アカウントの接続に関するトラブルシューティング

- アカウントの手動オンボーディング （1 ページ）
- 接続の段階的終了 （11 ページ）
- クラウドアカウント用の Terraform オンボーディングスクリプト （12 ページ）

アカウントの手動オンボーディング

アカウント オンボーディングに記載されている方法でクラウドサービスプロバイダーアカウントを Multicloud Defense にオンボーディングする場合は、アカウントを手動でオンボーディングする必要があります。別の方法として、次のオプションを使用できます。

GCP プロジェクトの手動オンボーディング

GCP の概要

GCP プロジェクトと GCP フォルダ

Multicloud Defense は現在、GCP プロジェクトと GCP フォルダの両方をサポートしていますが、これらのコンポーネントは個別にサポートされています。これらの両方のオプションについて、次の制限事項と例外に注意してください。

GCP プロジェクトには、仮想マシン、ストレージバケット、データベースなどの GCP リソースを含めることができます。すべての Google Cloud サービスを作成、有効化、および使用するために使用できます。

- プロジェクトは、Terraform、手動オンボーディング、およびスクリプト化されたオンボーディングを使用してオンボーディングできます。
- プロジェクトは、検出や調査などのオーケストレーションを必要とする環境に最適です。
- 各プロジェクトは Multicloud Defense ダッシュボードから個別に操作できます。

バージョン 23.10 では、Terraform を使用して GCP フォルダを接続できます。GCPフォルダには、プロジェクト、他のフォルダ、またはその両方の組み合わせが含まれます。組織リソースは、フォルダを使用して、階層内の組織リソースノードの下にプロジェクトをグループ化できます。

- `roles/compute.admin` 権限が有効になっていないフォルダは空と見なされ、使用されません。
- オンボーディングされたフォルダに関連付けられているプロジェクトは、アセットとトラフィックの検出にのみ使用されます。
- オンボーディングされたフォルダに関連付けられているプロジェクトは、サービス VPC のオーケストレーションまたはゲートウェイの作成には対応していません。
- GCP コンソールからフォルダに付与する権限は、フォルダレベルで付与する必要があります。そのため、Multicloud Defense のアクションはフォルダレベルでも行われます。

GCP フォルダをオンボードする場合は、「Terraform リポジトリ」を参照してください。

手順の概要

次に、GCP プロジェクトを接続する方法の概要を示します。シェル**スクリプト**が Multicloud Defense によって提供されるため、接続プロセスをウィザードの一部として簡単に進めることができます。次の手順はスクリプトによって自動化されるため、実行する必要はありません。

1. サービスアカウントを 2 つ作成します。
2. 次の API（Compute Engine、Secret Manager）を有効にします。
3. 次の 2 つの VPC（管理、データパス）を作成します。
4. データパス VPC で Multicloud Defense Gateway へのトラフィック（アプリケーション トラフィック）を許可するファイアウォールルールを作成します。
5. 管理 VPC で Multicloud Defense Controller から Multicloud Defense Gateway への管理トラフィックを許可するファイアウォールルールを作成します。

スクリプトが機能しない場合、または設定を手動で変更する必要がある場合は、GCP クラウドコンソールの Web UI または gcloud CLI を使用してこれらのアクションを実行できます。プロジェクトを接続する別のメソッドについては、GCP プロジェクトの手動オンボーディングを参照してください。

サービスアカウント

Multicloud Defense には、GCP プロジェクトで作成された 2 つのサービスアカウントが必要です。

- **Multicloud Defense-controller**：このアカウントは、Multicloud Defense Controller が GCP プロジェクトにアクセスしてリソース（Multicloud Defense Gateway）、Multicloud Defense Gateway のロードバランサを作成し、VPC、サブネット、セキュリティグループタグなどに関する情報を読み取るために使用されます。

- **Multicloud Defense-gateway**：このアカウントは Multicloud Defense Gateway（コンピューティング VM インスタンス）に割り当てられます。アカウントは、シークレットマネージャ（TLS 復号用の秘密キー）とストレージへのアクセスを提供します。

これらのサービスアカウントは、UI で使用可能なサービスを使用する方法と、クラウド サービス プロバイダーの CLI を使用する方法のいずれかの方法で作成できます。

