成する方法について説明します。

- 集計分析ルールを両方で使用する 2 つのテーブル (各メンバーから 1 つ)
- リスト分析ルールを両方で使用する 2 つのテーブル (各メンバーから 1 つ)
- 集計分析ルールを両方で使用する 2 つのテーブル (各メンバーから 1 つ) と、リスト分析ルールを両方で使用する 2 つのテーブル (各メンバーから 1 つ)

分析ビルダーを使用して 2 つのテーブルにクエリを実行する

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. [自身のメンバー能力] のステータスが [クエリ] になっているコラボレーションを選択します。
4. [クエリ] タブの [テーブル] に、2 つのテーブルとそれに関連する分析ルールの種類 (集計分析ルールまたはリスト分析ルール) が表示されます。

> **Note**
>
> 想定したテーブルが表示されない場合は、次のいずれかの理由が考えられます。

- テーブルが関連付けられていない。
- テーブルに分析ルールが設定されていない。

5. [分析] セクションで、[分析ビルダー UI] をオンにします。
6. クエリを作成します。

コラボレーションに、集計分析ルールを使用する 2 つのテーブルと、リスト分析ルールを使用する 2 つのテーブルが含まれている場合は、まず [集約] または [リスト] のどちらかを選択し、選択した分析ルールに基づいてプロンプトに従います。

2 つのテーブルが集計分析ルールを使用している場合	2 つのテーブルがリスト分析ルールを使用している場合
1. [メトリクスを選択] で、デフォルトで事前に選択されている集計メトリクスを確認し、必要に応じてメトリクスを削除します。 2. [レコードを照合] で、1 つ以上のレコードを選択します。 **Note** 分析ビルダーを使用する場合、1 組の列でのみ照合できます。 3. (オプション) [セグメントを追加 – オプション] で、1 つ以上のパラメータを選択します。 **Note** [セグメントを追加 – オプション] はテーブルにディメンションが指定	1. [属性を選択] で、デフォルトで事前に選択されているリスト属性を確認し、必要に応じてメトリクスを削除します。 2. [レコードを照合] で、1 つ以上のレコードを選択します。 **Note** 分析ビルダーを使用する場合、1 組の列でのみ照合できます。 3. (オプション) [フィルターを追加 – オプション] で [フィルターを追加] を選択し、パラメータ、演算子、および値を選択します。 さらにフィルターを追加するには、[別のフィルターを追加] を選択します。

2 つのテーブルが集計分析ルールを使用している場合	2 つのテーブルがリスト分析ルールを使用している場合
されている場合にのみ表示されます。 4. (オプション) [フィルターを追加 – オプション] で [フィルターを追加] を選択し、パラメータ、演算子、および値を選択します。 さらにフィルターを追加するには、[別のフィルターを追加] を選択します。 フィルターを削除するには、[削除] を選択します。 Note 集約クエリでは ORDER BY はサポートされていません。AND 演算子のみがフィルターでサポートされています。 5. (オプション) [説明を追加 – オプション] に、最近のクエリのリストでクエリを識別しやすくするための説明を入力します。	フィルターを削除するには、[削除] を選択します。 Note リストクエリでは、LIMIT はサポートされていません。AND 演算子のみがフィルターでサポートされています。 4. (オプション) [説明を追加 – オプション] に、最近のクエリのリストでクエリを識別しやすくするための説明を入力します。

7. [SQL コードをプレビュー] を展開します。

   a. 分析ビルダーから生成された SQL コードを表示します。

   b. SQL コードをコピーするには、[コピー] を選択します。

   c. SQL コードを編集するには、[SQL コードエディタで編集] を選択します。

8. [実行] を選択します。

> **Note**
>
> 結果を受け取れるメンバーがクエリ結果の設定を行っていないと、クエリを実行できません。

9. 引き続きパラメータを調整してクエリを再度実行するか、[+] ボタンを選択して新しいタブで新しいクエリを開始します。

> **Note**
>
> AWS Clean Rooms は、明確なエラーメッセージを提供することを目的としています。トラブルシューティングに役立つ情報がエラーメッセージに不足している場合は、アカウントチームに連絡し、エラーが発生した経緯とエラーメッセージ (ID を含む) の詳細を伝えてください。詳細については、「トラブルシューティング AWS Clean Rooms」を参照してください。

差分プライバシーによるデータクエリ

一般に、差分プライバシーが有効になっていても、クエリの記述と実行は変わりません。ただし、プライバシー予算が十分に残っていないと、クエリを実行できません。クエリを実行してプライバシー予算を消費すると、実行可能な集計のおおよその数と、それが将来のクエリにどのような影響を与えるかがわかります。

コラボレーションにおける差分プライバシーの影響を調べるには

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. [メンバー詳細] のステータスが [クエリの実行] になっているコラボレーションを選択します。
4. [クエリ] タブの [テーブル] に、残りのプライバシー予算が表示されます。これは、[残りの集計関数] の推定数と [使用されたユーティリティ] (パーセンテージで表示) として表示されます 。

> **Note**
>
> [残りの集計関数] の推定数と [使用されたユーティリティ] の割合は、クエリを実行できるメンバーにのみ表示されます。

5. [影響を表示] を選択すると、結果にどの程度のノイズが加わり、実行できる集計関数のおおよその数が表示されます。

最近のクエリの表示

過去 90 日間に実行されたクエリは、[最近のクエリ] タブで確認できます。

> **Note**
>
> 自分のメンバー能力が [データを寄稿] のみで、[クエリの計算コストを負担するメンバー] でもない場合、コンソールに [クエリ] タブは表示されません。

最近のクエリを表示するには

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きますAWS アカウント（まだ開いていない場合）。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. コラボレーションを選択します。
4. [クエリ] タブの [クエリ] に、過去 90 日間に実行されたクエリが表示されます。
5. 最近のクエリをステータス別にソートするには、[すべてのステータス] ドロップダウンリストからステータスを選択します。

   ステータスには、[送信済み]、[開始]、[キャンセル済み]、[成功]、[失敗]、[タイムアウト] があります。

クエリの詳細の表示

クエリの詳細は、クエリを行えるメンバーとして、または結果を受け取れるメンバーとして表示できます。

クエリの詳細を表示するには

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント（まだ開いていない場合）。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. コラボレーションを選択します。
4. [クエリ] タブで、次のいずれかを実行します。

   - 表示する特定のクエリのオプションボタンを選択して、[詳細を表示] を選択します。
   - [保護されたクエリ ID] を選択します。
5. [クエリの詳細] ページで次の操作を行います。

   - クエリを行えるメンバーの場合は、[クエリの詳細]、[SQL テキスト]、および [結果] を確認できます。

     結果を受け取れるメンバーにクエリ結果が送信されたことを通知するメッセージが表示されます。
   - 結果を受け取れるメンバーの場合は、[クエリの詳細] と [結果] を確認できます。

クエリ結果の受信

結果を受け取れるメンバーは、AWS Clean Rooms から、コラボレーションに参加したときに指定した Amazon S3 バケットでクエリの出力を受け取ることができます。

以下のトピックでは、AWS Clean Rooms コンソールを使用してクエリ結果を受け取る方法を説明します。

トピック

AWS Clean Rooms の API を直接呼び出すことで、または AWS SDK を使用することで、データに対してクエリを実行したり、クエリを表示したりする方法については、「AWS Clean Rooms API リファレンス」を参照してください。

クエリログ記録の詳細については、「クエリログイン AWS Clean Rooms」を参照してください。

> **Note**
>
> 暗号化されたデータテーブルにクエリを実行すると、暗号化された列の結果は暗号化されます。

クエリ結果の受信

クエリ結果は、AWS Clean Rooms コンソールの [クエリ] タブの [クエリ結果のデフォルト設定] セクションと [クエリ] セクションにあります。

クエリ結果を受け取るには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. [自身のメンバー能力] のステータスが [結果を受け取る] になっているコラボレーションを選択します。

4. クエリ結果を AWS Clean Rooms から直接受け取るには、[クエリ] タブの [クエリ] にある [保護されたクエリ ID] 列でクエリを選択します。
5. [クエリの詳細] ページの [結果] で、次のいずれかを実行します。

目的	選択内容
結果をコピーする。	[Copy] (コピー)
結果をダウンロードする。	ダウンロード **Note** デフォルトでは、ダウンロードされたファイルの名前は、AWS Clean Rooms でクエリが実行されたときに表示された `Query id` と同じものになります。
Amazon S3 で結果を表示する。	Amazon S3 で表示 別のタブで IAM コンソールが開きます。

6. 暗号化されたデータを使用している場合は、ここでデータテーブルを復号化できます。

   詳細については、「C3R 暗号化クライアントによるデータテーブルの復号化」を参照してください。

クエリ結果設定のデフォルト値の編集

結果を受け取れるメンバーは、AWS Clean Rooms コンソールでクエリ結果設定のデフォルト値を編集できます。

クエリ結果設定のデフォルト値を編集するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。

2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. [自身のメンバー能力] のステータスが [結果を受け取る] になっているコラボレーションを選択します。
4. [クエリ] タブの [クエリ結果設定] で、[編集] を選択します。
5. [クエリ結果の設定を編集] ページで、必要に応じて次のいずれかを変更します。

    a. [クエリ結果設定] で、[Amazon S3 内の結果の送信先] または [結果フォーマット] を変更します。

    b. [サービスアクセス] で [AWS Clean Rooms を認証する方法] を変更し、指定した Amazon S3 バケットとフォーマットでの書き込みを許可します。

    更新した [クエリ結果設定] がコラボレーションの詳細ページに表示されます。

他の AWS のサービスでのクエリ出力の使用

AWS Clean Rooms からのクエリ出力は、コンソールで表示され (コンソールを使用してクエリを実行した場合)、指定された Amazon S3 バケットにダウンロードされます。そこから、Amazon QuickSight や Amazon SageMaker などの他の AWS のサービスでもクエリ出力を使用できますが、使用するサービスは Amazon S3 からのデータをどのように使用するかによって異なります。

Amazon QuickSight の詳細については、Amazon QuickSight のドキュメントを参照してください。

Amazon SageMaker の詳細については、Amazon SageMaker のドキュメントを参照してください。

C3R 暗号化クライアントによるデータテーブルの復号化

Cryptographic Computing for Clean Rooms と C3R 暗号化クライアントを使用してデータテーブルを暗号化するコラボレーションでは、以下の手順に従ってください。この手順は、コラボレーションでデータにクエリを実行した後に使用します。

この手順には、共有シークレットキーとコラボレーション ID が必要です。

結果を受け取れるメンバーは、コラボレーションのデータを暗号化するときに使用されたものと同じ共有シークレットキーとコラボレーション ID を使用してデータを復号化します。

> **Note**
>
> AWS Clean Rooms のコラボレーションでは、クエリを実行したり結果を表示したりできるユーザーが既に制限されています。結果にアクセスできるユーザーが誰であっても、復号化を実行するには、データの暗号化に使用されたものと同じ共有シークレットキーとコラボレーション ID が必要になります。

暗号化されたデータテーブルを復号化するには

1. (オプション) C3R 暗号化クライアントで使用可能なコマンドを確認します。
2. (オプション) 目的のディレクトリに移動し、`ls` (macOS) または `dir` (Windows) を実行します。

   - c3r-cli.jar ファイルと暗号化されたクエリ結果データファイルが目的のディレクトリにあることを確認します。

     > **Note**
     >
     > クエリ結果が AWS Clean Rooms コンソールのインターフェイスからダウンロードされた場合は、ユーザーアカウントの [ダウンロード] フォルダにある可能性があります (例えば、Windows および macOS のユーザーディレクトリにある [ダウンロード] フォルダ)。クエリ結果ファイルは c3r-cli.jar と同じフォルダに移動することをお勧めします。

3. 共有シークレットキーを `C3R_SHARED_SECRET` 環境変数に保存します。詳細については、「ステップ 6: 共有シークレットキーを環境変数に保存する」を参照してください。

4. AWS Command Line Interface (AWS CLI) から、次のコマンドを実行します。

   java -jar c3r-cli.jar decrypt ***<name of input .csv file>*** --id=***<collaboration id>*** --output=***<output file name>***

5. 各###############を独自の情報に置き換えます。

   a. `id=` には、コラボレーション ID を入力します。

   b. `output=` には、出力ファイルの名前 (例えば `results-decrypted.csv`) を入力します。

      出力名を指定しないと、デフォルトの名前がターミナルに表示されます。

   c. CSV または Parquet 表示用の任意のアプリケーション (Microsoft Excel、テキストエディタ、その他のアプリケーションなど) を使用して、指定した出力ファイル内の復号化されたデータを表示します。

の管理 AWS Clean Rooms

以下のトピックでは、 AWS Clean Rooms コンソール AWS Clean Rooms を使用して でコラボレーション、メンバー、および設定済みテーブルを管理する方法について説明します。

SDK AWS Clean Rooms を使用して を管理する方法については、 AWS Clean Rooms API リファレンスを参照してください。 AWS SDKs

トピック

- AWS Clean Rooms でのコラボレーションの管理
- での設定済みテーブルの管理 AWS Clean Rooms

AWS Clean Rooms でのコラボレーションの管理

以下のトピックでは、コラボレーションクリエーターが AWS Clean Rooms コンソールを使用して AWS Clean Rooms コラボレーションを管理する方法について説明します。

AWS SDK を使用してコラボレーションを管理する方法については、「AWS Clean Rooms API リファレンス」を参照してください。

トピック

- コラボレーションの編集
- コラボレーションの削除
- コラボレーションの表示
- テーブルと分析ルールの表示
- 差分プライバシー使用状況ログの表示
- メンバーステータスの監視
- コラボレーションからメンバーを削除する
- コラボレーションからの退出
- 設定済みテーブルの関連付けの編集
- 設定済みテーブルの関連付けの解除
- 差分プライバシーポリシーの編集

- 差分プライバシーポリシーの削除
- 計算された差分プライバシーパラメータの表示

コラボレーションの編集

コラボレーションのさまざまな部分を編集する方法を説明します。

トピック

- コラボレーションの名前と説明の編集
- コラボレーションのタグ編集
- メンバーシップのタグ編集
- 関連付けられているテーブルのタグ編集
- 分析テンプレートのタグ編集
- 差分プライバシーポリシータグの編集

コラボレーションの名前と説明の編集

コラボレーションの作成後は、コラボレーションの名前と説明のみを編集できます。

> **Note**
>
> [クエリログ記録] を有効にしている場合は、クエリログを Amazon CloudWatch Logs アカウントに保存するかどうかを編集できます。

コラボレーションの名前と説明を編集するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. 作成したコラボレーションを選択します。
4. コラボレーションの詳細ページで [アクション] を選択し、[コラボレーションを編集] を選択します。
5. [詳細] で、コラボレーションの [名前] と [説明] を編集します。

6. [Save changes] (変更の保存) をクリックします。

コラボレーションのタグ編集

コラボレーションクリエーターは、コラボレーションの作成後に、コラボレーションリソースのタグを管理できます。

コラボレーションのタグを編集するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. 作成したコラボレーションを選択します。
4. 以下のうちのひとつを選択します。

状況	操作
コラボレーションのメンバーである場合	[詳細] タブを選択します。
コラボレーションクリエーターだが、コラボレーションのメンバーではない場合	ページの [タグ] セクションまで下方向にスクロールします。

5. [コラボレーションの詳細] で、[タグを管理] を選択します。
6. [Manage tags] (タグの管理) ページで、次の操作を実行できます。
   - タグを削除するには、[削除] を選択します。
   - タグを追加するには、[新しいタグの追加] を選択します。
   - 変更を保存するには、[変更内容を保存] を選択します。

メンバーシップのタグ編集

コラボレーションクリエーターは、コラボレーションの作成後に、メンバーシップリソースのタグを管理できます。

メンバーシップのタグを編集するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。

2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. 作成したコラボレーションを選択します。
4. [詳細] タブを選択します。
5. [メンバーシップの詳細] で [タグを管理] を選択します。
6. [メンバーシップタグを管理] ページで、次の操作を実行できます。

   - タグを削除するには、[削除] を選択します。
   - タグを追加するには、[新しいタグの追加] を選択します。
   - 変更を保存するには、変更の保存を選択します。

関連付けられているテーブルのタグ編集

コラボレーションクリエーターは、テーブルをコラボレーションに関連付けた後に、関連付けられているテーブルリソースのタグを管理できます。

関連付けられているテーブルのタグを編集するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. 作成したコラボレーションを選択します。
4. [Tables] (テーブル) タブを選択します。
5. [自分が関連付けたテーブル] で、テーブルを選択します。
6. 設定済みテーブルの詳細ページの [タグ] で、[タグを管理] を選択します。

   [Manage tags] (タグの管理) ページで、次の操作を実行できます。

   - タグを削除するには、[削除] を選択します。
   - タグを追加するには、[新しいタグの追加] を選択します。
   - 変更を保存するには、変更の保存を選択します。

分析テンプレートのタグ編集

コラボレーションクリエーターは、コラボレーションの作成後に、分析テンプレートリソースのタグを管理できます。

メンバーシップのタグを編集するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. 作成したコラボレーションを選択します。
4. [Templates (テンプレート)] タブを選択します。
5. [自分で作成した分析テンプレート] セクションで、分析テンプレートを選択します。
6. 分析テンプレートテーブルの詳細ページで、[タグ] セクションまで下方向にスクロールします。
7. [Manage tags (タグの管理)] を選択します。
8. [Manage tags] (タグの管理) ページで、次の操作を実行できます。
   - タグを削除するには、[削除] を選択します。
   - タグを追加するには、[新しいタグの追加] を選択します。
   - 変更を保存するには、変更の保存を選択します。

差分プライバシーポリシータグの編集

コラボレーションクリエーターは、コラボレーションの作成後に、分析テンプレートリソースのタグを管理できます。

メンバーシップのタグを編集するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. 編集する差分プライバシーポリシーを含むコラボレーションを選択します。
4. [Tables] (テーブル) タブを選択します。
5. [タグ] タブで、[タグ管理] を選択します。
6. [Manage tags] (タグの管理) ページで、次の操作を実行できます。
   - タグを削除するには、[削除] を選択します。
   - タグを追加するには、[新しいタグの追加] を選択します。
   - 変更を保存するには、変更の保存を選択します。

コラボレーションの削除

コラボレーションクリエーターは、作成したコラボレーションを削除できます。

> **Note**
>
> コラボレーションを削除すると、作成者もどのメンバーも、クエリを実行したり、結果を受け取ったり、データを寄稿したりできません。コラボレーションの各メンバーは、メンバーシップの一環として引き続き自分のデータにアクセスできます。

コラボレーションを削除するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. 削除するコラボレーションを選択します。
4. [アクション] で [コラボレーションを削除] を選択します。
5. 削除を確定し、[削除] を選択します。

コラボレーションの表示

コラボレーションクリエーターは、自分が作成したすべてのコラボレーションを表示できます。

コラボレーションを表示するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. [コラボレーション] ページの [最終使用日] に、直近に使用した 5 つのコラボレーションが表示されます。
4. [アクティブメンバーシップあり] タブには、[アクティブメンバーシップとのコラボレーション] のリストが表示されます。

   [名前]、[メンバーシップの作成日]、[あなたのメンバー情報] で並べ替えることができます。

   [検索] バーを使用してコラボレーションを検索できます。

5. [参加可能] タブには、[参加可能なコラボレーション] のリストが表示されます。
6. [利用できなくなりました] タブには、削除されたコラボレーションのリストと、[利用できなくなったコラボレーションのメンバーシップ] (削除されたメンバーシップ) が表示されます。

