管理ガイド

AWS AppFabric

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS AppFabric: 管理ガイド

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon の商標およびトレードドレスはAmazon 以外の製品およびサービスに使用することはできません。また、お客様に誤解を与える可能性がある形式で、または Amazon の信用を損なう形式で使用することもできません。Amazon が所有していないその他のすべての商標は Amazon との提携、関連、支援関係の有無にかかわらず、それら該当する所有者の資産です。

Table of Contents

AWS AppFabric とは 1
製品 1
利点 1
ユースケース 1
AppFabric の仕組み 2
料金 3
可用性 3
AWS AppFabric for security とは 4
利点 1
ユースケース 1
AppFabric for security にアクセスする 5
関連サービス 5
OCSF スキーマ 7
AppFabric の OCSF ベースのスキーマ 7
前提条件と推奨事項 7
にサインアップする AWS アカウント 8
管理アクセスを持つユーザーを作成する 8
(必須) アプリケーションの前提条件を完了させてください 10
(オプション) 出力場所を作成します 11
(オプション) AWS KMS キーを作成する 13
はじめに 13
前提条件 14
ステップ 1: アプリケーションバンドルを作成する 14
ステップ 2: アプリケーションを認可する 16
ステップ 3: 監査ログの取り込みの設定 18
ステップ 4: ユーザーアクセスツールを使用する 21
ステップ 5: セキュリティツールやその他の転送先にある AppFabric for security データに接続する 23
サポートされているアプリケーション 23
1Password 25
Asana 28
Azure Monitor 30
Atlassian Confluence 34
Atlassian Jira suite 37

Box 41
Cisco Duo 44
Dropbox 47
Genesys Cloud 50
GitHub 53
Google 分析 57
Google Workspace 60
HubSpot 63
IBM Security® Verify 66
AppFabric JumpCloud用に を設定する 70
Microsoft 365 72
Miro 75
Okta 79
OneLogin 82
PagerDuty 85
Ping Identity 87
Salesforce 90
ServiceNow 95
Singularity Cloud 99
Slack 101
Smartsheet 106
Terraform Cloud 109
Webex by Cisco 111
Zendesk 115
Zoom 118
互換性のあるセキュリティツール 121
Barracuda XDR 122
Dynatrace 123
Logz.io 124
Netskope 125
NetWitness 126
QuickSight 127
Rapid7 128
Security Lake 129
Singularity Cloud 151
Splunk 151

リソースの削除 ... 152
取り込み先の削除 ... 153
取り込みの削除 ... 153
アプリ認可の削除 ... 153
アプリバンドルの削除 ... 154
AWS AppFabric for productivity とは ... 155
利点 ... 1
ユースケース ... 1
AppFabric for productivity へのアクセス ... 5
アプリ開発者向けの開始方法 ... 158
前提条件 ... 14
ステップ 1. AppFabric for productivity の AppClient を作成する ... 159
ステップ 2. アプリケーションを認証し認可する ... 162
ステップ 3. AppFabric ユーザーポータル URL をアプリケーションに追加する ... 164
ステップ 4. AppFabric を使用してクロスアプリケーションのインサイトとアクションを表示する ... 165
ステップ 5. AppFabric にアプリケーションの検証をリクエストする ... 172
AppClients の管理 ... 174
トラブルシューティング ... 181
エンドユーザー向けの開始方法 ... 186
前提条件 ... 14
ステップ 1. AppFabric にサインイン ... 187
ステップ 2. インサイトを表示することをアプリに許可する ... 189
ステップ 3. アプリケーションを接続してインサイトとアクションを生成する ... 190
ステップ 4. 自分のアプリケーションでインサイトを確認しクロスアプリケーションアクションを実行する ... 193
アクセスを管理する ... 199
トラブルシューティング ... 200
AppFabric for productivity APIs ... 203
アクション ... 204
データ型 ... 220
一般的なエラー ... 227
AppFabric でのデータ処理 ... 227
保管中の暗号化 ... 228
転送中の暗号化 ... 228
用語と概念 ... 229

セキュリティ 232
データ保護 233
保管中の暗号化 234
転送中の暗号化 234
キー管理 234
キーポリシー 235
AppFabric が で許可を使用する方法 AWS KMS 236
AppFabricの暗号化キーのモニタリング 238
Identity and Access Management 240
対象者 240
アイデンティティを使用した認証 241
ポリシーを使用したアクセスの管理 244
AWS AppFabric と IAM の連携方法 247
アイデンティティベースのポリシーの例 254
サービスにリンクされたロールの使用 265
AWS マネージドポリシー 267
トラブルシューティング 273
コンプライアンス検証 275
セキュリティに関するベストプラクティス 276
管理者アクセスなしでアプリケーションを監視する 276
AppFabric イベントを監視する 276
耐障害性 276
インフラストラクチャセキュリティ 277
設定と脆弱性の分析 277
モニタリング 278
CloudWatch によるモニタリング 278
CloudTrail ログ 279
CloudTrail での AppFabric 情報 280
AppFabric のログ ファイルエントリーの理解 281
クォータ 283
ドキュメント履歴 285
..... cclxxxix

AWS AppFabric とは

AWS AppFabric は組織全体の Software as a Service (SaaS) アプリケーションをすばやく接続するため、IT チームとセキュリティチームは標準スキーマを使用してアプリケーションを簡単に管理および保護でき、従業員は生成 AI を使用して日常業務を迅速に完了できます。

トピック

- 製品
- 利点
- ユースケース
- AppFabric の仕組み
- 料金
- 可用性

製品

AWS AppFabric の 2 つの側面について説明します。AppFabric for security、効率的な管理とセキュリティのために設計された AppFabric for security、および生成 AI 機能で強化された AppFabric for productivity (プレビュー) です。詳細については、以下の各トピックを参照してください。

- AWS AppFabric for security とは
- AWS AppFabric for productivity とは

利点

AppFabricでは、以下を行えます。

- アプリケーションを数分以内に接続し、運用コストを削減します。
- SaaS アプリケーションデータ全体の可視性を高め、セキュリティ体制を強化します。
- 生成 AI を使用してアプリケーション全体のタスクを自動的に実行できます。

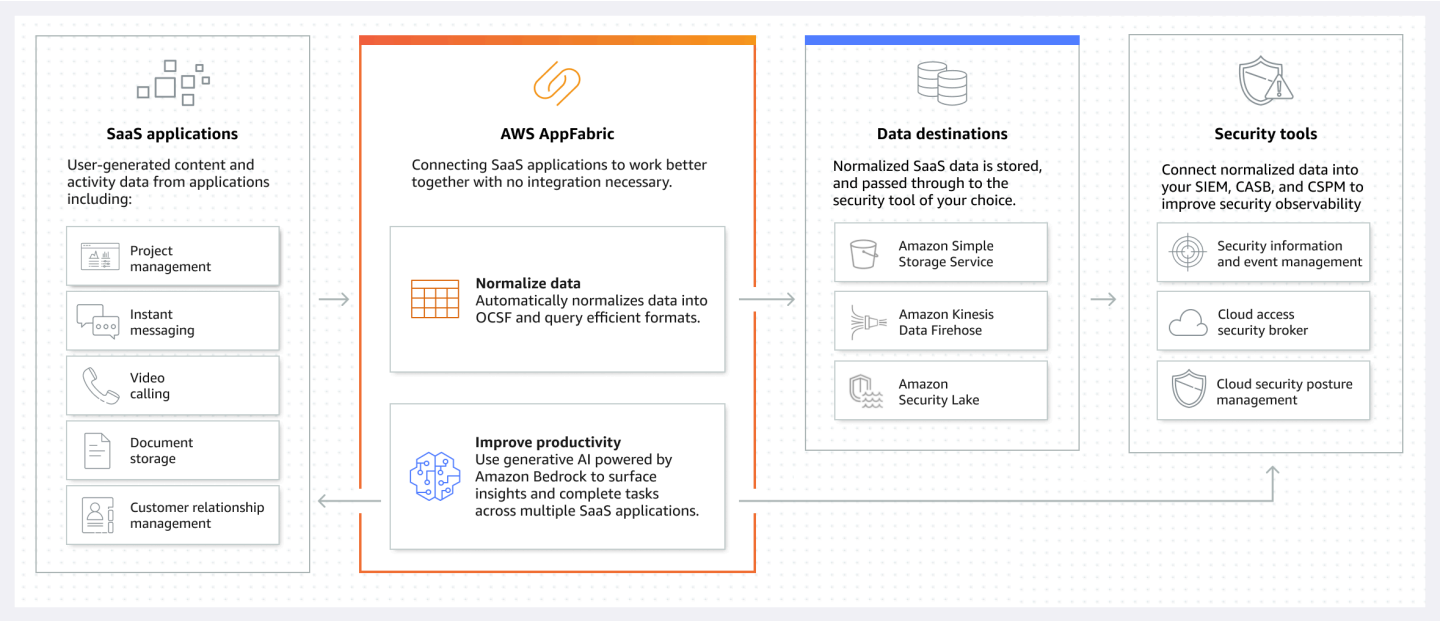
ユースケース

AppFabric を使用して次を実行できます。

- SaaS アプリケーションを迅速に接続する
  - AppFabric for security は、トップクラスの SaaS 生産性向上アプリケーションとセキュリティアプリケーションを相互にネイティブ接続し、フルマネージド型の SaaS 相互運用性ソリューションを提供します。
- セキュリティ体制を強化する
  - アプリケーションデータは自動的に正規化されるため、管理者は共通のポリシーを設定し、セキュリティアラートを標準化し、複数のアプリケーションにわたるユーザーアクセスを簡単に管理できます。
- 生産性を再構築する
  - AppFabric for productivity は、共通の生成 AI アシスタント機能により、従業員が迅速に回答を得て、タスク管理を自動化し、SaaS 生産性向上アプリケーション全体にわたるインサイトが生成できるようにします。

AppFabric の仕組み

AppFabric は、コーディングを必要とせずに複数の SaaS アプリケーションを迅速に接続できるため、生産性とセキュリティが向上します。AppFabric の利点を次の図に示します。



> **Note**
>
> AppFabric for Productivity は、現在プレビュー版でリリースされており、米国東部 (バージニア北部) AWS リージョンで利用できます。詳細については AWS リージョン、「」のAWS AppFabric エンドポイントとクォータ」を参照してくださいAWS 全般のリファレンス。

料金

AppFabric 料金の詳細と例については、「AWS AppFabricの料金」を参照してください。

可用性

AppFabric で現在サポートされている AWS リージョンとエンドポイントを確認するには、「 AWS 全般のリファレンス」のAWS AppFabric エンドポイントとクォータ」を参照してください。

AWS AppFabric for security とは

AWS AppFabric for security は、Software as a Service (SaaS) アプリケーションを組織全体にすばやく接続するため、IT チームとセキュリティチームは標準スキーマを使用してアプリケーションを簡単に管理および保護できます。

トピック

- 利点
- ユースケース
- AppFabric for security にアクセスする
- 関連サービス
- AWS AppFabric 用のオープンサイバーセキュリティスキーマフレームワーク
- AWS AppFabric を使用するための前提条件と推奨事項
- AWS AppFabric for security の使用を開始する
- AppFabric for security でサポートされているアプリケーション
- AppFabric for security の互換性のあるセキュリティツールとサービス
- セキュリティリソースの Delete AWS AppFabric

利点

AppFabric for security は以下の用途に使用できます。

- アプリケーションを数分以内に接続し、運用コストを削減します。
- SaaS アプリケーションデータ全体の可視性を高め、セキュリティ体制を強化します。

ユースケース

AppFabric for security は以下に使用できます。

- SaaS アプリケーションを迅速に接続する
  - AppFabric for security は、トップクラスの SaaS 生産性向上アプリケーションとセキュリティアプリケーションを相互にネイティブ接続し、フルマネージド型の SaaS 相互運用性ソリューションを提供します。

- セキュリティ体制を強化する
  - アプリケーションデータは自動的に正規化されるため、管理者は共通のポリシーを設定し、セキュリティアラートを標準化し、複数のアプリケーションにわたるユーザーアクセスを簡単に管理できます。

AppFabric for security にアクセスする

AppFabric for security は、米国東部 (バージニア北部) 、欧州 (アイルランド) 、アジアパシフィック (東京) で利用できます AWS リージョン。詳細については AWS リージョン、『』のAWS AppFabric エンドポイントとクォータ」を参照してくださいAWS 全般のリファレンス。

各リージョンでは、次のいずれかの方法で AppFabric for security にアクセスできます。

AWS Management Console

AWS Management Console は、 リソースの作成と管理 AWS に使用できるブラウザベースのインターフェイスです。AppFabric リソースには、AppFabric コンソールからアクセスできます。AppFabric コンソールを使用して、すべての AppFabric リソースを作成および管理することができます。

AppFabric API

AppFabricにプログラムからにアクセスするには、 AppFabric APIを使用します。これにより、サービスに HTTPS リクエストを直接発行できます。詳細については、 AWS AppFabric API リファレンスを参照してください。

AWS Command Line Interface (AWS CLI)

を使用すると AWS CLI、システムのコマンドラインでコマンドを発行して、AppFabric やその他のとやり取りできます AWS のサービス。コマンドラインツールは、タスクを実行するスクリプトを作成する場合にも便利です。のインストールと使用の詳細については AWS CLI、AWS Command Line Interface 「 バージョン 2 用ユーザーガイド」を参照してください。AppFabric の AWS CLI コマンドの詳細については、「 リファレンス」の AWS CLI AppFabric」セクションを参照してください。

関連サービス

AppFabric for security AWS のサービス では、以下を使用できます。

Amazon Data Firehose

Amazon Data Firehose は、データレイク、データストア、分析サービスにストリーミングデータを確実にキャプチャ、変換、配信する抽出、変換、ロード (ETL) サービスです。AppFabric を使用する場合、Open Cybersecurity Schema Framework (OCSF) の正規化された監査ログまたは raw 監査ログを JSON 形式で Firehose ストリームに送信先として出力することを選択できます。詳細については、「Firehose で出力場所を作成する」を参照してください。

Amazon Security Lake

Amazon Security Lake は、 AWS 環境、SaaS プロバイダー、オンプレミスおよびクラウドソースのセキュリティデータを、アカウントに保存されている専用のデータレイクに自動的に一元化します。AppFabric 監査ログデータを Security Lake と統合するには、Amazon Data Firehose を送信先として選択し、Security Lake で正しい形式とパスでデータを配信するように Firehose を設定します。詳細については、「Amazon Security Lake ユーザーガイド 」の「カスタムソースからのデータ収集」を参照してください。

Amazon Simple Storage Service

Amazon Simple Storage Service (Amazon S3) は、業界をリードするスケーラビリティ、データ可用性、セキュリティ、およびパフォーマンスを提供するオブジェクトストレージサービスです。AppFabric を使用する場合、新規または既存の Amazon S3 バケットを転送先とし、OCSF で正規化された監査ログ (JSON または Apache Parquet) とするか未加工の監査ログ (JSON)とするかを選択して出力することができます。詳細については、「Amazon S3 の出力場所の作成」を参照してください。

Amazon QuickSight

QuickSight は、ハイパースケールの統合ビジネスインテリジェンス (BI) でデータ駆動型組織を強化します。QuickSight を使用すると、最新のインタラクティブダッシュボード、ページ分割レポート、埋め込み分析、自然言語クエリを通じて、すべてのユーザーが同一の信頼できる情報源を使ってさまざまな分析ニーズに対応できるようになります。AppFabric ログが保存されている Amazon S3 バケットを情報源として選択することにより、QuickSightでAppFabricの監査ログデータを分析できます。詳細については、QuickSight ユーザーガイドのAmazon S3 ファイルを使用したデータセットの作成」を参照してください。また、Amazon S3 の AppFabric データを Amazon Athena にインポートし、QuickSight のデータソースとして Amazon Athena を選択することもできます。詳細については、QuickSight ユーザーガイドのAmazon Athena データを使用したデータセットの作成」を参照してください。

AWS Key Management Service

AWS Key Management Service （ AWS KMS) を使用すると、アプリケーションと 全体で暗号化キーを作成、管理、制御できます AWS のサービス。AppFabric でアプリバンドルを作成するときは、認証されたアプリケーションデータを安全に保護するための暗号化キーを設定します。このキーは AppFabric サービス内のデータを暗号化します。AppFabric は、ユーザーに代わって AppFabric が AWS 所有のキー 作成および管理している 、またはユーザーが作成および管理しているカスタマーマネージドキーを使用できます AWS KMS。詳細については、「 AWS KMS キーの作成」を参照してください。

AWS AppFabric 用のオープンサイバーセキュリティスキーマフレームワーク

Open Cybersecurity Schema Framework (OCSF) は、サイバーセキュリティ業界における AWS および主要なパートナーによる共同のオープンソースの取り組みです。OCSF は、一般的なセキュリティイベントの標準スキーマを提供し、スキーマの進化を容易にするバージョニング基準を定義し、セキュリティログの作成者と利用者を対象とした自己管理プロセスを組み込んでいます。OCSF の公開ソースコードは GitHubでホストされています。

AppFabric の OCSF ベースのスキーマ

AWS AppFabric for security OCSF 1.1 ベースのスキーマは、Software as a Service (SaaS) ポートフォリオの正規化され、一貫性があり、労力の少ないオブザーバビリティに対するニーズに対応するように特別に調整されています。AppFabric は、各フィールドとイベントに適したマッピングを決定します。AppFabric は、OCSF オープンソースコミュニティと協力して、OCSF を SaaS アプリケーションイベントに適用できるように、新しい OCSF イベントカテゴリ、イベントクラス、アクティビティ、オブジェクトを導入しました。AppFabric は、SaaS アプリケーションから受信した監査イベントを自動的に正規化し、このデータを の Amazon Simple Storage Service (Amazon S3) または Amazon Data Firehose サービスに配信します AWS アカウント。Amazon S3 の送信先では、2 つの正規化オプション (OCSF または Raw) と 2 つのデータ形式オプション (JSON またはParquet) を選択できます。Firehose に配信する場合、2 つの正規化オプション (OCSF または Raw) から選択することもできますが、データ形式は JSON に制限されます。

AWS AppFabric を使用するための前提条件と推奨事項

新規の AWS お客様は、 AWS AppFabric for security の使用を開始する前に、このページに記載されているセットアップの前提条件を完了してください。セットアップ手順には、 AWS Identity and Access Management (IAM) サービスを使用します。IAM の詳細については、「IAM ユーザーガイド」を参照してください。

トピック

- にサインアップする AWS アカウント
- 管理アクセスを持つユーザーを作成する
- (必須) アプリケーションの前提条件を完了させてください
- (オプション) 出力場所を作成します
- (オプション) AWS KMS キーを作成する

にサインアップする AWS アカウント

がない場合は AWS アカウント、次の手順を実行して作成します。

にサインアップするには AWS アカウント

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

   サインアップ手順の一環として、通話呼び出しまたはテキストメッセージを受け取り、電話キーパッドで検証コードを入力するように求められます。

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

AWS サインアッププロセスが完了すると、 から確認メールが送信されます。https://aws.amazon.com/ の [マイアカウント] をクリックして、いつでもアカウントの現在のアクティビティを表示し、アカウントを管理することができます。

管理アクセスを持つユーザーを作成する

にサインアップしたら AWS アカウント、日常的なタスクにルートユーザーを使用しないように AWS アカウントのルートユーザー、 を保護し AWS IAM Identity Center、 を有効にして管理ユーザーを作成します。

を保護する AWS アカウントのルートユーザー

1. ルートユーザーを選択し、 AWS アカウント E メールアドレスを入力して、アカウント所有者AWS Management Consoleとして にサインインします。次のページでパスワードを入力します。

   ルートユーザーを使用してサインインする方法については、AWS サインイン ユーザーガイドのルートユーザーとしてサインインするを参照してください。

2. ルートユーザーの多要素認証 (MFA) を有効にします。

   手順については、IAM ユーザーガイドの AWS アカウント 「ルートユーザー (コンソール) の仮想 MFA デバイスを有効にする」を参照してください。

管理アクセスを持つユーザーを作成する

1. IAM アイデンティティセンターを有効にします。

   手順については、「AWS IAM Identity Center ユーザーガイド」の「AWS IAM Identity Centerの有効化」を参照してください。

2. IAM アイデンティティセンターで、ユーザーに管理アクセスを付与します。

   を ID ソース IAM アイデンティティセンターディレクトリ として使用する方法のチュートリアルについては、「 AWS IAM Identity Center ユーザーガイド」の「デフォルトを使用してユーザーアクセスを設定する IAM アイデンティティセンターディレクトリ」を参照してください。

管理アクセス権を持つユーザーとしてサインインする

- IAM アイデンティティセンターのユーザーとしてサインインするには、IAM アイデンティティセンターのユーザーの作成時に E メールアドレスに送信されたサインイン URL を使用します。

  IAM Identity Center ユーザーを使用してサインインする方法については、AWS サインイン 「 ユーザーガイド」の AWS 「 アクセスポータルにサインインする」を参照してください。

追加のユーザーにアクセス権を割り当てる

1. IAM アイデンティティセンターで、最小特権のアクセス許可を適用するというベストプラクティスに従ったアクセス許可セットを作成します。

手順については、「AWS IAM Identity Center ユーザーガイド」の「権限設定を作成する」を参照してください。

2. グループにユーザーを割り当て、そのグループにシングルサインオンアクセス権を割り当てます。

   手順については、「AWS IAM Identity Center ユーザーガイド」の「グループの結合」を参照してください。

(必須) アプリケーションの前提条件を完了させてください

AppFabric for security を使用してアプリケーションからユーザー情報や監査ログを受信する際、アプリケーションの多くで特定のロールとプランタイプが必要になります。AppFabric for security で認可する各アプリケーションの前提条件を確認し、適切なプランとロールがあることを確認してください。アプリケーション別の前提条件の詳細については、「サポートされているアプリケーション」を参照するか、以下のアプリケーション別のトピックのいずれかを選択してください。

- AppFabric 1Password用に を設定する
- AppFabric Asana用に を設定する
- AppFabric Azure Monitor用に を設定する
- AppFabric Atlassian Confluence用に を設定する
- AppFabric Atlassian Jira suite用に を設定する
- AppFabric Box用に を設定する
- AppFabric Cisco Duo用に を設定する
- AppFabric Dropbox用に を設定する
- AppFabric Genesys Cloud用に を設定する
- AppFabric GitHub用に を設定する
- AppFabric Google Analytics用に を設定する
- AppFabric Google Workspace用に を設定する
- AppFabric HubSpot用に を設定する
- AppFabric IBM Security® Verify用に を設定する
- AppFabric JumpCloud用に を設定する
- AppFabric Microsoft 用に 365 を設定する

- AppFabric Miro用に を設定する
- AppFabric Okta用に を設定する
- AppFabric OneLogin by One Identity用に を設定する
- AppFabric PagerDuty用に を設定する
- AppFabric Ping Identity用に を設定する
- AppFabric Salesforce用に を設定する
- AppFabric ServiceNow用に を設定する
- AppFabric Singularity Cloud用に を設定する
- AppFabric Slack用に を設定する
- AppFabric Smartsheet用に を設定する
- AppFabric Terraform Cloud用に を設定する
- AppFabric Webex by Cisco用に を設定する
- AppFabric Zendesk用に を設定する
- AppFabric Zoom用に を設定する

(オプション) 出力場所を作成します

AppFabric for security は、監査ログの取り込み先として Amazon Simple Storage Service (Amazon S3) と Amazon Data Firehose をサポートしています。

Amazon S3

取り込み先を作成する際、AppFabric コンソールを使用して新しい Amazon S3 バケットを作成できます。また、Amazon S3 サービスを使用してバケットを作成することもできます。Amazon S3 サービスを使用してバケットを作成する場合は、AppFabric の取り込み先を作成する前にバケットを作成し、取り込み先を作成する際にバケットを選択する必要があります。既存のバケットの次の要件を満たしている限り AWS アカウント、 で既存の Amazon S3 バケットを使用できます。

- AppFabric for security では、Amazon S3 バケットが、Amazon S3 リソースと同じ AWS リージョン に存在する必要があります。
- は、次のいずれかを使用してバケットを暗号化できます。
  - Amazon S3 マネージドキーを用いたサーバー側の暗号化 (SSE-S3)
  - default AWS Key Management Service （ AWS KMS) を使用した () キーによるサーバー側の暗号化 (SSE-KMS) AWS マネージドキー `aws/s3`。

Amazon Data Firehose

セキュリティデータの AppFabric の取り込み先として Amazon Data Firehose を使用することを選択できます。Firehose を使用するには、取り込みを作成する AWS アカウント 前、または AppFabric で取り込み先を作成するときに、 に Firehose 配信ストリームを作成できます。Firehose 配信ストリームは、 AWS Management Console、、 AWS CLIまたは AWS APIs または SDKs を使用して作成できます。ストリーム設定の手順については、以下のトピックを参照してください。

- AWS Management Console 手順 – Amazon Data Firehose デベロッパーガイドの「Amazon Data Firehose 配信ストリームの作成」
- AWS CLI 手順 – AWS CLI コマンドリファレンスcreate-delivery-streamの「」
- AWS APIsと SDKsCreateDeliveryStream「」

Amazon Data Firehose を AppFabric for security 出力先として使用する場合の要件は次のとおりです。

- ストリームは、AppFabric for security リソース AWS リージョン と同じ に作成する必要があります。
- ソースとして [ダイレクト PUT] を選択する必要があります。
- AmazonKinesisFirehoseFullAccess AWS 管理ポリシーをユーザーにアタッチするか、以下のアクセス権限をユーザーにアタッチします。

```
{
    "Sid": "TagFirehoseDeliveryStream",
    "Effect": "Allow",
    "Action": ["firehose:TagDeliveryStream"],
    "Condition": {
        "ForAllValues:StringEquals": {"aws:TagKeys": "AWSAppFabricManaged"}
    },
    "Resource": "arn:aws:firehose:*:*:deliverystream/*"
}
```

Firehose は、 Splunkや などのさまざまなサードパーティーのセキュリティツールとの統合をサポートしていますLogz.io。これらのツールにデータを出力するように Amazon Kinesis を適切に設定する方法については、「Amazon Data Firehose デベロッパーガイド」の「送信先設定」を参照してください。

(オプション) AWS KMS キーを作成する

AppFabric for security のアプリケーションバンドルを作成する過程で、許可されたすべてのアプリケーションのデータを安全に保護するための暗号化キーを選択または設定します。このキーは、AppFabric サービス内のデータの暗号化に使用されます。

AppFabric for security はデフォルトでデータを暗号化します。AppFabric for security は、ユーザーに代わって AppFabric によって作成および管理される AWS 所有のキー 、または AWS Key Management Service () で作成および管理されるカスタマー管理キーを使用できますAWS KMS。AWS 所有のキー は、 が複数の で使用するために AWS のサービス 所有および管理する AWS KMS キーのコレクションです AWS アカウント。カスタマーマネージドキーは、 AWS アカウント ユーザーが作成、所有、管理する の AWS KMS キーです。 AWS 所有のキー およびカスタマーマネージドキーの詳細については、「 AWS Key Management Service デベロッパーガイド」の「カスタマーキーと AWS キー」を参照してください。

AppFabric for security 内のカスタマーマネージドキーを使用して認可トークンなどのデータを暗号化する場合は、AWS KMSを使用して作成できます。でカスタマーマネージドキーへのアクセスを許可するアクセス許可ポリシーの詳細については AWS KMS、このガイドの「キーポリシー」セクションを参照してください。

AWS AppFabric for security の使用を開始する

AWS AppFabric for security の使用を開始するには、先にアプリバンドルを作成してから、アプリケーションを認可しアプリバンドルに接続する必要があります。アプリ認証がアプリケーションに接続されると、監査ログの取り込みやユーザーアクセスなどの AppFabric for security の機能を使用できるようになります。

このセクションでは、 で AppFabric の使用を開始する方法について説明します AWS Management Console。

トピック

- 前提条件
- ステップ 1: アプリケーションバンドルを作成する
- ステップ 2: アプリケーションを認可する
- ステップ 3: 監査ログの取り込みの設定
- ステップ 4: ユーザーアクセスツールを使用する
- ステップ 5: セキュリティツールやその他の転送先にある AppFabric for security データに接続する

前提条件

開始する前に、まず AWS アカウント と管理ユーザーを作成する必要があります。詳細については、「にサインアップする AWS アカウント」および「管理アクセスを持つユーザーを作成する」を参照してください。

ステップ 1: アプリケーションバンドルを作成する

アプリバンドルには、AppFabric for security アプリの承認と取り込みがすべて保存されます。アプリバンドルを作成するには、認証されたアプリケーションデータを安全に保護するための暗号化キーを設定します。

1. https://console.aws.amazon.com/appfabric/ にある AppFabric コンソールを開きます。
2. ページの右上隅にある [リージョンの選択] セレクターで AWS リージョンを選択します。AppFabric は米国東部 (バージニア北部)、欧州 (アイルランド)、およびアジアパシフィック (東京) の各リージョンでのみご利用いただけます。
3. [開始方法] を選択します。
4. [開始方法] ページの [ステップ 1] を行います。[アプリバンドルの作成] で [アプリバンドルの作成] を選択します。
5. [暗号化] セクションで、認証されたすべてのアプリケーションからのデータを安全に保護するための暗号化キーを設定します。このキーは、AppFabric for security サービス内の、データの暗号化に使用されます。

   AppFabric for security はデフォルトでデータを暗号化します。AppFabric は、ユーザーに代わって AppFabric によって AWS 所有のキー 作成および管理される 、または () で AWS Key Management Service 作成および管理されるカスタマーマネージドキーを使用できますAWS KMS。
6. [AWS KMS キー] には、[使用 AWS 所有のキー] または [カスタマーマネージドキー] を選択します。

   カスタマーマネージドキーを選んで使用する場合は、Amazon リソースネーム (ARN) または使用したい既存のキーのキー ID のいずれかを入力するか、あるいは [AWS KMS キーの作成] を選択します。

   AWS 所有のキー またはカスタマーマネージドキーを選択するときは、次の点を考慮してください。

- AWS 所有のキー は、 が複数の で使用するために AWS のサービス 所有および管理する AWS Key Management Service （ AWS KMS) キーのコレクションです AWS アカウント。AWS 所有のキー は にはありませんが AWS アカウント、 は AWS 所有のキー を使用してアカウントのリソースを保護 AWS のサービス できます。アカウントのクォータには AWS KMS カウント AWS 所有のキー されません。キーまたはそのキーポリシーを作成または管理する必要はありません。のローテーションはサービス AWS 所有のキー によって異なります。AppFabric の AWS 所有のキー ローテーションの詳細については、「保管データ暗号化」を参照してください。
- カスタマーマネージドキーは、ユーザーが作成、所有、管理する の KMS キー AWS アカウント です。これらの AWS KMS キーは完全に制御できます。キーポリシー、 AWS Identity and Access Management (IAM) ポリシー、グラントを確立し維持することができます。それらを有効または無効にしたり、暗号化マテリアルをローテーションしたり、タグを追加したり、AWS KMS キーを参照するエイリアスを作成したり、 AWS KMS キーの削除をスケジュールしたりできます。カスタマーマネージドキーは、 AWS Management Console のカスタマーマネージドキーページに表示されます AWS KMS。

  カスタマーマネージドキーを明確に識別するには、`DescribeKey` オペレーションを使用します。カスタマーマネージドキーでは、`DescribeKey` レスポンスの `KeyManager` フィールドの値は `CUSTOMER` です。暗号化オペレーションではカスタマーマネージドキーを使用し、AWS CloudTrail ログでは使用状況を監査できます。と統合 AWS のサービス する多くの では AWS KMS、カスタマーマネージドキーを指定して、保存および管理されるデータを保護できます。カスタマーマネージドキーには、月額料金と AWS 無料利用枠を超える使用料が発生します。カスタマーマネージドキーは、アカウントの AWS KMS クォータに対してカウントされます。

AWS 所有のキー およびカスタマーマネージドキーの詳細については、「 AWS Key Management Service デベロッパーガイド」の「カスタマーキーと AWS キー」を参照してください。

> **Note**
>
> アプリバンドルが作成されると、AppFabric for security は AWS アカウント に AppFabric サービスにリンクされたロール (SLR) と呼ばれる特別な IAM ロールも作成します。これにより、サービスは Amazon CloudWatch にメトリクスを送信することができます。監査ログの送信先を追加すると、SLR は AppFabric for security サービスに AWS リソース (Amazon S3 バケット、Amazon Data Firehose 配信ストリーム) へのア

クセスを許可します。詳細については、「AppFabric のサービスリンクロールの使用」を参照してください。

7. (オプション) [タグ] で、アプリバンドルにタグを追加することができます。タグは、作成したリソースにメタデータを割り当てるキーと値のペアです。詳細については、「タグエディタユーザーガイド」の AWS「リソースのタグ付け」を参照してください。 AWS
8. アプリバンドルを作成するには、[アプリバンドルの作成] を選択します。

ステップ 2: アプリケーションを認可する

アプリバンドルが正常に作成されたら、AppFabric for security に各アプリケーションへの接続と操作を許可できるようになります。認証されたアプリケーションは暗号化され、アプリバンドルに保存されます。アプリバンドルごとに複数のアプリ認可を設定するには、アプリケーションごとに必要に応じてアプリ認可手順を繰り返します。

アプリケーションを認可する手順を開始する前に、AppFabric for security でサポートされているアプリケーションで各アプリケーションの前提条件 (必要なプランタイプなど) をよく確認してください。

1. [開始方法] ページの [ステップ 2] を行います。アプリケーションを承認し、アプリケーション認可の作成を選択します。
2. アプリ認可セクションで、アプリケーションドロップダウンから AppFabric for security が に接続するためのアクセス許可を付与するアプリケーションを選択します。表示されるアプリケーションは、現在 AppFabric for security でサポートされているものです。
3. アプリケーションを選択すると、必須の情報フィールドが表示されます。これらのフィールドには、テナント ID とテナント名のほか、クライアント ID、クライアントシークレット、または個人アクセストークンが含まれる場合があります。これらのフィールドの入力値はアプリケーションによって異なります。これらの値の検索方法に関するアプリケーション別の詳細な手順については、「AppFabric for security でサポートされているアプリケーション」を参照してください。
4. (オプション) タグには、アプリ認可にタグを追加するオプションがあります。タグは、作成したリソースにメタデータを割り当てるキーと値のペアです。詳細については、「タグエディタユーザーガイド」の AWS「リソースのタグ付け」を参照してください。 AWS
5. 「アプリ認可の作成」を選択します。
6. ポップアップウィンドウが表示されたら (接続中のアプリケーションによる)、[許可] を選択して AppFabric for security のアプリケーションへの接続を許可します。