GCP Cloud コンソールを使用した Multicloud Defense Controller サービスアカウントの作成

Multicloud Defense Controller サービスアカウントは、GCP プロジェクトのリソースにアクセスして管理するために Multicloud Defense Controller によって使用されます。アカウントを作成し、キーを生成する必要があります。キーは、コントローラへのアカウントオンボーディングの一部としてコントローラに追加されます。

手順

ステップ 1 GCP プロジェクトで [IAM] を開きます。

ステップ 2 [サービスアカウント（Service Account）] をクリックします。

ステップ 3 サービスアカウントを作成します。

ステップ 4 名前と ID（Multicloud Defense-controller など）を入力し、[作成（Create）] をクリックします。

ステップ 5 [コンピューティング管理者（Computing Admin）] ロールと [サービスアカウントユーザー（Service Account User）] ロールを追加します。

ステップ 6 [続行（Continue）] をクリックします。

ステップ 7 [完了（Done）] をクリックします。

（注）
ユーザーを追加する必要はありません。

ステップ 8 新しく作成したアカウントをクリックし、[キー（Keys）] までスクロールして、[キーの追加（Add Key）] ドロップダウンで [新しいキーの作成（Create New Key）] を選択します。

ステップ 9 [JSON]（デフォルトオプション）を選択し、[作成（Create）] をクリックします。

ステップ 10 ファイルがコンピュータにダウンロードされます。このファイルを保存します。

GCP Cloud Console を使用した Multicloud Defense ファイアウォール サービス アカウントの作成

Multicloud Defense ファイアウォール サービス アカウントは、GCP プロジェクト内で実行されている Multicloud Defense Gateway インスタンスによって使用されます。ゲートウェイは、TLS 復号のために Secret Manager に保存されている秘密キーにアクセスし、PCAP ファイルなどを保存するためのストレージにアクセスする必要があります（ユーザーが設定した場合）。また、多くのゲートウェイには、Multicloud Defense Gateway から GCP ロギングインスタンスにログを送信するためのログ作成者権限が必要です（ユーザーが設定した場合）。

このサービスアカウントを作成する 2 つの方法を次に示します。

手順

ステップ 1 GCP プロジェクトで [IAM] を開きます。
ステップ 2 [サービスアカウント（Service Account）] をクリックします。
ステップ 3 サービスアカウントを作成します。
ステップ 4 名前と ID（Multicloud Defense-firewall など）を入力し、[作成（Create）] をクリックします。
ステップ 5 [シークレットマネージャ（Secret Manager）]、[シークレットアクセサ（Secret Accessor）]、および[ログ作成者（Logs Writer）] ロールを追加します。
ステップ 6 [続行（Continue）] をクリックします。
ステップ 7 [完了（Done）] をクリックします。

（注）
ユーザーを追加する必要はありません。

API の有効化

GCP コンソールまたはクラウド サービス プロバイダーの CLI を使用して、Multicloud Defense Controller と GCP アカウント間で通信するための API を有効にできます。

API の有効化：GCP Cloud Console を使用

Multicloud Defense Controller が Multicloud Defense Gateway（仮想マシン、ロードバランサ）を作成できるように、プロジェクト/アカウントで API を有効にします。

手順

ステップ 1 検索バーで **Compute Engine API** を検索します。
ステップ 2 [有効（Enable）] をクリックします。
ステップ 3 検索バーで **Secret Manager API** を検索します。
ステップ 4 [有効（Enable）] をクリックします。
ステップ 5 検索バーで **Identity and Access Management (IAM) API** を検索します。
ステップ 6 [有効（Enable）] をクリックします。
ステップ 7 検索バーで **Cloud Resource Manager API** を検索します。
ステップ 8 [有効（Enable）] をクリックします。

VPC セットアップ

Multicloud Defense Gateway インスタンスはエッジモードまたはハブで展開できます。エッジモードでは、ゲートウェイインスタンスはアプリケーションと同じ VPC で実行されます。このドキュメントでは、Multicloud Defense Gateway 展開をエッジモードで展開するための準備に焦点を当てます。

VPC とサブネット

Multicloud Defense Gateway を展開すると、Multicloud Defense Controller は [管理（management）] および [データパス（datapath）] VPC 情報を要求します。Multicloud Defense Gateway インスタンスには 2 つのネットワーク インターフェイスが必要です。GCP では、VM インスタンスのネットワーク インターフェイスは、単に異なるサブネットに存在できる他のクラウドプロバイダーとは異なり、異なる VPC に存在する必要があります。アプリケーションが実行されている VPC がすでにある場合は、 [データパス（datapath）] VPC とサブネットが設定されています。管理用に別の VPC を作成する（または別の既存の VPC を使用する）必要があります。自動作成されたサブネットを使用することも、手動で作成することもできます。