テーブルと分析ルールの表示

コラボレーションに関連付けられたテーブルと分析ルールを表示するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. コラボレーションを選択します。
4. [Tables] (テーブル) タブを選択します。
5. 以下のうちのひとつを選択します。

   a. コラボレーションに関連付けられている自分のテーブルを表示するには、[自分が関連付けたテーブル] でテーブル (青色のテキスト) を選択します。

   b. コラボレーションに関連付けられている他のテーブルを表示するには、[コラボレーターが関連付けたテーブル] でテーブル (青色のテキスト) を選択します。
6. テーブルの詳細と分析ルールが、テーブルの詳細ページに表示されます。

差分プライバシー使用状況ログの表示

差分プライバシーでデータを保護しているコラボレーションメンバーは、差分プライバシーでコラボレーションを作成した後で、プライバシー予算の使用状況をモニタリングできます。

集計が行われた回数と使用されたプライバシー予算の金額を確認するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. コラボレーションを選択します。
4. [Tables] (テーブル) タブを選択します。
5. [使用状況ログを表示] (青色のテキスト) を選択します。
6. プライバシー予算や提供したユーティリティの提供量など、使用状況の詳細を表示します。

メンバーステータスの監視

コラボレーションクリエーターは、コラボレーションの作成後に、[メンバー] タブですべてのメンバーのステータスを監視できます。

メンバーのステータスを確認するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. 作成したコラボレーションを選択します。
4. [メンバー] タブを選択します。
5. 各メンバーの [メンバーステータス] を表示します。

コラボレーションからメンバーを削除する

> **Note**
>
> メンバーを削除すると、そのメンバーが関連付けられているデータセットもすべてコラボレーションから削除されます。

コラボレーションからメンバーを削除するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. 作成したコラボレーションを選択します。
4. [メンバー] タブを選択します。
5. 削除するメンバーの横にあるオプションボタンを選択します。

   > **Note**
   >
   > コラボレーションクリエーターは自分のアカウント ID を選択できません。

6. [削除] を選択します。

7. ダイアログボックスのテキスト入力フィールドに「**confirm**」と入力して、メンバーの削除を確定します。

> **Note**
>
> クエリの計算コストを負担するメンバーを削除すると、コラボレーション内でそれ以上クエリを実行できなくなります。

コラボレーションからの退出

コラボレーションメンバーは、自分のメンバーシップを削除することでコラボレーションから退出できます。コラボレーションクリエーターは、コラボレーションを削除しないとコラボレーションから退出できません。

> **Note**
>
> メンバーシップを削除すると、コラボレーションから退出することになり、再び参加することはできません。クエリの計算コストを負担するメンバーがメンバーシップを削除すると、それ以上クエリを実行できなくなります。

コラボレーションから退出するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. [アクティブメンバーシップあり] で、自分がメンバーになっているコラボレーションを選択します。
4. [アクション] を選択します。
5. [メンバーシップを削除] を選択します。
6. ダイアログボックスのテキスト入力フィールドに「**confirm**」と入力してコラボレーションからの退出を確定し、[メンバーシップを空にして削除] を選択します。

   メンバーシップが削除されたことを示すメッセージがコンソールに表示されます。

   コラボレーションクリエーターには、[メンバーステータス] に [退出済み] と表示されます。

設定済みテーブルの関連付けの編集

コラボレーションメンバーは、作成した設定済みテーブルの関連付けを編集できます。

設定済みテーブルの関連付けを編集するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. コラボレーションを選択します。
4. [テーブル] タブを選択します。
5. [自分が関連付けたテーブル] で、テーブルを選択します。
6. テーブルの詳細ページで、[テーブルの関連付けの詳細] まで下方向にスクロールします。
7. [編集] を選択します。
8. [設定済みのテーブル関連付けを編集] ページで、[説明] または [サービスアクセス情報] を更新します。
9. [Save changes] (変更の保存) をクリックします。

設定済みテーブルの関連付けの解除

コラボレーションメンバーは、設定済みテーブルとコラボレーションの関連付けを解除できます。このアクションにより、クエリを行えるメンバーはテーブルでクエリを実行できなくなります。

設定済みテーブルの関連付けを解除するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. コラボレーションを選択します。
4. [テーブル] タブを選択します。
5. [自分が関連付けたテーブル] で、関連付けを解除するテーブルの横にあるオプションボタンを選択します。
6. [Disassociate] (関連付け解除) を選択します。
7. ダイアログボックスで [関連付け解除] を選択すると、設定済みテーブルの関連付けの解除が確定し、クエリを行えるメンバーがテーブルでクエリを実行できなくなります。

差分プライバシーポリシーの編集

差分プライバシーポリシーを設定した後は、プライバシーのニーズをより適切に反映するようにいつでも更新できます。

差分プライバシーポリシーを編集するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. コラボレーションを選択します。
4. コラボレーションページの [テーブル] タブの [自分が関連付けたテーブル] で、[編集] を選択します。
5. [差分プライバシーの編集] ページで、次のプロパティの新しい値を選択します。

   - [プライバシー予算] — コラボレーション中の任意の時点で、スライダーバーを動かして予算を増減できます。クエリを実行できるメンバーがデータのクエリを開始した後は、予算を減らすことはできません。[プライバシー予算] が増加した場合、AWS Clean Rooms は、新しく追加されたプライバシー予算を使用する前に、既存の予算を完全に消費するまで使用し続けます。
   - [クエリごとに追加されるノイズ] — コラボレーション中の任意の時点で、スライダーバーを動かして [クエリごとに追加されるノイズ] を増減できます。

   > **Note**
   >
   > [インタラクティブサンプル] を選択して、[プライバシー予算] と [クエリごとに追加されるノイズ] の値が異なると、実行できる集計関数の数にどのように影響するかを調べることができます。

   [プライバシー予算の更新] の値は変更できません。選択を変更するには、差分プライバシーポリシーを削除し、新しく作成する必要があります。
6. [Save changes] (変更の保存) をクリックします。

差分プライバシーポリシーが正常に編集されたことを示す確認メッセージが表示されます。

差分プライバシーポリシーの削除

差分プライバシーポリシーはコラボレーションの [テーブル] タブから削除できます。

差分プライバシーポリシーを削除するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. コラボレーションを選択します。
4. コラボレーションページの [テーブル] タブで、[差分プライバシーポリシー] の横にある [削除] を選択します。
5. 差分プライバシーポリシーを削除してよい場合は、[削除] を選択します。

差分プライバシーポリシーを削除すると、そのポリシーのプライバシー予算使用状況ログにアクセスできなくなります。差分プライバシーポリシーが削除されると、差分プライバシーがオンになっているテーブルをクエリできなくなります。

計算された差分プライバシーパラメータの表示

差分プライバシーに関する専門知識を持つユーザーは、コラボレーションの [クエリ] タブから計算された差分プライバシーパラメータを表示できます。

計算された差分プライバシーパラメータを表示するには

1. 自身の AWS アカウントで AWS Management Consoleにサインインし、AWS Clean Rooms コンソールを開きます (まだ開いていない場合)。
2. 左のナビゲーションペインで、[コラボレーション] を選択します。
3. コラボレーションを選択します。
4. [クエリ] タブの [結果] セクションで、[計算された差分プライバシーパラメータを表示] を選択します。

[計算された差分プライバシーパラメータ] のテーブルでは、集計関数の感度値が表示されます。これは、1 人のユーザーのレコードが追加、削除、または変更された場合に関数の結果が変化する可能性のある最大値として定義されます。リストには、次の差分プライバシーパラメータが含まれます。

- [ユーザー寄与度上限] (UCL) は、SQL クエリでユーザーが入力する行の最大数です。例えば、各ユーザーが複数のインプレッションを持つことができる特定のキャンペーンで一致したインプレッションの合計数をカウントする場合、差分プライバシーの計算が正確になるように、AWS Clean Rooms の差分プライバシーは、単一のユーザーのインプレッション数を制限する必要があります。つまり、あるユーザーのインプレッション数が上限を超える場合、AWS Clean Rooms は自動的に、計算された UCL 値に従ってそのユーザーのインプレッションの均一なランダムサンプルを取得し、クエリの実行中にそのユーザーの残りのインプレッションを除外します。一意のユーザー数をカウントする場合、UCL 値は 1 に等しくなります。これは、1 人のユーザーを追加、削除、または変更することで、個々のユーザーのカウントが最大で 1 だけ変更される可能性があるためです。
- [最小値] は、`sum()` のような集計関数内で使用される式の下限値です。例えば、式が `purchase_value` という列の場合、最小値はその列の下限値です。
- [最大値] は、`sum()` のような集計関数内で使用される式の上限値です。例えば、式が `purchase_value` という列の場合、最大値はその列の上限値です。

[計算された差分プライバシーパラメータ] のテーブルでは、これらのパラメータを使用してクエリ結果のノイズの総量の理解を深めることができます。例えば、[クエリごとに追加されるノイズ] が 30 ユーザーに設定されている状態で `COUNT DISTINCT (user_id)` クエリを実行すると、`COUNT DISTINCT` の感度が 1 であるため、AWS Clean Rooms 差分プライバシーは高い確率で -30 ~ 30 の範囲のランダムなノイズを追加します。同じ設定の `COUNT` クエリの場合、AWS Clean Rooms 差分プライバシーは、1 人のユーザーがクエリ結果に複数の行を寄稿する可能性があるため、ユーザーの寄与度上限に応じてスケールされる統計的ノイズを追加します。`SUM (purchase_value)` のようなすべての列の値が正の値である `SUM` クエリの場合、ノイズの合計は、ユーザー寄与度上限に最大値を掛けた値によってスケーリングされます。AWS Clean Rooms差分プライバシーは、クエリの実行時に自動的に感度パラメータを計算してノイズを追加し、プライバシー予算を使い果たします。感度パラメータはデータに依存するため、プライバシー予算を使い果たす必要があります。

での設定済みテーブルの管理 AWS Clean Rooms

以下のトピックでは、 AWS Clean Rooms コンソール AWS Clean Rooms を使用して で設定済みテーブルを管理する方法について説明します。

AWS SDKsAWS Clean Rooms リファレンスを参照してください。

トピック

- 設定済みテーブルの詳細の編集

- 設定済みテーブルのタグの編集
- 設定済みテーブルの分析ルールの編集
- 設定済みテーブルの分析ルールの削除

設定済みテーブルの詳細の編集

コラボレーションメンバーは、設定済みテーブルの詳細を編集できます。

設定済みテーブルの詳細を編集するには

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. コンソールの左のナビゲーションペインで、[設定済みのテーブル] を選択します。
3. 自分が作成した設定済みテーブルを選択します。
4. 設定済みテーブルの詳細ページで、[設定済みテーブルの詳細] まで下方向にスクロールします。
5. [編集] を選択します。
6. 設定済みテーブルの [名前] または [説明] を変更します。
7. [変更を保存] を選択します。

設定済みテーブルのタグの編集

コラボレーションメンバーは、設定済みテーブルの作成後に、[設定済みのテーブル] タブで設定済みテーブルリソースのタグを管理できます。

設定済みテーブルのタグを編集するには

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. コンソールの左のナビゲーションペインで、[設定済みのテーブル] を選択します。
3. 自分が作成した設定済みテーブルを選択します。
4. 設定済みテーブルの詳細ページで、[タグ] セクションまで下方向にスクロールします。
5. [Manage tags (タグの管理)] を選択します。
6. [Manage tags] (タグの管理) ページで、次の操作を実行できます。
    - タグを削除するには、[削除] を選択します。

- タグを追加するには、[新しいタグの追加] を選択します。
- 変更を保存するには、変更の保存を選択します。

設定済みテーブルの分析ルールの編集

設定済みテーブルの分析ルールを編集するには

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. コンソールの左のナビゲーションペインで、[設定済みのテーブル] を選択します。
3. 自分が作成した設定済みテーブルを選択します。
4. 設定済みテーブルの詳細ページで、[集計分析ルール]、[分析ルールを一覧表示]、または [カスタム分析ルール] セクションまで下方向にスクロールします (どれを選択するかは、設定済みテーブルにどのタイプの分析ルールを選択したかによって異なります)。
5. [編集] を選択します。
6. [分析ルールを編集] ページで、次の操作を実行できます。
   - 次の方法で [分析ルール定義] を変更します。
     - JSON エディタを変更します。
     - [ファイルからインポート] を選択して、新しい分析ルール定義をアップロードします。
   - 以下のオプションから選択して、コラボレーションでメンバーに表示される内容をプレビューします。
     - テーブルビュー
     - JSON
     - クエリの例
7. [変更を保存] を選択して、変更を保存します。

設定済みテーブルの分析ルールの削除

> **Warning**
> このアクションは元に戻すことができず、関連するすべてのリソースに影響します。

設定済みテーブルの分析ルールを削除するには

1. にサインイン AWS Management Console し、 でAWS Clean Rooms コンソールを開きます AWS アカウント （ まだ開いていない場合 ）。
2. コンソールの左のナビゲーションペインで、[設定済みのテーブル] を選択します。
3. 自分が作成した設定済みテーブルを選択します。
4. 設定済みテーブルの詳細ページで、[集計分析ルール]、[分析ルールを一覧表示]、または [カスタム分析ルール] セクションまで下方向にスクロールします (どれを選択するかは、設定済みテーブルにどのタイプの分析ルールを選択したかによって異なります)。
5. [削除] を選択します。
6. 分析ルールを削除してよい場合は、[削除] を選択します。

トラブルシューティング AWS Clean Rooms

このセクションでは、 の使用時に発生する可能性のある一般的な問題 AWS Clean Rooms とその修正方法について説明します。

問題

クエリが参照する 1 つ以上のテーブルに、関連付けられたサービスロールでアクセスできない。テーブル/ロールの所有者が、サービスロールにテーブルへのアクセス許可を付与する必要がある。

- サービスロールのアクセス許可が必要に応じて設定されていることを確認します。詳細については、「セットアップ AWS Clean Rooms」を参照してください。

基になるデータセットの 1 つに、サポートされていないファイル形式が使用されている。

- データセットがサポートされているファイル形式のいずれかであることを確認します。
  - Parquet
  - RCFile
  - TextFile
  - SequenceFile
  - RegexSerde
  - OpenCSV
  - AVRO
  - JSON

詳細については、「のデータ形式 AWS Clean Rooms」を参照してください。

Cryptographic Computing for Clean Rooms の使用時に、期待どおりのクエリ結果が得られない。

Cryptographic Computing for Clean Rooms (C3R) を使用している場合は、暗号化された列がクエリで正しく使用されていることを確認します。

- sealed列は SELECT 句でのみ使用されます。
- fingerprint列は JOIN 句 (および特定の条件下では GROUP BY 句) でのみ使用されます。
- コラボレーション設定で、同じ名前のfingerprint列でしかJOINingを実行できないように指定されている場合があります。

詳細については、「暗号コンピューティング」および「the section called “列タイプ”」を参照してください。

のセキュリティ AWS Clean Rooms

のクラウドセキュリティが最優先事項 AWS です。 AWS のお客様は、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からメリットを得られます。

セキュリティは、 AWS とユーザーの間で共有される責任です。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 — クラウドで AWS サービスを実行するインフラストラクチャを保護する責任 AWS は AWS にあります。 AWS また、 では、安全に使用できるサービスも提供しています。コンプライアンスAWS プログラムコンプライアンスプログラム の一環として、サードパーティーの監査者は定期的にセキュリティの有効性をテストおよび検証。に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 Clean Rooms、「コンプライアンスプログラムによる AWS 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 AWS サービスによって決まります。また、お客様は、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、 の使用時に責任共有モデルを適用する方法を理解するのに役立ちます AWS Clean Rooms。セキュリティおよびコンプライアンスの目的 AWS Clean Rooms を達成するためにを設定する方法を示します。また、 リソースのモニタリングや保護に役立つ他の AWS のサービスの使用方法についても AWS Clean Rooms 説明します。

内容

でのデータ保護 AWS Clean Rooms

責任 AWS 共有モデル、 でのデータ保護に適用されます AWS Clean Rooms。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。お客様は、このインフラストラクチャでホストされてい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のよくある質問」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 は必須であり TLS 1.3 がお勧めです。
- 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。
- 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 などの高度なマネージド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 140-2 検証済みの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) 140-2」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は、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配置しないことを強くお勧めします。これは、コンソール、API、AWS Clean Rooms 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 名前に使用する自由記述のテキストフィールドやタグに入力したデータは、課金や診断ログに使用される場合があります。外部サーバーへの URL を提供する場合は、そのサーバーへのリクエストを検証するための認証情報を URL に含めないように強くお勧めします。

保管中の暗号化

AWS Clean Rooms は、追加の設定を必要とせずに、保管中のすべてのサービスメタデータを常に暗号化します。この暗号化は、 を使用する場合に自動的に行われます AWS Clean Rooms。

Clean Rooms ML は、保管中のサービスに保存されているすべてのデータを で暗号化します AWS KMS。独自の KMS キーを提供することを選択した場合、類似モデルと類似セグメント生成ジョブの内容は KMS キーで保存時に暗号化されます。

> **Note**
>
> Amazon S3 の暗号化オプションを使用して、保管中のデータを保護できます。詳細については、「Amazon S3 ユーザーガイド」の「Amazon S3 マネージドキーによるサーバー側の暗号化 (SSE-S3) の指定」を参照してください。

転送中の暗号化

AWS Clean Rooms は、転送中の暗号化に Transport Layer Security (TLS) とクライアント側の暗号化を使用します。との通信 AWS Clean Rooms は常に HTTPS 経由で行われるため、データは転送中に常に暗号化されます。これには、クリーンルーム ML を使用する際に転送中のすべてのデータが含まれます。

基になるデータの暗号化

基になるデータを暗号化する方法の詳細については、「Cryptographic Computing for Clean Rooms」を参照してください。

でのデータ保持 AWS Clean Rooms

類似モデルを作成すると、Clean Rooms ML はトレーニングデータを読み取り、ML モデルに適した形式に変換し、トレーニングされたモデルパラメータを Clean Rooms ML に保存します。Clean Rooms ML はトレーニングデータのコピーを保持しません。 AWS Clean Rooms SQL クエリは、クエリの実行後にデータを保持しません。次に、クリーンルーム ML はトレーニング済みモデルを使用して、すべてのユーザーの動作を要約します。Clean Rooms ML は、類似モデルがアクティブである限り、各ユーザーのユーザーレベルのデータセットをデータに保存します。

類似セグメント生成ジョブを開始すると、クリーンルーム ML はシードデータを読み取り、関連する類似モデルから動作概要を読み取り、 AWS Clean Rooms サービス内に保存される類似セグメントを作成します。Clean Rooms ML はシードデータのコピーを保持しません。Clean Rooms ML は、ジョブがアクティブである限り、ジョブのユーザーレベルの出力を保存します。

類似モデルまたは類似セグメント生成ジョブデータを削除する場合は、API を使用して削除します。Clean Rooms ML は、モデルまたはジョブに関連付けられているすべてのデータを非同期的に削

除します。このプロセスが完了すると、Clean Rooms ML はモデルまたはジョブのメタデータを削除し、API に表示されなくなります。Clean Rooms ML は、ディザスタリカバリの防止のために、削除されたデータを 3 日間保持します。ジョブまたはモデルが API に表示されなくなり、3 日が経過すると、モデルまたはジョブに関連するすべてのデータが完全に削除されます。