アプリ認可が成功すると、「開始方法」ページにアプリ認可の成功メッセージが表示されます。

7. アプリ認可のステータスは、ナビゲーションペインに一覧表示されているアプリ認可ページで、各アプリケーションのステータスでいつでも確認できます。接続ステータスは、AppFabric for security がアプリケーションに接続するためのアプリ認可が付与され、完了していることを意味します。
8. 関連するエラーを修正するために実行できるトラブルシューティング手順を含め、考えられるアプリ認可ステータスを以下の表に示します。

ステータス名	ステータス情報	トラブルシューティングのステップ
[保留中]	[保留中] というステータスは、アプリケーションのアプリ認可は作成されているが、AppFabric for security がまだアプリケーションに接続されていないことを意味します。	このステータスが表示されたら、アプリ認可ページのアクションドロップダウンから接続を選択して接続を開始します。この エラーが解決されない場合は、ブラウザのポップアップブロッカーが無効になっていないか確認してください。ポップアップウィンドウに [400 不良なリクエスト] などのエラーメッセージが表示される場合は、テナント ID、クライアント ID、クライアントシークレットなどのすべての情報が正しく入力されていることを確認してください。また、アプリケーションのアプリ認可が正しく作成されていない可能性もあります。詳細については、「サポートされるアプリケーション」を参照してください。
[接続が検証できませんでした]	「接続が検証できませんでした」のステータスは、AppF	テナント ID、クライアント ID、クライアントシークレッ

ステータス名	ステータス情報	トラブルシューティングのステップ
	abric for security がアプリ認可とアプリケーションとの接続を検証できないことを意味します。	トなどのすべての情報がアプリ認可用に正しく入力されていることを確認してください。
[トークンの自動ローテーションに失敗しました]	「トークンの自動ローテーションに失敗しました」のステータスは、アプリ認可が正常に接続された後に OAuth 更新トークンが失敗したことを意味します。	このエラーが解決されない場合は、アプリケーションの認証アプリを確認してください。詳細については、「サポートされるアプリケーション」を参照してください。

9. 他のアプリケーションを認可するには、必要に応じてステップ 1 ~ 8 を繰り返します。

ステップ 3: 監査ログの取り込みの設定

アプリバンドルで少なくとも 1 つのアプリ認可を作成したら、監査ログの取り込みを設定できるようになります。監査ログの取り込みにより、認証アプリからの監査ログが消費され、オープンサイバーセキュリティスキーマフレームワーク (OCSF) に標準化されます。次に、それらは AWS内の1つまたは複数の転送先に配信されます。Raw JSON ファイルを転送先に配信することもできます。

1. [開始方法] ページの [ステップ 3] を行います。[監査ログ取り込みの設定] セクションで、[取り込みの Quick Setup] を選択します。

> **Note**
>
> セットアップを迅速に行うには、[開始方法] ページからのみアクセスできる [取り込みの Quick Setup] ページを使用して、同じ転送先に対する複数のアプリ認証の取り込みを一度に作成します。たとえば、同じ Amazon S3 バケットまたは Amazon Data Firehose データストリームなどです。
> ナビゲーションペインからアクセスできる [取り込み] ページから取り込みを作成することもできます。[取り込み] ページでは、異なる転送先への取り込みを一度に 1 つずつ設定できます。[取り込み] ページでは、取り込みのタグを作成することもできます。以下は、[取り込みのクイックセットアップ] ページの説明です。

2. [アプリ認証の選択] で、監査ログの取り込みを作成したいアプリ認証を選択します。App authorizations ドロップダウンに表示されるテナント名は、AppFabric for security で のアプリケーション認可を以前に作成したアプリケーションのテナント名です。
3. [転送先の追加] では、選択したアプリケーションの監査ログの取り込み先を選択します。送信先オプションには、Amazon S3 - 既存のバケット、Amazon S3 - 新しいバケット、Amazon Data Firehose などがあります。複数のテナント名を選択した場合、選択した転送先がアプリケーション認可の取り込みのたびに適用されます。
4. 転送先を選択すると、追加の必須フィールドが表示されます。
   a. [Amazon S3 — 新規バケット] を転送先として選択した場合は、作成したい S3 バケットの名前を入力する必要があります。Amazon S3 バケットの作成に関する詳しい手順については、「出力先の作成」を参照してください。
   b. [Amazon S3 — 既存のバケット] を送信先として選択した場合は、使用したい Amazon S3 バケットの名前を選択します。
   c. 送信先として Amazon Data Firehose を選択した場合は、Firehose 配信ストリーム名ドロップダウンリストから配信ストリームの名前を選択します。Amazon Data Firehose 配信ストリームを作成する方法の詳細については、「出力先の作成」およびAppFabric for security」に必要なアクセス許可ポリシーを書き留めてください。
5. Schema & Format では、監査ログを Raw - JSON、OCSF - JSON、OCSF - Amazon S3 バケットParquetの場合は 、Raw - JSON または OCSF-JSON for Firehose に保存できます。

   Raw データ形式では、監査ログデータがデータ文字列から JSON に変換されます。OCSF データ形式は、監査ログデータを AppFabric for security のオープンサイバーセキュリティスキーマフレームワーク (OCSF) スキーマに正規化します。AppFabric がOCSF を使用する方法については、「AWS AppFabric 用のオープンサイバーセキュリティスキーマフレームワーク」を参照してください。一度に取り込むことができるスキーマと形式のデータタイプは 1 つだけです。スキーマと形式のデータタイプを追加する場合は、取り込み作成プロセスを繰り返すことで追加の取り込み先を設定できます。
6. (オプション) 取り込みにタグを追加する場合は、ナビゲーションペインの [取り込み] ページに移動します。「取り込みの詳細」ページに移動するには、テナント名を選択します。[タグ] で、取り込みにタグを追加することができます。タグは、作成したリソースにメタデータを割り当てるキーと値のペアです。詳細については、「タグエディタユーザーガイド」の AWS「リソースのタグ付け」を参照してください。 AWS
7. [取り込みの設定] を選択します。

取り込みの設定が正常に完了すると、[開始方法] ページに [取り込みが作成されました] という成功メッセージが表示されます。

8. また、ナビゲーションペインの [取り込み] ページで、取り込みの状態と取り込み先のステータスをいつでも確認できます。このページでは、アプリ認可の作成時に作成されたテナント名、転送先、および取り込みの状態を確認することができます。取り込みの状態が [有効] の場合は、取り込みが有効になっていることを意味します。このページでアプリ認可のテナント名を選択すると、転送先の詳細やステータスなど、そのアプリ認可の詳細ページが表示されます。取り込み先のステータスが [有効] の場合は、その取り込み先が適切に設定され、有効になっていることを意味します。アプリ認可のステータスが接続済みで、取り込み先のステータスがアクティブである場合、監査ログを処理して配信する必要があります。アプリ認可ステータスまたは取り込み先ステータスがいずれかの「失敗」状態である場合、取り込みステータスが有効になっていても監査ログは処理も配信もされません。アプリケーション認可の失敗を修正するには、「ステップ 2」を参照してください。アプリケーションを認可する。

9. エラーステータスを修正するために実行できるトラブルシューティング手順を含め、考えられる取り込み先と取り込み先ステータスを以下の表に示します。

状態またはステータス名	説明	トラブルシューティングのステップ
[Disabled] (無効)	取り込みが [無効] の状態になっている場合、取り込みは無効になっています。	取り込みを有効にするには、[取り込み] ページの [アクション] ドロップダウンから [有効にする] を選択します。
失敗	取り込み先が [失敗] の状態になっている場合、取り込み先が監査ログを受け付けていないことを意味します。例えば、保存場所がいっぱいのためにこの状態になることがあります。	これらの問題を解決するには、Amazon S3 または Firehose コンソールに移動します。

ステップ 4: ユーザーアクセスツールを使用する

AppFabric for security ユーザーアクセスツールを使用すると、セキュリティチームと IT 管理者チームは、従業員の会社のメールアドレスを使った簡単な検索を実行することで特定のアプリケーションへのアクセス権を持つ人をすばやく確認することができます。このアプローチは、ユーザーのプロビジョニング解除など、SaaS アプリケーション全体にわたるユーザーアクセスを手動で確認または監査する必要があるタスクに費やす時間を削減するのに役立ちます。ユーザーが特定できたら、AppFabric for security はアプリケーション内のユーザー名と、アプリケーションが提供している場合はアプリ内ユーザーステータス (有効など) を表示します。AppFabric for security はアプリバンドル内のすべての認証済みアプリケーションを検索して、ユーザーがアクセスできるアプリケーションのリストを返します。

1. [開始方法] ページの [ステップ 4] を行います。[ユーザーアクセスツール] を使用して、[ユーザーの検索] を選択します。
2. [メールアドレス] フィールドに、ユーザーのメールアドレスを入力し、[検索] を選択します。
3. [検索結果] セクションには、ユーザーがアクセスできるすべての認証済みアプリケーションのリストが表示されます。アプリケーション内のユーザー名とステータス (可能な場合) を表示するには、検索結果を選択します。
4. 検索結果列に [ユーザーが見つかりました] というメッセージが表示されている場合は、そのユーザーはリストに表示されているアプリにアクセスできることを意味します。考えられる検索結果、エラー、およびエラーに対処するために実行できるアクションを以下の表で示します。

検索結果	説明
ユーザーが見つかりません	使用されたメールアドレスを持つユーザーが見つかりません。
認可トークンが見つかりません。アプリケーションのアプリ認可に接続します。	テナント ID、クライアント ID、クライアントシークレットなどのすべての情報がアプリ認可用に正しく入力されていることを確認してください。
認可トークンは取り消されました。アプリケーションのアプリ認可に接続します。	テナント ID、クライアント ID、クライアントシークレットなどのすべての情報がアプリ認可用に正しく入力されていることを確認してください。

検索結果	説明
認可トークンをローテーションできませんでした。アプリケーションのアプリ認可に接続します。	アプリ認可が正常に接続された後、OAuth 更新トークンは失敗しました。この エラーが解決されない場合は、アプリケーションの認証アプリを確認してください。詳細については、「サポートされるアプリケーション」を参照してください。
必要な許可が見つかりません。アプリケーションのアプリ認可に接続します。	テナント ID、クライアント ID、クライアントシークレットなどのすべての情報がアプリ認可用に正しく入力されていることを確認してください。
アプリ認可が無効です。	テナント ID、クライアント ID、クライアントシークレットなどのすべての情報がアプリ認可用に正しく入力されていることを確認してください。
アクセス許可が不十分なため、アプリケーション API を呼び出すことができませんでした。	テナント ID、クライアント ID、クライアントシークレットなどのすべての情報がアプリ認可用に正しく入力されていることを確認してください。
アプリケーションリクエスト制限を超えました。	これはアプリケーションから受け取ったエラーメッセージです。後でもう一度メールアドレスを検索してください。
アプリケーションに内部サーバーエラーが発生しました	これはアプリケーションから受け取ったエラーメッセージです。後でもう一度メールアドレスを検索してください。
アプリケーションに不正なゲートウェイエラーが発生しました	これはアプリケーションから受け取ったエラーメッセージです。後でもう一度メールアドレスを検索してください。

検索結果	説明
アプリケーションはリクエストを処理する準備ができていません	これはアプリケーションから受け取ったエラーメッセージです。後でもう一度メールアドレスを検索してください。
アプリケーションに不正なリクエストエラーが発生しました。	これはアプリケーションから受け取ったエラーメッセージです。後でもう一度メールを検索してください。
アプリケーションでサービス使用不可エラーが発生しました。	これはアプリケーションから受け取ったエラーメッセージです。後でもう一度メールを検索してください。

ステップ 5: セキュリティツールやその他の転送先にある AppFabric for security データに接続する

AppFabric からの正規化された (または raw) アプリケーションデータは、、、、、、、、 などのセキュリティツールSplunkや独自のセキュリティソリューションなど、Amazon S3 からのデータ取り込みと Firehose Barracuda XDR Dynatrace Logz.io Netskope NetWitness Rapid7との統合をサポートする任意のツールと互換性があります。AppFabric から正規化された (または未加工の) アプリケーションデータを取得するには、前のステップ 1 ~ 3 に従います。特定のセキュリティツールやサービスの設定方法の詳細については、「互換性のあるセキュリティツールとサービス」を参照してください。

AppFabric for security でサポートされているアプリケーション

AWS AppFabric for security は、以下のアプリケーションとの統合をサポートしています。AppFabric for security を設定して接続する方法の詳細を見るには、アプリケーションの名前を選択します。

トピック

- AppFabric 1Password用に を設定する
- AppFabric Asana用に を設定する
- AppFabric Azure Monitor用に を設定する
- AppFabric Atlassian Confluence用に を設定する

- AppFabric Atlassian Jira suite用に を設定する
- AppFabric Box用に を設定する
- AppFabric Cisco Duo用に を設定する
- AppFabric Dropbox用に を設定する
- AppFabric Genesys Cloud用に を設定する
- AppFabric GitHub用に を設定する
- AppFabric Google Analytics用に を設定する
- AppFabric Google Workspace用に を設定する
- AppFabric HubSpot用に を設定する
- AppFabric IBM Security® Verify用に を設定する
- AppFabric JumpCloud用に を設定する
- AppFabric Microsoft 用に 365 を設定する
- AppFabric Miro用に を設定する
- AppFabric Okta用に を設定する
- AppFabric OneLogin by One Identity用に を設定する
- AppFabric PagerDuty用に を設定する
- AppFabric Ping Identity用に を設定する
- AppFabric Salesforce用に を設定する
- AppFabric ServiceNow用に を設定する
- AppFabric Singularity Cloud用に を設定する
- AppFabric Slack用に を設定する
- AppFabric Smartsheet用に を設定する
- AppFabric Terraform Cloud用に を設定する
- AppFabric Webex by Cisco用に を設定する
- AppFabric Zendesk用に を設定する
- AppFabric Zoom用に を設定する

AppFabric 1Password用に を設定する

1Password は、すべてのオンラインアカウントに強力なパスワードを作成、保存、使用するのに役立つパスワードマネージャーです。また、暗号化を使用してデータを保護し、違反について警告し、パスワードを共有することもできます。

AWS AppFabric for security を使用すると、 からログとユーザーデータを監査し1Password、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- 1Password での AppFabric のサポート
- AppFabric を 1Password アカウントに接続する

1Password での AppFabric のサポート

AppFabric は、1Password からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して 1Password からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- 有効な有料 1Password Business または Enterprise サブスクリプションプランが必要です。詳細については、 1Passwordウェブサイトの1Password「Enterprise」を参照してください。
- 1Password アカウントには管理者ロールまたはチーム所有者が必要です。詳細については、1Passwordサポートウェブサイトの「グループ」を参照してください。

レート制限に関する考慮事項

1Password AuditLog Events API は、リクエストを 1 分あたり 600 件、1 時間あたり最大 30,000 件に制限します。これらの制限を超えると、エラーが返されます。詳細については、「 Events 1Password API reference」の「API Rate limits」を参照してください。 1Password

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によ

るものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を 1Password アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、1Passwordを使用して AppFabric を認可する必要があります。AppFabric 1Passwordで を認可するために必要な情報を見つけるには、次のステップを使用します。

個人用1Passwordアクセストークンを作成する

1Password は、パブリッククライアントの個人用アクセストークンをサポートしています。個人用アクセストークンを生成するには、次の手順を実行します。

1. 1Password アカウントにサインインします。
2. ナビゲーションペインで統合を選択します。
3. 既存の統合が存在する場合は、ディレクトリを選択します。それ以外の場合は、次の手順に進んでください。
4. イベントレポート統合でその他を選択します。
5. 統合の追加ページで、セキュリティ情報とイベント管理 (SIEM) システム名 (AppFabric Secure など) を入力します。
6. 統合の追加を選択し、トークンの設定ページで次の手順を実行します。
   a. AppFabric セキュア環境で使用するトークン名を指定します。
   b. 「有効期限後」ドロップダウンリストで「Never」を選択することをお勧めします。 他の値が選択されている場合、 は有効期限が経過した後にトークンを1Password取り消します。
   c. レポートするイベントセクションで、サインイン試行、アイテム使用状況イベント、監査イベントを選択します。
7. 問題トークンを選択してトークンを作成します。
8. 保存 1Password を選択し、次のステップを完了します。
   a. タイトルは、システム名とトークン名に基づいて自動的に入力されます。
   b. ボールトの選択でプライベートを選択します。
   c. [Save] を選択します。

詳細については、 1Passwordウェブサイトの1Password「イベントレポートの開始方法」を参照してください。

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。AppFabric のテナント ID は1Passwordサインインアドレスになります。テナント ID を検索するには、次のステップを実行します。

1. 1Password アカウントにサインインします。
2. ナビゲーションペインで [設定] を選択します。
3. 1Password サインインがページに表示されます。例えば、example-account.1password.com です。

テナント名

この一意の 1Password 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

サービスアカウントトークン

AppFabric 1Passwordアプリ認可に入力するには、 サービスアカウントの1Passwordサービスアカウントトークンが必要です。サービスアカウントトークンをお持ちでない場合は、次に説明する手順に従ってください。

AppFabric はサービスアカウントトークンをリクエストします。AppFabric のサービスアカウントトークンは、作成した個人用アクセストークンです。1Password ポータルで次の手順を実行して、個人用アクセストークンを見つけます。

1. Dashboard を選択します。
2. People を選択します。
3. アカウント所有者名を選択します。
4. [プライベート] を選択してください。
5. ボールトの表示 を選択します。
6. トークン名を選択します。

クライアント認可

テナント ID、テナント名、サービスアカウントトークンを使用して AppFabric でアプリ認可を作成します。次に、接続を選択して認可をアクティブ化します。

AppFabric Asana用に を設定する

Asana は、日常業務から部門横断的な戦略的イニシアチブに至るまで、個人、チーム、組織が仕事を調整できるよう支援する業務管理プラットフォームです。誰もがコミュニケーション、コラボレーション、仕事の調整を行える、生き生きとしたわかりやすいシステムを提供します。Asana を使うと、チームは重要なビジネスツールを 1 か所に統合できるため、どこにいても仕事を進めることができます。

AWS AppFabric for security を使用すると、 からログとユーザーデータを監査しAsana、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Asana での AppFabric のサポート
- AppFabric を Asana アカウントに接続する

Asana での AppFabric のサポート

AppFabric は、Asana からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Asana からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- Asana の エンタープライズアカウントが必要です。Asana エンタープライズアカウントの作成またはアップグレードに関する詳細については、Asana ウェブサイトの「Asana エンタープライズ」を参照してください。
- Asana アカウントにはスーパー管理者ロールを持つユーザーが必要です。ロールの詳細については、Asana ウェブサイトの「Asana の管理者およびスーパー管理者」ロールを参照してください。

レート制限に関する考慮事項

Asana は、Asana API にレート制限を課します。AsanaAPI のレート制限の詳細については、「Asana デベロッパーガイド」ウェブサイトの「レート制限」を参照してください。AppFabric と既存の Asana アプリケーションの組み合わせが制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Asana アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Asanaを使用して AppFabric を認可する必要があります。AppFabric Asanaで認可するために必要な情報を見つけるには、次のステップを使用します。

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。AppFabric のテナント ID は、Asana ではドメイン ID と呼ばれています。ドメイン ID を見つけるには、Asana ホーム画面で以下の指示に従ってください。

1. アカウントのプロフィール画像を選択し、[管理コンソール] を選択します。
2. [設定] を選択します。
3. [ドメイン設定] までスクロールします。
4. このセクションのドメイン ID を [AppFabric テナント ID 設定] に入力します。

テナント名

この一意の Asana 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

サービスアカウントトークン

AppFabric Asanaアプリ認可に入力するには、 サービスアカウントのAsanaサービスアカウントトークンが必要です。サービスアカウントトークンをお持ちでない場合は、次に説明する手順に従ってください。

1. サービスアカウントを作成するには、「Asana ガイド」ウェブサイトの「サービスアカウント」の指示に従います。

2. [サービスアカウントの追加] ページが初めて表示されたときに、[サービスアカウントの追加] ページの下部にあるトークンをコピーして保存します。
3. トークンを保存する前に [サービスアカウントの追加] ページを閉じた場合は、サービスアカウントを編集し、新しいトークンを生成して保存する必要があります。

AppFabric Azure Monitor用に を設定する

Azure Monitor は、クラウド環境とオンプレミス環境からモニタリングデータを収集、分析、対応するための包括的なモニタリングソリューションです。を使用してAzure Monitor、アプリケーションとサービスの可用性とパフォーマンスを最大化できます。これにより、アプリケーションのパフォーマンスを理解し、システムイベントに手動およびプログラムで対応できます。

Azure Monitor は、複数の Azure および Azure 以外のサブスクリプションとテナントにわたって、システムのすべてのレイヤーとコンポーネントからデータを収集して集約します。これは、データを相関、分析、視覚化、および/または応答できる共通のツールセットで使用するために、共通のデータプラットフォームに保存されます。他の Microsoft ツールと Microsoft 以外のツールを統合することもできます。Azure Monitor アクティビティログは、サブスクリプションレベルのイベントに関するインサイトを提供するプラットフォームログです。アクティビティログには、リソースが変更されたときや仮想マシンが開始されたときなどの情報が含まれます。

AWS AppFabric for security を使用すると、 からログとユーザーデータを監査しAzure Monitor、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Azure Monitor での AppFabric のサポート
- AppFabric を Azure Monitor アカウントに接続する

Azure Monitor での AppFabric のサポート

AppFabric は、次のAzure Monitorサービスからユーザー情報と監査ログを受信できます。

- Azure Monitor
- API Management
- Microsoft Sentinel
- Security Center

前提条件

AppFabric を使用して Azure Monitor からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- 無料トライアルまたはpay-as-you-goのサブスクリプションのMicrosoft Azureアカウントが必要です。
- そのサブスクリプション内のイベントを取得するには、少なくとも 1 つのサブスクリプションが必要です。

レート制限に関する考慮事項

Azure Monitor は、リクエストを行うセキュリティプリンシパル (ユーザーまたはアプリケーション) とサブスクリプション ID またはテナント ID にレート制限を課します。Azure Monitor API レート制限の詳細については、Azure Monitor「 デベロッパーウェブサイト」の「 がリクエストAzure Resource Managerを調整する方法を理解する」を参照してください。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Azure Monitor アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Azure Monitorを使用して AppFabric を認可する必要があります。AppFabric Azure Monitorで を承認するために必要な情報を見つけるには、次のステップを使用します。

OAuth アプリケーションの作成

AppFabric は OAuth2 Azure Monitorを使用して と統合します。で OAuth2 アプリケーションを作成するには、次の手順を実行しますAzure Monitor。

1. Microsoft Azure ポータルに移動し、サインインします。
2. Microsoft Entra ID に移動します。
3. アプリ登録を選択します。

4. 「新規登録」を選択します。
5. OAuth Azure Monitor クライアントなどのクライアントの名前を入力します。これは登録されたアプリケーションの名前になります。
6. サポートされているアカウントタイプがシングルテナントに設定されていることを確認します。
7. リダイレクト URI では、プラットフォームとして Web を選択し、リダイレクト URI を追加します。リダイレクト URI には次の形式を使用します。

   ```
   https://<region>.console.aws.amazon.com/appfabric/oauth2
   ```

   このアドレス AWS リージョン の *<region>*は、AppFabric アプリバンドルを設定した のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。そのリージョンのリダイレクト URL は `https://`*`us-east-1`*`.console.aws.amazon.com/appfabric/oauth2` です。

   認証レスポンスは、ユーザーの認証に成功すると、指定された URI に送信されます。これを指定することはオプションであり、後で変更できますが、ほとんどの認証シナリオでは値が必要です。
8. [登録] を選択します。
9. 登録されたアプリで、証明書とシークレットを選択し、新しいクライアントシークレットを選択します。
10. シークレットの説明を追加します。
11. シークレットの有効期限を選択します。ドロップダウンから任意のプリセット期間を選択するか、カスタム期間を設定できます。
12. [Add] (追加) を選択します。クライアントシークレット値は、作成直後にのみ表示できます。ページを離れる前に、必ずシークレットを安全な場所に保存してください。

必要なアクセス許可

OAuth アプリケーションには以下のアクセス許可を追加する必要があります。アクセス許可を追加するには、「 Microsoft Entraデベロッパーガイド」の「ウェブ API にアクセスするためのアクセス許可を追加する」セクションの手順に従います。

- Microsoft Graph ユーザーアクセス API > User.Read.All (委任タイプを選択)
- Microsoft Graph ユーザーアクセス API > offline_access (委任タイプを選択)
- Azure サービス管理監査ログ API > user_impersonation (委任タイプを選択)

アクセス許可を追加した後、アクセス許可に対する管理者の同意を付与するには、 Microsoft Entraデベロッパーガイドの「管理者の同意ボタン」セクションの指示に従います。

アプリ権限

AppFabric は、Azure Monitorアカウントからのユーザー情報と監査ログの受信をサポートしています。から監査ログとユーザーデータの両方を受信するにはAzure Monitor、2 つのアプリケーション認可を作成する必要があります。1 つはアプリケーション認可ドロップダウンリストAzure Monitorで、もう 1 つはアプリケーション認可ドロップダウンリストで Azure Monitor Audit Logs という名前です。両方のアプリ認証には、同じテナント ID、クライアント ID、およびクライアントシークレットを使用できます。から監査ログを受信するにはAzure Monitor、 Azure Monitorと Azure Monitor Audit Logs の両方のアプリ認可が必要です。ユーザーアクセスツールを単独で使用するには、Azure Monitorアプリケーション認可のみが必要です。

テナント ID

AppFabric はテナント ID を要求します。Azure Monitor でクライアント ID を検索するには、次の手順を実行します。

1. Microsoft Azure ポータルに移動します。
2. Azure Active Directory に移動します。
3. アプリ登録セクションで、以前に作成されたアプリを選択します。
4. 概要セクションで、ディレクトリ (テナント) ID フィールドからテナント ID をコピーします。

テナント名

この一意のAzure Monitorサブスクリプションを識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

> **Note**
> テナント名は、数字、小文字/大文字、およびピリオド (.)、アンダースコア (_)、ダッシュ (-)、空白の特殊文字で構成される最大 2,048 文字にする必要があります。

クライアント ID

AppFabric はクライアント ID を要求します。でクライアント ID を検索するには、次の手順を実行しますAzure Monitor。

1. Microsoft Azure ポータルに移動します。
2. Azure Active Directory に移動します。
3. アプリ登録セクションで、以前に作成されたアプリを選択します。
4. 概要セクションで、アプリケーション (クライアント) ID フィールドからクライアント ID をコピーします。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。登録された OAuth アプリのクライアントシークレットは、OAuth アプリ作成セクションのステップ 11 で生成したものです。OAuth アプリの作成中に生成されたクライアントシークレットを紛失した場合は、OAuth アプリの作成セクションのステップ 8 ~ 11 を繰り返して、新しいシークレットを再生成します。

アプリ認可

AppFabric でアプリ認可を作成すると、 から認可を承認Microsoft Azureするためのポップアップウィンドウが表示されます。ウィンドウからアカウントにサインインし、許可を選択して AppFabric 認可を承認します。

AppFabric Atlassian Confluence用に を設定する

すべての作業を 1 か所で作成、コラボレーション、整理できます。Confluence は、知識とコラボレーションとが融合するチームワークのスペースです。ダイナミックページでは、チームは、あらゆるプロジェクトやアイデアを作成、記録し、コラボレーションすることができます。スペースでは、チームは、作業を構築、整理、共有することができ、チームメンバー全員が組織内の情報を把握したり仕事で最善を尽くすために必要な情報にアクセスしたりできます。

AWS AppFabric for security を使用すると、 から監査ログとユーザーデータを受信しConfluence、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Atlassian Confluence での AppFabric のサポート
- AppFabric を Atlassian Confluence アカウントに接続する

Atlassian Confluence での AppFabric のサポート

AppFabric は、Atlassian Confluence からの監査ログの受信をサポートしています。

前提条件

AppFabric を使用して Atlassian Confluence からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- 監査ログにアクセスするには、スタンダード、プレミアム、エンタープライズのいずれかのアカウントが必要です。該当する Confluence プランタイプを作成またはアップグレードする際の詳細については、Atlassian のウェブサイトの「Confluence Pricing」を参照してください。
- 監査ログにアクセスするには、お使いのアカウントの、管理者のアクセス許可が必要になります。ロールの詳細については、Atlassian Support Webサイトの「ユーザーに管理者権限を付与する」を参照してください。

レート制限に関する考慮事項

Confluence は、Atlassian Confluence API にレート制限を課します。AppFabric と既存の Atlassian Confluence API アプリケーションの組み合わせが Atlassian Confluence の制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Atlassian Confluence アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Atlassian Confluenceを使用して AppFabric を認可する必要があります。AppFabric Atlassian Confluenceで を承認するために必要な情報を見つけるには、次のステップを使用します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して Atlassian Confluence と統合されます。Atlassian Confluence で OAuth アプリケーションを作成するには、以下の手順に従います。

1. Atlassian 開発者コンソールに移動します。

2. 右上にあるプロフィールアイコンを選択し、[開発者コンソール] を選択します。
3. [マイアプリ] の横にある [作成]、[OAuth 2.0 統合] を選択します。
4. 左側のナビゲーションペインで [アクセス権限] を選択し、Confluence API の横にある [追加] を選択します。
5. [クラシックスコープ] で、[ユーザーの読み取り] (`read:confluence-user`) を選択します。
6. [詳細スコープ] で [監査記録を表示] (`read:audit-log:confluence`) を選択します。
7. 左側のナビゲーションペインで [承認] を選択し、[OAuth 2.0 (3LO)] の横にある [追加] を選択します。
8. [コールバック URL] テキストボックスに、リダイレクト URL を以下の形式で入力し、[変更を保存] を選択します。

   ```
   https://<region>.console.aws.amazon.com/appfabric/oauth2
   ```

   この URL で、*<#####>*は、AppFabric アプリバンドルを構成した AWS リージョン のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。そのリージョンのリダイレクト URL は `https://us-east-1.console.aws.amazon.com/appfabric/oauth2` です。

必要範囲

Atlassian Confluence OAuth アプリケーションに次のスコープのいずれか 1 つを入力する必要があります。スコープに関する詳細は、Atlassian 開発者ウェブサイトの「Scopes for OAuth 2.0 (3LO) and Forge apps」を参照してください。可能な場合はクラシックスコープを使用します。

- クラシックスコープ:
  - `read:confluence-user`
- 詳細なスコープ:
  - `read:audit-log:confluence`

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。AppFabric のテナント ID は [Atlassian Confluenceインスタンスサブドメイン] です。[Atlassian Confluence インスタンスのサブドメイン] は、ブラウザのアドレスバーの [https://] と [.atlassian.net] との間にあります。

テナント名

この一意の Atlassian Confluence 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。Atlassian Confluenceでクライアント ID を検索するには以下の手順を使用してください。

1. Atlassian 開発者コンソールに移動します。
2. 右上にあるプロフィールアイコンを選択し、[開発者コンソール]、[自分のアプリケーション] の順に選択します。
3. AppFabric との接続に使用するOAuthアプリを選択します。
4. [設定] ページのクライアント ID を AppFabric のクライアント ID フィールドに入力します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。以下の手順でAtlassian Confluenceのクライアント シークレット を検索してください。

1. Atlassian 開発者コンソールに移動します。
2. 右上にあるプロフィールアイコンを選択し、[開発者コンソール]、[自分のアプリケーション] の順に選択します。
3. AppFabric との接続に使用するOAuthアプリを選択します。
4. AppFabric の [クライアントシークレット] フィールドに [設定] ページからシークレットを入力します。

認可を承認します

AppFabric でアプリ認可を作成すると、 から認可を承認Atlassian Confluenceするためのポップアップウィンドウが表示されます。AppFabric 認可を承認するには、許可を選択します。

AppFabric Atlassian Jira suite用に を設定する

Atlassianはすべてのチームの可能性を解き放ちます。同社のアジャイルとDevOps、ITサービス管理、ワークマネジメントソフトウェアは、チームが共有作業の整理、議論、完了に役立ちます。

フォーチュン500企業の過半数や、NASA、Kiva、Deutsche Bank、Salesforce などを含む世界中のあらゆる規模の24万社を超える企業が、チームの連携を強化し、質の高い結果を予定通りに達成するため Atlassian のソリューションに頼っています。Jira Software、Confluence、Jira Service Management、Trello、Bitbucket および Jira Align などの Atlassian 製品について詳しくは Atlassian をご覧ください。

AWS AppFabric for security を使用して、 Jira suite (以外) からログとユーザーデータを監査し Jira Align、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Jira suite での AppFabric のサポート
- AppFabric を Jira アカウントに接続する

Jira suite での AppFabric のサポート

AppFabric は、Jira Align を除いて、Jira suite からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Jira suite からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- Jiraスタンダードプラン以上に加入している必要があります。Jiraプランの機能について詳しくは、Jiraソフトウェア、Jiraサービス管理、Jiraワークマネジメント、Jiraプロダクトディスカバリーの料金ページをご覧ください。
- Jiraアカウントには組織管理ロールを持つユーザーが必要です。ロールの詳細については、Atlassian Support Webサイトの「ユーザーに管理者権限を付与する」を参照してください。

レート制限に関する考慮事項

Jira が、JiraAPI にレート制限を課します。Jira suite API のレート制限について詳しくは、Web サイトの「Atlassian 開発者ガイド」の「レート制限」を参照してください。AppFabric と既存の Jira API アプリケーションの組み合わせが制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Jira アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Jiraを使用して AppFabric を認可する必要があります。AppFabric Jiraで を承認するために必要な情報を見つけるには、次のステップを使用します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して Jira suite と統合されます。Jiraで OAuth アプリケーションを作成するには、以下の手順に従います。

1. Atlassian開発者コンソールに移動します。
2. [マイアプリ] の横にある [作成]、[OAuth 2.0 統合] を選択します。
3. アプリに名前を付け、[作成] を選択します。
4. 認可セクションに移動し、OAuth 2.0 の横にある追加を選択します。
5. [コールバック URL] フィールドに以下の形式の URL を入力し、[変更を保存] を選択します。