データパス VPC はアプリケーションが実行されている VPC であり、以降のセクションではそのようなものとして言及されます。

各 VPC で、Multicloud Defense にはデータパス用の 1 つのサブネットと管理用の 1 つのサブネットが必要です。

[管理（management）] サブネットは、インターネットへのデフォルトルートを持つルートテーブルに関連付ける必要があるパブリックサブネットです。Multicloud Defense Gateway インスタンスには、Multicloud Defense Controller との通信に使用する、このサブネットにアタッチされたインターフェイスがあります。このインターフェイスは、Multicloud Defense Controller と Multicloud Defense Gateway インスタンス間のポリシーのプッシュやその他の管理およびテレメトリアクティビティに使用されます。お客様のアプリケーションのトラフィックは、このインターフェイスとサブネットを**通過しません**。インターフェイスは、**Multicloud Defense-management** ネットワークタグ（またはチームの要件に基づく任意のタグ）に関連付けられます。これについては、以下のネットワークタグのセクションで説明します。

[データパス（datapath）] サブネットは、インターネットへのデフォルトルートを持つルートテーブルに関連付ける必要があるパブリックサブネットです。Multicloud Defense Controller は、このサブネットにネットワーク ロード バランサ（NLB）を作成します。さらに、Multicloud Defense Gateway インスタンスには、このサブネットにアタッチされたインターフェイスがあります。お客様のアプリケーションのトラフィックは、このインターフェイスを**通過します**。セキュリティポリシーは、このインターフェイスを介して入力されるトラフィックに適用されます。インターフェイスは、**Multicloud Defense-datapath** ネットワークタグ（またはチームの要件に基づく任意のタグ）に関連付けられます。これについては、以下のネットワークタグのセクションで説明します。

CLI を使用した *VPC* とサブネットの例

独自のコマンドを実行して GCP アカウントの VPC を作成する場合、以下のコマンドを一例として使用します。これらの特定のコマンド用に Google Cloud Shell ウィンドウを開きます。

手順

ステップ 1 VPC **apps** とサブネット **apps-us-east1** を作成します。

ステップ 2 VPC `Multicloud Defense-mgmt` とサブネット `Multicloud Defense-mgmt-us-east1` を作成します。

ステップ 3 target-tags を `Multicloud Defense-mgmt` とした VPC `Multicloud Defense-mgmt`に少なくとも 2 つのファイアウォールルールを作成します。

1. すべてのアウトバウンドトラフィックを許可する出力ルール：
2. ファイアウォール インスタンスへの SSH を許可する入力ルール：

ステップ 4 VPC **apps** 用に少なくとも 3 つのファイアウォールルールを作成します。以下は例として使用してください。

1. target-tags を `Multicloud Defense-datapath` とする、すべてのアウトバウンドトラフィックを許可する 1 つの出力ルール：
2. target-tags を `Multicloud Defense-datapath` とする、非ロードバランサを介してゲートウェイインスタンスへの HTTP および HTTPS を許可する 1 つの入力ルール：
3. target-tags を `app-instance` とする、すべてのアウトバウンドトラフィックを許可する 1 つの出力ルール：
4. `target-tags` を `app-instance` とする、`tcp:8000` を許可する 1 つの入力ルール：

```
gcloud config set project <project-name> # incase the project is not set in the gcloud cli shell
gcloud compute networks create apps --subnet-mode custom
gcloud compute networks subnets create apps-us-east1 --network apps --range 10.0.0.0/24 --region
us-east1
gcloud compute networks create ciscomcd-mgmt --subnet-mode custom
gcloud compute networks subnets create ciscomcd-mgmt-us-east1 --network ciscomcd-mgmt --range
172.16.0.0/24 --region us-east1
gcloud compute firewall-rules create ciscomcd-mgmt-out --direction EGRESS --network ciscomcd-mgmt
\
    --target-tags ciscomcd-mgmt --allow tcp,udp
gcloud compute firewall-rules create ciscomcd-mgmt-in --direction INGRESS --network ciscomcd-mgmt
\
    --target-tags ciscomcd-mgmt --allow tcp:22
gcloud compute firewall-rules create ciscomcd-datapath-out --direction EGRESS --network apps \
    --target-tags ciscomcd-datapath --allow tcp,udp
gcloud compute firewall-rules create ciscomcd-datapath-in --direction INGRESS --network apps \
    --target-tags ciscomcd-datapath --allow tcp:80,tcp:443
gcloud compute firewall-rules create app-instance-out --direction EGRESS --network apps \
    --target-tags app-instance --allow tcp,udp
gcloud compute firewall-rules create app-instance-in --direction INGRESS --network apps \
    --target-tags app-instance --allow tcp:8000,tcp:22
```