でのデータコラボレーションのベストプラクティス AWS Clean Rooms

このトピックでは、 AWS Clean Roomsでデータコラボレーションを行うときのベストプラクティスについて説明します。

AWS Clean Rooms は、コラボレーションで機密データを保護する機能を強化するために設定できる AWS 責任共有モデル AWS Clean Rooms .offers 分析ルールに従います。で設定した分析ルール AWS Clean Rooms は、設定した制限 (クエリコントロールとクエリ出力コントロール) を適用します。制限の決定とそれに応じた分析ルールの設定はユーザーの責任となります。

データコラボレーションには、 の使用だけでなく、 の使用も含まれます AWS Clean Rooms。データコラボレーションの利点を最大化するために、 および AWS Clean Rooms 特に分析ルールを使用して、次のベストプラクティスを実行することをお勧めします。

トピック

- でのベストプラクティス AWS Clean Rooms
- AWS Clean Roomsで分析ルールを使用する際のベストプラクティス

でのベストプラクティス AWS Clean Rooms

各データコラボレーションのリスクを評価し、それを外部および内部のコンプライアンスプログラムやポリシーなどのプライバシー要件と比較する責任はユーザーにあります。を使用して追加のアクションを実行することをお勧めします AWS Clean Rooms。これらの措置によって、リスクをさらに管理し、第三者がデータを再特定しようとする試み (差分攻撃やサイドチャネル攻撃など) を防止できる場合があります。

例えば、コラボレーションを開始する前に、相手のコラボレーターに対してデューデリジェンスを実施し、法的契約を結ぶことを検討してください。自社データの使用状況を監視するために、 AWS Clean Roomsの使用に際して他の監査メカニズムを採用することも検討してください。

AWS Clean Roomsで分析ルールを使用する際のベストプラクティス

の分析ルール AWS Clean Rooms を使用すると、設定済みテーブルにクエリコントロールを設定して、実行できるクエリを制限できます。例えば、設定済みテーブルを結合する方法や、選択できる列に関するクエリコントロールを設定できます。また、出力行の集約しきい値などのクエリ結果コントロールを設定して、クエリ出力を制限することもできます。クエリに含まれる設定済みテーブルでメンバーが設定した分析ルールを満たしていない行については、クエリが一切拒否され、結果から除外されます。

設定済みテーブルで分析ルールを使用する際には、次の 10 のベストプラクティスに従うことをお勧めします。

- クエリのユースケース (オーディエンスプランニングやアトリビューションなど) ごとに、個別の設定済みテーブルを作成します。同じ AWS Glue テーブルを基に、複数の設定済みテーブルを作成できます。
- コラボレーションでのクエリに必要な列 (ディメンション列、リスト列、結合列など) を分析ルールに指定します。これにより、差分攻撃や、他のメンバーによってデータのリバースエンジニアリングが行われるリスクを軽減できる可能性があります。許可リスト列の機能を使用して、今後クエリを実行できるようにしたい他の列を記録しておきます。特定のコラボレーションに使用できる列をカスタマイズするには、基盤となる同じテーブルを持つ追加の設定済み AWS Glue テーブルを作成します。
- コラボレーションでの分析に必要な関数を分析ルールに指定します。これにより、個々のデータポイントの情報が提示されるまれな関数エラーによるリスクを軽減できます。特定のコラボレーションに使用できる関数をカスタマイズするには、基になる同じ AWS Glue テーブルを使用して追加の設定済みテーブルを作成します。
- 行レベルの値が機密であるすべての列に集約制約を追加します。これには、設定済みテーブル内の列のうち、コラボレーションの他のメンバーのテーブル、および集約制約として分析ルールにも存在する列が含まれます。また、設定済みテーブル内のクエリを実行できない列、つまり、設定済みテーブルには存在するが分析ルールには存在しない列も含まれます。集約制約により、クエリ結果がコラボレーション外部のデータに関連付けられるリスクを軽減できます。
- テスト用のコラボレーションと分析ルールを作成して、指定された分析ルールで作成した制限をテストします。
- コラボレーターが設定したテーブルと、設定済みテーブルにあるメンバーの分析ルールを確認して、コラボレーションに際して合意された内容と一致していることを確認します。これにより、他のメンバーが独自のデータを設計して、合意されていないクエリを実行するリスクを軽減できます。

- 分析ルールの設定後に、設定済みテーブルで有効になっている、提示されているクエリの例 (コンソールのみ) を確認します。

> **Note**
>
> 提示されているクエリの例以外に、分析ルールや、他のコラボレーションメンバーのテーブルと分析ルールに基づいて、その他のクエリも利用できます。

- コラボレーションの設定済みテーブルの分析ルールは、追加または更新が可能です。追加や更新を行う際は、設定済みテーブルが関連付けられているすべてのコラボレーションと、その結果生じる影響を確認してください。これにより、すべてのコラボレーションで古い分析ルールが使用されることを回避できます。
- コラボレーションで実行されるクエリを確認して、そのクエリがコラボレーションに際して合意されたユースケースまたはクエリと一致していることを確認します ([クエリログ記録] 機能がオンになっている場合は、クエリがクエリログに表示されます)。これにより、合意されていない分析をメンバーが実行するリスクや、サイドチャネル攻撃などの潜在的な攻撃のリスクを軽減できます。
- コラボレーションメンバーの分析ルールやクエリで使用される設定済みテーブルの列を確認して、コラボレーションに際して合意された内容と一致していることを確認します (クエリのログ記録機能がオンになっている場合は、クエリがクエリログに表示されます)。これにより、他のメンバーが独自のデータを設計して、合意されていないクエリを実行するリスクを軽減できます。

の Identity and Access Management AWS Clean Rooms

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 Clean Rooms リソースの使用を承認する (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 AWS のサービス 使用できる です。

トピック

- AWS Clean Rooms ID とアクセスのトラブルシューティング
- サービス間の混乱した代理の防止
- AWS Clean Rooms ML の IAM 動作

対象者

AWS Identity and Access Management (IAM) の使用方法は、 で行う作業によって異なります AWS Clean Rooms。

サービスユーザー – AWS Clean Rooms サービスを使用してジョブを実行する場合、管理者から必要な認証情報とアクセス許可が与えられます。さらに多くの AWS Clean Rooms 機能を使用して作業を行う場合は、追加のアクセス許可が必要になる場合があります。アクセスの管理方法を理解すると、管理者から適切な権限をリクエストするのに役に立ちます。 AWS Clean Rooms機能にアクセスできない場合は、「AWS Clean Rooms ID とアクセスのトラブルシューティング」を参照してください。

サービス管理者 – 社内の AWS Clean Rooms リソースを担当している場合は、通常、 へのフルアクセスがあります AWS Clean Rooms。サービスユーザーがどの AWS Clean Rooms 機能やリソースにアクセスするかを決めるのは管理者の仕事です。その後、IAM 管理者にリクエストを送信して、サービスユーザーの権限を変更する必要があります。このページの情報を点検して、IAM の基本概念を理解してください。会社で IAM を で使用する方法の詳細については、 AWS Clean Rooms「」を参照してくださいと IAM の AWS Clean Rooms 連携方法。

IAM 管理者 - 管理者は、 AWS Clean Roomsへのアクセスを管理するポリシーの書き込み方法の詳細について確認する場合があります。IAM で使用できる AWS Clean Rooms 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 Clean Rooms。

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。として、IAM ユーザーとして AWS アカウントのルートユーザー、または IAM ロールを引き受けて認証（ にサインイン AWS) される必要があります。

ID ソースを介して提供された認証情報を使用して、フェデレーティッド ID AWS として にサインインできます。 AWS IAM Identity Center（ IAM Identity Center) ユーザーまたは会社のシングルサインオン認証は、フェデレーティッド ID の例です。フェデレーティッド ID としてサインインする場

合、IAM ロールを使用して、前もって管理者により ID フェデレーションが設定されています。フェデレーション AWS を使用して にアクセスすると、間接的にロールを引き受けることになります。

ユーザーのタイプに応じて、 AWS Management Console または AWS アクセスポータルにサインインできます。へのサインインの詳細については AWS、「 ユーザーガイド」の「 へのサインイン AWS アカウント方法AWS サインイン 」を参照してください。

AWS プログラムで にアクセスする場合、 は Software Development Kit (SDK) とコマンドラインインターフェイス (CLI) AWS を提供し、認証情報を使用してリクエストに暗号で署名します。 AWS ツールを使用しない場合は、リクエストに自分で署名する必要があります。推奨される方法を使用して自分でリクエストに署名する方法の詳細については、「AWS 全般のリファレンス」の「署名バージョン 4 の署名プロセス」を参照してください。

使用する認証方法を問わず、追加セキュリティ情報の提供をリクエストされる場合もあります。例えば、 AWS では、多要素認証 (MFA) を使用してアカウントのセキュリティを向上させることをお勧めします。詳細については、「AWS IAM Identity Center ユーザーガイド」の「多要素認証」および「IAM ユーザーガイド」の「AWSでの多要素認証 (MFA) の使用」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、アカウント内のすべての 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 1 つのサインインアイデンティティから始めます。この ID は AWS アカウント ルートユーザーと呼ばれ、アカウントの作成に使用した E メールアドレスとパスワードでサインインすることによってアクセスできます。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。ルートユーザーの認証情報を保護し、この認証情報はルートユーザーのみが実行できるタスクに使用します。ルートユーザーとしてサインインする必要があるタスクの完全なリストについては、「AWS 全般のリファレンス」の「AWS アカウントのルートユーザー の認証情報と IAM ID」を参照してください。

フェデレーティッドアイデンティティ

ベストプラクティスとして、管理者アクセスを必要とするユーザーを含む人間のユーザーに、一時的な認証情報を使用して にアクセスするための ID プロバイダーとのフェデレーションの使用を要求 AWS のサービス します。

フェデレーティッド ID は、エンタープライズユーザーディレクトリ、ウェブ ID プロバイダー、AWS Directory Service、アイデンティティセンターディレクトリ、または ID ソースを通じて提供された認証情報 AWS のサービス を使用して にアクセスするユーザーです。フェデレーティッド ID がにアクセスすると AWS アカウント、ロールを引き受け、ロールは一時的な認証情報を提供します。

アクセスを一元管理する場合は、 AWS IAM Identity Centerを使用することをお勧めします。IAM Identity Center でユーザーとグループを作成することも、独自の ID ソース内のユーザーとグループのセットに接続して同期して、すべての AWS アカウント とアプリケーションで使用することもできます。IAM Identity Center の詳細については、「AWS IAM Identity Center ユーザーガイド」の「IAM Identity Center 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、単一のユーザーまたはアプリケーションに対して特定のアクセス許可 AWS アカウント を持つ 内のアイデンティティです。可能であれば、パスワードやアクセスキーなどの長期的な認証情報を保有する IAM ユーザーを作成する代わりに、一時認証情報を使用することをお勧めします。ただし、IAM ユーザーでの長期的な認証情報が必要な特定のユースケースがある場合は、アクセスキーをローテーション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「長期的な認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションする」を参照してください。

IAM グループは、IAM ユーザーの集団を指定するアイデンティティです。グループとしてサインインすることはできません。グループを使用して、複数のユーザーに対して一度に権限を指定できます。多数のユーザーグループがある場合、グループを使用することで権限の管理が容易になります。例えば、IAMAdmins という名前のグループを設定して、そのグループに IAM リソースを管理する許可を与えることができます。

ユーザーは、ロールとは異なります。ユーザーは 1 人の人または 1 つのアプリケーションに一意に関連付けられますが、ロールはそれを必要とする任意の人が引き受けるようになっています。ユーザーには永続的な長期の認証情報がありますが、ロールでは一時的な認証情報が提供されます。詳細については、「IAM ユーザーガイド」の「IAM ユーザー (ロールではなく) の作成が適している場合」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 AWS アカウント を持つ 内のアイデンティティです。これは IAM ユーザーに似ていますが、特定のユーザーには関連付けられていません。ロール を切り替える AWS Management Console ことで、 で IAM ロールを一時的に引き受けることができます。ロールを引き受けるには、 または AWS API AWS CLI オペレーションを呼び出すか、カスタム URL を使用します。ロールを使用する方法の詳細については、「IAM ユーザーガイド」の「IAM ロールの使用」を参照してください。

IAM ロールと一時的な認証情報は、次の状況で役立ちます:

- フェデレーションユーザーアクセス – フェデレーティッド ID に許可を割り当てるには、ロールを作成してそのロールの許可を定義します。フェデレーティッド ID が認証されると、その ID はロールに関連付けられ、ロールで定義されている許可が付与されます。フェデレーションの詳細については、「IAM ユーザーガイド」の「Creating a role for a third-party Identity Provider」(サードパーティーアイデンティティプロバイダー向けロールの作成) を参照してください。IAM Identity Center を使用する場合は、許可セットを設定します。アイデンティティが認証後にアクセスできるものを制御するため、IAM Identity Center は、権限セットを IAM のロールに関連付けます。アクセス許可セットの詳細については、「AWS IAM Identity Center ユーザーガイド」の「アクセス許可セット」を参照してください。
- 一時的な IAM ユーザー権限 - IAM ユーザーまたはロールは、特定のタスクに対して複数の異なる権限を一時的に IAM ロールで引き受けることができます。
- クロスアカウントアクセス - IAM ロールを使用して、自分のアカウントのリソースにアクセスすることを、別のアカウントの人物 (信頼済みプリンシパル) に許可できます。クロスアカウントアクセス権を付与する主な方法は、ロールを使用することです。ただし、一部の では AWS のサービス、 (ロールをプロキシとして使用する代わりに) ポリシーをリソースに直接アタッチできます。クロスアカウントアクセスのロールとリソースベースのポリシーの違いについては、「IAM ユーザーガイド」の「IAM でのクロスアカウントリソースアクセス」を参照してください。
- クロスサービスアクセス — 一部の は、他の の機能 AWS のサービス を使用します AWS のサービス。例えば、あるサービスで呼び出しを行うと、通常そのサービスによって Amazon EC2 でアプリケーションが実行されたり、Amazon S3 にオブジェクトが保存されたりします。サービスでは、呼び出し元プリンシパルの許可、サービスロール、またはサービスリンクロールを使用してこれを行う場合があります。
  - 転送アクセスセッション (FAS) — IAM ユーザーまたはロールを使用して でアクションを実行する場合 AWS、ユーザーはプリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストリクエストリクエストと組み合わせて使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。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。
  - サービスロール - 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 のサービスにアクセス許可を委任するロールの作成」を参照してください。

- サービスにリンクされたロール – 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、サービスリンクロールの許可を表示できますが、編集することはできません。
- Amazon EC2 で実行されているアプリケーション – IAM ロールを使用して、EC2 インスタンスで実行され、 AWS CLI または AWS API リクエストを行うアプリケーションの一時的な認証情報を管理できます。これは、EC2 インスタンス内でのアクセスキーの保存に推奨されます。 AWS ロールを EC2 インスタンスに割り当て、そのすべてのアプリケーションで使用できるようにするには、インスタンスにアタッチされたインスタンスプロファイルを作成します。インスタンスプロファイルにはロールが含まれ、EC2 インスタンスで実行されるプログラムは一時的な認証情報を取得できます。詳細については、「IAM ユーザーガイド」の「Amazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用して許可を付与する」を参照してください。

IAM ロールと IAM ユーザーのどちらを使用するかについては、「IAM ユーザーガイド」の「(IAM ユーザーではなく) IAM ロールをいつ作成したら良いのか?」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、 AWS ID 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ているときにアクセス許可を定義する のオブジェクトです。 は、プリンシパル (ユーザー、ルートユーザー、またはロールセッション) AWS 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。ポリシーでの権限により、リクエストが許可されるか拒否されるかが決まります。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構造と内容の詳細については、「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 AWS JSON 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。

すべての IAM エンティティ (ユーザーまたはロール) は、許可のない状態からスタートします。デフォルトでは、ユーザーは何もできず、自分のパスワードを変更することすらできません。何かを実行する許可をユーザーに付与するには、管理者がユーザーに許可ポリシーをアタッチする必要があります。また、管理者は、必要な許可があるグループにユーザーを追加できます。管理者がグループに許可を付与すると、そのグループ内のすべてのユーザーにこれらの許可が付与されます。

IAM ポリシーは、オペレーションの実行方法を問わず、アクションの許可を定義します。例えば、`iam:GetRole` アクションを許可するポリシーがあるとします。そのポリシーを持つユーザーは、 AWS Management Console、、 AWS CLIまたは AWS API からロール情報を取得できます。

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ポリシーは、IAM ユーザー、ユーザーのグループ、ロールなど、アイデンティティにアタッチできる JSON 権限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 ポリシーの作成」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、さらにインラインポリシーまたはマネージドポリシーに分類できます。インラインポリシーは、単一のユーザー、グループ、またはロールに直接埋め込まれます。管理ポリシーは、 AWS アカウント内の複数のユーザー、グループ、およびロールにアタッチできるスタンドアロンポリシーです。管理ポリシーには、 AWS 管理ポリシーとカスタマー管理ポリシーが含まれます。マネージドポリシーまたは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「マネージドポリシーとインラインポリシーの比較」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。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制御できます。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は、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。

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、一般的ではない追加のポリシータイプをサポートします。これらのポリシータイプでは、より一般的なポリシータイプで付与された最大の権限を設定できます。

- アクセス許可の境界 - アクセス許可の境界は、アイデンティティベースのポリシーによって IAM エンティティ (IAM ユーザーまたはロール) に付与できる権限の上限を設定する高度な機能です。エンティティに許可の境界を設定できます。結果として許可される範囲は、エンティティのアイデンティティベースポリシーとその許可の境界の共通部分になります。`Principal` フィールドでユーザーまたはロールを指定するリソースベースのポリシーでは、許可の境界は制限されません。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。アクセス許可の境界の詳細については、「IAM ユーザーガイド」の「IAM エンティティのアクセス許可の境界」を参照してください。
- サービスコントロールポリシー (SCPs) – SCPs は、 の組織または組織単位 (OU) に対する最大アクセス許可を指定する JSON ポリシーです AWS Organizations。 AWS Organizations は、 AWS アカウント ビジネスが所有する複数の をグループ化して一元管理するためのサービスです。組織内のすべての機能を有効にすると、サービスコントロールポリシー (SCP) を一部またはすべてのアカウントに適用できます。SCP は、各 を含むメンバーアカウントのエンティティのアクセス許可を制限します AWS アカウントのルートユーザー。Organizations と SCP の詳細については、AWS Organizations ユーザーガイドの「SCP の仕組み」を参照してください。
- セッションポリシー - セッションポリシーは、ロールまたはフェデレーションユーザーの一時的なセッションをプログラムで作成する際にパラメータとして渡す高度なポリシーです。結果としてセッションの権限は、ユーザーまたはロールのアイデンティティベースポリシーとセッションポリシーの共通部分になります。また、リソースベースのポリシーから権限が派生する場合もあります。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。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権限を理解するのがさらに難しくなります。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。

と IAM の AWS Clean Rooms 連携方法

IAM を使用して へのアクセスを管理する前に AWS Clean Rooms、 で使用できる IAM 機能について学びます AWS Clean Rooms。

で使用できる IAM の機能 AWS Clean Rooms

IAM 機能	AWS Clean Rooms サポート
アイデンティティベースのポリシー	Yes
リソースベースのポリシー	部分的
ポリシーアクション	Yes
ポリシーリソース	はい
ポリシー条件キー (サポート固有)	部分的
ACL	No
ABAC (ポリシー内のタグ)	はい
一時的な認証情報	はい
転送アクセスセッション (FAS)	はい
サービスロール	あり
サービスリンクロール	いいえ

