AWS Application Discovery Service

ユーザーガイド

AWS Application Discovery Service: ユーザーガイド

Table of Contents

AWS Application Discovery Service とは

AWS Application Discovery Service では、オンプレミスサーバーに関する使用状況と設定のデータを収集することで、AWS クラウドへの移行を計画できます。Application Discovery Service は AWS Migration Hub と統合されています。これにより、移行ステータスの情報が 1 つのコンソールに集約されるため、移行の追跡が簡単になります。検出したサーバーを表示し、これらをアプリケーションとしてグループ化して、各アプリケーションの移行ステータスをホームリージョンの Migration Hub コンソールから追跡できます。

検出されたすべてのデータは、AWS Migration Hub ホームリージョンに保存されます。したがって、検出および移行アクティビティを実行する前に、Migration Hub コンソールまたは CLI コマンドを使用してホームリージョンを設定する必要があります。データは、Microsoft Excel や AWS 分析ツール (Amazon Athena や Amazon QuickSight など) にエクスポートして分析できます。

Application Discovery Service API を使用して、検出したサーバーのシステムパフォーマンスと使用状況データをエクスポートできます。このデータをコストモデルに取り込むことで、これらのサーバーを AWS で実行した場合のコストを計算します。さらに、サーバー間に存在するネットワーク接続に関するデータをエクスポートできます。この情報により、サーバー間のネットワーク依存関係を確認し、サーバーをアプリケーションとしてグループ化して、移行計画に役立てることができます。

> Note
>
> データはホームリージョンに保存されるため、検出プロセスを開始する前に、ホームリージョンを AWS Migration Hub で設定する必要があります。ホーム領域の操作の詳細については、「ホームリージョン」を参照してください。

Application Discovery Service は、2 つの方法で、オンプレミスのサーバーを検出して関連データを収集します。

- エージェントレス検出を実行するには、VMware vCenter を通じて AWS エージェントレス検出コネクタ (OVA ファイル) をデプロイします。検出コネクタを設定すると、vCenter に関連付けられている仮想マシン (VM) とホストが識別されます。検出コネクタは、次の静的構成データを収集します。サーバのホスト名、IP アドレス、MAC アドレス、ディスクリソースの割り当て さらに、VM ごとに使用状況データを収集し、CPU、RAM、ディスク I/O などのメトリクスの平均とピークの使用率を計算します。
- エージェントベース検出を実行するには、VM と物理サーバーのそれぞれに AWS Application Discovery Agent をデプロイします。エージェントのインストーラは Windows および Linux オペレーティングシステムで使用できます。これにより、静的な設定データ、詳細な時系列のシステムパフォーマンス情報、着信/発信のネットワーク接続、および実行中のプロセスが収集されます。

Application Discovery Service は、AWS パートナーネットワーク (APN) のパートナーが提供するアプリケーション検出ソリューションと統合できます。これらのサードパーティー製ソリューションを使用すると、検出コネクタや検出エージェントを使用せずに、オンプレミス環境の詳細を Migration Hub に直接インポートできます。サードパーティーのアプリケーション検出ツールは、AWS Application Discovery Service をクエリしたり、パブリック API を使用して Application Discovery Service データベースに書き込んだりできます。このようにして、Migration Hub にデータをインポートして表示できるため、アプリケーションをサーバーに関連付けたり、移行を追跡したりできます。

VMware の検出に関する詳細

VMware vCenter 環境で実行されている仮想マシン (VM) がある場合は、検出コネクタを使用してシステム情報を収集できます。各 VM にエージェントをインストールする必要はありません。代わりに、このオン

プレミスアプライアンスを vCenter 内にロードし、このアプライアンスですべてのホストと VM を検出することを許可します。

Discovery Connector は、vCenter で実行されている各 VM のシステムパフォーマンス情報とリソース使用状況をキャプチャします。ただし、各 VM の「内部を見る」ことはできません。したがって、各 VM で実行されているプロセスや使用されているネットワーク接続を判断することはできません。移行計画に役立てるために、このレベルの詳細が必要な場合や既存の VM の一部の詳細を確認する場合は、必要に応じて Discovery Agent をインストールできます。

また、VMware でホストされている VM の場合は、検出コネクタと検出エージェントの両方を使用して検出を同時に実行できます。各検出ツールで収集される正確なデータタイプの詳細については、「検出コネクタによって収集されたデータ (p. 20)」と「エージェント検出によって収集されたデータ (p. 7)」を参照してください。

コネクタとエージェントの比較

次の表に、2 つの主要な Application Discovery Service ツールの簡単な比較を示します。

	検出コネクタ	検出エージェント
サポートされるサーバータイプ VMware 仮想マシン 物理サーバー	 はい いいえ	 はい はい
デプロイメント サーバーごと vCenter ごと	 いいえ はい	 はい いいえ
収集されるデータ 静的な設定データ VM 使用率メトリクス 時系列のパフォーマンス情報 ネットワーク着信/発信接続 実行中のプロセス	 はい はい いいえ いいえ いいえ	 はい いいえ はい (エクスポートのみ) はい (エクスポートのみ) はい (エクスポートのみ)
サポートされる OS	VMware vCenter (V5.5、V6、および V6.5) で実行されている任意の OS	サポートされている Linux および Windows オペレーティングシステムのリストについては、検出エージェントのインストールの前提条件 (p. 8)。

Assumptions

Application Discovery Service を使用するには、以下を前提とします。

- サインアップしている必要がありますAWS。詳細については、「AWS Application Discovery Service のセットアップ (p. 4)」を参照してください。
- Migration Hub ホームリージョンが選択されました。詳細については、ホームリージョンに関するドキュメントを参照してください。

期待する内容は次のとおりです。

- Migration Hub ホームリージョンは、Application Discovery Service が検出および計画データを保存する唯一のリージョンです。
- 検出エージェント、コネクタ、およびインポートは、選択した Migration Hub ホームリージョンでのみ使用できます。
- Application Discovery Service を使用できる AWS リージョンのリストについては、Amazon Web Services 全般のリファレンス。

AWS Application Discovery Service のセットアップ

AWS Application Discovery Service を初めて使用する場合は、事前に以下のタスクをすべて実行してください。

ステップ 1: AWS へのサインアップ

このセクションでは、AWS アカウントにサインアップします。すでに AWS アカウントをお持ちの場合は、この手順をスキップしてください。

Amazon Web Services サインアップすると (AWS)、あなたのAWSアカウントは AWS Application Discovery Service など、すべての AWS のサービスに自動的にサインアップします。料金が発生するのは、実際に使用したサービスの分のみです。

AWS アカウントを作成するには

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、電話のキーパッドを用いて確認コードを入力することが求められます。

次のタスクで AWS アカウント番号が必要となるので、メモしておいてください。

ステップ 2: IAM ユーザーの作成

AWS アカウントを初めて作成するときは、アカウント内のすべての AWS サービスとリソースに対して完全なアクセス権限を持つシングルサインイン ID を取得します。このアイデンティティは、AWS アカウントのルートユーザーと呼ばれます。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により AWS Management Console にサインインすると、アカウント内のすべての AWS リソースへのフルアクセスが許可されます。

日常的なタスクには (それが管理タスクであっても)、ルートユーザーを使用しないよう強くお勧めします。代わりに、セキュリティに関するベストプラクティスに従ってください個々の IAM ユーザーの作成を作成し、AWS Identity and Access Management(IAM) 管理者ユーザー。その後、root ユーザーの認証情報を安全な場所に保管し、それらを使用して少数のアカウントおよびサービス管理タスクのみを実行します。

管理者ユーザーの作成に加え、管理者以外の IAM ユーザーも作成する必要があります。以下のトピックでは、両タイプの IAM ユーザーの作成方法について説明します。

トピック

IAM 管理者ユーザーの作成

既定では、管理者アカウントは Application Discovery Service へのアクセスに必要なすべてのポリシーを継承します。

管理者ユーザーを作成するには

- AWS アカウントで管理ユーザーを作成します。手順については、以下を参照してください。最初の IAM ユーザーおよび管理者グループの作成()IAM ユーザーガイド。

IAM 管理者以外のユーザーの作成

管理者以外の IAM ユーザーを作成するときは、セキュリティのベストプラクティス最小権限を付与するで、ユーザーに最低限の権限を付与します。

IAM 管理ポリシーを使用して、管理者以外の IAM ユーザーによる Application Discovery Service へのアクセスレベルを定義します。Application Discovery Service 管理ポリシーについては、」Application Discovery Service での AWS 管理 (事前定義) ポリシー (p. 79)。

管理者以外のユーザーを作成するには

- AWS アカウントで管理ユーザーを作成します。手順については、以下を参照してください。最初の IAM 委任ユーザーおよびグループの作成()IAM ユーザーガイド。

AWS アプリケーション検出エージェント

-AWS検出エージェントはAWS検出や移行のためにオンプレミスサーバーや VM にインストールされるソフトウェアです。エージェントは、システム設定、システムパフォーマンス、実行中のプロセス、およびシステム間のネットワーク接続の詳細をキャプチャします。エージェントは、ほとんどの Linux および Windows オペレーティングシステムをサポートし、オンプレミスの物理サーバー、Amazon EC2 インスタンス、仮想マシンにデプロイできます。

> **Note**
>
> 検出エージェントを展開する前に、選択する必要があります。Migration Hub ホーム。ホームリージョンにはエージェントを登録する必要があります。

検出エージェントはローカル環境で実行され、ルート権限を必要とします。検出エージェントを開始すると、ホームリージョンと安全に接続され、Application Discovery Service に登録されます。

- たとえば、場合`eu-central-1`がホームリージョンの場合、に登録されます。`arsenal-discovery.eu-central-1.amazonaws.com`Application Discovery Service と。
- または、us-west-2 を除く他のすべてのリージョンで、必要に応じてホームリージョンが置き換えられます。
- もし`us-west-2`がホームリージョンの場合、に登録されます。`arsenal.us-west-2.amazonaws.com`Application Discovery Service と。

仕組み

登録後、エージェントは、該当するホストや VM のためにデータ収集を開始します。エージェントは、15 分間隔でApplication Discovery Service に ping を送信して設定情報を確認します。

収集されるデータには、システム仕様、時系列の使用状況やパフォーマンスのデータ、ネットワーク接続、処理データなどが含まれます。この情報を使用して IT アセットとネットワーク依存関係をマッピングできます。これらのすべてのデータポイントは、これらのサーバーを AWS で実行する場合のコストを算定したり、移行を計画したりするのに役立ちます。

データは、Transport Layer Security (TLS) を通じて、検出エージェントからアプリケーション検出サービスに安全に転送されます。エージェントは、新しいバージョンが利用可能になると自動的にアップグレードするように設定されています。必要に応じて、この設定は変更できます。

> **Tip**
>
> Discovery Agent をダウンロードしてインストールを開始する前に、必ず「」に目を通し、必須の前提条件を確認してください。検出エージェントのインストールの前提条件 (p. 8)

トピック

- エージェントデータ収集 (p. 18)

エージェント検出によって収集されたデータ

AWS Application Discovery Agent は、オンプレミスサーバーや VM にインストールされるソフトウェアです。検出エージェントは、システム構成、時系列の使用状況やパフォーマンスのデータ、処理データ、送信制御プロトコル (TCP) ネットワーク接続を収集します。このセクションでは、収集されるデータについて説明します。

検出エージェントで収集されるデータのテーブル凡例:

- ホストという用語は、物理サーバーまたは VM を指します。
- 収集されたデータは、特に断らない限り、キロバイト (KB) 単位です。
- Migration Hub コンソールで対応するデータはメガバイト (MB) 単位です。
- ポーリング間隔は約 15 分です。
- アスタリスク (*) が付いているデータフィールドは、エージェントの API エクスポート関数から生成される `.csv` ファイルでのみ使用できます。

データフィールド	説明
agentAssignedProcessId*	エージェントによって検出されたプロセスのプロセス ID
agentId	エージェント固有の ID
agentProvidedTimeStamp*	エージェントの監視日時 (mm/dd/yyyy hh:mm:ss am/pm)
cmdLine*	コマンドラインに入力されるプロセス
cpuType	ホストで使用される CPU (中央処理装置) のタイプ
destinationIp*	パケットを送信する先のデバイスの IP アドレス
destinationPort*	データ/リクエストを送信する先のポート番号
family*	ルーティングファミリーのプロトコル
freeRAM (MB)	アプリケーションで即時に使用できる無料 RAM およびキャッシュ RAM (MB 単位)
gateway*	ネットワークのノードアドレス
hostName	データを収集したホストの名前
hypervisor	ハイパーバイザーのタイプ
ipAddress	ホストの IP アドレス
ipVersion*	IP バージョン番号
isSystem*	OS がプロセスを所有しているかどうかを示すブール属性
macAddress	ホストの MAC アドレス

データフィールド	説明
name*	収集されているホスト、ネットワーク、メトリクスなどのデータの名前
netMask*	ネットワークホストが属する IP アドレスプレフィックス
osName	ホストのオペレーティングシステムの名前
osVersion	ホストのオペレーティングシステムのバージョン
パス	コマンドラインから発信されるコマンドのパス
sourceIp*	IP パケットの送信元デバイスの IP アドレス
sourcePort*	データ/リクエストの送信元のポート番号
timestamp*	報告された属性がエージェントでログに記録された日時
totalCpuUsagePct	ポーリング間隔中のホストの CPU 使用率
totalDiskBytesReadPerSecond (Kbps)	ホストのディスク空き領域の合計量
totalDiskBytesWrittenPerSecond (Kbps)	ホストのディスクの合計サイズ
totalDiskFreeSize (GB)	ディスク空き容量 (GB 単位)
totalDiskReadOpsPerSecond	1 秒あたりの読み取り I/O オペレーションの合計数
totalDiskSize (GB)	ディスクの合計容量 (GB 単位)
totalDiskWriteOpsPerSecond	1 秒あたりの書き込み I/O オペレーションの合計数
totalNetworkBytesReadPerSecond (Kbps)	1 秒あたりに読み取られたバイトスループットの合計値
totalNetworkBytesWrittenPerSecond (Kbps)	1 秒あたりに書き込まれたバイトスループットの合計値
totalNumCores	CPU 内の独立した処理装置の合計数
totalNumCpus	CPU の合計数
totalNumDisks	ホストの物理ハードディスクの数
totalNumLogicalProcessors*	物理コアの合計数と各コアで実行できるスレッド数を乗算した値
totalNumNetworkCards	サーバーのネットワークカードの合計数
totalRAM (MB)	ホストで使用可能な RAM の合計量
transportProtocol*	トランスポートプロトコルの使用タイプ

検出エージェントのインストールの前提条件

AWS アプリケーション検出エージェント (検出エージェント) を正常にインストールする前に実行する必要がある前提条件とタスクを次に示します。

- 設定する必要がありますAWS Migration Hub ホーム探索エージェントのインストールを開始する前に。
- 1.x バージョンのエージェントがインストールされている場合は、最新バージョンをインストールする前に削除する必要があります。
- エージェントがインストールされているホストがLinuxを実行している場合は、ホストが少なくともIntel i686 CPUアーキテクチャ（P6マイクロアーキテクチャとも呼ばれる）をサポートしていることを確認します。
- 使用しているオペレーティングシステム (OS) 環境がサポートされていることを確認します。

  Linux

  Amazon Linux 2012.03、2015.03

  Amazon Linux 2 (2018 年 9 月 25 日更新以降)

  Ubuntu 12.04、14.04

  Red Hat Enterprise Linux 5.11、6.10、7.3、7.3、7.7、8.1

  CentOS 5.11、6.9、7.3

  SUSE 11 SP4, 12 SP5

  Windows

  Windows Server 2003 R2 SP2

  Windows Server 2008 R1 SP2、2008 R2 SP1

  Windows Server 2012 R1、2012 R2

  Windows Server 2016

  Windows Server 2019
- ネットワークからの発信接続が制限されている場合は、ファイアウォール設定を更新する必要があります。エージェントには、TCP ポート 443 を介した `arsenal` へのアクセスが必要です。着信ポートを開く必要はありません。

  たとえば、ホームリージョンが `eu-central-1` の場合、`https://arsenal-discovery.eu-central-1.amazonaws.com:443` を使用できます。
- 自動アップグレードが機能するには、ホームリージョンの Amazon S3 へのアクセスが必要です。
- を作成するAWS Identity and Access Management(IAM) ユーザーをコンソールで管理し、既存の`AWSApplicationDiscoveryAgentAccess`IAM 管理ポリシー。このポリシーにより、ユーザーはお客様に代わって必要なエージェントアクションを実行できます。管理ポリシーの詳細については、「Application Discovery Service での AWS 管理 (事前定義) ポリシー (p. 79)」を参照してください。
- ネットワークタイムプロトコル (NTP) サーバーからの時刻のずれを確認し、必要に応じて修正します。時刻の同期が正しくないと、エージェント登録コールが失敗します。

  Note

  検出エージェントには 32 ビットのエージェント実行可能ファイルは、32 ビットと 64 ビットのオペレーティングシステムで動作します。実行可能ファイルを 1 つにすることで、デプロイに必要なインストールパッケージの数が減ります。この実行可能エージェントは、Linux および Windows OS で動作します。これについては、以降のそれぞれのインストールセクションで説明します。

Linux にエージェントをインストールする

Linux で次の手順を完了します。必ず、お客様のことを確認します。Migration Hub ホームが設定されている場合、この手順を開始する前に設定されています。

Note

以前の Linux バージョンを使用している場合は、「古い Linux プラットフォームでの要件 (p. 11)」を参照してください。

データセンターに AWS アプリケーション検出エージェントをインストールするには

1. Linux ベースのサーバーまたは VM にサインインし、エージェントコンポーネントを格納するための新しいディレクトリを作成します。
2. 新しいディレクトリに切り替え、コマンドラインまたはコンソールからインストールスクリプトをダウンロードします。

   a. コマンドラインからダウンロードするには、次のコマンドを実行します。

   ```
   curl -o ./aws-discovery-agent.tar.gz https://s3-us-west-2.amazonaws.com/aws-discovery-agent.us-west-2/linux/latest/aws-discovery-agent.tar.gz
   ```

   b. Migration Hub コンソールからダウンロードするには、次の操作を行います。

      i. コンソールを開いて、[Discovery Tools (検出ツール)] ページに移動します。

      ii. [Discovery Agent (検出エージェント)] ボックスで、[Download agent (エージェントのダウンロード)] を選択し、表示されたリストボックスで [Linux] を選択します。ダウンロードがすぐに開始されます。

3. 次の 3 つのコマンドを使用して、インストールパッケージの暗号署名を確認します。

   ```
   curl -o ./agent.sig https://s3.us-west-2.amazonaws.com/aws-discovery-agent.us-west-2/linux/latest/aws-discovery-agent.tar.gz.sig
   ```

   ```
   curl -o ./discovery.gpg https://s3.us-west-2.amazonaws.com/aws-discovery-agent.us-west-2/linux/latest/discovery.gpg
   ```

   ```
   gpg --no-default-keyring --keyring ./discovery.gpg --verify agent.sig aws-discovery-agent.tar.gz
   ```

   エージェントパブリックキー (discovery.gpg) のフィンガープリントは、`7638 F24C 6717 F97C 4F1B 3BC0 5133 255E 4DF4 2DA2` です。

4. 次に示すように、tarball から抽出します。

   ```
   tar -xzf aws-discovery-agent.tar.gz
   ```

5. エージェントをインストールするには、次のインストール方法のいずれかを選択します。

目的	操作
検出エージェントのインストール	エージェントをインストールするには、次の例に示すように agent install コマンドを実行します。この例では、*##########*をホームリージョンの名前に置き換えます。*aws-###### ID*をアクセスキー ID に置き換え、*aws##### ###*をシークレットアクセスキーに置き換えます。 `sudo bash install -r your-home-region -k aws-access-key-id -s aws-secret-access-key` デフォルトでは、エージェントは、アップデートが利用可能になると、自動的にダウンロードして適用します。

目的	操作
	このデフォルト設定の使用が推奨されます。 ただし、エージェントがアップデートを自動的にダウンロードして適用しないようにするには、`-u false`パラメーターを使用します。
（オプション）検出エージェントをインストールし、非透過プロキシを構成する	非透過プロキシを設定するには、agent install コマンドに次のパラメータを追加します。 • -eプロキシのパスワード。 • -fプロキシポート番号。 • -gプロキシスキーム。 • -iプロキシユーザー名。 非透過プロキシパラメータを使用した agent install コマンドの例を以下に示します。 `sudo bash install -r your-home-region -k aws-access-key-id -s aws-secret-access-key -d myproxy.mycompany.com -e mypassword -f proxy-port-number -g https -i myusername` 認証が不要なプロキシの場合は、`-e`および`-i`パラメータ。 install コマンドの例では、`https`で、プロキシが HTTP を使用する場合は、`http`向けの`-g`パラメータ値。

6. ネットワークからの発信接続が制限されている場合は、ファイアウォール設定を更新する必要があります。エージェントには、TCP ポート 443 を介した `arsenal` へのアクセスが必要です。着信ポートを開く必要はありません。

   たとえば、ホームリージョンが `eu-central-1` の場合、`https://arsenal-discovery.eu-central-1.amazonaws.com:443` を使用できます。

トピック

古い Linux プラットフォームでの要件

一部の古い Linux プラットフォーム (SUSE 10、CentOS 5、RHEL 5 など) はサポートが終了しているか、最低限のサポート対象となります。これらのプラットフォームは、古い暗号スイートがあり、エージェントの更新スクリプトでインストールパッケージをダウンロードできない場合があります。

Curl

アプリケーション検出エージェントは、AWS サーバーとのセキュアな通信のために `curl` を要求します。一部の古いバージョンの `curl` は、最新のウェブサービスと安全に通信することはできません。

すべてのオペレーションで`curl` バージョンが含まれるアプリケーション検出エージェントを使用するには、`-c true` パラメータでインストールスクリプトを実行します。

認証機関バンドル

以前の Linux システムの認証機関 (CA) バンドルは古いため、安全なインターネット通信が確保できない場合があります。

すべてのオペレーションで CA バンドルが含まれるアプリケーション検出エージェントを使用するには、`-b true` パラメータでインストールスクリプトを実行します。

これらのインストールスクリプトオプションは一緒に使用できます。次のコマンド例では、両方のスクリプトパラメータがインストールスクリプトに渡されます。

```
sudo bash install -r your-home_region -k aws-access-key-id -s aws-secret-access-key -c true
 -b true
```

Linux で検出エージェントプロセスを管理する

検出エージェントの動作は、システムレベルで管理できます。`systemd`,`Upstart`, または`System V init`ツール。以下のタブは、それぞれのツールでサポートされているタスクのコマンドの概要を示しています。

systemd

Application Discovery Agent の管理コマンド

タスク	コマンド
エージェントが実行されていることを確認	`sudo systemctl status aws-discovery-daemon.service`
エージェントの開始	`sudo systemctl start aws-discovery-daemon.service`
エージェントの停止	`sudo systemctl stop aws-discovery-daemon.service`
エージェントの再起動	`sudo systemctl restart aws-discovery-daemon.service`

Upstart

Application Discovery Agent の管理コマンド

タスク	コマンド
エージェントが実行されていることを確認	`sudo initctl status aws-discovery-daemon`

タスク	コマンド
エージェントの開始	`sudo initctl start aws-discovery-daemon`
エージェントの停止	`sudo initctl stop aws-discovery-daemon`
エージェントの再起動	`sudo initctl restart aws-discovery-daemon`

System V init

Application Discovery Agent の管理コマンド

タスク	コマンド
エージェントが実行されていることを確認	`sudo /etc/init.d/aws-discovery-daemon status`
エージェントの開始	`sudo /etc/init.d/aws-discovery-daemon start`
エージェントの停止	`sudo /etc/init.d/aws-discovery-daemon stop`
エージェントの再起動	`sudo /etc/init.d/aws-discovery-daemon restart`

Linux で検出エージェントをアンインストールする

このセクションでは、Linux で検出エージェントをアンインストールする方法について説明します。

yum パッケージマネージャを使用している場合にエージェントをアンインストールするには

- yum を使用している場合は、次のコマンドを使用してエージェントをアンインストールします。

```
rpm -e --nodeps aws-discovery-agent
```

apt-get パッケージマネージャーを使用している場合にエージェントをアンインストールするには

- 次のコマンドを使用して apt-get を使用している場合は、エージェントをアンインストールします。

```
apt-get remove aws-discovery-agent:i386
```

zypper パッケージマネージャーを使用している場合にエージェントをアンインストールするには

- 次のコマンドを使用して、zypper を使用する場合にエージェントをアンインストールします。

```
zypper remove aws-discovery-agent
```

Linux におけるエージェントのトラブルシューティング

Linux で検出エージェントをインストール中または使用中に問題が発生した場合には、ログと設定に関する次のガイダンスを参照してください。エージェントの潜在的な問題や Application Discovery Service への接続についてのトラブルシューティングでは、AWS Support からこれらのファイルが要求されることがあります。

- ログファイル

  検出エージェントのログファイルは次のディレクトリにあります。

  ```
  /var/log/aws/discovery/
  ```

  ログファイル名は、メインデーモン、自動アップグレード、またはインストール時のいずれかで生成されたかを示します。

- 設定ファイル

  検出エージェントバージョン 2.0.1617.0 以降の構成ファイルは、次のディレクトリにあります。

  ```
  /etc/opt/aws/discovery/
  ```