   ```
   https://<region>.console.aws.amazon.com/appfabric/oauth2
   ```

   この URL で、<#####> iは、AppFabric アプリバンドルを構成した AWS リージョン のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。そのリージョンのリダイレクト URL は `https://us-east-1.console.aws.amazon.com/appfabric/oauth2` です。
6. 設定セクションに移動し、クライアント ID とクライアントシークレットをコピーして、AppFabric アプリ認可に使用するように保存します。

必要範囲

Jira OAuth アプリの [許可] ページに次のスコープを追加する必要があります。

- [クラシックスコープ] で:

  - Jira API > `read:jira-user`
- グラニュラースコープでは:
  - Jira API > `read:audit-log:jira`
  - Jira API > `read:user:jira`

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。AppFabric のテナント ID は [Jiraインスタンスサブドメイン] です。[Jira インスタンスのサブドメイン] は、ブラウザのアドレスバーの [https://] と [.atlassian.net] との間にあります。

テナント名

この固有のJiraサーバーを識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。以下の手順で、Jiraでクライアント ID を検索してください。

1. Atlassian開発者コンソールに移動します。
2. AppFabric との接続に使用するOAuthアプリを選択します。
3. [設定] ページのクライアント ID を AppFabric のクライアント ID フィールドに入力します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。AppFabric の [クライアントシークレット] は Jiraの [シークレット] です。以下の手順で Jira の [シークレット] を検索してください。

1. Atlassian開発者コンソールに移動します。
2. AppFabric との接続に使用するOAuthアプリを選択します。
3. AppFabric の [クライアントシークレット] フィールドに [設定] ページからシークレットを入力します。

認可を承認します

AppFabric でアプリ認可を作成すると、 から認可を承認Jiraするためのポップアップウィンドウが表示されます。AppFabric 認可を承認するには、許可を選択します。

AppFabric Box用に を設定する

Box は、組織がコンテンツのライフサイクル全体を管理し、どこからでも安全に作業し、best-of-breedアプリケーションと統合できるようにする単一のプラットフォームである、主要なコンテンツクラウドです。

AWS AppFabric を使用すると、 から監査ログとユーザーデータを受信しBox、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Box での AppFabric のサポート
- AppFabric を Box アカウントに接続する

Box での AppFabric のサポート

AppFabric は、Box からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Box からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- 監査ログにアクセスするには、ビジネス、ビジネスプラス、エンタープライズ、またはエンタープライズプラスプランへの有効な有料サブスクリプションが必要です。
- 管理者権限を持つユーザーが必要です。
- 設定タブからアプリケーションのクライアントシークレットを表示およびコピーするには、Boxアカウントで 2 要素認証を有効にする必要があります。

レート制限に関する考慮事項

Box は、Box API にレート制限を課します。Box API レート制限の詳細については、 Boxデベロッパーガイドウェブサイトの「レート制限」を参照してください。AppFabric と既存の Box アプリ

ケーションの組み合わせが制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントで送信先に配信されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Box アカウントに接続する

AppFabric サービス内でアプリバンドルを作成したら、 で AppFabric を承認する必要がありますBox。AppFabric Boxで を認可するために必要な情報を見つけるには、次のステップを使用します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して Box と統合されます。で OAuth アプリケーションを作成するには、次のステップを使用します。詳細についてはBox、 Boxウェブサイトの「OAuth アプリケーションの作成」を参照してください。

1. にログインBoxし、 デベロッパーコンソールに移動します。
2. [新しいアプリの作成] を選択します。
3. アプリケーションタイプのリストからカスタムアプリケーションを選択します。モーダルが表示され、次のステップの選択を求められます。
4. アプリの名前と説明を入力します。
5. 目的のドロップダウンリストから統合を選択します。
   a. カテゴリドロップダウンリストからセキュリティとコンプライアンスを選択します。
   b. どの外部システムと統合していますか？テキストボックスAWS AppFabric Secureに「」と入力します。
6. クライアント ID とクライアントシークレットを使用してアプリケーション ID を検証する場合は、サーバー認証 (クライアント認証情報付与) を選択します。
7. [Create App (アプリの作成)] を選択します。
8. [設定] タブを選択します。
9. ページのアプリアクセスレベルセクションで、アプリ + エンタープライズアクセスを選択します。

10. ページの「アプリケーションスコープ」セクションで、「ユーザーの管理」と「エンタープライズプロパティの管理」を選択します。
11. [Save Changes] を選択します。

    Box 管理者は、アプリケーションを使用する前にBox、管理者コンソール内でアプリケーションを認可する必要があります。認可をリクエストするには、次の手順を実行します。

    a. デベロッパーコンソールでアプリケーションの認可タブを選択します。
    b. レビューと送信を選択して、承認のためにBoxエンタープライズ管理者に E メールを送信します。詳細については、「 Boxガイド」の「認可」を参照してください。

    > **Note**
    > 送信後に変更があった場合は、アプリを再送信する必要があります。

必要範囲

以下のアプリケーションスコープが必要です。スコープの詳細については、Box ドキュメントウェブサイトの「スコープ」を参照してください。

- エンタープライズプロパティを管理する (`manage_enterprise_properties`)
- ユーザーを管理する (`manage_managed_users`)

アプリ権限

テナント ID

AppFabric はテナント ID をリクエストします。AppFabric のテナント ID はBoxエンタープライズ ID です。Box エンタープライズ ID は、管理者コンソールの「アカウントと請求 > アカウント情報 > エンタープライズ ID」にあります。詳細については、Box ドキュメントウェブサイトの「エンタープライズ ID」を参照してください。

テナント名

この一意の Box 組織を識別する名前を入力します。AppFabric はテナント名を使用して、アプリ認可とアプリ認可から作成された取り込みにラベルを付けます。

クライアント ID とクライアントシークレット

1. にログインBoxし、 デベロッパーコンソールに移動します。
2. ナビゲーションメニューでアプリを選択します。
3. AppFabric の接続に使用する OAuth アプリケーションを選択します。
4. [設定] タブを選択します。
5. ページの「Oauth 2.0 認証情報」セクションまでスクロールします。
6. OAuth クライアント ID のクライアント ID を AppFabric のクライアント ID フィールドに入力します。
7. クライアントシークレットの取得を選択します。
8. OAuth クライアントシークレットから AppFabric のクライアントシークレットフィールドにクライアントシークレットを入力します。

AppFabric Cisco Duo用に を設定する

Cisco Duo は、強力な多層防御と革新的な機能を提供する主要なアクセス管理スイートを使用して、侵害から保護します。これにより、正当なユーザーが に入り、悪意のある攻撃者を防ぐことができます。侵害を懸念し、迅速にソリューションを必要とする組織にとって、 はユーザーの生産性を向上させながら、強力なセキュリティCisco Duoを迅速に実現します。

AWS AppFabric for security を使用すると、 から監査ログとユーザーデータを受信しCisco Duo、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Cisco Duo での AppFabric のサポート
- AppFabric をCisco Duoアカウントに接続する

Cisco Duo での AppFabric のサポート

AppFabric は、Cisco Duo からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Cisco Duo からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- 監査ログにアクセスするには、Duo Essentials、Duo Advantage、または Duo Premier エディションへのアクティブなサブスクリプションが必要です。または、 Advantage または Premier トライアルの新規のお客様も にアクセスできます。Cisco Duo エディションの詳細については、「エディションと料金表」を参照してください。
- Admin API を作成または変更するには、所有者ロールを持つ管理者である必要があります。
- 管理者 API の監査ログにアクセスするには、読み取りログリソースの付与「」アクセス許可を追加する必要があります。

レート制限に関する考慮事項

Cisco Duo は、Cisco Duo API にレート制限を課します。Cisco Duo API レート制限の詳細については、「認証ログ」の「レート制限」を参照してください。AppFabric と既存の Cisco Duo API アプリケーションの組み合わせが Cisco Duo の制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。レート制限の引き上げが必要な場合は、Cisco Duo にお問い合わせください。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric をCisco Duoアカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Cisco Duoを使用して AppFabric を認可する必要があります。AppFabric Cisco Duoで を認可するために必要な情報を見つけるには、次のステップを使用します。

Cisco Duo Admin API アプリケーションを作成する

AppFabric は API サービストークンCisco Duoを使用して と統合します。でアプリケーションを作成するにはCisco Duo、次の手順を実行します。

- Cisco Duo Admin API アプリケーションを作成するには、Cisco DuoAdmin API の最初のステップの手順に従います。

必要なアクセス許可

Cisco Duo アプリケーションには、次のスコープを追加する必要があります。

- 読み取りログを付与する
- 読み取りリソースを付与する

アプリ権限

テナント ID

AppFabric はテナント ID をリクエストします。テナント ID はCisco Duoホスト名にあります。でホスト名を検索するにはCisco Duo、次の手順に従います。

1. Cisco Duo 管理者ログインページに移動し、サインインします。
2. アプリケーションに移動し、アプリケーションを保護するを選択します。
3. アプリケーションリストで Admin API のエントリを見つけ、右端に保護を選択してアプリケーションを設定し、API ホスト名を取得します。
4. API ホスト名の形式は でapi-*<tenant-id>*.duosecurity.com、 *<tenant-id>*はテナント ID です。

テナント名

この一意の Cisco Duo 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

サービストークン

AppFabric はサービストークンをリクエストします。サービストークンは、次の形式のコロン区切りの統合キーとシークレットキーです。

```
integrationkey:secretkey
```

で統合キーとシークレットキーを検索するにはCisco Duo、次の手順を実行します。

1. Cisco Duo 管理者ログインページに移動し、サインインします。
2. アプリケーションに移動し、アプリケーションを保護するを選択します。

3. 「アプリケーションを保護する」をクリックし、アプリケーションリストで Admin API の エントリを見つけます。右端にある Protect をクリックしてアプリケーションを設定します。スコープセクションまでスクロールし、 **`Grant read log`** と を追加します**`Grant read resource`**。

AppFabric Dropbox用に を設定する

Dropbox は、何に取り組んでいるのか、どこで働いているか、どのツールを使っているかに関わらず、従業員が一丸となることで、組織がより良い仕事をより早く成し遂げられるように支援します。ユーザーは、シンプルで安全な方法でコンテンツを共有できるようになるため、イノベーションと効率性を加速することができます。Dropbox は 1 か所で生活を整理し、仕事をスムーズに進められる場所を提供します。180 か国、7 億人以上の登録ユーザーを有する Dropbox は、より賢明な働き方をデザインすることを使命としています。

AWS AppFabric for security を使用すると、 からログとユーザーデータを監査しDropbox、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Dropbox での AppFabric のサポート
- AppFabric を Dropbox アカウントに接続する

Dropbox での AppFabric のサポート

AppFabric は、Dropbox からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Dropbox からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- Dropbox ビジネスアカウントを持っている必要があります。Dropbox ビジネスアカウントの作成またはアップグレードの詳細については、Dropbox ウェブサイトの「Dropbox ビジネス」を参照してください。
- Dropbox アカウントにはチーム管理者ロールを持つユーザーが必要です。ロールに関する詳細については、「Dropbox ヘルプセンター」ウェブサイトの「Dropbox チームの管理者権限を変更する方法」を参照してください。

レート制限に関する考慮事項

Dropbox は、Dropbox API にレート制限を課します。Dropbox API のレート制限に関する詳細については、「Dropbox パフォーマンスガイド」ウェブサイトの「レート制限」を参照してください。AppFabric と既存の Dropbox API アプリケーションの組み合わせが制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Dropbox アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Dropboxを使用して AppFabric を認可する必要があります。AppFabric Dropboxで を承認するために必要な情報を見つけるには、次のステップを使用します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して Dropbox と統合されます。Dropbox で OAuth アプリケーションを作成するには、以下の手順に従います。

1. https://www.dropbox.com/developers/apps のDropboxアプリコンソールで [アプリの作成] を選択します。
2. 新しいアプリケーション設定ページで、API の [範囲指定アクセス] を選択します。
3. 次に、Dropbox アクセスの種類として [Full] を選択します。
4. OAuth アプリケーションに名前を付け、[アプリの作成] を選択して OAuth アプリケーションの初期設定を完了します。
5. アプリケーション情報ページの 「OAuth2 リダイレクト URI」 フィールドに、以下の形式のリダイレクト URL を入力します。

   ```
   https://<region>.console.aws.amazon.com/appfabric/oauth2
   ```

   この URL AWS リージョン では、 *<region>*は AppFabric アプリバンドルを設定した のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。その

リージョンのリダイレクト URL は `https://us-east-1.console.aws.amazon.com/appfabric/oauth2` です。

6. [追加] を選択します。
7. AppFabricアプリ認可に使用するアプリキーとアプリシークレットをコピーして保存します。
8. [設定] タブの他のフィールドはすべてデフォルト値のままでかまいません。

必要範囲

アプリ情報画面の [許可] タブを使用して、次の範囲を Dropbox アプリに入力します。

- `account_info.read`
- `team_data.member`
- `events.read`
- `members.read`
- `team_info.read`

終了したら、[送信] を選択します。

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。チーム名など、Dropbox アカウントを一意に識別する任意の値を入力します。

テナント名

この一意の Dropbox アカウントを識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。AppFabric のクライアント ID は Dropbox アプリケーションキーです。Dropbox アプリキーを確認するには、以下のステップに従います。

1. https://www.dropbox.com/developers/apps の Dropbox アプリコンソールに移動します。
2. AppFabric との接続に使用するアプリを検索します。

3. アプリ情報ページの [ステータス] セクションでアプリキーを検索します。
4. AppFabric の [クライアント ID] フィールドに Dropbox アプリのアプリキーを入力します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。AppFabric のクライアントシークレットは、Dropbox アプリシークレットです。Dropbox アプリシークレットを確認するには、以下のステップに従います。

1. https://www.dropbox.com/developers/apps の Dropbox アプリコンソールに移動します。
2. AppFabric との接続に使用するアプリを検索します。
3. アプリ情報ページの [ステータス] セクションでアプリシークレットを検索します。
4. AppFabric の [クライアントシークレット] フィールドに Dropbox アプリのアプリシークレットを入力します。

認可を承認します

AppFabric でアプリ認可を作成すると、 から認可を承認Dropboxするためのポップアップウィンドウが表示されます。AppFabric 認可を承認するには、許可を選択します。

AppFabric Genesys Cloud用に を設定する

Genesys Cloud は、使いやすいオールインワンのインターフェイスで、デジタルチャネルと音声チャネルを横断するスムーズな会話を実現します。これにより企業は、従業員と顧客に優れたエクスペリエンスを提供し、導入の迅速化、複雑さの解消、管理の簡素化といった数多くのメリットを享受できます。

AWS AppFabric for security を使用すると、 から監査ログとユーザーデータを受信しGenesys Cloud、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Genesys Cloud での AppFabric のサポート
- AppFabric を Genesys Cloud アカウントに接続する

Genesys Cloud での AppFabric のサポート

AppFabric は、Genesys Cloud からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Genesys Cloud からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- Genesys Cloud アカウントが必要です。
- Genesys Cloudアカウントには管理者ロールを持つユーザーが必要です。

レート制限に関する考慮事項

Genesys Cloud は、Genesys Cloud API にレート制限を課します。Genesys Cloud API のレート制限に関する詳細は、Genesys Cloud Developer ウェブサイトの「Rate limits」を参照してください。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Genesys Cloud アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Genesys Cloudを使用して AppFabric を認可する必要があります。AppFabric Genesys Cloudで を承認するために必要な情報を見つけるには、次のステップを使用します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して Genesys Cloud と統合されます。Genesys Cloudで OAuth アプリケーションを作成するには、以下の手順に従います。

1. Genesys Cloud Resource Center ウェブサイトの「Create an OAuth Client」の手順に従います。

   [許可のタイプ] では [コード承認] を選択します。
2. 次の形式のリダイレクト URL を承認済みリダイレクト URL として使用します。

```
https://<region>.console.aws.amazon.com/appfabric/oauth2
```

この URL で、<#####>は、AppFabric アプリバンドルを構成した AWS リージョン のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは us-east-1 です。そのリージョンのリダイレクト URL は https://us-east-1.console.aws.amazon.com/appfabric/oauth2 です。

3. [スコープ] ボックスを選択すると、アプリケーションで使用できるスコープのリストが表示されます。scope audits:readonlyと を選択しますusers:readonly。スコープの詳細については、「Genesys Cloud Developer Center」の「OAuth Scopes」を参照してください。
4. [保存] をクリックします。Genesys Cloud に、クライアント ID とクライアントシークレット (トークン) が作成されます。

必要範囲

Genesys Cloud OAuth アプリケーションに次の範囲を入力する必要があります。

- audits:readonly
- users:readonly

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。AppFabric のテナント ID は Genesys Cloud インスタンス名です。ブラウザのアドレスバーにテナント ID が表示されます。例えば、usw2.pure.cloudは次のURLhttps://login.usw2.pure.cloudのテナントIDです。

テナント名

この一意の Genesys Cloud 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。Genesys Cloudでクライアント ID を検索するには以下の手順を使用してください。

1. [管理者] を選択します。

2. [統合] で [OAuth] を選択します。
3. OAuth クライアントを選択し、クライアント ID を取得します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。以下の手順でGenesys Cloudのクライアントシークレット を検索してください。

1. [管理者] を選択します。
2. [統合] で [OAuth] を選択します。
3. OAuth クライアントを選択し、クライアントシークレットを取得します。

AppFabric GitHub用に を設定する

GitHubは、Git を使用してソフトウェア開発とバージョン管理を行うためのプラットフォームおよびクラウドベースのサービスで、開発者はコードを保存および管理できます。Git の分散型バージョン管理に加えて、アクセス制御、バグトラッキング、ソフトウェア機能要求、タスク管理、継続的インテグレーション、すべてのプロジェクトの Wiki を提供します。

AWS AppFabric for security を使用すると、 から監査ログとユーザーデータを受信しGitHub、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- GitHub での AppFabric のサポート
- AppFabric を GitHub アカウントに接続する

GitHub での AppFabric のサポート

AppFabric は、GitHub からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して GitHub からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- 監査ログにアクセスするには、エンタープライズアカウントが必要です。

- エンタープライズ監査ログにアクセスするには、エンタープライズアカウントの管理者ロールが必要です。
- 組織から監査ログを取得するには、組織のオーナーである必要があります。

レート制限に関する考慮事項

GitHub は、GitHub API にレート制限を課します。GitHub API レート制限の詳細については、GitHubウェブサイトの「API リクエストの制限と割り当て」を参照してください。AppFabric と既存の GitHub API アプリケーションの組み合わせが GitHub's の制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を GitHub アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、GitHubを使用して AppFabric を認可する必要があります。AppFabric GitHubで を承認するために必要な情報を見つけるには、次のステップを使用します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して GitHub と統合されます。GitHub で OAuth アプリケーションを作成するときは、以下の手順に従います。この詳細については、GitHub ウェブサイトの「Creating GitHubs Apps」を参照してください。

1. ページの右上にある [プロフィール写真] を選択し、[設定] を選択します。
2. 左側のナビゲーションペインの [デベロッパー設定] を選択します。
3. 左のナビゲーションペインから、[OAuth アプリケーション] を選択します。
4. [新規 OAuth アプリケーション] を選択します。

> **Note**
>
> OAuth アプリをまだ作成していない場合、このボタンには [新規アプリケーションの登録] というラベルが表示されます。

5. [アプリケーション名] テキストボックスにアプリケーションの名前を入力します。
6. [ホームページ URL] テキストボックスに、アプリケーションインスタンスの完全な URL を入力します。
7. (オプション) [アプリケーションの説明] テキストボックスにアプリの説明を入力します。ユーザーにはこの説明が表示されます。
8. [承認コールバック URL] テキストボックスに、次の形式の URL を入力します。

```
https://<region>.console.aws.amazon.com/appfabric/oauth2
```

   この URL で、*<#####>* iは、AppFabric アプリバンドルを構成した AWS リージョン のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。そのリージョンのリダイレクト URL は `https://us-east-1.console.aws.amazon.com/appfabric/oauth2` です。
9. OAuth アプリがデバイスフローを使用してユーザーの識別と承認を行う場合は、[デバイスフローを有効にする] を選択します。デバイスフローについて詳しくは、GitHub ウェブサイトの「OAuth アプリの承認」を参照してください。
10. [アプリケーションの登録] を選択します。

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。テナント ID は以下のいずれかの形式で指定する必要があります。

エンタープライズ監査ログ:

エンタープライズアカウントが所有するすべての組織のアクションを集約して知りたい場合は、エンタープライズの監査ログを使用してください。

エンタープライズ監査ログを使用するには、テナント ID がアカウントのエンタープライズ ID です。ブラウザのアドレスバーにエンタープライズ ID が表示されます。

例えば、*exampleenterprise*は次のURLhttps://github.com/settings/enterprises/*examplenterprise*のエンタープライズIDです。

エンタープライズ監査ログのテナント ID を指定するときは、プレフィックスを付ける必要がありますenterprise:。そのため、前の例ではと指定しますenterprise:examplenterprise。

組織監査ログ:

組織のメンバーが実行したアクションを知りたい場合は、組織の監査ログを組織管理者として使用してください。アクションを実行したユーザー、アクション内容、実行日時などの詳細が含まれます。

組織の監査ログを使用するには、テナント ID が組織 ID です。ブラウザのアドレスバーに組織 ID が表示されます。例えば、*exampleorganization*は次のURLhttps://github.com/settings/organizations/*exampleorganization*の組織IDです。

組織監査ログのテナント ID を指定するときは、プレフィックスを付ける必要がありますorganization:。そのため、前の例ではorganization:exampleorganizationと指定します。

テナント名

この固有のGitHubエンタープライズまたは組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。GitHubでクライアント ID を検索するには以下の手順を使用してください。

1. ページの右上にある [プロフィール写真] を選択し、[設定] を選択します。
2. 左側のナビゲーションペインの [デベロッパー設定] を選択します。
3. 左のナビゲーションペインから、[OAuth アプリケーション] を選択します。
4. 特定の OAuth アプリを選択し、[クライアントID] の値を探します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。GitHub以下の手順でクライアントシークレット を検索してください。

1. ページの右上にある [プロフィール写真] を選択し、[設定] を選択します。

2. 左側のナビゲーションペインの [デベロッパー設定] を選択します。
3. 左のナビゲーションペインから、[OAuth アプリケーション] を選択します。
4. 特定の OAuth アプリを選択し、[クライアントシークレット] の値を探します。既存のクライアントシークレットが見つからない場合は、新しいクライアントシークレットを生成する必要がある場合があります。

認可を承認します

AppFabric でアプリ認可を作成すると、 から認可を承認GitHubするためのポップアップウィンドウが表示されます。AppFabric 認可を承認するには、許可を選択します。

OAuthアプリ へのアクセス制限が有効になっている場合は、組織が OAuth アプリケーションへのアクセスを許可していることを確認してください。

AppFabric Google Analytics用に を設定する

Google Analytics は、検索エンジンの最適化 (SEO) とマーケティングの目的で統計と基本的な分析ツールを提供するウェブ分析サービスです。 Google Analyticsは、ウェブサイトのパフォーマンスを追跡し、訪問者のインサイトを収集するために使用されます。これは、組織がユーザートラフィックのトップソースを決定し、マーケティング活動やキャンペーンの成功を測定し、目標の完了 (購入、カートへの製品の追加など) を追跡し、ユーザーエンゲージメントのパターンや傾向を検出し、人口統計などの他の訪問者情報を取得するのに役立ちます。小規模および中規模の小売ウェブサイトは、さまざまな顧客行動分析を取得および分析Google Analyticsするために を使用することがよくあります。これは、マーケティングキャンペーンの改善、ウェブサイトトラフィックの促進、訪問者の保持の向上に使用できます。

AWS AppFabric for security を使用すると、 からログとユーザーデータを監査しAzure Monitor、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Google Analytics での AppFabric のサポート
- AppFabric を Google Analytics アカウントに接続する

Google Analytics での AppFabric のサポート

AppFabric は、Google Analytics からの監査ログの受信をサポートしています。

前提条件

AppFabric を使用して Google Analytics からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- Google Analytics アカウントの管理者である必要があります。
- AppFabric がログを配信するには、Google Cloudプロジェクトで Google Analytics Admin API を有効にする必要があります。OAuth Google Analytics アプリケーションを設定するときは、必ず新しいプロジェクトを使用してください。

レート制限に関する考慮事項

Google Analytics は、Google Analytics API にレート制限を課します。Google Analytics API レート制限の詳細については、Google Analytics ウェブサイトの「制限とクォータ」を参照してください。AppFabric と既存の Google Analytics API アプリケーションの組み合わせが制限を超えると、AppFabric に表示される監査ログが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Google Analytics アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Google Analyticsを使用して AppFabric を認可する必要があります。AppFabric Google Analyticsで を承認するために必要な情報を見つけるには、次の手順を実行します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して Google Analytics と統合されます。で OAuth アプリケーションを作成するには、次の手順を実行しますGoogle Analytics。

1. OAuth 同意画面を設定するには、Google ウェブサイトの Google デベロッパーガイドのOAuth 同意画面を設定する」の手順に従います。
2. ユーザータイプの外部を選択する
3. AppFabric の OAuth 認証情報を設定するには、Google デベロッパーガイドの「アクセス認証情報の作成」ページのOAuth クライアント ID 認証情報」セクションの手順に従います。

4. 次の形式のリダイレクト URL を使用します。

```
https://<region>.console.aws.amazon.com/appfabric/oauth2
```

このアドレス AWS リージョン の `<region>`は、AppFabric アプリバンドルを設定した のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。そのリージョンのリダイレクト URL は `https://us-east-1.console.aws.amazon.com/appfabric/oauth2` です。

必要範囲

OAuth Google Analytics アプリケーションには、次のスコープを追加する必要があります。

```
https://www.googleapis.com/auth/analytics.edit
```

アプリ権限

テナント ID

AppFabric はテナント ID をリクエストします。AppFabric のテナント ID は、Google Analytics アカウント ID です。

1. Google Analytics ホームページに移動します。
2. ナビゲーションペインで管理者を選択します。
3. アカウント ID は、Account > Account Settings > Account details > Account ID にあります。

テナント名

この一意の Google Analytics 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。でクライアント ID を検索するには、次の手順に従いますGoogle Analytics。

1. 認証情報ページに移動します。
2. OAuth 2.0 クライアント IDs セクションで、作成したクライアント ID を選択します。
3. クライアント ID は、ページの追加情報セクションに一覧表示されます。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。でクライアントシークレットを検索するには、次の手順に従いますGoogle Analytics。

1. 認証情報ページに移動します。
2. OAuth 2.0 クライアント IDs セクションで、クライアント名を選択します。
3. クライアントシークレットは、ページのクライアントシークレットセクションに一覧表示されます。

アプリ認可

AppFabric でアプリ認可を作成すると、 から認可を承認Google Analyticsするためのポップアップウィンドウが表示されます。許可を選択して AppFabric 認可を承認するには。

AppFabric Google Workspace用に を設定する

Google Workspace は Google が開発、販売しているクラウドコンピューティング、生産性向上ツール、コラボレーションツール、ソフトウェア、製品のコレクションです。

AWS AppFabric for security を使用すると、 からログとユーザーデータを監査しGoogle Workspace、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Google Workspace での AppFabric のサポート
- AppFabric を Google Workspace アカウントに接続する

Google Workspace での AppFabric のサポート

AppFabric は、Google Workspace からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Google Workspace からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- Google Workspace エンタープライズスタンダードプランへの加入が必要です。Google Workspace エンタープライズスタンダードプランの作成またはアップグレードの詳細については、「Google Workspace プラン」ウェブサイトを参照してください。
- Google Workspace には管理者ロールを持つユーザーが必要です。
- AppFabric がログを配信できるようにするには、Google クラウドプロジェクトで Google 管理者 SDK API を有効にする必要があります。詳細については、「Google Workspaceデベロッパーガイド 」の「Google Workspace API の有効化」を参照してください。

レート制限に関する考慮事項

Google Workspace は、Google Workspace API にレート制限を課します。Google Workspace API レート制限の詳細については、Google Workspace ウェブサイトに掲載されている「Google Workspace 管理者ガイド」の「制限とクォータ」を参照してください。AppFabric と既存の Google Workspace API アプリケーションの組み合わせが制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

ほとんどの監査イベントには最大 30 分の遅延、特定の監査イベントが宛先に配信されるまでには最大 4 時間の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。詳細については、Google WorkSpace 管理者ヘルプウェブサイトの「データ保持とラグタイム」を参照してください。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、 にお問い合わせくださいサポート。

AppFabric を Google Workspace アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Google Workspaceを使用して AppFabric を認可する必要があります。AppFabric Google Workspaceで を承認するために必要な情報を見つけるには、次のステップを使用します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して Google Workspace と統合されます。Google Workspace で OAuth アプリケーションを作成するには、以下の手順に従います。

1. OAuth 同意画面を設定するには、Google Workspace ウェブサイトに掲載されている「Google Workspace デベロッパーガイド」の「OAuth 同意画面の設定」の指示に従ってください。

   [ユーザータイプ] に [内部] を選択します。

2. AppFabric の OAuth 認証情報を設定するには、「Google Workspace デベロッパーガイド」の「アクセス認証情報の作成」ページにある「OAuth クライアント ID 認証情報」セクションの手順に従います。
3. 次の形式のリダイレクト URL を使用します。

```
https://<region>.console.aws.amazon.com/appfabric/oauth2
```

この URL AWS リージョン では、 `<region>`は AppFabric アプリバンドルを設定した のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。そのリージョンのリダイレクト URL は `https://us-east-1.console.aws.amazon.com/appfabric/oauth2` です。

必要範囲

Google Workspace OAuth アプリケーションに次の範囲を入力する必要があります。

- `https://www.googleapis.com/auth/admin.reports.audit.readonly`
- `https://www.googleapis.com/auth/admin.directory.user`

これらの範囲が表示されない場合は、Admin SDK API を Google クラウド API ライブラリに追加してください。

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。AppFabric のテナント ID は Google Workspace プロジェクトIDです。プロジェクト ID を確認するには、Google API コンソールヘルプウェブサイトの「プロジェクト ID の検索」を参照してください。

テナント名

この一意の Google Workspace を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。クライアント ID を確認するには、以下のステップに従います。

1. 「Google Workspace デベロッパーガイド」の「認証情報の管理」ページにある「認証情報の表示」セクションの情報を使用してクライアント ID を検索します。
2. OAuth クライアントのクライアント ID を AppFabric の [クライアント ID] フィールドに入力します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。クライアントシークレットを確認するには、以下の手順に従います。

1. 「Google Workspace デベロッパーガイド」の「認証情報の管理」ページにある「認証情報の表示」セクションの情報を使用してクライアントシークレットを検索します。
2. クライアントシークレットをリセットする必要がある場合は、「Google Workspace デベロッパーガイド」の「認証情報の管理」ページにある「クライアントシークレットのリセット」セクションの手順に従ってください。
3. AppFabric のクライアントシークレットを [クライアントシークレット] フィールドに入力します。

認可を承認します

AppFabric でアプリ認可を作成すると、 から認可を承認Google Workspaceするためのポップアップウィンドウが表示されます。AppFabric 認可を承認するには、許可を選択します。

AppFabric HubSpot用に を設定する

HubSpot は、マーケティング、営業、コンテンツ管理、カスタマーサービスをつなげるために必要な、ソフトウェア、統合、リソースのすべてを備える顧客プラットフォームです。HubSpot のコネクテッドプラットフォームを使うことで、ユーザーは最も重要なこと、つまり顧客、に焦点を当てることにより、ビジネスをより早く成長させることができます。

AWS AppFabric for security を使用すると、 から監査ログとユーザーデータを受信しHubSpot、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- HubSpot での AppFabric のサポート
- AppFabric を HubSpot アカウントに接続する

HubSpot での AppFabric のサポート

AppFabric は、HubSpot からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して HubSpot からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- 監査ログにアクセスするには、HubSpot の Enterprise サブスクリプションのアカウントが必要になります。HubSpot サブスクリプションに関する詳細は、HubSpot のナレッジベースの「Manage your HubSpot subscription」を参照してください。
- デベロッパーアカウントと、そのアカウントに関連付けられたアプリケーションが必要になります。
- アプリケーションを HubSpot アカウントにインストールするには、スーパー管理者であるか、または、App Marketplace Access のアクセス権限と、アプリケーションが要求するスコープを受け入れるためのユーザー権限を持っている必要があります。

レート制限に関する考慮事項

HubSpot は、HubSpot API にレート制限を課します。HubSpot API のレート制限 (OAuth を使用するアプリケーションの制限を含む) に関する詳細は、HubSpot ウェブサイトの「Rate limits」を参照してください。AppFabric と既存の HubSpot API アプリケーションの組み合わせが HubSpot の制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を HubSpot アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、HubSpotを使用して AppFabric を認可する必要があります。AppFabric HubSpotで を承認するために必要な情報を見つけるには、次のステップを使用します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して HubSpot と統合されます。HubSpotで OAuth アプリケーションを作成するには、以下の手順に従います。

1. HubSpot ウェブサイトの「HubSpot ガイド」にある「 Create a public app」のセクションの指示に従います。
2. [認証] タブから、必要範囲 に記載されている 3 つのスコープを追加します。
3. [リダイレクト URL] で、以下の形式のリダイレクト URL を使用します。

   ```
   https://<region>.console.aws.amazon.com/appfabric/oauth2
   ```

   この URL で、`<#####>`は、AppFabric アプリバンドルを構成した AWS リージョン のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。そのリージョンのリダイレクト URL は `https://us-east-1.console.aws.amazon.com/appfabric/oauth2` です。
4. [アプリの作成] を選択します。

必要範囲

HubSpot OAuth アプリケーションに次の範囲を入力する必要があります。

- `settings.users.read`
- `crm.objects.owners.read`
- `account-info.security.read`

アプリ権限

テナント ID

この一意の HubSpot 組織を識別する ID を入力します。例えば、HubSpot アカウント ID などです。

テナント名

この一意の HubSpot 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。HubSpotでクライアント ID を検索するには以下の手順を使用してください。

1. HubSpot ログインページに進み、デベロッパーアカウントの認証情報を使用してサインインします。
2. [アプリケーション] メニューで自分のアプリケーションを選択します。
3. [認証] タブで、クライアント ID の値を探します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。以下の手順でHubSpotのクライアント シークレット を検索してください。

1. HubSpot ログインページに進み、デベロッパーアカウントの認証情報を使用してサインインします。
2. [アプリケーション] メニューで自分のアプリケーションを選択します。
3. [認証] タブで、クライアントシークレットの値を探します。

認可を承認します

AppFabric でアプリ認可を作成すると、 から認可を承認HubSpotするためのポップアップウィンドウが表示されます。(デベロッパーアカウントではなく) エンタープライズアカウントの認証情報を使用してアカウントにサインインし、AppFabric 認可を承認します。[許可] を選択します。