AWS Clean Rooms およびその他の がほとんどの IAM 機能と AWS のサービス 連携する方法の概要を把握するには、IAM ユーザーガイドのAWS のサービス「IAM と連携する」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 Clean Rooms

アイデンティティベースポリシーをサポートする	Yes

アイデンティティベースポリシーは、IAM ユーザー、ユーザーのグループ、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 ポリシーの作成」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。プリンシパルは、それが添付されているユーザーまたはロールに適用されるため、アイデンティティベースのポリシーでは指定できません。JSON ポリシーで使用できるすべての要素については、「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 Clean Rooms

AWS Clean Rooms 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 Clean Rooms。

内のリソースベースのポリシー AWS Clean Rooms

リソースベースのポリシーのサポート	部分的

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制御できます。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は、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、アカウント全体、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。リソースベースのポリシーにクロスアカウントのプリンシパルを追加しても、信頼関係は半分しか確立されない点に注意してください。プリンシパルとリソースが異なる にある場合 AWS アカウント、信頼されたアカウントの IAM 管理者は、プリンシパルエンティティ (ユーザーまたはロール) にリソースへのアクセス許可も付与する必要があります。IAM 管理者は、アイデンティティベースのポリシーをエンティティにアタッチすることで権限を付与します。ただし、リソースベースのポリシーで、同じアカウントのプリンシパルへのアクセス権が付与されている場合は、アイデンティティベースのポリシーをさらに付与する必要はありません。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。

この AWS Clean Rooms サービスは、設定された類似モデル管理リソースポリシー と呼ばれるリソースベースのポリシーのタイプを 1 つだけサポートします。これは、設定された類似モデルにアタッチされます。このポリシーは、設定された類似モデルに対してアクションを実行できるプリンシパルを定義します。

リソースベースのポリシーを設定済みの類似モデルにアタッチする方法については、「」を参照してくださいAWS Clean Rooms ML の IAM 動作。

のポリシーアクション AWS Clean Rooms

ポリシーアクションに対するサポート	はい

管理者は AWS JSON 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んなリソースにどんな条件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。ポリシーアクションの名前は通常、関連付けられた AWS API オペレーションと同じです。一致する API オペレーションのない許可のみのアクションなど、いくつかの例外があります。また、ポリシーに複数のアクションが必要なオペレーションもあります。これらの追加アクションは、依存アクションと呼ばれます。

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権限を付与するポリシーで使用されます。

AWS Clean Rooms アクションのリストを確認するには、「サービス認証リファレンス」の「 で定義されるアクション AWS Clean Rooms」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 Clean Rooms を使用します。

```
cleanrooms
```

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": [
      "cleanrooms:action1",
      "cleanrooms:action2"
         ]
```

AWS Clean Rooms 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 Clean Rooms。

のポリシーリソース AWS Clean Rooms

ポリシーリソースに対するサポート	はい

管理者は AWS JSON 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どのような条件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。ステートメントには、Resource または NotResource要素を含める必要があります。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。これは、リソースレベルの許可と呼ばれる特定のリソースタイプをサポートするアクションに対して実行できます。

オペレーションのリスト化など、リソースレベルの権限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 Clean Rooms リソースタイプとその ARNs」の「 で定義されるリソース AWS Clean Rooms」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 Clean Roomsで定義されるアクション」を参照してください。

AWS Clean Rooms 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 Clean Rooms。

のポリシー条件キー AWS Clean Rooms

サービス固有のポリシー条件キーのサポート	部分的

管理者は AWS JSON 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んなリソースにどんな条件でアクションを実行できるかということです。

Condition 要素 (または Condition ブロック) を使用すると、ステートメントが有効な条件を指定できます。Condition 要素はオプションです。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することで、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。

1 つのステートメントに複数の Condition 要素を指定するか、1 つの Condition 要素に複数のキーを指定すると、 AWS は AND 論理演算子を使用してそれらを評価します。1 つの条件キーに複数の値を指定すると、 は論理ORオペレーションを使用して条件 AWS を評価します。ステートメントの権限が付与される前にすべての条件が満たされる必要があります。

条件を指定する際にプレースホルダー変数も使用できます。例えば IAM ユーザーに、IAM ユーザー名がタグ付けされている場合のみリソースにアクセスできる権限を付与することができます。詳細については、「IAM ユーザーガイド」の「IAM ポリシーの要素: 変数およびタグ」を参照してください。

AWS は、グローバル条件キーとサービス固有の条件キーをサポートします。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 Clean Rooms ML がポリシー条件キーを使用する方法については、「」を参照してくださいAWS Clean Rooms ML の IAM 動作。

ACLs AWS Clean Rooms

ACL のサポート	No

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許可を持つかをコントロールします。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。

での ABAC AWS Clean Rooms

ABAC のサポート (ポリシー内のタグ)	はい

属性ベースのアクセス制御 (ABAC) は、属性に基づいてアクセス許可を定義するアクセス許可戦略です。では AWS、これらの属性はタグ と呼ばれます。タグは、IAM エンティティ (ユーザーまた

はロール) および多くの AWS リソースにアタッチできます。エンティティとリソースのタグ付けは、ABAC の最初の手順です。その後、プリンシパルのタグがアクセスしようとしているリソースのタグと一致した場合にオペレーションを許可するように ABAC ポリシーをします。

ABAC は、急成長する環境やポリシー管理が煩雑になる状況で役立ちます。

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はいです。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 とは?」を参照してください。ABAC をセットアップするステップ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制御 (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 Clean Rooms

一時的な認証情報のサポート	はい

一部の AWS のサービス は、一時的な認証情報を使用してサインインすると機能しません。一時的な認証情報 AWS のサービス を使用する などの詳細については、IAM ユーザーガイドのAWS のサービス 「IAM と連携する 」を参照してください。

ユーザー名とパスワード以外の AWS Management Console 方法で にサインインする場合、一時的な認証情報を使用します。例えば、会社の Single Sign-On (SSO) リンク AWS を使用して にアクセスすると、そのプロセスによって一時的な認証情報が自動的に作成されます。また、ユーザーとしてコンソールにサインインしてからロールを切り替える場合も、一時的な認証情報が自動的に作成されます。ロールの切り替えに関する詳細については、「IAM ユーザーガイド」の「ロールへの切り替え (コンソール)」を参照してください。

一時的な認証情報は、 AWS CLI または AWS API を使用して手動で作成できます。その後、これらの一時的な認証情報を使用して . AWS recommends にアクセスできます AWS。これは、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することを推奨しています。詳細については、「IAM の一時的セキュリティ認証情報」を参照してください。

の転送アクセスセッション AWS Clean Rooms

転送アクセスセッション (FAS) をサポート	はい

IAM ユーザーまたはロールを使用して でアクションを実行すると AWS、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストのリクエストと組み合わせて使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。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。

AWS Clean Roomsのサービスロール

サービスロールに対するサポート	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 のサービスにアクセス許可を委任するロールの作成」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 Clean Rooms 機能が破損する可能性があります。が指示する場合以外 AWS Clean Rooms は、サービスロールを編集しないでください。

のサービスにリンクされたロール AWS Clean Rooms

サービスにリンクされたロールのサポート	いいえ

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権限を表示できますが、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。表の中から、Service-linked role (サービスにリンクされたロール) 列に Yes と記載されたサービスを見つけます。サービスリンク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 Clean Rooms

デフォルトでは、 ユーザーおよびロールには、 AWS Clean Rooms リソースを作成または変更する権限はありません。また、、 AWS Command Line Interface （ AWS CLI ） AWS Management Console、または AWS API を使用してタスクを実行することはできません。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。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き受けることができます。

これらサンプルの 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 ポリシーの作成」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 Clean Rooms、 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、「サービス認証リファレンス」の「 のアクション、リソース、および条件キー AWS Clean Rooms」を参照してください。 ARNs

トピック

- ポリシーのベストプラクティス
- AWS Clean Rooms コンソールを使用する
- 自分の権限の表示をユーザーに許可する

ポリシーのベストプラクティス

ID ベースのポリシーは、ユーザーのアカウントで誰かが AWS Clean Rooms リソースを作成、アクセス、または削除できるどうかを決定します。これらのアクションを実行すると、 AWS アカウントに料金が発生する可能性があります。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を付与するには、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して、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「AWS ジョブ機能の管理ポリシー」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。例えば、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、 などの特定の を介してサービスアクションが使用される場合に 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 AWS CloudFormation。詳細については、「IAM ユーザーガイド」の [IAM JSON policy elements: Condition] (IAM JSON ポリシー要素: 条件) を参照してください。
- IAM Access Analyzer 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 Access Analyzer 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 ポリシーの検証」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 保護 API アクセスの設定」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 Clean Rooms コンソールを使用する

AWS Clean Rooms 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 Clean Rooms 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 Clean Rooms 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 Clean Rooms *FullAccess*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 マネージドポリシー AWS Clean Rooms

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、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るため、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。ユースケース別にカスタマーマネージドポリシーを定義して、マネージドポリシー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 管理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) が更新されます。 は、新しい AWS のサービス が起動されたとき、または既存のサービスで新しい API AWS オペレーションが使用可能になったときに、 AWS 管理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 管理ポリシー」を参照してください。

AWS 管理ポリシー: **AWSCleanRoomsReadOnlyAccess**

`AWSCleanRoomsReadOnlyAccess` をIAM プリンシパルにアタッチできます。

このポリシーは、`AWSCleanRoomsReadOnlyAccess` コラボレーションのリソースとメタデータへの読み取り専用のアクセス許可を付与します。

許可の詳細

このポリシーには、以下の許可が含まれています。

- CleanRoomsRead - プリンシパルにサービスへの読み取り専用アクセスを許可します。
- ConsoleDisplayTables – プリンシパルが、基盤となる AWS Glue テーブルに関するデータをコンソールに表示するために必要な AWS Glue メタデータに読み取り専用でアクセスできるようにします。
- ConsoleLogSummaryQueryLogs - プリンシパルにクエリログの閲覧を許可します。
- ConsoleLogSummaryObtainLogs - プリンシパルにログ結果の取得を許可します。

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "CleanRoomsRead",
   "Effect": "Allow",
   "Action": [
    "cleanrooms:BatchGet*",
    "cleanrooms:Get*",
    "cleanrooms:List*"
   ],
   "Resource": "*"
  },
  {
   "Sid": "ConsoleDisplayTables",
   "Effect": "Allow",
   "Action": [
    "glue:GetDatabase",
    "glue:GetDatabases",
    "glue:GetTable",
    "glue:GetTables",
    "glue:GetPartition",
    "glue:GetPartitions",
    "glue:GetSchema",
    "glue:GetSchemaVersion",
    "glue:BatchGetPartition"
   ],
   "Resource": "*"
  },
  {
   "Sid": "ConsoleLogSummaryQueryLogs",
   "Effect": "Allow",
   "Action": [
    "logs:StartQuery"
```

```
    ],
    "Resource": "arn:aws:logs:*:*:log-group:/aws/cleanrooms*"
   },
   {
    "Sid": "ConsoleLogSummaryObtainLogs",
    "Effect": "Allow",
    "Action": [
     "logs:GetQueryResults"
    ],
    "Resource": "*"
   }
  ]
}
```

AWS 管理ポリシー: **AWSCleanRoomsFullAccess**

AWSCleanRoomsFullAccess をIAM プリンシパルにアタッチできます。

このポリシーは、 AWS Clean Rooms コラボレーション内のリソースとメタデータへのフルアクセス (読み取り、書き込み、更新) を許可する管理アクセス許可を付与します。このポリシーには、クエリを実行するためのアクセス許可が含まれています。

許可の詳細

このポリシーには、以下の許可が含まれています。

- CleanRoomsAccess – のすべてのリソースに対するすべてのアクションへのフルアクセスを許可します AWS Clean Rooms。
- PassServiceRole - 名前に「cleanrooms」が含まれるサービスのみにサービスロールを渡すためのアクセス許可 (PassedToService 条件) を付与します。
- ListRolesToPickServiceRole — プリンシパルが を使用するときにサービスロールを選択できるように、すべてのロールを一覧表示できるようにします AWS Clean Rooms。
- GetRoleAndListRolePoliciesToInspectServiceRole - IAM のサービスロールと対応するポリシーを表示することをプリンシパルに許可します。
- ListPoliciesToInspectServiceRolePolicy - IAM のサービスロールと対応するポリシーを表示することをプリンシパルに許可します。
- GetPolicyToInspectServiceRolePolicy - IAM のサービスロールと対応するポリシーを表示することをプリンシパルに許可します。

- ConsoleDisplayTables – プリンシパルが、基盤となる AWS Glue テーブルに関するデータをコンソールに表示するために必要な AWS Glue メタデータに読み取り専用でアクセスできるようにします。
- ConsolePickQueryResultsBucketListAll - 使用可能なすべての Amazon S3 バケットのリストから、クエリ結果を書き込む S3 バケットを選択することをプリンシパルに許可します。
- SetQueryResultsBucket – クエリ結果を書き込む S3 バケットを選択することをプリンシパルに許可します。
- ConsoleDisplayQueryResults – S3 バケットから読み取ったクエリ結果を顧客に示すことをプリンシパルに許可します。
- WriteQueryResults – クエリ結果を顧客所有の S3 バケットに書き込むことをプリンシパルに許可します。
- EstablishLogDeliveries — プリンシパルがクエリログを顧客の Amazon CloudWatch Logs ロググループに配信できるようにします。
- SetupLogGroupsDescribe — プリンシパルが Amazon CloudWatch Logs ロググループの作成プロセスを使用できるようにします。
- SetupLogGroupsCreate — プリンシパルが Amazon CloudWatch Logs ロググループを作成できるようにします。
- SetupLogGroupsResourcePolicy — プリンシパルが Amazon CloudWatch Logs ロググループにリソースポリシーを設定できるようにします。
- ConsoleLogSummaryQueryLogs - プリンシパルにクエリログの閲覧を許可します。
- ConsoleLogSummaryObtainLogs - プリンシパルにログ結果の取得を許可します。

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "CleanRoomsAccess",
   "Effect": "Allow",
   "Action": [
    "cleanrooms:*"
   ],
   "Resource": "*"
  },
  {
   "Sid": "PassServiceRole",
   "Effect": "Allow",
```

```
    "Action": [
     "iam:PassRole"
    ],
    "Resource": "arn:aws:iam::*:role/service-role/*cleanrooms*",
    "Condition": {
     "StringEquals": {
      "iam:PassedToService": "cleanrooms.amazonaws.com"
     }
    }
   },
   {
    "Sid": "ListRolesToPickServiceRole",
    "Effect": "Allow",
    "Action": [
     "iam:ListRoles"
    ],
    "Resource": "*"
   },
   {
    "Sid": "GetRoleAndListRolePoliciesToInspectServiceRole",
    "Effect": "Allow",
    "Action": [
     "iam:GetRole",
     "iam:ListRolePolicies",
     "iam:ListAttachedRolePolicies"
    ],
    "Resource": "arn:aws:iam::*:role/service-role/*cleanrooms*"
   },
   {
    "Sid": "ListPoliciesToInspectServiceRolePolicy",
    "Effect": "Allow",
    "Action": [
     "iam:ListPolicies"
    ],
    "Resource": "*"
   },
   {
    "Sid": "GetPolicyToInspectServiceRolePolicy",
    "Effect": "Allow",
    "Action": [
     "iam:GetPolicy",
     "iam:GetPolicyVersion"
    ],
    "Resource": "arn:aws:iam::*:policy/*cleanrooms*"
```

```
  },
  {
   "Sid": "ConsoleDisplayTables",
   "Effect": "Allow",
   "Action": [
    "glue:GetDatabase",
    "glue:GetDatabases",
    "glue:GetTable",
    "glue:GetTables",
    "glue:GetPartition",
    "glue:GetPartitions",
    "glue:GetSchema",
    "glue:GetSchemaVersion",
    "glue:BatchGetPartition"
   ],
   "Resource": "*"
  },
  {
   "Sid": "ConsolePickQueryResultsBucketListAll",
   "Effect": "Allow",
   "Action": [
    "s3:ListAllMyBuckets"
   ],
   "Resource": "*"
  },
  {
   "Sid": "SetQueryResultsBucket",
   "Effect": "Allow",
   "Action": [
    "s3:GetBucketLocation",
    "s3:ListBucketVersions"
   ],
   "Resource": "arn:aws:s3:::cleanrooms-queryresults*"
  },
  {
   "Sid": "WriteQueryResults",
   "Effect": "Allow",
   "Action": [
    "s3:ListBucket",
    "s3:PutObject"
   ],
   "Resource": "arn:aws:s3:::cleanrooms-queryresults*",
   "Condition": {
    "ForAnyValue:StringEquals": {
```

```
      "aws:CalledVia": "cleanrooms.amazonaws.com"
     }
    }
   },
   {
    "Sid": "ConsoleDisplayQueryResults",
    "Effect": "Allow",
    "Action": [
     "s3:GetObject"
    ],
    "Resource": "arn:aws:s3:::cleanrooms-queryresults*"
   },
   {
    "Sid": "EstablishLogDeliveries",
    "Effect": "Allow",
    "Action": [
     "logs:CreateLogDelivery",
     "logs:GetLogDelivery",
     "logs:UpdateLogDelivery",
     "logs:DeleteLogDelivery",
     "logs:ListLogDeliveries"
    ],
    "Resource": "*",
    "Condition": {
     "ForAnyValue:StringEquals": {
      "aws:CalledVia": "cleanrooms.amazonaws.com"
     }
    }
   },
   {
    "Sid": "SetupLogGroupsDescribe",
    "Effect": "Allow",
    "Action": [
     "logs:DescribeLogGroups"
    ],
    "Resource": "*",
    "Condition": {
     "ForAnyValue:StringEquals": {
      "aws:CalledVia": "cleanrooms.amazonaws.com"
     }
    }
   },
   {
    "Sid": "SetupLogGroupsCreate",
```