上記のコマンドを実行すると、**apps** VPC で VM インスタンスを作成し、ポート 8000 でテスト用 Web アプリケーションを起動できます。

```
gcloud compute instances create app-instance1 \
```

```
  --zone=us-east1-b \
  --image-project=ubuntu-os-cloud \
  --image-family=ubuntu-2004-lts \
  --network apps \
  --subnet=apps-us-east1 \
  --tags=app-instance
gcloud compute ssh app-instance1 --zone us-east1-b
echo hello world > index.html
python3 -m http.server 8000
```

ネットワークタグ（GCP ゲートウェイ用）

管理およびデータパスのネットワークタグは、上記のサブネットセクションで説明されているように、Multicloud Defense Gateway インスタンスのそれぞれのインターフェイスに関連付けられます。

[管理（management）] VPC でゲートウェイルールを作成し、**Multicloud Defense-management** ネットワークタグに関連付けます。これにより、ゲートウェイインスタンスがコントローラと通信するように、すべてのアウトバウンドトラフィックが許可される必要があります。必要に応じて、インバウンドルールでは、ポート 22（SSH）を有効にしてゲートウェイインスタンスへの SSH アクセスを許可します。Multicloud Defense ファイアウォールが正常に機能するのに SSH は**必須ではありません**。

[データパス（datapath）] VPC でゲートウェイルールを作成し、**Multicloud Defense-datapath** ネットワークタグに関連付けます。これにより、有効にするすべてのサービスの Multicloud Defense Gateway へのトラフィックを許可する必要があります。

たとえば、アプリケーションがポート 3000 で実行されていて、Multicloud Defense Gateway によってポート 443 でプロキシ接続されている場合、ポート 443 は Multicloud Defense-datapath ネットワーク セキュリティ タグで開く必要があります。

ゲートウェイの作成

Multicloud Defense Gateway 作成ページでは、次のパラメータを使用します。

1. データパス VPC：**apps**。
2. データパス ネットワーク タグ：**Multicloud Defense-datapath**。
3. 管理 VPC：**Multicloud Defense-mgmt**。
4. 管理ネットワークタグ：**Multicloud Defense-mgmt**。
5. **us-east1-b** ゾーンを活用します。
6. 管理サブネット：**Multicloud Defense-mgmt-us-east1**。
7. データパスサブネット：**apps-us-east1**。

他のリージョンにサブネットを作成して、マルチ可用性ゾーンモードで Multicloud Defense Gateway をテストできます。

Azure サブスクリプションの手動オンボーディング

Multicloud Defense Controller ダッシュボードで提供されるスクリプトで Azure サブスクリプションに直接接続できない場合は、次の手順を実行してサブスクリプションを手動で接続します。

（任意）Key Vault および BLOB ストレージへのアクセス用のユーザー割り当てマネージド ID

Multicloud Defense Gateway は、オプションで Azure Key Vault と統合して TLS 証明書を取得したり、BLOB ストレージと統合して PCAP（パケットキャプチャ）ファイルを保存したりすることができます。ユーザー割り当てマネージド ID は、これらのサービスへのアクセス権を付与するために使用されます。

Azure ポータルで、[マネージドID（Managed Identities）] に移動して ID を作成します。

または、Azure Cloud Shell で次のコマンドを実行します。

```
az identity create -g <RESOURCE GROUP> -n <USER ASSIGNED IDENTITY NAME>
```