AppFabric IBM Security® Verify用に を設定する

IBM Security® Verify ファミリーは、アイデンティティガバナンスの管理、ワークフォースとコンシューマーのアイデンティティとアクセスの管理、特権アカウントの制御のための、自動化されたクラウドベースのオンプレミス機能を提供します。クラウドまたはオンプレミスのソリューションをデプロイする必要があるかどうかにかかわらず、 は信頼を確立し、ワークフォースとコンシューマーの両方に対する内部脅威から保護するIBM Security® Verifyのに役立ちます。

AWS AppFabric for security を使用すると、 から監査ログとユーザーデータを受信しIBM Security® Verify、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- IBM Security® Verify での AppFabric のサポート
- AppFabric を IBM Security® Verify アカウントに接続する

IBM Security® Verify での AppFabric のサポート

AppFabric は、IBM Security® Verify からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して IBM Security® Verify からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- 監査ログにアクセスするには、IBM Security® VerifySaaS アカウントが必要です。
- 監査ログにアクセスするには、SaaS IBM Security® Verify アカウントに管理者ロールが必要です。

レート制限に関する考慮事項

IBM Security® Verify は、IBM Security® Verify API にレート制限を課します。IBM Security® Verify API レート制限の詳細については、「IBM 規約」を参照してください。AppFabric と既存の IBM Security® Verify API アプリケーションの組み合わせがIBM Security® Verify制限を超えると、AppFabric に表示される監査ログが遅延する可能性があります。

データ遅延に関する考慮事項

監査イベントでは、送信先に配信されるまでに最大 30 分の遅延が発生することがあります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を IBM Security® Verify アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、IBM Security® Verifyを使用して AppFabric を認可する必要があります。AppFabric IBM Security® Verifyで を認可するために必要な情報を確認するには、次の手順を実行します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して IBM Security® Verify と統合されます。で OAuth アプリケーションを作成するにはIBM Security® Verify、IBM ドキュメントウェブサイトの「API クライアントの作成」を参照してください。

1. 初回ログインには、登録済みの E メールアドレスに送信されたログイン URL と認証情報を使用します。
2. で管理コンソールにアクセスしますhttps://<hostname>.verify.ibm.com/ui/admin/。詳細については、「IBM Security® Verify へのアクセス」を参照してください。
3. 管理コンソールの Security < API Access < API Client で、Add を選択します。
4. 次のオプションを選択します。これらは、監査ログとユーザーの詳細を読み取るために必要です。
   - レポートの読み取り
   - ユーザーおよびグループの読み取り
5. クライアント認証メソッドのデフォルトオプションのままにします。

   カスタムスコープフィールドを編集しないでください。
6. [次へ] を選択します。
7. IP フィルターフィールドを編集しないでください。
8. [次へ] を選択します。
9. 追加プロパティフィールドを編集しないでください。
10. [次へ] を選択します。
11. 名前と説明を指定します。説明はオプションです。
12. API クライアントの作成を選択します。

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。テナント ID はIBM Security® Verify標準 URL にあります。たとえば、https://hostname.verify.ibm.com/URL では、テナント ID は より前 .verify.ibm.com (以前の####を使用している場合ice.ibmcloud.comは より前) にあるホスト名です。バニティ URL を使用している場合は、IBM Security® Verifyサポートチームに連絡して標準 URL を取得してください。

テナント名

この一意のIBM Security® Verifyテナントを識別する名前を入力します。AppFabric はテナント名を使用して、アプリケーション認可とアプリケーション認可から作成された取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。IBM Security® Verifyでクライアント ID を検索するには以下の手順を使用してください。

1. 初回ログインには、登録済みの E メールアドレスに送信されたログイン URL と認証情報を使用します。
2. で管理コンソールにアクセスします`https://<hostname>.verify.ibm.com/ui/admin/`。詳細については、「IBM Security® Verify へのアクセス」を参照してください。
3. 管理コンソールの Security < API Access < API Client で、特定の OAuth アプリの横にある省略記号 (⋮) を選択します。
4. 接続の詳細を選択します。
5. API 認証情報でクライアント ID を見つけます。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。以下の手順でIBM Security® Verifyのクライアント シークレット を検索してください。

1. 初回ログインには、登録済みの E メールアドレスに送信されたログイン URL と認証情報を使用します。
2. で管理コンソールにアクセスします`https://<hostname>.verify.ibm.com/ui/admin/`。詳細については、「IBM Security® Verify へのアクセス」を参照してください。
3. 管理コンソールの Security < API Access < API Client で、特定の OAuth アプリの横にある省略記号 (⋮) を選択します。
4. 接続の詳細を選択します。
5. API 認証情報でクライアントシークレットを見つけます。

AppFabric JumpCloud用に を設定する

JumpCloud Inc. は、アイデンティティ管理用のクラウドベースのディレクトリプラットフォームを提供する米国のエンタープライズソフトウェア会社です。ID 管理を一元化して簡素化することで、ユーザーはプラットフォーム、プロトコル、プロバイダー、場所に関係なく、単一の認証情報セットでシステム、アプリケーション、ネットワーク、ファイルサーバーに安全にアクセスできます。

AWS AppFabric を使用すると、JumpCloud から監査ログとユーザーデータを受信し、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- JumpCloud での AppFabric のサポート
- AppFabric を JumpCloud アカウントに接続する

JumpCloud での AppFabric のサポート

AppFabric は、JumpCloud からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して JumpCloud からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- 有効な有料JumpCloudサブスクリプションプランが必要です。詳細については、 JumpCloudウェブサイトのSelect a package that's right for you「」を参照してください。
- 「請求のある管理者」ロールが必要です。

レート制限に関する考慮事項

JumpCloud はレート制限を公開していません。サポートケースを作成するか、JumpCloudカスタマーチームに連絡する必要があります。AppFabric と既存の JumpCloud API アプリケーションの組み合わせがJumpCloud's制限を超えると、AppFabric に表示される監査ログが遅延する可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションによって利用可能になった監査イベントの遅延と、データ損失を減らすために

講じられた予防策によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を JumpCloud アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、JumpCloudを使用して AppFabric を認可する必要があります。AppFabric JumpCloudで を承認するために必要な情報を確認するには、次のセクションのステップに従います。

JumpCloud アカウントから Organization トークンを作成する

AppFabric は API JumpCloud キーを使用して と統合します。JumpCloud で API キーを作成するには、次の手順に従います。

1. 管理者としてアカウントにサインインしますJumpCloud。
2. 管理者ポータルで、右上にあるアカウントのイニシャルを選択し、メニューから My API Key を選択します。
3. 新しい API キーの生成を選択するか、既存のキーを選択します。

> **Note**
>
> JumpCloud は 1 つのアクティブな API キーのみを許可します。新しい API キーを生成すると、現在の API キーへのアクセスが取り消されます。これにより、前の API キーを使用するすべての呼び出しにアクセスできなくなります。以前の API キーを使用する既存の統合は、新しいキー値で更新する必要があります。

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。ここで「組織 ID」はテナント ID になります。「組織 ID」を検索するには、次の手順に従います。

1. JumpCloud アカウントにサインインします。
2. ナビゲーションペインで、設定、組織プロファイル、全般を選択します。
3. 「目の」アイコンを選択して、不明瞭なビューを削除します。
4. ID をコピーするには、「ダブルページ」アイコンを選択します。

テナント名

この一意の JumpCloud 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

サービスアカウントトークン

AppFabric は、ユーザーのサービスアカウントトークンをリクエストします。AppFabric では、これはこのトピックの前半で で作成した組織 API JumpCloud アカウントから Organization トークンを作成するトークンです。

AppFabric Microsoft 用に 365 を設定する

Microsoft 365 は、Microsoft が所有する生産性向上ソフトウェア、コラボレーション、クラウドベースサービスの製品ファミリーです。

AWS AppFabric for security を使用すると、365 Microsoft のログとユーザーデータを監査し、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Microsoft 365 での AppFabric のサポート
- AppFabric を Microsoft 365 アカウントに接続する

Microsoft 365 での AppFabric のサポート

AppFabric は、Microsoft 365 からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Microsoft 365 からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- Microsoft365 エンタープライズプランのサブスクリプションが必要です。Microsoft 365 エンタープライズプランの作成またはアップグレードについての詳細は、Microsoft ウェブサイトの「Microsoft365 エンタープライズプラン」を参照してください。
- 管理者権限を持つユーザーが含まれている Microsoft 365 アカウントが必要です。
- 組織の監査ログを有効にする必要があります。詳細については、Microsoft ウェブサイトの「監査のオンとオフを切り替える」を参照してください。

レート制限に関する考慮事項

Microsoft 365 は、Microsoft 365 API にレート制限を課しています。Microsoft 365 API のレート制限の詳細については、Microsoft ウェブサイトの Microsoft Graph 文書の「Microsoft Graph サービス固有のスロットリング制限」を参照してください。AppFabric と既存の Microsoft 365 API アプリケーションの組み合わせが制限を超えると、AppFabric での監査ログの表示が遅延する可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Microsoft 365 アカウントに接続する

AppFabric サービス内で App バンドルを作成したら、365 で AppFabric Microsoft を承認する必要があります。AppFabric で Microsoft 365 を認可するために必要な情報を確認するには、次の手順を実行します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用してMicrosoft 365 と統合されます。Microsoft 365 で OAuth アプリケーションを作成するには、以下の手順に従います。

1. Microsoft ウェブサイトに掲載されている「Azure Active Directory 開発者ガイド」の「アプリケーションの登録」セクションの指示に従ってください。

   [サポートされているアカウントタイプ] の設定では、[この組織ディレクトリのアカウントのみ] を選択します。

2. 「Azure Active Directory 開発者ガイド」の「リダイレクト URI の追加」セクションの指示に従ってください。

   [ウェブプラットフォーム] を選択します。

   ```
   https://<region>.console.aws.amazon.com/appfabric/oauth2
   ```

   この URL AWS リージョン では、 *<region>*は AppFabric アプリバンドルを設定した のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。その

リージョンのリダイレクト URL は `https://us-east-1.console.aws.amazon.com/appfabric/oauth2` です。

ウェブプラットフォームの他の入力フィールドはスキップできます。

3. 「Azure Active Directory 開発者ガイド」の「クライアントシークレットの追加」セクションの指示に従ってください。

必要なアクセス許可

OAuth アプリケーションには以下のアクセス許可を追加する必要があります。許可を追加するには、「Azure Active Directory 開発者ガイド」の「ウェブ API にアクセスするためのアクセス許可を追加する」セクションの指示に従ってください。

- `Microsoft Graph API` > `User.Read` (自動的に追加されます)
- `Office 365 Management APIs` > `ActivityFeed.Read` (委任タイプを選択)
- `Office 365 Management APIs` > `ActivityFeed.ReadDlp` (委任タイプを選択)
- `Office 365 Management APIs` > `ServiceHealth.Read` (委任タイプを選択)

アクセス許可の追加後にその許可に対する管理者の同意を付与するには、「Azure Active Directory 開発者ガイド」の「管理者同意ボタン」セクションの指示に従ってください。

アプリ権限

AppFabric は、Microsoft 365 アカウントからのユーザー情報と監査ログの受信をサポートします。365 から監査ログとユーザーデータの両方を受信するには、2 Microsoft つのアプリ認可を作成する必要があります。1 つはアプリ認可ドロップダウンリストから Microsoft 365 という名前のもの、もう 1 つはアプリ認可ドロップダウンリストから Microsoft 365 監査ログという名前のものです。両方のアプリ認証には、同じテナント ID、クライアント ID、およびクライアントシークレットを使用できます。Microsoft 365 から監査ログを受信するには、[Microsoft 365] および [Microsoft 365 監査ログ] の両方のアプリ認証が必要です。ユーザーアクセスツールを単独で使用するには、Microsoft365 アプリケーション認可のみが必要です。

テナント ID

AppFabric はテナント ID を要求します。AppFabric のテナント ID は、Azure Active Directory のテナント ID です。Azure Active Directory のテナント ID を確認するには、Microsoft ウェブサイトの「Azure 製品ドキュメント」の「Azure Active Directory テナント ID を確認する方法」を参照してください。

テナント名

この一意の Microsoft 365 アカウントを識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。AppFabric のクライアント ID は、Microsoft 365 アプリケーション (クライアント) ID です。Microsoft 365 アプリケーション (クライアント) ID を確認するには、以下の手順に従います。

1. AppFabric で使用する OAuth アプリケーションの概要ページを開きます。
2. アプリケーション (クライアント) ID が [Essentials] の下に表示されます。
3. OAuth クライアントのアプリケーション (クライアント) ID を AppFabric の [クライアント ID] フィールドに入力します。

クライアントシークレット

AppFabric は、クライアントのシークレットを要求します。Microsoft 365 がこの値を表示するのは、OAuth アプリケーションのクライアントシークレットを最初に作成したときだけです。まだ行っていない場合に新しいクライアントシークレットを生成するには、以下の手順に従います。

1. クライアントシークレットを作成するには、「Azure Active Directory 開発者ガイド」の「クライアントシークレットの追加」セクションの指示に従ってください。
2. [値] フィールドの内容を AppFabric の [クライアントシークレット] フィールドに入力します。

認可を承認します

AppFabric でアプリ認可を作成すると、認可を承認するためのポップアップウィンドウが 365 Microsoft から表示されます。AppFabric 認可を承認するには、許可を選択します。

AppFabric Miro用に を設定する

Miroは、あらゆる規模の分散型チームが次の大きなものを構築できるようにする、イノベーションのためのオンラインワークスペースです。プラットフォームの無限のキャンバスにより、チームは魅力的なワークショップや会議を開催したり、製品をデザインしたり、アイデアをブレインストーミングしたりすることができます。 Miroサンフランシスコとアムステルダムに共同本社を置き、Fortune 100 企業の 99% を含め、世界中で 5,000 万人以上のユーザーにサービスを提供してい

ます。 Miro2011 年に設立され、現在、世界 12 の拠点に 1,500 人以上の従業員を擁しています。詳細については、「Miro」を参照してください。

Miroダイアグラム作成、ワイヤーフレーミング、リアルタイムのデータ視覚化、ワークショップの円滑化、アジャイルプラクティス、ワークショップ、インタラクティブなプレゼンテーションの組み込みサポートなど、イノベーションのために設計されたコラボレーション機能がすべて含まれています。 Miro最近、Miro AI を活用したマッピングとダイアグラム作成、Miroクラスタリングと要約、コンテンツ生成といった機能を拡張する AI が発表されました。 Miro組織はスタンドアロンツールの数を減らし、情報の断片化とコスト削減を可能にします。

AWS AppFabric for security を使用すると、 からログとユーザーデータを監査しMiro、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Miro での AppFabric のサポート
- AppFabric を Miro アカウントに接続する

Miro での AppFabric のサポート

AppFabric は、Miro からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Miro からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- Miro Enterprise プランが必要です。Miro プランタイプの詳細については、Web Miroサイトの料金ページを参照してください。Miro
- Miroアカウントには会社管理者ロールを持つユーザーが必要です。ロールについて詳しくは、Miro ヘルプセンター Web サイトの「Miro のロール」の「会社レベル」セクションを参照してください。
- Miroアカウントには Enterprise Developer チームが必要です。開発者チームの作成について詳しくは、Miro Help Center Web サイトの「エンタープライズ開発者チーム」を参照してください。

レート制限に関する考慮事項

Miro は、Miro API にレート制限を課します。Miro API のレート制限について詳しくは、Miro Web サイトの「Miro 開発者ガイド」の「レート制限」を参照してください。AppFabric と既存の Miro API アプリケーションの組み合わせが制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Miro アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Miroを使用して AppFabric を認可する必要があります。AppFabric Miroで を認可するために必要な情報を確認するには、次の手順を使用します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して Miro と統合されます。Miroで OAuth アプリケーションを作成するには、以下の手順に従います。

1. OAuth アプリケーションを作成するには、Miro Help Center Web サイトのエンタープライズデベロッパーチームの記事の「アプリの作成とインストール」セクションの指示に従ってください。
2. アプリケーション作成ダイアログで、エンタープライズ組織の開発者チームを選択した後、ユーザー認可トークンの有効期限チェックボックスをオンにします。

   > **Note**
   > このオプションはアプリの作成後に変更できないため、アプリを作成する前に行う必要があります。

3. アプリページの [OAuth 2.0 用リダイレクト URI] セクションに次の形式の URL を入力します。

```
https://<region>.console.aws.amazon.com/appfabric/oauth2
```

この URL で、*<#####>*は、AppFabric アプリバンドルを構成した AWS リージョン のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。そのリージョンのリダイレクト URL は `https://us-east-1.console.aws.amazon.com/appfabric/oauth2` です。

4. AppFabric アプリ認可に使用するクライアント ID とクライアントシークレットをコピーして保存します。

必要範囲

Miro OAuth `Permissions` アプリページのセクションに次のスコープを追加する必要があります。

- `auditlogs:read`
- `organizations:read`

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。AppFabric のテナント ID はMiroチームIDです。Miro チーム ID の確認方法については、「よくある質問」セクションの「新しく Miro の管理者になりました。」を参照してください。Miro ヘルプセンターの Web サイトのどこから始めればいいですか?

テナント名

この一意の Miro 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。クライアント ID を確認するには、以下のステップに従います。

1. Miroプロファイル設定に移動します。
2. [マイアプリ] タブを選択します。
3. AppFabric との接続に使用するアプリを選択します。
4. [アプリ認証情報] セクションのクライアント ID を AppFabric の [クライアント ID] フィールドに入力します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。クライアントシークレットを確認するには、以下の手順に従います。

1. Miroプロファイル設定に移動します。
2. [マイアプリ] タブを選択します。
3. AppFabric との接続に使用するアプリを選択します。
4. [アプリ認証情報] セクションのクライアントシークレットを AppFabric の [クライアントシークレット] フィールドに入力します。

認可を承認します

AppFabric でアプリ認可を作成すると、 から認可を承認Miroするためのポップアップウィンドウが表示されます。AppFabric 認可を承認するには、許可を選択します。

AppFabric Okta用に を設定する

Oktaは世界のアイデンティティ企業です。Oktaは、独立系の主要なアイデンティティパートナーとして、誰もがどこでも、あらゆるデバイスやアプリであらゆるテクノロジーを安全に使用できるようにします。最も信頼されているブランドは、安全なアクセス、認証、自動化を実現するOktaを信頼しています。Oktaワークフォース ID クラウドとカスタマー ID クラウドの中核をなす柔軟性と中立性により、ビジネスリーダーや開発者は、カスタマイズ可能なソリューションと 7,000 を超える事前構築済みの統合により、イノベーションに注力し、デジタルトランスフォーメーションを加速できます。 Oktaはアイデンティティが自分のものである世界を構築しています。詳細については、okta.com.を参照してください。

AWS AppFabric for security を使用すると、 からログとユーザーデータを監査しOkta、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Okta での AppFabric のサポート
- AppFabric を Okta アカウントに接続する

Okta での AppFabric のサポート

AppFabric は、Okta からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Okta からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- AppFabric Okta はどのプランタイプでも使用できます。
- Oktaアカウントにはスーパー管理者ロールを持つユーザーが必要です。
- AppFabric でアプリ認可を承認するユーザーには、Oktaアカウントでスーパー管理者ロールも必要です。

レート制限に関する考慮事項

Okta は、Okta API にレート制限を課します。Okta API のレート制限について詳しくは、Okta Web サイトの「Okta 開発者ガイド」の「レート制限」を参照してください。AppFabric と既存の Okta API アプリケーションの組み合わせが Okta の制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Okta アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Oktaを使用して AppFabric を認可する必要があります。AppFabric Oktaで を承認するために必要な情報を見つけるには、次のステップを使用します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して Okta と統合されます。AppFabric に接続する OAuth アプリケーションを作成するには、Oktaヘルプセンター Web サイトの「OIDC アプリインテグレーションの作成」の指示に従ってください。この構成には、AppFabricの考慮事項に注意してください。

1. [アプリケーションタイプ] には、[Webアプリケーション] を選択します。
2. Grant タイプで、認可コードと更新トークンを選択します。
3. [サインインリダイレクト URI] と [サインアウトリダイレクト URI] には、次の形式のリダイレクト URL を使用します。

```
https://<region>.console.aws.amazon.com/appfabric/oauth2
```

この URL で、`<#####>`は、AppFabric アプリバンドルを構成した AWS リージョン のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。そのリージョンのリダイレクト URL は `https://us-east-1.console.aws.amazon.com/appfabric/oauth2` です。

4. [信頼できるオリジン] の設定は省略できます。
5. [制限付きアクセス設定] で、Okta 組織内の全員にアクセス権を付与します。

> **Note**
>
> OAuth アプリケーションの初回作成時にこのステップを省略しても、アプリケーション設定ページの [割り当て] タブを使用して、組織内の全員をグループとして割り当てることができます。

6. その他のオプションはすべて、デフォルト値のままにしておくことができます。

必要範囲

Okta OAuth アプリケーションに次の範囲を入力する必要があります。

- `okta.logs.read`
- `okta.users.read`

アプリ権限

テナント ID

AppFabric はテナント ID をリクエストします。AppFabric のテナント ID はOktaドメインです。Okta ドメインの検索について詳しくは、Okta ウェブサイトの「Okta 開発者ガイド」の「Okta ドメインを探す」を参照してください。

テナント名

この一意の Okta 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。Oktaでクライアント ID を検索するには以下の手順を使用してください。

1. Okta開発者コンソールに移動します。
2. [アプリケーション] タブを選択します。
3. アプリケーションを選択し、[一般] タブを選択します。
4. [クライアント認証情報] セクションまでスクロールします。
5. AppFabric の [クライアント ID] フィールドに OAuth クライアントのクライアント ID を入力します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。以下の手順でOktaのクライアント シークレット を検索してください。

1. Okta開発者コンソールに移動します。
2. [アプリケーション] タブを選択します。
3. アプリケーションを選択し、[一般] タブを選択します。
4. [クライアント認証情報] セクションまでスクロールします。
5. OAuth アプリケーションのクライアントシークレットを AppFabric の [クライアントシークレット] フィールドに入力します。

認可を承認します

AppFabric でアプリ認可を作成すると、 から認可を承認Oktaするためのポップアップウィンドウが表示されます。AppFabric 認可を承認するには、許可を選択します。Okta 認可を承認するユーザーには、 でスーパー管理者権限が必要ですOkta。

AppFabric OneLogin by One Identity用に を設定する

OneLogin by One Identity は、従業員、顧客、パートナーのすべてのデジタル ID をシームレスに管理する、最新のクラウドベースのアクセス管理ソリューションです。OneLogin は、安全なシングルサインオン (SSO)、多要素認証 (MFA)、適応型認証、デスクトップレベルの MFA、AD や LDAP、G

Suite その他外部ディレクトリとの統合、ID ライフサイクル管理など、さまざまな機能を備えています。を使用するとOneLogin、最も一般的な攻撃から組織を保護できるため、セキュリティの向上、ユーザーエクスペリエンスの摩擦がなく、規制要件への準拠につながります。

AWS AppFabric for security を使用すると、 から監査ログとユーザーデータを受信しOneLogin、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- OneLogin by One Identity での AppFabric のサポート
- AppFabric を OneLogin by One Identity アカウントに接続する

OneLogin by One Identity での AppFabric のサポート

AppFabric は、OneLogin by One Identity からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して OneLogin by One Identity からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- OneLogin の Advanced または Professional のアカウントが必要です。
- 管理者/委任管理者の権限を持つユーザーが必要です。

レート制限に関する考慮事項

OneLogin by One Identity は、OneLogin API にレート制限を課します。OneLogin API レート制限の詳細については、「OneLogin API Reference」の「Get Rate Limit」を参照してください。AppFabric と既存の OneLogin API アプリケーションの組み合わせが OneLogin の制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。ただし、OneLogin レート制限は増やすことができます。サポートが必要な場合は OneLogin by One Identity アカウントマネージャー、または One Identity にお問い合わせください。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によ

るものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を OneLogin by One Identity アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、OneLogin by One Identityを使用して AppFabric を認可する必要があります。AppFabric OneLoginで認可するために必要な情報を見つけるには、次のステップを使用します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して OneLogin by One Identity と統合されます。OneLoginで OAuth アプリケーションを作成するには、以下の手順に従います。

1. OneLogin のログインページに進み、サインインします。
2. [デベロッパー] メニューから [API 認証情報] を選択します。
3. [新しい認証情報] を選択し、新しい認証情報の名前を入力して、[すべて読み取る] を選択します。
4. [保存] をクリックします。OneLogin に、クライアント ID とクライアントシークレットが作成されます。

必要範囲

OneLogin by One Identity OAuth アプリケーションに次の範囲を入力する必要があります。

- すべて読み取ります。スコープとクライアント認証情報の詳細については、「OneLogin API Reference」の「Working with API Credentials」を参照してください。

アプリ権限

テナント ID

AppFabric はテナント ID をリクエストします。AppFabric のテナント ID は [インスタンスサブドメイン] です。ブラウザのアドレスバーにテナント ID が表示されます。例えば、subdomainは次のURLhttps://*subdomain*.onelogin.comのテナントIDです。

テナント名

この一意の OneLogin by One Identity 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。OneLogin by One Identityでクライアント ID を検索するには以下の手順を使用してください。

1. OneLogin のログインページに進み、サインインします。
2. [デベロッパー] メニューから [API 認証情報] を選択します。
3. API 認証情報を選択してクライアント ID を取得します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。以下の手順でOneLogin by One Identityのクライアント シークレット を検索してください。

1. OneLogin のログインページに進み、サインインします。
2. [デベロッパー] メニューから [API 認証情報] を選択します。
3. API 認証情報を選択してクライアントシークレットを取得します。

クライアントアプリケーションの認可

AppFabric で、テナント ID と名前、クライアント ID と名前を使用してアプリケーションの認可を作成します。[接続] を選択して認可を有効にします。

AppFabric PagerDuty用に を設定する

PagerDuty は、兆候を発見したらすぐ行動に移すことで、顧客に影響がおよぶ問題を最小限に抑え、すばやい問題解決と業務効率の向上につなげる、デジタルオペレーション管理プラットフォームです。CloudWatch、GuardDuty、CloudTrail、Personal Health Dashboard と統合します。

AWS AppFabric for security を使用すると、 から監査ログとユーザーデータを受信しPagerDuty、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- PagerDuty での AppFabric のサポート
- AppFabric を PagerDuty アカウントに接続する

PagerDuty での AppFabric のサポート

AppFabric は、PagerDuty からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して PagerDuty からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- 監査ログにアクセスするには、PagerDuty のビジネスプランまたはデジタルオペレーションプランに加入している必要があります。
- ユーザーは、PagerDuty アカウントのグローバル管理者かアカウントオーナーである必要があります。

レート制限に関する考慮事項

PagerDuty は、PagerDuty API にレート制限を課します。PagerDuty API のレート制限の詳細については、「PagerDuty デベロッパープラットフォーム」の「REST API Rate Limits」を参照してください。AppFabric と既存の PagerDuty API アプリケーションの組み合わせが PagerDuty の制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を PagerDuty アカウントに接続する

PagerDuty プラットフォームは API アクセスキーをサポートしています。API アクセスキーを生成するには、次のステップを実行します。

API アクセスキーを作成する

AppFabric は、パブリッククライアント用の API アクセスキーを使用して PagerDuty と統合されています。PagerDuty で API アクセスキーを生成するには、次のステップを実行します。

1. PagerDuty のログインページに進み、サインインします。

2. [統合]、[API アクセスキー] の順に選択します。
3. [新しい API キーを作成] を選択します。
4. 説明を入力し、[読み取り専用 API キー] を選択します。
5. [Create Key] (キーを作成) を選択します。
6. API キーをコピーし、保存します。このキーは、後ほど、AppFabric で使用します。API キーを保存する前にページを閉じる場合は、新しい API キーを生成して保存する必要があります。PagerDuty API のレート制限を他の統合と共有しないようにするため、このキーは AppFabric 専用にします。

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。PagerDuty アカウントのテナント ID は、お使いのアカウントのベース URL です。この情報は、PagerDuty にログインし、ウェブブラウザのアドレスバーからコピーすることで確認できます。テナント ID は、次のいずれかの形式に従っている必要があります。

- 米国のアカウントの場合、`subdomain.pagerduty.com`
- EU のアカウントの場合、`subdomain.eu.pagerduty.com`

テナント名

この一意の PagerDuty 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

サービスアカウントトークン

AppFabric は、ユーザーのサービスアカウントトークンをリクエストします。AppFabric のサービスアカウントトークンは、API アクセスキーを作成する で作成した API アクセスキーです。

AppFabric Ping Identity用に を設定する

Ping Identity で私たちは、すべてのユーザーに安全かつシームレスなデジタル体験を、妥協なく実現することは可能だと考えます。だからこそ Ping Identity は、ユーザーのデジタルインタラクションを保護すると同時にスムーズなユーザーエクスペリエンスを実現するために、フォーチュン 100 企業の半数以上から選ばれているのです。2023 年 8 月 23 日、Ping Identity と ForgeRock は、より多

くの選択肢、より深い専門知識、より完全な ID ソリューションをお客様とパートナーにお届けするために提携しました。

AWS AppFabric for security を使用すると、 から監査ログとユーザーデータを受信しPing Identity、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Ping Identity での AppFabric のサポート
- AppFabric を Ping Identity アカウントに接続する

Ping Identity での AppFabric のサポート

AppFabric は、Ping Identity からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Ping Identity からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- Essential、Plus、Premium Ping Identity のいずれかのアカウントが必要です。該当する Ping Identity プランタイプの作成またはアップグレードの詳細については、「Ping Identity Web サイトのPing Identityすべての機能の価格表」を参照してください。
- Ping Identity アカウントには Identity Data Read Only のロールが必要です。アカウントには、アプリケーションにロールを付与することで、ロールを追加することができます。ロールの詳細については、Ping Identity サポートのウェブサイトの「Roles」を参照してください。

レート制限に関する考慮事項

Ping Identity はレート制限を公開していません。サポートケースを作成するか、Ping Identity カスタマーサクセスチームに連絡してください。AppFabric と既存の Ping Identity API アプリケーションの組み合わせが Ping Identity の制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によ

るものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Ping Identity アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Ping Identityを使用して AppFabric を認可する必要があります。AppFabric Ping Identityで を承認するために必要な情報を見つけるには、次のステップを使用します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して Ping Identity と統合されます。Ping Identityで OAuth アプリケーションを作成するには、以下の手順に従います。

1. Ping Identity ウェブサイトの「PingOne for Developers」ガイドにある「Create an application connection」のセクションの指示に従います。
2. アプリケーションを作成したら、付与のタイプをカスタマイズします。
   a. アプリケーションにサインインしたら、[設定] タブを選択し、鉛筆アイコンをクリックして既存の設定を変更します。
   b. Grant Type で、認可コードを選択します。[PKCE 実行] は [オプション] のままにしておきます。
   c. [更新トークン] を選択し、更新期間を選択します。
3. [リダイレクト URL/コールバック URL] では、次の形式のリダイレクト URL を使用します。

   ```
   https://<region>.console.aws.amazon.com/appfabric/oauth2
   ```

   この URL で、`<#####>`は、AppFabric アプリバンドルを構成した AWS リージョン のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。そのリージョンのリダイレクト URL は `https://us-east-1.console.aws.amazon.com/appfabric/oauth2` です。

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。AppFabric のテナント ID は Ping Identity インスタンス名です。ブラウザのアドレスバーにテナント ID が表示されます。例えば、`API_PATH`/v1/environments/`environmentID`。ここでは、`API_PATH` は PingOne サーバーのリージョンドメ

イン (`api.pingone.com` など) を表し、*`environmentID`* は、アプリケーションの環境プロパティで示された環境 ID を表します。環境プロパティの詳細については、Ping Identity のウェブサイトの「Environment Properties」を参照してください。

テナント名

この一意の Ping Identity 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。Ping Identityでクライアント ID を検索するには以下の手順を使用してください。

1. PingOne の管理コンソールにサインインし、[アプリケーション] を選択します。
2. リストの中からアプリケーションを選択します。
3. [概要] タブを選択し、[クライアント ID] の値を探します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。以下の手順でPing Identityのクライアントシークレット を検索してください。

1. PingOne の管理コンソールにサインインし、[アプリケーション] を選択します。
2. リストの中からアプリケーションを選択します。
3. [概要] タブを選択し、[クライアントシークレット] の値を探します。

認可を承認します

AppFabric でアプリ認可を作成すると、 から認可を承認Ping Identityするためのポップアップウィンドウが表示されます。AppFabric 認可を承認するには、許可を選択します。

AppFabric Salesforce用に を設定する

Salesforce は、企業がより多くの見込み客を見つけ、より多くの取引を成立させ、優れたサービスで顧客を驚かせるのに役立つように設計されたクラウドベースのソフトウェアを作成します。Salesforce'sCustomer 360 は、一連の製品を提供し、販売、サービス、マーケティング、コマース、IT の各チームを顧客情報の単一の共有ビューに統合し、組織が顧客や従業員との関係を成長させます。

AWS AppFabric を使用すると、 から監査ログとユーザーデータを受信しSalesforce、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Salesforce での AppFabric のサポート
- AppFabric を Salesforce アカウントに接続する

Salesforce での AppFabric のサポート

AppFabric は、Salesforce からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Salesforce からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- のパフォーマンス、エンタープライズ、または無制限エディションが必要ですSalesforce。これらのエディションのいずれかにアップグレードSalesforceするには、 にお問い合わせください。
- AppFabric が からのログイベントの完全なセットを含む時間単位のイベントログファイルを転送する場合はSalesforce、 の Shield 機能の一部として Event Monitoring をサブスクライブする必要がありますSalesforce。それ以外の場合、AppFabric はSalesforce's標準の日次ログファイルから制限付きイベント (ログイン、ログアウト、InsecureExternalAssets、API の合計使用量、CORS 違反、HostnameRedirects ELF Events など) を転送します。Salesforce アカウントが Shield 機能に既にサブスクライブされているかどうかを確認するには、「Setup > Event Manager」を参照してください。19 件以上のイベントが表示された場合、アカウントは Event Monitoring にサブスクライブされます。Event Monitoring がない場合は、 に連絡してこのアドオンのサブスクリプションを購入できますSalesforce。
- Salesforce 設定でイベントログファイルの生成をオプトインする必要があります。
- システム管理者プロファイルを使用して OAuth アプリケーションを作成し、AppFabric の同じ認証情報でログインする必要があります。