```
    "Effect": "Allow",
    "Action": [
     "logs:CreateLogGroup"
    ],
    "Resource": "arn:aws:logs:*:*:log-group:/aws/cleanrooms*",
    "Condition": {
     "ForAnyValue:StringEquals": {
      "aws:CalledVia": "cleanrooms.amazonaws.com"
     }
    }
   },
   {
    "Sid": "SetupLogGroupsResourcePolicy",
    "Effect": "Allow",
    "Action": [
      "logs:DescribeResourcePolicies",
     "logs:PutResourcePolicy"
    ],
    "Resource": "*",
    "Condition": {
     "ForAnyValue:StringEquals": {
      "aws:CalledVia": "cleanrooms.amazonaws.com"
     }
    }
   },
   {
    "Sid": "ConsoleLogSummaryQueryLogs",
    "Effect": "Allow",
    "Action": [
     "logs:StartQuery"
    ],
    "Resource": "arn:aws:logs:*:*:log-group:/aws/cleanrooms*"
   },
   {
    "Sid": "ConsoleLogSummaryObtainLogs",
    "Effect": "Allow",
    "Action": [
     "logs:GetQueryResults"
    ],
    "Resource": "*"
   }
  ]
}
```

AWS 管理ポリシー: **AWSCleanRoomsFullAccessNoQuerying**

IAM principals に AWSCleanRoomsFullAccessNoQuerying をアタッチできます。

このポリシーは、 AWS Clean Rooms コラボレーション内のリソースとメタデータへのフルアクセス (読み取り、書き込み、更新) を許可する管理アクセス許可を付与します。このポリシーでは、クエリを実行するためのアクセス許可は除外されます。

許可の詳細

このポリシーには、以下の許可が含まれています。

- CleanRoomsAccess – コラボレーションでのクエリを除く AWS Clean Rooms、 のすべてのリソースに対するすべてのアクションへのフルアクセスを許可します。
- CleanRoomsNoQuerying – StartProtectedQuery と UpdateProtectedQuery を明示的に拒否し、クエリを禁止します。
- PassServiceRole - 名前に「cleanrooms」が含まれるサービスのみにサービスロールを渡すためのアクセス許可 (PassedToService 条件) を付与します。
- ListRolesToPickServiceRole — プリンシパルが を使用するときにサービスロールを選択できるように、すべてのロールを一覧表示できるようにします AWS Clean Rooms。
- GetRoleAndListRolePoliciesToInspectServiceRole - IAM のサービスロールと対応するポリシーを表示することをプリンシパルに許可します。
- ListPoliciesToInspectServiceRolePolicy - IAM のサービスロールと対応するポリシーを表示することをプリンシパルに許可します。
- GetPolicyToInspectServiceRolePolicy - IAM のサービスロールと対応するポリシーを表示することをプリンシパルに許可します。
- ConsoleDisplayTables – プリンシパルが、基盤となる AWS Glue テーブルに関するデータをコンソールに表示するために必要な AWS Glue メタデータに読み取り専用でアクセスできるようにします。
- EstablishLogDeliveries — プリンシパルがクエリログを顧客の Amazon CloudWatch Logs ロググループに配信できるようにします。
- SetupLogGroupsDescribe — プリンシパルが Amazon CloudWatch Logs ロググループの作成プロセスを使用できるようにします。
- SetupLogGroupsCreate — プリンシパルが Amazon CloudWatch Logs ロググループを作成できるようにします。

- SetupLogGroupsResourcePolicy — プリンシパルが Amazon CloudWatch Logs ロググループにリソースポリシーを設定できるようにします。
- ConsoleLogSummaryQueryLogs - プリンシパルにクエリログの閲覧を許可します。
- ConsoleLogSummaryObtainLogs - プリンシパルにログ結果の取得を許可します。
- cleanrooms – サービス内のコラボレーション、分析テンプレート、設定済みテーブル、メンバーシップ、関連リソースを管理します AWS Clean Rooms 。これらのリソースに関する情報の作成、更新、削除、一覧表示、取得など、さまざまなオペレーションを実行します。
- iam – cleanrooms「」を含む名前のサービスロールを AWS Clean Rooms サービスに渡します。ロール、ポリシーを一覧表示し、サービスに関連する AWS Clean Rooms サービスロールとポリシーを検査します。
- glue – からデータベース、テーブル、パーティション、スキーマに関する情報を取得します AWS Glue。これは、 AWS Clean Rooms サービスが基盤となるデータソースを表示して操作するために必要です。
- logs – CloudWatch Logs のログ配信、ロググループ、およびリソースポリシーを管理します。AWS Clean Rooms サービスに関連するログをクエリして取得します。これらのアクセス許可は、サービス内のモニタリング、監査、トラブルシューティングの目的で必要です。

また、このポリシーは、アクション cleanrooms:StartProtectedQueryおよび を明示的に拒否cleanrooms:UpdateProtectedQueryし、ユーザーが保護されたクエリを直接実行または更新できないようにします。これは、制御された AWS Clean Rooms メカニズムを通じて実行する必要があります。

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "CleanRoomsAccess",
   "Effect": "Allow",
   "Action": [
    "cleanrooms:BatchGetCollaborationAnalysisTemplate",
    "cleanrooms:BatchGetSchema",
    "cleanrooms:BatchGetSchemaAnalysisRule",
    "cleanrooms:CreateAnalysisTemplate",
    "cleanrooms:CreateCollaboration",
    "cleanrooms:CreateConfiguredTable",
    "cleanrooms:CreateConfiguredTableAnalysisRule",
    "cleanrooms:CreateConfiguredTableAssociation",
    "cleanrooms:CreateMembership",
```

```
    "cleanrooms:DeleteAnalysisTemplate",
    "cleanrooms:DeleteCollaboration",
    "cleanrooms:DeleteConfiguredTable",
    "cleanrooms:DeleteConfiguredTableAnalysisRule",
    "cleanrooms:DeleteConfiguredTableAssociation",
    "cleanrooms:DeleteMember",
    "cleanrooms:DeleteMembership",
    "cleanrooms:GetAnalysisTemplate",
    "cleanrooms:GetCollaboration",
    "cleanrooms:GetCollaborationAnalysisTemplate",
    "cleanrooms:GetConfiguredTable",
    "cleanrooms:GetConfiguredTableAnalysisRule",
    "cleanrooms:GetConfiguredTableAssociation",
    "cleanrooms:GetMembership",
    "cleanrooms:GetProtectedQuery",
    "cleanrooms:GetSchema",
    "cleanrooms:GetSchemaAnalysisRule",
    "cleanrooms:ListAnalysisTemplates",
    "cleanrooms:ListCollaborationAnalysisTemplates",
    "cleanrooms:ListCollaborations",
    "cleanrooms:ListConfiguredTableAssociations",
    "cleanrooms:ListConfiguredTables",
    "cleanrooms:ListMembers",
    "cleanrooms:ListMemberships",
    "cleanrooms:ListProtectedQueries",
    "cleanrooms:ListSchemas",
    "cleanrooms:UpdateAnalysisTemplate",
    "cleanrooms:UpdateCollaboration",
    "cleanrooms:UpdateConfiguredTable",
    "cleanrooms:UpdateConfiguredTableAnalysisRule",
    "cleanrooms:UpdateConfiguredTableAssociation",
    "cleanrooms:UpdateMembership",
    "cleanrooms:ListTagsForResource",
    "cleanrooms:UntagResource",
    "cleanrooms:TagResource"
   ],
   "Resource": "*"
  },
  {
   "Sid": "CleanRoomsNoQuerying",
   "Effect": "Deny",
   "Action": [
    "cleanrooms:StartProtectedQuery",
    "cleanrooms:UpdateProtectedQuery"
```

```
    ],
    "Resource": "*"
   },
   {
    "Sid": "PassServiceRole",
    "Effect": "Allow",
    "Action": [
     "iam:PassRole"
    ],
    "Resource": "arn:aws:iam::*:role/service-role/*cleanrooms*",
    "Condition": {
     "StringEquals": {
      "iam:PassedToService": "cleanrooms.amazonaws.com"
     }
    }
   },
   {
    "Sid": "ListRolesToPickServiceRole",
    "Effect": "Allow",
    "Action": [
     "iam:ListRoles"
    ],
    "Resource": "*"
   },
   {
    "Sid": "GetRoleAndListRolePoliciesToInspectServiceRole",
    "Effect": "Allow",
    "Action": [
     "iam:GetRole",
     "iam:ListRolePolicies",
     "iam:ListAttachedRolePolicies"
    ],
    "Resource": "arn:aws:iam::*:role/service-role/*cleanrooms*"
   },
   {
    "Sid": "ListPoliciesToInspectServiceRolePolicy",
    "Effect": "Allow",
    "Action": [
     "iam:ListPolicies"
    ],
    "Resource": "*"
   },
   {
    "Sid": "GetPolicyToInspectServiceRolePolicy",
```

```
    "Effect": "Allow",
    "Action": [
     "iam:GetPolicy",
     "iam:GetPolicyVersion"
    ],
    "Resource": "arn:aws:iam::*:policy/*cleanrooms*"
   },
   {
    "Sid": "ConsoleDisplayTables",
    "Effect": "Allow",
    "Action": [
     "glue:GetDatabase",
     "glue:GetDatabases",
     "glue:GetTable",
     "glue:GetTables",
     "glue:GetPartition",
     "glue:GetPartitions",
     "glue:GetSchema",
     "glue:GetSchemaVersion",
     "glue:BatchGetPartition"
    ],
    "Resource": "*"
   },
   {
    "Sid": "EstablishLogDeliveries",
    "Effect": "Allow",
    "Action": [
     "logs:CreateLogDelivery",
     "logs:GetLogDelivery",
     "logs:UpdateLogDelivery",
     "logs:DeleteLogDelivery",
     "logs:ListLogDeliveries"
    ],
    "Resource": "*",
    "Condition": {
     "ForAnyValue:StringEquals": {
      "aws:CalledVia": "cleanrooms.amazonaws.com"
     }
    }
   },
   {
    "Sid": "SetupLogGroupsDescribe",
    "Effect": "Allow",
    "Action": [
```

```
    "logs:DescribeLogGroups"
   ],
   "Resource": "*",
   "Condition": {
    "ForAnyValue:StringEquals": {
     "aws:CalledVia": "cleanrooms.amazonaws.com"
    }
   }
  },
  {
   "Sid": "SetupLogGroupsCreate",
   "Effect": "Allow",
   "Action": [
    "logs:CreateLogGroup"
   ],
   "Resource": "arn:aws:logs:*:*:log-group:/aws/cleanrooms*",
   "Condition": {
    "ForAnyValue:StringEquals": {
     "aws:CalledVia": "cleanrooms.amazonaws.com"
    }
   }
  },
  {
   "Sid": "SetupLogGroupsResourcePolicy",
   "Effect": "Allow",
   "Action": [
     "logs:DescribeResourcePolicies",
    "logs:PutResourcePolicy"
   ],
   "Resource": "*",
   "Condition": {
    "ForAnyValue:StringEquals": {
     "aws:CalledVia": "cleanrooms.amazonaws.com"
    }
   }
  },
  {
   "Sid": "ConsoleLogSummaryQueryLogs",
   "Effect": "Allow",
   "Action": [
    "logs:StartQuery"
   ],
   "Resource": "arn:aws:logs:*:*:log-group:/aws/cleanrooms*"
  },
```

```
  {
   "Sid": "ConsoleLogSummaryObtainLogs",
   "Effect": "Allow",
   "Action": [
    "logs:GetQueryResults"
   ],
   "Resource": "*"
  }
 ]
}
```

AWS マネージドポリシー: **AWSCleanRoomsMLReadOnlyAccess**

AWSCleanRoomsMLReadOnlyAccess をIAM プリンシパルにアタッチできます。

このポリシーは、AWSCleanRoomsMLReadOnlyAccess コラボレーションのリソースとメタデータへの読み取り専用のアクセス許可を付与します。

このポリシーには、以下の権限が含まれています。

- CleanRoomsConsoleNavigation – AWS Clean Rooms コンソールの画面を表示するアクセス許可を付与します。
- CleanRoomsMLRead – プリンシパルに Clean Rooms ML サービスへの読み取り専用アクセスを許可します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CleanRoomsConsoleNavigation",
            "Effect": "Allow",
            "Action": [
                "cleanrooms:GetCollaboration",
                "cleanrooms:GetConfiguredAudienceModelAssociation",
                "cleanrooms:GetMembership",
                "cleanrooms:ListAnalysisTemplates",
                "cleanrooms:ListCollaborationAnalysisTemplates",
                "cleanrooms:ListCollaborationConfiguredAudienceModelAssociations",
                "cleanrooms:ListCollaborations",
                "cleanrooms:ListConfiguredTableAssociations",
                "cleanrooms:ListConfiguredTables",
```

```
                "cleanrooms:ListMembers",
                "cleanrooms:ListMemberships",
                "cleanrooms:ListProtectedQueries",
                "cleanrooms:ListSchemas",
                "cleanrooms:ListTagsForResource"
            ],
            "Resource": "*"
        },
        {
            "Sid": "CleanRoomsMLRead",
            "Effect": "Allow",
            "Action": [
                "cleanrooms-ml:Get*",
                "cleanrooms-ml:List*"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS 管理ポリシー: **AWSCleanRoomsMLFullAccess**

AWSCleanRoomsMLFullAcces をIAM プリンシパルにアタッチできます。このポリシーは、Clean Rooms ML に必要なリソースとメタデータへのフルアクセス (読み取り、書き込み、更新) を許可する管理アクセス許可を付与します。

許可の詳細

このポリシーには、以下の許可が含まれています。

- CleanRoomsMLFullAccess – すべての Clean Rooms ML アクションへのアクセスを許可します。
- PassServiceRole - 名前に「cleanrooms-ml」が含まれるサービスのみにサービスロールを渡すためのアクセス許可 (PassedToService 条件) を付与します。
- CleanRoomsConsoleNavigation – AWS Clean Rooms コンソールの画面を表示するアクセス許可を付与します。
- CollaborationMembershipCheck – コラボレーション内でオーディエンス生成 (類似セグメント) ジョブを開始すると、クリーンルーム ML サービスは ListMembersを呼び出して、コラボレーションが有効で、発信者がアクティブなメンバーであり、設定されたオーディエンスモデル所有者がアクティブなメンバーであることを確認します。このアクセス許可は常に必要です。コンソールナビゲーション SID はコンソールユーザーにのみ必要です。

- AssociateModels – プリンシパルが Clean Rooms ML モデルをコラボレーションに関連付けることを許可します。
- TagAssociations – 類似モデルとコラボレーションの関連付けにタグを追加することをプリンシパルに許可します。
- ListRolesToPickServiceRole — プリンシパルが を使用するときにサービスロールを選択できるように、すべてのロールを一覧表示できるようにします AWS Clean Rooms。
- GetRoleAndListRolePoliciesToInspectServiceRole - IAM のサービスロールと対応するポリシーを表示することをプリンシパルに許可します。
- ListPoliciesToInspectServiceRolePolicy - IAM のサービスロールと対応するポリシーを表示することをプリンシパルに許可します。
- GetPolicyToInspectServiceRolePolicy - IAM のサービスロールと対応するポリシーを表示することをプリンシパルに許可します。
- ConsoleDisplayTables – プリンシパルが、基盤となる AWS Glue テーブルに関するデータをコンソールに表示するために必要な AWS Glue メタデータに読み取り専用でアクセスできるようにします。
- ConsolePickOutputBucket – 設定済みのオーディエンスモデル出力用の Amazon S3 バケットを選択することをプリンシパルに許可します。
- ConsolePickS3Location – 設定済みのオーディエンスモデル出力のバケット内の場所を選択することをプリンシパルに許可します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CleanRoomsMLFullAccess",
            "Effect": "Allow",
            "Action": [
                "cleanrooms-ml:*"
            ],
            "Resource": "*"
        },
        {
            "Sid": "PassServiceRole",
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
```

```
            "Resource": [
                "arn:aws:iam::*:role/cleanrooms-ml*"
            ],
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "cleanrooms-ml.amazonaws.com"
                }
            }
        },
        {
            "Sid": "CleanRoomsConsoleNavigation",
            "Effect": "Allow",
            "Action": [
                "cleanrooms:GetCollaboration",
                "cleanrooms:GetConfiguredAudienceModelAssociation",
                "cleanrooms:GetMembership",
                "cleanrooms:ListAnalysisTemplates",
                "cleanrooms:ListCollaborationAnalysisTemplates",
                "cleanrooms:ListCollaborationConfiguredAudienceModelAssociations",
                "cleanrooms:ListCollaborations",
                "cleanrooms:ListConfiguredTableAssociations",
                "cleanrooms:ListConfiguredTables",
                "cleanrooms:ListMembers",
                "cleanrooms:ListMemberships",
                "cleanrooms:ListProtectedQueries",
                "cleanrooms:ListSchemas",
                "cleanrooms:ListTagsForResource"
            ],
            "Resource": "*"
        },
        {
            "Sid": "CollaborationMembershipCheck",
            "Effect": "Allow",
            "Action": [
                "cleanrooms:ListMembers"
            ],
            "Resource": "*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "aws:CalledVia": ["cleanrooms-ml.amazonaws.com"]
                }
            }
        },
        {
```

```
            "Sid": "AssociateModels",
            "Effect": "Allow",
            "Action": [
                "cleanrooms:CreateConfiguredAudienceModelAssociation"
            ],
            "Resource": "*"
        },
        {
            "Sid": "TagAssociations",
            "Effect": "Allow",
            "Action": [
                "cleanrooms:TagResource"
            ],
            "Resource": "arn:aws:cleanrooms:*:*:membership/*/
configuredaudiencemodelassociation/*"
        },
        {
            "Sid": "ListRolesToPickServiceRole",
            "Effect": "Allow",
            "Action": [
                "iam:ListRoles"
            ],
            "Resource": "*"
        },
        {
            "Sid": "GetRoleAndListRolePoliciesToInspectServiceRole",
            "Effect": "Allow",
            "Action": [
                "iam:GetRole",
                "iam:ListRolePolicies",
                "iam:ListAttachedRolePolicies"
            ],
            "Resource": [
                "arn:aws:iam::*:role/service-role/cleanrooms-ml*",
                "arn:aws:iam::*:role/role/cleanrooms-ml*"
            ]
        },
        {
            "Sid": "ListPoliciesToInspectServiceRolePolicy",
            "Effect": "Allow",
            "Action": [
                "iam:ListPolicies"
            ],
            "Resource": "*"
```

```
        },
        {
            "Sid": "GetPolicyToInspectServiceRolePolicy",
            "Effect": "Allow",
            "Action": [
                "iam:GetPolicy",
                "iam:GetPolicyVersion"
            ],
            "Resource": "arn:aws:iam::*:policy/*cleanroomsml*"
        },
        {
            "Sid": "ConsoleDisplayTables",
            "Effect": "Allow",
            "Action": [
                "glue:GetDatabase",
                "glue:GetDatabases",
                "glue:GetTable",
                "glue:GetTables",
                "glue:GetPartition",
                "glue:GetPartitions",
                "glue:GetSchema",
                "glue:GetSchemaVersion",
                "glue:BatchGetPartition"
            ],
            "Resource": "*"
        },
        {
            "Sid": "ConsolePickOutputBucket",
            "Effect": "Allow",
            "Action": [
                "s3:ListAllMyBuckets"
            ],
            "Resource": "*"
        },
        {
            "Sid": "ConsolePickS3Location",
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Resource": "arn:aws:s3:::*cleanrooms-ml*"
        }
    ]
```