Azure Key Vault で TLS 証明書のシークレットを作成する方法については、Azure Key Vaultを参照してください。

Microsoft Entra ID へのアプリケーションの登録

Multicloud Defense アプリケーションを Entra ID に登録するには、次の手順を実行します。

手順

ステップ 1 Azure ポータルから [Microsoft Entra ID] に移動します。

ステップ 2 [アプリケーションの登録（App Registration）] を選択します。

ステップ 3 [新規登録（New Registration）] をクリックします。

ステップ 4 新規登録するアプリケーションを示す名前を入力します（例：Multicloud Defense Controller）。 [サポートされているアカウントタイプ（Supported Account Types）] で、2 番目のオプションである [任意の組織ディレクトリのアカウント（Accounts in any organizational directory）] を選択します。

ステップ 5 組織に適したオプションを選択します。[リダイレクトURI（Redirect URI）] は、アプリケーション登録の作成時には必要ないことに注意してください。

ステップ 6 [登録（Register）] をクリックします。

ステップ 7 新しく作成したアプリケーションの下にある左側のナビゲーションバーで、[証明書およびシークレット（Certificates & secrets）] をクリックします。

ステップ 8 [+新しいクライアントシークレット（+ New Client Secret）] をクリックし、[クライアントシークレットの追加（Add a client secret）] ダイアログに必要な情報を入力します。

- [説明（Description）]：説明を追加します（例：Multicloud Defense-controller-secret1）

- [有効期限（Expires）] : [なし（Never）] を選択します。この選択はいつでも行うことができます。現在のシークレットが期限切れになったら、新しいシークレットを作成する必要があります）

ステップ 9 [追加（Add）] をクリックします。クライアントシークレットが [値（Value）] 列の下に入力されます。

ステップ 10 クライアントシークレットをメモ帳にコピーします。これは 1 回だけ表示され、再度表示されることはありません。

ステップ 11 左側のナビゲーションバーで、[概要（Overview）] をクリックします。

ステップ 12 アプリケーション（クライアント）ID とディレクトリ（テナント）ID をメモ帳にコピーします。

アプリケーションに割り当てるカスタムロールの作成

CloudFormation テンプレートによって次のロールが作成されます。

- [カスタムロール（CustomRole）] : カスタムロールは、インベントリ情報を読み取り、リソース（VM、ロードバランサなど）を作成する権限をアプリケーションに付与します。カスタムロールは複数の方法で作成できます。

Multicloud Defense Controller 用に作成されたアプリケーションに割り当てられる**カスタムロール**を作成します。カスタムロールは、インベントリ情報を読み取り、リソース（VM、ロードバランサなど）を作成する権限をアプリケーションに付与します。カスタムロールは複数の方法で作成できます。

手順

ステップ 1 [サブスクリプション（Subscriptions）] に移動し、[アクセス制御（IAM）（Access Control (IAM)）] をクリックします。

ステップ 2 [ロール（Roles）] をクリックし、上部のメニューバーで**[+追加（+Add）]**>**[カスタムロールの追加（Add Custom Role）]**をクリックします。

ステップ 3 カスタムロールに名前を付けます（例：`Multicloud Defense-controller-role`）。

ステップ 4 JSON 編集画面が表示されるまで、[次へ（Next）] をクリックし続けます。

ステップ 5 画面で [編集（Edit）] をクリックし、JSON テキストの [権限>アクション（permissions > actions）] セクションで、角カッコの間に次のコンテンツをコピーして貼り付けます（インデントを維持する必要はありません）。

```
"Microsoft.ApiManagement/service/*",
"Microsoft.Compute/disks/*",
"Microsoft.Compute/images/read",
"Microsoft.Compute/sshPublicKeys/read",
"Microsoft.Compute/virtualMachines/*",
"Microsoft.ManagedIdentity/userAssignedIdentities/read",
"Microsoft.ManagedIdentity/userAssignedIdentities/assign/action",
"Microsoft.Network/loadBalancers/*",
"Microsoft.Network/natGateways/*",
"Microsoft.Network/networkinterfaces/*",
"Microsoft.Network/networkSecurityGroups/*",
```

```
"Microsoft.Network/publicIPAddresses/*",
"Microsoft.Network/routeTables/*",
"Microsoft.Network/virtualNetworks/*",
"Microsoft.Network/virtualNetworks/subnets/*",
"Microsoft.Resources/subscriptions/resourcegroups/*",
"Microsoft.Storage/storageAccounts/blobServices/*",
"Microsoft.Storage/storageAccounts/listkeys/action",
"Microsoft.Network/networkWatchers/*",
"Microsoft.Network/applicationSecurityGroups/*",
"Microsoft.Compute/diskEncryptionSets/read",
"Microsoft.Insights/Metrics/Read"
```