> **Note**
>
> API の合計使用量、CORS 違反レコード、ホスト名リダイレクト、安全でない外部アセット、ログイン、ログアウトイベントは、 のサポートされているエディションで追加料金な

しで利用できますSalesforce。残りのイベントタイプを購入するSalesforceには、 にお問い合わせください。Salesforce イベントタイプの詳細については、 Salesforceウェブサイトの[EventLogFile でサポートされているイベントタイプ]」を参照してください。
AppFabric は、ログファイルインスタンスごとにイベントタイプごとに最大 100,000 個のイベントをサポートできます (Event Monitoring アドオンサブスクリプションに応じて、毎日または毎時) 。しきい値を超えるログファイルがあると、ログファイル全体が取り込みから除外される可能性があります。

レート制限に関する考慮事項

Salesforce は、Salesforce API にレート制限を課します。Salesforce API レート制限の詳細については、 Salesforceウェブサイトの「API リクエストの制限と割り当て」を参照してください。AppFabric と既存の Salesforce API アプリケーションの組み合わせがSalesforce's制限を超えると、AppFabric に表示される監査ログが遅延する可能性があります。

データ遅延に関する考慮事項

監査イベントが宛先に配信されるように、1 日あたりのログファイルには最大 6 時間の遅延、1 時間あたりのログファイルには最大 29 時間の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Salesforce アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Salesforceを使用して AppFabric を認可する必要があります。AppFabric Salesforceで を認可するために必要な情報を確認するには、次のステップを実行します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して Salesforce と統合されます。Salesforceで OAuth アプリケーションを作成するには、以下の手順に従います。

1. Salesforceアカウントにログインします。
2. Salesforce ドキュメントの説明に従って、セットアップページに移動します。
3. クイック検索で App Manager を検索します。

4. 新しい接続されたアプリを選択します。
5. フォームフィールドに必要な情報を入力します。
6. OAuth 設定を有効にする を選択します。
7. 必ず以下のオプションをオフにしてください。
   - サポートされている認可フローにコード交換 (PKCE) 拡張機能の証明キーを要求する
   - ウェブサーバーフローにシークレットを要求する
   - 更新トークンフローにシークレットを要求する
   - トークンローテーションの更新を有効にする
8. コールバック URL テキストボックスに次の形式の URL を入力し、変更の保存を選択します。

   ```
   https://<region>.console.aws.amazon.com/appfabric/oauth2
   ```

   この URL で、*<#####>* iは、AppFabric アプリバンドルを構成した AWS リージョン のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。そのリージョンのリダイレクト URL は `https://us-east-1.console.aws.amazon.com/appfabric/oauth2` です。
9. 必要に応じてスコープを入力します (次の必要範囲セクションで説明) 。他のすべてのフィールドはデフォルト値のままにできます。
10. [保存] を選択します。
11. 次の手順を実行して、新しい OAuth アプリの更新トークンポリシーを確認します。
    a. Setup ページで、Quick Find テキストボックスに「Connected Apps」と入力し、「Manage Connected Apps」を選択します。
    b. 新しく作成したアプリの横にある編集 を選択します。
    c. 取り消されたオプションが選択されるまで、更新トークンが有効であることを確認します。
    d. 変更内容を保存します。
12. 監査ログが生成されていることを確認するには、次の手順を実行します。
    a. セットアップページで、クイック検索テキストボックスにイベントログファイルを入力し、イベントログファイルブラウザを選択します。
    b. イベントログがイベントログファイルブラウザにリストされていることを確認します。
13. 作成したアプリに移動し、ドロップダウンから表示を選択します。
14. [コンシューマーの詳細を管理] を選択します。

ID を検証する必要がある新しいタブにリダイレクトされます。そのタブで、コンシューマーキーとコンシューマーシークレットの値を書き留めます。サインインするには、後でこれらが必要になります。

必要範囲

Salesforce OAuth アプリケーションに次の範囲を入力する必要があります。

- APIs () を使用してユーザーデータを管理しますAPI。
- リクエストはいつでも実行します (`refresh_token` および `offline_access`) 。

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。AppFabric のテナント ID は、SalesforceMy Domain のサブドメインです。My Domain サブドメインは、ブラウザのアドレスバーで `https://` と の間で確認できます`.my.salesforce.com`。

Salesforce マイドメインを検索するには、Salesforceホーム画面から次の手順を使用します。

1. Salesforce ドキュメントの説明に従って、セットアップページに移動します。
2. クイック検索で会社設定を検索し、結果でマイドメインを選択します。

テナント名

この一意の Salesforce 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。Salesforceでクライアント ID を検索するには以下の手順を使用してください。

1. セットアップページに移動します。
2. Setup を選択し、App Manager を選択します。
3. 作成したアプリを選択し、ドロップダウンメニューから表示を選択します。
4. [コンシューマーの詳細を管理] を選択します。新しいタブにリダイレクトされます。

5. ID を確認し、コンシューマーキーの値を探します。
6. AppFabric のクライアント ID フィールドにコンシューマーキーを入力します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。AppFabric のクライアントシークレットは、のコンシューマーシークレットですSalesforce。でシークレットを検索するにはSalesforce、次の手順を実行します。

1. セットアップページに移動します。
2. Setup を選択し、App Manager を選択します。
3. 作成したアプリを選択し、ドロップダウンメニューから表示を選択します。
4. [コンシューマーの詳細を管理] を選択します。新しいタブにリダイレクトされます。
5. ID を確認し、コンシューマーシークレットの値を探します。
6. AppFabric のクライアントシークレットフィールドにコンシューマーシークレットを入力します。

認可を承認します

AppFabric でアプリ認可を作成すると、 から認可を承認Salesforceするためのポップアップウィンドウが表示されます。承認ページで、Salesforceシステム管理者ロール、または承認中にイベントログファイルの表示と API 有効Salesforceユーザーアクセス許可を持つユーザーを使用してください。許可 を選択して AppFabric 認可を承認します。

AppFabric ServiceNow用に を設定する

ServiceNowは、企業の IT 運用を自動化するクラウドベースのサービスの大手プロバイダです。ServiceNow の ITOM により、企業は仮想インフラストラクチャやクラウドインフラストラクチャを含む IT 環境全体を完全に可視化して制御できます。サービスのマッピング、提供、保証を簡素化し、ITサービスとインフラストラクチャのデータを単一の記録システムに統合します。また、イベント、インシデント、問題、構成、変更管理などの主要プロセスを自動化および合理化します。

AWS AppFabric for security を使用すると、 から監査ログとユーザーデータを受信しServiceNow、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- ServiceNow での AppFabric のサポート
- データ遅延に関する考慮事項
- AppFabric を ServiceNow アカウントに接続する

ServiceNow での AppFabric のサポート

AppFabric は、ServiceNow からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して ServiceNow からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- AppFabric はどのServiceNowプランタイプでも使用できます。
- ServiceNowアカウントには管理者ロールを持つユーザーが必要です。
- ServiceNow インスタンスが必要です。

レート制限に関する考慮事項

ServiceNow は、ServiceNow API にレート制限を課します。API のレート制限について詳しくは、ServiceNow Web サイトのインバウンドREST API レート制限 を参照してください。AppFabric と既存の ServiceNow API アプリケーションの組み合わせが制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を ServiceNow アカウントに接続する

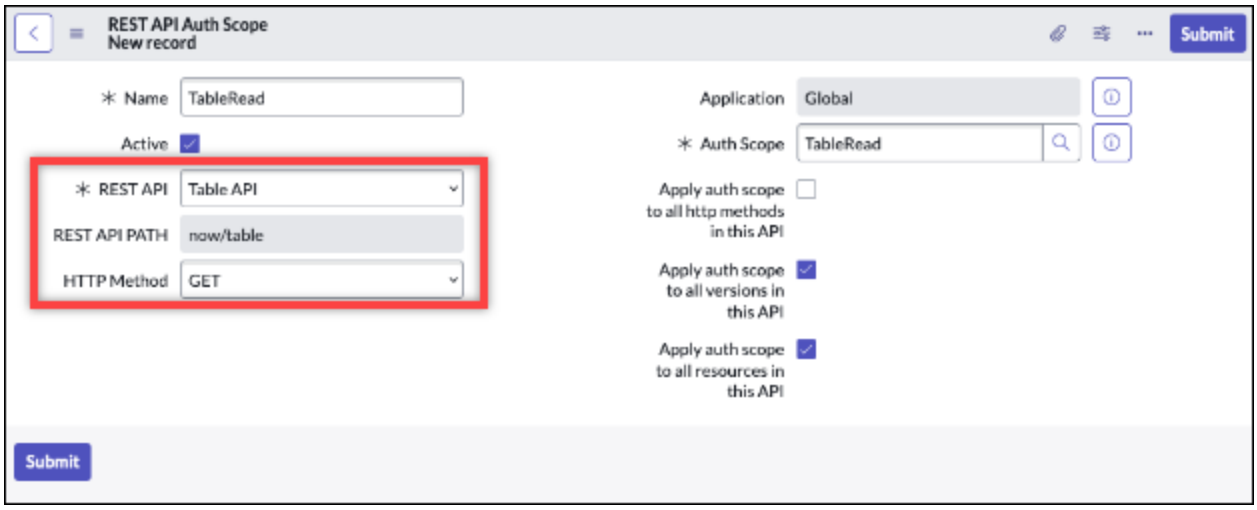
AppFabric サービス内でアプリケーションバンドルを作成した後で、ServiceNowを使用して AppFabric を認可する必要があります。AppFabric ServiceNowで を認可するために必要な情報を見つけるには、次のステップを使用します。

OAuth アプリケーションの作成

Now Platform は、パブリッククライアントがアクセストークンを生成するための OAuth 2.0 - 認可グラントタイプをサポートしています。

1. OAuth アプリケーションを登録します。この場合、以下の3ステップに従います。これらの手順を完了する方法について詳しくは、ServiceNow の ウェブサイトの「Register your application with ServiceNow」を参照してください。

   a. アプリを登録し、次の図のように [REST API PATH] は [now/table]、[HTTP メソッド]は [GET] として、[認証スコープ] が [Table API] にアクセスできるようにします。



   b. 認可コードを生成します。

   c. 認可コードを使用してベアラートークンを生成します。

2. 次の形式のリダイレクト URL を使用します。

```
https://<region>.console.aws.amazon.com/appfabric/oauth2
```

   この URL で、<#####> iは、AppFabric アプリバンドルを構成した AWS リージョン のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは us-east-1 です。そのリージョンのリダイレクト URL は https://us-east-1.console.aws.amazon.com/appfabric/oauth2 です。

アプリ権限

テナント ID

AppFabric はテナント ID をリクエストします。AppFabric のテナント ID はインスタンス名です。ブラウザのアドレスバーにテナント ID が表示されます。例えば、*example*は次の URL`https://`*example*`.service-now.com`のテナントIDです。

テナント名

この一意の ServiceNow 組織を識別する名前を入力します。AppFabric は、テナント名を使用してアプリ認可と、アプリ認可から作成されたすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。ServiceNowでクライアント ID を検索するには以下の手順を使用してください。

1. ServiceNow コンソールに移動します。
2. [システム OAuth]、[アプリケーションレジストリ] の順に選択します。
3. アプリケーションを選択します。
4. AppFabric の [クライアント ID] フィールドに OAuth クライアントのクライアント ID を入力します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。ServiceNow以下の手順でクライアントシークレット を検索してください。

1. ServiceNow コンソールに移動します。
2. [システム OAuth]、[アプリケーションレジストリ] の順に選択します。
3. アプリケーションを選択します。
4. OAuth アプリケーションのクライアントシークレットを AppFabric の [クライアントシークレット] フィールドに入力します。

認可を承認します

AppFabric でアプリ認可を作成すると、 から認可を承認ServiceNowするためのポップアップウィンドウが表示されます。許可 を選択して AppFabric 認可を承認します。

AppFabric Singularity Cloud用に を設定する

Singularity Cloud プラットフォームは、あらゆる段階で、あらゆるカテゴリの脅威から企業を保護します。その特許取得済みの人工知能は、既知の署名やパターンから、ゼロデイ攻撃やランサムウェアなどの最も高度な攻撃まで、セキュリティを拡張します。

AWS AppFabric を使用すると、 から監査ログとユーザーデータを受信しSingularity Cloud、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

> **Note**
>
> Singularity Cloud ドキュメントは、Singularity Cloudアカウントにサインインした後にのみアクセスできます。したがって、このSingularity Cloudドキュメントのドキュメントに直接リンクすることはできません。

トピック

- Singularity Cloud での AppFabric のサポート
- AppFabric を Singularity Cloud アカウントに接続する

Singularity Cloud での AppFabric のサポート

AppFabric は、Singularity Cloud からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して からサポートされている宛先Singularity Cloudに監査ログを転送するには、Singularity Cloudアカウントに管理者ロールが必要です。Singularity Cloud API レート制限の詳細については、Singularity Cloud アカウントにサインインし、ドキュメントセクションを参照し、ロールを検索します。

レート制限に関する考慮事項

Singularity Cloud は、Singularity Cloud API にレート制限を課します。Singularity Cloud API レート制限の詳細については、Singularity Cloud アカウントにサインインし、ドキュメントセクションを参照し、API レート制限を検索します。

データ遅延に関する考慮事項

監査イベントが送信先に配信されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Singularity Cloud アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Singularity Cloudを使用して AppFabric を認可する必要があります。AppFabric Singularity Cloudで認可するために必要な情報を見つけるには、次のステップを使用します。

の API トークンを作成する Singularity Cloud

サービスユーザーに関連付けられている API トークンを作成するには、次の手順を実行します。API トークンは、特定のコンソールユーザーまたは E メールアドレスにリンクされません。

> **Note**
>
> 新しいユーザーを作成するか、サービスユーザーをコピーして、サービスユーザー API トークンの有効期限が切れる前または後に新しい API トークンを取得します。

1. Singularity Cloud アカウントにサインインします。
2. 設定ツールバーで、ユーザーを選択し、サービスユーザーを選択します。
3. アクションを選択し、新しいサービスユーザーの作成を選択します。
4. 新しいサービスユーザーの作成 ページで、サービスユーザーの名前、説明、有効期限を入力します。
5. [Next (次へ)] を選択します。
6. アクセス範囲の選択セクションで、スコープを選択します。
   - アクセスレベルのアカウントを選択します。
   - 監査ログを取得するアカウントを選択します。
7. [ユーザーを作成] をクリックします。

   API トークンが生成されます。ウィンドウが開き、トークンを最後に表示できることを示すメッセージがトークン文字列に表示されます。

8. (オプション) API トークンのコピーを選択し、安全な場所に保存します。
9. [閉じる] を選択します。

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。AppFabric のテナント ID は、サービスにサインインするSentinel Oneウェブサイトアドレスのサブドメインになります。例えば、`example-company-1.sentinelone.net`住所でSingularity Cloudアカウントにサインインした場合、テナント ID は ですexample-company-1。

テナント名

この一意の Singularity Cloud 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

サービスアカウントトークン

このガイドの の API トークンを作成する Singularity Cloudセクションのステップを使用して生成したトークンを使用します。トークンを紛失したり、トークンを見つけられない場合は、同じステップを再度実行して新しいトークンを生成できます。

> **Note**
>
> AppFabric が監査ログを取り込む間に Singularity Cloud コンソールで新しい API トークンが生成された場合、取り込みは停止します。このような場合は、新しい API トークンでアプリ認可を更新して、監査ログの取り込みを再開する必要があります。

AppFabric Slack用に を設定する

Slack は人々のワーキングライフをよりシンプルに、より楽しく、より生産的なものにすることを使命としています。シームレスな検索とナレッジ共有およびコード不要の自動化に加え、チームのつながりを強化して目標達成に向けた協力体制を支援することでパフォーマンスを向上させる、顧客企業向けの生産性プラットフォームです。Salesforce の一部として、Slack は Salesforce Customer 360 に深く統合されているため、営業、サービス、マーケティングの各チーム全体の生産性が大幅に向上します。Slack の詳細を確認して無料で使い始めるには、slack.com をご覧ください。

AWS AppFabric for security を使用すると、 からログとユーザーデータを監査しSlack、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Slack での AppFabric のサポート
- AppFabric を Slack アカウントに接続する

Slack での AppFabric のサポート

AppFabric は、Slack からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Slack からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- Slack でのエンタープライズグリッドプランへの加入が必要です。詳細については、Slack のウェブサイト「Slack Enterprise Grid」を参照してください。
- Slack アカウントには組織の所有者ロールを持つユーザーが必要です。ロールの詳細については、Slack のウェブサイトの Slack ヘルプセンター にある「Types of roles in Slack」を参照してください。

レート制限に関する考慮事項

Slack は、Slack API にレート制限を課します。Slack API レート制限の詳細については、Slack ウェブサイトの「Slack API 使用ガイド」にある「レート制限」を参照してください。AppFabric と既存の Slack API アプリケーションの組み合わせが制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Slack アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Slackを使用して AppFabric を認可する必要があります。AppFabric Slackで を承認するために必要な情報を見つけるには、次のステップを使用します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して Slack と統合されます。OAuth アプリを作成するには、アプリマニフェストを使用する方法と、ゼロから作成する方法の 2 つがあります。Slack で OAuth アプリケーションを作成するには、以下の手順に従います。

Using an app manifest

1. ブラウザでSlack アプリ管理 UI に移動します。
2. [新しいアプリの作成] を選択します。
3. [アプリマニフェストから] を選択します。
4. AppFabric を承認するワークスペースを選択します。
5. [以下にアプリマニフェストを入力] ボックスで [JSON] を選択し、既存の JSON を次のものに置き換えます。*<region>* を適切な AWS リージョン（例:) に置き換えます*us-east-1*。

```
{
    "display_information": {
        "name": "AppFabric"
    },
    "oauth_config": {
        "redirect_urls": [
            "https://<region>.console.aws.amazon.com/appfabric/oauth2"
        ],
        "scopes": {
            "user": [
                "auditlogs:read",
                "users:read.email",
                "users:read"
            ]
        }
    },
    "settings": {
        "org_deploy_enabled": false,
```

```
        "socket_mode_enabled": false,
        "token_rotation_enabled": true
    }
}
```

6. [基本情報] ページからクライアント ID とクライアントシークレットをコピーして保存します。
7. `auditLogs:read` の範囲では、アプリのパブリックディストリビューションを有効にする必要があります。詳しくは、Slack のウェブサイト「Enabling public distribution」を参照してください。

From scratch

1. [アプリの作成] 画面で [ゼロから作成] を選択します。
2. アプリに名前を付け、ワークスペースを選択します。
3. [基本情報] ページからクライアント ID とクライアントシークレットをコピーして保存します。
4. [OAuth および許可] ページで、[トークンローテーションによる高度なトークンセキュリティ] オプションを選択します。
5. [OAuth および許可] ページの [リダイレクト URL] セクションに、次の形式の URL を追加します。

   ```
   https://<region>.console.aws.amazon.com/appfabric/oauth2
   ```

   この URL AWS リージョン では、 *<region>*は AppFabric アプリバンドルを設定した のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。そのリージョンのリダイレクト URL は `https://`*`us-east-1`*`.console.aws.amazon.com/appfabric/oauth2` です。
6. `auditLogs:read` の範囲では、アプリのパブリックディストリビューションを有効にする必要があります。詳しくは、Slack のウェブサイト「Enabling public distribution」を参照してください。

必要範囲

> **Note**
>
> このセクションは、OAuth アプリをゼロから作成することを選択した場合にのみ適用されます。アプリマニフェストを使用してアプリケーション認可を作成することを選択した場合は、このセクションをスキップしてください。

Slack OAuthアプリケーションの [OAuth および許可] ページで次のユーザートークンの範囲を追加する必要があります。

- `auditlogs:read`
- `users:read.email`
- `users:read`

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。AppFabricのテナント ID は Slack ワークプレイス IDです。テナント ID を取得するには、Slack ウェブサイトの Slack ヘルプセンター」にある「Locate your Slack URL」の手順に従ってください。Slack ワークスペース URL の形式は、`examplecorp.slack.com` または `examplecorp.enterprise.slack.com` に似ています。必要なテナント ID は、`.slack.com` または `.enterprise.slack.com` が付いていない `examplecorp` です。

テナント名

Slack ワークスペース ID を識別する名前を入力します。AppFabric は、テナント名を使用してアプリ認可と、アプリ認可から作成されたすべての取り込みにラベルを付けます

クライアント ID

AppFabric は Slack OAuth アプリケーションからクライアント ID を要求します。クライアント ID を確認するには、以下のステップに従います。

1. ブラウザでSlack アプリ管理 UI に移動します。
2. AppFabricに使用する OAuth アプリケーションを選択します。

3. [基本情報] ページのクライアント ID を AppFabric の [クライアント ID] フィールドに入力します。

クライアントシークレット

AppFabric は Slack OAuth アプリケーションにクライアントシークレット を要求します。クライアントシークレット を確認するには、以下の手順に従います。

1. ブラウザでSlack アプリ管理 UI に移動します。
2. AppFabric に使用する OAuth アプリケーションを選択します。
3. [基本情報] ページのクライアントシークレットを AppFabric の [クライアントシークレット] フィールドに入力します。

認可を承認します

AppFabric でアプリ認可を作成すると、 から認可を承認Slackするためのポップアップウィンドウが表示されます。AppFabric 認可を承認するには、許可を選択します。

AppFabric Smartsheet用に を設定する

Smartsheet は、企業全体で仕事、人材、テクノロジーを連携させる上で役立つワークマネジメントプラットフォームです。 Smartsheet は誰もがプロジェクト管理、ワークフローの自動化、大規模なソリューションの迅速な構築を行えるように支援するエンタープライズグレードの堅牢な機能セットを提供し、セキュリティとコンプライアンスを維持しながらイノベーションを実現する環境を作り出します。

AWS AppFabric for security を使用すると、 からログとユーザーデータを監査しSmartsheet、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Smartsheet での AppFabric のサポート
- AppFabric を Smartsheet アカウントに接続する

Smartsheet での AppFabric のサポート

AppFabric は、Smartsheet からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Smartsheet からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- Smartsheet ビジネス、エンタープライズ、またはアドバンスアカウントが必要です。Smartsheet アカウントの作成またはアップグレードの詳細については、Smartsheet ウェブサイトの「Smartsheet pricing」または「Smartsheet Advance」を参照してください。
- Smartsheet開発者登録プロセスを完了する必要があります。

レート制限に関する考慮事項

Smartsheet は、Smartsheet API にレート制限を課します。Smartsheet API レート制限の詳細については、Smartsheet ウェブサイトの Smartsheet API Reference の「Rate limiting」を参照してください。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Smartsheet アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Smartsheetを使用して AppFabric を認可する必要があります。AppFabric Smartsheetで を認可するために必要な情報を確認するには、次の手順を実行します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して Smartsheet と統合されます。Smartsheet で OAuth アプリケーションを作成するには、以下の手順に従います。

1. Smartsheet アカウントの開発者ツールに移動します。
2. 開発者ツールの画面で、[新規アプリの作成] を選択します。
3. [新規アプリの作成] 画面のすべての入力フィールドに入力します。
4. [アプリの URL] と [アプリの連絡先/サポート] には任意の一意の値を使用してください。

5. 次の形式のリダイレクト URL をアプリリダイレクト URL として使用します。

```
https://<region>.console.aws.amazon.com/appfabric/oauth2
```

この URL AWS リージョン では、 *<region>*は AppFabric アプリバンドルを設定した のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。そのリージョンのリダイレクト URL は `https://us-east-1.console.aws.amazon.com/appfabric/oauth2` です。

6. [保存] を選択します。
7. アプリクライアント ID およびアプリシークレットをコピーして保存します。

必要範囲

Smartsheet OAuth の構成に範囲を明示的に追加する必要はありません。AppFabric は、認可リクエストの以下のスコープをSmartsheetアカウントにリクエストします。

- `READ_EVENTS`
- `READ_USERS`

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。AppFabric のテナント ID は、Smartsheet アカウント ID です。

テナント名

AppFabric はテナント ID を要求します。Smartsheet アカウントを一意に識別する任意の値を入力します。

クライアント ID

AppFabric はクライアント ID を要求します。AppFabric のクライアント ID は、Smartsheet アプリクライアント ID です。Smartsheet でアプリクライアント ID を確認するには、以下の手順に従います。

1. Smartsheet アカウントの開発者ツールに移動します。

2. AppFabric との接続に使用する OAuth アプリケーションを選択します。
3. [アプリプロファイル] 画面のアプリクライアント ID を、 AppFabric の [クライアント ID] フィールドに入力します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。AppFabric のクライアントシークレットは、Smartsheet アプリシークレットです。Smartsheet でアプリシークレットを確認するには、次のステップに従います。

1. Smartsheet アカウントの開発者ツールに移動します。
2. AppFabric との接続に使用する OAuth アプリケーションを選択します。
3. [アプリのプロファイル] 画面のアプリシークレットを、AppFabric の [クライアントシークレット] フィールドに入力します。

認可を承認します

AppFabric でアプリ認可を作成すると、 から認可を承認Smartsheetするためのポップアップウィンドウが表示されます。AppFabric 認可を承認するには、許可を選択します。

AppFabric Terraform Cloud用に を設定する

HashiCorp Terraform Cloud は、世界中で広く使用されているマルチクラウドプロビジョニング製品です。Terraform エコシステムには、3,000 を超えるプロバイダー、14,000 のモジュール、2 億 5,000 万件のダウンロードがあります。 Terraform Cloudは、 を導入する最も速い方法であり Terraform、プラクティショナー、チーム、グローバル企業がインフラストラクチャの作成と共同作業を行い、セキュリティ、コンプライアンス、運用上の制約のリスクを管理するために必要なすべてを提供します。

AWS AppFabric for security を使用すると、 から監査ログとユーザーデータを受信しTerraform Cloud、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Terraform Cloud での AppFabric のサポート
- AppFabric を Terraform Cloud アカウントに接続する

Terraform Cloud での AppFabric のサポート

AppFabric は、Terraform Cloud からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Terraform Cloud からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- 監査ログにアクセスするには、 Terraform Cloud Plus Edition プランがあり、組織の所有者である必要があります。Terraform Cloud プランの詳細については、 HashiCorp Terraformウェブサイトの「 のTerraform料金」を参照してください。
- TBD 監査ログは、Terraform Cloudアカウントから作成できる組織で使用できます。

レート制限に関する考慮事項

Terraform Cloud は、Terraform Cloud API にレート制限を課します。Terraform Cloud API レート制限の詳細については、 Terraform CloudウェブサイトのTerraform Cloud「開発者管理全般設定」の「API レート制限」を参照してください。AppFabric と既存の Terraform Cloud API アプリケーションの組み合わせが Terraform Cloud の制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Terraform Cloud アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Terraform Cloudを使用して AppFabric を認可する必要があります。AppFabric Terraform Cloudで を認可するために必要な情報を確認するには、次の手順を実行します。

組織 API トークンを作成する

AppFabric は、組織 API トークンTerraform Cloudを使用して と統合します。Terraform Cloud 組織 API トークンの詳細については、「組織 API トークン」を参照してください。組織を作成するには、「組織の作成」の手順に従います。で組織 API トークンを作成するにはTerraform Cloud、次のステップを使用します。

1. Terraform Cloud サインインページに移動し、サインインします。
2. 左側のパネルで組織、設定を選択し、API トークンを選択します。
3. Organization Tokens で、Create an organization token を選択し、Generate token を選択します。
4. (オプション) トークンの有効期限の日時を入力するか、有効期限のないトークンを作成します。
5. トークンをコピーして保存します。このキーは、後ほど、AppFabric で使用します。トークンを保存する前にページを閉じる場合は、古いトークンを取り消して新しいトークンを作成する必要があります。

アプリ権限

テナント ID

AppFabric はテナント ID をリクエストします。Terraform Cloud アカウントのテナント ID は、アカウントの現在の組織 URL です。これは、Terraform Cloud組織にログインし、現在の組織 URL をコピーすることで確認できます。テナント ID は、次のいずれかの形式に従っている必要があります。

```
https://app.terraform.io/app/organization_URL
```

テナント名

この一意の Terraform Cloud 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

サービスアカウントトークン

AppFabric は、ユーザーのサービスアカウントトークンをリクエストします。AppFabric のサービスアカウントトークンは、 で作成した組織 API トークンです組織 API トークンを作成する。

AppFabric Webex by Cisco用に を設定する

Cisco は、インターネットを支えるテクノロジーの世界的リーダーです。Cisco はアプリケーションの新たな概念をもたらし、データを保護し、インフラストラクチャを変革させ、グローバルでインクルーシブな未来に向けてチームを強化することで、新しい可能性を広げます。

Webex by Cisco について

Webex は、ビデオ会議、通話、メッセージング、イベント、コンタクトセンターや専用コラボレーションデバイスなどの顧客体験ソリューションを含む、クラウドベースのコラボレーションソリューションの大手プロバイダです。Webex は、インクルーシブなコラボレーション体験を重視し、AI と機械学習を活用したイノベーションにより地理、言語、性格、テクノロジーへの精通度といった障壁を排除するイノベーションを推進しています。同社のソリューションは、そのセキュリティとプライバシーバイデザインに支えられています。Webex は、単一のアプリケーションとインターフェースを通じて世界をリードするビジネスアプリや生産性向上アプリとの連携を提供します。詳細については、「webex.com」を参照してください。

AWS AppFabric for security を使用すると、 からログとユーザーデータを監査しWebex、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Webex での AppFabric のサポート
- AppFabric を Webex アカウントに接続する

Webex での AppFabric のサポート

AppFabric は、Webex からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Webex からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- コラボレーションフレックスプラン、Meet プラン、Call プラン、またはそれ以上が必要です。該当する Webex プランタイプの作成またはアップグレードの詳細については、「Webex Web サイトのWebexすべての機能の価格表」を参照してください。
- Cisco AuditLog API のいずれかが提供するセキュリティ監査イベントにアクセスするには、アカウントに Pro Pack ライセンスが必要です。
- 組織管理者 > 完全な管理者権限を持つユーザーが必要です。
- 完全な管理者権限を持つ管理者ロールの設定では、コンプライアンスオフィサーオプションが有効になっている必要があります。

レート制限に関する考慮事項

Webex は、Webex API にレート制限を課します。Webex API のレート制限の詳細については、Webex Web サイトの「Webex 開発者ガイド」の「レート制限」を参照してください。AppFabric と既存の Webex API アプリケーションの組み合わせが制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Webex アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Webexを使用して AppFabric を認可する必要があります。AppFabric Webexで を認可するために必要な情報を確認するには、次のステップを使用します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して Webex と統合されます。Webex で OAuth アプリケーションを作成するには、以下の手順に従います。

1. デベロッパーガイドの「統合と認可」ページの「統合の登録」セクションの指示に従ってください。 Webex
2. 次の形式のリダイレクト URL を使用します。

   ```
   https://<region>.console.aws.amazon.com/appfabric/oauth2
   ```

   この URL AWS リージョン では、 *<region>*は AppFabric アプリバンドルを設定した のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。そのリージョンのリダイレクト URL は `https://us-east-1.console.aws.amazon.com/appfabric/oauth2` です。

必要範囲

Webex OAuth アプリケーションに次の範囲を入力する必要があります。

- `spark-compliance:events_read`
- `audit:events_read`
- `spark-admin:people_read`

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。AppFabric のテナント ID は、Webex の組織 ID です。Webex 組織 ID を確認する方法については、Webex ヘルプセンターウェブサイトの「CiscoWebex Control Hub で組織 ID を検索する」を参照してください。

テナント名

この一意の Webex インスタンスを識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はWebex クライアント ID を要求します。Webex クライアント ID を確認するには、以下のステップに従います。

1. https://developer.webex.com で Webex アカウントにサインインします。
2. 右上のアバターを選択します。
3. [My Webex アプリ] を選択します。
4. AppFabric に使用する OAuth2 アプリケーションを選択します。
5. このページのクライアント ID を AppFabric の [クライアント ID] フィールドに入力します。

クライアントシークレット

AppFabric は、Webex クライアントシークレットを要求します。Webex がクライアントシークレットを表示するのは、OAuth アプリケーションを最初に作成したときのみです。最初のクライアントシークレットを保存しなかった場合に新しいクライアントシークレットを生成するには、以下の手順に従います。

1. https://developer.webex.com で Webex アカウントにサインインします。
2. 右上のアバターを選択します。

3. [My Webex アプリ] を選択します。
4. AppFabric に使用する OAuth2 アプリケーションを選択します。
5. このページで、新しいクライアントシークレットを生成します。
6. AppFabric の [クライアントシークレット] フィールドに新しいクライアントシークレットを入力します。

認可を承認します

AppFabric でアプリ認可を作成すると、 から認可を承認Webexするためのポップアップウィンドウが表示されます。AppFabric 認可を承認するには、承認を選択します。

AppFabric Zendesk用に を設定する

2007 年に世界中のあらゆる企業がカスタマーサービスをオンライン化できるようにすることで、Zendeskがカスタマーエクスペリエンス革命を始めました。現在では、Zendeskは、あらゆる場所ですべての人に優れたサービスを提供し、何十億もの会話を支えています。電話、チャット、電子メール、メッセージング、ソーシャルチャネル、コミュニティ、レビューサイト、ヘルプセンターを通じて、10万を超えるブランドと数億人の顧客をつなげています。 Zendesk製品は愛されるための愛を込めて作られています。同社はデンマークのコペンハーゲンで設立され、カリフォルニアで建設・ 栽培され、現在では世界中で6,000人以上の従業員を雇用しています。

AWS AppFabric for security を使用すると、 からログとユーザーデータを監査しZendesk、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Zendesk での AppFabric のサポート
- AppFabric を Zendesk アカウントに接続する

Zendesk での AppFabric のサポート

AppFabric は、Zendesk からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Zendesk からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- Zendeskスイートエンタープライズアカウント、エンタープライズプラスアカウント、またはZendeskSupport エンタープライズアカウントが必要です。Zendesk Enterprise アカウントの作成またはアップグレードについて詳しくは、Zendesk ウェブサイトの「Zendesk プランタイプの確認」を参照してください。
- Zendeskアカウントには管理者ロールを持つユーザーが必要です。ロールの詳細については、Zendeskウェブサイトの「Zendeskサポート ユーザーのロールについて」を参照してください。