```
}
```

AWS Clean RoomsAWS 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した AWS Clean Rooms 以降の の AWS マネージド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通知を受け取るには、 AWS Clean Rooms 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
AWSCleanRoomsFullAccessNoQuerying – 既存のポリシーの更新	cleanrooms:BatchGetSchemaAnalysisRuleがCleanRoomsAccessに追加されました。	2024 年 5 月 13 日
AWSCleanRoomsFullAccess – 既存のポリシーの更新	コンソールの有無にかかわらずクエリ結果バケットを設定するためにアクセス許可が必要なため、このポリシーの SetQueryResultsBucketのステートメント ID を AWSCleanRoomsFullAccess から ConsolePickQueryResultsBucket に更新して、アクセス許可をより適切に表現しました。	2024 年 3 月 21 日
AWSCleanRoomsMLReadOnlyAccess - 新しいポリシー AWSCleanRoomsMLFullAccess - 新しいポリシー	AWS Clean Rooms ML をサポートするために AWSCleanRoomsMLReadOnlyAccess と AWSCleanRoomsMLFullAccess が追加されました。	2023 年 11 月 29 日
AWSCleanRoomsFullAccessNoQuerying – 既存のポリシーの更新	cleanrooms:CreateAnalysisTemplate、、cleanrooms:GetAnalysisTemplatecleanrooms:UpdateAnalysisTemplate、、 cleanrooms:DeleteAnalysisTemplate、cleanrooms:ListAnalysisTemplates、 をcleanrooms:GetCollaborationAnalysisTemplatecleanrooms:ListCollaborationAnalysisTemplatesCleanRoomsA	2023 年 7 月 31 日

変更	説明	日付
	ccessに追加してcleanrooms:BatchGetCollaborationAnalysisTemplate、新しい分析テンプレート機能を有効にしました。	
AWSCleanRoomsFullAccessNoQuerying – 既存のポリシーの更新	cleanrooms:ListTagsForResource、cleanrooms:UntagResource、および cleanrooms:TagResource が CleanRoomsAccess に追加され、リソースのタグ付けが可能になりました。	2023 年 3 月 21 日
AWS Clean Rooms が変更の追跡を開始しました	AWS Clean Rooms が AWS マネージドポリシーの変更の追跡を開始しました。	2023 年 1 月 12 日

AWS Clean Rooms ID とアクセスのトラブルシューティング

次の情報は、 と 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 Clean Rooms 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ない AWS Clean Rooms
- iam を実行する権限がありません。PassRole
- 自分の 以外のユーザーに自分の AWS Clean Rooms リソース AWS アカウント へのアクセスを許可したい

でアクションを実行する権限がない AWS Clean Rooms

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、そのアクションを実行できるようにポリシーを更新する必要があります。

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 cleanrooms:*GetWidget* 権限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:
 cleanrooms:GetWidget on resource: my-example-widget
```

この場合、Mateo のポリシーでは、*my-example-widget* アクションを使用して cleanrooms:*GetWidget* リソースにアクセスすることを許可するように更新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。サインイン認証情報を提供した担当者が管理者です。

iam を実行する権限がありません。PassRole

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 Clean Rooms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、そのサービスに既存のロールを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 Clean Rooms でアクションを実行しよう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。メアリーには、ロールをサービスに渡す許可がありません。

```
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。サインイン認証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 Clean Rooms 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。

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 Clean Rooms 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と IAM の AWS Clean Rooms 連携方法。
- 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認証されたユーザーへのアクセスの提供 (ID フェデレーション) を参照してください。
- クロスアカウントアクセスでのロールとリソースベースのポリシーの使用の違いの詳細については、「IAM ユーザーガイド」の「IAM ロールとリソースベースのポリシーとの相違点」を参照してください。

サービス間の混乱した代理の防止

混乱した代理問題は、アクションを実行するためのアクセス許可を持たないエンティティが、より特権のあるエンティティにアクションの実行を強制できてしまう場合に生じる、セキュリティ上の問題です。では AWS、サービス間のなりすましにより、混乱した代理問題が発生する可能性があります。サービス間でのなりすましは、1 つのサービス (呼び出し元サービス) が、別のサービス (呼び出し対象サービス) を呼び出すときに発生する可能性があります。呼び出し元サービスは、本来ならアクセスすることが許可されるべきではない方法でその許可を使用して、別のお客様のリソースに対する処理を実行するように操作される場合があります。これを防ぐために、 AWS には、アカウント内のリソースへのアクセス権が付与されたサービスプリンシパルですべてのサービスのデータを保護するために役立つツールが用意されています。

リソースポリシーで `aws:SourceArn` のグローバル条件コンテキストキーを使用して、 AWSClean Rooms が別のサービスに付与するアクセス許可をそのリソースに制限することをお勧めします。クロスサービスのアクセスにリソースを 1 つだけ関連付けたい場合は、`aws:SourceArn` を使用します。

混乱した代理問題から保護するための最も効果的な方法は、リソースの完全な ARN を指定して `aws:SourceArn` グローバル条件コンテキストキーを使用することです。では AWSClean Rooms、`sts:ExternalId` 条件キーと比較する必要があります。

`aws:SourceArn` の値は、引き受けられたロールのメンバーシップの ARN に設定する必要があります。

次の例では、 AWSClean Rooms で aws:SourceArn グローバル条件コンテキストキーを使用して、混乱した代理問題を回避する方法を示します。

> **Note**
>
> このサンプルポリシーは、 AWSClean Rooms が顧客データへのアクセスに使用するサービスロールの信頼ポリシーに適用されます。
> *MembershipID* の値は、コラボレーションにおける AWSClean Rooms メンバーシップ ID です。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowIfExternalIdMatches",
            "Effect": "Allow",
            "Principal": {
                "Service": "cleanrooms.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringLike": {
                    "sts:ExternalId": "arn:aws:*:aws-region:*:dbuser:*/membershipID*"
                }
            }
        },
        {
            "Sid": "AllowIfSourceArnMatches",
            "Effect": "Allow",
            "Principal": {
                "Service": "cleanrooms.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "ForAnyValue:ArnEquals": {
                    "aws:SourceArn": "arn:aws:cleanrooms:aws-
region:123456789012:membership/membershipID"
                }
            }
        }
    ]
```

```
}
```

AWS Clean Rooms ML の IAM 動作

クロスアカウントジョブ

Clean Rooms ML では、ある によって作成された特定のリソース AWS アカウント に、別の によって自分のアカウントで安全にアクセスできます AWS アカウント。A のクライアントが AWS アカウント B AWS アカウント が所有するConfiguredAudienceModelリソースStartAudienceGenerationJobで を呼び出すと、クリーンルーム ML はジョブに 2 つの ARNs を作成します。1 つの ARN は A に、もう 1 AWS アカウント つの ARN は AWS アカウント B にあります。ARNsは、 を除いて同じです AWS アカウント。

Clean Rooms ML は、両方のアカウントがジョブに独自の ARNs を作成します。例えば、両方のアカウントでタグベースのアクセスコントロールを使用し、 AWS 組織からのポリシーを適用できます。ジョブは両方のアカウントのデータを処理するため、どちらのアカウントでもジョブとそれに関連するデータを削除できます。どちらのアカウントも、もう一方のアカウントによるジョブの削除をブロックすることはできません。

ジョブの実行は 1 つだけで、どちらのアカウントも ListAudienceGenerationJobs を呼び出してジョブを確認できます。どちらのアカウントもGet、独自の AWS アカウント ID を持つ ARN を使用して、ジョブで Delete、、および Export APIs を呼び出すことができます。

他の AWS アカウント ID で ARN を使用する場合、 はジョブにアクセス AWS アカウント できません。

ジョブの名前は AWS アカウント内で一意である必要があります。 AWS アカウント B の名前は *$accountA -$name* です。 AWS アカウント A で選択された名前は、ジョブが AWS アカウント B で表示されるときに A のプレフィックスが付けられ AWS アカウント ます。

クロスアカウントStartAudienceGenerationJobを成功させるには、 AWS アカウント B で次の例のようなリソースポリシーを使用して、 AWS アカウント B の新しいジョブと AWS アカウント B ConfiguredAudienceModelの の両方でそのアクションを許可する必要があります。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Clean-Rooms-<CAMA ID>",
```

```
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "accountA"
                ]
            },
            "Action": [
                "cleanrooms-ml:StartAudienceGenerationJob"
            ],
            "Resource": [
                "arn:aws:cleanrooms-ml:us-west-1:AccountB:configured-audience-
model/id",
                "arn:aws:cleanrooms-ml:us-west-1:AccountB:audience-generation-job/*"
            ],
            // optional - always set by AWS Clean Rooms
"Condition":{"StringEquals":{"cleanrooms-ml:CollaborationId":"UUID"}}
        }
    ]
}
```

AWS Clean Rooms ML API を使用して、 を true manageResourcePoliciesに設定して設定された類似モデルを作成する場合、 はこのポリシー AWS Clean Rooms を作成します。

さらに、A の発信者の ID AWS アカウント ポリシーには、 に対する アクセスStartAudienceGenerationJob許可が必要ですarn:aws:cleanrooms-ml:us-west-1:AccountA:audience-generation-job/*。そのため、アクション には、A ジョブ、AWS アカウント B ジョブ、 AWS アカウント B StartAudienceGenerationJob AWS アカウント の 3 つの IAM リソースがありますConfiguredAudienceModel。

> **Warning**
>
> ジョブ AWS アカウント を開始した は、ジョブに関する AWS CloudTrail 監査ログイベントを受け取ります。ConfiguredAudienceModel を所有する AWS CloudTrail は AWS アカウント 監査ログイベントを受信しません。

ジョブのタグ付け

CreateConfiguredAudienceModel の childResourceTagOnCreatePolicy=FROM_PARENT_RESOURCE パラメータを設定すると、設定した類似モデルから作成されたアカウント内のすべての類似セグメント生成ジョブには、設定した類

似モデルと同じタグがデフォルトで割り当てられます。設定した類似モデルが親で、類似セグメント生成ジョブが子です。

自身のアカウント内でジョブを作成する場合、ジョブのリクエストタグは親タグよりも優先されます。他のアカウントが作成したジョブが、自身のアカウントにタグを作成することはありません。childResourceTagOnCreatePolicy=FROM_PARENT_RESOURCE を設定し、別のアカウントでジョブを作成した場合、そのジョブのコピーは 2 つあります。アカウントのコピーには親リソースタグが付けられ、ジョブ送信者のアカウントのコピーにはリクエストのタグが付けられます。

共同作業者の検証

AWS Clean Rooms コラボレーションの他のメンバーにアクセス許可を付与する場合、リソースポリシーには条件キー を含める必要がありますcleanrooms-ml:CollaborationId。これにより、collaborationIdパラメータがStartAudienceGenerationJobリクエストに含まれていることが強制されます。collaborationId パラメータがリクエストに含まれると、Clean Rooms ML はコラボレーションが存在すること、ジョブ送信者がコラボレーションのアクティブなメンバーであること、設定された類似モデル所有者がコラボレーションのアクティブなメンバーであることを検証します。

が設定された類似モデルリソースポリシー AWS Clean Rooms を管理する場合 (manageResourcePoliciesパラメータはCreateConfiguredAudienceModelAssociation リクエスト TRUE にあります)、この条件キーはリソースポリシーで設定されます。したがって、collaborationIdで を指定する必要がありますStartAudienceGenerationJob。

クロスアカウントアクセス

アカウント間で呼び出せるのは StartAudienceGenerationJob のみです。他のすべての Clean Rooms ML APIsは、自分のアカウントのリソースでのみ使用できます。これにより、トレーニングデータ、類似モデルの設定、その他の情報は非公開のままになります。

Clean Rooms ML は、アカウント間で Amazon S3 または AWS Glue ロケーションを公開することはありません。トレーニングデータの場所、設定済みの類似モデルの出力場所、および類似セグメント生成ジョブシードの場所は、どのアカウントでも表示されません。別のアカウントが送信したオーディエンス生成ジョブを Get した場合、サービスにはシードロケーションは表示されません。

のコンプライアンス検証 AWS Clean Rooms

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、コンプライアンスプログラムAWS のサービス による対象範囲内のコンプライアンスプログラムを参照

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、「 でのレポートのダウンロード AWS Artifact」の」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。 は、コンプライアンスに役立つ以下のリソース AWS を提供しています。

- セキュリティとコンプライアンスのクイックスタートガイド – これらのデプロイガイドでは、アーキテクチャ上の考慮事項について説明し、セキュリティとコンプライアンスに重点を置いたベースライン環境 AWS を にデプロイする手順について説明します。
- アマゾン ウェブ サービスにおける HIPAA セキュリティとコンプライアンスのアーキテクチャ — このホワイトペーパーでは、企業が AWS を使用して HIPAA 対象アプリケーションを作成する方法について説明します。

> **Note**
>
> すべて AWS のサービス HIPAA の対象となるわけではありません。詳細については、「HIPAA 対応サービスのリファレンス」を参照してください。

- AWS コンプライアンスリソース – このワークブックとガイドのコレクションは、お客様の業界や地域に適用される場合があります。
- AWS カスタマーコンプライアンスガイド — コンプライアンスの観点から責任共有モデルを理解します。このガイドでは、ガイダンスを保護し AWS のサービス 、複数のフレームワーク (米国国立標準技術研究所 (NIST)、Payment Card Industry Security Standards Council (PCI)、国際標準化機構 (ISO) を含む) のセキュリティコントロールにマッピングするためのベストプラクティスをまとめています。
- 「 デベロッパーガイド」の「 ルールによるリソースの評価」 – この AWS Config サービスは、リソース設定が社内プラクティス、業界ガイドライン、および規制にどの程度準拠しているかを評価します。 AWS Config
- AWS Security Hub – これにより AWS のサービス 、 内のセキュリティ状態を包括的に確認できます AWS。Security Hub では、セキュリティコントロールを使用して AWS リソースを評価し、セキュリティ業界標準とベストプラクティスに対するコンプライアンスをチェックします。サポートされているサービスとコントロールのリストについては、「Security Hub のコントロールリファレンス」を参照してください。

- Amazon GuardDuty – これにより AWS アカウント、疑わしいアクティビティや悪意のあるアクティビティがないか環境を監視することで、、、ワークロード、コンテナ、データに対する潜在的な脅威 AWS のサービス を検出します。 GuardDuty は、特定のコンプライアンスフレームワークで義務付けられている侵入検知要件を満たすことで、PCI DSS などのさまざまなコンプライアンス要件への対応に役立ちます。
- AWS Audit Manager – これにより AWS のサービス 、 AWS 使用状況を継続的に監査し、リスクの管理方法と規制や業界標準への準拠を簡素化できます。

の耐障害性 AWS Clean Rooms

AWS グローバルインフラストラクチャは、 AWS リージョンとアベイラビリティーゾーンを中心に構築されています。リージョンには、低レイテンシー、高いスループット、そして高度の冗長ネットワークで接続されている複数の物理的に独立および隔離されたアベイラビリティーゾーンがあります。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 リージョンと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。

のインフラストラクチャセキュリティ AWS Clean Rooms

マネージドサービスである AWS Clean Rooms は、 AWS グローバルネットワークセキュリティで保護されています。 AWS セキュリティサービスと がインフラストラクチャ AWS を保護する方法については、AWS 「 クラウドセキュリティ」を参照してください。インフラストラクチャセキュリティのベストプラクティスを使用して AWS 環境を設計するには、「 Security Pillar AWS Well-Architected Framework」の「Infrastructure Protection」を参照してください。

が AWS 公開している API コールを使用して、ネットワーク AWS Clean Rooms 経由で にアクセスします。クライアントは以下をサポートする必要があります:

- Transport Layer Security (TLS)。TLS 1.2 は必須で TLS 1.3 がお勧めです。
- DHE (楕円ディフィー・ヘルマン鍵共有) や ECDHE (楕円曲線ディフィー・ヘルマン鍵共有) などの完全前方秘匿性 (PFS) による暗号スイート。これらのモードは、Java 7 以降など、ほとんどの最新システムでサポートされています。

また、リクエストには、アクセスキー ID と、IAM プリンシパルに関連付けられているシークレットアクセスキーを使用して署名する必要があります。または、AWS Security Token Service (AWS STS) を使用して、一時的なセキュリティ認証情報を生成し、リクエストに署名することもできます。

ネットワークセキュリティ

クエリの実行中に が S3 バケットから AWS Clean Rooms 読み取ると、 と Amazon S3 間の AWS Clean Rooms トラフィックは AWS プライベートネットワークを介して安全にルーティングされます。処理中のトラフィックは Amazon Signature Version 4 プロトコル (SIGv4) を使用して署名され、HTTPS を使用して暗号化されます。このトラフィックは、設定済みテーブルに設定した IAM サービスロールに基づいて承認されます。

エンドポイント AWS Clean Rooms を介してプログラムで に接続できます。サービスエンドポイントのリストについては、「AWS 全般のリファレンス」の「AWS Clean Rooms のエンドポイントとクォータ」を参照してください。

すべてのサービスエンドポイントは HTTPS 限定です。Amazon Virtual Private Cloud (VPC) エンドポイントは、VPC AWS Clean Rooms から に接続し、インターネット接続を希望しない場合に使用できます。詳細については、「 AWS PrivateLink ガイド」の「 を通じて AWS のサービスにアクセスする AWS PrivateLink」を参照してください。

aws:SourceVpce context キーを使用する IAM ポリシーを IAM プリンシパルに割り当てることで、IAM プリンシパルが VPC エンドポイント AWS Clean Rooms 経由でのみ を呼び出すことができるように制限し、インターネット経由で呼び出すことはできません。

インターフェイスエンドポイント (AWS PrivateLink) を使用して AWS Clean Rooms または AWS Clean Rooms ML にアクセスする

AWS PrivateLink を使用して、Virtual Private Cloud (VPC) と AWS Clean Rooms または AWS Clean Rooms ML の間にプライベート接続を作成できます。インターネットゲートウェイ AWS Clean Rooms 、NAT デバイス、VPN 接続、または AWS Direct Connect 接続を使用せずに、VPC 内にあるかのように または AWS Clean Rooms ML にアクセスできます。VPC のインスタンスは、パブリック IP アドレスがなくても AWS Clean Roomsにアクセスできます。

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 Clean Rooms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「AWS のサービス でアクセスする」を参照してください。

に関する考慮事項 AWS Clean Rooms

のインターフェイスエンドポイントを設定する前に AWS Clean Rooms、「 AWS PrivateLink ガイド」の「考慮事項」を参照してください。

AWS Clean Rooms および AWS Clean Rooms ML 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は、 AWS Clean Rooms または AWS Clean Rooms ML ではサポートされていません。デフォルトでは、インターフェイスエンドポイントを介して AWS Clean Rooms および AWS Clean Rooms ML へのフル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を介した AWS Clean Rooms または AWS Clean Rooms ML 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 Clean Rooms

Amazon VPC コンソール AWS Clean Rooms または AWS Command Line Interface () を使用して、または AWS Clean Rooms ML 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 Clean Rooms 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.cleanrooms
```