  2.0.1617.0 より前のバージョンの検出エージェントの構成ファイルは、次のディレクトリにあります。

  ```
  /var/opt/aws/discovery/
  ```

- 旧バージョンの検出エージェントを削除する手順については、「」を参照してください。検出エージェントのインストールの前提条件 (p. 8)。

Windows にエージェントをインストールする

Windows にエージェントをインストールするには、次の手順を実行します。必ず、お客様のことを確認します。Migration Hub ホームが設定されている場合、この手順を開始する前に設定されています。

データセンターに AWS アプリケーション検出エージェントをインストールするには

1. のダウンロードWindows エージェントのインストーラ Windows 内でインストーラーをダブルクリックして実行しないでください。

   > **Important**
   >
   > Windows 内でインストーラーをダブルクリックして実行すると、インストールは失敗します。エージェントのインストールはコマンドプロンプトからのみ可能です (インストーラーをダブルクリックしてしまった場合は、[プログラムの追加と削除] に移動し、エージェントをアンインストールしてから残りのインストール手順を続行する必要があります)。
   > Windows エージェントインストーラーがホスト上の Visual C + ランタイムのバージョンを検出しない場合、エージェントソフトウェアをインストールする前に Visual C + 2015—2019 ランタイムが自動的にインストールされます。

2. 管理者としてコマンドプロンプトを開き、インストールパッケージを保存した場所に移動します。
3. エージェントをインストールするには、次のインストール方法のいずれかを選択します。

目的	操作
検出エージェントのインストール	エージェントをインストールするには、次の例に示すように agent install コマンドを実行します。この例では、*your-home-region*をホームリージョンの名前に置き換えます。*aws-access-key-id*をアクセスキー ID に置き換え、*aws-secret-access-key*をシークレットアクセスキーに置き換えます。 オプションで、フォルダのパスを指定して、エージェントのインストール場所を設定できます。*C:\install-location*INSTALLLOCATIONパラメータの。たとえば、`INSTALLLOCATION="C:\install-location"` と指定します。結果のフォルダ階層は [INSTALLLOCATION パス]\ AWS ディスカバリーになります。デフォルトでは、インストール場所は`Program Files`folder 必要に応じて、を使用できます。`LOGANDCONFIGLOCATION`をクリックして、エージェントログフォルダと設定ファイルのデフォルトディレクトリ (ProgramData) を上書きします。作成されるフォルダ階層は`[LOGANDCONFIGLOCATION path]\AWS Discovery`。 `.\AWSDiscoveryAgentInstaller.exe REGION="your-home-region" KEY_ID="aws-access-key-id" KEY_SECRET="aws-secret-access-key" /quiet` デフォルトでは、エージェントは、アップデートが利用可能になると、自動的にダウンロードして適用します。 このデフォルト設定の使用が推奨されます。 ただし、エージェントがアップデートを自動的にダウンロードして適用しないようにするには、agent installコマンドを実行するときに、次のパラメータを含めます。`AUTO_UPDATE=false` **Warning** 自動アップグレードを無効にすると、最新のセキュリティパッチがインストールされなくなります。

目的	操作
（オプション）検出エージェントをインストールし、非透過プロキシを構成する	非透過プロキシを設定するには、次のパブリックプロパティを agent install コマンドに追加します。 • PROXY— プロキシホストの名前。 • SCHEME— プロキシスキーム • PROXY— プロキシポート番号 • PROXY— プロキシユーザー名 • PASSWORD— プロキシユーザーのパスワードです。 非透過プロキシプロパティを使用する agent install コマンドの例を以下に示します。 `.\AWSDiscoveryAgentInstaller.exe REGION="your-home-region" KEY_ID="aws-access-key-id" KEY_SECRET="aws-secret-access-key" PROXY_HOST="myproxy.mycompany.com" PROXY_SCHEME="https" PROXY_PORT="proxy-port-number" PROXY_USER="myusername" PROXY_PASSWORD="mypassword" /quiet` 認証が不要なプロキシの場合は、PROXY_USERおよびPROXY_PASSWORDプロパティ。install コマンドの例では、https。プロキシが HTTP を使用する場合は、http向けのPROXY_SCHEME値.

4. ネットワークからの発信接続が制限されている場合は、ファイアウォール設定を更新する必要があります。エージェントには、TCP ポート 443 を介した arsenal へのアクセスが必要です。着信ポートを開く必要はありません。

   たとえば、ホームリージョンがeu-central-1を使用する場合は、次のようにします。https://arsenal-discovery.*eu-central-1*.amazonaws.com:443

パッケージの署名と自動アップグレード

Windows Server 2008 以降では、Amazon は SHA256 証明書で Application Discovery Service エージェントインストールパッケージを暗号化して証明します。Windows Server 2008 SP2 での SHA2 署名付き自動更新では、ホストに SHA2 署名認証をサポートする修正プログラムがインストールされていることを確認します。マイクロソフトの最新のサポート修正プログラムWindows Server 2008 SP2 で SHA2 認証をサポートします。

> Note
>
> Windows 2003 用の SHA256 サポートの修正プログラムは、マイクロソフトから公開されなくなりました。これらの修正が Windows 2003 ホストにまだインストールされていない場合は、手動アップグレードが必要です。

アップグレードを手動で実行するには

1. のダウンロードWindows エージェントのアップデーター。

2. 管理者としてコマンドプロンプトを開きます。
3. アップデータが保存されているの場所に移動します。
4. 次のコマンドを実行します。

```
AWSDiscoveryAgentUpdater.exe /Q
```

Windows で検出エージェントプロセスを管理する

検出エージェントの動作は、Windows Server Manager Services コンソールを通じてシステムレベルで管理できます。次の表に管理方法を示します。

タスク	サービス名	サービス状況/アクション
エージェントが実行されていることを確認	AWS検出エージェント AWS検出アップデーター	開始
エージェントの開始	AWS検出エージェント AWS検出アップデーター	[Start (開始)] を選択
エージェントの停止	AWS検出エージェント AWS検出アップデーター	[Stop (停止)] を選択
エージェントの再起動	AWS検出エージェント AWS検出アップデーター	[Restart (再起動)] を選択

Windows で検出エージェントをアンインストールするには

1. Windows でコントロールパネルを開きます。
2. [プログラム] を選択します。
3. [プログラムと機能] を選択します。
4. SelectAWS検出エージェント。
5. [アンインストール] を選択します。

> **Note**
>
> アンインストール後にエージェントを再インストールするように選択した場合は、/repairおよび/norestartオプション。

```
.\AWSDiscoveryAgentInstaller.exe REGION="your-home-region" KEY_ID="aws-access-key-id" KEY_SECRET="aws-secret-access-key" /quiet /repair /norestart
```

Windows で検出エージェントをコマンドラインでアンインストールするには

1. 右クリックを起動。
2. 選択コマンドプロント。
3. Windows で検出エージェントをアンインストールするには、次のコマンドを使用します。

```
wmic product where name='AWS Discovery Agent' call uninstall
```

Windows におけるエージェントのトラブルシューティング

インストール中あるいは Windows で AWS アプリケーション検出エージェントを使用中に問題が発生した場合には、ログと設定に関する次のガイダンスを参照してください。AWS Supportは、エージェントや Application Discovery Service への接続についてのトラブルシューティングでは、これらのファイルを要求することがあります。

- インストールログ記録

  agent install コマンドが失敗したと表示される場合があります。たとえば、Windows Services Manager の失敗により、検出サービスは作成されていないと表示される場合があります。このような場合は、コマンドに /log install.log を追加して、詳細なインストールログを生成します。
- 運用ログ

  Windows Server 2008 以降の場合、エージェントログファイルは次のディレクトリにあります。

  ```
  C:\ProgramData\AWS\AWS Discovery\Logs
  ```

  Windows Server 2003 の場合、エージェントログファイルは次のディレクトリにあります。

  ```
  C:\Documents and Settings\All Users\Application Data\AWS\AWSDiscovery\Logs
  ```

  ログファイル名は、メインサービス、自動アップグレード、またはインストール時のいずれかで生成されたかを示します。

- 設定ファイル

  Windows Server 2008 以降の場合、エージェント設定ファイルは次の場所にあります。

  ```
  C:\ProgramData\AWS\AWS Discovery\config
  ```

  Windows Server 2003 の場合、エージェント設定ファイルは次の場所にあります。

  ```
  C:\Documents and Settings\All Users\Application Data\AWS\AWS Discovery\config
  ```

- 以前のバージョンの検出エージェントを削除する手順については、「」を参照してください。検出エージェントのインストールの前提条件 (p. 8)。

エージェントデータ収集

Discovery Agent を展開して構成した後、データ収集が停止した場合は、再起動できます。データ収集を開始または停止するには、コンソールを使用するか、を通じて API コールを行います。AWS CLI。これらの方法の両方の方法については、以下の手順で説明します。

Using the Migration Hub Console

次の手順は、Discovery Agent データ収集プロセスを開始または停止する方法を示しています。データコレクターページMigration Hub。

データ収集を開始または停止するには

1. ナビゲーションペインで、[Data Collectors (データコレクタ)] を選択します。
2. [Agents (エージェント)] タブを選択します。
3. 起動または停止するエージェントのチェックボックスをオンにします。

   Tip

   複数のエージェントをインストールしている場合でも、データの収集を特定のホストに限定するには、Hostname列で、エージェントがインストールされているホストを特定します。
4. [Start data collection (データ収集の開始)] または [Stop data collection (データ収集の停止)] を選択します。

Using the AWS CLI

探索エージェントのデータ収集プロセスを開始または停止するには、AWS CLIをインストールするには、最初にをインストールする必要がありますAWS CLIを選択し、選択した CLI を使用する必要がありますMigration Hub ホーム。

をインストールするにはAWS CLIデータ収集を開始または停止するには

1. OS のタイプ (Windows または Mac/Linux) に適切な AWS CLI をインストールします (まだインストールしていない場合)。フレームワークの使用の詳細については、AWS Command Line Interfaceユーザーガイド手順については、「」
2. コマンドプロンプト (Windows) またはターミナル (MAC/Linux) を開きます。

   a. `aws configure` を入力して、[Enter] を押します。
   b. AWS アクセスキー ID と AWS シークレットアクセスキーを入力します。
   c. デフォルトリージョン名に、*`us-west-2`* などホームリージョンを入力します。(この例では、ホームリージョンが `us-west-2` であると仮定しています)。
   d. デフォルトの出力形式として「`text`」と入力します。
3. データ収集を停止または開始するエージェントの ID を確認するには、次のコマンドを入力します。

```
aws discovery describe-agents
```

4. エージェントによるデータ収集を開始するには、次のコマンドを入力します。

```
aws discovery start-data-collection-by-agent-ids --agent-ids <agent ID>
```

エージェントによるデータ収集を停止するには、次のコマンドを入力します。

```
aws discovery stop-data-collection-by-agent-ids --agent-ids <agent ID>
```

AWS エージェントレス検出コネクタ

Agentless Discovery ではAWS検出コネクタ。-AWSDiscovery Connector は、VMware 仮想マシン (VM) に関する情報のみを収集できる VMware アプライアンスです。Open Virtualization Archive (OVA) ファイルを使用して VMware vCenter Server 環境に VM として Discovery Connector をインストールします。Discovery Connectorは、オペレーティングシステムの種類を問わず、VMware メタデータに依存してサーバー情報を収集するため、最初のオンプレミスインフラストラクチャの評価所要時間を最小限に抑えます。

検出コネクタを展開する前に、[Discovery Connector を選択します。Migration Hub。コネクタをホームリージョンに登録する必要があります。探索コネクタをデプロイして設定すると、Application Discovery Service エンドポイントに登録され、定期的に (約 60 分間隔) でサーバーに ping を発行して設定情報を確認します。

- たとえば、次の場合`eu-central-1`がホームリージョンの場合、が`arsenal-discovery.eu-central-1.amazonaws.com`Application Discovery Service
- または、us-west-2 を除く他のすべてのリージョンで、必要に応じてホームリージョンが置き換えられます。
- もし`us-west-2`がホームリージョンの場合、が`arsenal.us-west-2.amazonaws.com`Application Discovery Service

仕組み

登録後、コネクタは VMware vCenter Server に接続され、指定の vCenter に管理されるすべての VM とホストに関するデータを収集します。収集されたデータは、Secure Sockets Layer (SSL) 暗号化を使用して Application Discovery Service に送信されます。コネクタは、コネクタの新しいバージョンが利用可能になると自動的にアップグレードするように設定されています。この設定はいつでも変更できます。

トピック

検出コネクタによって収集されたデータ

検出コネクタは、VMware vCenter Server ホストおよび仮想マシンに関する情報を収集します。ただし、このデータをキャプチャできるのは、VMware vCenter Server ツールがインストールされている場合に限ります。使用している AWS アカウントにこのタスクに必要なアクセス許可があることを確認するには、「Application Discovery Service での AWS 管理 (事前定義) ポリシー (p. 79)」を参照してください。

次に、Discovery Connector で収集される情報のインベントリを示します。

検出コネクタで収集されるデータのテーブル凡例:

- 収集されたデータは、特に断らない限り、キロバイト (KB) 単位です。
- Migration Hub コンソールの対応するデータはメガバイト (MB) 単位です。

- アスタリスク (*) が付いているデータフィールドは、コネクタの API エクスポート関数から生成される .csv ファイルでのみ使用できます。
- ポーリング間隔は約 60 分です。
- データフィールドは二重アスタリスク (**) で表され、現在 null 値を返します。

データフィールド	説明
applicationConfigurationId*	VM をグループ化する移行アプリケーションの ID
avgCpuUsagePct	ポーリング間隔中の平均 CPU 使用率
avgDiskBytesReadPerSecond	ポーリング間隔中にディスクから読み取られた平均バイト数
avgDiskBytesWrittenPerSecond	ポーリング間隔中にディスクに書き込まれた平均バイト数
avgDiskReadOpsPerSecond**	1 秒あたりの null の読み取り I/O オペレーションの平均数
avgDiskWriteOpsPerSecond**	1 秒あたりの書き込み I/O オペレーションの平均数
avgFreeRAM	平均空き RAM (MB 単位)
avgNetworkBytesReadPerSecond	1 秒あたりに読み取られたバイトスループットの平均値
avgNetworkBytesWrittenPerSecond	1 秒あたりに書き込まれたバイトスループットの平均値
configId	検出した VM に Application Discovery Service から割り当てられたID
configType	検出したリソースのタイプ
connectorId	Discovery Connector 仮想アプライアンスの ID
cpuType	VM の場合は CPU、ホストの場合は実際のモデル
datacenterId	vCenter の ID
hostId*	VM ホストの ID
hostName	仮想化ソフトウェアを実行しているホストの名前
hypervisor	ハイパーバイザーのタイプ
id	サーバーの ID
lastModifiedTimeStamp*	データのエクスポート前の直近にデータを収集した日時
macAddress	VM の MAC アドレス
manufacturer	仮想化ソフトウェアのメーカー
maxCpuUsagePct	ポーリング期間の最大 CPU 使用率
maxDiskBytesReadPerSecond	ポーリング期間のディスクから読み取られた最大バイト数

データフィールド	説明
maxDiskBytesWrittenPerSecond	ポーリング期間のディスクに書き込まれた最大バイト数
maxDiskReadOpsPerSecond**	読み取り I/O オペレーションの最大数 (1 秒あたり)
maxDiskWriteOpsPerSecond**	書き込み I/O オペレーションの最大数 (1 秒あたり)
maxNetworkBytesReadPerSecond	読み取られたバイトスループットの最大値 (1 秒あたり)
maxNetworkBytesWrittenPerSecond	書き込まれたバイトスループットの最大値 (1 秒あたり)
memoryReservation*	VM へのメモリの超過割り当てを避けるための制限
moRefId	vCenter マネージド型オブジェクトの一意のリファレンス ID
name*	VM またはネットワークの名前 (ユーザー指定)
numCores	CPU 内の独立した処理装置の数
numCpus	VM の CPU の数
numDisks**	VM のディスクの数
numNetworkCards**	VM のネットワークカードの数
osName	VM のオペレーティングシステムの名前
osVersion	VM のオペレーティングシステムのバージョン
portGroupId*	VLAN のメンバーポートのグループの ID
portGroupName*	VLAN のメンバーポートのグループの名前
powerState*	電力のステータス
serverId	検出した VM に Application Discovery Service から割り当てられたID
smBiosId*	システム管理 BIOS の ID//バージョン
state*	Discovery Connector 仮想アプライアンスのステータス
tagKey	サーバーに関するカスタムデータまたはメタデータを保存するためのユーザー定義キー
tagValue	サーバーに関するキーのカスタムデータまたはメタデータを詳細に定義するためのユーザー定義値
toolsStatus	VMware ツールの運用状態 (詳細なリストについては、「データコレクタの表示とソート (p. 55)」を参照)
totalDiskSize	ディスクの合計容量 (MB 単位)
totalRAM	VM で使用可能な RAM の合計量 (MB)

データフィールド	説明
type	ホストのタイプ
vCenterId	VM 固有の ID 番号
vCenterName*	vCenter ホストの名前
virtualSwitchName*	仮想スイッチの名前
vmFolderPath	VM ファイルのディレクトリパス
vmName	仮想マシンの名前

Discovery Connector

ダウンロード、セットアップ、およびデータ収集の開始

エージェントレス検出を設定するには、仮想アプライアンスであるDiscovery Connector (Discovery Connector) をオンプレミス環境の VMware vCenter Server ホストにダウンロードしてデプロイする必要があります。検出コネクタは、Open Virtualization Archive (OVA) ファイルであり、オンプレミスの VMware 環境にインストールする必要があります。

> Reminder
>
> Discovery Connector は、VMware vCenter バージョン V5.5、V6、V6.5、および V6.7 をサポートしています。

このセクション以降では、Discovery Connector を使用してダウンロード、デプロイ、設定、およびデータ収集の開始を行うための手順を示します。

Discovery Connector OVA ファイルをダウンロードして、そのチェックサムを検証するには

1. VMware 管理者として vCenter にサインインし、Discovery Connector OVA ファイルをダウンロードする先のディレクトリに切り替えます。
2. のダウンロード検出コネクタOVA。
3. システム環境で使用するハッシュアルゴリズムに応じて、MD5 または SHA256 をダウンロードし、チェックサム値が含まれているファイルを取得します。この値を使用して、前のステップでダウンロードした `AWSDiscoveryConnector.ova` ファイルを検証します。
4. Linux のバリエーションに応じて、適切なバージョンの MD5 コマンドまたは SHA256 コマンドを実行して、`AWSDiscoveryConnector.ova` ファイルの暗号署名が、ダウンロードした各 MD5 / SHA256 ファイルの値と一致することを確認します。

```
$ md5sum AWSDiscoveryConnector.ova
```

```
$ sha256sum AWSDiscoveryConnector.ova
```

検出コネクタの展開

前のセクションでは、AWS Agentless Discovery Connector をOpen Virtualization Archive (OVA) ファイルにダウンロードしました。このセクションでは、ダウンロードした Discovery Connector の仕様と、VMware 環境に展開する方法を示します。

検出コネクタ仮想マシンの仕様

- オペレーティングシステム—FreeBSD 11 (64 ビット)
- RAM—2 GB
- ディスクストレージ

  が vCenter で仮想マシンとして OVA を展開する場合、vCenter クライアントには次の 2 つのプロビジョニングオプションがあります。
  - シンプロビジョニング— 約 7.8 GB
  - シックプロビジョニング— 約 299.0 GB (推奨オプション)

VMware 環境に検出コネクタ OVA ファイルをデプロイするステップを次に示します。

検出コネクタを展開するには

1. VMware 管理者として vCenter にサインインします。
2. 選択ファイル,OVF テンプレートのデプロイ前のセクションでダウンロードした OVA ファイルを選択し、ウィザードを閉じます。
3. [Disk Format] ページで、いずれかのシックプロビジョニングディスクタイプを選択します。最高のパフォーマンスと信頼性を備えているため、[Thick Provision Eager Zeroed] を選択することをお勧めします。ただし、ディスクのフォーマットに数時間かかります。[Thin Provision] は選択しないでください。このオプションではデプロイは速くなりますが、ディスクパフォーマンスは大幅に低下します。詳細については、VMware のドキュメントでサポートされている仮想ディスクのタイプを確認してください。
4. vSphere クライアントのインベントリツリーで、新しくデプロイされたテンプレートのコンテキスト (右クリック) メニューを見つけて開き、[Power]、[Power On] の順に選択します。
5. テンプレートのコンテキスト (右クリック) メニューを再度開き、[Open Console] を選択します。コンソールに、コネクタコンソールの IP アドレスが表示されます。IP アドレスをメモしておきます。コネクタのセットアッププロセスを完了するために必要になります。

を設定するAWS検出コネクタ

セットアッププロセスを完了するには、次の手順と、必要に応じてオプションのコネクタ設定タスクを実行します。

> Reminder
>
> 手順を開始する前に、Migration Hubまだ行っていない場合は、[] を選択します。

コンソールを使用してコネクタを設定するには

1. ウェブブラウザで、アドレスバーに次の URL を入力します。**`https://`***`<ip_address>`***`/`ここで、とします。ip_addressは、以前に保存したコネクタコンソールの IP アドレスです。
2. 選択今すぐ始める次に、指示に従って以下の設定ページを完了します。ライセンス契約,パスワードを作成する, およびネットワーク情報。
3. リポジトリの []ログのアップロードとアップグレードページで、[] を選択することをお勧めします。ログを自動的にアップロードする。自動アップロードによってログが利用可能になった場合、AWS はコネクタの問題をトラブルシューティングするのに役立ちます。

   [AWS Agentless Discovery Connector の自動アップグレード] 機能がデフォルトで有効になっています。コネクタの最新バージョンを実行すると、最新のセキュリティパッチがインストールされます。自動アップグレードはいつでも無効にできます。「での自動アップグレードの無効化AWS検出コネクタ (p. 27)」を参照してください。

4. リポジトリの []検出コネクタのセットアップページで、以下のステップを実行します。

   a. [vCenter 認証情報の設定] で、次の操作を行います。

      i. [vCenter Host] には、VMware vCenter サーバーホストのホスト名または IP アドレスを入力します。

      ii. [vCenter Username] には、コネクタが vCenter との通信に使用するローカルまたはドメインユーザーの名前を入力します。ドメインユーザーの場合、domain\username または username@domain 形式を使用します。

      iii. [vCenter Password] で、ローカルユーザーまたはドメインユーザーのパスワードを入力します。

      iv. vCenter で SSL 証明書検証をバイパスするには、[Ignore security certificate] を選択します。

   b. []設定AWS認証情報に、割り当てられた IAM ユーザーの認証情報を入力します。`AWSAgentlessDiscoveryService`IAM 管理ポリシー。管理ポリシーの詳細については、「Application Discovery Service での AWS 管理 (事前定義) ポリシー (p. 79)」を参照してください。

      続いて、[次へ] を選択します。

   c. []データをパブリッシュする場所の構成で、ローカルファイルまたは特定の AWS リージョンエンドポイントへの発行を選択します。ローカルファイルに公開することを選択した場合、Discovery Connector はオンプレミスサーバーに関するデータを AWS に送信しません。ただし、Discovery Connector は、コネクタ自体に関するデータを引き続き AWS に送信します。

      次に [] を選択します。次AWS Agentless Discovery Connector コンソールに戻ります。

次のトピックでは、オプションのコネクタ設定タスクについて説明します。

トピック

コネクタの静的 IP アドレスの設定

環境で静的 IP アドレスを使用する必要がある場合は、次の手順に従ってください。

コネクタの静的 IP アドレスを設定するには

1. コネクタの仮想マシンコンソールを開き、パスワード **ec2pass** で **ec2-user** としてログインします。求められたら、新しいパスワードを指定します。
2. コマンド **sudo setup.rb** を実行し、設定メニューを表示するよう求められたら、ec2-user のパスワードを入力します。
3. 「**2**」と入力して [Reconfigure network settings (ネットワーク設定の再構成)] を選択します。現在のネットワーク情報と、ネットワーク設定を変更するサブメニューが表示されます。
4. 前のステップで生成されたサブメニューで、「**2**」と入力し、[Set up a static IP (静的 IP のセットアップ)] を選択します。これにより、ネットワーク設定を指定するフォームが表示されます。

   - 各フィールドに適切な値を入力し、Enter キーを押します。次のような出力が表示されます。ここで nnn.nnn.nnn.nnn には、各フィールドに入力したアドレス番号が入ります。

```
Setting up static IP:
     1. Enter IP address: <nnn.nnn.nnn.nnn>
     2. Enter netmask: <nnn.nnn.nnn.nnn>
     3. Enter gateway: <nnn.nnn.nnn.nnn>
```

```
     4. Enter DNS 1: <nnn.nnn.nnn.nnn>
     5. Enter DNS 2: <nnn.nnn.nnn.nnn>

Static IP address configured.
```

データ収集の範囲の制御

vCenter ユーザーは、Application Discovery Service を使用してインベントリを行うために、各 ESX ホストまたは VM に対する読み取り専用アクセス許可が必要です。許可設定を使用すると、データ収集に組み込まれるホストと VM を制御できます。現在の vCenter のすべてのホストと仮想マシンをインベントリ対象にするか、ケースバイケースで許可を付与することができます。

> Note
>
> セキュリティのベストプラクティスとして、Discovery Connector の vCenter ユーザーに余分なアクセス許可を付与しないようお勧めします。