レート制限に関する考慮事項

Zendesk は、Zendesk API にレート制限を課します。Zendesk API のレート制限の詳細については、Zendesk Web サイトの「Zendesk 開発者ガイド」の「レート制限」を参照してください。AppFabric と既存の Zendesk API アプリケーションの組み合わせが制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に転送されるまでに最大 30 分の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。ただし、これはアカウントレベルでカスタマイズできる場合があります。サポートが必要な場合は、サポート にお問い合わせください。

AppFabric を Zendesk アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Zendeskを使用して AppFabric を認可する必要があります。AppFabric Zendeskで を承認するために必要な情報を見つけるには、次のステップを使用します。

OAuth アプリケーションの作成

AppFabric は OAuth を使用して Zendesk と統合されます。Zendeskでは、以下の設定で OAuth アプリケーションを作成する必要があります。

1. Zendeskサポート ebサイトの「アプリケーションでのOAuth認証の使用」の「WZendeskへのアプリケーションの登録」セクションの指示に従ってください。
2. 次の形式のリダイレクト URL を使用します。

```
https://<region>.console.aws.amazon.com/appfabric/oauth2
```

この URL AWS リージョン では、 `<region>`は AppFabric アプリバンドルを設定した のコードです。例えば、米国東部 (バージニア北部) リージョンのコードは `us-east-1` です。そのリージョンのリダイレクト URL は `https://us-east-1.console.aws.amazon.com/appfabric/oauth2` です。

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。AppFabric のテナント ID はZendeskサブドメインです。Zendesk サブドメインを見つける方法について詳しくは、Zendesk Support Web サイトの「Zendesk サブドメインはどこで見つけられますか?」を参照してください。

テナント名

この一意の Zendesk 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。AppFabric のクライアント ID は Zendesk API 固有の識別子です。Zendesk 固有の識別子を見つけるには、次の手順に従います。

1. Zendeskアカウントの「管理センター」に移動します。
2. [アプリとインテグレーション] を選択します。
3. [API」, Zendesk[API] .を選択します。
4. [OAuth クライアント] タブを選択します。
5. AppFabricに対して作成した OAuthアプリケーションを選択します。
6. AppFabric の [クライアント ID] フィールドに OAuth クライアントの固有識別子を入力します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。AppFabric Zendesk のクライアントシークレットはシークレットトークンです。Zendesk OAuth アプリケーションを初めて作成したときに、Zendesk がシークレットトークンを提示するのは 1 回だけです。最初のシークレットトークンを保存しなかった場合に新しいシークレットトークンを生成するには、以下の手順に従います。

1. Zendeskアカウントの「管理センター」に移動します。

2. [アプリとインテグレーション] を選択します。
3. [API」, Zendesk[API] .を選択します。
4. [OAuth クライアント] タブを選択します。
5. AppFabricに対して作成した OAuthアプリケーションを選択します。
6. [シークレットトークン] フィールドの横にある [再作成] ボタンを選択します。
7. AppFabric の [クライアントシークレット] フィールドに新しいシークレットトークンを入力します。

認可を承認します

AppFabric でアプリ認可を作成すると、 から認可を承認Zendeskするためのポップアップウィンドウが表示されます。AppFabric 認可を承認するには、許可を選択します。

AppFabric Zoom用に を設定する

Zoom は、企業や個人間の接続をより簡単かつ没入的に、そしてよりダイナミックに行えるオールインワンのインテリジェントなコラボレーションプラットフォームです。Zoom は人中心のテクノロジーとして、チームチャット、電話、会議、オムニチャネルクラウドコンタクトセンター、スマートレコーディング、ホワイトボードなどのソリューションを 1 つの提供サービスにまとめ、有意義なつながりを可能にし、最新のコラボレーションとヒューマンイノベーションを推進します。

AWS AppFabric for security を使用すると、 からログとユーザーデータを監査しZoom、データを Open Cybersecurity Schema Framework (OCSF) 形式に正規化して、Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose ストリームにデータを出力できます。

トピック

- Zoom での AppFabric のサポート
- AppFabric を Zoom アカウントに接続する

Zoom での AppFabric のサポート

AppFabric は、Zoom からのユーザー情報と監査ログの受信をサポートします。

前提条件

AppFabric を使用して Zoom からサポートされている宛先に監査ログを転送するには、以下の要件を満たす必要があります。

- Zoom のプロ、ビジネス、エデュケーション、またはエンタープライズプランに加入している必要があります。
- Zoom管理者ロールには、Server to Server OAuth アプリケーションを作成する許可が必要です。Server to Server OAuth アプリケーションの有効化に関する詳細については、Zoom ウェブサイトに掲載されている「Zoomデベロッパーガイド」の「Server to Server OAuth」ページにある「許可の有効化」セクションを参照してください。
- Zoom管理者ロールには、管理アクティビティログを表示したり、監査アクティビティにサインイン/サインアウトしたりする許可が必要です。監査アクティビティを閲覧する許可を有効にする方法の詳細については、Zoom サポートウェブサイトの「ロール管理の使用」と「管理者アクティビティログの使用」を参照してください。

レート制限に関する考慮事項

Zoom は、Zoom API にレート制限を課します。Zoom API 制限の詳細については、「Zoomデベロッパーガイド」の「レート制限」を参照してください。AppFabric と既存の Zoom アプリケーションの組み合わせが制限を超えると、AppFabric に監査ログが表示されるのが遅れる可能性があります。

データ遅延に関する考慮事項

監査イベントが取り込み先に配信されるまでに約 24 時間の遅延が発生する場合があります。これは、アプリケーションで利用できる監査イベントの遅延と、データ損失を減らすための予防措置によるものです。

AppFabric を Zoom アカウントに接続する

AppFabric サービス内でアプリケーションバンドルを作成した後で、Zoomで AppFabric を認可する必要があります。AppFabric Zoomで を承認するために必要な情報を見つけるには、次のステップを使用します。

Server-to-server OAuth アプリケーションを作成する

AppFabric は、server-to-server OAuth とアプリの認証情報を使用して Zoom と統合します。Zoom で server-to-server OAuth アプリケーションを作成するには、「Zoomデベロッパーガイド」の「Server-to-Server OAuth アプリの作成」の手順に従ってください。AppFabric は Zoom Webhook をサポートしていないため、 Webhook サブスクリプションの追加に関するセクションはスキップできます。

必要範囲

Zoom には、きめ細かなスコープ (新しく作成されたアプリケーションの場合) と従来のスコープ (以前に作成されたアプリケーションの場合) の 2 種類のスコープがあります。

Zoom server-to-server OAuth アプリケーションには、次の詳細なスコープを追加する必要があります。

- `report:read:user_activities:admin`
- `report:read:operation_logs:admin`
- `user:read:email:admin`
- `user:read:user:admin`

以前に作成したアプリケーションを使用している場合は、次のクラシックスコープを追加する必要があります。

- `report:read:admin`
- `user:read:admin`

アプリ権限

テナント ID

AppFabric はテナント ID を要求します。AppFabric のテナント ID Zoom はアカウント ID です。Zoom アカウント ID を確認するには、以下のステップに従います。

1. Zoom Marketplace にアクセスします。
2. [管理] を選択します。
3. AppFabric に使用する server-to-server OAuth アプリケーションを選択します。
4. [アプリ認証情報] ページのアカウント ID を AppFabric の [テナント ID] フィールドに入力します。

テナント名

この一意の Zoom 組織を識別する名前を入力します。AppFabric は、テナント名を使用して、アプリ認可と、アプリ認可から作成されるすべての取り込みにラベルを付けます。

クライアント ID

AppFabric はクライアント ID を要求します。Zoom クライアント ID を確認するには、以下のステップに従います。

1. Zoom Marketplace にアクセスします。
2. [管理] を選択します。
3. AppFabric に使用する server-to-server OAuth アプリケーションを選択します。
4. [アプリ認証情報] ページのクライアント ID を AppFabric の [クライアント ID] フィールドに入力します。

クライアントシークレット

AppFabric はクライアントシークレットを要求します。Zoom クライアントシークレットを確認するには、以下のステップに従います。

1. Zoom Marketplace にアクセスします。
2. [管理] を選択します。
3. AppFabric に使用する server-to-server OAuth アプリケーションを選択します。
4. [アプリ認証情報] ページのクライアントシークレットを AppFabric の [クライアント シークレット] フィールドに入力します。

監査ログの配信

Zoom は、24 時間ごとに API にアクセスして監査ログを利用できるようにします。AppFabric で監査ログを表示する際にZoom について表示されるデータは、前日のアクティビティに関するものです。

AppFabric for security の互換性のあるセキュリティツールとサービス

AWS AppFabric for security は、以下のセキュリティツールおよびサービスとの統合をサポートしています。AppFabric for security を設定して接続する方法の詳細を読むには、サービス名を選択します。

トピック

- Barracuda XDR
- Dynatrace
- Logz.io
- Netskope
- NetWitness
- Amazon QuickSight
- Rapid7
- Amazon Security Lake
- Singularity Cloud
- Splunk

Barracuda XDR

Barracuda Networks は、ビジネスのジャーニーに合わせて成長し変化する革新的なソリューションにより、E メール、ネットワーク、データ、アプリケーションを保護する、クラウドファーストなセキュリティソリューションを提供している、信頼できるパートナーであり業界をリードするプロバイダーです。Barracuda XDR は、高度なテクノロジーと、セキュリティオペレーションセンター (SOC) のセキュリティアナリストチームとを組み合わせた、オープンな、拡張された、検知および対応ソリューションです。Barracuda XDR のプラットフォームは、40 を超える統合データソースの、一日数十億件に上る未加工のイベントを分析し、MITRE ATT&CK® フレームワークに対応した広範な脅威検出ルールとともに、これまでよりもスピーディに脅威を検出し、より短時間で対応します。

AWS AppFabric 監査ログの取り込みに関する考慮事項

以下のセクションでは、Barracuda XDR で使用する AppFabric 出力スキーマ、出力形式、出力先について説明します。

スキーマと形式

Barracuda XDR は、以下の AppFabric 出力スキーマと形式をサポートしています。

- OCSF - JSON: AppFabric はオープンサイバーセキュリティスキーマフレームワーク (OCSF) を使用してデータを正規化し、データを JSON 形式で出力します。

出力場所

Barracuda XDR は、Amazon Security Lake の監査ログの受信をサポートしています。AppFabric から Barracuda XDR にデータを送信するときは、以下の手順に従います。

1. Amazon Security Lake にデータを送信する: Amazon Data Firehose を介して Amazon Security Lake にデータを送信するように AppFabric を設定します。詳細については、「Amazon Security Lake」を参照してください。
2. Barracuda XDR にデータを送信する: Amazon Security Lake から監査ログを受信するように Barracuda XDR を設定します。詳細については、「Setting Up and Using Amazon Security Lake」を参照してください。

Dynatrace

Dynatrace® Platform は、広範で深いオブザーバビリティと継続的なランタイムアプリケーションセキュリティを高度な AIOps と組み合わせて、データからの回答とインテリジェントな自動化を提供します。これにより、イノベーターはクラウド運用をモダナイズして自動化し、ソフトウェアをより迅速かつ安全に提供し、完璧なデジタルエクスペリエンスを確保できます。

AWS AppFabric 監査ログの取り込みに関する考慮事項

以下のセクションでは、 で使用する AppFabric 出力スキーマ、出力形式、出力先について説明しますDynatrace Platform。

スキーマと形式

は、次の AppFabric 出力スキーマと形式Dynatrace Platformをサポートしています。

- OCSF - JSON: AppFabric はオープンサイバーセキュリティスキーマフレームワーク (OCSF) を使用してデータを正規化し、データを JSON 形式で出力します。

出力場所

は、次の AppFabric 出力場所からの監査ログの受信Dynatrace Platformをサポートします。

- Amazon Simple Storage Service (Amazon S3)
  - 監査ログを含む Amazon S3 バケットからデータを受信するDynatrace Platformように を設定するには、 の Dynatrace の S3 Log Forwarder プロジェクトの手順に従ってくださいGitHub。

Logz.io

Logz.io は、オブザーバビリティとセキュリティを高コストで低価値の負担となるものから、より優れたビジネス成果を実現する高価値でコスト効率の高いものに変えることで、クラウドネイティブ企業がLogz.io Open 360 Platformを通して自社の環境を監視し保護できるよう、支援しています。

Logz.io クラウドSIEMは、高速クエリ、多次元検出、カスタマイズ可能で詳細なセキュリティコンテンツを通じて、データ過負荷や遍在するサイバースキルギャップなどの今日の主要なセキュリティ課題に直接対処し、データ量に関わらず、またパフォーマンスを低下させることなく、クラウド環境全体の監視と調査を支援します。

Logz.io ソリューションは、複雑さとコストを抑えながら、高度な脅威分析と調査を実現する目的で構築されました。ノイズが多いデータを減らすことを目的として備えられた専任のセキュリティアナリスト、サービスとしての脅威コンテンツ、AIベースの機能を活用することによって、チームは現実の脅威に迅速に対処するための有益な情報に集中できます。

AWS AppFabric 監査ログの取り込みに関する考慮事項

以下のセクションでは、Logz.io で使用する AppFabric 出力スキーマ、出力形式、出力先について説明します。

スキーマと形式

Logz.io は、以下の AppFabric 出力スキーマと形式をサポートしています。

- Raw - JSON
  - AppFabric は、ソースアプリケーションが使用していた元のスキーマの出力データを JSON 形式で出力します。
- OCSF - JSON
  - AppFabric はオープンサイバーセキュリティスキーマフレームワーク (OCSF) を使用してデータを正規化し、データを JSON 形式で出力します。

出力場所

Logz.io は、以下の AppFabric 出力場所をサポートしています。

- Amazon Data Firehose
  - Firehose 配信ストリームを設定して にデータを送信するにはLogz.io、「Amazon Data Firehose デベロッパーガイドLogz.io」の「送信先の選択」の手順に従います。

- Amazon Simple Storage Service (Amazon S3)
  - 監査ログを含む Amazon S3 バケットからデータを受信するように Logz.ioを設定するには、Logz.io ウェブサイトの「Amazon S3 バケットの設定」の手順に従ってください。

Netskope

サイバーセキュリティの世界的リーダーであるNetskopeは、組織がゼロトラストの原則を適用してデータを保護できるように、クラウド、データ、ネットワークのセキュリティを再定義しています。Netskope高速で使いやすいこのプラットフォームは、どこにいても、人、デバイス、データに最適なアクセスとゼロトラストセキュリティを提供します。 Netskopeはクラウド、ウェブ、プライベートアプリケーションのアクティビティにおいて、リスクの軽減、パフォーマンスの向上、比類のない可視化を実現します。進化する脅威、新たなリスク、テクノロジーの変化、組織やネットワークの変化、新しい規制要件に対応するために、フォーチュン100社のうち25社以上を含む数千のお客様が、Netskopeとその強力なNewEdgeネットワークを信頼しています。お客様が SASE ジャーニーでどのような状況にも対応できるようにNetskope が支援する方法については、netskope.comをご覧ください。

AWS AppFabric 監査ログの取り込みに関する考慮事項

以下のセクションでは、Netskope で使用する AppFabric 出力スキーマ、出力形式、出力先について説明します。

スキーマと形式

Netskope は、以下の AppFabric 出力スキーマと形式をサポートしています。

- Raw - JSON
  - AppFabric は、ソースアプリケーションが使用していた元のスキーマの出力データを JSON 形式で出力します。
- OCSF - JSON
  - AppFabric はオープンサイバーセキュリティスキーマフレームワーク (OCSF) を使用してデータを正規化し、データを JSON 形式で出力します。

出力場所

Netskope は、以下の AppFabric 出力場所をサポートしています。

- Amazon Simple Storage Service (Amazon S3)

- 監査ログを含む Amazon S3 Netskope バケットからデータを受信するように設定するには、Netskopeウェブサイトの「Amazon Web Services S3 のデータ保護」の指示に従ってください。

NetWitness

NetWitness は、Extended Detection and Response (XDR) ソフトウェアの大手開発業者です。セキュリティ意識の高い同社のグローバル顧客層は、巧妙で攻撃的な攻撃者に対する防御に NetWitness XDR を活用しています。デジタル攻撃を検知、調査、対応するための業界で最も完全で統合され、成熟したプラットフォームを備えた NetWitness XDR は、最新かつ効果的な SOC の統一基盤となっています。

高度にモジュール化されたアーキテクチャにより、NetWitness XDR はクラウド、オンプレミス、モバイルワーカーやリモートワーカーなど、発生場所や相手を問わず脅威を検出します。NetWitness Platform XDR は、適用された脅威インテリジェンスとユーザー行動分析を組み合わせて完全な可視性を提供し、脅威の検出、アクティビティの優先順位付け、調査、対応の自動化を可能にします。これらすべてに基づき、セキュリティアナリストはより優れた、より迅速な効率性をもって、ビジネスに影響を及ぼす脅威の先手を打つセキュリティ運用を実行できます。

AWS AppFabric 監査ログの取り込みに関する考慮事項

以下のセクションでは、NetWitness で使用する AppFabric 出力スキーマ、出力形式、出力先について説明します。

スキーマと形式

NetWitness は、以下の AppFabric 出力スキーマと形式をサポートしています。

- Raw - JSON
  - AppFabric は、ソースアプリケーションが使用していた元のスキーマの出力データを JSON 形式で出力します。
- OCSF - JSON
  - AppFabric はオープンサイバーセキュリティスキーマフレームワーク (OCSF) を使用してデータを正規化し、データを JSON 形式で出力します。

出力場所

NetWitness は、以下の AppFabric 出力場所をサポートしています。

- Amazon Simple Storage Service (Amazon S3)
  - 監査ログを含む Amazon S3 バケットからデータを受信するように NetWitness を設定するには、NetWitness ウェブサイトのNetWitnessプラットフォーム統合 ページにある S3 ユニバーサルコネクタイベントソースログ設定ガイドの指示に従ってください。

Amazon QuickSight

ハイパースケールの統合ビジネスインテリジェンス (BI) を備えたAmazon QuickSightは、データ主導型の組織を強化します。QuickSight を使用すると、最新のインタラクティブダッシュボード、ページ分割レポート、埋め込み分析、自然言語クエリを通じて、すべてのユーザーが同一の信頼できる情報源を使ってさまざまな分析ニーズに対応できるようになります。 AWS AppFabric for セキュリティログがソースとして保存されている Amazon Simple Storage Service (Amazon S3) バケットを選択することで、QuickSight で AppFabric 監査ログデータを分析できます。

AppFabric 監査ログの取り込みに関する考慮事項

以下のセクションでは、QuickSight で使用する AppFabric 出力スキーマ、出力形式、出力先について説明します。

スキーマと形式

QuickSightは、以下の AppFabric 出力スキーマと形式をサポートしています。

- Raw - JSON
  - AppFabric は、ソースアプリケーションが使用していた元のスキーマの出力データを JSON 形式で出力します。
- OCSF - JSON
  - AppFabric はオープンサイバーセキュリティスキーマフレームワーク (OCSF) を使用してデータを正規化し、データを JSON 形式で出力します。

出力場所

QuickSightは、以下の AppFabric 出力場所をサポートしています。

- Amazon S3
  - Amazon S3 ファイルを使用してデータセットを作成することで、Amazon S3 から QuickSight にデータを直接取り込むことができます。ターゲットファイルセットが QuickSight データ

ソースクォータを超えないことを確認するには、QuickSight ユーザーガイドの「データソースクォータ」を参照してください。

- ファイルセットが Amazon S3 データソースの QuickSight クォータを超える場合は、Amazon Athena と AWS Glue テーブルを使用して Amazon S3 にデータを取り込むことができます。QuickSight データセットで Athena を使用すると、追加コストが発生します。Athena料金の詳細については、Athena料金表を参照ください。

  Athena を使用するには:

  1. Athena ユーザーガイド の「AWS Glue を使用してAmazon S3 のデータソースに接続する」の指示に従ってください。
  2. QuickSight ユーザーガイドの「Athena データを使用したデータセットの作成」の手順に従います。

Rapid7

Rapid7, Inc. の使命は、よりシンプルで利用しやすいサイバーセキュリティを提供して安全なデジタル世界を創造することです。Rapid7 は最高クラスのテクノロジー、最先端の研究、幅広い戦略的専門知識でセキュリティ専門家が最新のアタックサーフェスを管理できるよう支援します。Rapid7 の包括的なセキュリティソリューションは、世界の 10,000 社を超えるお客様がクラウドリスク管理と脅威検出を統合してアタックサーフェスを減らし、脅威を迅速かつ正確に排除できるようサポートしています。

AWS AppFabric 監査ログの取り込みに関する考慮事項

以下のセクションでは、Rapid7 で使用する AppFabric 出力スキーマ、出力形式、および出力先について説明します。

スキーマと形式

Rapid7 は、以下の AppFabric 出力スキーマと形式をサポートしています。

- Raw - JSON
  - AppFabric は、ソースアプリケーションが使用していた元のスキーマの出力データを JSON 形式で出力します。
- OCSF - JSON
  - AppFabric はオープンサイバーセキュリティスキーマフレームワーク (OCSF) を使用してデータを正規化し、データを JSON 形式で出力します。

出力場所

Rapid7 は、以下の AppFabric 出力場所をサポートしています。

- Amazon Simple Storage Service (Amazon S3)
  - 監査ログを含む Amazon S3 バケットからデータを受信するように Rapid7 を設定するには、Rapid7 ブログウェブサイトの投稿「How to Monitor Your Amazon S3 Activity with InsightIDR」の手順に従ってください。

Amazon Security Lake

Amazon Security Lake は、 AWS 環境、Software as a Service (SaaS) プロバイダー、オンプレミス、クラウドソースのセキュリティデータを、 に保存されている専用のデータレイクに自動的に一元化します AWS アカウント。Security Lake を使用すると、組織全体のセキュリティデータをより完全に把握できます。Security Lake は、オープンソースのセキュリティイベントスキーマであるオープンサイバーセキュリティスキーマフレームワーク (OCSF) を採用しています。OCSF サポートにより、このサービスは のセキュリティデータと幅広いエンタープライズセキュリティデータソースを正規化 AWS し、組み合わせます。

AppFabric 監査ログの取り込みに関する考慮事項

Security Lake にカスタムソースを追加 AWS アカウント することで、 の Amazon Security Lake に SaaS 監査ログを取得できます。以下のセクションでは、Security Lake で使用する AppFabric 出力スキーマ、出力形式、および出力先について説明します。

スキーマと形式

Security Lake は以下の AppFabric 出力スキーマと形式をサポートしています。

- OCSF - JSON
  - AppFabric はオープンサイバーセキュリティスキーマフレームワーク (OCSF) を使用してデータを正規化し、データを JSON 形式で出力します。

出力場所

Security Lake は、AppFabric 取り込み出力場所として Amazon Data Firehose 配信ストリームを使用してAppFabric をカスタムソースとしてサポートします。 AWS Glue テーブルと Firehose 配信ストリームを設定し、Security Lake でカスタムソースを設定するには、次の手順を使用します。

AWS Glue テーブルを作成する

1. Amazon Simple Storage Service (Amazon S3) に移動し、任意の名前を付けたバケットを作成します。
2. AWS Glue コンソールに移動します。
3. [データカタログ] の場合は、[テーブル] セクションに移動して [テーブルの追加] を選択します。
4. このテーブルに任意の名前を付けます。
5. ステップ 1 で作成した Amazon S3 バケットを選択します。
6. データ形式には [JSON] を選択し、[次へ] を選択します。
7. [スキーマの選択または定義] ページで、[スキーマを JSON として編集] を選択します。
8. 次のスキーマを入力し、 AWS Glue テーブル作成プロセスを完了します。

```
[
    {
        "Name": "message",
        "Type": "string"
    },
    {
        "Name": "process",
        "Type":
 "struct<name:string,pid:int,user:struct<name:string,type:string,domain:string,uid:string,t
    },
    {
        "Name": "status",
        "Type": "string"
    },
    {
        "Name": "time",
        "Type": "bigint"
    },
    {
        "Name": "device",
        "Type":
 "struct<name:string,owner:struct<name:string,type:string,uid:string,type_id:int,risk_level
    },
    {
        "Name": "metadata",
        "Type":
 "struct<version:string,product:struct<name:string,version:string,uid:string,data_classific
    },
```

```
    {
        "Name": "severity",
        "Type": "string"
    },
    {
        "Name": "duration",
        "Type": "int"
    },
    {
        "Name": "type_name",
        "Type": "string"
    },
    {
        "Name": "activity_id",
        "Type": "int"
    },
    {
        "Name": "type_uid",
        "Type": "int"
    },
    {
        "Name": "observables",
        "Type": "array<struct<name:string,type:string,type_id:int,value:string>>"
    },
    {
        "Name": "category_name",
        "Type": "string"
    },
    {
        "Name": "class_uid",
        "Type": "int"
    },
    {
        "Name": "category_uid",
        "Type": "int"
    },
    {
        "Name": "class_name",
        "Type": "string"
    },
    {
        "Name": "timezone_offset",
        "Type": "int"
    },
```

```
    {
        "Name": "end_time",
        "Type": "bigint"
    },
    {
        "Name": "activity_name",
        "Type": "string"
    },
    {
        "Name": "cloud",
        "Type":
 "struct<account:struct<name:string,type:string,uid:string,type_id:int>,project_uid:string,
    },
    {
        "Name": "query_info",
        "Type": "struct<name:string,uid:string,query_string:string>"
    },
    {
        "Name": "query_result",
        "Type": "string"
    },
    {
        "Name": "query_result_id",
        "Type": "int"
    },
    {
        "Name": "severity_id",
        "Type": "int"
    },
    {
        "Name": "status_code",
        "Type": "string"
    },
    {
        "Name": "status_detail",
        "Type": "string"
    },
    {
        "Name": "status_id",
        "Type": "int"
    },
    {
        "Name": "network_interfaces",
```

```
        "Type":
"array<struct<name:string,type:string,hostname:string,mac:string,type_id:int,ip:string>>"
    },
    {
        "Name": "file",
        "Type":
"struct<attributes:int,name:string,type:string,path:string,type_id:int,accessor:struct<nam
    },
    {
        "Name": "actor",
        "Type":
"struct<process:struct<pid:int,file:struct<name:string,size:bigint,type:string,version:str
    },
    {
        "Name": "dst_endpoint",
        "Type":
"struct<owner:struct<name:string,type:string,uid:string,type_id:int,full_name:string,risk_
    },
    {
        "Name": "src_endpoint",
        "Type":
"struct<name:string,owner:struct<name:string,type:string,domain:string,uid:string,org:stru
    },
    {
        "Name": "user",
        "Type":
"struct<name:string,type:string,groups:array<struct<name:string,uid:string>>,type_id:int>"
    },
    {
        "Name": "resource",
        "Type":
"struct<version:string,uid:string,agent_list:array<struct<name:string,type:string,uid:stri
    },
    {
        "Name": "privileges",
        "Type": "array<string>"
    },
    {
        "Name": "action",
        "Type": "string"
    },
    {
        "Name": "action_id",
        "Type": "int"
```

```
    },
    {
        "Name": "protocol_ver",
        "Type": "string"
    },
    {
        "Name": "proxy",
        "Type":
"struct<name:string,port:int,type:string,ip:string,hostname:string,uid:string,type_id:int,
    },
    {
        "Name": "client_hassh",
        "Type":
"struct<algorithm:string,fingerprint:struct<value:string,algorithm:string,algorithm_id:int
    },
    {
        "Name": "authorizations",
        "Type": "array<string>"
    },
    {
        "Name": "proxy_tls",
        "Type":
"struct<version:string,certificate:struct<version:string,uid:string,subject:string,issuer:
    },
    {
        "Name": "load_balancer",
        "Type":
"struct<name:string,classification:string,dst_endpoint:struct<owner:struct<type:string,dom
    },
    {
        "Name": "disposition_id",
        "Type": "int"
    },
    {
        "Name": "disposition",
        "Type": "string"
    },
    {
        "Name": "proxy_traffic",
        "Type": "struct<bytes:bigint,packets:int>"
    },
    {
        "Name": "auth_type_id",
        "Type": "int"
```

```
    },
    {
        "Name": "proxy_http_response",
        "Type": "struct<code:int,message:string,status:string,length:int>"
    },
    {
        "Name": "server_hassh",
        "Type":
"struct<algorithm:string,fingerprint:struct<value:string,algorithm:string,algorithm_id:int
    },
    {
        "Name": "auth_type",
        "Type": "string"
    },
    {
        "Name": "firewall_rule",
        "Type": "struct<version:string,uid:string>"
    },
    {
        "Name": "proxy_connection_info",
        "Type":
"struct<direction:string,direction_id:int,protocol_num:int,protocol_ver:string>"
    },
    {
        "Name": "connection_info",
        "Type": "struct<direction:string,direction_id:int>"
    },
    {
        "Name": "api",
        "Type":
"struct<request:struct<data:string,uid:string>,response:struct<error:string,code:int,messa
    },
    {
        "Name": "attacks",
        "Type":
"array<struct<version:string,tactics:array<struct<name:string,uid:string>>,technique:struc
    },
    {
        "Name": "raw_data",
        "Type": "string"
    },
    {
        "Name": "email_uid",
        "Type": "string"
```

```
    },
    {
        "Name": "malware",
        "Type":
"array<struct<name:string,path:string,uid:string,classification_ids:array<int>,cves:array<
    },
    {
        "Name": "start_time_dt",
        "Type": "string"
    },
    {
        "Name": "direction",
        "Type": "string"
    },
    {
        "Name": "smtp_hello",
        "Type": "string"
    },
    {
        "Name": "unmapped",
        "Type": "string"
    },
    {
        "Name": "direction_id",
        "Type": "int"
    },
    {
        "Name": "email_auth",
        "Type":
"struct<spf:string,dkim:string,dkim_domain:string,dkim_signature:string,dmarc:string,dmarc
    },
    {
        "Name": "email",
        "Type":
"struct<uid:string,from:string,to:array<string>,data_classification:struct<category:string
    },
    {
        "Name": "impact_id",
        "Type": "int"
    },
    {
        "Name": "resources",
        "Type":
"array<struct<owner:struct<name:string,type:string,uid:string,type_id:int,full_name:string
```

```
    },
    {
        "Name": "finding_info",
        "Type":
"struct<title:string,uid:string,attacks:array<struct<version:string,tactics:array<struct<n
    },
    {
        "Name": "evidences",
        "Type":
"array<struct<process:struct<name:string,pid:int,file:struct<name:string,type:string,versi
    },
    {
        "Name": "impact",
        "Type": "string"
    },
    {
        "Name": "count",
        "Type": "int"
    },
    {
        "Name": "confidence_id",
        "Type": "int"
    },
    {
        "Name": "enrichments",
        "Type":
"array<struct<data:string,name:string,type:string,value:string,provider:string>>"
    },
    {
        "Name": "rcode",
        "Type": "string"
    },
    {
        "Name": "app_name",
        "Type": "string"
    },
    {
        "Name": "rcode_id",
        "Type": "int"
    },
    {
        "Name": "query",
        "Type":
"struct<type:string,hostname:string,class:string,opcode_id:int,packet_uid:int>"
```

```
    },
    {
        "Name": "proxy_endpoint",
        "Type":
"struct<name:string,owner:struct<name:string,type:string,domain:string,uid:string,groups:a
    },
    {
        "Name": "response_time",
        "Type": "bigint"
    },
    {
        "Name": "delay",
        "Type": "int"
    },
    {
        "Name": "start_time",
        "Type": "bigint"
    },
    {
        "Name": "proxy_http_request",
        "Type":
"struct<version:string,url:struct<port:int,scheme:string,path:string,hostname:string,query
    },
    {
        "Name": "version",
        "Type": "string"
    },
    {
        "Name": "stratum",
        "Type": "string"
    },
    {
        "Name": "stratum_id",
        "Type": "int"
    },
    {
        "Name": "dispersion",
        "Type": "int"
    },
    {
        "Name": "traffic",
        "Type":
"struct<bytes_out:int,chunks:bigint,bytes:int,packets:int,packets_in:bigint>"
    },
```

```
    {
        "Name": "precision",
        "Type": "int"
    },
    {
        "Name": "size",
        "Type": "int"
    },
    {
        "Name": "actual_permissions",
        "Type": "int"
    },
    {
        "Name": "base_address",
        "Type": "string"
    },
    {
        "Name": "requested_permissions",
        "Type": "int"
    },
    {
        "Name": "end_time_dt",
        "Type": "string"
    },
    {
        "Name": "compliance",
        "Type":
 "struct<control:string,status:string,standards:array<string>,status_id:int>"
    },
    {
        "Name": "remediation",
        "Type": "struct<desc:string>"
    },
    {
        "Name": "kb_article_list",
        "Type":
 "array<struct<os:struct<name:string,type:string,type_id:int,cpe_name:string,edition:string
    },
    {
        "Name": "peripheral_device",
        "Type":
 "struct<name:string,class:string,uid:string,model:string,serial_number:string,vendor_name:
    },
    {
```

```
        "Name": "time_dt",
        "Type": "string"
    },
    {
        "Name": "group",
        "Type": "struct<name:string,type:string,uid:string>"
    },
    {
        "Name": "users",
        "Type":
 "array<struct<name:string,type:string,uid:string,type_id:int,risk_level:string,risk_level_
    },
    {
        "Name": "confidence_score",
        "Type": "int"
    },
    {
        "Name": "state",
        "Type": "string"
    },
    {
        "Name": "state_id",
        "Type": "int"
    },
    {
        "Name": "evidence",
        "Type": "string"
    },
    {
        "Name": "confidence",
        "Type": "string"
    },
    {
        "Name": "risk_level",
        "Type": "string"
    },
    {
        "Name": "risk_score",
        "Type": "int"
    },
    {
        "Name": "impact_score",
        "Type": "int"
    },
```

```
    {
        "Name": "risk_level_id",
        "Type": "int"
    },
    {
        "Name": "finding",
        "Type":
"struct<title:string,uid:string,modified_time:bigint,modified_time_dt:string,first_seen_ti
    },
    {
        "Name": "user_result",
        "Type":
"struct<name:string,type:string,uid:string,type_id:int,account:struct<name:string,uid:stri
    },
    {
        "Name": "codes",
        "Type": "array<int>"
    },
    {
        "Name": "command",
        "Type": "string"
    },
    {
        "Name": "type",
        "Type": "string"
    },
    {
        "Name": "kernel",
        "Type": "struct<name:string,type:string,type_id:int>"
    },
    {
        "Name": "http_response",
        "Type":
"struct<code:int,status:string,http_headers:array<struct<name:string,value:string>>>"
    },
    {
        "Name": "http_request",
        "Type":
"struct<url:struct<scheme:string,path:string,hostname:string,query_string:string,category_
    },
    {
        "Name": "tls",
        "Type":
"struct<version:string,certificate:struct<subject:string,issuer:string,fingerprints:array<
```

```
    },
    {
        "Name": "web_resources",
        "Type":
"array<struct<name:string,type:string,data_classification:struct<category:string,category_
    },
    {
        "Name": "http_cookies",
        "Type":
"array<struct<name:string,value:string,is_http_only:boolean,is_secure:boolean,samesite:str
    },
    {
        "Name": "type_id",
        "Type": "int"
    },
    {
        "Name": "databucket",
        "Type":
"struct<name:string,type:string,file:struct<attributes:int,name:string,owner:struct<name:s
    },
    {
        "Name": "table",
        "Type": "struct<uid:string,created_time_dt:string>"
    },
    {
        "Name": "session",
        "Type":
"struct<count:int,uid:string,uuid:string,issuer:string,created_time:bigint,is_remote:boole
    },
    {
        "Name": "certificate",
        "Type":
"struct<version:string,uid:string,subject:string,issuer:string,fingerprints:array<struct<v
    },
    {
        "Name": "is_mfa",
        "Type": "boolean"
    },
    {
        "Name": "logon_type_id",
        "Type": "int"
    },
    {
        "Name": "auth_protocol_id",
```

```
        "Type": "int"
    },
    {
        "Name": "logon_type",
        "Type": "string"
    },
    {
        "Name": "is_remote",
        "Type": "boolean"
    },
    {
        "Name": "is_cleartext",
        "Type": "boolean"
    },
    {
        "Name": "auth_protocol",
        "Type": "string"
    },
    {
        "Name": "is_renewal",
        "Type": "boolean"
    },
    {
        "Name": "lease_dur",
        "Type": "int"
    },
    {
        "Name": "relay",
        "Type":
 "struct<name:string,type:string,ip:string,mac:string,namespace:string,type_id:int>"
    },
    {
        "Name": "transaction_uid",
        "Type": "string"
    },
    {
        "Name": "file_result",
        "Type":
 "struct<name:string,size:int,type:string,path:string,desc:string,product:struct<name:strin
    },
    {
        "Name": "file_diff",
        "Type": "string"
    },
```

```
    {
        "Name": "create_mask",
        "Type": "string"
    },
    {
        "Name": "web_resources_result",
        "Type":
 "array<struct<type:string,data_classification:struct<category:string,category_id:int,confi
    },
    {
        "Name": "app",
        "Type":
 "struct<name:string,version:string,uid:string,data_classification:struct<category:string,c
    },
    {
        "Name": "src_url",
        "Type": "string"
    },
    {
        "Name": "priority_id",
        "Type": "int"
    },
    {
        "Name": "verdict",
        "Type": "string"
    },
    {
        "Name": "desc",
        "Type": "string"
    },
    {
        "Name": "verdict_id",
        "Type": "int"
    },
    {
        "Name": "priority",
        "Type": "string"
    },
    {
        "Name": "finding_info_list",
        "Type":
 "array<struct<title:string,uid:string,attacks:array<struct<version:string,tactics:array<st
    },
    {
```

```
        "Name": "expiration_time_dt",
        "Type": "string"
    },
    {
        "Name": "expiration_time",
        "Type": "bigint"
    },
    {
        "Name": "comment",
        "Type": "string"
    },
    {
        "Name": "entity",
        "Type": "struct<data:string,name:string,version:string,uid:string>"
    },
    {
        "Name": "entity_result",
        "Type":
"struct<data:string,name:string,type:string,version:string,uid:string>"
    },
    {
        "Name": "module",
        "Type":
"struct<type:string,file:struct<name:string,type:string,path:string,desc:string,type_id:in
    },
    {
        "Name": "exit_code",
        "Type": "int"
    },
    {
        "Name": "injection_type",
        "Type": "string"
    },
    {
        "Name": "injection_type_id",
        "Type": "int"
    },
    {
        "Name": "request",
        "Type": "struct<uid:string>"
    },
    {
        "Name": "response",
        "Type": "struct<error:string,code:int,message:string,error_message:string>"
```

```
    },
    {
        "Name": "driver",
        "Type":
"struct<file:struct<name:string,type:string,version:string,path:string,type_id:int,parent_
    },
    {
        "Name": "prev_security_states",
        "Type": "array<string>"
    },
    {
        "Name": "security_states",
        "Type": "array<string>"
    },
    {
        "Name": "folder",
        "Type":
"struct<name:string,type:string,path:string,desc:string,type_id:int,mime_type:string,paren
    },
    {
        "Name": "url",
        "Type":
"struct<port:int,scheme:string,path:string,hostname:string,query_string:string,resource_ty
    },
    {
        "Name": "tunnel_type_id",
        "Type": "int"
    },
    {
        "Name": "tunnel_type",
        "Type": "string"
    },
    {
        "Name": "protocol_name",
        "Type": "string"
    },
    {
        "Name": "job",
        "Type":
"struct<name:string,file:struct<name:string,type:string,path:string,signature:struct<certi
    },
    {
        "Name": "num_trusted_items",
        "Type": "int"
```

```
    },
    {
        "Name": "command_uid",
        "Type": "string"
    },
    {
        "Name": "num_registry_items",
        "Type": "int"
    },
    {
        "Name": "num_network_items",
        "Type": "int"
    },
    {
        "Name": "schedule_uid",
        "Type": "string"
    },
    {
        "Name": "num_resolutions",
        "Type": "int"
    },
    {
        "Name": "scan",
        "Type": "struct<name:string,type:string,type_id:int>"
    },
    {
        "Name": "num_detections",
        "Type": "int"
    },
    {
        "Name": "num_processes",
        "Type": "int"
    },
    {
        "Name": "num_files",
        "Type": "int"
    },
    {
        "Name": "total",
        "Type": "int"
    },
    {
        "Name": "num_folders",
        "Type": "int"
```

```
    },
    {
        "Name": "dce_rpc",
        "Type":
 "struct<command:string,flags:array<string>,command_response:string,opnum:int,rpc_interface
    },
    {
        "Name": "share",
        "Type": "string"
    },
    {
        "Name": "client_dialects",
        "Type": "array<string>"
    },
    {
        "Name": "open_type",
        "Type": "string"
    },
    {
        "Name": "tree_uid",
        "Type": "string"
    },
    {
        "Name": "share_type_id",
        "Type": "int"
    },
    {
        "Name": "share_type",
        "Type": "string"
    },
    {
        "Name": "dialect",
        "Type": "string"
    },
    {
        "Name": "cis_benchmark_result",
        "Type": "struct<name:string>"
    },
    {
        "Name": "vulnerabilities",
        "Type":
 "array<struct<references:array<string>,severity:string,affected_packages:array<struct<name
    },
    {
```

```
        "Name": "service",
        "Type": "struct<name:string,uid:string>"
    },
    {
        "Name": "data_security",
        "Type":
 "struct<category:string,pattern_match:string,category_id:int,confidentiality:string,confid
    },
    {
        "Name": "database",
        "Type":
 "struct<name:string,type:string,uid:string,type_id:int,data_classification:struct<category
    }
]
```