ステップ 6 **任意**：Multicloud Defense で複数のサブスクリプションを使用する場合は、`assignableScopes` で JSON を編集して別のサブスクリプション品目を追加か、またはすべてのサブスクリプションでカスタムロールを使用できるように *****（星印）に変更する必要があります。

ステップ 7 テキストボックスの上部にある [保存（Save）] をクリックします。

ステップ 8 [確認して作成（Review + Create）] をクリックして、ロールを作成します。

ステップ 9 カスタムロールが作成されたら、[アクセス制御（IAM）（Access Control (IAM)）] に戻ります。

ステップ 10 上部のメニューバーで、**[追加（Add）]** > **[ロール割り当ての追加（Add role assignment）]**をクリックします。

ステップ 11 [ロール（Role）] ドロップダウンで、上で作成したカスタムロールを選択します。

ステップ 12 [アクセス権の割り当て先（Assign Access To）] ドロップダウンリストはデフォルト値のままとします（Azure AD ユーザー、グループ、またはサービスプリンシパル）。

ステップ 13 [選択（Select）] テキストボックスに、先ほど作成したアプリケーションの名前（例：`Multicloud Defensecontrollerapp`）を入力し、[保存（Save）] をクリックします。

ステップ 14 [サブスクリプション（Subscription）] ページで、左側のメニューバーの [概要（Overview）] をクリックし、サブスクリプション ID をメモ帳にコピーします。

Multicloud Defense Controller のオンボーディングに必要な値

次に進む前に、以下に示す情報を用意してください。

- サブスクリプション ID（サブスクリプションの概要ページから）
- ディレクトリ（テナント）ID（Azure AD アプリの概要ページから）
- アプリケーション（クライアント）ID（Azure AD アプリの概要ページから）
- クライアントシークレット（クライアントシークレットの作成時にコピーしたもの）

マーケットプレイスの利用規約に同意する

Multicloud Defense Controller は、Azure マーケットプレイスの Multicloud Defense 仮想マシン（VM）イメージを使用してゲートウェイインスタンスを作成します。サブスクリプションごとに利用規約に同意する必要があります。Azure ポータル Web サイト（上部のメニューバーの右側）から Azure Cloud Shell を開きます。bash シェルを選択する、または bash シェルに切り替えて、次のコマンドを実行します（subscription-id を前のセクションでコピーしたサブスクリプション ID に置き換えます）。

```
az vm image terms accept --publisher valtix --offer datapath --plan valtix_dp_image
--subscription subscription-id
```

接続の段階的終了

Multicloud Defense Gatewayは、次のような複数の理由で、確立されたフローの終了を選択することがあります。

- ポリシーに基づく終了。たとえば、FQDNフィルタリングはフローが確立された後にのみ適用できます。
- IDS/IPSは、クライアントまたはサーバーによって送信されたフロー内のパケットを安全でないと見なし、確立されたフローの終了を選択することがあります。
- Multicloud Defense Gatewayのプロキシサービスが、フローが確立された後にフローの終了を決定した場合。
- Multicloud Defense Gateway TCPスタックのタイマーの1つが、フローがアクティブまたは動作中でないと判断した場合。
- PRSの更新、ゲートウェイ設定の変更など、特定の設定を変更している際のフローの終了。
- ゲートウェイがデコミッションされたときのフローの終了（コントローラが開始した無効化/アップグレード/スケールイン）。

上記のいずれかの理由でMulticloud Defense Gatewayが確立されたフローの終了を選択した場合、現時点では、クライアントとサーバーに終了について通知せずにフローが終了します（[拒否時のリセット（Reset on Deny）] がオンになっているFQDNフィルタリングがある場合を除く）。これにより、クライアントとサーバーはTCPまたはアプリケーションのタイムアウトに依存して接続の切断を検出することになり、アプリケーション障害を引き起こします。

TCPフローの場合、Multicloud Defense Gatewayは、ゲートウェイがフローを停止したときにクライアント（イニシエータ）にTCPリセットを送信する段階的終了メカニズムを導入します。これにより、クライアントTCPスタックは接続を迅速に終了でき、アプリケーションは中断されたフローの再確立を試行できるため、トラフィックの中断を最小限に抑えることができます。これは、Multicloud Defense Gatewayによって処理されるすべての種類のフロー（転送、フォワードプロキシ、およびリバースプロキシ）に適用されます。