次の手順では、細分化がおおまかなものから細かいものまでの設定シナリオを順に説明します。

に関するデータを検出するにはall現在の vCenter の ESX ホストと VM

1. VMware vSphere クライアントでは、[vCenter] を選択してから [Hosts and Clusters] または [VMs and Templates] を選択します。
2. [Manage]、[Permissions] の順に選択します。
3. vCenter ユーザーを選択して右クリックでコンテキストメニューを開き、[Change Role] を選択します。
4. [Assigned Role] ペインで [Read-only] を選択します。
5. [Propagate to children]、[OK] を選択します。

に関するデータを検出するにはspecificESX ホストとallその子オブジェクトの

1. VMware vSphere クライアントでは、[vCenter] を選択してから [Hosts and Clusters] または [VMs and Templates] を選択します。
2. [Related Objects]、[Hosts] の順に選択します。
3. ホスト名を右クリックしてコンテキストメニューを開き、[All vCenter Actions]、[Add Permission] の順に選択します。
4. [Add Permission] で、vCenter ユーザーをホストに追加します。[Assigned Role] では、[Read-only] を選択します。
5. [Propagate to children]、[OK] を選択します。

に関するデータを検出するspecificESX ホストまたは子 VM

1. VMware vSphere クライアントでは、[vCenter] を選択してから [Hosts and Clusters] または [VMs and Templates] を選択します。
2. [Related Objects] を選択します。
3. [Hosts] (vCenter に認識される ESX ホストのリストを表示) または [Virtual Machines] (すべてのホスト ESX ホストにわたる VM のリストを表示) を選択します。
4. ホストあるいは VM 名を右クリックしてコンテキストメニューを開き、[All vCenter Actions]、[アクセス許可の追加] の順に選択します。
5. [Add Permission] で、vCenter ユーザーをホストまたは VM に追加します。[Assigned Role] では、[読み取り専用] を選択します。

6. [OK] を選択します。

   Note

   [Propagate to children] を選択した場合でも引き続き、ケースバイケースで、ESX ホストと VM から読み取り専用アクセス許可を削除することができます。このオプションは、他の ESX ホストや VM に適用される、継承された許可には影響しません。

での自動アップグレードの無効化AWS検出コネクタ

最新バージョンのを確実に実行するにはAWSDiscovery Connector の場合、自動アップグレード機能はインストール時にデフォルトで有効になります。ただし、以下に示すように、自動アップグレード機能を無効にすることもできます。

自動アップグレードを無効にするには

1. ウェブブラウザで、アドレスバーに次の URL を入力します。**`https://`***`<ip_address>`***`/`**ここで、とします。ip_addressの IP アドレスです。AWS検出コネクタ。
2. 検出コネクタコンソールで、アクション] で、自動アップグレードの無効化。

   Warning

   自動アップグレードを無効にすると、最新のセキュリティパッチがインストールされなくなります。

検出コネクタのデータ収集

VMware 環境にDiscovery Connector をデプロイして構成した後、データ収集が停止した場合は、再起動できます。データ収集を開始または停止するには、コンソールを使用するか、を使用して API コールを行います。AWS CLI。これらの方法のどちらも、以下の手順で説明します。

Using the Migration Hub Console

次の手順は、Discovery Connector データ収集プロセスを開始または停止する方法を示しています。データコレクターページMigration Hub。

データ収集を開始または停止するには

1. ナビゲーションペインで、[Data Collectors (データコレクタ)] を選択します。
2. [Connectors (コネクタ)] タブを選択します。
3. 開始または停止するコネクタのチェックボックスをオンにします。
4. [Start data collection (データ収集の開始)] または [Stop data collection (データ収集の停止)] を選択します。

   Note

   コネクタでデータ収集を開始した後にインベントリ情報が表示されない場合は、コネクタが vCenter Server に登録済みであることを確認します。

Using the AWS CLI

探索コネクタのデータ収集プロセスを開始するには、AWS CLIとすると、AWS CLI最初にを環境にインストールし、選択したを使用するように CLI を設定する必要があります。Migration Hub。

AWS CLI をインストールしてデータ収集を開始するには

1. オペレーティングシステム (Linux、macOS、または Windows) 用の AWS CLI をインストールします。フレームワークの使用の詳細については、AWS Command Line Interfaceユーザーガイド手順については、「」
2. コマンドプロンプト (Windows) またはターミナル (Linux/macOS) を開きます。
   a. `aws configure` を入力して、[Enter] を押します。
   b. AWS アクセスキー ID と AWS シークレットアクセスキーを入力します。
   c. デフォルトリージョン名に、`us-west-2` などホームリージョンを入力します。
   d. デフォルトの出力形式として「`text`」と入力します。
3. データ収集を開始または停止するコネクタの ID を確認するには、次のコマンドを入力してコネクタの ID を確認します。

```
aws discovery describe-agents --filters
 condition=EQUALS,name=hostName,values=connector
```

4. コネクタによるデータ収集を開始するには、次のコマンドを入力します。

```
aws discovery start-data-collection-by-agent-ids --agent-ids <connector ID>
```

   Note

   コネクタでデータ収集を開始した後にインベントリ情報が表示されない場合は、コネクタが vCenter Server に登録済みであることを確認します。

   コネクタによるデータ収集を停止するには、次のコマンドを入力します。

```
aws discovery stop-data-collection-by-agent-ids --agent-ids <connector ID>
```

検出コネクタのトラブルシューティング

このセクションには、Application Discovery Service に関する既知の問題のトラブルシューティングに役立つトピックが含まれています。

検出コネクタが到達できない問題を修正するAWSセットアップ中に

設定時にAWSエージェントレス検出コネクタは、コンソールに、次のエラーメッセージが表示されることがあります。

到達できませんでしたAWS

AWSに到達できません (接続リセット)。ネットワークとプロキシの設定を確認してください。

このエラーは、検出コネクタが HTTPS 接続を確立するために失敗したために発生します。`ec2.amazonaws.com`セットアッププロセス中に。接続を確立できない場合、検出コネクタの構成は失敗します。

への接続を修正するにはAWS

1. ファイアウォールが送信トラフィックをブロックしているかどうかを確認します。`ec2.amazonaws.com`。ブロックされている場合は、ブロックを解除します。ファイアウォールを更新した後、検出コネクタを再構成します。

2. ファイアウォールを更新しても接続の問題が解決しない場合は、コネクタ仮想マシンにアウトバウンドネットワーク接続があることを確認します。仮想マシンにアウトバウンド接続がある場合は、`aws.amazon.com`および`ec2.amazonaws.com`を実行してTelnetポート80および443に。

```
telnet ec2.amazonaws.com 80
```

3. 仮想マシンからのアウトバウンド接続が有効になっている場合は、AWS サポートを参照して、さらにトラブルシューティングを行ってください。

異常なコネクタの修正

各検出コネクタのHealth 情報は、データコレクターページMigration Hub。[Health (ヘルス)] ステータスが [Unhealthy (異常)] のコネクタを検索すると、問題のあるコネクタを特定できます。次の手順では、コネクタコンソールにアクセスしてヘルスの問題を特定する方法の概要を示します。

コネクタコンソールへのアクセス

1. Web ブラウザーで Migration Hub コンソールを開き、[] を選択します。データコレクター左側のナビゲーションから「」を選択します。
2. [] からコネクタタブで、[IP addressの健全性ステータスを持つ各コネクタの異常。
3. コネクタ仮想マシンに接続できる任意のコンピュータでブラウザを開き、コネクタコンソールの URL を入力します。`https://`*`ip_address_of_connector`*ここで、とします。`ip_address_of_connector`は、正常でないコネクタの IP アドレスです。
4. コネクタの構成時に設定されたコネクタ管理コンソールのパスワードを入力します。

コネクタコンソールにアクセスすると、異常なステータスを解決するためのアクションを実行できます。ここで選択することができます情報の表示forvCenter の接続をクリックすると、診断メッセージを含むダイアログボックスが表示されます。[View Info (情報を表示)] リンクは、バージョン 1.0.3.12 以降のコネクタでのみ使用できます。

ヘルスの問題を修正した後、コネクタは vCenter サーバーとの接続を再確立し、コネクタのステータスが [HEALTHY (正常)] ステータスに変わります。問題が解決しない場合は、AWS サポートまでお問い合わせください。

異常なコネクタの最も一般的な原因は、IP アドレスの問題と認証情報の問題です。以下のセクションは、これらの問題を解決し、コネクタを正常な状態に戻すのに役立ちます。

トピック

- IP アドレスの問題 (p. 29)
- 認証情報の問題 (p. 30)

IP アドレスの問題

コネクタのセットアップ中に提供された vCenter エンドポイントの形式が正しくないか、無効な場合、または vCenter サーバーが現在ダウンしていて到達不可能な場合、コネクタが異常なステータスになる可能性があります。この場合、[情報の表示forvCenter の接続をクリックすると、「vCenter サーバの動作ステータスを確認するか、[設定の編集] を選択して vCenter エンドポイントを更新します。

次の手順は、IP アドレスの問題を解決するのに役立ちます。

1. コネクタコンソール (`https://`*`ip_address_of_connector`*) から、[Edit Settings (設定の編集)] を選択します。
2. 左側のナビゲーションから、[] を選択します。ステップ 5: 検出コネクタのセットアップ。

3. [Configure vCenter credentials (vCenter 認証情報の設定)] で、[vCenter Host (vCenter ホスト)] の IP アドレスをメモします。
4. のような別のコマンドラインツールを使用するpingまたはtraceroute関連付けられた vCenter サーバーがアクティブであり、IP がコネクタ VM から到達可能であることを確認します。

   - IP アドレスが正しくなく、vCenter サービスがアクティブな場合は、コネクタコンソールで IP アドレスを更新し、[Next (次へ)] を選択します。
   - IP アドレスは正しいが、vCenter サーバーが非アクティブの場合は、アクティブにします。
   - IP アドレスが正しく、vCenter サーバーがアクティブな場合は、ファイアウォールの問題により侵入ネットワーク接続がブロックされているかどうかを確認します。ブロックされている場合は、コネクタ VM からの着信接続を許可するようにファイアウォール設定を更新します。

認証情報の問題

コネクタのセットアップ中に提供された vCenter ユーザーの認証情報が無効であるか、vCenter の読み取りおよび表示アカウント権限がない場合、コネクタは異常な状態になる可能性があります。この場合、[情報の表示forvCenter の接続をクリックすると、「[設定の編集] を選択して、アカウントの vCenter ユーザー名とパスワードを読み取り、表示権限で更新します」というメッセージが表示されたダイアログボックスが表示されます。

次の手順は、認証情報の問題を解決するのに役立ちます。前提条件として、vCenter サーバーでアカウントの読み取り権限と表示権限を持つ vCenter ユーザーを作成していることを確認します。

1. コネクタコンソール (`https://ip_address_of_connector`) から、[Edit Settings (設定の編集)] を選択します。
2. 左側のナビゲーションから、[] を選択します。ステップ 5: 検出コネクタのセットアップ。
3. [Configure vCenter credentials (vCenter 認証情報の設定)] で、読み取り権限と表示権限を持つ vCenter ユーザーの認証情報を指定して、[vCenter Username (vCenter ユーザ名)] と [vCenter Password (vCenter パスワード)] を更新します。
4. [Next (次へ)] を選択して設定を完了します。

スタンドアロン ESX ホストのサポート

Discovery Connector は、スタンドアロンの ESX ホストをサポートしていません。ESX ホストは vCenter Server インスタンスの一部であることが必要です。

コネクタの問題に対する追加のサポートの取得

ヘルプが必要な問題が発生した場合は、AWS サポートにお問い合わせください。AWS サポートから、コネクタのログを送信するよう求められる場合があります。ログを取得するには、次の操作を行います。

- ログインし直すAWSエージェントレス検出コネクタコンソール (設定 (p. 24)) を選択し、ログバンドルのダウンロード。
- ログバンドルのダウンロードが完了したら、AWS サポートの指示に従って送信します。

Migration Hub インポート

Migration Hub のインポートを使用すると、Discovery Connector や Discovery Agent を使用しなくても、オンプレミス環境の詳細を Migration Hub に直接インポートできるため、インポートしたデータから直接移行の評価と計画を実行できます。デバイスをアプリケーションとしてグループ化し、それらの移行ステータスを追跡することもできます。

インポートリクエストを開始するには

- 特別な形式のカンマ区切り値 (CSV) インポートテンプレートをダウンロードします。
- 既存のオンプレミスサーバーデータを入力します。
- 移行ハブコンソールを使用して、Migration Hub にアップロードします。AWS CLIまたは AWS SDK のいずれかを使用します。

複数のインポートリクエストを送信できます。各リクエストは順番に処理されます。インポートリクエストのステータスは、コンソールまたはインポート API を使用していつでも確認できます。

インポートリクエストが完了したら、インポートされた各レコードの詳細を表示することができます。Migration Hub コンソール内から、使用状況データ、タグ、およびアプリケーションマッピングを直接表示します。インポート中にエラーが発生した場合は、成功したレコードと失敗したレコードの数や、失敗した各レコードのエラー詳細を確認できます。

エラー処理: エラーログと失敗したレコードのファイルを CSV ファイルとして圧縮アーカイブにダウンロードするためのリンクが用意されています。これらのファイルを使用して、エラーを修正してから、インポートリクエストを再送信します。

インポートされたレコード、インポートされたサーバー、および保持できる削除されたレコードの数には、制限が適用されます。詳細については、「AWS Application Discovery Service のクォータ (p. 95)」を参照してください。

サポートされているインポートファイルフィールド

Migration Hub のインポートを使用すると、任意のソースからデータをインポートできます。提供されるデータは、CSV ファイルでサポートされている形式である必要があります。また、データには、サポートされている範囲を持つサポートされているフィールドのみが含まれている必要があります。

次の表のインポートフィールド名の横にあるアスタリスクは、必須フィールドであることを示しています。インポートファイルの各レコードには、サーバーまたはアプリケーションを一意に識別するために、必須フィールドが 1 つ以上含まれている必要があります。必須フィールドが 1 つもないレコードはインポートできません。

> **Note**
>
> VMware.MoRefId または VMWare.VCenterId を使用してレコードを識別している場合は、同じレコードに両方のフィールドが必要です。

インポートフィールド名	説明	例
ExternalId*	各レコードに一意であることをマークすることができるカスタム識別子。たとえば、[ExternalId] は、データセンター内のサーバーのインベントリ ID を指します。	Inventory Id 1 Server 2 CMBD Id 3

インポートフィールド名	説明	例
SMBiosId	システム管理 BIOS (SMBIOS) ID。	
IPAddress*	サーバーの IP アドレスのカンマ区切りリスト (引用符で囲む)。	192.0.0.2 "10.12.31.233, 10.12.32.11"
MACAddress*	サーバーの MAC アドレスのカンマ区切りリスト (引用符で囲む)。	00:1B:44:11:3A:B7 "00-15-E9-2B-99-3C, 00-14-22-01-23-45"
HostName*	サーバーのホスト名。この値には完全修飾ドメイン名 (FQDN) を使用することをお勧めします。	ip-1-2-3-4 localhost.domain
VMware.MoRefId*	マネージド型オブジェクトのリファレンス ID。VMware.VCenterId で指定する必要があります。	
VMware.VCenterId*	仮想マシンの一意の ID。VMware.MoRefId で指定する必要があります。	
CPU.NumberOfProcessors	CPU の数。	4
CPU.NumberOfCores	物理コアの合計数。	8
CPU.NumberOfLogicalCores	サーバー内のすべての CPU で同時に実行できるスレッドの合計数。一部の CPU は、単一の CPU コアにおける複数のスレッドの同時実行をサポートしています。このような場合、この数は物理 (または仮想) コアの数よりも大きくなります。	16
OS.Name	オペレーティングシステムの名前。	Linux Windows.Hat
OS.Version	オペレーティングシステムのバージョン。	16.04.3 NT 6.2.8
VMware.VMName	仮想マシンの名前。	Corp1
RAM.TotalSizeInMB	サーバーで使用可能な合計 RAM (MB)。	64 128
RAM.UsedSizeInMB.Avg	サーバーで使用されている RAM の平均容量 (MB)。	64 128

インポートフィールド名	説明	例
RAM.UsedSizeInMB.Max	サーバーで使用できる RAM の最大容量 (MB)。	64 128
CPU.UsagePct.Avg	検出ツールでデータを収集していたときの平均 CPU 使用率。	45 23.9
CPU.UsagePct.Max	検出ツールでデータを収集していたときの最大 CPU 使用率。	55.34 24
DiskReadsPerSecondInKB.Avg	1 秒あたりのディスク読み取りの平均数 (KB)。	1159 84506
DiskWritesPerSecondInKB.Avg	1 秒あたりのディスク書き込みの平均数 (KB)。	199 6197
DiskReadsPerSecondInKB.Max	1 秒あたりのディスク読み取りの最大数 (KB)。	37892 869962
DiskWritesPerSecondInKB.Max	1 秒あたりのディスク書き込みの最大数 (KB)。	18436 1808
DiskReadsOpsPerSecond.Avg	1 秒あたりのディスク読み取り操作の平均回数。	45 28
DiskWritesOpsPerSecond.Avg	1 秒あたりのディスク書き込み操作の平均回数。	8 3
DiskReadsOpsPerSecond.Max	1 秒あたりのディスク読み取りオペレーションの最大数。	1083 176
DiskWritesOpsPerSecond.Max	1 秒あたりのディスク書き込みオペレーションの最大数。	535 71
NetworkReadsPerSecondInKB.Avg	1 秒あたりのネットワーク読み取りオペレーションの平均数 (KB)。	45 28
NetworkWritesPerSecondInKB.Avg	1 秒あたりのネットワーク書き込みオペレーションの平均数 (KB)。	8 3
NetworkReadsPerSecondInKB.Max	1 秒あたりのネットワーク読み取りオペレーションの最大数 (KB)。	1083 176
NetworkWritesPerSecondInKB.Max	1 秒あたりのネットワーク書き込みオペレーションの最大数 (KB)。	535 71

インポートフィールド名	説明	例
アプリケーション	このサーバーを含むアプリケーションのカンマ区切りリスト (引用符で囲む)。この値には、既存のアプリケーションや、インポート時に作成された新規アプリケーションを含めることができます。	Application1 "Application2, Application3"
タグ	name:value 形式のタグのカンマ区切りリスト。 Important タグに機密情報 (個人データなど) を保存しないでください。	"zone:1, critical:yes" "zone:3, critical:no, zone:1"

インポートテンプレートで定義されているすべてのフィールドにデータが入力されていなくても、各レコードに 1 つ以上の必須フィールドが含まれていれば、データをインポートすることができます。重複は、外部または内部の一致キーを使用して、複数のインポートリクエスト間で管理されます。独自の一致キー `External ID` を入力する場合は、このフィールドでレコードを一意に識別してインポートします。一致キーが指定されていない場合、インポートテンプレートの一部の列から派生した内部生成の一致キーがインポートに使用されます。この一致の詳細については、「検出されたサーバーやアプリケーションの一致ロジック (p. 40)」を参照してください。

Note

Migration Hub のインポートでは、インポートテンプレートで定義されていないフィールドはサポートされていません。カスタムフィールドは無視され、インポートもされません。

インポートのアクセス許可の設定

データをインポートする前に、アップロードに必要な Amazon S3 アクセス許可が IAM ユーザーにあることを確認します (`s3:PutObject`) 、インポートファイルを Amazon S3 にインポートして、オブジェクト (`s3:GetObject`). また、IAM ポリシーを作成し、AWS アカウントでインポートを実行する IAM ユーザーにアタッチすることで、プログラムによるアクセス (AWS CLI の場合) またはコンソールアクセスを確立する必要があります。

Console Permissions

コンソールを使用して AWS アカウントでインポートリクエストを行う IAM ユーザーのアクセス許可ポリシーを編集するには、次の手順を使用します。

ユーザーにアタッチされている管理ポリシーを編集するには

1. AWS Management Console にサインインして、IAM コンソール (https://console.aws.amazon.com/iam/) を開きます。
2. ナビゲーションペインで [Users] (ユーザー) を選択します。
3. アクセス許可ポリシーを変更する対象のユーザーの名前を選択します。
4. [アクセス許可] タブを選択後、[アクセス許可の追加] を選択します。
5. [Attach existing policies directly (既存のポリシーを直接アタッチ)]、[ポリシーの作成] の順に選択します。

a. 表示された [ポリシーの作成] ページで [JSON] を選択し、次のポリシーに貼り付けます。バケットの名前を、IAM ユーザーがインポートファイルをアップロードする実際のバケットの名前に置き換えることを忘れないでください。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation",
        "s3:ListAllMyBuckets"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": ["s3:ListBucket"],
      "Resource": ["arn:aws:s3:::importBucket"]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObject"
      ],
      "Resource": ["arn:aws:s3:::importBucket/*"]
    }
  ]
}
```

b. [ポリシーの確認] を選択します。

c. ポリシーに新しい [名前] と説明 (オプション) を入力してから、ポリシーの概要を確認します。

d. [Create policy] を選択します。

6. [] に戻ります。アクセス許可の付与AWS アカウントでインポートリクエストを行うユーザーの IAM コンソールページ。
7. ポリシーのテーブルを更新し、先ほど作成したポリシーの名前を検索します。
8. [Next: (次へ:)] を選択します 確認.
9. [Add permissions] を選択します。

AWS CLI Permissions

AWS アカウントでインポートリクエストを行う IAM ユーザーのアクセス許可ポリシーを編集するには、次の手順を使用します。AWS CLI。

ユーザーにアタッチされている管理ポリシーを編集するには

1. の使用`aws iam create-policy` AWS CLIのコマンドを実行して、次のアクセス許可を持つ IAM ポリシーを作成します。バケットの名前を、IAM ユーザーがインポートファイルをアップロードする実際のバケットの名前に置き換えることを忘れないでください。

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

```
      "Action": ["s3:ListBucket"],
      "Resource": ["arn:aws:s3:::importBucket"]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObject"
      ],
      "Resource": ["arn:aws:s3:::importBucket/*"]
    }
  ]
}
```

このコマンドの使用の詳細については、「」を参照してください。create-policy()AWS CLIコマンドリファレンス。

2. の使用`aws iam attach-user-policy` AWS CLIのコマンドを実行して、最後のステップで作成したポリシーを、AWS アカウントでインポートリクエストを行う IAM ユーザーにアタッチします。AWS CLI。このコマンドの使用の詳細については、「」を参照してください。attach-user-policy()AWS CLIコマンドリファレンス。

ポリシーが IAM ユーザーに追加され、インポートプロセスを開始できるようになりました。ユーザーが指定した Amazon S3 バケットにオブジェクトをアップロードすると、ユーザーがそのオブジェクトを読み取ることができるように、オブジェクトセットに対するデフォルトのアクセス許可の設定は保持されることに注意してください。

インポートファイルを Amazon S3 にアップロードする

次に、CSV 形式のインポートファイルを Amazon S3 にアップロードして、インポートできるようにする必要があります。開始する前に、インポートファイルを格納する Amazon S3 バケットを事前に作成または選択しておく必要があります。

Console S3 Upload

インポートファイルを Amazon S3 にアップロードするには

1. [] にサインインします。AWS Management ConsoleAmazon S3 コンソール () を開きます。https://console.aws.amazon.com/s3/。
2. [Bucket name (バケット名)] リストで、オブジェクトのアップロード先のバケットの名前を選択します。
3. [Upload (アップロード)] を選択します。
4. [Upload (アップロード)] ダイアログボックスで、[Add files (ファイルの追加)] を選択してアップロードするファイルを選択します。
5. アップロードするファイルを選択し、続いて [Open (オープン)] を選択します。
6. [Upload (アップロード)] を選択します。
7. ファイルがアップロードされたら、バケットのダッシュボードからデータファイルオブジェクトの名前を選択します。
8. オブジェクトの詳細ページの [概要] タブから、[オブジェクト URL] をコピーします。この情報は、インポートリクエストを作成するときに必要になります。
9. に戻り、S3 のデータファイルリンク[] フィールドをクリックします。新しいインポートの開始ページで.

AWS CLI S3 Upload

インポートファイルを Amazon S3 にアップロードするには

1. ターミナルウィンドウを開き、インポートファイルを保存したディレクトリに移動します。
2. 次のコマンドを入力します。

```
aws s3 cp ImportFile.csv s3://BucketName/ImportFile.csv
```

3. これにより、次の結果が返ります。