Security Lake でカスタムソースを作成する

1. Amazon Security Lake コンソールに移動します。
2. ナビゲーションペインで [カスタムソース] を選択します。
3. [カスタムソースの作成] を選択します。
4. カスタムソースの名前を入力し、適用可能な OCSF イベントクラスを選択します。

> **Note**
>
> AppFabric は、[アカウントの変更] 、[認証] 、[ユーザーアクセス管理] 、[グループ管理] 、[ウェブリソースアクティビティ] 、[ウェブリソースアクセスアクティビティ] の各イベントクラスを使用します。

5. AWS アカウント ID と外部 ID の両方に、 AWS アカウント ID を入力します。続いて、[作成] を選択します。
6. カスタムソースの Amazon S3 の場所を保存します。これを使用して Amazon Data Firehose 配信ストリームを設定します。

Firehose で配信ストリームを作成する

1. Amazon Data Firehose コンソールに移動します。
2. [配信ストリームの作成] を選択します。
3. [ソース] には [Direct PUT] を選択します。

4. [宛先] には、[S3] を選択します。
5. [レコードを変換および転換] セクションで、[レコード形式の変換を有効にする] を選択し、出力形式として Apache Parquet を選択します。
6. AWS Glue テーブルで、前の手順で作成した AWS Glue テーブルを選択し、最新バージョンを選択します。
7. [送信先の設定] には、 Security Lake カスタムソースで作成した Amazon S3 バケットを選択します。
8. [動的パーティショニング] には [有効] を選択します。
9. [JSON のインライン解析] には [有効] を選択します。
   - [キー名] には、`eventDayValue` と入力します。
   - [JQ 式] には、`(.time/1000)|strftime("%Y%m%d")` と入力します。
10. [S3 バケットプレフィックス] には、以下の値を入力します。

```
ext/<custom source name>/region=<region>/accountId=<account_id>/eventDay=!
{partitionKeyFromQuery:eventDayValue}/
```

   <custom source name>、*<region>*、*<account_id>* を Security Lake のカスタムソース名 AWS リージョン と AWS アカウント ID に置き換えます。

11. [S3 バケットエラー出力プレフィックス] には、以下の値を入力します。

```
ext/AppFabric/error/
```

12. [再試行時間] には [300] を選択します。
13. [バッファサイズ] には [128 MiB] を選択します。
14. [バッファ間隔] には [60秒] を選択します。
15. Firehose 配信ストリームの作成プロセスを完了します。

AppFabric 取り込みの作成

Amazon Security Lake にデータを送信するには、前に作成した Firehose 配信ストリームを出力場所として使用する取り込みを AppFabric コンソールで作成する必要があります。Firehose を出力場所として使用するように AppFabric 取り込みを設定する方法の詳細については、「出力場所の作成」を参照してください。

Singularity Cloud

Singularity Cloud プラットフォームは、あらゆる段階で、あらゆるカテゴリの脅威から企業を保護します。その特許取得済みの AI (人工知能) は、既知の署名やパターンから、ゼロデイ攻撃やランサムウェアなどの最も高度な攻撃にセキュリティを拡張します。

AWS AppFabric 監査ログの取り込みに関する考慮事項

以下のセクションでは、Singularity Cloud で使用する AppFabric 出力スキーマ、出力形式、出力先について説明します。

スキーマと形式

Singularity Cloud は、以下の AppFabric 出力スキーマと形式をサポートしています。

OCSF - JSON: AppFabric はオープンサイバーセキュリティスキーマフレームワーク (OCSF) を使用してデータを正規化し、データを JSON 形式で出力します。

出力場所

Singularity Cloud は、次の AppFabric 出力場所からの監査ログの受信をサポートします。

- Amazon Simple Storage Service (Amazon S3)
  - 監査ログを含む Amazon S3 バケットからデータを受信するSingularity Cloudように を設定するには、 Singularity Cloud’sドキュメントの指示に従います。

Splunk

Splunkは組織のレジリエンスを高めるのに役立ちます。主要組織は、Splunkのセキュリティとオブザーバビリティの統合プラットフォームを使用して、デジタルシステムの安全性と信頼性を維持しています。セキュリティ、インフラストラクチャ、アプリケーションの問題が重大なインシデントになるのを防ぎ、デジタルの混乱による衝撃を吸収し、デジタルトランスフォーメーションを加速するために、組織は Splunk を信頼しています。

AWS AppFabric 監査ログの取り込みに関する考慮事項

以下のセクションでは、Splunk で使用する AppFabric 出力スキーマ、出力形式、出力先について説明します。

スキーマと形式

Splunk は、以下の AppFabric 出力スキーマと形式をサポートしています。

- Raw - JSON
  - AppFabric は、ソースアプリケーションが使用していた元のスキーマの出力データを JSON 形式で出力します。
- OCSF - JSON
  - AppFabric はオープンサイバーセキュリティスキーマフレームワーク (OCSF) を使用してデータを正規化し、データを JSON 形式で出力します。
- OCSF-Parquet
  - AppFabric はオープンサイバーセキュリティスキーマフレームワーク (OCSF) を使用してデータを正規化し、データを Apache Parquet 形式で出力します。

出力場所

Splunk は、以下の AppFabric 出力場所をサポートしています。

- Amazon Data Firehose
  - 監査ログを含む Firehose ストリームから監査ログを受信するSplunkように を設定するには、SplunkウェブサイトのSplunk「Amazon Data Firehose のアドオン」の指示に従います。
- Amazon Simple Storage Service (Amazon S3)
  - 監査ログを含む Amazon S3 バケットからデータを受信するようにSplunkを設定するには、Splunk ウェブサイトの「AWS用 Splunk アドオンの SQS ベースの S3 入力の設定」の手順に従ってください。

セキュリティリソースの Delete AWS AppFabric

AWS AppFabric for security を引き続き使用しない場合は、追加料金が発生しないように、セットアップ時に作成した出力場所のデータと AppFabric for security リソースを必ず削除してください。AppFabric リソースをクリーンアップするには、Software as a Service (SaaS) アプリケーションごとにリソースを作成した逆の順序でリソースを削除する必要があります: Ingestion destinations > Ingestions > App authorization > App Bundles

最後のアプリ認可を削除した後、アプリバンドルを削除できます。

トピック

- 取り込み先の削除
- 取り込みの削除
- アプリ認可の削除
- アプリバンドルの削除

取り込み先の削除

取り込みの作成時に出力場所を選択した場合、 AppFabric for security がユーザーに代わって取り込み先を作成します。取り込み先を削除するには、以下の手順に従います。

1. https://console.aws.amazon.com/appfabric/ にある AppFabric コンソールを開きます。
2. [はじめに] ページで、左側のメニューを展開します。
3. [取り込み] を選択します。
4. アプリ認可を選択します。
5. 削除する取り込み先の横にあるオプションボタンを選択し、[削除] を選択します。
6. 取り込み先ダイアログボックスで、[削除] を選択して確定します。
7. すべての取り込み先で上記の手順を繰り返します。

取り込みの削除

取り込みを削除するには、次の手順に従います。

1. [はじめに] ページで、左側のメニューを展開します。
2. [取り込み] を選択します。
3. アプリ認可の横にあるオプションボタンを選択します。
4. [アクション] ドロップダウンメニューを選択します。
5. [削除] を選択します。
6. 取り込みの削除ダイアログボックスで、[削除] を選択して確定します。

アプリ認可の削除

アプリ認可を削除するには、次の手順に従います。

1. [はじめに] ページで、左側のメニューを展開します。

2. [アプリ認証] を選択します。
3. 削除したいアプリ認可の横にあるオプションボタンを選択します。
4. [アクション] ドロップダウンメニューを選択します。
5. [削除] を選択します。
6. 取り込みの削除ダイアログボックスで、[削除] を選択して確定します。

アプリバンドルの削除

アプリバンドルを削除するには、次の手順に従います。

1. [はじめに] ページで、左側のメニューを展開します。
2. [アプリバンドル] を選択します。
3. [削除] ボタンを選択します。
4. [delete] と入力して確定し、[削除] を選択します。

AWS AppFabric for productivity とは

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

> **Note**
>
> Amazon Bedrock を利用: 自動不正検出 AWS を実装します。 AWS AppFabric for productivity は Amazon Bedrock 上に構築されているため、ユーザーは Amazon Bedrock に実装されているコントロールを継承して、AI の安全性、セキュリティ、責任ある使用を強化します。

AWS AppFabric for productivity (プレビュー) は、複数のアプリケーションからコンテキストを使用してインサイトとアクションを生成することで、サードパーティーアプリケーションのエンドユーザーの生産性を再考するのに役立ちます。アプリケーションデベロッパーは、他のアプリケーションのユーザーデータにアクセスすることがアプリケーションの生産性を高めるうえで重要であることを認識していますが、各アプリケーションとの統合を構築して管理することは望んでいません。AppFabric for productivity を使用すると、アプリケーションデベロッパーは、クロスアプリケーションのデータインサイトやアクションを生成する、生成 AI を活用した API を使って、新規または既存の生成 AI アシスタントを通じてより充実したエンドユーザーエクスペリエンスを実現できます。AppFabric for productivity によって複数のアプリケーションのデータが統合されるため、デベロッパーは 2 地点間統合を構築または管理する必要がなくなります。アプリケーションデベロッパーは、AppFabric for productivity を自社のアプリケーションの UI に直接埋め込むことができ、エンドユーザーに一貫したエクスペリエンスを提供すると同時に、他のアプリケーションの関連するコンテキストを表示することができます。

AppFabric for productivity は、Asana、Atlassian Jira Suite、Google Workspace、Microsoft 365、Miro、Slack、Smartsheet など、一般によく利用されているアプリケーションのデータを接続します。AppFabric for productivity を使用すれば、アプリデベロッパーは、ユーザーの採用率、満足度、ロイヤルティを高めるよりパーソナライズされたアプリケーション体験を、これまでよりも簡単に構築できるようになります。一方でエンドユーザーは、さまざまなアプリケーションを横断して、必要とするインサイトに作業の流れを止めることなくアクセスできるようになります。

トピック

- 利点
- ユースケース

- AppFabric for productivity へのアクセス
- アプリケーション開発者向けの AppFabric for productivity (プレビュー) の使用を開始する
- エンドユーザー向けの AppFabric for productivity (プレビュー) の使用を開始する
- AppFabric for productivity APIsプレビュー)
- AppFabric でのデータ処理

利点

AppFabric for productivity を使用すると、アプリケーションデベロッパーは、クロスアプリケーションのデータインサイトやアクションを生成する API を使って、新規または既存の生成 AI アシスタントを通じてより充実したエンドユーザーエクスペリエンスを実現できます。

- クロスアプリケーションのユーザーデータの単一ソース: AppFabric for productivity によって複数のアプリケーションのデータが統合されるため、デベロッパーは 2 地点間統合を構築または管理する必要がなくなります。SaaS アプリケーションのデータは、異質なデータタイプを、どのアプリケーションにも理解可能な形式に自動的に正規化することで、他のアプリケーションで使用できるように処理されます。これにより、アプリケーションのデベロッパーはより多くのデータを組み込んでエンドユーザーの生産性を高めることができます。
- ユーザーエクスペリエンスを完全にコントロール: デベロッパーは、AppFabric for productivity を自社のアプリケーションの UI に直接埋め込むことで、ユーザーエクスペリエンスを完全にコントロールしながら、アプリケーション全体のコンテキストに基づいてパーソナライズされたインサイトや推奨されるアクションをエンドユーザーに提供することができます。エンドユーザーは、好みの SaaS アプリケーションや、タスク実行のために選択したアプリケーションから、AppFabric for productivity を利用できます。それにより、アプリケーションを切り替える手間を省いて時間を節約し、作業の流れを常に把握することが可能になります。
- 市場化にかかる時間を短縮: アプリケーションデベロッパーは、生成されたユーザーデータに関するユーザーレベルのインサイトを 1 度の API コールで取得できます。モデルを微調整したり、カスタムプロンプトを作成したり、複数のアプリケーションを横断して統合を構築したりする必要がありません。AppFabric はこうした複雑さを取り除き、アプリケーションデベロッパーによる生成 AI 機能のよりスピーディな構築、組み込み、強化を可能にします。それによりアプリケーションデベロッパーは、最も重要なタスクにリソースを集中させることが可能になります。
- アーティファクトのリファレンスによりユーザーの信頼を構築: AppFabric for productivity は、アウトプットの一部として、インサイトの生成に使用した関連するアーティファクトまたはソースファイルを表示し、LLM アウトプットに対するエンドユーザーの信頼を構築します。

- ユーザー権限をシンプル化: インサイトの生成に使用したユーザーアーティファクトは、ユーザーがアクセスできるものに基づきます。AppFabric for productivity は、ISVの許可とアクセス制御を信頼できるソースとして使用します。

ユースケース

アプリケーションデベロッパーは、AppFabric for productivity を使用することでアプリケーションの生産性を刷新できます。AppFabric for productivity には、エンドユーザーの生産性向上に役立つ、以下のユースケースに焦点を絞った 2 つの API が用意されています。

- 1 日の作業の優先付け
  - 実行可能なインサイト API は、アプリケーション (E メール、カレンダー、メッセージ、タスクなど) を横断してタイムリーにインサイトを表示することで、1 日の業務を効率よく管理することを可能にする API です。さらにユーザーは、メール作成、会議のスケジュール設定、アクションアイテムの作成といったアプリケーションを横断したアクションを、自分が選んだアプリケーションから実行することができます。例えば、夜間にカスタマーエスカレーションを受けた従業員は、夜間の会話の要旨を確認できるだけでなく、推奨されるアクションを確認して、その顧客のアカウントマネージャーとのミーティングを設定することもできます。アクションの必須フィールド (タスク名、所有者、メールの送信者/受信者など) は自動入力され、入力された内容はアクションの実行前に編集することが可能です。
- 次回のミーティングの準備
  - 会議準備の API は、会議の内容を要約したり、E メールやメッセージその他のアプリケーションを横断して関連性の高いアーティファクトを表示したりすることで、会議に向けた準備に役立つ API です。ユーザーはすぐに会議に向けた準備ができ、コンテンツを見つけるためにアプリケーション間を切り替える手間が省けます。

AppFabric for productivity へのアクセス

AppFabric for Productivity は、現在プレビュー版でリリースされており、米国東部 (バージニア北部) AWS リージョンで利用できます。詳細については AWS リージョン、『』のAWS AppFabric エンドポイントとクォータ」を参照してくださいAWS 全般のリファレンス。

各リージョンでは、次のいずれかの方法で AppFabric for productivity にアクセスできます。

- アプリケーションデベロッパーとして
  - アプリケーション開発者向けの AppFabric for productivity (プレビュー) の使用を開始する

- エンドユーザーとして
  - エンドユーザー向けの AppFabric for productivity (プレビュー) の使用を開始する

アプリケーション開発者向けの AppFabric for productivity (プレビュー) の使用を開始する

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

このセクションでは、アプリケーションデベロッパーが AWS AppFabric for productivity (プレビュー) をアプリケーションに統合するのに役立ちます。 AWS AppFabric for productivity を使用すると、デベロッパーは、複数のアプリケーション間で E メール、カレンダーイベント、タスク、メッセージなどから AI を活用したインサイトやアクションを生成することで、ユーザーにとってよりリッチなアプリケーションエクスペリエンスを構築できます。サポートされているアプリケーションのリストは、「AWS AppFabric Supported Applications」を参照してください。

AppFabric for Productivity は、アプリケーションデベロッパーに、安全で統制のとれた環境で構築や実験へのアクセスを提供します。AppFabric for productivity の使用を開始するときは、まず AppClient を作成し、テストユーザーを 1 人登録します。この方法は、ユーザーが、アプリケーションと AppFabric 間の認証およびコミュニケーションの流れを理解しテストすることができるようにすることを目的としています。1 人のユーザーでテストしたら、アプリケーションを AppFabric に送信して検証を行い、その後、アクセスを他のユーザーに広げます (「ステップ 5. AppFabric にアプリケーションの検証をリクエストする」を参照)。アプリケーションデベロッパーとエンドユーザーを保護し彼らのデータを保護するため、AppFabric では、アプリケーションの情報を検証してから採用範囲を拡大できるようにしています。そうすることで、責任ある方法でユーザーによる利用を広げるための準備をします。

トピック

- 前提条件
- ステップ 1. AppFabric for productivity の AppClient を作成する
- ステップ 2. アプリケーションを認証し認可する
- ステップ 3. AppFabric ユーザーポータル URL をアプリケーションに追加する
- ステップ 4. AppFabric を使用してクロスアプリケーションのインサイトとアクションを表示する
- ステップ 5. AppFabric にアプリケーションの検証をリクエストする

- AppFabric for productivity AppClients の管理
- AppFabric for productivity の AppClients AppClients のトラブルシューティング

前提条件

開始する前に、 を作成する必要があります AWS アカウント。詳細については、「にサインアップする AWS アカウント」を参照してください。また、以下の "appfabric:CreateAppClient" IAM ポリシーにアクセスできるユーザーを 1 人以上作成する必要があります。これにより、ユーザーはアプリケーションを AppFabric に登録することができます。AppFabric for productivity の機能へのアクセス権限を付与する方法の詳細については、「AppFabric for productivity IAM ポリシーの例」を参照してください。管理ユーザーを作成しておくことは何かと便利ですが、初期設定ではその作成は必須ではありません。詳細については、「管理アクセスを持つユーザーを作成する」を参照してください。

AppFabric for productivity は、プレビュー中は米国東部 (バージニア北部) のみで利用できます。以下のステップを開始する前に、ご自身の現在地がこの地域であることを確認します。

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "appfabric:CreateAppClient"
            ],
            "Resource": ["arn:aws:appfabric:*:*:appclient/*"]
        }
    ],
    "Version": "2012-10-17"
}
```

ステップ 1. AppFabric for productivity の AppClient を作成する

AppFabric for productivity インサイトをアプリケーション内で表示できるようにするには、先に AppFabric の AppClient を作成しておく必要があります。AppClient とは、AppFabric for productivity のゲートウェイであり、アプリケーションと AppFabric 間の安全なコミュニケーションを可能にする、セキュアな OAuth アプリケーションクライアントとして機能します。AppClient を作成すると AppClient ID が付与されます。これは、AppFabric が、アプリケーションおよびユーザーの AWS アカウントと連携していることを認識できるようにするための、不可欠な一意の識別子です。

AppFabric for Productivity は、アプリケーションデベロッパーに、安全で統制のとれた環境で構築や実験へのアクセスを提供します。AppFabric for productivity の使用を開始するときは、まず AppClient を作成し、テストユーザーを 1 人登録します。この方法は、ユーザーが、アプリケーションと AppFabric 間の認証およびコミュニケーションの流れを理解しテストすることができるようにすることを目的としています。1 人のユーザーでテストしたら、アプリケーションを AppFabric に送信して検証を行い、その後、アクセスを他のユーザーに広げます (「ステップ 5. AppFabric にアプリケーションの検証をリクエストする」を参照)。アプリケーションデベロッパーとエンドユーザーを保護し彼らのデータを保護するため、AppFabric では、アプリケーションの情報を検証してから採用範囲を拡大できるようにしています。そうすることで、責任ある方法でユーザーによる利用を広げるための準備をします。

AppClient を作成するには、 AWS AppFabric `CreateAppClient` API オペレーションを使用します。AppClient を後で更新する必要がある場合は、`UpdateAppClient` API オペレーションを使用すれば redirectUrls を変更するだけで済みます。appName や説明などお使いの AppClient に関連付けられているその他のパラメータを変更する必要がある場合は、その AppClient を削除し、新たに作成する必要があります。詳細については、「CreateAppClient」を参照してください。

API を使用して、Python、Node.js、Java、C#、Go、Rust などの複数のプログラミング言語`CreateAppClient`を使用して、 AWS サービスにアプリケーションを登録できます。詳細については、「IAM ユーザーガイド」の「リクエスト署名の例」を参照してください。この API オペレーションを実行するには、アカウント署名バージョン 4 の認証情報を使用する必要があります。署名バージョン 4 の詳細については、IAM ユーザーガイドの AWS API リクエストの署名を参照してください。

リクエストフィールド

- `appName` - AppFabric ユーザーポータルの同意ページでユーザーに表示されるアプリケーションの名前です。同意ページでは、エンドユーザーは、アプリケーション内に AppFabric インサイトを表示する許可を求められます。同意ページの詳細については、「ステップ 2. インサイトを表示することをアプリに許可する」を参照してください 。
- `description` - アプリケーションの説明です。
- `redirectUrls` - 承認後にエンドユーザーがリダイレクトされる URI です。redirectUrls は 5 個まで追加できます。例えば、`https://localhost:8080`。
- `starterUserEmails` - アプリケーションが検証されるまでの間、インサイトを受け取るためのアクセスが許可されるユーザーの E メールアドレスです。使用できるアドレスは 1 つのみです。例: `anyuser@example.com`

- `customerManagedKeyIdentifier` (オプション) - データの暗号化に使用されるカスタマーマネージドキー (KMS が生成) の ARN です。指定しない場合、 AWS AppFabric マネージドキーが使用されます。 AWS 所有のキー およびカスタマーマネージドキーの詳細については、「AWS Key Management Service 開発者ガイド」の「カスタマーキーと AWS キー」を参照してください。

レスポンスフィールド

- `appClientArn` - AppClient ID を含む Amazon リソースネーム (ARN) です。例えば、AppClient ID は `a1b2c3d4-5678-90ab-cdef-EXAMPLE11111` です。
- `verificationStatus` - AppClient の検証ステータスです。
  - `pending_verification` - AppClient の検証が、AppFabric でまだ進行中です。AppClient の検証が終わるまでは、この AppClient を使用できるのは (`starterUserEmails` で指定された) 1 人のユーザーのみとなります。このユーザーは、AppFabric ユーザーポータル (ステップ 3. AppFabric ユーザーポータル URL をアプリケーションに追加する で紹介) で、アプリケーションが検証されていないことを示す通知を閲覧できます。
  - `verified` - AppFabric による検証プロセスが正常に完了し、AppClient は検証済みです。
  - `rejected` - AppClient の検証プロセスが AppFabric によって拒否されました。検証プロセスが再開され正常に完了するまでは、他のユーザーが AppClient を使用することはできません。

```
curl --request POST \
  --header "Content-Type: application/json" \
  --header "X-Amz-Content-Sha256: <sha256_payload>" \
  --header "X-Amz-Security-Token: <security_token>" \
  --header "X-Amz-Date: 20230922T172215Z" \
  --header "Authorization: AWS4-HMAC-SHA256 ..." \
  --url https://appfabric.<region>.amazonaws.com/appclients/ \
  --data '{
    "appName": "Test App",
    "description": "This is a test app",
    "redirectUrls": ["https://localhost:8080"],
    "starterUserEmails": ["anyuser@example.com"],
    "customerManagedKeyIdentifier": "arn:aws:kms:<region>:<account>:key/<key>"
}'
```

アクションが成功すると、サービスは HTTP 200 レスポンスを返します。

```
{
    "appClientConfigSummary": {
```

```
        "appClientArn": "arn:aws:appfabric:<region>:<account>:appclient/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "verificationStatus": "pending_verification"
    }
}
```

ステップ 2. アプリケーションを認証し認可する

OAuth 2.0 の認可フローを確立して、アプリケーションが AppFabric のインサイトを安全に統合できるようにします。まず認可コードを作成します。このコードがアプリケーションの ID を検証します。詳細については、「承認」を参照してください。次に、この認可コードをアクセストークンと交換します。これにより、アプリケーション内の AppFabric インサイトを取得して表示するアクセス権限がアプリケーションに付与されます。詳細については、「トークン」を参照してください。

アプリケーションを承認するアクセス権限を付与する方法の詳細については、「アプリケーションを承認するためのアクセスを許可する」を参照してください。

1. 認可コードを作成するには、 AWS AppFabric `oauth2/authorize` API オペレーションを使用します。

   リクエストフィールド

   - `app_client_id`(必須) - ステップ 1. AppClient の作成 で作成した AWS アカウント の AppClient ID です。AppClient を作成する。例えば、`a1b2c3d4-5678-90ab-cdef-EXAMPLE11111`。
   - `redirect_uri` (必須) - ステップ 1 で使用した認可後にエンドユーザーを にリダイレクトする URI。AppClient を作成する。例えば、`https://localhost:8080`。
   - `state` (必須) - リクエストとコールバック間の状態を維持するための一意の値です。例えば、`a8904edc-890c-1005-1996-29a757272a44`。

   ```
   GET https://productivity.appfabric.<region>.amazonaws.com/oauth2/authorize?
   app_client_id=a1b2c3d4-5678-90ab-cdef-EXAMPLE11111\
   redirect_uri=https://localhost:8080&state=a8904edc-890c-1005-1996-29a757272a44
   ```

2. 認証後、ユーザーは指定した URI にリダイレクトされ、認可コードがクエリパラメータとして返されます。例えば、`code=mM0NyJ9.MEUCIHQQgV3ChXGs2LRwxLtpsgya3ybfPYXfX-sxTAdRF-gDAiEAxX7BYKlD9krG3J2VtprOjVXZ0FSUX9whdekqJ-oampc` です。

```
https://localhost:8080/?code=mM0NyJ9.MEUCIHQQgV3ChXGs2LRwxLtpsgya3ybfPYXfX-
sxTAdRF-gDAiEAxX7BYKlD9krG3J2VtprOjVXZ0FSUX9whdekqJ-
oampc&state=a8904edc-890c-1005-1996-29a757272a44
```

3. AppFabric `oauth2/token` API オペレーションを使用して、この認可コードをアクセストークンと交換します。

   このトークンは API リクエストに使用され、AppClient の検証が完了するまでは `starterUserEmails` に対して有効です。AppClient の検証が完了した後は、このトークンはどのユーザーにでも使用できます。この API オペレーションを実行するには、アカウント署名バージョン 4 の認証情報を使用する必要があります。署名バージョン 4 の詳細については、IAM ユーザーガイドの AWS API リクエストの署名を参照してください。

   リクエストフィールド

   - `code` (必須) - 最後のステップで認証した後に受け取った認可コード。例えば、`mM0NyJ9.MEUCIHQQgV3ChXGs2LRwxLtpsgya3ybfPYXfX-sxTAdRF-gDAiEAxX7BYKlD9krG3J2VtprOjVXZ0FSUX9whdekqJ-oampc`。
   - `app_client_id`(必須) - ステップ 1. AppClient の作成 で作成した AWS アカウント の AppClient ID です。AppClient を作成する。例えば、`a1b2c3d4-5678-90ab-cdef-EXAMPLE11111`。
   - `grant_type` (必須) - 値は `authorization_code` でなければなりません。
   - `redirect_uri` (必須) - ステップ 1 で使用した認可後にユーザーを にリダイレクトする URI。AppClient を作成する。こちらは、認可コードの作成に使用したものと同じリダイレクト URI である必要があります。例えば、`https://localhost:8080`。

   レスポンスフィールド

   - `expires_in` - トークンの有効期限が切れるまでの残り時間です。デフォルトの有効期限は 12 時間です。
   - `refresh_token` - 最初の /token リクエストで受け取った更新トークンです。
   - `token` - 最初の /token リクエストで受け取ったトークンです。
   - `token_type` - この値は `Bearer` になります。
   - `appfabric_user_id` - AppFabric ユーザー ID です。この値は、リクエストが `authorization_code` グラントタイプを使用している場合のみ返されます。