次のサービス名を使用して、 AWS Clean Rooms ML 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.cleanrooms-ml
```

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 Clean Rooms への API リクエストを実行できます。例えば、`cleanrooms-ml.us-east-1.amazonaws.com`です。

モニタリング AWS Clean Rooms

モニタリングは、 AWS Clean Rooms およびその他の AWS ソリューションの信頼性、可用性、およびパフォーマンスを維持する上で重要な部分です。 は、 をモニタリングし AWS Clean Rooms、問題が発生したときに報告し、必要に応じて自動アクションを実行するために、以下のモニタリングツール AWS を提供します。

- Amazon CloudWatch Logs を使用すると、Amazon EC2 インスタンスやその他のソースからログファイルをモニタリング、保存 AWS CloudTrail、アクセスできます。Amazon CloudWatch Logs は、ログファイル内の情報をモニタリングし、特定のしきい値に達し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。

  Clean Rooms ML では、特定の API アクションに対してクロスアカウントジョブを使用できます。ジョブ AWS アカウント を開始した は、ジョブの AWS CloudTrail 監査ログイベントを受け取ります。詳細については、「AWS Clean Rooms ML の IAM 動作」を参照してください。
- 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。を呼び出したユーザーとアカウント AWS、呼び出し元のソース IP アドレス、呼び出しが発生した日時を特定できます。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。

AWS Clean Roomsを使用したAWS CloudTrailAPI コールのログ記録

AWS Clean Rooms は、ユーザー、ロール、または AWS のサービスが AWS Clean Rooms で実行したアクションを記録するサービスである AWS CloudTrail と統合されています。CloudTrail は、AWS Clean Rooms 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 Clean Rooms コンソールの呼び出しと、AWS Clean Rooms 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、AWS Clean Rooms のイベントなど、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 証跡を設定しない場合でも、CloudTrail コンソールの [Event history (イベント履歴)] で最新のイベントを表示できます。CloudTrail によって収集された情報を使用して、AWS Clean Rooms に対して行われた要求、要求が行われた IP アドレス、要求を行った人、要求が行われた日時、および追加の詳細を判別できます。

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。

CloudTrail での AWS Clean Rooms 情報

CloudTrail は、アカウント作成時に AWS アカウント で有効になります。AWS Clean Rooms でアクティビティが発生すると、そのアクティビティは [イベント履歴] のその他の AWS のサービスイベントと共に 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。

AWS Clean Rooms 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、追跡を作成します。追跡により、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 ログで収集されたイベントデータをより詳細に分析し、それを基にアクションを取るために他の AWS のサービス を設定できます。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の CloudTrail ログファイルの受信
- 複数のアカウントからの CloudTrail ログファイルの受信

すべての AWS Clean Rooms アクションは CloudTrail によってログに記録され、AWS Clean RoomsAPI リファレンスに記録されます。

各イベントまたはログエントリには、リクエストの生成者に関する情報が含まれます。同一性情報は次の判断に役立ちます。

- リクエストが、ルートユーザーまたは IAM ユーザーのどちらの認証情報を使用して送信されたかどうか。
- 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一時的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 のサービス によって送信されたかどうか。

詳細については、「CloudTrail userIdentity 要素」を参照してください。

AWS Clean Rooms ログファイルエントリの理解

追跡は、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 のログファイルには、単一か複数のログエントリがあります。イベントはあらゆるソース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、リクエストのパラメータなどの情報が含まれます。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。

AWS Clean Rooms の CloudTrail イベントの例

次の例は、以下の CloudTrail イベントを示しています。

トピック

- StartProtectedQuery (成功)
- StartProtectedQuery (失敗)

StartProtectedQuery (成功)


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "EXAMPLE_PRINCIPAL_ID",
        "arn": "arn:aws:sts::123456789012:assumed-role/query-runner/jdoe",
        "accountId": "123456789012",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "EXAMPLE_PRINCIPAL_ID",
                "arn": "arn:aws:iam::123456789012:role/query-runner",
                "accountId": "123456789012",
                "userName": "query-runner"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-04-07T19:34:32Z",
                "mfaAuthenticated": "false"
            }
        }
    },
```

```
    "eventTime": "2023-04-07T19:53:32Z",
    "eventSource": "cleanrooms.amazonaws.com",
    "eventName": "StartProtectedQuery",
    "awsRegion": "us-east-2",
    "sourceIPAddress": "203.0.113.1",
    "userAgent": "aws-internal/3",
    "requestParameters": {
        "resultConfiguration": {
            "outputConfiguration": {
                "s3": {
                    "resultFormat": "CSV",
                    "bucket": "cleanrooms-queryresults-jdoe-test",
                    "keyPrefix": "test"
                }
            }
        },
        "sqlParameters": "***",
        "membershipIdentifier": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "type": "SQL"
    },
    "responseElements": {
        "Access-Control-Expose-Headers": "x-amzn-RequestId,x-amzn-ErrorType,x-amzn-
ErrorMessage,Date",
        "protectedQuery": {
            "createTime": 1680897212.279,
            "id": "f5988bf1-771a-4141-82a8-26fcc4e41c9f",
            "membershipArn": "arn:aws:cleanrooms:us-east-2:123456789012:membership/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "membershipId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "resultConfiguration": {
                "outputConfiguration": {
                    "s3": {
                        "bucket": "cleanrooms-queryresults-jdoe-test",
                        "keyPrefix": "test",
                        "resultFormat": "CSV"
                    }
                }
            },
            "sqlParameters": "***",
            "status": "SUBMITTED"
        }
    },
    "requestID": "7464211b-2277-4b55-9723-fb4f259aefd2",
    "eventID": "f7610f5e-74b9-420f-ae43-206571ebcbf7",
```

```
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

StartProtectedQuery (失敗)

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "EXAMPLE_PRINCIPAL_ID",
        "arn": "arn:aws:sts::123456789012:assumed-role/query-runner/jdoe",
        "accountId": "123456789012",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "EXAMPLE_PRINCIPAL_ID",
                "arn": "arn:aws:iam::123456789012:role/query-runner",
                "accountId": "123456789012",
                "userName": "query-runner"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-04-07T19:34:32Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-04-07T19:47:27Z",
    "eventSource": "cleanrooms.amazonaws.com",
    "eventName": "StartProtectedQuery",
    "awsRegion": "us-east-2",
    "sourceIPAddress": "203.0.113.1",
    "userAgent": "aws-internal/3",
    "errorCode": "ValidationException",
    "requestParameters": {
        "resultConfiguration": {
            "outputConfiguration": {
                "s3": {
```

```
                    "resultFormat": "CSV",
                    "bucket": "cleanrooms-queryresults-jdoe-test",
                    "keyPrefix": "test"
                }
            }
        },
        "sqlParameters": "***",
        "membershipIdentifier": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "type": "SQL"
    },
    "responseElements": {
        "Access-Control-Expose-Headers": "x-amzn-RequestId,x-amzn-ErrorType,x-amzn-
ErrorMessage,Date",
        "message": "Column(s) [identifier] is not allowed in select"
    },
    "requestID": "e29f9f74-8299-4a83-9d18-5ddce7302f07",
    "eventID": "c8ee3498-8e4e-44b5-87e4-ab9477e56eb5",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

AWS Clean Rooms によるリソースの作成 AWS CloudFormation

AWS Clean Rooms AWS リソースのモデル化と設定を支援するサービスと統合されています。 AWS CloudFormationこの統合により、リソースとインフラストラクチャの作成、管理に費やす時間を短縮できます。 AWS 必要なすべてのリソースを記述し、 AWS CloudFormation それらのリソースを自動的にプロビジョニングして構成するテンプレートを作成します。リソースの例には、コラボレーション、設定済みテーブル、設定済みテーブルの関連付け、メンバーシップなどがあります。

を使用すると AWS CloudFormation、 AWS Clean Rooms テンプレートを再利用してリソースを一貫して繰り返し設定できます。リソースの記述を 1 回行ってから、 AWS アカウント 同じリソースを複数回の AD で何度もプロビジョニングします。 AWS リージョン

AWS Clean RoomsAWS CloudFormation とテンプレート

AWS Clean Rooms 関連サービスのリソースをプロビジョニングして設定するには、AWS 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には、 AWS CloudFormation スタックにプロビジョニングするリソースが記述されています。JSON や YAML に慣れていない場合は、 AWS CloudFormation Designer を使用してテンプレートを使い始めることができます。 AWS CloudFormation 詳細については、「AWS CloudFormation ユーザーガイド」の「AWS CloudFormation Designer とは」を参照してください。

AWS Clean Rooms でのコラボレーション、設定済みテーブル、設定済みテーブル関連付け、メンバーシップの作成をサポートします。 AWS CloudFormationコラボレーション、設定済みテーブル、設定済みテーブルの関連付け、メンバーシップの 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 Clean Rooms 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。

- 分析テンプレート

  名前、説明、形式、ソース、パラメータ、 AWS Clean Rooms タグを含む分析テンプレートを指定します。

  詳細については、次のトピックを参照してください。

「AWS Clean Rooms ユーザーガイド」の「AWS::CleanRooms::AnalysisTemplate」

「CreateAnalysisTemplate API リファレンス」の「AWS Clean Rooms 」

- コラボレーション

  名前、説明、タイプ、パラメーター、 AWS Clean Rooms タグを含むコラボレーションを指定します。

  詳細については、次のトピックを参照してください。

  「AWS CloudFormation ユーザーガイド」の「AWS::CleanRooms::Collaboration」

  「CreateCollaboration API リファレンス」の「AWS Clean Rooms 」

- 設定済みテーブル

  許可する列、分析方法 AWS Clean Rooms、説明、名前、テーブル参照、プライバシーバジェット、タグなど、設定済みのテーブルを指定します。設定済みテーブルは、で使用するように設定された、 AWS Glue Data Catalog 内の既存のテーブルへの参照を表します AWS Clean Rooms。設定済みテーブルには、データの使用方法を決定する分析ルールが含まれています。

  詳細については、次のトピックを参照してください。

  「AWS CloudFormation ユーザーガイド」の「AWS::CleanRooms::ConfiguredTable」

  「CreateConfiguredTable API リファレンス」の「AWS Clean Rooms 」

- 設定済みテーブルの関連付け

  ID、説明、メンバーシップ ID AWS Clean Rooms、名前、ロール、Amazon リソースネーム (ARN)、タグなど、設定済みのテーブル関連付けを指定します。設定済みテーブルの関連付けにより、設定済みテーブルとコラボレーションがリンクします。

  詳細については、次のトピックを参照してください。

  「AWS CloudFormation ユーザーガイド」の「AWS::CleanRooms::ConfiguredTableAssociation」

  「CreateConfiguredTableAssociation API リファレンス」の「AWS Clean Rooms 」

- メンバーシップ

  特定のコラボレーション ID のメンバーシップを指定して、 AWS Clean Roomsのコラボレーションに参加します。

詳細については、次のトピックを参照してください。

「AWS CloudFormation ユーザーガイド」の「AWS::CleanRooms::Membership」

「CreateMembership API リファレンス」の「AWS Clean Rooms 」

- プライバシー予算テンプレート

  AWS Clean Rooms プライバシー予算、クエリごとに追加されるノイズ、毎月のプライバシー予算更新を含むプライバシー予算テンプレートを指定します。

  詳細については、次のトピックを参照してください。

  「AWS CloudFormation ユーザーガイド」の「AWS::CleanRooms::PrivacyBudgetTemplate」

  「CreatePrivacyBudgetTemplate API リファレンス」の「AWS Clean Rooms 」
- トレーニングデータセットを作成します。

  Clean Room ML AWS Glue モデルのトレーニングデータセットをテーブルから指定します。

  詳細については、次のトピックを参照してください。

  「AWS CloudFormation ユーザーガイド」の「AWS::CleanRoomsML::TrainingDataset」

  CreateTrainingDataset『クリーンルーム ML API リファレンス』にあります。

詳細についてはこちらをご覧ください。 AWS CloudFormation

詳細については AWS CloudFormation、以下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
- AWS 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 Clean Rooms

には、 ごとに、以前 AWS アカウント は制限と呼ばれていたデフォルトのクォータがあります AWS のサービス。特に明記されていない限り、各クォータは に固有です AWS リージョン。一部のクォータの増加を要求できますが、他のクォータは増加できません。

のクォータを表示するには AWS Clean Rooms、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 Clean Rooms] 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。クォータが Service Quotas でまだ利用できない場合は、サービス制限引き上げフォーム を使用します。

には、 に関連する次のクォータ AWS アカウント があります AWS Clean Rooms。

リソース	デフォルト	[Description] (説明)
コラボレーションあたりの招待済みメンバー	5	各コラボレーションで招待されるメンバーの最大数
アカウントあたりのメンバーシップ	100	1 つのアカウントのメンバーシップの最大数
アカウントあたりの作成済みコラボレーション	10	各アカウントで作成されるコラボレーションの最大数
アカウントあたりの設定済みテーブル	60	1 つのアカウントで作成できる設定済みテーブルの最大数
メンバーシップあたりのテーブルの関連付け	25	アクティブなメンバーシップあたりの関連付けられたテーブルの最大数
メンバーシップあたりの同時実行クエリ	5	メンバーシップあたりの継続的な同時クエリの最大数
設定済みテーブル許可リストあたりの列	100	設定された各テーブルでホワイトリストに登録できる列の最大数

リソース	デフォルト	[Description] (説明)
保護されたクエリあたりの設定済みテーブル	15	保護されたクエリで設定されたテーブルの最大数
メンバーシップあたりの分析テンプレート	25	メンバーシップあたりの分析テンプレートの最大数
メンバーシップごとに設定された類似モデル (オーディエンスモデル) の関連付け	5	メンバーシップごとに設定されている類似モデルの関連付けの最大数。

リソースパラメータの制限

リソース	デフォルト	[Description] (説明)
分析ルールサイズ	100 KB	分析ルールの JSON の最大サイズ
クエリテキストの長さ	90 KB (差分プライバシークエリの場合は 8 KB)	SQL クエリステートメントの最大テキスト長
クエリ実行時間	12 時間ごと	タイムアウトまでのクエリの最大実行時間
クエリデータファイルの出力サイズ	6.2 GB	保護されたクエリからの出力ファイルの最大サイズ

エンドポイントクォータごとに、アカウントごとに次の API トランザクション/秒 (TPS) AWS アカウント があります。

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
`BatchGetCollaborationAnalysisTemplate` リクエストのレート	5 TPS	1 秒あたりの `BatchGetCollaborationAnalys`

リソース	[Rate limit] (レート制限)	説明
		isTemplate API コールの最大数
BatchGetSchema リクエストのレート	5 TPS	1 秒あたりの BatchGetSchema API コールの最大数
CreateAnalysisTemplate リクエストのレート	5 TPS	1 秒あたりの CreateAnalysisTemplate API コールの最大数
CreateCollaboration リクエストのレート	5 TPS	1 秒あたりの CreateCollaboration API コールの最大数
CreateConfiguredAudienceModelAssociation リクエストのレート	5 TPS	1 秒あたりの CreateConfiguredAudienceModelAssociation コールの最大数
CreateConfiguredTable リクエストのレート	5 TPS	1 秒あたりの CreateConfiguredTable コールの最大数
CreateConfiguredTableAnalysisRule リクエストのレート	5 TPS	1 秒あたりの CreateConfiguredTableAnalysisRule コールの最大数
CreateConfiguredTableAssociation リクエストのレート	5 TPS	1 秒あたりの CreateConfiguredTableAssociation コールの最大数
CreateMembership リクエストのレート	5 TPS	1 秒あたりの CreateMembership コールの最大数
CreatePrivacyBudgetTempate リクエストのレート	5 TPS	1 秒あたりの CreatePrivacyBudgetTemplate コールの最大数

リソース	[Rate limit] (レート制限)	説明
DeleteAnalysisTemplate リクエストのレート	5 TPS	1 秒あたりの DeleteAnalysisTemplate コールの最大数
DeleteCollaboration リクエストのレート	5 TPS	1 秒あたりの DeleteCollaboration コールの最大数
DeleteConfiguredAudienceModelAssociation リクエストのレート	5 TPS	1 秒あたりの DeleteConfiguredAudienceModelAssociation コールの最大数
DeleteConfiguredTable リクエストのレート	5 TPS	1 秒あたりの DeleteConfiguredTable コールの最大数
DeleteConfiguredTableAnalysisRule リクエストのレート	5 TPS	1 秒あたりの DeleteConfiguredTableAnalysisRule コールの最大数
DeleteConfiguredTableAssociation リクエストのレート	5 TPS	1 秒あたりの DeleteConfiguredTableAssociation コールの最大数
DeleteMember リクエストのレート	5 TPS	1 秒あたりの DeleteMember コールの最大数
DeleteMembership リクエストのレート	5 TPS	1 秒あたりの DeleteMembership コールの最大数
DeletePrivacyBudgetTemplate リクエストのレート	5 TPS	1 秒あたりの DeletePrivacyBudgetTemplate コールの最大数

リソース	[Rate limit] (レート制限)	説明
GetAnalysisTemplate リクエストのレート	5 TPS	1 秒あたりの GetAnalysisTemplate コールの最大数
GetCollaboration リクエストのレート	5 TPS	1 秒あたりの GetCollaboration コールの最大数
GetCollaborationConfiguredAudienceModelAssociation リクエストのレート	5 TPS	1 秒あたりの GetCollaborationConfiguredAudienceModelAssociation コールの最大数
GetCollaborationPrivacyBudgetTemplate リクエストのレート	5 TPS	1 秒あたりの GetCollaborationPrivacyBudgetTemplate コールの最大数