```
upload: .\ImportFile.csv to s3://BucketName/ImportFile.csv
```

4. 返った Amazon S3 オブジェクトの完全なパスをコピーします。この情報は、インポートリクエストを作成するときに必要になります。

データのインポート

Migration Hub コンソールからインポートテンプレートをダウンロードし、既存のオンプレミスサーバーデータを入力したら、Migration Hub へのデータのインポートを開始することができます。次の 2 通りの方法があります。コンソールを使用するか、を通じて API コールを行います。AWS CLI。各方法の手順は以下のとおりです。

Console Import

データインポートをツールページMigration Hub。

データのインポートを開始するには

1. ナビゲーションペインの [Discover (検出)] で [Tools (ツール)] を選択します。
2. インポートテンプレートに入力していない場合は、[] ダイアログボックスから、テンプレートをダウンロードすることができます。インポートテンプレート()インポートボックスに移動するとそのように表示されます。ダウンロードしたテンプレートを開き、既存のオンプレミスサーバーデータを入力します。インポートテンプレートは、Amazon S3 バケットにあるダウンロードすることもできます。https://s3.us-west-2.amazonaws.com/templates-7cffcf56-bd96-4b1c-b45b-a5b42f282e46/import_template.csv
3. [インポート[] ボタンをクリックします。インポートボックスが開きます。このボックスでは、インポートページの下のツール。
4. [新しいインポートの開始] を選択します。
5. 次の画面の [インポート名] フィールドで、インポートの名前を指定します。
6. [S3 のデータファイルリンク] フィールドに入力します。このステップを行うには、インポートデータファイルを Amazon S3 にアップロードする必要があります。詳細については、「インポートファイルを Amazon S3 にアップロードする (p. 36)」を参照してください。
7. 右下エリアにある [インポート] を選択します。[インポート] ページが開きます。テーブルには、インポートとそのステータスが表示されます。

前の手順に従って、データのインポートを開始したら、各インポートリクエストの詳細 (例: 進行状況のステータス、完了時間、レコードの成功/失敗数 (ダウンロード可能)) が [インポート] ページに表示されます。この画面から、[] に移動することもできます。サーバーページの下の検出をクリックして、実際にインポートされたデータを表示します。

[サーバー] ページでは、検出されたすべてのサーバー (デバイス) とインポート名を確認できます。からナビゲートすると、インポート(インポート履歴) ページに表示されるインポートの名前を選択して、名前列をクリックすると、[] に移動します。サーバーページが開き、選択したインポートのデー

タセットに基づいてフィルタが適用され、その特定のインポートに属するデータのみが表示されます。

アーカイブは、.zip 形式で提供され、`errors-file` と `failed-entries-file` の 2 つのファイルが含まれます。エラーファイルには、失敗した各行に関連付けられたエラーメッセージのリストと、インポートに失敗したデータファイルの関連付けられた列の名前が含まれます。このファイルを使用して、問題の発生原因をすばやく特定することができます。失敗したエントリファイルには、失敗した各行と提供されたすべての列が含まれます。このファイルのエラーファイルで変更を呼び出し、修正した情報を使用してファイルのインポートを再試行することができます。

AWS CLI Import

AWS CLI からデータのインポートプロセスを開始するには、最初に AWS CLI を環境にインストールする必要があります。詳細については、「」を参照してください。AWS コマンドラインインターフェイスのインストール()AWS Command Line Interfaceユーザーガイド。

Note

インポートテンプレートに入力していない場合は、Amazon S3 バケットにあるインポートテンプレート () をダウンロードすることができます。https://s3.us-west-2.amazonaws.com/templates-7cffcf56-bd96-4b1c-b45b-a5b42f282e46/import_template.csv

データのインポートを開始するには

1. ターミナルウィンドウを開いて、次のコマンドを入力します。

```
aws discovery start-import-task --import-url s3://BucketName/ImportFile.csv --
name ImportName
```

2. これにより、インポートタスクが作成され、次のステータス情報が返ります。

```
{
    "task": {
        "status": "IMPORT_IN_PROGRESS",
        "applicationImportSuccess": 0,
        "serverImportFailure": 0,
        "serverImportSuccess": 0,
        "name": "ImportName",
        "importRequestTime": 1547682819.801,
        "applicationImportFailure": 0,
        "clientRequestToken": "EXAMPLE1-abcd-1234-abcd-EXAMPLE1234",
        "importUrl": "s3://BucketName/ImportFile.csv",
        "importTaskId": "import-task-EXAMPLE1229949eabfEXAMPLE03862c0"
    }
}
```

Migration Hub インポート要求の追跡

Migration Hub のインポートリクエストの状態は、コンソール (AWS CLI、または AWS SDK のいずれかを使用します。

Console Tracking

[] から、インポートダッシュボードMigration Hub 次の要素があります。

- 名前— インポートリクエストの名前。
- ID のインポート— インポートリクエストの一意の ID。
- インポート時間— インポートリクエストが作成された日時。

- インポートステータス— インポートリクエストのステータスです。これには、次のいずれかの値を指定できます。
  - インポート中— このデータファイルは現在インポート中です。
  - インポート済み— データファイル全体が正常にインポートされました。
  - エラーとともにインポートされました— データファイル内の 1 つ以上のレコードをインポートできませんでした。失敗したレコードを解決するには、インポートタスクの [Download failed records (失敗したレコードのダウンロード)] を選択し、失敗したエントリの csv ファイルのエラーを解消してから、再度インポートを行います。
  - インポート失敗— インポートされたデータファイル内のどのレコードもインポートされませんでした。失敗したレコードを解決するには、インポートタスクの [Download failed records (失敗したレコードのダウンロード)] を選択し、失敗したエントリの csv ファイルのエラーを解消してから、再度インポートを行います。
- インポートされたレコード— 正常にインポートされた特定のデータファイル内のレコード数。
- 失敗したレコード— インポートされなかった特定のデータファイル内のレコード数。

CLI Tracking

インポートタスクのステータスは、AWS CLI の `aws discovery describe-import-tasks` コマンドを使用して追跡できます。

1. ターミナルウィンドウを開いて、次のコマンドを入力します。

   ```
   aws discovery describe-import-tasks
   ```

2. これにより、すべてのインポートタスクのリストが JSON 形式で返り、ステータスやその他の関連情報が含まれます。必要に応じて、インポートタスクのサブセットが返るように結果をフィルタリングすることができます。

インポートタスクを追跡すると、返った `serverImportFailure` 値がゼロより大きいことがわかります。この場合、インポートファイルには、インポートできなかったエントリが 1 つ以上含まれています。この問題を解消するには、失敗したレコードのアーカイブをダウンロードして、中のファイルを確認し、変更した failed-entries.csv ファイルを使用してインポートリクエストを行います。

インポートタスクを作成したら、データ移行の管理と追跡に役立つ他の操作を実行できます。たとえば、特定のリクエストに対して失敗したレコードのアーカイブをダウンロードできます。失敗したレコードのアーカイブを使用して、インポートの問題を解消する方法については、「失敗したインポートレコードの問題のトラブルシューティング (p. 99)」を参照してください。

検出したデータの表示、エクスポート、および検索

-AWS検出コネクタとAWSDiscovery Agentは両方とも、平均とピークの使用率に基づいてシステムパフォーマンスのデータを提供します。収集したシステムパフォーマンスのデータを使用して、ハイレベルな TCO (総所有コスト) を算定できます。検出エージェントは、システムパフォーマンス情報、着信/発信のネットワーク接続、およびサーバーで実行中のプロセスに関するより詳細な時系列データを収集します。このデータを使用して、サーバー間のネットワーク依存関係を確認し、関連するサーバーをアプリケーションとしてグループ化して移行計画に役立てることができます。

このセクションでは、コンソールとの両方から、検出コネクタと検出エージェントで検出したデータを表示および操作する手順を示します。AWS CLI。

トピック

コンソールを使用した収集データの表示

Discovery ConnectorとDiscovery Agentの両方で、データ収集プロセスの開始後に、コンソールを使用してサーバーと VM に関する収集データを表示できます。コンソールには、データ収集の開始から約 15 分後にデータが表示されます。経由で API コールを実行して収集したデータをエクスポートし、csv 形式でデータを表示することもできます。AWS CLI。収集データのエクスポートについては、次のセクション「収集データのエクスポート (p. 41)」で説明します。

検出したサーバーに関する収集データを表示するには

1. コンソールのナビゲーションペインで、[Servers (サーバー)] を選択します。検出したサーバーがサーバリストに表示されます。
2. 収集データの詳細を表示するには、[Server info (サーバー情報)] 列のサーバー名のリンクを選択します。表示される画面で、システム情報やパフォーマンスメトリクスなどの詳細情報を確認できます。

コンソールを使用して Discovery Connectors や Discovery Agent で検出したサーバーの表示、ソート、およびタグ付けを行う方法については、「」を参照してください。 AWS Application Discovery Service Console (p. 52)。

検出されたサーバーやアプリケーションの一致ロジック

AWS Application Discovery Service には、検出したサーバーが既存のエントリと一致した場合に識別する一致ロジックが組み込まれています。この口ジックで一致が見つかると、検出済みの既存のサーバーの情報は、新しい値で更新されます。この一致ロジックは、Migration Hub のインポート、検出コネク

タ、検出エージェント、およびその他の移行ツールなど、複数のソースの重複したサーバーを処理します。Migration Hub のインポートの詳細については、「」を参照してください。AWS Migration Hub のインポート。

サーバーが検出されると、インポートされたサーバーが存在していないことを確認するために、各エントリは、以前にインポートされたレコードと照合されます。一致が見つからない場合は、新しいレコードが作成され、一意の新しいサーバー ID が割り当てられます。一致が見つからない場合でも新しいエントリは作成されますが、既存のサーバーと同じ一意のサーバー ID が割り当てられます。Migration Hub コンソールでこのサーバーを表示している場合は、サーバーに対して 1 つの一意のエントリのみ表示されます。

このエントリに関連付けられたサーバー属性は、使用可能な以前のレコードや、新しくインポートされたレコードの属性値が表示されるようにマージされます。複数のソースの特定のサーバー属性の値が複数ある場合 (インポートおよび Discovery Agent によって検出された特定のサーバーに関連付けられた `Total RAM` の 2 つの異なる値など)、サーバーの一致レコードには、最後に更新された値が表示されます。

一致フィールド

次のフィールドは、検出ツールの使用時にサーバーを一致させるために使用されます。

- ExternalId— サーバーの一致に使用する主要フィールドです。このフィールドの値が、別のと同一である場合`ExternalId`別のエントリで、他のフィールドが一致しているかどうかに関係なく、Application Discovery Service は 2 つのエントリを一致させます。
- IPAddress
- HostName
- MacAddress
- VMware.MORefIdおよびvCenterIDApplication Discovery Service で一致を行うには、これらの値はいずれも、別のエントリの各フィールドと同一である必要があります。

収集データのエクスポート

検出コネクタと Discovery Agent の両方について、データ収集プロセスの開始後に、サーバーと VM に関する収集データをエクスポートできます。このデータは、コンソールを操作するか、経由で API コールを実行することでエクスポートできます。AWS CLIデータの収集に使用した検索ツールによります。

- 検出エージェントの場合、コンソールまたはから収集データをエクスポートできます。AWS CLI。
- 検出コネクタの場合、からのみ収集データをエクスポートできます。AWS CLI。

各方法の手順を示します。次のいずれかを選択して展開してください。

すべてのサーバーに関するシステムパフォーマンスデータのエクスポート

ホストと VM で実行されている検出コネクタと検出エージェントからの収集データは、AWS CLI。AWS CLI をまだインストールしていない場合は、最初に環境にインストールする必要があります。

AWS CLI をインストールして収集データをエクスポートするには

1. OS のタイプ (Windows または Mac/Linux) に適切な AWS CLI をインストールします (まだインストールしていない場合)。フレームワークの使用の詳細については、AWS Command Line Interfaceユーザーガイド手順については、こちらを参照してください
2. コマンドプロンプト (Windows) またはターミナル (MAC/Linux) を開きます。

a. `aws configure` を入力して、[Enter] を押します。
b. AWS アクセスキー ID と AWS シークレットアクセスキーを入力します。
c. デフォルトのリージョン名として「`us-west-2`」と入力します。
d. デフォルトの出力形式として「`text`」と入力します。

3. 次のコマンドを入力してエクスポート ID を生成します。

```
aws discovery start-export-task
```

4. 次のコマンドで、前のステップで生成したエクスポート ID を使用し、パラメータ `"configurationsDownloadUrl"` の値として S3 URL を生成します。

```
aws discovery describe-export-tasks --export-ids <export ID>
```

5. 前のステップで生成した URL をコピーしてブラウザに貼り付け、検出したサーバーの収集データの zip ファイルをダウンロードします。

コンソールを使用したエージェントの収集データのエクスポート

エージェントの収集データをコンソールからエクスポートする場合、特定のサーバーの詳細ページから利用できるエージェントは 1 つに限られます。サーバーのエクスポートジョブは、画面下部の [Exports (エクスポート)] に表示されます。まだエクスポートジョブがない場合、テーブルは空になります。サーバーデータのエクスポートは、最大 5 つまで同時に実行できます。

検出したサーバーに関する収集データをエクスポートするには

1. ナビゲーションペインで、[Servers (サーバー)] を選択します。
2. [Server info (サーバー情報)] 列で、データをエクスポートするサーバーのリンクを選択します。
3. 画面下部の [Exports (エクスポート)] セクションで、[Export server details (サーバー詳細のエクスポート)] を選択します。
4. [Export server details (サーバー詳細のエクスポート)] で、[Start date (開始日)] と [Time (時刻)] を入力します。

   Note

   開始時刻は、現在の時刻から 72 時間より前にすることはできません。

5. ジョブを開始するには、[Export (エクスポート)] を選択します。最初のステータスは [In-progress (進行中)] です。ステータスを更新するには、[Exports (エクスポート)] セクションの更新アイコンをクリックします。
6. エクスポートジョブが完了したら、[Download (ダウンロード)] を選択して .zip ファイルを保存します。
7. 保存されたファイルを解凍します。エクスポートデータは、次のような .csv ファイルのセットに含まれています。

   - *<AWS ##### ID>*_destinationProcessConnection.csv
   - *<AWS ##### ID>*_networkInterface.csv
   - *<AWS ##### ID>*_osInfo.csv
   - *<AWS ##### ID>*_process.csv
   - *<AWS ##### ID>*_sourceProcessConnection.csv
   - *<AWS ##### ID>*_systemPerformance.csv

   .csv ファイルを Microsoft Excel で開き、エクスポートしたサーバーデータを確認できます。

複数のファイルの 1 つは JSON ファイルであり、これにはエクスポートタスクとその結果に関するデータが含まれています。

Note

Amazon EC2 インスタンスの推奨事項の生成とエクスポートの詳細については、AWS Migration Hubコンソールについては、」Amazon EC2 インスタンスの推奨事項()AWS Migration Hubユーザーガイド。

Amazon Athena でのデータ探索

Amazon Athena でのデータ探索では、Discovery Agents によって検出されたすべてのオンプレミスサーバーから収集されたデータを 1 つの場所で分析できます。Amazon Athena でのデータ探索をMigration Hub コンソールから (または、StartContinousExport API を使用して) 有効にし、エージェントのデータ収集を有効にすると、エージェントによって収集されたデータは、定期的に S3 バケットに自動的に保存されます。

次に Amazon Athena にアクセスして、各サーバーの時系列のシステムパフォーマンス、各サーバーで実行されているプロセスのタイプ、および、複数のサーバー間のネットワーク依存関係を分析する事前定義されたクエリを実行できます。さらに、Amazon Athena を使用して独自のカスタムクエリを記述して、設定管理データベース (CMDB) のエクスポートなど、既存のデータソースをアップロードしたり、検出されたサーバーを実際のビジネスアプリケーションと関連付けたりすることができます。Athena データベースと Amazon QuickSight を統合することもでき、クエリ出力を視覚化したり、追加の分析を実行できます。

Steps

1. Amazon Athena でのデータ探索の有効化 (p. 43)
2. Amazon Athena でのデータ探索の操作 (p. 44)

Amazon Athena でのデータ探索の有効化

Amazon Athena でのデータ探索を有効にするには、Migration Hub コンソールを使用するかからの API 呼び出しを使用して、継続的なエクスポートを有効にします。AWS CLI。Amazon Athena で検出されたデータを表示および探索するには、データ探索をオンにする必要があります。

継続的なエクスポートを有効にすると、サービスにリンクされたロール`AWSServiceRoleForApplicationDiscoveryServiceContinuousExport`は自動的にアカウントで使用されます。このサービスにリンクされたロールの詳細については、「Application Discovery Service のサービスにリンクされたロールのアクセス許可 (p. 86)」を参照してください。

以下の手順では、Amazon Athena でのデータ探索を有効にする方法を示しています。コンソールおよびAWS CLI。

Enable with the console

Amazon Athena でのデータ探索は、「データ収集の開始」を選択するか、「Amazon Athena でのデータ探索」というラベルが付いた切り替えをクリックすると、継続的なエクスポートが暗黙的に有効になり、有効になります。データコレクターページMigration Hub。

コンソールから Amazon Athena でデータ探索を有効にするには

1. ナビゲーションペインで、[Data Collectors (データコレクタ)] を選択します。
2. [Agents (エージェント)] タブを選択します。

3. 選択データ収集の開始をクリックするか、データの収集が有効になっている場合、[Amazon Athena でのデータ探査を切り替え。
4. 前のステップで作成したダイアログボックスで、関連するコストに同意するチェックボックスをオンにして、[Continue (続行)] または [Enable (有効)] を選択します。

   Note

   エージェントは、「継続的なエクスポート」モードで実行されていて、Amazon Athena で検出されたデータを表示および操作できます。初めて有効にする場合は、Amazon Athena にデータが表示されるまでに、最長で 30 分かかる場合があります。

Enable with the AWS CLI

Amazon Athena でのデータ探索は、からのAPI呼び出しをスローして、継続的なエクスポートを明示的に有効にし、を有効にします。AWS CLI。これを行うには、まず AWS CLI が環境にインストールされている必要があります。

をインストールするにはAWS CLIし、Amazon Athena でのデータ探索を有効にします。

1. オペレーティングシステム (Linux、macOS、または Windows) 用の AWS CLI をインストールします。フレームワークの使用の詳細については、AWS Command Line Interfaceユーザーガイド手順については、
2. コマンドプロンプト (Windows) またはターミナル (Linux/macOS) を開きます。
   a. `aws configure` を入力して、[Enter] を押します。
   b. AWS アクセスキー ID と AWS シークレットアクセスキーを入力します。
   c. デフォルトのリージョン名として「`us-west-2`」と入力します。
   d. デフォルトの出力形式として「`text`」と入力します。
3. 次のコマンドを入力します。

```
aws discovery start-continuous-export
```

   Note

   エージェントは、「継続的なエクスポート」モードで実行されていて、Amazon Athena で検出されたデータを表示および操作できます。初めて有効にする場合は、Amazon Athena にデータが表示されるまでに、最長で 30 分かかる場合があります。

Amazon Athena でのデータ探索の操作

Amazon Athena でデータ探索を有効にすると、Athena で直接データをクエリすることで、エージェントによって検出された詳細な最新データの調査および操作を開始できます。このデータを使用して、スプレッドシートの作成、コスト分析の実行、視覚化プログラムへのクエリの移植などを行うことができます。

このセクションのトピックでは、AWS への ローカル環境の移行を評価および計画するために、Athena でデータを操作する方法について説明します。

トピック

Amazon Athena でのデータの直接調査

以下の手順では、すべてのエージェントデータを Athena コンソール内で直接参照します。Athena にデータがない場合、または Amazon Athena でのデータ探索を有効にしていない場合は、「」で説明されているように、Amazon Athena でのデータ探索を有効にするように求めるダイアログボックスが表示されます。Amazon Athena でのデータ探索の有効化 (p. 43)。

Athena でエージェントによって検出されたデータを直接検索するには

1. AWS Migration Hub コンソールを開き、ナビゲーションペインで [Servers (サーバー)] を選択します。
2. Amazon Athena コンソールを開くには、Amazon Athena データを探る。
3. [Query Editor (クエリエディタ)] ページのナビゲーションペインの [Database (データベース)] で、application_discovery_service_database が選択されていることを確認します。

   Note

   [Tables (テーブル)] で、以下のテーブルは、エージェントによってグループ化されたデータセットを表しています。

   - os_info_agent
   - network_interface_agent
   - sys_performance_agent
   - processes_agent
   - inbound_connection_agent
   - outbound_connection_agent
   - id_mapping_agent
4. Athena クエリエディタで SQL クエリを記述して実行することで、Amazon Athena コンソールでデータをクエリします。たとえば、以下のクエリを使用して、検出されたすべてのサーバー IP アドレスを確認できます。

   ```
   SELECT * FROM network_interface_agent;
   ```

   クエリの例については、「Athena で使用する事前定義されたクエリ (p. 46)」を参照してください。

Amazon Athena データの視覚化

データを視覚化するには、Amazon QuickSight などの視覚化プログラム、Cytoscape、yEd、Gelphi などのオープンソースの視覚化ツールにクエリを移植します。ネットワーク図、要約グラフなどのグラフィカルな表現をレンダリングするには、これらのツールを使用します。この方法を使用して、Athena を視覚化プログラムに接続すると、収集したデータにアクセスし、そのデータをソースとしてビジュアライゼーションを生成できます。

Amazon QuickSight を使用して Amazon Athena データを視覚化するには

1. にサインインするAmazon QuickSight。
2. [Connect to another data source or upload a file (別のデータソースに接続するか、ファイルをアップロードします)] を選択します。
3. 選択Athena。-新しい Athena データソースダイアログボックスが表示されます。
4. [Data source name (データソース名)] フィールドに名前を入力します。
5. [Create data source] を選択します。
6. [Choose your table (テーブルの選択)] ダイアログボックスで、[Agents-servers-os] テーブルを選択して、[Select (選択)] を選択します。

7. [Finish data set creation (データセット作成の終了)] ダイアログボックスで、[Import to SPICE for quicker analytics (SPICE にインポートしてクイック分析)] を選択して、[Visualize (視覚化)] を選択します。

   ビジュアライゼーションがレンダリングされます。

Athena で使用する事前定義されたクエリ

このセクションでは、TCO 分析やネットワークの可視化などの一般的なユースケースを実行する、一連の事前定義されたクエリを示します。これらのクエリをそのまま、あるいは必要に応じて変更して使用できます。

事前定義されたクエリを使用するには

1. AWS Migration Hub コンソールを開き、ナビゲーションペインで [Servers (サーバー)] を選択します。
2. Amazon Athena コンソールを開くには、Amazon Athena データを探る。
3. [Query Editor (クエリエディタ)] ページのナビゲーションペインの [Database (データベース)] で、application_discovery_service_database が選択されていることを確認します。
4. クエリエディタでプラス記号 (+) を選択して、新しいクエリのタブを作成します。
5. 「事前に定義されたクエリ (p. 46)」からいずれかのクエリをコピーします。
6. 作成した新しいクエリタブのクエリウィンドウにそのクエリを貼り付けます。
7. [Run Query] を選択します。

事前に定義されたクエリ

タイトルを選択すると、クエリに関する情報が表示されます。

サーバーの IP アドレスおよびホスト名を取得する

このビューヘルパー関数では、特定のサーバーの IP アドレスとホスト名を取得します。このビューは他のクエリで使用できます。ビューを作成する方法については、」CREATE()Amazon Athena ユーザーガイド。

```
CREATE OR REPLACE VIEW hostname_ip_helper AS
SELECT DISTINCT
  "os"."host_name"
, "nic"."agent_id"
, "nic"."ip_address"
FROM
  os_info_agent os
, network_interface_agent nic
WHERE ("os"."agent_id" = "nic"."agent_id");
```

エージェントの有無に関係なくサーバーを識別する

このクエリは、データ検証を実行するのに役立ちます。ネットワーク内の多数のサーバーにエージェントをデプロイした場合は、このクエリを使用して、エージェントが配置されていない他のサーバーがネットワーク内にあるかどうかを確認できます。このクエリでは、インバウンドとアウトバウンドのネットワークトラフィックを調べ、プライベート IP アドレスについてのみトラフィックをフィルタリングします。つまり、192、10、172 で始まる IP アドレスです。

```
SELECT DISTINCT "destination_ip" "IP Address" ,
         (CASE
```

```
    WHEN (
    (SELECT "count"(*)
    FROM network_interface_agent
    WHERE ("ip_address" = "destination_ip") ) = 0) THEN
        'no'
        WHEN (
        (SELECT "count"(*)
        FROM network_interface_agent
        WHERE ("ip_address" = "destination_ip") ) > 0) THEN
            'yes' END) "agent_running"
    FROM outbound_connection_agent
WHERE ((("destination_ip" LIKE '192.%')
        OR ("destination_ip" LIKE '10.%'))
        OR ("destination_ip" LIKE '172.%'))
UNION
SELECT DISTINCT "source_ip" "IP ADDRESS" ,
         (CASE
    WHEN (
    (SELECT "count"(*)
    FROM network_interface_agent
    WHERE ("ip_address" = "source_ip") ) = 0) THEN
        'no'
        WHEN (
        (SELECT "count"(*)
        FROM network_interface_agent
        WHERE ("ip_address" = "source_ip") ) > 0) THEN
            'yes' END) "agent_running"
    FROM inbound_connection_agent
WHERE ((("source_ip" LIKE '192.%')
        OR ("source_ip" LIKE '10.%'))
        OR ("source_ip" LIKE '172.%'));
```

エージェントを含むサーバーのシステムパフォーマンスデータを分析する

このクエリを使用して、エージェントがインストールされているオンプレミスサーバーのシステムパフォーマンスと使用パターンデータを分析できます。このクエリでは、system_performance_agent テーブルと os_info_agent テーブルを組み合わせて、各サーバーのホスト名を識別します。このクエリでは、エージェントが稼働しているすべてのサーバーの時系列の使用状況データ (15 分間隔) が返ります。