また、Multicloud Defense Gatewayのデータプレーンが（ソフトウェアの問題が原因で）予期せずダウンした場合、このリセットメカニズムは適用されません。クライアントは引き続きアプリケーションのタイムアウトに依存して復旧することになります。

トラブルシューティング

Multicloud Defense GatewayによってTCPリセットで終了されたフローを検索するには、トラフィックの概要を（コントローラから）CSVとしてダウンロードし、RESETを検索します。これは入力フローの最後の接続状態になります。自然に終了した接続の場合、この状態が最後

の状態になることはありません。TCP 以外のフローでは、最後の接続状態は常に AGED OUT です。

クラウドアカウント用の Terraform オンボーディングスクリプト

オンボーディングウィザードまたは手動プロセスを使用する代わりに、Terraform スクリプトを使用してクラウド サービス プロバイダーのアカウントをオンボーディングできます。

Terraform について

Multicloud Defense のお客様は、**Terraform プロバイダー**を使用して以下の操作を行うことができます。[検出（Discover）]：パブリッククラウドアカウントをオンボードし、継続的なアセットの可視性を取得し、侵害の兆候（IoC）を検出します。[展開（Deploy）]：Multicloud Defense Gateway は、入力、出力、および East-West トラフィックを保護します。[防御（Defend）]：継続的に検出されるクラウドアセットを使用したマルチクラウド（AWS、Azure、GCP、OCI）の動的ポリシーを利用します。

注目 Multicloud Defense Controller バージョン 23.10 以降、Terraform プロバイダーを使用して GCP フォルダおよび GCP プロジェクトを接続できます。詳細については、Terraform リポジトリ（13 ページ）を参照してください。

Multicloud Defense Terraform プロバイダーは、Terraform レジストリから入手できる「検証済み」プロバイダーです。お客様は、Multicloud Defense の Terraform プロバイダーを使用してその運用にセキュリティを組み込むことができます。つまり、クラウドアカウントを Multicloud Defense にオンボードし、Multicloud Defense Gateway を展開して、インターネットからの侵入攻撃から保護し（WAF、IDS/IPS、Geo-IP）、出力トラフィックのデータ漏えいを食い止め（TLS 復号、IDS/IPS、AV、DLP、FQDN/URL フィルタリング）、VPC/VNets 間の East-West 攻撃を防ぎます。セキュリティポリシーは、クラウドアセットタグ（「dev」、「test」、「prod」、「pci」、「web」、「app1」など）に基づいて指定できます。

詳細については、以下を参照してください。

- Multicloud Defense 向けの Terraform プロバイダーのダウンロード。
- GitHub の例。
- Terraform に関する Multicloud Defense のブログ。

Terraform リポジトリ

ユース ケース	説明	Github リポジトリ
AWS オンボーディング	これは、Terraform を使用した AWS アカウントのオンボーディング用です。	AWS Github リポジトリ
AWS 検出 CFT	この CFT 展開には、Multicloud Defense の検出機能を使用するために必要なすべての権限が含まれます。完全な機能セットについては、ネイティブな製品 CFT を使用してください。	AWS 検出 Github リポジトリ
AWS 検出	これは、Terraform を使用した検出専用モードの AWS アカウントのオンボーディング用です。	AWS Github リポジトリ
Azure オンボーディング	これは、Terraform を使用した Azure サブスクリプションのオンボーディング用です。	Azure Github リポジトリ
GCP プロジェクトオンボーディング	これは、Terraform を使用した GCP プロジェクトのオンボーディング用です。	GCP Github リポジトリ
GCP フォルダのオンボーディング	これは、Terraform を使用した GCP フォルダのオンボーディング用です。	GCP Github リポジトリ

Terraform ブロックとしての設定のエクスポート

お客様は、セキュリティプロファイルを Multicloud Defense Controller から Terraform のリソースブロックにエクスポートできます。設定を Terraform ブロックにエクスポートするには、目的のセキュリティプロファイルに移動して選択し、[エクスポート（Export）] ボタンをクリックします。これにより、選択したオブジェクト/セキュリティプロファイルの Terraform ブロックを持つファイルがダウンロードされます。

すべてのオブジェクトとプロファイルは、次の例外を除き、Terraform のエクスポートをサポートします。

- ゲートウェイ
- サービス VPC/VNet

- 診断

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。