GetConfiguredAudienceModelAssociation リクエストのレート	5 TPS	1 秒あたりの GetConfiguredAudienceModelAssociation コールの最大数
GetConfiguredTable リクエストのレート	5 TPS	1 秒あたりの GetConfiguredTable コールの最大数
GetConfiguredTableAnalysisRule リクエストのレート	5 TPS	1 秒あたりの GetConfiguredTableAnalysisRule コールの最大数
GetConfiguredTableAssociation リクエストのレート	20 TPS	1 秒あたりの GetConfiguredTableAssociation コールの最大数
GetMembership リクエストのレート	5 TPS	1 秒あたりの GetMembership コールの最大数

リソース	[Rate limit] (レート制限)	説明
GetPrivacyBudgetTemplate リクエストのレート	5 TPS	1 秒あたりの GetPrivacyBudgetTemplate コールの最大数
GetProtectedQuery リクエストのレート	20 TPS	1 秒あたりの GetProtectedQuery コールの最大数
GetSchema リクエストのレート	5 TPS	1 秒あたりの GetSchema コールの最大数
GetSchemaAnalysisRule リクエストのレート	5 TPS	1 秒あたりの GetSchemaAnalysisRule コールの最大数
ListAnalysisTemplates リクエストのレート	5 TPS	1 秒あたりの ListAnalysisTemplates コールの最大数
ListCollaborationConfiguredAudienceModelAssociations リクエストのレート	5 TPS	1 秒あたりの ListCollaborationConfiguredAudienceModelAssociations コールの最大数
ListCollaborationPrivacyBudgets リクエストのレート	5 TPS	1 秒あたりの ListCollaborationPrivacyBudgets コールの最大数
ListCollaborationPrivacyBudgetTemplates リクエストのレート	5 TPS	1 秒あたりの ListCollaborationPrivacyBudgetTemplates コールの最大数
ListCollaborations リクエストのレート	5 TPS	1 秒あたりの ListCollaborations コールの最大数

リソース	[Rate limit] (レート制限)	説明
ListConfiguredAudienceModelAssociations リクエストのレート	5 TPS	1 秒あたりの ListConfiguredAudienceModelAssociations コールの最大数
ListConfiguredTableAssociations リクエストのレート	5 TPS	1 秒あたりの ListConfiguredTableAssociations コールの最大数
ListConfiguredTables リクエストのレート	5 TPS	1 秒あたりの ListConfiguredTables コールの最大数
ListMembers リクエストのレート	5 TPS	1 秒あたりの ListMembers コールの最大数
ListMemberships リクエストのレート	5 TPS	1 秒あたりの ListMemberships コールの最大数
ListPrivacyBudgets リクエストのレート	5 TPS	1 秒あたりの ListPrivacyBudgets コールの最大数
ListPrivacyBudgetTemplates リクエストのレート	5 TPS	1 秒あたりの ListPrivacyBudgetTemplates コールの最大数
ListProtectedQueries リクエストのレート	5 TPS	1 秒あたりの ListProtectedQueries コールの最大数
ListSchemas リクエストのレート	5 TPS	1 秒あたりの ListSchemas コールの最大数
StartProtectedQuery リクエストのレート	5 TPS	1 秒あたりの StartProtectedQuery コールの最大数

リソース	[Rate limit] (レート制限)	説明
UpdateAnalysisTemplate リクエストのレート	5 TPS	1 秒あたりの UpdateAnalysisTemplate コールの最大数
UpdateCollaboration リクエストのレート	5 TPS	1 秒あたりの UpdateCollaboration コールの最大数
UpdateConfiguredAudienceModelAssociation リクエストのレート	5 TPS	1 秒あたりの UpdateConfiguredAudienceModelAssociation コールの最大数
UpdateConfiguredTable リクエストのレート	5 TPS	1 秒あたりの UpdateConfiguredTable コールの最大数
UpdateConfiguredTableAnalysisRule リクエストのレート	5 TPS	1 秒あたりの UpdateConfiguredTableAnalysisRule コールの最大数
UpdateConfiguredTableAssociation リクエストのレート	5 TPS	1 秒あたりの UpdateConfiguredTableAssociation コールの最大数
UpdatePrivacyBudgetTemplate リクエストのレート	5 TPS	1 秒あたりの UpdatePrivacyBudgetTemplate コールの最大数

AWS Clean Rooms ML API スロットリングクォータ

リソース	[Rate limit] (レート制限)	説明
CreateAudienceModel リクエストのレート	1 TPS レート、3 TPS バースト	1 秒あたりの CreateAudienceModel API コールの最大数

リソース	[Rate limit] (レート制限)	説明
CreateConfiguredAudienceModel リクエストのレート	10 TPS	1 秒あたりの CreateConfiguredAudienceModel API コールの最大数
CreateTrainingDataset リクエストのレート	10 TPS	1 秒あたりの CreateTrainingDataset API コールの最大数
DeleteAudienceGenerationJob リクエストのレート	2 TPSレート、10TPSバースト	1 秒あたりの DeleteAudienceGenerationJob API コールの最大数
DeleteAudienceModel リクエストのレート	2 TPSレート、10TPSバースト	1 秒あたりの DeleteAudienceModel API コールの最大数
DeleteConfiguredAudienceModel リクエストのレート	10 TPS	1 秒あたりの DeleteConfiguredAudienceModel API コールの最大数
DeleteConfiguredAudienceModelPolicy リクエストのレート	25 TPS	1 秒あたりの DeleteConfiguredAudienceModelPolicy API コールの最大数
DeleteTrainingDataset リクエストのレート	10 TPS	1 秒あたりの DeleteTrainingDataset API コールの最大数
GetAudienceGenerationJob リクエストのレート	50 TPS	1 秒あたりの GetAudienceGenerationJob API コールの最大数
GetAudienceModel リクエストのレート	50 TPS	1 秒あたりの GetAudienceModel API コールの最大数

リソース	[Rate limit] (レート制限)	説明
GetConfiguredAudienceModel リクエストのレート	50 TPS	1 秒あたりの GetConfiguredAudienceModel API コールの最大数
GetConfiguredAudienceModelPolicy リクエストのレート	50 TPS	1 秒あたりの GetConfiguredAudienceModelPolicy API コールの最大数
GetTrainingDataset リクエストのレート	50 TPS	1 秒あたりの GetTrainingDataset API コールの最大数
ListAudienceExportJobs リクエストのレート	50 TPS	1 秒あたりの ListAudienceExportJobs API コールの最大数
ListAudienceGenerationJobs リクエストのレート	50 TPS	1 秒あたりの ListAudienceGenerationJobs API コールの最大数
ListAudienceModels リクエストのレート	50 TPS	1 秒あたりの ListAudienceModels API コールの最大数
ListConfiguredAudienceModels リクエストのレート	50 TPS	1 秒あたりの ListConfiguredAudienceModels API コールの最大数
ListTagsForResource リクエストのレート	50 TPS	1 秒あたりの ListTagsForResource API コールの最大数
ListTrainingDatasets リクエストのレート	50 TPS	1 秒あたりの ListTrainingDatasets API コールの最大数

リソース	[Rate limit] (レート制限)	説明
PutConfiguredAudienceModelPolicy リクエストのレート	25 TPS	1 秒あたりの PutConfiguredAudienceModelPolicy API コールの最大数
StartAudienceExportJob リクエストのレート	1 TPS レート、3 TPS バースト	1 秒あたりの StartAudienceExportJob API コールの最大数
StartAudienceGenerationJob リクエストのレート	1 TPS レート、5 TPS バースト	1 秒あたりの StartAudienceGenerationJob API コールの最大数
TagResource リクエストのレート	10 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	50 TPS	1 秒あたりの UntagResource API コールの最大数
UpdateConfiguredAudienceModel リクエストのレート	10 TPS	1 秒あたりの UpdateConfiguredAudienceModel API コールの最大数

名前	デフォルト	引き上げ可能	説明
オーディエンス生成ジョブあたりのアクティブなオーディエンスのエクスポートジョブ	サポートされている各リージョン: 25	いいえ	オーディエンス生成ジョブのアクティブなオーディエンスエクスポートジョブの最大数

名前	デフォルト	引き上げ可能	説明
顧客あたりの保留中/進行中のオーディエンスのエクスポートジョブ	サポートされている各リージョン: 20	いいえ	顧客あたりの保留中/進行中のオーディエンスのエクスポートジョブの最大数
顧客あたりの保留中/進行中のオーディエンス生成ジョブ	サポートされている各リージョン: 10	はい	顧客あたりの保留中/進行中のオーディエンス生成ジョブの最大数
顧客あたりの保留中/進行中のオーディエンスモデル	サポートされている各リージョン: 2	はい	顧客あたりの保留中/進行中のオーディエンスモデルトレーニングジョブの最大数

クリーンルーム ML クォータ

リソース	デフォルト	[Description] (説明)
データセット	ジョブあたり	
インタラクションの最大数	200 億	トレーニングデータ内のインタラクションの最大数。より大きな入力はサンプルダウンされます。
インタラクションの最小数	100 万件	
類似モデルトレーニングの個別ユーザーの最大数	100 万件	さらに多く含まれる場合は、インタラクションの数でランク付けされた上位 1 億件のみが使用されます。

リソース	デフォルト	[Description] (説明)
類似モデルのトレーニングに必要な個別ユーザーの最小数	100,000	
エクスポート類似セグメント (オーディエンス) ジョブの最大ユーザー数	10,000	
モデルトレーニングに使用される個別アイテムの最大数。	100 万件	最大 5,000 万項目まで含めることができますが、使用されるのは最も一般的な 100 万項目だけです。
トレーニングデータセット内の特徴量列の最大数。	10	
ユーザーあたりの個別の項目の最小数	2	AWS Clean Rooms ML では、各行またはユーザーに、繰り返し項目を含む 2 つ以上の項目が必要です。
シードオーディエンスの最大サイズ	500,000	
シードオーディエンスの最小サイズ	500	トレーニングデータプロバイダーは、この値を最低 25 に設定できます。
API	お客様 1 人あたり	
アクティブなトレーニングデータセットの総数。	500	
アクティブな類似モデル (オーディエンスモデル) の総数	500	

リソース	デフォルト	[Description] (説明)
設定済みのアクティブな類似モデル (オーディエンスモデル) の総数	10,000	
完了した類似セグメント (オーディエンス) 生成ジョブの総数	無制限	
エクスポートが完了した類似セグメント (オーディエンス) ジョブの総数	無制限	
類似モデル (オーディエンスモデル) 生成ジョブの最大所要時間	1 日 (24 時間)	
類似セグメント (オーディエンス) 生成ジョブの最大所要時間	10 時間	シードを指定すると、Clean Rooms ML が類似セグメントを生成するまでに最大 10 時間かかります。
セグメント (オーディエンス) サイズのビンの最小割合	1%	
セグメント (オーディエンス) サイズのビンの最大割合	20%	
セグメント (オーディエンス) サイズのビンの最小絶対サイズ	個別ユーザー数の 1%	
セグメント (オーディエンス) サイズのビンの最大絶対サイズ	個別ユーザー数の 20%	

AWS Clean Rooms 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 Clean Rooms。

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AWSCleanRoomsFullAccessNoQuerying マネージドポリシーに追加されました:cleanrooms:BatchGetSchemaAnalysisRule	2024 年 5 月 13 日
AWS Clean Rooms ML が完全に利用可能になりました	AWS Clean Rooms ML は、データを相互に共有しなくても、データ内の類似ユーザーを識別するためのプライバシー強化方法を提供します。	2024 年 4 月 3 日
既存のポリシーの更新	AWSCleanRoomsFullAccess 管理ポリシーのステートメント ID が から ConsolePickQueryResultsBucket に更新されSetQueryResultsBucket、アクセス許可以降のアクセス許可をより適切に表現できるようになりました。	2024 年 3 月 21 日
AWS Clean Rooms ML の新しい マネージドポリシー	AWSCleanRoomsMLReadOnlyAccess と AWSCleanRoomsMLFullAccess という 2 つの新し	2023 年 11 月 29 日

	いマネージドポリシーが追加されました。	
AWS Clean Rooms ML (プレビュー)	AWS Clean Rooms ML は、データを相互に共有しなくても、データ内の類似ユーザーを識別するためのプライバシー強化方法を提供します。	2023 年 11 月 29 日
AWS Clean Rooms 差分プライバシー (プレビュー)	お客様は、 AWS Clean Rooms 差分プライバシーを使用してユーザーのプライバシーを保護することができるようになりました。	2023 年 11 月 29 日
支払い設定	コラボレーションクリエーターが、クエリを行えるメンバーまたはコラボレーション内の別のメンバーを、クエリの計算コストの請求先として設定できるようになりました。	2023 年 11 月 14 日
クエリ実行時間 - 更新	クエリ実行時間のタイムアウトまでの上限が 4 時間から 12 時間に更新されました。	2023 年 10 月 6 日

AWS CloudFormation リソース - 更新	AWS Clean Rooms では、、AWS::CleanRooms::MembershipProtectedQueryOutputConfiguration 、および の新しいリソースが追加されましたAWS::CleanRooms::MembershipProtectedQueryResultConfiguration AWS::CleanRooms::MembershipProtectedQueryS3OutputConfiguration 。	2023 年 9 月 7 日
AWS CloudFormation リソース - 更新	AWS Clean Rooms では、AWS::CleanRooms::AnalysisTemplate と の新しいリソースが追加されましたAWS::CleanRooms::ConfiguredTableAnalysisRuleCustom 。	2023 年 8 月 31 日
メンバー能力の個別指定	コラボレーションクリエーターが、クエリを行えるメンバー を 1 人と、結果を受け取れる別のメンバーを 1 人指定できるようになりました。これにより、コラボレーションクリエーターは、クエリを実行するメンバーがクエリ結果にアクセスできないように設定できます。	2023 年 8 月 30 日
AWS Clean Rooms 用語集	AWS Clean Rooms 用語集を追加するためのドキュメントのみの更新。	2023 年 8 月 30 日

Apache Iceberg テーブルのサポート (プレビュー)	AWS Clean Rooms でApache Icebergテーブル (プレビュー) がサポートされるようになりました。	2023 年 8 月 25 日
クォータの更新	アカウントごとのメンバーシップの新しいデフォルトクォータを反映して、「クォータ」セクションが更新されました。	2023 年 8 月 9 日
既存のポリシーの更新	AWSCleanRoomsFullAccessNoQuerying マネージドポリシーに次の新しいアクセス許可が追加されました。cleanrooms:CreateAnalysisTemplate 、cleanrooms:GetAnalysisTemplate 、cleanrooms:UpdateAnalysisTemplate 、 cleanrooms:DeleteAnalysisTemplate 、cleanrooms:ListAnalysisTemplates 、cleanrooms:GetCollaborationAnalysisTemplate 、cleanrooms:BatchGetCollaborationAnalysisTemplate 、および cleanrooms:ListCollaborationAnalysisTemplates 。	2023 年 7 月 31 日

分析テンプレートとカスタム分析ルール	AWS Clean Rooms で、分析テンプレートとカスタム分析ルールがサポートされるようになりました。分析テンプレートにより、コラボレーターは独自のカスタム SQL クエリを作成またはインポートしてコラボレーションで使用できます。カスタム分析ルールを使用すると、テーブル所有者は設定済みテーブルに対するカスタム SQL クエリを承認できます。	2023 年 7 月 31 日
分析ルールでの OR 論理条件のサポート	AWS Clean Rooms 分析ルールが OR JOIN句の論理条件をサポートするようになりました。	2023 年 6 月 29 日
CloudFormation 統合	AWS Clean Rooms が と統合されるようになりました AWS CloudFormation。	2023 年 6 月 15 日
分析ビルダー	クエリを実行して結果を受け取ることができるメンバーは、[分析ビルダー UI] を使用して、SQL コードを記述することなく一部のテーブルでクエリを実行できるようになりました。	2023 年 6 月 15 日
SQL 関数	ドキュメントのみの更新。サポートされている SQL 関数を明確にしました。	2023 年 5 月 5 日

トラブルシューティング	ドキュメントのみの更新。一般的な問題に関するトラブルシューティングセクションを追加しました。	2023 年 4 月 27 日
でサポートされているデータ型 AWS Clean Rooms	ドキュメントのみの更新により、サポートされている AWS Glue Data Catalog データ型を一覧表示する新しいセクションが追加されました。	2023 年 4 月 26 日
AWS CloudTrail イベントの例	(成功) と StartProtectedQuery (StartProtectedQuery失敗) の CloudTrail イベントの例を追加するためのドキュメントのみの更新。	2023 年 4 月 20 日
既存のポリシーの更新	`AWSCleanRoomsFullAccessNoQuerying` マネージドポリシーに次の新しいアクセス許可が追加されました。`cleanrooms:ListTagsForResource` 、`cleanrooms:UntagResource` 、および `cleanrooms:TagResource` 。詳細については、「AWS マネージドポリシー」を参照してください。	2023 年 3 月 21 日
一般提供	AWS Clean Rooms が一般公開されました。	2023 年 3 月 21 日
プレビューリリース	AWS Clean Rooms ユーザーガイドのプレビューリリース	2023 年 1 月 12 日

AWS Clean Rooms 用語集

この用語集を参照して、 AWS Clean Roomsで使用される用語を理解してください。

集計分析ルール

COUNT、SUM、または AVG 関数を使用して任意のディメンションで分析を集約するクエリを許可するクエリ制限です。このようなクエリでは行レベルの情報は明らかにはなりません。

キャンペーン計画、メディアリーチ、頻度測定、コンバージョン測定などのユースケースに対応します。

その他の分析ルールタイプとして、カスタムとリストがあります。

分析ルール

特定のタイプのクエリを許可するクエリ制限です。

分析ルールのタイプによって、設定済みテーブルに対して実行できる分析の種類が決まります。タイプごとに事前定義されたクエリ構造があります。テーブル列を構造内でどのように使用するかは、クエリコントロールを通じて制御します。

分析ルールには、集計、リスト、カスタムの 3 種類があります。

分析テンプレート

コラボレーションに固有の、事前に承認された、再利用可能なクエリです。

でサポートされているカスタム SQL クエリをサポートします AWS Clean Rooms。

SQL クエリで通常リテラル値が表示される部分にパラメータを含めることができます。サポートされているパラメータタイプの詳細については、「AWS Clean Rooms SQL リファレンス」の「データ型」を参照してください。

分析テンプレートはカスタム分析ルールでのみ使用できます。

C3R 暗号化クライアント

Cryptographic Computing for Clean Rooms (C3R) の暗号化クライアントです。

データの暗号化と復号化に使用される C3R は、コマンドラインインターフェイスを備えたクライアント側の暗号化 SDK です。

クリアテキスト列

JOIN または SELECT SQL 構文のどちらのためにも暗号化によって保護されていない列です。

クリアテキスト列は SQL クエリのどの部分でも使用できます。

コラボレーション

メンバーが設定済みテーブル AWS Clean Rooms に対して SQL クエリを実行できる安全な論理境界。

コラボレーションはコラボレーションクリエーターが作成します。

コラボレーションに参加できるのは、コラボレーションに招待されたメンバーだけです。

コラボレーションには、データに対してクエリを行えるメンバー、結果を受け取れるメンバー、クエリの計算コストを負担するメンバーをそれぞれ 1 人ずつ指定できます。

すべてのメンバーは、コラボレーションに参加する前に、コラボレーションに招待された参加者のリストを確認できます。

コラボレーションクリエーター

コラボレーションを作成するメンバーです。

コラボレーションクリエーターは 1 つのコラボレーションにつき 1 人だけです。

コラボレーションからメンバーを削除したり、コラボレーションを削除したりできるのは、コラボレーションクリエーターだけです。

設定済みテーブル

各設定済みテーブル AWS Glue Data Catalog は、 で使用するために設定された 内の既存のテーブルへの参照を表します AWS Clean Rooms。設定済みテーブルには、データの使用方法を決定する分析ルールが含まれています。

現在、 は、 を介してカタログ化された Amazon Simple Storage Service (Amazon S3) に保存されているデータの関連付け AWS Clean Rooms をサポートしています AWS Glue。

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

設定済みテーブルは 1 つ以上のコラボレーションに関連付けることができます。

> **Note**
>
> AWS Clean Rooms は現在、 に登録されている Amazon S3 バケットの場所をサポートしていません AWS Lake Formation。

カスタム分析ルール

事前に承認された特定のクエリ (分析テンプレート) を許可したり、データを使用するクエリの提供を特定のアカウントに許可したりするクエリ制限です。

ファーストタッチアトリビューション、増分分析、オーディエンス発掘分析などのユースケースに対応します。

差分プライバシーをサポートします。

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。

差分プライバシー

特定の個人について学習した結果を受け取ることができるメンバーからコラボレーションデータを保護する、数学的に厳格な手法。

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。

フィンガープリント列

JOIN SQL 構文のために暗号化によって保護されている列です。

リスト分析ルール

クエリを行えるメンバーのテーブルとこのテーブルとの重複部分について、行レベルの属性分析を出力するクエリを許可するクエリ制限です。

エンリッチメント、オーディエンス構築、サプレッションなどのユースケースに対応します。

メンバー

コラボレーション に参加している AWS 顧客。

メンバーは、それぞれの AWS アカウントを使用して識別されます。

メンバー全員がデータを寄稿できます。

クエリを行えるメンバー

コラボレーションにおいてデータにクエリを実行できるメンバーです。

クエリを行えるメンバーはコラボレーションごとに 1 人だけであり、そのメンバーは変更できません。

管理ユーザーは AWS Identity and Access Management (IAM) アクセス許可を使用して、コラボレーション内のデータをクエリできる IAM プリンシパル (ユーザーやロールなど) を制御できます。詳細については、「データを読み取るサービスロールの作成」を参照してください。

結果を受け取れるメンバー

クエリ結果を受信できるメンバーです。結果を受け取れるメンバーは、クエリ結果の設定として Amazon S3 送信先とクエリ結果の形式を指定します。

結果を受け取れるメンバーはコラボレーションごとに 1 人だけであり、そのメンバーは変更できません。

クエリの計算コストを負担するメンバー

クエリの計算コストの支払いを担当するメンバーです。

クエリの計算コストを負担するメンバーはコラボレーションごとに 1 人だけであり、そのメンバーは変更できません。

コラボレーションクリエーターが、クエリの計算コストを負担するメンバーとして誰も指定していない場合は、クエリを行えるメンバーがデフォルトの支払い担当となります。

コラボレーションで実行されたクエリの請求書は、クエリの計算コストを負担するメンバーに送られます。

メンバーシップ

メンバーがコラボレーションに参加したときに作成されるリソースです。

メンバーがコラボレーションに関連付けるすべてのリソースは、メンバーシップの一部であるか、メンバーシップに関連付けられます。

メンバーシップを所有するメンバーだけが、そのメンバーシップのリソースを追加、削除、編集できます。

シール列

SELECT SQL 構文のために暗号化によって保護されている列です。

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。