```
SELECT "OS"."os_name" "OS Name" ,
    "OS"."os_version" "OS Version" ,
    "OS"."host_name" "Host Name" ,
     "SP"."agent_id" ,
     "SP"."total_num_cores" "Number of Cores" ,
     "SP"."total_num_cpus" "Number of CPU" ,
     "SP"."total_cpu_usage_pct" "CPU Percentage" ,
     "SP"."total_disk_size_in_gb" "Total Storage (GB)" ,
     "SP"."total_disk_free_size_in_gb" "Free Storage (GB)" ,
     ("SP"."total_disk_size_in_gb" - "SP"."total_disk_free_size_in_gb") "Used Storage" ,
     "SP"."total_ram_in_mb" "Total RAM (MB)" ,
     ("SP"."total_ram_in_mb" - "SP"."free_ram_in_mb") "Used RAM (MB)" ,
     "SP"."free_ram_in_mb" "Free RAM (MB)" ,
     "SP"."total_disk_read_ops_per_sec" "Disk Read IOPS" ,
     "SP"."total_disk_bytes_written_per_sec_in_kbps" "Disk Write IOPS" ,
     "SP"."total_network_bytes_read_per_sec_in_kbps" "Network Reads (kbps)" ,
     "SP"."total_network_bytes_written_per_sec_in_kbps" "Network Write (kbps)"
FROM "sys_performance_agent" "SP" , "OS_INFO_agent" "OS"
WHERE ("SP"."agent_id" = "OS"."agent_id") limit 10;
```

ポート番号とプロセスの詳細に基づくサーバー間のアウトバウンド通信の追跡

このクエリでは、ポート番号とプロセスの詳細と共に、各サービスのアウトバウンドトラフィックの詳細が返されます。

クエリを実行する前に、まだ行っていない場合は、IANA からダウンロードした IANA ポートレジストリデータベースを含む iana_service_ports_import テーブルを作成する必要があります。このテーブルを作成する方法については、「IANA ポートレジストリインポートテーブルの作成 (p. 50)」を参照してください。

iana_service_ports_import テーブルが作成されたら、アウトバウンドトラフィックを追跡する 2 つのビューヘルパー関数を作成します。ビューを作成する方法については、」CREATE()Amazon Athena ユーザーガイド。

アウトバウンド追跡ヘルパー関数を作成するには

1. https://console.aws.amazon.com/athena/ で Athena コンソールを開きます。
2. の作成valid_outbound_ips_helperすべての個別のアウトバウンド先 IP アドレスのリストを取得する以下のヘルパー関数を使用して、ビューを表示します。

```
CREATE OR REPLACE VIEW valid_outbound_ips_helper AS
SELECT DISTINCT "destination_ip"
FROM outbound_connection_agent;
```

3. アウトバウンドトラフィックの通信頻度を決定する以下のヘルパー関数を使用して、ビュー outbound_query_helper を作成します。

```
CREATE OR REPLACE VIEW outbound_query_helper AS
SELECT "agent_id" ,
         "source_ip" ,
         "destination_ip" ,
         "destination_port" ,
         "agent_assigned_process_id" ,
         "count"(*) "frequency"
FROM outbound_connection_agent
WHERE (("ip_version" = 'IPv4')
        AND ("destination_ip" IN
    (SELECT *
    FROM valid_outbound_ips_helper )))
GROUP BY  "agent_id", "source_ip", "destination_ip", "destination_port",
 "agent_assigned_process_id";
```

4. iana_service_ports_import テーブルと 2 つのヘルパー関数を作成したら、以下のクエリを実行して、各サービスのアウトバウンドトラフィックの詳細をポート番号とプロセスの詳細と共に取得できます。

```
SELECT hip1.host_name "Source Host Name",
         outbound_connections_results0.source_ip "Source IP Address",
         hip2.host_name "Destination Host Name",
         outbound_connections_results0.destination_ip "Destination IP Address",
         outbound_connections_results0.frequency "Connection Frequency",
         outbound_connections_results0.destination_port "Destination Communication
 Port",
         outbound_connections_results0.servicename "Process Service Name",
         outbound_connections_results0.description "Process Service Description"
FROM
    (SELECT DISTINCT o.source_ip,
         o.destination_ip,
         o.frequency,
         o.destination_port,
         ianap.servicename,
         ianap.description
    FROM outbound_query_helper o, iana_service_ports_import ianap
    WHERE o.destination_port = TRY_CAST(ianap.portnumber AS integer)) AS
 outbound_connections_results0 LEFT OUTER
JOIN hostname_ip_helper hip1
    ON outbound_connections_results0.source_ip = hip1.ip_address LEFT OUTER
```

```
JOIN hostname_ip_helper hip2
    ON outbound_connections_results0.destination_ip = hip2.ip_address
```

ポート番号およびプロセスの詳細に基づくサーバー間のインバウンド通信の追跡

このクエリでは、ポート番号とプロセスの詳細と共に、各サービスのインバウンドトラフィックに関する情報が返されます。

このクエリを実行する前に、まだ行っていない場合は、IANA からダウンロードした IANA ポートレジストリデータベースを含む `iana_service_ports_import` テーブルを作成する必要があります。このテーブルを作成する方法については、「IANA ポートレジストリインポートテーブルの作成 (p. 50)」を参照してください。

`iana_service_ports_import` テーブルが作成されたら、インバウンドトラフィックを追跡する 2 つのビューヘルパー関数を作成します。ビューを作成する方法については、」CREATE()Amazon Athena ユーザーガイド。

インポートの追跡ヘルパー関数を作成するには

1. https://console.aws.amazon.com/athena/ で Athena コンソールを開きます。
2. すべての個別のインバウンド元 IP アドレスのリストを取得する以下のヘルパー関数を使用して、ビュー `valid_inbound_ips_helper` を作成します。

```
CREATE OR REPLACE VIEW valid_inbound_ips_helper AS
SELECT DISTINCT "source_ip"
FROM inbound_connection_agent;
```

3. インバウンドトラフィックの通信頻度を決定する以下のヘルパー関数を使用して、ビュー `inbound_query_helper` を作成します。

```
CREATE OR REPLACE VIEW inbound_query_helper AS
SELECT "agent_id" ,
         "source_ip" ,
         "destination_ip" ,
         "destination_port" ,
         "agent_assigned_process_id" ,
         "count"(*) "frequency"
FROM inbound_connection_agent
WHERE (("ip_version" = 'IPv4')
        AND ("source_ip" IN
    (SELECT *
    FROM valid_inbound_ips_helper )))
GROUP BY  "agent_id", "source_ip", "destination_ip", "destination_port",
 "agent_assigned_process_id";
```

4. `iana_service_ports_import` テーブルと 2 つのヘルパー関数を作成したら、以下のクエリを実行して、各サービスのインバウンドトラフィックの詳細をポート番号とプロセスの詳細と共に取得できます。

```
SELECT hip1.host_name "Source Host Name",
         inbound_connections_results0.source_ip "Source IP Address",
         hip2.host_name "Destination Host Name",
         inbound_connections_results0.destination_ip "Destination IP Address",
         inbound_connections_results0.frequency "Connection Frequency",
         inbound_connections_results0.destination_port "Destination Communication
 Port",
         inbound_connections_results0.servicename "Process Service Name",
         inbound_connections_results0.description "Process Service Description"
FROM
```

```
    (SELECT DISTINCT i.source_ip,
         i.destination_ip,
         i.frequency,
         i.destination_port,
         ianap.servicename,
         ianap.description
    FROM inbound_query_helper i, iana_service_ports_import ianap
    WHERE i.destination_port = TRY_CAST(ianap.portnumber AS integer)) AS
 inbound_connections_results0 LEFT OUTER
JOIN hostname_ip_helper hip1
    ON inbound_connections_results0.source_ip = hip1.ip_address LEFT OUTER
JOIN hostname_ip_helper hip2
    ON inbound_connections_results0.destination_ip = hip2.ip_address
```

ポート番号から実行中のソフトウェアを識別する

このクエリでは、ポート番号に基づいて実行中のソフトウェアが識別されます。

このクエリを実行する前に、まだ行っていない場合は、IANA からダウンロードした IANA ポートレジストリデータベースを含む `iana_service_ports_import` テーブルを作成する必要があります。このテーブルを作成する方法については、「IANA ポートレジストリインポートテーブルの作成 (p. 50)」を参照してください。

以下のクエリを実行して、ポート番号に基づき、実行中のソフトウェアを識別します。

```
SELECT DISTINCT
  "o"."host_name" "Host Name"
, "ianap"."servicename" "Service"
, "ianap"."description" "Description"
, "con"."destination_port"
, "count"("con"."destination_port") "Destination Port Count"
FROM
  inbound_connection_agent con
, os_info_agent o
, iana_service_ports_import ianap
, network_interface_agent ni
WHERE (((((("con"."destination_ip" = "ni"."ip_address") AND (NOT ("con"."destination_ip"
 LIKE '172%'))) AND (("con"."destination_port" = "ianap"."portnumber") AND
 ("ianap"."transportprotocol" = 'tcp'))) AND ("con"."agent_id" = "o"."agent_id")) AND
 ("o"."agent_id" = "ni"."agent_id"))
GROUP BY "o"."host_name", "ianap"."servicename", "ianap"."description",
 "con"."destination_port"
ORDER BY "Destination Port Count" DESC;
```

IANA ポートレジストリインポートテーブルの作成

事前定義されたクエリによっては、Internet Assigned Numbers Authority (IANA) からダウンロードした情報を含む `iana_service_ports_import` という名前のテーブルが必要になる場合があります。

iana_service_ports_import テーブルを作成するには

1. IANA ポートレジストリデータベースのダウンロードCSVファイルからサービス名とトランスポートプロトコルのポート番号レジストリでのiana.org。
2. ファイルを Amazon S3 にアップロードします。詳細については、「S3 バケットにファイルとフォルダをアップロードする方法」を参照してください。
3. Athena にという名前の新しいテーブルを作成します。`iana_service_ports_import`。手順については、以下を参照してください。テーブルを作成する()Amazon Athena ユーザーガイド。以下の例では、`my_bucket_name` を、前の手順で CSV ファイルをアップロードした S3 バケットの名前に置き換える必要があります。

```
CREATE EXTERNAL TABLE IF NOT EXISTS iana_service_ports_import (
         ServiceName STRING,
         PortNumber INT,
         TransportProtocol STRING,
         Description STRING,
         Assignee STRING,
         Contact STRING,
         RegistrationDate STRING,
         ModificationDate STRING,
         Reference STRING,
         ServiceCode STRING,
         UnauthorizedUseReported STRING,
         AssignmentNotes STRING
)
ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe'
WITH SERDEPROPERTIES (
  'serialization.format' = ',',
  'quoteChar' = '"',
  'field.delim' = ','
) LOCATION 's3://my_bucket_name/'
TBLPROPERTIES ('has_encrypted_data'='false',"skip.header.line.count"="1");
```

AWS Application Discovery Service Console

AWS Application Discovery Service は AWS Migration Hub と統合されており、Migration Hub 内でデータコレクタ、サーバー、およびアプリケーションを表示および管理できます。Application Discovery Service コンソールを使用するときに Migration Hub コンソールにリダイレクトされます。Migration Hub コンソールを使用するのに、追加のステップやセットアップは不要です。

このセクションでは、コンソールを使用して検出コネクタと検出エージェントを管理および監視する方法について説明します。

トピック

メインダッシュボード

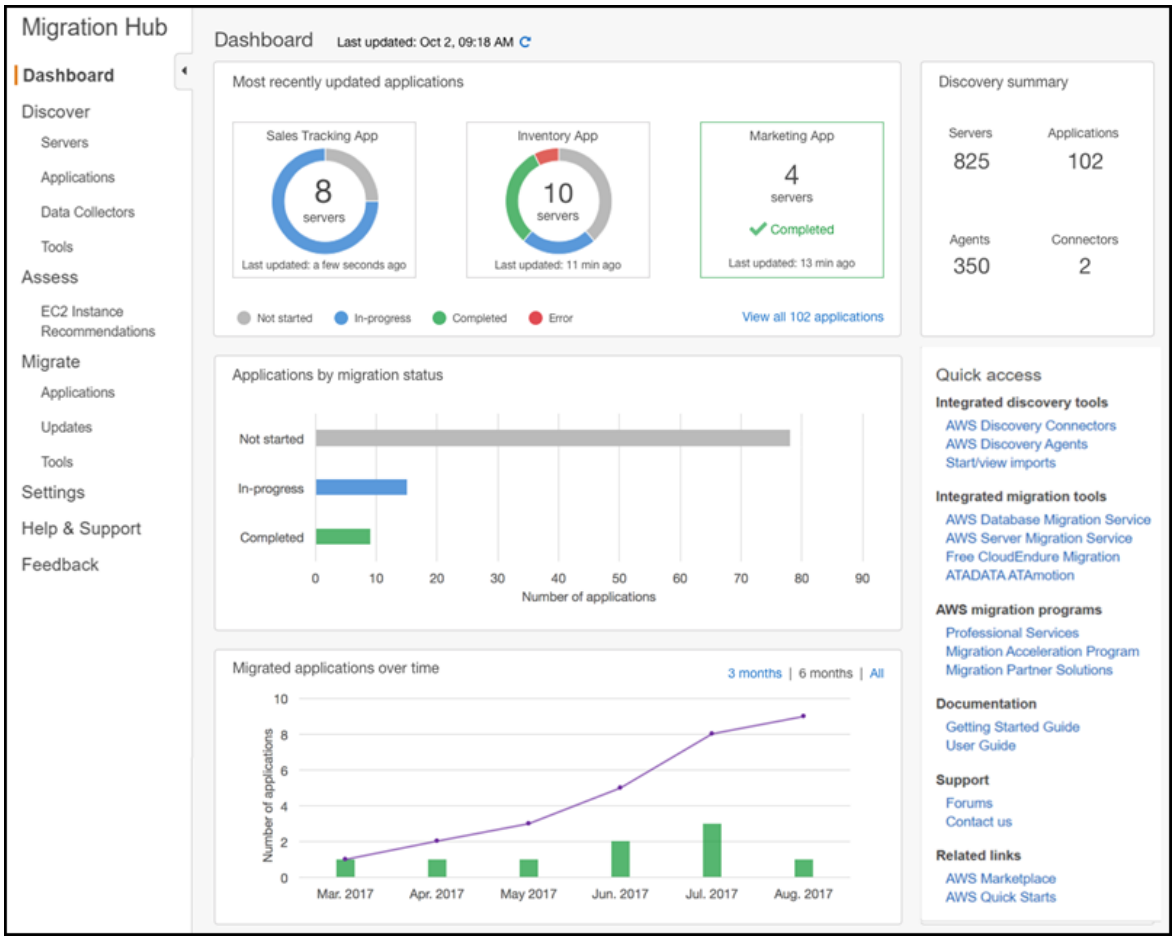
メインダッシュボードは、AWS Migration Hub コンソールのホームページでデフォルトで開きます。または、ダッシュボードナビゲーションペインで、[] を選択します。Migration Hub のメインダッシュボードでは、サーバー、アプリケーション、データコレクタ (検出コネクタや検出エージェントなど) のハイレベルな統計情報を表示できます。

トピック

メインダッシュボード

メインダッシュボードでは、中央にある [Discover (検出)] ダッシュボードと [Migrate (移行)] ダッシュボードからのデータを収集します。メインダッシュボードには、ステータスと情報のペインが 4 つあり、クイックアクセス用のリンクのリストもあります。各ペインでは、直近に更新されたアプリケーションのステータスの概要を確認できます。また、すべてのアプリケーションにすばやくアクセスしたり、異なる状態のアプリケーションの概要を取得したり、時間の経過とともに移行の進行状況を追跡したりできます。

メインダッシュボードにアクセスするには、ダッシュボードMigration Hub ホームページの左側にあるナビゲーションペインから必要な情報に移動できます。



ダッシュボードとナビゲーションペインからのナビゲーション

ダッシュボードでデータの概要を確認したら、さらに詳細を参照できます。これを行うには、メインダッシュボードの該当するステータスや情報のボックスから直接移動します。

次の表は、ダッシュボードから必要な情報に移動する手順の一覧です。Migration Hub ホームページの左側にあるナビゲーションペインから必要な情報に移動する手順も示してあります。

確認する項目	ダッシュボードでの操作	ナビゲーションペインでの操作
すべてのサーバー	メインダッシュボードの [Discovery summary (検出の概要)] ボックスで、[Servers (サーバー)] を選択します。	1. ナビゲーションペインで、[Servers (サーバー)] を選択します。
すべてのエージェント	メインダッシュボードの [Discovery summary (検出の概要)] ボックスで、[Agents (エージェント)] を選択します。	1. ナビゲーションペインで、[Data Collectors (データコレクタ)] を選択します。 2. [Agents (エージェント)] タブを選択します。

確認する項目	ダッシュボードでの操作	ナビゲーションペインでの操作
すべてのコネクタ	メインダッシュボードの [Discovery summary (検出の概要)] ボックスで、[Connectors (コネクタ)] を選択します。	1. ナビゲーションペインで、[Data Collectors (データコレクタ)] を選択します。 2. [Connectors (コネクタ)] タブを選択します。
すべてのアプリケーション	メインダッシュボードまたは [Migrate (移行)] ダッシュボードの [Most recently updated applications (直近に更新されたアプリケーション)] ボックスで、[View all applications (すべてのアプリケーションの表示)] を選択します。 [Discover (検出)] ダッシュボードの [Servers & Applications (サーバーとアプリケーション)] ボックスで、[View all applications (すべてのアプリケーションの表示)] を選択します。	1. ナビゲーションペインで [Migrate (移行)] の [Applications (アプリケーション)] を選択します。 2. [アプリケーション] を選択します。
アプリケーションの詳細: • 移行ステータス • サーバーリスト	メインダッシュボードまたは [Migrate (移行)] ダッシュボードの [Most recently updated applications (直近に更新されたアプリケーション)] ボックスで、アプリケーションのステータスボックスを選択します。	1. ナビゲーションペインで [Migrate (移行)] を選択します。 2. [アプリケーション] を選択します。 3. [Application Name (アプリケーション名)] 列で、アプリケーション名を選択します。
サーバーの詳細: • 基本的な情報 • パフォーマンス情報	1. メインダッシュボードまたは [Migrate (移行)] ダッシュボードの [Most recently updated applications (直近に更新されたアプリケーション)] ボックスで、アプリケーションを選択します。 2. [Server ID (サーバー ID)] 列で、サーバー名を選択します。	1. ナビゲーションペインで、[Servers (サーバー)] を選択します。 2. [Server ID (サーバー ID)] 列で、サーバー名を選択します。

データ収集ツール

検出コネクタと検出エージェントは、既存のインフラストラクチャを検出するために、Application Discovery Service で使用するデータ収集ツールです。検出コネクタと検出エージェントは、「AWS エージェントレス検出コネクタ (p. 20)」と「AWS アプリケーション検出エージェント (p. 6)」の説明に従ってダウンロードおよびデプロイできます。

これらのデータ収集ツールは、Application Discovery Service のリポジトリにデータを保存し、各サーバーおよび各サーバーで実行されているプロセスの詳細を提供します。これらのツールのいずれかを展開して、Migration Hub コンソールからのデータ収集を開始、停止、および表示できます。

トピック

データコレクタの開始と停止

Discovery ConnectorとDiscovery Agent のいずれをデプロイした場合でも、データコレクタページ Migration Hub。

データ収集ツールを開始または停止するには

1. ナビゲーションペインで、[Data Collectors (データコレクタ)] を選択します。
2. [Connectors (コネクタ)] タブまたは [Agents (エージェント)] タブを選択します。
3. 開始または停止する収集ツールのチェックボックスをオンにします。
4. [Start data collection (データ収集の開始)] または [Stop data collection (データ収集の停止)] を選択します。

データコレクタの表示とソート

複数のデータコレクタをデプロイした場合は、検出コネクタまたは検出エージェントをソートできます。データコレクタコンソールの [] ページを開きます。これを行うには、検索バーでフィルタを適用します。検索とフィルタ処理は、[Data Collectors (データコレクタ)] で指定したほとんどの条件で実行できます。

次の表は、検索条件、演算子、値、および値の定義の一覧です。

検索条件	演算子	値: 定義
収集ステータス		開始済み: データは収集されており、Application Discovery Service スケジュールされた開始: データ収集の開始が予定済みです。データ Service に送信され、ステータスが開始。 停止済み: データは収集されておらず、Application Discovery Servic スケジュールされた停止: データ収集の停止が予定済みです。データ Service への送信が停止され、ステータスが停止。
ヘルス	== !=	正常: データ収集が有効になっていません。ツールは正常に機能して 非正常の状態: ツールはエラー状態です。データは収集されていな 不明: 接続が確立されていない状態が 1 時間を超えています。 シャットダウン: システム、サービス、またはデーモンのシャットタ 通信は「シャットダウン中」になりました。再起動やツールのアッ タスは最初のレポートサイクルで別の状態に変わります。 実行中: データ収集が有効になっています。ツールは正常に機能して
Hostname		エージェントの場合、エージェントがインストールされているホス された任意のホスト名です。 コネクタの場合は該当しません。
IP アドレス		収集ツールのインストール先の事前設定されたリストから選択され
コネクタ/エージェント ID	==	収集ツールのインストール元の事前設定されたリストから選択され ト ID です。

検索フィルタを適用してデータコレクタをソートするには

1. ナビゲーションペインで、[Data Collectors (データコレクタ)] を選択します。
2. [Connectors (コネクタ)] タブまたは [Agents (エージェント)] タブを選択します。
3. 検索バー内をクリックし、リストから検索条件を選択します。
4. 次のリストから演算子を選択します。
5. 最後のリストから値を選択します。

サーバーデータの表示、エクスポート、および検索

[Servers (サーバー)] ページには、データ収集ツールが認識している各サーバーインスタンスのシステム設定およびパフォーマンスのデータが表示されます。ここで、サーバー情報の表示、フィルタを使用したサーバーのソート、キーと値のペアを使用したサーバーのタグ付け、およびサーバーとシステムの詳細情報のエクスポートを行うことができます。

トピック

サーバーの表示とソート

データ収集ツールで検出したサーバーの情報を表示し、フィルタを使用してサーバーをソートできます。

サーバーの表示

データ収集ツールで検出したサーバーの全般表示と詳細表示を取得できます。

検出したサーバーを表示するには

1. ナビゲーションペインで、[Servers (サーバー)] を選択します。検出したサーバーがサーバリストに表示されます。
2. 各サーバーの詳細情報を表示するには、[Server info (サーバー情報)] 列でサーバーのリンクを選択します。このサーバーを説明する画面が表示されます。

サーバーの詳細画面には、システムとパフォーマンスのメトリクスが表示されます。ネットワークの依存関係やプロセスの情報をエクスポートするためのボタンも表示されます。サーバーの詳細情報をエクスポートするには、「サーバーデータのエクスポート (p. 57)」を参照してください。

検索フィルタによるサーバーのソート

特定のサーバーを簡単に見つけるには、収集ツールで検出したすべてのサーバーに検索フィルタを適用してソートします。検索とフィルタ処理は、さまざまな条件で実行できます。

検索フィルタを適用してサーバーをソートするには

1. ナビゲーションペインで、[Servers (サーバー)] を選択します。
2. 検索バー内をクリックし、リストから検索条件を選択します。
3. 次のリストから演算子を選択します。

4. 選択した検索条件の値を大文字と小文字を区別して入力し、Enter キーを押します。
5. 複数のフィルタを適用するには、ステップ 2〜4 を繰り返します。

サーバーのタグ付け

移行計画と情報の整理に役立てるために、サーバーごとに複数のタグを作成できます。タグは、ユーザー定義のキーと値のペアであり、サーバーに関するカスタムデータやメタデータを保存できます。個別または複数のサーバーに 1 回のオペレーションでタグを付けることができます。Application Discovery Service スタグは AWS タグと似ていますが、この 2 タイプ相互でタグを交換することはできません。

メイン [サーバー] ページから複数のタグを 1 つ以上のサーバーに対して追加または削除できます。選択したサーバーに対して 1 つ以上のタグを追加または削除するには、サーバーの詳細ページを使用します。複数のサーバーに対するタグ付け作業は、作業の種類を問わず、1 回のオペレーションで実行できます。また、タグを削除することもできます。

1 つ以上のサーバーにタグを追加するには

1. ナビゲーションペインで、[Servers (サーバー)] を選択します。
2. [Server info (サーバー情報)] 列で、タグを追加するサーバーのリンクを選択します。複数のサーバーに同時にタグを追加するには、各サーバーのチェックボックス内をクリックします。
3. [Add tag] を選択します。
4. ダイアログボックスで、[Key (キー)] フィールドに値を入力し、必要に応じて [Value (値)] フィールドに値を入力します。

   さらにタグを追加するには、[Additional tag (追加のタグ)] を選択して情報を追加します。
5. [Add Tags (タグの追加)] を選択します。緑色の確認メッセージが画面の上部に表示されます。
6. 個別のサーバーの詳細ページからタグを追加するには、[Actions (アクション)]、[Add tag (タグの追加)] の順に選択し、上記のステップを繰り返します。

1 つ以上のサーバーからタグを追加するには

1. ナビゲーションペインで、[Servers (サーバー)] を選択します。
2. [Server info (サーバー情報)] 列で、タグを削除するサーバーのリンクを選択します。複数のサーバーから同時にタグを削除するには、各サーバーのチェックボックス内をクリックします。
3. [Actions (アクション)]、[Remove tag (タグの削除)] の順に選択します。
4. 削除する各タグを選択するか、[select all (すべて選択)] を選択します。
5. [削除] を選択します。緑色の確認メッセージが画面の上部に表示されます。
6. 個別のサーバーの詳細ページからタグを削除するには、[Actions (アクション)]、[Remove tag (タグの削除)] の順に選択し、上記のステップを繰り返します。

サーバーデータのエクスポート

1 つのサーバーのネットワーク依存関係とプロセスの情報をエクスポートするには、サーバーの詳細画面を使用できます。サーバーのエクスポートジョブは、サーバーの詳細画面で [Exports (エクスポート)] セクションのテーブルにあります。まだエクスポートジョブがない場合、テーブルは空になります。データ収集を最大 5 つまで同時にエクスポートできます。

> **Note**
>
> コンソールからエクスポートできるサーバーデータは、そのサーバーで実行されているエージェントで収集されたものに限ります。コネクタで収集されたデータをダウンロードする場合は、「すべてのサーバーに関するシステムパフォーマンスデータのエクスポート (p. 41)」を参照し

てください。エージェントがインストールされているすべてのサーバーからデータを一括エクスポートする場合は、「Amazon Athena でのデータ探索 (p. 43)」を参照してください。

サーバーの詳細データをエクスポートするには

1. ナビゲーションペインで、[Servers (サーバー)] を選択します。
2. [Server info (サーバー情報)] 列で、データをエクスポートするサーバーの ID を選択します。
3. 画面下部の [Exports (エクスポート)] セクションで、[Export server details (サーバー詳細のエクスポート)] を選択します。
4. [Export server details (サーバー詳細のエクスポート)] で、[Start date (開始日)] と [Time (時刻)] を入力します。

   Note

   開始時刻は、現在の時刻から 72 時間より前にすることはできません。
5. ジョブを開始するには、[Export (エクスポート)] を選択します。最初のステータスは [In-progress (進行中)] です。ステータスを更新するには、[Exports (エクスポート)] セクションの更新アイコンをクリックします。
6. エクスポートジョブが完了したら、[Download (ダウンロード)] を選択して .zip ファイルを保存します。
7. 保存されたファイルを解凍します。エクスポートデータは、次のような .csv ファイルのセットに含まれています。

   - *<AWS account ID>*_destinationProcessConnection.csv
   - *<AWS account ID>*_networkInterface.csv
   - *<AWS account ID>*_osInfo.csv
   - *<AWS account ID>*_process.csv
   - *<AWS account ID>*_sourceProcessConnection.csv
   - *<AWS account ID>*_systemPerformance.csv

   .csv ファイルを Microsoft Excel で開き、エクスポートしたサーバーデータを確認できます。

   複数のファイルの 1 つは JSON ファイルであり、これにはエクスポートタスクとその結果に関するデータが含まれています。

Athena でのデータ探索

Amazon Athena でのデータ探索では、Discovery Agent によって検出された、すべてのオンプレミスサーバーから収集されたデータを 1 つの場所で分析できます。Migration Hub コンソールから (または、StartContinousPort API を使用して) Amazon Athena でのデータ探索が有効になり、エージェントのデータ収集が有効になり、エージェントによって収集されたデータは、定期的に S3 バケットに自動的に保存されます。詳細については、「Amazon Athena でのデータ探索 (p. 43)」を参照してください。

Applications

一部の検出したサーバーは、グループとして移行することで、引き続き動作できます。この場合、検出したサーバーをアプリケーションとして論理的に定義してグループ化できます。

グループ化のプロセスの一環として、タグの検索、フィルタ処理、および追加を行うことができます。

サーバーを新規または既存のアプリケーションにグループ化するには

1. ナビゲーションペインで、[Servers (サーバー)] を選択します。

2. サーバーリストで、新規または既存のアプリケーションにグループ化する各サーバーを選択します。

   グループに含めるサーバーを選択しやすくするために、サーバーリストで任意の条件を指定して検索およびフィルタできます。検索バー内をクリックしてリストから項目を選択し、次のリストから演算子を選択して、条件を入力します。
3. オプション: 選択したサーバーごとに、タグの追加で、[] に値を入力します。キーと入力し、オプションで値。
4. [Group as application (アプリケーションとしてグループ化する)] を選択してアプリケーションを作成します。または、既存のアプリケーションに追加します。
5. [Group as application (アプリケーションとしてグループ化する)] ダイアログボックスで、[Group as a new application (新規アプリケーションとしてグルーかする)] または [Add to an existing application (既存のアプリケーションに追加する)] を選択し ます。

   a. [Group as a new application (新規アプリケーションとしてグルーかする)] を選択した場合は、[Application name (アプリケーション名)] に名前を入力します。必要に応じて、[Application description (アプリケーションの説明)] に説明を入力できます。

   b. [Add to an existing application (既存のアプリケーション追加する)] を選択した場合は、リストで追加先のアプリケーションの名前を選択します。
6. [Save] (保存) をクリックします。

検出された構成項目のクエリ

構成項目は、エージェントまたはコネクタによってデータセンター内で検出された IT アセットです。Application Discovery Service を使用すると、フィルタを指定し、サーバー、アプリケーション、プロセス、および接続アセットの特定の構成項目をクエリできます。

次のセクションの表に、2 つの Application Discovery Service アクションで使用できる入力フィルタと出力ソートのオプションを示します。

- DescribeConfigurations
- ListConfigurations

フィルタリングおよびソートのオプションは、適用するアセットのタイプ (サーバー、アプリケーション、プロセス、接続) 別に整理されています。

の使用DescribeConfigurationsAction

DescribeConfigurations アクションは、設定 ID のリストの属性を取得します。提供される ID はすべて、アセットタイプ (サーバー、アプリケーション、プロセス、または接続) が同じである必要があります。出力フィールドは、選択されたアセットタイプに固有です。たとえば、サーバー設定項目の出力には、ホスト名、オペレーティングシステム、ネットワークカード数など、サーバーに関する属性のリストが含まれています。コマンド構文の詳細については、「DescribeConfigurations」を参照してください。

-DescribeConfigurationsアクションはフィルタリングをサポートしていません。

の出力フィールド**DescribeConfigurations**

以下の表は、アセットタイプ別に整理された、DescribeConfigurations アクションでサポートされる出力フィールドの一覧です。必須とマークされたものは、常に出力に存在します。

サーバーアセット

フィールド	必須
server.agentId	
server.applications	
server.applications.hasMoreValues	
server.configurationId	x
server.cpuType	
server.hostName	
server.hypervisor	
server.networkInterfaceInfo	
server.networkInterfaceInfo.hasMoreValues	
server.osName	
server.osVersion	

フィールド	必須
server.tags	
server.tags.hasMoreValues	
server.timeOfCreation	x
server.type	
server.performance.avgCpuUsagePct	
server.performance.avgDiskReadIOPS	
server.performance.avgDiskReadsPerSecondInKB	
server.performance.avgDiskWriteIOPS	
server.performance.avgDiskWritesPerSecondInKB	
server.performance.avgFreeRAMInKB	
server.performance.avgNetworkReadsPerSecondInKB	
server.performance.avgNetworkWritesPerSecondInKB	
server.performance.maxCpuUsagePct	
server.performance.maxDiskReadIOPS	
server.performance.maxDiskReadsPerSecondInKB	
server.performance.maxDiskWriteIOPS	
server.performance.maxDiskWritesPerSecondInKB	
server.performance.maxNetworkReadsPerSecondInKB	
server.performance.maxNetworkWritesPerSecondInKB	
server.performance.minFreeRAMInKB	
server.performance.numCores	
server.performance.numCpus	
server.performance.numDisks	
server.performance.numNetworkCards	
server.performance.totalRAMInKB	

アセットの処理

フィールド	必須
process.commandLine	
process.configurationId	x
process.name	

フィールド	必須
process.path	
process.timeOfCreation	x

アプリケーションアセット

フィールド	必須
application.configurationId	x
application.description	
application.lastModifiedTime	x
application.name	x
application.serverCount	x
application.timeOfCreation	x

の使用ListConfigurationsAction

ListConfigurations アクションは、フィルタで指定した条件に従って、構成項目のリストを取得します。コマンド構文の詳細については、「ListConfigurations」を参照してください。

ListConfigurations の出力フィールド

以下の表は、アセットタイプ別に整理された、ListConfigurations アクションでサポートされる出力フィールドの一覧です。必須とマークされたものは、常に出力に存在します。

サーバーアセット

フィールド	必須
server.configurationId	x
server.agentId	
server.hostName	
server.osName	
server.osVersion	
server.timeOfCreation	x
server.type	

アセットの処理

フィールド	必須
process.commandLine	

フィールド	必須
process.configurationId	x
process.name	
process.path	
process.timeOfCreation	x
server.agentId	
server.configurationId	x

アプリケーションアセット

フィールド	必須
application.configurationId	x
application.description	
application.name	x
application.serverCount	x
application.timeOfCreation	x
application.lastModifiedTime	x

接続アセット

フィールド	必須
connection.destinationIp	x
connection.destinationPort	x
connection.ipVersion	x
connection.latestTimestamp	x
connection.occurrence	x
connection.sourceIp	x
connection.transportProtocol	
destinationProcess.configurationId	
destinationProcess.name	
destinationServer.configurationId	
destinationServer.hostName	
sourceProcess.configurationId	
sourceProcess.name	

フィールド	必須
sourceServer.configurationId	
sourceServer.hostName	

ListConfigurations でサポートされているフィルタ

以下の表は、アセットタイプ別に整理された、ListConfigurations アクションでサポートされるフィルタの一覧です。フィルタと値は、サポートされている論理条件のいずれかによって定義されたキー/値の関係にあります。指定したフィルタの出力は並べ替えることができます。

サーバーアセット

フィルタ	サポートされる条件	サポートされる値	サポートされるソート
server.configurationId	• EQUALS • NOT_EQUALS • EQ • NE	• 任意の有効なサーバ設定 ID	なし
server.hostName	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
server.osName	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
server.osVersion	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
server.agentId	• EQUALS • NOT_EQUALS • EQ • NE	• 文字列	なし
server.connectorId	• EQUALS • NOT_EQUALS • EQ • NE	• 文字列	なし

フィルタ	サポートされる条件	サポートされる値	サポートされるソート
server.type	• EQUALS • NOT_EQUALS • EQ • NE	次のいずれかの値を持つ文字列: • EC2 • OTHER • VMWARE_VM • VMWARE_HOST • VMWARE_VM_TEMPLATE	なし
server.vmWareInfo.moref	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	なし
server.vmWareInfo.vcenterUuid	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	なし
server.vmWareInfo.hostId	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	なし
server.networkInterfaceInfo.portGroupId	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	なし
server.networkInterfaceInfo.portGroupName	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	なし

フィルタ	サポートされる条件	サポートされる値	サポートされるソート
server.networkInterfaceInfo.virtualSwitchName	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	なし
server.networkInterfaceInfo.ipAddress	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	なし
server.networkInterfaceInfo.macAddress	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	なし
server.performance.avgCpuUsagePct	• GE • LE • GT • LT	• 割合 (%)	なし
server.performance.totalDiskFreeSizeInKB	• GE • LE • GT • LT	• Double	なし
server.performance.avgFreeRAMInKB	• GE • LE • GT • LT	• Double	なし
server.tag.value	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	なし

フィルタ	サポートされる条件	サポートされる値	サポートされるソート
server.tag.key	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	なし
server.application.name	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	なし
server.application.description	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	なし
server.application.configurationId	• EQUALS • NOT_EQUALS • EQ • NE	• 任意の有効なアプリケーション構成 ID	なし
server.process.configurationId	• EQUALS • NOT_EQUALS • EQ • NE	• ProcessId	なし
server.process.name	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	なし
server.process.commandLine	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	なし

アプリケーションアセット

フィルタ	サポートされる条件	サポートされる値	サポートされるソート
application.configurationId	• EQUALS • NOT_EQUALS • EQ • NE	• ApplicationId	なし
application.name	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
application.description	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
application.serverCount	フィルタリングはサポートされていません。	フィルタリングはサポートされていません。	• ASC • DESC
application.timeOfCreation	フィルタリングはサポートされていません。	フィルタリングはサポートされていません。	• ASC • DESC
application.lastModifiedTime	フィルタリングはサポートされていません。	フィルタリングはサポートされていません	• ASC • DESC
server.configurationId	• EQUALS • NOT_EQUALS • EQ • NE	• serverId	なし

アセットの処理

フィルタ	サポートされる条件	サポートされる値	サポートされるソート
process.configurationId	• EQUALS • NOT_EQUALS • EQ • NE	• ProcessId	

フィルタ	サポートされる条件	サポートされる値	サポートされるソート
process.name	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
process.commandLine	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
server.configurationId	• EQUALS • NOT_EQUALS • EQ • NE	• serverId	
server.hostName	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
server.osName	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
server.osVersion	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
server.agentId	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	

接続アセット

フィルタ	サポートされる条件	サポートされる値	サポートされるソート
connection.sourceIp	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• IP	• ASC • DESC
connection.destinationIp	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• IP	• ASC • DESC
connection.destinationPort	• EQUALS • NOT_EQUALS • EQ • NE	• 整数	• ASC • DESC
sourceServer.configuredId	• EQUALS • NOT_EQUALS • EQ • NE	• serverId	
sourceServer.hostName	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
destinationServer.osName	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
destinationServer.osVersion	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
destinationServer.agentId	• EQUALS	• 文字列	

フィルタ	サポートされる条件	サポートされる値	サポートされるソート
	• NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS		
sourceProcess.configurationId	• EQUALS • NOT_EQUALS • EQ • NE	• ProcessId	
sourceProcess.name	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
sourceProcess.commandLine	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
destinationProcess.configurationId	• EQUALS • NOT_EQUALS • EQ • NE	• ProcessId	
destinationProcess.name	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC
destinationprocess.commandLine	• EQUALS • NOT_EQUALS • EQ • NE • CONTAINS • NOT_CONTAINS	• 文字列	• ASC • DESC

AWS Application Discovery Service でのセキュリティ

AWS では、クラウドのセキュリティが最優先事項です。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利点を得られます。

セキュリティは、AWS とお客様の間の共有責任です。共有責任モデルでは、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して説明しています。

- クラウドのセキュリティ–AWSを実行するインフラストラクチャを保護する責任を担います。AWSのサービスAWSクラウドです。AWSまた、安全に使用できるサービスも用意されています。セキュリティの有効性は、AWS 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審査機関によって定期的にテストおよび検証されています。
- クラウドのセキュリティ— お客様の責任はAWSサービスを使用します。また、お客様は、お客様のデータの機密性、組織の要件、および適用可能な法律および規制などの他の要因についても責任を担います。

このドキュメントは、Application Discovery Service を使用する際に適用される責任共有モデルについての理解の助けとなることを目的としています。以下のトピックでは、セキュリティとコンプライアンスの目的を達成するように Application Discovery Service を設定する方法を示します。また、Application Discovery Service 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説明します。

トピック

AWS Application Discovery Service 向けの Identity and Access Management

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コントロールするために役立つ AWS のサービスです。IAM 管理者は、認証済み(サインイン) と承認済み(アクセス許可を持つ) Application Discovery Service リソースを使用します。IAM は、追加料金なしで使用できる AWS のサービスです。

トピック

- AWS Application Discovery Service の Identity and Access のトラブルシューティング (p. 91)

Audience

使用方法AWS Identity and Access Management(IAM) は、Application Discovery Service で行う作業によって異なります。

サービスユーザー— Application Discovery Service を使用してジョブを実行する場合は、必要なアクセス許可と認証情報を管理者が用意します。作業を行うためにより多くの Application Discovery Service 機能を使用するにつれて、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、管理者から適切なアクセス許可をリクエストするのに役に立ちます。Application Discovery Service の機能にアクセスできない場合は、」を参照してください。AWS Application Discovery Service の Identity and Access のトラブルシューティング (p. 91)。

サービス管理者— 社内の Application Discovery Service リソースを担当している場合は、通常、Application Discovery Service へのフルアクセスがあります。従業員は、従業員にアクセスを許可する Application Discovery Service 機能とリソースを決定します。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許可を変更する必要があります。このページの情報を確認して、IAM の基本概念を理解してください。会社で IAM を利用する方法の詳細については、を参照してください。AWS Application Discovery Service で IAM を使用する方法 (p. 77)。

IAM 管理者IAM 管理者は、Application Discovery Service へのアクセスを管理するポリシーの作成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 Application Discovery Service 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。AWS Application Discovery Service アイデンティティベースのポリシーの例 (p. 80)。

アイデンティティを使用した認証

認証は、アイデンティティ認証情報を使用して AWS にサインインする方法です。AWS Management Console を使用したサインインの詳細については、IAM ユーザーガイドの「IAM ユーザーまたはルートユーザーとしての AWS Management Console へのサインイン」を参照してください。

AWS アカウントのルートユーザーもしくは IAM ユーザーとして、または IAM ロールを引き受けることによって認証される (AWS にサインインする) 必要があります。会社のシングルサインオン認証を使用することも、Google や Facebook を使用してサインインすることもできます。このような場合、管理者は以前に IAM ロールを使用して ID フェデレーションを設定しました。他の会社の認証情報を使用して AWS にアクセスした場合、ロールを間接的に割り当てられています。

AWS Management Console に直接サインインするには、ルートユーザーの E メールまたは IAM ユーザー名とパスワードを使用します。ルートユーザーまたは IAM ユーザーのアクセスキーを使用して AWS にプログラム的にアクセスできます。AWS では、SDK とコマンドラインツールを提供し、お客様の認証情報を使用して、リクエストに暗号で署名できます。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。これには、インバウンド API リクエストを認証するためのプロトコル、署名バージョン 4 を使用します。リクエストの認証の詳細については、AWS の全般リファレンスの「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、追加のセキュリティ情報の提供を要求される場合もあります。たとえば、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高めることを推奨しています。詳細については、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

AWS アカウント を初めて作成する場合は、このアカウントのすべての AWS のサービスとリソースに対して完全なアクセス権限を持つシングルサインインアイデンティティで始めます。このアイデンティティは 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メールアドレスとパスワードでのサインインによりアクセスされます。強くお勧めしているのは、日常的なタスクには、それが管理者

タスクであっても、ルートユーザーを使用しないことです。代わりに、最初の IAM ユーザーを作成するためにのみ、ルートユーザーを使用するというベストプラクティスに従います。その後、ルートユーザーの認証情報を安全な場所に保管し、それらを使用して少数のアカウントおよびサービス管理タスクのみを実行します。

IAM ユーザーとグループ

IAM ユーザーは、単一の人物またはアプリケーションに対する特定の許可を持つ AWS アカウント内のアイデンティティです。IAM ユーザーは、ユーザー名とパスワード、アクセスキーのセットなど、長期的な認証情報を持つことができます。アクセスキーを生成する方法の詳細については、IAM ユーザーガイドの「IAM ユーザーのアクセスキーの管理」を参照してください。IAM ユーザーにアクセスキーを生成するとき、必ずキーペアを表示して安全に保存してください。後になって、シークレットアクセスキーを回復することはできません。新しいアクセスキーペアを生成する必要があります。

IAM グループは、IAM ユーザーのコレクション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、一度に複数のユーザーに対してアクセス許可を指定できます。多数の組のユーザーがある場合、グループを使用すると管理が容易になります。たとえば、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アクセス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 人の特定の人または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利用できます。詳細については、IAM ユーザーガイドの「IAM ユーザーの作成が適している場合 (ロールではなく)」を参照してください。

IAM ロール

IAM ロールは、特定のアクセス許可を持つ、AWS アカウント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。AWS Management Consoleロールを切り替えることによって、 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、AWS CLI または AWS API 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。

- 一時的な IAM ユーザーアクセス許可 – IAM ユーザーは、特定のタスクに対して複数の異なるアクセス許可を一時的に IAM ロールで引き受けることができます。
- フェデレーティッドユーザーアクセス – IAM ユーザーを作成する代わりに、AWS Directory Service、エンタープライズユーザーディレクトリ、またはウェブ ID プロバイダーからの既存のアイデンティティを使用できます。このようなユーザーはフェデレーティッドユーザーと呼ばれます。AWS では、ID プロバイダーを通じてアクセスがリクエストされたとき、フェデレーティッドユーザーにロールを割り当てます。フェデレーティッドユーザーの詳細については、IAM ユーザーガイドのフェデレーティッドユーザーとロールを参照してください。
- 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。ロールは、クロスアカウントアクセスを許可する主な方法です。ただし、一部の AWS のサービスでは、(ロールをプロキシとして使用する代わりに) リソースにポリシーを直接アタッチできます。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。
- クロスサービスアクセス – 一部の AWS のサービスは、AWS の他のサービスの機能を使用します。たとえば、サービスで呼び出しを行う場合、そのサービスでは Amazon EC2 でアプリケーションを実行したり、Amazon S3 にオブジェクトを保存したりするのが一般的です。サービスは、呼び出し元プリンシパルのアクセス許可、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。
  - プリンシパル許可 – IAM ユーザーまたはロールを使用して AWS でアクションを実行する場合、そのユーザーはプリンシパルとみなされます。ポリシーは、プリンシパルにアクセス許可を付与します。

一部のサービスを使用する場合、別のサービスで別のアクションをトリガーするアクションを実行することがあります。この場合、両方のアクションを実行するための許可が必要です。アクションがポリシーで追加の依存アクションを必要とするかどうかを確認するには、サービス認証リファレンスのAWS Application Discovery Service のアクション、リソース、および条件キー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権限を委任するロールの作成」を参照してください。
- サービスリンクロール – サービスリンクロールは、AWS のサービスにリンクされたサービスロールの一種です。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、IAM アカウント内に表示され、サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールのアクセス許可を表示できますが、編集することはできません。

- Amazon EC2 で実行されているアプリケーション – EC2 インスタンスで実行され、AWS CLI または AWS API リクエストを作成しているアプリケーションの一時的な認証情報を管理するには、IAM ロールを使用できます。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認証情報を取得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「Amazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用してアクセス許可を付与する」を参照してください。

IAM ロールを使用するか IAM ユーザーを使用するかどうかについては、IAM ユーザーガイドの IAM ユーザーの作成が適している場合 (ロールではなく)」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

AWS でのアクセスは、ポリシーを作成し、それらを IAM アイデンティティまたは AWS リソースにアタッチすることで制御できます。ポリシーは AWS のオブジェクトであり、ID やリソースに関連付けて、これらのアクセス許可を定義します。ルートユーザーまたは IAM ユーザーとしてサインインすることも、IAM ロールを引き受けることもできます。リクエストを行うと、AWS は関連するアイデンティティベースまたはリソースベースのポリシーを評価します。ポリシーでのアクセス許可により、リクエストが許可されるか拒否されるかが決まります。大半のポリシーは JSON ドキュメントとして AWS 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、誰が何にアクセスす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、アクセス許可のない状態からスタートします。言い換えると、デフォルト設定では、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。何かを実行するアクセス許可をユーザーに付与するには、管理者がユーザーにアクセス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、必要なアクセス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アクセス許可を付与すると、そのグループ内のすべてのユーザーにこれらのアクセス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、アクションのアクセス許可を定義します。たとえば、`iam:GetRole` アクションを許可するポリシーがあるとします。このポリシーがあるユーザーは、AWS Management Console、AWS CLI、または AWS API からロールの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のポリシーは、IAM user 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 アクセス許可ポリシードキュメントです。これらのポリシーは、ユー

ザーとロールが実行できるアクション、リソース、および条件を制御します。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管理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ています。管理ポリシーは、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。AWS アカウント。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。管理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 ユーザーガイドの管理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、リソースにアタッチする 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の例は、IAM ロールの信頼ポリシーおよび Amazon S3 バケットポリシーです。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、ポリシーは、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を定義します。リソースベースのポリシーで、プリンシパルを指定する必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーティッドユーザー、または AWS のサービスを含めることができます。

リソースベースのポリシーは、そのサービス内にあるインラインポリシーです。リソースベースのポリシーで IAM からの AWS 管理ポリシーを使用することはできません。

アクセスコントロールリスト (ACL)

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制御します。ACL はリソースベースのポリシーに似ていますが、JSON ポリシードキュメント形式は使用しません。

Amazon S3、AWS WAF、および Amazon VPC は、ACL をサポートするサービスの例です。ACL の詳細については、Amazon Simple Storage Service デベロッパーガイドの「アクセスコントロールリスト (ACL) の概要」を参照してください。

その他のポリシータイプ

AWS では、別のあまり一般的ではないポリシータイプもサポートしています。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許可を設定できます。

- 許可の境界 – 許可の境界は、ID ベースのポリシーが IAM エンティティ (IAM ユーザーまたはロール) に付与できる許可の上限を設定する高度な機能です。エンティティの許可の境界を設定できます。結果として得られる許可は、エンティティの ID ベースのポリシーとその許可の境界の共通部分です。`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は、アクセス許可の境界では制限されません。これらのポリシーのいずれかを明示的に拒否した場合、その許可は無効になります。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 (SCP) – SCP は、AWS Organizations の組織または組織単位 (OU) の最大許可を指定する JSON ポリシーです。AWS Organizations は、ビジネスが所有する複数の AWS アカウントをグループ化し、一元的に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、サービス制御ポリシー (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、メンバーアカウントのエンティティに対するアクセス許可を制限します (各など)。AWS アカウントルートユーザー。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ティッドユーザーの一時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。結果として得られるセッションの許可は、ユーザーまたはロールの ID ベースのポリシーとセッションポリシーの共通部分

です。また、リソースベースのポリシーから許可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、その許可は無効になります。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許可を理解するのがさらに複雑になります。複数のポリシータイプが関連するとき、リクエストを許可するかどうかを AWS が決定する方法の詳細については、IAM ユーザーガイド のポリシーの評価ロジックを参照してください。

AWS Application Discovery Service で IAM を使用する方法

IAM を使用してApplication Discovery Service へのアクセスを管理する前に、Application Discovery Service で使用できる IAM 機能について理解しておく必要があります。Application Discovery Service などの高レベルのビューを取得するにはAWSサービスについては、「」を参照してください。AWSIAM と連携するサービスのIAM ユーザーガイド。

トピック

Application Discovery Service ID ベースのポリシー

IAM ID 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、またアクションを許可または拒否する条件を指定できます。Application Discovery Service では、特定のアクション、リソース、および条件キーをサポートしています。JSON ポリシーで使用す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。

Actions

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。

JSON ポリシーの `Action` 要素は、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を記述します。ポリシーアクションの名前は通常、関連する AWS API オペレーションと同じです。一致する API オペレーションを持たないアクセス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、関連付けられた操作を実行するための許可を付与するポリシーで使用されます。

Application Discovery Service のポリシーアクションは、アクションの前にプレフィックス () を使用します。`discovery:`。 ポリシーステートメントには、`Action`または`NotAction`要素。Application Discovery Service は、このサービスで実行できるタスクを記述する独自のアクションのセットを定義します。

単一のステートメントに複数のアクションを指定するには、次のようにコンマで区切ります。

```
"Action": [
```

```
      "discovery:action1",
      "discovery:action2"
```

ワイルドカード (*) を使用して複数のアクションを指定することができます。例えば、Describe という単語で始まるすべてのアクションを指定するには、次のアクションを含めます。

```
"Action": "discovery:Describe*"
```

Application Discovery Service のアクションの一覧を表示するには、」を参照してください。で定義されるアクションAWS Application Discovery ServiceのIAM ユーザーガイド。

Resources

Application Discovery Service は、ポリシー内のリソース ARN の指定をサポートしていません。

条件キー

Application Discovery Service にはサービス固有条件キーがありませんが、いくつかのグローバル条件キーの使用がサポートされています。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。

Examples

Application Discovery Service 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、」AWS Application Discovery Service アイデンティティベースのポリシーの例 (p. 80)。

Application Discovery Service のリソースベースのポリシー

Application Discovery Service は、リソースベースのポリシーをサポートしていません。

Application Discovery Service スタグに基づく承認

Application Discovery Service は、リソースのタグ付けやタグに基づいたアクセスの制御をサポートしていません。

Application Discovery Service IAM ロール

IAM ロールは、特定のアクセス許可を持つ AWS アカウント内のエンティティです。

Application Discovery Service による一時的な認証情報の使用

Application Discovery Service では、一時的な認証情報の使用をサポートしていません。

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によって、AWS サービスが他のサービスのリソースにアクセスして自動的にアクションを完了できます。サービスにリンクされたロールは、IAM アカウント内に表示され、サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールのアクセス許可を表示できますが、編集することはできません。

Application Discovery Service はサービスにリンクされたロールをサポートします Application Discovery Service 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、」Application Discovery Service のサービスにリンクされたロールの使用 (p. 86)。

サービスロール

この機能では、サービスのロールをユーザーに代わって引き受けることをサービスに許可します。このロールにより、サービスはユーザーに代わって他のサービスのリソースにアクセスし、アクションを実行できます。サービスロールは、IAM アカウントに表示され、サービスによって所有されます。つまり、IAM 管理者は、このロールのアクセス許可を変更できます。ただし、これを行うことにより、サービスの機能が損なわれる場合があります。

Application Discovery Service はサービスのロールをサポートします

Application Discovery Service での AWS 管理 (事前定義) ポリシー

AnAWS管理ポリシーである。AWS Identity and Access Managementによって作成および管理される (IAM) ポリシーAWS。管理ポリシーは、一般的なユースケースに必要なアクセス許可を付与するように事前定義されています。管理ポリシーの詳細については、」を参照してください。管理ポリシーとインラインポリシーのIAM ユーザーガイド。

-AWSこのトピックに一覧表示されている管理ポリシーは、AWS Application Discovery Service へのアクセスをコントロールするために使用されます。管理者AWSアカウントはデフォルトで、Application Discovery Service へのアクセスに必要なすべてのポリシーを継承します。

アカウントが管理者以外のアカウントである場合、Application Discovery Service にアクセスするには、以下の管理ポリシーを 1 つ以上追加することを管理者にリクエストする必要があります。管理ポリシーをアカウントにアタッチする方法については、「IAM 管理者以外のユーザーの作成 (p. 5)」を参照してください。

AWSApplicationDiscoveryServiceFullAccess

> -`AWSApplicationDiscoveryServiceFullAccess`ポリシーでは、IAM ユーザーアカウントに、Application Discovery Service Migration Hub API へのアクセスを許可します。
>
> このポリシーがアタッチされている IAM ユーザーアカウントは、Application Discovery Service の設定、エージェントの開始と停止、エージェントレス検出の開始と停止、AWSDiscovery Service Database このポリシーの例については、「Application Discovery Service へのフルアクセスの付与 (p. 81)」を参照してください。

AWSApplicationDiscoveryAgentAccess

> -`AWSApplicationDiscoveryAgentAccess`ポリシーは、Application Discovery Agent に、Application Discovery Service に登録して通信するためのアクセス許可
>
> このポリシーをアタッチする対象のユーザーは、その認証情報が Application Discovery Agent で使用されるすべてのユーザーです。
>
> このポリシーは、ユーザーに Arsenal へのアクセス権も付与します。Arsenal は、AWS で管理およびホストされるエージェントサービスです。Arsenal はデータをクラウド内の Application Discovery Service に転送します。このポリシーの例については、「Discovery Agent へのアクセスの許可 (p. 82)」を参照してください。

AWSAgentlessDiscoveryService

> -`AWSAgentlessDiscoveryService`ポリシーでは、AWSVMware vCenter Server で実行されている Agentless Discovery Connector に、Application Discovery Service を使用してコネクタヘルスメトリクスを登録、通信、共有するためのアクセス権を付与します。
>
> このポリシーをアタッチする対象のユーザーは、その認証情報がコネクタで使用されるすべてのユーザーです。

このポリシーの例については、「AWS Agentless Discovery Connector のアクセスの許可 (p. 82)」を参照してください。

ApplicationDiscoveryServiceContinuousExportServiceRolePolicy

IAM アカウントに`AWSApplicationDiscoveryServiceFullAccess`ポリシーのアタッチ済み`ApplicationDiscoveryServiceContinuousExportServiceRolePolicy`Amazon Athena でデータ探索をオンにすると、アカウントに自動的にアタッチされます。

このポリシーにより、AWS Application Discovery Service は Amazon Kinesis Data Firehose ストリームを作成し、AWS Application Discovery Service エージェントによって収集されたデータを、AWSアカウント.

さらに、このポリシーは、AWS Glue Data Catalogapplication_discovery_service_database という名前の新しいデータベースを使用する 、および、エージェントによって収集されたデータをマッピングするテーブルスキーマを作成します。このポリシーの例については、「 エージェントデータ収集のためのアクセス許可の付与 (p. 84)」を参照してください。

AWSDiscoveryContinuousExportFirehosePolicy

-`AWSDiscoveryContinuousExportFirehosePolicy`ポリシーは、Amazon Athena でデータ探索を使用するために必要です。これにより、Amazon Kinesis Data Firehose は、Application Discovery Service から収集したデータを Amazon S3 に書き込むことができます。このポリシーの使用方法については、「 AWSApplicationDiscoveryServiceFirehose ロールの作成 (p. 80)」を参照してください。このポリシーの例については、「 データ収集のためのアクセス許可の付与 (p. 85)」を参照してください。

AWSApplicationDiscoveryServiceFirehose ロールの作成

管理者は、お客様の IAM ユーザーアカウントに管理ポリシーをアタッチします。[] を使用する場合`AWSDiscoveryContinuousExportFirehosePolicy`ポリシーを作成する場合、管理者は最初にAWSApplicationDiscoveryServiceFirehoseを信頼されたエンティティとして Kinesis Data Firehose と結合し、`AWSDiscoveryContinuousExportFirehosePolicy`次の手順に示すように、ロールに追加します。

を作成するにはAWSApplicationDiscoveryServiceFirehoseIAM ロール

1. IAM コンソールで、[] を選択します。ロールナビゲーションペインで、[] を選択します。
2. [ロールの作成] を選択します。
3. [Kinesis] を選択します。
4. ユースケースとして、[Kinesis Firehose] を選択します。
5. [Next: (次へ:)] を選択します アクセス許可.
6. [フィルタポリシー] で、[AWSDiscoveryContinuousExportFirehosePolicy] を検索します。
7. 横のチェックボックスをオンにします。AWSDiscoveryContinuousExportFirehosePolicy[] を選択してから、[次へ: 確認.
8. [AWSApplicationDiscoveryServiceFirehose] をロール名として入力し、[ロールの作成] を選択します。

AWS Application Discovery Service アイデンティティベースのポリシーの例

IAM ユーザーおよびロールには、Application Discovery Service リソースを作成または変更するアクセス許可はありません。また、AWS Management Console や AWS CLI、AWS API を使用してタスクを実行することもできません。IAM 管理者は、ユーザーとロールに必要な、指定されたリソースで特定の API オ

ペレーションを実行するアクセス許可をユーザーとロールに付与する IAM ポリシーを作成する必要があります。続いて、管理者はそれらのアクセス許可が必要な IAM ユーザーまたはグループにそのポリシーをアタッチします。

JSON ポリシードキュメントのこれらの例を使用して、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「JSON タブでのポリシーの作成」を参照してください。

トピック

ポリシーのベストプラクティス

ID ベースのポリシーは非常に強力です。これらは、アカウント内で誰かが Application Discovery Service リソースの作成、アクセス、または削除を実行できるか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 アカウントに追加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のポリシーを作成または編集するときは、以下のガイドラインと推奨事項に従います。

- の使用を開始します。AWSマネージドポリシー— Application Discovery Service 使用を迅速に開始するには、AWS管理ポリシーを使用して、従業員に必要なアクセス許可を付与します。これらのポリシーはアカウントですでに有効になっており、 によって管理および更新されていますAWS 詳細については、IAM ユーザーガイドの「AWS 管理ポリシーを使用したアクセス許可の使用開始」を参照してください。
- 最小権限を付与する – カスタムポリシーを作成するときは、タスクの実行に必要な許可のみを付与します。最小限のアクセス許可から開始し、必要に応じて追加のアクセス許可を付与します。この方法は、寛容なアクセス許可で始め、後でそれらを強化しようとするよりも安全です。詳細については、IAM ユーザーガイドの「最小限の特権を認める」を参照してください。
- 機密性の高い操作に MFA を有効にする – 追加セキュリティとして、機密性の高いリソースまたは API 操作にアクセスするために IAM ユーザーに対して、多要素認証 (MFA) の使用を要求します。詳細については、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。
- 追加のセキュリティとしてポリシー条件を使用する – 実行可能な範囲内で、ID ベースのポリシーでリソースへのアクセスを許可する条件を定義します。例えば、要求が発生しなければならない許容 IP アドレスの範囲を指定するための条件を記述できます。指定された日付または時間範囲内でのみリクエストを許可する条件を書くことも、SSL や MFA の使用を要求することもできます。詳細については、「」を参照してください。IAM JSON ポリシー要素: 条件のIAM ユーザーガイド。

Application Discovery Service へのフルアクセスの付与

AWSApplicationDiscoveryServiceFullAccess 管理ポリシーでは、IAM ユーザーアカウントに、Application Discovery Service Migration Hub API へのアクセスを許可します。

このポリシーがアカウントにアタッチされている IAM ユーザーは、Application Discovery Service の設定、エージェントの開始と停止、エージェントレス検出の開始と停止、AWSDiscovery Service Database このポリシーの詳細については、Application Discovery Service での AWS 管理 (事前定義) ポリシー (p. 79) を参照してください。

Example AWSApplicationDiscoveryServiceFullAccess Policy

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "mgh:*",
                "discovery:*"
            ],
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Action": [
                "iam:GetRole"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Discovery Agent へのアクセスの許可

AWSApplicationDiscoveryAgentAccess 管理ポリシーでは、Application Discovery Agent に、Application Discovery Service 登録して通信するためのアクセス権を付与します。このポリシーの詳細については、Application Discovery Service での AWS 管理 (事前定義) ポリシー (p. 79) を参照してください。

このポリシーをアタッチする対象のユーザーは、その認証情報が Application Discovery Agent で使用されるすべてのユーザーです。

このポリシーは、ユーザーに Arsenal へのアクセス権も付与します。Arsenal は、AWS で管理およびホストされるエージェントサービスです。Arsenal はデータをクラウド内の Application Discovery Service に転送します。

Example AWSApplicationDiscoveryAgentAccess Policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "arsenal:RegisterOnPremisesAgent"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS Agentless Discovery Connector のアクセスの許可

AWSagentLessDiscoveryService 管理ポリシーは、AWSVMware vCenter Server で実行されている Agentless Discovery Connector。コネクタヘルスメトリクスの登録、やり取り、共有のためのアクセス権を付与します。このポリシーの詳細については、Application Discovery Service での AWS 管理 (事前定義) ポリシー (p. 79) を参照してください。

Example AWSAgentlessDiscoveryService Policy

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

```
"Statement": [
    {
        "Effect": "Allow",
        "Action": [
            "awsconnector:RegisterConnector",
            "awsconnector:GetConnectorHealth"
        ],
        "Resource": "*"
    },
    {
        "Effect": "Allow",
        "Action": "iam:GetUser",
        "Resource": "*"
    },
    {
        "Effect": "Allow",
        "Action": [
            "s3:GetObject",
            "s3:ListBucket"
        ],
        "Resource": [
            "arn:aws:s3:::connector-platform-upgrade-info/*",
            "arn:aws:s3:::connector-platform-upgrade-info",
            "arn:aws:s3:::connector-platform-upgrade-bundles/*",
            "arn:aws:s3:::connector-platform-upgrade-bundles",
            "arn:aws:s3:::connector-platform-release-notes/*",
            "arn:aws:s3:::connector-platform-release-notes",
            "arn:aws:s3:::prod.agentless.discovery.connector.upgrade/*",
            "arn:aws:s3:::prod.agentless.discovery.connector.upgrade"
        ]
    },
    {
        "Effect": "Allow",
        "Action": [
            "s3:PutObject",
            "s3:PutObjectAcl"
        ],
        "Resource": [
            "arn:aws:s3:::import-to-ec2-connector-debug-logs/*"
        ]
    },
    {
        "Effect": "Allow",
        "Action": [
            "SNS:Publish"
        ],
        "Resource": "arn:aws:sns:*:*:metrics-sns-topic-for-*"
    },
    {
        "Sid": "Discovery",
        "Effect": "Allow",
        "Action": [
            "Discovery:*"
        ],
        "Resource": "*"
    },
    {
        "Sid": "arsenal",
        "Effect": "Allow",
        "Action": [
            "arsenal:RegisterOnPremisesAgent"
        ],
        "Resource": "*"
    }
]
```

```
}
```

エージェントデータ収集のためのアクセス許可の付与

ApplicationDiscoveryServiceContinuousExportServiceRolePolicy では、AWS Application Discovery ServiceAmazon Kinesis Data Firehose ストリームを作成し、Application Discovery Service エージェントによって収集されたデータを変換して、AWSアカウント.

さらに、このポリシーでは、application_discovery_service_database という名前の新しいデータベースを使用する AWS Glue データカタログ、およびエージェントによって収集されたデータをマッピングするためのテーブルスキーマを作成します。

このポリシーの使用方法については、「Application Discovery Service での AWS 管理 (事前定義) ポリシー (p. 79)」を参照してください。

Example ApplicationDiscoveryServiceContinuousExportServiceRolePolicy

```
{
   "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "glue:CreateDatabase",
                "glue:UpdateDatabase",
                "glue:CreateTable",
                "glue:UpdateTable",
                "firehose:CreateDeliveryStream",
                "firehose:DescribeDeliveryStream",
                "logs:CreateLogGroup"
            ],
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Action": [
                "firehose:DeleteDeliveryStream",
                "firehose:PutRecord",
                "firehose:PutRecordBatch",
                "firehose:UpdateDestination"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:firehose:*:*:deliverystream/aws-application-discovery-
service*"
        },
        {
            "Action": [
                "s3:CreateBucket",
                "s3:ListBucket",
                "s3:PutBucketLogging",
                "s3:PutEncryptionConfiguration"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:s3:::aws-application-discovery-service*"
        },
        {
            "Action": [
                "s3:GetObject"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:s3:::aws-application-discovery-service*/*"
        },
        {
            "Action": [
```

```
                "logs:CreateLogStream",
                "logs:PutRetentionPolicy"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:logs:*:*:log-group:/aws/application-discovery-service/
firehose*"
        },
        {
            "Action": [
                "iam:PassRole"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:iam::*:role/AWSApplicationDiscoveryServiceFirehose",
            "Condition": {
                "StringLike": {
                    "iam:PassedToService": "firehose.amazonaws.com"
                }
            }
        },
        {
            "Action": [
                "iam:PassRole"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:iam::*:role/service-role/
AWSApplicationDiscoveryServiceFirehose",
            "Condition": {
                "StringLike": {
                    "iam:PassedToService": "firehose.amazonaws.com"
                }
            }
        }
    ]
}
```

データ収集のためのアクセス許可の付与

Amazon Athena でデータ探索を使用するには、AWSDiscoveryContinuousExportFirehosePolicy ポリシーが必要です。これにより、Amazon Kinesis Data Firehose は、Application Discovery Service から収集したデータを Amazon S3 に書き込むことができます。このポリシーの使用方法については、「AWSApplicationDiscoveryServiceFirehose ロールの作成 (p. 80)」を参照してください。

Example AWSDiscoveryContinuousExportFirehosePolicy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "glue:GetTableVersions"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:AbortMultipartUpload",
                "s3:GetBucketLocation",
                "s3:GetObject",
                "s3:ListBucket",
                "s3:ListBucketMultipartUploads",
                "s3:PutObject"
```

```
            ],
            "Resource": [
                "arn:aws:s3:::aws-application-discovery-service-*",
                "arn:aws:s3:::aws-application-discovery-service-*/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:PutLogEvents"
            ],
            "Resource": [
                "arn:aws:logs:*:*:log-group:/aws/application-discovery-service/
firehose:log-stream:*"
            ]
        }
    ]
}
```

Application Discovery Service のサービスにリンクされたロールの使用

AWS Application Discovery Service は AWS Identity and Access Management (IAM) サービスリンクロールを使用します。サービスにリンクされたロールは、Application Discovery Service に直接リンクされた一意のタイプの IAM ロールです。サービスにリンクされたロールは、Application Discovery Service によって事前定義されており、その他のAWSのサービスをお客様に代わります。

サービスにリンクされたロールを使用すると、必要なアクセス許可を手動で追加する必要がなくなるため、Application Discovery Service の設定が簡単になります。Application Discovery Service は、サービスにリンクされたロールのアクセス許可を定義します。特に定義されていない限り、Application Discovery Service のみがそのロールを引き受けることができます。定義される許可は、信頼ポリシーと許可ポリシーに含まれており、その許可ポリシーを他の IAM エンティティにアタッチすることはできません。

サービスリンクロールを削除するには、まずその関連リソースを削除します。これにより、リソースへの意図しないアクセスによるアクセス許可の削除が防止され、Application Discovery Service リソースが保護されます。

トピック

- Application Discovery Service のサービスにリンクされたロールのアクセス許可 (p. 86)
- Application Discovery Service のサービスにリンクされたロールの作成 (p. 89)
- Application Discovery Service のサービスにリンクされたロールの削除 (p. 90)

サービスリンクロールをサポートする他のサービスについては、「IAM と連携する AWS のサービス」でサービスリンクロール列がはいになっているサービスを検索してください。サービスにリンクされたロールに関するドキュメントをサービスで表示するには、[はい] リンクを選択します。

Application Discovery Service のサービスにリンクされたロールのアクセス許可

Application Discovery Service、と呼ばれるサービスにリンクされたロールを使用します。AWSservicerOLEアプリケーション検出サービスネットワークネットワークネットワークポート—によって使用または管理される AWS のサービスおよびリソースへのアクセスを可能にします。AWS Application Discovery Service。

AWSServiceRoleForApplicationDiscoveryServiceContinuousExport サービスにリンクされたロールは、ロールを引き受ける上で次のサービスを信頼します。

- `continuousexport.discovery.amazonaws.com`

ロールのアクセス許可ポリシーは、Application Discovery Service が次のアクションを完了することを許可します。

glue

`CreateDatabase`

`UpdateDatabase`

`CreateTable`

`UpdateTable`

firehose

`CreateDeliveryStream`

`DeleteDeliveryStream`

`DescribeDeliveryStream`

`PutRecord`

`PutRecordBatch`

`UpdateDestination`

s3

`CreateBucket`

`ListBucket`

`GetObject`

ログ

`CreateLogGroup`

`CreateLogStream`

`PutRetentionPolicy`

iam

`PassRole`

これは、上記のアクションが適用されるリソースを示す全ポリシーです。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "glue:CreateDatabase",
                "glue:UpdateDatabase",
                "glue:CreateTable",
                "glue:UpdateTable",
                "firehose:CreateDeliveryStream",
                "firehose:DescribeDeliveryStream",
```

```
                "logs:CreateLogGroup"
            ],
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Action": [
                "firehose:DeleteDeliveryStream",
                "firehose:PutRecord",
                "firehose:PutRecordBatch",
                "firehose:UpdateDestination"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:firehose:*:*:deliverystream/aws-application-discovery-
service*"
        },
        {
            "Action": [
                "s3:CreateBucket",
                "s3:ListBucket",
                "s3:PutBucketLogging",
                "s3:PutEncryptionConfiguration"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:s3:::aws-application-discovery-service*"
        },
        {
            "Action": [
                "s3:GetObject"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:s3:::aws-application-discovery-service*/*"
        },
        {
            "Action": [
                "logs:CreateLogStream",
                "logs:PutRetentionPolicy"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:logs:*:*:log-group:/aws/application-discovery-service/
firehose*"
        },
        {
            "Action": [
                "iam:PassRole"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:iam::*:role/AWSApplicationDiscoveryServiceFirehose",
            "Condition": {
                "StringLike": {
                    "iam:PassedToService": "firehose.amazonaws.com"
                }
            }
        },
        {
            "Action": [
                "iam:PassRole"
            ],
            "Effect": "Allow",
            "Resource": "arn:aws:iam::*:role/service-role/
AWSApplicationDiscoveryServiceFirehose",
            "Condition": {
                "StringLike": {
                    "iam:PassedToService": "firehose.amazonaws.com"
                }
            }
```

```
        }
    ]
}
```

サービスにリンクされたロールの作成、編集、削除を IAM エンティティ (ユーザー、グループ、ロールなど) に許可するには、アクセス許可を設定する必要があります。詳細については、IAM ユーザーガイドの「サービスリンクロールの許可」を参照してください。

Application Discovery Service のサービスにリンクされたロールの作成

サービスにリンクされたロールを手動で作成する必要はありません。継続的な暗黙的エクスポートをオンにするか、「データ収集の開始」を選択した後に、データコレクタページで表示されるダイアログボックスでオプションを確認するときに、AWSServiceRoleForApplicationsExport サービスリンクロールが自動的に作成されます。Athena でのデータ探索」、または b) AWS CLI を使用して StartContinuousExport API を呼び出すときに呼び出されます。

> Important
>
> このサービスにリンクされたロールがアカウントに表示されるのは、このロールでサポートされている機能を使用する別のサービスでアクションが完了した場合です。詳細については、「AWS アカウントに新しいロールが表示される」を参照してください。

Migration Hub コンソールからサービスにリンクされたロールを作成する

Migration Hub コンソールを使用して、AWSServiceRoleForApplicationDiscoveryServiceContinuousExport サービスにリンクされたロールを作成できます。

サービスにリンクされたロールを作成するには (コンソール)

1. ナビゲーションペインで、[Data Collectors (データコレクタ)] を選択します。
2. [Agents (エージェント)] タブを選択します。
3. [] を切り替えAthena でのデータ探索スライダーをオンに切り替えます。
4. 前のステップで作成したダイアログボックスで、関連するコストに同意するチェックボックスをオンにして、[Continue (続行)] または [Enable (有効)] を選択します。

AWS CLI からサービスにリンクされたロールを作成する

Application Discovery Service コマンドは、AWS Command Line InterfaceAWSServiceRoleForApplicationDiscoveryServiceContinuousExport サービスにリンクされたロールを作成します。

このサービスにリンクされたロールは、AWS CLI(AWS CLIはまず環境にインストールされている必要があります)。

コンソールから連続エクスポートを開始してサービスにリンクされたロール (CLI) を作成するには AWS CLI

1. オペレーティングシステム (Linux、macOS、または Windows) 用の AWS CLI をインストールします。フレームワークの使用の詳細については、AWS Command Line Interfaceユーザーガイド手順については、を参照してください。
2. コマンドプロンプト (Windows) またはターミナル (Linux/macOS) を開きます。
   a. `aws configure` を入力して、[Enter] を押します。
   b. AWS アクセスキー ID と AWS シークレットアクセスキーを入力します。

c. デフォルトのリージョン名として「us-west-2」と入力します。
d. デフォルトの出力形式として「text」と入力します。

3. 次のコマンドを入力します。

```
aws discovery start-continuous-export
```

IAM コンソールを使用して、サービスにリンクされたロールを作成するには、検出サービス - 継続的エクスポートユースケース。IAM CLI または IAM API で、サービスにリンクされたロールをサービス名 (continuousexport.discovery.amazonaws.com) で作成します。詳細については、IAM ユーザーガイドの「サービスにリンクされたロールの作成」を参照してください。このサービスにリンクされたロールを削除する場合、この同じプロセスを使用して、もう一度ロールを作成できます。

Application Discovery Service のサービスにリンクされたロールの削除

サービスリンクロールを必要とする機能またはサービスが不要になった場合には、そのロールを削除することをお勧めします。そうすることで、積極的にモニタリングまたは保守されていない未使用のエンティティを排除できます。ただし、手動で削除する前に、サービスにリンクされたロールをクリーンアップする必要があります。

サービスにリンクされたロールのクリーンアップ

IAM を使用してサービスにリンクされたロールを削除するには、最初に、そのロールで使用されているリソースをすべて削除する必要があります。

> Note
>
> リソースを削除する際に、Application Discovery Service でロールが使用されている場合、削除は失敗することがあります。失敗した場合は、数分待ってから再度オペレーションを実行してください。

AWSServiceRoleForApplicationDiscoveryServiceRoleForApplicationDiscoveryServiceContinuousExport サービスにリンクされたロールが使用する Application Discovery Service リソースを削除するには

1. ナビゲーションペインで、[Data Collectors (データコレクタ)] を選択します。
2. [Agents (エージェント)] タブを選択します。
3. [] を切り替えAthena でのデータ探索スライダーをオフに切り替えます。

AWSServiceRoleForApplicationDiscoveryServiceContinuousExport サービスにリンクされたロールが使用する Application Discovery Service リソースを削除するにはAWS CLI

1. オペレーティングシステム (Linux、macOS、または Windows) 用の AWS CLI をインストールします。フレームワークの使用の詳細については、AWS Command Line Interfaceユーザーガイド手順については、を参照してください。
2. コマンドプロンプト (Windows) またはターミナル (Linux/macOS) を開きます。

   a. aws configure を入力して、[Enter] を押します。
   b. AWS アクセスキー ID と AWS シークレットアクセスキーを入力します。
   c. デフォルトのリージョン名として「us-west-2」と入力します。
   d. デフォルトの出力形式として「text」と入力します。

3. 次のコマンドを入力します。

```
aws discovery stop-continuous-export --export-id <export ID>
```

- 停止する継続的なエクスポートのエクスポート ID がわからない場合は、次のコマンドを入力して継続的なエクスポートの ID を確認します。

  ```
  aws discovery describe-continuous-exports
  ```

4. 次のコマンドを入力して、返されるステータスが「INACTIVE」であることを確認し、継続エクスポートが停止したことを確認します。

```
aws discovery describe-continuous-export
```

サービスにリンクされたロールを手動で削除する

IAM コンソール、IAM CLI、または IAM API を使用して、AWSServiceRoleForApplicationDiscoveryServiceContinuousExport サービスにリンクされたロールを削除できます。このサービスにリンクされたロールが必要な Discovery Service-Continuous Export 機能が不要になった場合には、そのロールを削除することをお勧めします。そうすることで、積極的にモニタリングまたは保守されていない未使用のエンティティを排除できます。詳細については、IAM ユーザーガイドの「サービスにリンクされたロールの削除」を参照してください。

Note

削除する前に、まずサービスにリンクされたロールをクリーンアップする必要があります。「サービスにリンクされたロールのクリーンアップ (p. 90)」を参照してください。

AWS Application Discovery Service の Identity and Access のトラブルシューティング

次の情報は、Application Discovery Service と IAM の使用時に発生する可能性のある、一般的な問題の診断や修復に役立ちます。

トピック

- iam:PassRole を実行する権限がない (p. 91)

iam:PassRole を実行する権限がない

`iam:PassRole` アクションを実行する権限がないというエラーが表示された場合、管理者に問い合わせ、サポートを依頼する必要があります。お客様のユーザー名とパスワードを発行したのが、担当の管理者です。管理者に、ロールを Application Discovery Service に渡すことができるようにポリシーを更新することを要請します。

一部の AWS 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せずに、既存のロールをサービスに渡すことができます。そのためには、サービスにロールを渡すアクセス許可が必要です。

以下の例のエラーは、という IAM ユーザーが発生します。`marymajor`コンソールを使用して、Application Discovery Service でアクションを実行しようします。ただし、アクションでは、サービスロールによって

付与された許可がサービスにある必要があります。メアリーには、ロールをサービスに渡すアクセス許可がありません。

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

この場合、メアリーは担当の管理者に `iam:PassRole` アクションを実行できるようにポリシーの更新を依頼します。

AWS Application Discovery Service でのロギングとモニタリング

AWS Application Discovery Service はと統合されていますAWS CloudTrail。CloudTrail を使用して、トラブルシューティングや監査の目的で、アカウントアクティビティの記録、継続的な監視、保持を行うことができます。CloudTrail は、AWS マネジメントコンソール、AWS SDK、コマンドラインツールを通じて実行されたアクションを含め、AWS アカウントのアクティビティについて、イベント履歴を提供します。このセクションのトピックでは、Application Discovery Service で CloudTrail を使用する方法について説明します。

トピック

Application Discovery Service API コールのログAWS CloudTrail

AWS Application Discovery Serviceはと統合されます。AWS CloudTrailは、ユーザー、ロール、またはAWSApplication Discovery Service Service で Service CloudTrail のすべての API コールをイベントとして Application Discovery Service にキャプチャします。キャプチャされたコールには、Application Discovery Service コンソールからのコールと、Application Discovery Service API オペレーションへのコードコールが含まれます。

証跡を作成する場合は、Application Discovery Service 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 で収集された情報を使用して、Application Discovery Service に対するリクエスト、リクエスト元の IP アドレス、リクエスト者、リクエスト日時などの詳細を確認できます。

CloudTrail の詳細については、『AWS CloudTrailユーザーガイド。

CloudTrail での Application Discovery Service 情報

CloudTrail は、AWSアカウント作成時にアカウントを作成します。Application Discovery Service でアクティビティが発生すると、そのアクティビティは [その他の CloudTrail イベントに記録されます。AWSのサービスイベントイベント履歴。最近のイベントは、AWS アカウントで表示、検索、ダウンロードできます。詳細については、「CloudTrail イベント履歴でのイベントの表示」を参照してください。

のイベントの継続的な記録についてはAWSアカウント (Application Discovery Service のイベントなど) は、証跡を作成します。証跡により、CloudTrail はログファイルを Amazon S3 バケットに配信できます。デフォルトでは、コンソールで作成した証跡がすべての AWS リージョンに適用されます。証跡では、すべてのリージョンのイベントがログに記録されます。AWSパーティションを作成し、指定した Amazon S3 バケットにログファイルを渡します。さらに、その他のAWSサービスを使用して、CloudTrail ログで収集されたデータに基づき、詳細な分析やアクティビティを行うことができます。詳細については、以下を参照してください。

- 証跡を作成するための概要
- CloudTrail のサポート対象サービスと統合
- Amazon SNS の CloudTrail 通知の設定
- 「複数のリージョンから CloudTrail ログファイルを受け取る」および「複数のアカウントから CloudTrail ログファイルを受け取る」

Application Discovery Service アクションはすべて CloudTrail によってログに記録され、これらのアクションはApplication Discovery Service API。例えば、`CreateTags`、`DescribeTags`、`GetDiscoverySummary` の各アクションを呼び出すと、CloudTrail ログファイルにエントリが生成されます。

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。この ID 情報は以下のことを確認するうえで役立ちます。

- リクエストが、root 認証情報と AWS Identity and Access Management (IAM) ユーザー認証情報のどちらを使用して送信されたか.
- リクエストが、ロールとフェデレーティッドユーザーのどちらの一時的なセキュリティ認証情報を使用して送信されたか。
- リクエストが、別の AWS 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。

Application Discovery Service ログファイルエントリについて

証跡は、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できる設定です。CloudTrail ログファイルには、1 つ以上のログエントリがあります。イベントは任意の発生元からの 1 つ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に関する情報が含まれます。CloudTrail ログファイルは、パブリック API 呼び出し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。

以下の例は、`DescribeTags` アクションの CloudTrail ログエントリを示しています。


```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAJBHMC4H6EKEXAMPLE:sample-user",
        "arn": "arn:aws:sts::444455556666:assumed-role/ReadOnly/sample-user",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDAJQABLZS4A3QDU576Q",
                "arn": "arn:aws:iam::444455556666:role/ReadOnly",
                "accountId": "444455556666",
                "userName": "sampleAdmin"
            },
            "webIdFederationData": {},
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2020-05-05T15:19:03Z"
            }
        }
    },
    "eventTime": "2020-05-05T17:02:40Z",
    "eventSource": "discovery.amazonaws.com",
    "eventName": "DescribeTags",
```

```
    "awsRegion": "us-west-2",
    "sourceIPAddress": "20.22.33.44",
    "userAgent": "Coral/Netty4",
    "requestParameters": {
        "maxResults": 0,
        "filters": [
            {
                "values": [
                    "d-server-0315rfdjreyqsq"
                ],
                "name": "configurationId"
            }
        ]
    },
    "responseElements": null,
    "requestID": "mgh-console-eb1cf315-e2b4-4696-93e5-b3a3b9346b4b",
    "eventID": "7b32b778-91c9-4c75-9cb0-6c852791b2eb",
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

AWS Application Discovery Service のクォータ

サービスクォータコンソールには、AWS Application Discovery Service のクォータに関する情報が表示されます。Service Quotas コンソールを使用して、デフォルトのサービスクォータを表示することや、要求クォータの増加調整可能なクォータ用。

現在、増加できるクォータはアカウントあたりのインポートサーバー。

Application Discovery Service には、次の既定のクォータがあります。

- アカウントあたり 1,000 アプリケーション。

  このクォータに達したが、新しいアプリケーションをインポートする場合は、`DeleteApplications`API アクション。詳細については、「」を参照してください。DeleteApplications()Application Discovery Service API リファレンス。
- 各インポートファイルの最大ファイルサイズは 10 MB です。
- アカウントあたり 25,000 のインポートサーバーレコード。
- 1日あたり 25,000 のインポートレコードの削除。
- 1 アカウントあたり 10,000 台のインポートされたサーバー (このクォータを増やすことを要求できます)。
- 1,000 アクティブエージェント。データを収集してApplication Discovery Service に送信するエージェント。
- 10,000 非アクティブエージェント。応答するもののデータ収集は行わないエージェント。
- アプリケーションあたり 400 サーバー。
- サーバあたり 30 個のタグ

Amazon Athena でのデータ探索のトラブルシューティング

このセクションでは、AWS Application Discovery Service の一般的な問題の修正方法について説明します。

トピック

データ探索によるデータ収集の停止

データ探索を停止するには、Migration Hub コンソールの [Discover] > [Data Collectors] > [Agents] タブでトグルスイッチをオフにするか、`StopContinuousExport`API. データ収集を停止するには最大 30 分かかる場合があります。このステージでは、コンソールのトグルスイッチおよび、`DescribeContinuousExport` API 呼び出しで、データ探索の状態は「停止中」と表示されます。

> Note
>
> コンソールページをリフレッシュした後、切り替えのスイッチがオフにならずエラーメッセージがスローされるか、`DescribeContinuousExport` API が、「Stop_Failed」を返す場合は、再度コンソールでトグルスイッチをオフにするか `StopContinuousExport` API を呼び出します。「データ探索」で、引き続きエラーが表示され、正常に停止できない場合は、AWS サポートにお問い合わせください。

または、次の手順で説明されているようにデータ収集を手動で停止できます。

オプション 1: エージェントデータ収集の停止

ADS エージェントを使用した検出がすでに完了していて、ADS データベースリポジトリで追加データをさらに収集しない場合:

1. Migration Hub コンソールから、[Discover] > [Data Collectors] >
2. 実行中の既存のすべてのエージェントを選択して、[Stop Data Collection (データ収集の停止)] を選択します。

   これにより、ADS データリポジトリおよび S3 バケットの両方で、エージェントにより、新しいデータが収集されていないことを確認できます。既存のデータには引き続きアクセスできます。

オプション 2: データ探索の Amazon Kinesis Data Streams を削除する

ADS データリポジトリのデータはエージェントを使用して引き続き収集するが、データ探索を使用して Amazon S3 バケットのデータは収集しない場合は、データ探索によって作成された Amazon Kinesis Data Firehose ストリームを手動で削除できます。

1. Amazon Kinesis にログインするには、AWSコンソールを選択し、データFirehoseナビゲーションペインで [] を選択します。
2. データ探索機能によって作成された次のストリームを削除します。

   - `aws-application-discovery-service-id_mapping_agent`
   - `aws-application-discovery-service-inbound_connection_agent`
   - `aws-application-discovery-service-network_interface_agent`
   - `aws-application-discovery-service-os_info_agent`
   - `aws-application-discovery-service-outbound_connection_agent`
   - `aws-application-discovery-service-processes_agent`
   - `aws-application-discovery-service-sys_performance_agent`

データ探索によって収集されたデータの削除

データ探索によって収集されたデータを削除するには

1. Amazon S3 に保存されている検出エージェントデータを削除します。

   Application Discovery Service (ADS) で収集されたデータは S3 バケットに、`aws-application-discover-discovery-service-`*`uniqueid`* という名前で保存されます。

   > **Note**
   >
   > Amazon Athena でのデータ探索が有効になっているときに、Amazon S3 バケットまたはバケット内のオブジェクトのいずれかを削除するとエラーが発生することがあります。新しい検出エージェントデータが S3 に引き続き送信されます。Athena でも、削除されたデータには以後アクセスできなくなります。

2. 削除 AWS Glue Data Catalog.

   Amazon Athena でのデータ探索が有効になっている場合、定期的な間隔で ADS エージェントを使用して収集されたデータを保存する Amazon S3 バケットがアカウント内に作成されます。さらに、それはまた、AWS Glue Data Catalogを使用して、Amazon Athena から Amazon S3 バケットに保存されているデータをクエリできます。Amazon Athena のデータ探索を無効にすると、新しいデータは Amazon S3 バケットに保存されませんが、以前に収集されたデータは保持されます。このデータが不要になり、アカウントを Amazon Athena のデータ探索が有効になっている前の状態に戻すには

   a. Amazon S3 にアクセスしてください。AWSコンソールを使用して、「aws-application-discover-discovery-service-uniqueid」という名前のバケットを手動で削除します。
   b. application-discovery-service-database データベースとそのすべてのテーブルを削除して、AWS Glue データカタログのデータ探索を手動で削除できます。

      - `os_info_agent`
      - `network_interface_agent`
      - `sys_performance_agent`
      - `processes_agent`
      - `inbound_connection_agent`
      - `outbound_connection_agent`
      - `id_mapping_agent`

AWS Application Discovery Service からのデータの削除

Application Discovery Service からすべてのデータを削除するには、AWS サポートし、完全なデータ削除を要求します。

Amazon Athena でのデータ探索に関する一般的な問題を修正

このセクションでは、Amazon Athena でのデータ探索の一般的な問題の修正方法について説明します。

トピック

サービスにリンクされたロールおよび必要な AWS リソースが作成できなかったため、Amazon Athena でのデータ探索を開始できません

Amazon Athena でデータ探索を有効にすると、サービスリンクロールAWSServiceRoleForApplicationDiscoveryServiceContinuousExportを作成することを可能にするあなたのアカウントで、必要なAWSAmazon S3 バケット、Amazon Kinesis ストリーム、および Amazon Athena でエージェントが収集したデータにアクセスできるようにするためのリソースAWS Glue Data Catalog。アカウントに、このロールを作成する Amazon Athena のデータ探索の適切なアクセス許可がない場合は、初期化に失敗します。Application Discovery Service での AWS 管理 (事前定義) ポリシー (p. 79) を参照してください。

Amazon Athena に新しいエージェントデータが表示されません

新しいデータが Athena に取り込まれない場合、エージェントの開始から 30 分以上が経過している場合、およびデータ探索のステータスがアクティブな場合は、以下に示すソリューションを確認してください。

- AWS 検出エージェント

  エージェントの [Collection (収集)] ステータスが [Started (開始)] とマークされ、[Health (状態)] ステータスが [Running (実行中)] とマークされていることを確認してください。

- Kinesis ロール

  アカウントに `AWSApplicationDiscoveryServiceFirehose` ロールがあることを確認します。

- Kinesis Data Firehose ステータス

  以下の Kinesis Data Firehose 配信ストリームが正常に動作していることを確認します。

  - `aws-application-discovery-service/os_info_agent`
  - `aws-application-discovery-service-network_interface_agent`
  - `aws-application-discovery-service-sys_performance_agent`
  - `aws-application-discovery-service-processes_agent`
  - `aws-application-discovery-service-inbound_connection_agent`

- `aws-application-discovery-service-outbound_connection_agent`
- `aws-application-discovery-service-id_mapping_agent`

- AWS Glue Data Catalog

  AWS Glue に `application-discovery-service-database` データベースがあることを確認します。AWS Glue に以下のテーブルが存在することを確認します。

  - `os_info_agent`
  - `network_interface_agent`
  - `sys_performance_agent`
  - `processes_agent`
  - `inbound_connection_agent`
  - `outbound_connection_agent`
  - `id_mapping_agent`

- Amazon S3 バケット

  という名前の Amazon S3 バケットがあることを確認します。`aws-application-discovery-service-`*`uniqueid`*アカウントにを入力します。バケット内のオブジェクトが移動または削除された場合は、Athena で適切に表示されません。

- オンプレミスサーバー

  サーバーが実行されていて、エージェントが AWS Application Discovery Service にデータを収集して送信できることを確認します。

Amazon S3、Amazon Kinesis Data Firehose、またはにアクセスするためのアクセス許可が不足しています。AWS Glue

を使用している場合AWS Organizations、Amazon Athena でのデータ探索の初期化に失敗する場合は、Amazon S3、Amazon Kinesis Data Firehose、Athena、またはAWS Glue。

これらのサービスにアクセスできるようにするためには、管理者権限を持つ IAM ユーザーが必要です。管理者は、このアクセス権限を付与するために、ユーザーのアカウントを使用できます。「Application Discovery Service での AWS 管理 (事前定義) ポリシー (p. 79)」を参照してください。

Amazon Athena でのデータ探索が正常に動作することを確認するために、AWSAmazon Athena S3バケット、Amazon Kinesis Data Firehose ストリーム、およびAWS Glue Data Catalog。これらのリソースを誤って削除または変更してしまった場合は、データ探索を停止して起動すると、これらのリソースが自動的に再作成されます。データ探索によって作成された Amazon S3 バケットを削除した場合、バケット内に収集されたデータが失われる場合があります。

失敗したインポートレコードの問題のトラブルシューティング

Migration Hub のインポートを使用すると、Discovery Connector や Discovery Agent を使用しなくても、オンプレミス環境の詳細を Migration Hub そのため、インポートデータを使用して、直接、移行の評価お

よび計画を行うこともできます。デバイスをアプリケーションとしてグループ化し、それらの移行ステータスを追跡することもできます。

データをインポートする際、エラーが発生する可能性があります。通常、これらのエラーは、次のいずれかの原因により発生します。

- インポート関連のクォータに達しました— インポートタスクに関連付けられたクォータがあります。クォータを超えるインポートタスクリクエストを行った場合、そのリクエストは失敗し、エラーが返ります。詳細については、「

  サービスクォータコンソールには、AWS Application Discovery Service のクォータに関する情報が表示されます。Service Quotas コンソールを使用して、デフォルトのサービスクォータを表示することや、要求クォータの増加調整可能なクォータ用。

  現在、増加できるクォータはアカウントあたりのインポートサーバー。

  Application Discovery Service には、次の既定のクォータがあります。

  - アカウントあたり 1,000 アプリケーション。

    このクォータに達したが、新しいアプリケーションをインポートする場合は、`DeleteApplications`API アクション。詳細については、「」を参照してください。DeleteApplications()Application Discovery Service API リファレンス。
  - 各インポートファイルの最大ファイルサイズは 10 MB です。
  - アカウントあたり 25,000 のインポートサーバーレコード。
  - 1日あたり 25,000 のインポートレコードの削除。
  - 1 アカウントあたり 10,000 台のインポートされたサーバー (このクォータを増やすことを要求できます)。
  - 1,000 アクティブエージェント。データを収集してApplication Discovery Service に送信するエージェント。
  - 10,000 非アクティブエージェント。応答するもののデータ収集は行わないエージェント。
  - アプリケーションあたり 400 サーバー。
  - サーバあたり 30 個のタグ

  (p. 95)」を参照してください。
- 余分なカンマ (,) がインポートファイルに挿入されました— CSV ファイル内のカンマは、フィールドと次のフィールドを区別するために使用されます。フィールド内にカンマを入れることはサポートされていません。カンマを入れるとフィールドが分割されます。これが原因で、フォーマットエラーのカスケードが生じることがあります。カンマはフィールド間でのみ使用され、インポートファイルで使用することはできません。
- フィールドに、サポート範囲外の値が含まれています— いくつかのフィールド (`CPU.NumberOfCores`には、サポート範囲内の値が必要です。サポートされている範囲よりも多い、または少ない場合、レコードはインポートされません。

インポートリクエストでエラーが発生した場合は、インポートタスクの失敗したレコードをダウンロードしてそれらを解決し、失敗したエントリの CSV ファイルでエラーを解決してから再度インポートします。

Console

失敗したレコードのアーカイブをダウンロードするには

1. サインインします。AWS Management Consoleを開き、Migration Hub コンソールhttps://console.aws.amazon.com/migrationhub。
2. 左側のナビゲーションペインの [Discover (検出)] で [Tools (ツール)] を選択します。

3. [検出ツール] から、[view imports (インポートの表示)] を選択します。
4. [インポート] ダッシュボードから、[失敗したレコード] をいくつか含むインポートリクエストに関連付けられたラジオボタンを選択します。
5. ダッシュボードのテーブルの上から、[失敗したレコードのダウンロード] を選択します。これにより、アーカイブファイルをダウンロードするためのブラウザのダウンロードダイアログボックスが開きます。

AWS CLI

失敗したレコードのアーカイブをダウンロードするには

1. ターミナルウィンドウを開いて、次のコマンドを入力します。ここで、*ImportName* is the name of the import task with the failed entries that you want to correct.

```
aws discovery describe-import-tasks - -name ImportName
```

2. 出力から、errorsAndFailedEntriesZip で返る値の内容全体をコピーします (引用符で囲まない)。
3. ウェブブラウザを開き、その内容を URL のテキストボックスに貼り付け、ENTER を押します。これにより、失敗したレコードのアーカイブ (.zip 形式で圧縮) がダウンロードされます。

失敗したレコードのアーカイブがダウンロードされました。次に、中の 2 つのファイルを抽出してエラーを修正します。エラーがサービスベースの制限に関連付けられている場合は、制限の引き上げをリクエストするか、アカウントを制限以下にするのに十分な関連リソースを削除する必要があります。アーカイブには次のファイルがあります。

- errors-file.csv— このファイルはエラーログを示します。これを使用して、行、列名、列名、ExternalId、および失敗した各エントリの失敗した各レコードについて、エラーメッセージの説明を示します。
- failed-entries-file.csv— このファイルには、元のインポートファイルの失敗したエントリのみが含まれています。

発生した非制限ベースのエラーを修正するには、errors-file.csv を使用して、failed-entries-file.csv ファイルの問題を修正してから、そのファイルをインポートします。ファイルのインポート手順については、「データのインポート (p. 37)」を参照してください。

AWS Application Discovery Service のドキュメント履歴

- API バージョン: 2015-11-01
- ユーザーガイド最終更新日: 2019 年 11 月 14 日

次の表は、重要な変更点をまとめたものです。AWS Migration Hubユーザーガイド2019年1月18日以降。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。

update-history-change	update-history-description	update-history-date
ホームリージョンの紹介 (p. 102)	Migration Hub ホームリージョンは、ポートフォリオ全体の検出および移行計画情報の単一リポジトリと、複数の AWS リージョンへの移行の単一ビューを提供します。	2019 年 11 月 20 日
Migration Hub のインポート機能の紹介 (p. 102)	Migration Hub では、オンプレミスのサーバーおよびアプリケーションに関する情報 (例:サーバーの仕様および使用状況のデータ) を Migration Hub にインポートできます。このデータを使用して、アプリケーション移行のステータスを追跡することもできます。詳細については、「」を参照してください。Migration Hub インポート。	2019 年 1 月 18 日

次の表は、のドキュメントのリリースの一覧です。AWS Migration Hubユーザーガイド2019 年 1 月 18 日までの:

変更	説明	日付
新機能	Amazon Athena でのデータ探索のサポートドキュメントを更新し、トラブルシューティングに関する章を追加しました。	2018 年 8 月 09 日
主な改訂	使用と出力に関する詳細を書き直し、ドキュメント全体を再構成しました。	2018 年 5 月 25 日
検出エージェント 2.0	新しく改善したアプリケーション検出エージェントをリリースしました。	2017 年 10 月 19 日
コンソール	AWS Management Console が追加されました。	2016 年 19 月 12 日

変更	説明	日付
エージェントレス検出	このリリースでは、エージェントレス検出のセットアップおよび設定方法について説明しています。	2016 年 7 月 28 日
Microsoft Windows Server の新しい詳細とコマンド問題の修正	この更新では、Microsoft Windows Server の詳細を追加しています。また、さまざまなコマンド問題の修正について説明しています。	2016 年 5 月 20 日
初版発行	これはApplication Discovery Service Guide.	2016 年 12 月 5 日

AWS 用語集

最新の AWS の用語については、AWS 全般のリファレンスの「AWS 用語集」を参照してください。

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。