```
curl --location \
"https://appfabric.<region>.amazonaws.com/oauth2/token" \
--header "Content-Type: application/json" \
--header "X-Amz-Content-Sha256: <sha256_payload>" \
--header "X-Amz-Security-Token: <security_token>" \
--header "X-Amz-Date: 20230922T172215Z" \
--header "Authorization: AWS4-HMAC-SHA256 ..." \
--data "{
    \"code\": \"mM0NyJ9.MEUCIHQQgV3ChXGs2LRwxLtpsgya3ybfPYXfX-sxTAdRF-
gDAiEAxX7BYKlD9krG3J2VtprOjVXZ0FSUX9whdekqJ-oampc",
    \"app_client_id\": \"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111\",
    \"grant_type\": \"authorization_code\",
    \"redirect_uri\": \"https://localhost:8080\"
}"
```

アクションが成功すると、サービスは HTTP 200 レスポンスを返します。

```
{
    "expires_in": 43200,
    "refresh_token": "apkaeibaerjr2example",
    "token": "apkaeibaerjr2example",
    "token_type": "Bearer",
    "appfabric_user_id" : "<userId>"
}
```

ステップ 3. AppFabric ユーザーポータル URL をアプリケーションに追加する

エンドユーザーは、インサイトの生成に使われる、自分のアプリケーションから得られるデータへのアクセスを、AppFabric に許可する必要があります。AppFabric は、専用のユーザーポータル (ポップアップ画面) を構築してそこでエンドユーザーにアプリケーションを認可させることで、アプリケーションデベロッパーがこのプロセスを自分で管理する煩雑さを解消します。AppFabric for productivity を有効にする準備が整うとユーザーはこのユーザーポータルに誘導されます。ユーザーはそこで、インサイトやクロスアプリケーションアクションの生成に使用されるアプリケーションを、接続したり管理したりできます。ログインすると、ユーザーはアプリケーションを AppFabric for productivity に接続できるようになり、自分のアプリケーションでインサイトやアクションを確認できます。アプリケーションを AppFabric for productivity と連携させるには、特定の AppFabric

URL をアプリケーションに追加する必要があります。これは、ユーザーがアプリケーションから AppFabric ユーザーポータルに直接アクセスできるようにするために欠かせないステップです。

1. アプリケーションの設定に進み、リダイレクト URL を追加するセクションを探します。
2. 該当するセクションが見つかったら、以下の AppFabric URL をリダイレクト URL としてアプリケーションに追加します。

```
https://userportal.appfabric.<region>.amazonaws.com/eup_login
```

URL を追加すると、ユーザーを AppFabric ユーザーポータルに誘導するようにアプリケーションで設定されます。ここで、ユーザーはログインし、AppFabric for productivity のインサイトの生成に使用されるアプリケーションを接続して管理することができます。

ステップ 4. AppFabric を使用してクロスアプリケーションのインサイトとアクションを表示する

ユーザーがアプリケーションを接続すると、ユーザーのインサイトを活用でき、アプリケーションとコンテキストを切り替える手間が省けて生産性を高めることができます。AppFabric は、ユーザーが持つアクセス権限に基づいてインサイトを生成します。AppFabric は、AppFabric AWS アカウント が所有する にユーザーデータを保存します。AppFabric によるデータの使用方法についての詳細は、「AppFabric でのデータ処理」を参照してください。

ユーザーレベルのインサイトとアクションをアプリケーション内で生成して表示するときは、AI を活用した以下の API を使用できます。

- `ListActionableInsights` — 詳細については、以下の「実行可能なインサイト」のセクションを参照してください。
- `ListMeetingInsights` — 詳細については、本ガイドで後述する「会議の準備」のセクションを参照してください。

実行可能なインサイト (**`ListActionableInsights`**)

`ListActionableInsights` は、アプリケーション (E メール、カレンダー、メッセージ、タスクなど) を横断するアクティビティに基づいて実行可能なインサイトを表示する、日々の業務を効率よく管理するのに役立つ API です。表示されるインサイトには、インサイトの生成に使用されたアーティファクトへの埋め込みリンクも含まれているため、ユーザーはインサイトの生成に使用さ

れたデータをすばやく確認できます。さらにこの API は、インサイトに基づく推奨されるアクションを返すことができるため、ユーザーは自分のアプリケーションからクロスアプリケーションアクションを実行することができます。具体的には、この API は Asana、Google Workspace、Microsoft 365、Smartsheet などのプラットフォームと連携し、ユーザーがメール送信、カレンダーイベントの作成、タスクの作成などを行うことを可能にします。大規模言語モデル (LLM) では、推奨されるアクション (メール本文やタスク名など) に自動的に詳細を追加できます。ユーザーはこれを実行前にカスタマイズできるため、意思決定を簡略化し、生産性を高めることができます。エンドユーザーがアプリケーションを承認する場合と同様に、AppFabric では、ユーザーは同じ専用ポータルを使用してクロスアプリケーションアクションを表示、編集、実行できます。アクションを実行する場合、AppFabric は、ユーザーを AppFabric ユーザーポータルにリダイレクトして、アクションの詳細を確認してからこれを実行できるようにすることを ISV に要求しています。AppFabric で生成されるアクションのすべてに、固有の URL があります。この URL は `ListActionableInsights` API レスポンスのレスポンスで使用できます。

以下は、サポートされているクロスアプリケーションアクションと、どのアプリケーションでサポートされているかをまとめたものです。

- E メールの送信 (Google Workspace、Microsoft 365)
- カレンダーイベントの作成 (Google Workspace、Microsoft 365)
- タスクの作成 (Asana、Smartsheet)

リクエストフィールド

- `nextToken` (オプション) - 次回のインサイトのセットを取得するためのページネーショントークンです。
- `includeActionExecutionStatus` - アクションの実行ステータスのリストを受け入れるフィルターです。これらのアクションは渡されたステータス値に基づいてフィルタリングされます。使用できる値: `NOT_EXECUTED` | `EXECUTED`

リクエストヘッダー

- 承認ヘッダーは `Bearer Token` 値とともに渡す必要があります。

レスポンスフィールド

- `insightId` - 生成されたインサイトの一意の ID です。

- `insightContent` - インサイトの概要と、インサイトの生成に使用されたアーティファクトへの埋め込みリンクを返します。注: こちらは、埋め込みリンク (<a>タグ) を含む HTML コンテンツです。
- `insightTitle` - 生成されたインサイトの件名です。
- `createdAt` - インサイトが生成された日時です。
- `actions` - 生成されたインサイトで推奨されるアクションのリストです。アクションオブジェクト:
  - `actionId` - 生成されたアクションの一意の ID です。
  - `actionIconUrl` - アクションの実行が推奨されているアプリケーションのアイコン URL です。
  - `actionTitle` - 生成されたアクションの件名です。
  - `actionUrl` - エンドユーザーが AppFabric のユーザーポータルでアクションを表示して実行するための一意の URL です。注: アクションを実行する場合、ISV のアプリケーションはこの URL を使用してユーザーを AppFabric ユーザーポータル (ポップアップ画面) にリダイレクトします。
  - `actionExecutionStatus` - アクションのステータスを示す列挙型です。指定できる値は `EXECUTED | NOT_EXECUTED` です。
- `nextToken` (オプション) - 次回のインサイトのセットを取得するためのページネーショントークンです。こちらはオプションのフィールドで、null が返された場合は、ロードするインサイトがそれ以上ないことを意味します。

詳細については、「ActionableInsights」を参照してください。

```
curl -v --location \
  "https://productivity.appfabric.<region>.amazonaws.com"\
"/actionableInsights" \
  --header "Authorization: Bearer <token>"
```

アクションが成功すると、サービスは HTTP 200 レスポンスを返します。

```
200 OK

{
    "insights": [
        {
            "insightId": "7tff3412-33b4-479a-8812-30EXAMPLE1111",
```

```
            "insightContent": "You received an email from James
            regarding providing feedback
            for upcoming performance reviews.",
            "insightTitle": "New feedback request",
            "createdAt": 2022-10-08T00:46:31.378493Z,
            "actions": [
                {
                    "actionId": "5b4f3412-33b4-479a-8812-3EXAMPLE2222",
                    "actionIconUrl": "https://d3gdwnnn63ow7w.cloudfront.net/
eup/123.svg",
                    "actionTitle": "Send feedback request email",
                    "actionUrl": "https://userportal.appfabric.us-east-1.amazonaws.com/
action/action_id_1"
                    "actionExecutionStatus": "NOT_EXECUTED"
                }
            ]
        },
        {
            "insightId": "2dff3412-33b4-479a-8812-30bEXAMPLE3333",
            "insightContent":"Steve sent you an email asking for details on project.
 Consider replying to the email.",
            "insightTitle": "New team launch discussion",
            "createdAt": 2022-10-08T00:46:31.378493Z,
            "actions": [
                {
                    "actionId": "74251e31-5962-49d2-9ca3-1EXAMPLE1111",
                    "actionIconUrl": "https://d3gdwnnn63ow7w.cloudfront.net/
eup/123.svg",
                    "actionTitle": "Reply to team launch email",
                    "actionUrl": "https://userportal.appfabric.us-east-1.amazonaws.com/
action/action_id_2"
                    "actionExecutionStatus": "NOT_EXECUTED"
                }
            ]
        }
    ],
    "nextToken": null
}
```

会議の準備 (**ListMeetingInsights**)

ListMeetingInsights は、会議の内容を要約したり、E メールやメッセージその他のアプリケーションを横断して関連性の高いアーティファクトを表示したりすることで、今後の会議に向けて準備

するのに役立つ API です。ユーザーはすぐに会議に向けた準備ができ、コンテンツを見つけるためにアプリケーション間を切り替える手間が省けます。

リクエストフィールド

- `nextToken` (オプション) - 次回のインサイトのセットを取得するためのページネーショントークンです。

リクエストヘッダー

- 承認ヘッダーは `Bearer Token` 値とともに渡す必要があります。

レスポンスフィールド

- `insightId` - 生成されたインサイトの一意の ID です。
- `insightContent` - インサイトの説明で、詳細は文字列の形式で強調表示されます。例えば、なぜこのインサイトが重要なのか、など。
- `insightTitle` - 生成されたインサイトの件名です。
- `createdAt` - インサイトが生成された日時です。
- `calendarEvent` - ユーザーが注目すべき重要なカレンダーイベントまたは会議です。カレンダーイベントオブジェクト:
  - `startTime` - イベントの開始時刻です。
  - `endTime` - イベントの終了時刻です。
  - `eventUrl` - ISV アプリケーションのカレンダーイベントの URL です。
- `resources` - インサイトの生成に関連する他のリソースを含むリストです。リソースオブジェクト:
  - `appName` - リソースが属するアプリケーションの名前です。
  - `resourceTitle` - リソースの件名です。
  - `resourceType` - リソースのタイプです。指定できる値は `EMAIL | EVENT | MESSAGE | TASK` です。
  - `resourceUrl` - アプリケーション内のリソース URL です。
  - `appIconUrl` - リソースが属するアプリケーションの画像 URL です。

- nextToken (オプション) - 次回のインサイトのセットを取得するためのページネーショントークンです。こちらはオプションのフィールドで、null が返された場合は、ロードするインサイトがそれ以上ないことを意味します。

詳細については、「MeetingInsights」を参照してください。

```
curl --location \
  "https://productivity.appfabric.<region>.amazonaws.com"\
"/meetingContexts" \
  --header "Authorization: Bearer <token>"
```

アクションが成功すると、HTTP 201 レスポンスが返されます。

```
200 OK

{
    "insights": [
        {
            "insightId": "74251e31-5962-49d2-9ca3-15EXAMPLE4444"
            "insightContent": "Project demo meeting coming up soon. Prepare
 accordingly",
            "insightTitle": "Demo meeting next week",
            "createdAt": 2022-10-08T00:46:31.378493Z,
            "calendarEvent": {
                    "startTime": {
                        "timeInUTC": 2023-10-08T10:00:00.000000Z,
                        "timeZone": "UTC"
                     },
                    "endTime": {
                        "timeInUTC": 2023-10-08T11:00:00.000000Z,
                        "timeZone": "UTC"
                     },
                    "eventUrl": "http://someapp.com/events/1234",
            }
            "resources": [
                {
                    "appName": "SOME_EMAIL_APP",
                    "resourceTitle": "Email for project demo",
                    "resourceType": "EMAIL",
                    "resourceUrl": "http://someapp.com/emails/1234",
                    "appIconUrl":"https://d3gdwnnn63ow7w.cloudfront.net/eup/123.svg"
                }
```

```
            ]
        },
        {
            "insightId": "98751e31-5962-49d2-9ca3-15EXAMPLE5555"
            "insightContent": "Important code complete task is now due. Consider
 updating the status.",
            "insightTitle": "Code complete task is due",
            "createdAt": 2022-10-08T00:46:31.378493Z,
            "calendarEvent":{
                    "startTime": {
                        "timeInUTC": 2023-10-08T10:00:00.000000Z,
                        "timeZone": "UTC"
                     },
                    "endTime": {
                        "timeInUTC": 2023-10-08T11:00:00.000000Z,
                        "timeZone": "UTC"
                     },
                    "eventUrl": "http://someapp.com/events/1234",
            },
            "resources": [
                {
                    "appName": "SOME_TASK_APPLICATION",
                    "resourceTitle": "Code Complete task is due",
                    "resourceType": "TASK",
                    "resourceUrl": "http://someapp.com/task/1234",
                    "appIconUrl": "https://d3gdwnnn63ow7w.cloudfront.net/eup/123.svg"
                }
            ]
        }
    ],
    "nextToken": null
}
```

インサイトやアクションに関するフィードバックを提供してください。

生成されたインサイトやアクションに関するフィードバックを送るときは、AppFabric PutFeedback API オペレーションを使用します。この機能をアプリケーションに埋め込むと、特定の InsightId または ActionId に対するフィードバック評価 (1 ~ 5、値が大きい高いほど評価が高い) を送信できます。

リクエストフィールド

- `id` - フィードバックの送信対象となるオブジェクトの識別子です。InsightId か ActionId のいずれかになります。
- `feedbackFor` - フィードバックの送信対象となるリソースタイプです。使用できる値: `ACTIONABLE_INSIGHT` | `MEETING_INSIGHT` | `ACTION`
- `feedbackRating` - 1 から 5 までの評価です。値が大きいほど評価が高いことを意味します。

レスポンスフィールド

- レスポンスフィールドはありません。

詳細については、「PutFeedback」を参照してください。

```
curl --request POST \
  --url "https://productivity.appfabric.<region>.amazonaws.com"\
"/feedback" \
  --header "Authorization: Bearer <token>" \
  --header "Content-Type: application/json" \
  --data '{
    "id": "1234-5678-9012",
    "feedbackFor": "ACTIONABLE_INSIGHT"
    "feedbackRating": 3
}'
```

アクションが成功した場合、サービスは空の HTTP 本文を持つ HTTP 201 レスポンスを返します。

ステップ 5. AppFabric にアプリケーションの検証をリクエストする

ここに至るまでに、アプリケーションの UI を、AppFabric のクロスアプリケーションのインサイトとアクションを埋め込むように更新し、1 人のユーザーに関するインサイトを取得しました。テストに満足し、AppFabric によって機能強化したエクスペリエンスを他のユーザーにも広げたい場合は、アプリケーションを AppFabric に送信し、レビューと検証を受けます。アプリケーションデベロッパーとエンドユーザーを保護し彼らのデータを保護するため、AppFabric では、アプリケーションの情報を検証してから採用範囲を拡大できるようにしています。そうすることで、責任ある方法でユーザーによる利用を広げるための準備をします。

検証プロセスの開始

appfabric-appverification@amazon.com にメールを送信し、アプリケーションの検証をリクエストして、検証プロセスを開始します。

E メールの本文には次の情報を含めます。

- AWS アカウント ID
- 検証を依頼するアプリケーションの名称
- 自分の AppClient ID
- 自分の連絡先情報

また、可能であれば、以下の情報も含めると、優先順位や影響を評価する際に役立ちます。

- アクセスを許可するユーザー数 (推計)
- リリース日

> **Note**
> AWS アカウント マネージャーまたは AWS パートナー開発マネージャーがいる場合は、E メールでコピーしてください。追加しておくと、検証プロセスをスピーディに進めることができます。

検証基準

検証プロセスを開始する前に、次の基準を満たしている必要があります。

- AppFabric for productivity AWS アカウント を使用するには、有効な を使用する必要があります

また、以下の基準のうち 1 つ以上を満たしている必要があります。

- 組織は、少なくともAWS 「 Select」階層 AWS Partner Network を持つ の AWS パートナーです。詳細については、「AWS サービスパートナーティア」を参照してください。
- 所属する組織が、過去 3 年間で AppFabric サービスに 10,000 ドル以上を費やしていること。
- アプリケーションが AWS Marketplaceに掲載されていること。詳細については、「AWS Marketplace」を参照してください。

検証ステータスの更新の待機

アプリケーションの審査が完了すると、メールで返信があり、AppClient のステータスが `pending_verification` から `verified` に変わります。アプリケーションが却下された場合は、検証プロセスを改めて行う必要があります。

AppFabric for productivity AppClients の管理

> AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

AppFabric for productivity の AppClients を管理することで、認証および認可のプロセスのスムーズなオペレーションと管理を実現できます。

AppClient の詳細を取得する

AppClient ステータスの確認など AppClient に関する詳細を表示するときは、AppFabric `GetAppClient` API オペレーションを使用します。詳細については、「GetAppClient」を参照してください。

AppClient の詳細を取得するには、少なくとも IAM ポリシーの許可が必要です。`"appfabric:GetAppClient"`詳細については、「AppClients の詳細を得るためのアクセスを許可する」を参照してください。

リクエストフィールド

- `appClientId` - AppClient ID です。

レスポンスフィールド

- `appName` - AppFabric ユーザーポータルの同意ページでユーザーに表示されるアプリケーションの名前です。
- `customerManagedKeyIdentifier` (オプション) - データの暗号化に使用されるカスタマーマネージドキー (KMS が生成) の ARN です。指定しない場合、 AWS AppFabric マネージドキーが使用されます。
- `description` - アプリケーションの説明です。
- `redirectUrls` - 承認後にエンドユーザーがリダイレクトされる URI です。redirectUrls は 5 個まで追加できます。例えば、`https://localhost:8080`。

- `starterUserEmails` - アプリケーションが検証されるまでの間、インサイトを受け取るためのアクセスが許可されるユーザーの E メールアドレスです。使用できるアドレスは 1 つのみです。例えば、`anyuser@example.com`。
- `verificationStatus` - AppClient の検証ステータスです。
  - `pending_verification` - AppClient の検証が、AppFabric でまだ進行中です。AppClient の検証が終わるまでは、この AppClient を使用できるのは (`starterUserEmails` で指定された) 1 人のユーザーのみとなります。
  - `verified` - AppFabric による検証プロセスが正常に完了し、AppClient は検証済みです。
  - `rejected` - AppClient の検証プロセスが AppFabric によって拒否されました。検証プロセスが再開され正常に完了するまでは、他のユーザーが AppClient を使用することはできません。

```
curl --request GET \
  --header "Content-Type: application/json" \
  --header "X-Amz-Content-Sha256: <sha256_payload>" \
  --header "X-Amz-Security-Token: <security_token>" \
  --header "X-Amz-Date: 20230922T172215Z" \
  --header "Authorization: AWS4-HMAC-SHA256 ..." \
  --url https://appfabric.<region>.amazonaws.com/appclients/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111
```

アクションが成功すると、サービスは HTTP 200 レスポンスを返します。

```
200 OK

{
    "appClient": {
        "appName": "Test App",
        "arn": "arn:aws:appfabric:<region>:111122223333:appclient/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
        "customerManagedKeyArn": "arn:aws:kms:<region>:111122223333:key/<key>",
        "description": "This is a test app",
        "redirectUrls": [
            "https://localhost:8080"
        ],
        "starterUserEmails": [
            "anyuser@example.com"
        ],
        "verificationDetails": {
            "verificationStatus": "pending_verification"
```

```
        }
    }
}
```

AppClients の一覧表示

AppClients のリストを表示するときは、AppFabric `ListAppClients` API オペレーションを使用します。AppFabric では、1 つの AppClient につき 1 つのみが許可されます AWS アカウント。この制限は将来変更される可能性があります。詳細については、「ListAppClients」を参照してください。

AppClient をリスト化するには、少なくとも `"appfabric:ListAppClients"` IAM ポリシーの許可が必要です。詳細については、「AppClients を一覧表示するためのアクセスを許可する」を参照してください。

リクエストフィールド

- 必須フィールドはありません。

レスポンスフィールド

- `appClientARN` - AppClient ID を含む Amazon リソースネーム (ARN) です。例えば、AppClient ID は `a1b2c3d4-5678-90ab-cdef-EXAMPLE11111` です。
- `verificationStatus` - AppClient の検証ステータスです。
  - `pending_verification` - AppClient の検証が、AppFabric でまだ進行中です。AppClient の検証が終わるまでは、この AppClient を使用できるのは (`starterUserEmails` で指定された) 1 人のユーザーのみとなります。
  - `verified` - AppFabric による検証プロセスが正常に完了し、AppClient は検証済みです。
  - `rejected` - AppClient の検証プロセスが AppFabric によって拒否されました。検証プロセスが再開され正常に完了するまでは、他のユーザーが AppClient を使用することはできません。

```
curl --request GET \
  --header "Content-Type: application/json" \
  --header "X-Amz-Content-Sha256: <sha256_payload>" \
  --header "X-Amz-Security-Token: <security_token>" \
  --header "X-Amz-Date: 20230922T172215Z" \
  --header "Authorization: AWS4-HMAC-SHA256 ..." \
  --url https://appfabric.<region>.amazonaws.com/appclients
```

アクションが成功すると、サービスは HTTP 200 レスポンスを返します。

```
200 OK

{
    "appClientList": [
        {
            "appClientArn": "arn:aws:appfabric:<region>:111122223333:appclient/
a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
            "verificationStatus": "pending_verification"
        }
    ]
}
```

AppClient を更新する

AppClient に対応付けられた redirectUrls を更新するときは、AppFabric `UpdateAppClient` API オペレーションを使用します。AppName、starterUserEmailsなど、その他のパラメータを変更する必要がある場合は、AppClient を削除して新しい AppClient を作成します。詳細については、「UpdateAppClient」を参照してください。

AppClient を更新するには、少なくとも `"appfabric:UpdateAppClient"` IAM ポリシーの許可が必要です。詳細については、「AppClients を更新するためのアクセスを許可する」を参照してください。

リクエストフィールド

- `appClientId` (必須) - redirectUrls を更新する AppClient ID です。
- `redirectUrls` (必須) - 更新された redirectUrls のリストです。redirectUrls は 5 個まで追加できます。

レスポンスフィールド

- `appName` - AppFabric ユーザーポータルの同意ページでユーザーに表示されるアプリケーションの名前です。
- `customerManagedKeyIdentifier` (オプション) - データの暗号化に使用されるカスタマーマネージドキー (KMS が生成) の ARN です。指定しない場合、 AWS AppFabric マネージドキーが使用されます。
- `description` - アプリケーションの説明です。

- `redirectUrls` - 承認後にエンドユーザーがリダイレクトされる URI です。例えば、`https://localhost:8080`。
- `starterUserEmails` - アプリケーションが検証されるまでの間、インサイトを受け取るためのアクセスが許可されるユーザーの E メールアドレスです。使用できるアドレスは 1 つのみです。例えば、`anyuser@example.com`。
- `verificationStatus` - AppClient の検証ステータスです。
  - `pending_verification` - AppClient の検証が、AppFabric でまだ進行中です。AppClient の検証が終わるまでは、この AppClient を使用できるのは (`starterUserEmails` で指定された) 1 人のユーザーのみとなります。
  - `verified` - AppFabric による検証プロセスが正常に完了し、AppClient は検証済みです。
  - `rejected` - AppClient の検証プロセスが AppFabric によって拒否されました。検証プロセスが再開され正常に完了するまでは、他のユーザーが AppClient を使用することはできません。

```
curl --request PATCH \
  --header "Content-Type: application/json" \
  --header "X-Amz-Content-Sha256: <sha256_payload>" \
  --header "X-Amz-Security-Token: <security_token>" \
  --header "X-Amz-Date: 20230922T172215Z" \
  --header "Authorization: AWS4-HMAC-SHA256 ..." \
  --url https://appfabric.<region>.amazonaws.com/appclients/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111 \
  --data '{
    "redirectUrls": ["https://localhost:8081"]
}'
```

アクションが成功すると、サービスは HTTP 200 レスポンスを返します。

```
200 OK

{
    "appClient": {
        "appName": "Test App",
        "arn": "arn:aws:appfabric:<region>:111122223333:appclient/a1b2c3d4-5678-90ab-
cdef-EXAMPLE11111",
        "customerManagedKeyArn": "arn:aws:kms:<region>:111122223333:key/<key>",
        "description": "This is a test app",
        "redirectUrls": [
            "https://localhost:8081"
        ],
```

```
        "starterUserEmails": [
            "anyuser@example.com"
        ],
        "verificationDetails": {
            "verificationStatus": "pending_verification"
        }
    }
}
```

AppClient を削除する

不要になった AppClients を削除するときは、AppFabric `DeleteAppClient` API オペレーションを使用します。詳細については、「DeleteAppClient」を参照してください。

AppClient を削除するには、少なくとも `"appfabric:DeleteAppClient"` IAM ポリシーの許可が必要です。詳細については、「AppClients を削除するためのアクセスを許可する」を参照してください。

リクエストフィールド

- `appClientId` - AppClient ID です。

レスポンスフィールド

- レスポンスフィールドはありません。

```
curl --request DELETE \
  --header "Content-Type: application/json" \
  --header "X-Amz-Content-Sha256: <sha256_payload>" \
  --header "X-Amz-Security-Token: <security_token>" \
  --header "X-Amz-Date: 20230922T172215Z" \
  --header "Authorization: AWS4-HMAC-SHA256 ..." \
  --url https://appfabric.<region>.amazonaws.com/appclients/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111
```

アクションが成功した場合、サービスは空の HTTP 本文を持つ HTTP 204 レスポンスを返します。

エンドユーザー用トークンを更新する

AppClient がエンドユーザー用に取得するトークンは、有効期限が切れると更新できます。これは トークン API と grant_type `refresh_token` とを組み合わせて使用することで実行できま

す。grant_type が authorization_code である場合は、使用すべき refresh_token は、トークン API のレスポンスの一部として返されます。デフォルトの有効期限は 12 時間です。更新 API を呼び出すには、"appfabric:Token" IAM ポリシーの許可が必要です。詳細については、「トークン」および「AppClients を更新するためのアクセスを許可する」を参照してください。

リクエストフィールド

- refresh_token (必須) - 最初の /token リクエストで受け取った更新トークンです。
- app_client_id (必須) - AWS アカウント用に作成された AppClient リソースの ID です。
- grant_type (必須) - refresh_token でなければなりません。

レスポンスフィールド

- expires_in - トークンの有効期限が切れるまでの残り時間です。デフォルトの有効期限は 12 時間です。
- refresh_token - 最初の /token リクエストで受け取った更新トークンです。
- token - 最初の /token リクエストで受け取ったトークンです。
- token_type - この値は Bearer になります。
- appfabric_user_id - AppFabric ユーザー ID です。この値は、リクエストが authorization_code グラントタイプを使用している場合のみ返されます。

```
curl --location \
"https://appfabric.<region>.amazonaws.com/oauth2/token" \
--header "Content-Type: application/json" \
--header "X-Amz-Content-Sha256: <sha256_payload>" \
--header "X-Amz-Security-Token: <security_token>" \
--header "X-Amz-Date: 20230922T172215Z" \
--header "Authorization: AWS4-HMAC-SHA256 ..." \
--data "{
    \"refresh_token\": \"<refresh_token>",
    \"app_client_id\": \"a1b2c3d4-5678-90ab-cdef-EXAMPLE11111\",
    \"grant_type\": \"refresh_token\"
}"
```

アクションが成功すると、サービスは HTTP 200 レスポンスを返します。

```
200 OK
```

```
{
    "expires_in": 43200,
    "token": "apkaeibaerjr2example",
    "token_type": "Bearer",
    "appfabric_user_id" : "${UserID}"
}
```

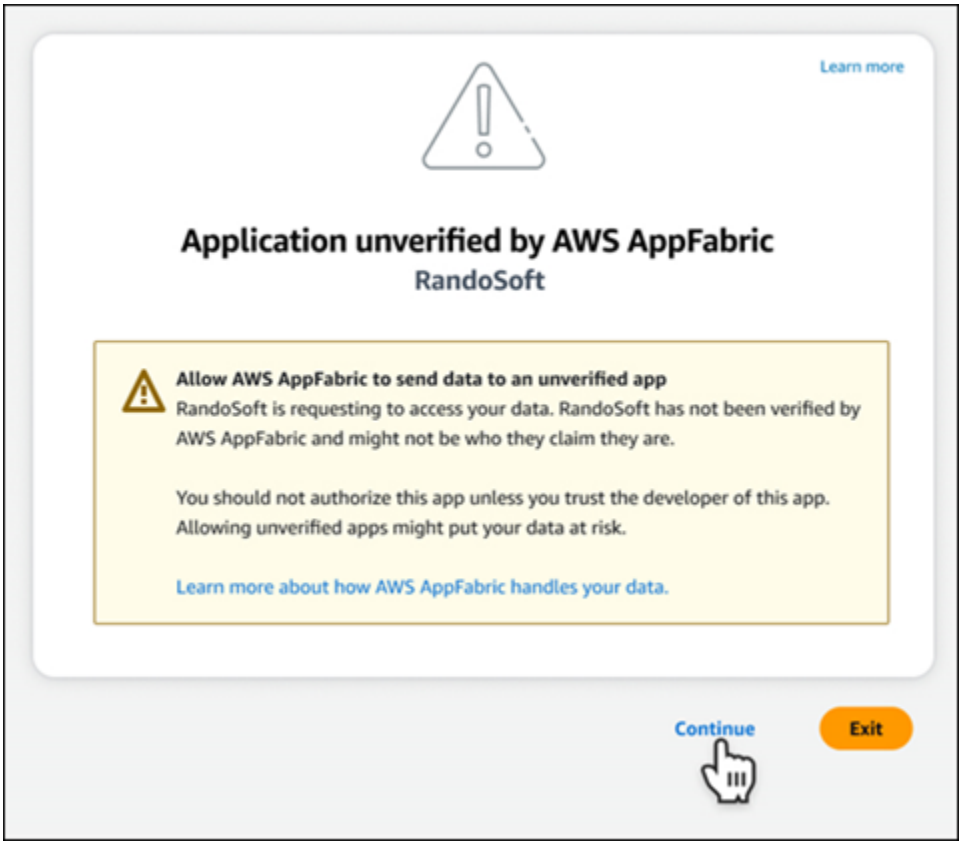
AppFabric for productivity の AppClients AppClients のトラブルシューティング

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

このセクションでは、AppFabric for productivity のよくあるエラーとトラブルシューティングについて説明します。

未検証のアプリケーション

AppFabric for productivity を使用してエクスペリエンスを強化するアプリケーションデベロッパーは、エンドユーザーに機能を公開する前に検証プロセスを経る必要があります。アプリケーションはすべて、未検証の状態からスタートし、検証プロセスが完了した場合のみ、検証済みの状態になります。つまり、AppClient の作成時に使用した `starterUserEmails` にはこのメッセージが表示されます。



`CreateAppClient` エラー

ServiceQuotaExceededException

AppClient の作成時に次の例外が発生した場合、 AWS アカウントごとに作成できる AppClient の数を超えています。上限は 1 です。HTTP ステータスコード: 402

```
ServiceQuotaExceededException / SERVICE_QUOTA_EXCEEDED
You have exceeded the number of AppClients that can be created per AWS Account. The
 limit is 1.
HTTP Status Code: 402
```

`GetAppClient` エラー

ResourceNotFoundException

AppClient の詳細を取得するときに以下の例外が表示される場合は、正しい AppClient 識別子が入力されていることを確認します。こちらのエラーは、指定した AppClient が見つからなかったことを意味します。

```
ResourceNotFoundException / APP_CLIENT_NOT_FOUND
The specified AppClient is not found. Ensure you’ve entered the correct AppClient
 identifier.
HTTP Status Code: 404
```

DeleteAppClient エラー

ConflictException

AppClient を削除するときに以下の例外が表示された場合は、別の削除リクエストが進行中です。そちらが完了するまで待機してから、再度お試しください。HTTP ステータスコード: 409

```
ConflictException
Another delete request is in progress. Wait until it completes then try again.
HTTP Status Code: 409
```

ResourceNotFoundException

AppClient を削除するときに以下の例外が表示される場合は、正しい AppClient 識別子が入力されていることを確認します。こちらのエラーは、指定した AppClient が見つからなかったことを意味します。

```
ResourceNotFoundException / APP_CLIENT_NOT_FOUND
The specified AppClient is not found. Ensure you’ve entered the correct AppClient
 identifier.
HTTP Status Code: 404
```

UpdateAppClient エラー

ResourceNotFoundException

AppClient を更新するときに以下の例外が表示される場合は、正しい AppClient 識別子が入力されていることを確認します。こちらのエラーは、指定した AppClient が見つからなかったことを意味します。

```
ResourceNotFoundException / APP_CLIENT_NOT_FOUND
The specified AppClient is not found. Ensure you’ve entered the correct AppClient
 identifier.
HTTP Status Code: 404
```

Authorize エラー

ValidationException

API パラメータのいずれかが、API の仕様で定義されている制限事項を満たしていない場合、以下の例外が発生することがあります。

```
ValidationException
HTTP Status Code: 400
```

理由 1: AppClient ID が指定されていない

リクエストにパラメータに `app_client_id` がありません。AppClient がまだ作成されていない場合は作成するか、既存の `app_client_id` を使用して再度試します。AppClient ID を見つけるには、ListAppClient API オペレーションを使用します。

理由 2: AppFabric がカスタマーマネージドキーにアクセスできない

```
Message: AppFabric couldn't access the customer managed key configured for AppClient.
```

おそらく、最近アクセス権限が変更されたため、AppFabric は現在、カスタマーマネージドキーにアクセスすることができません。指定したキーが存在することを確認し、AppFabric に適切なアクセス権限が付与されていることを確認します。

理由 3: 指定したリダイレクト URL が無効

```
Message: Redirect url invalid
```

リクエストのリダイレクト URL が正しいことを確認します。AppClient を作成または更新したときに指定した、リダイレクト URL のいずれかと一致する必要があります。許可済みのリダイレクト URL を一覧表示するには、GetAppClient API オペレーションを実行します。

Token エラー

TokenException

いくつかの理由から、以下のエラーが発生する場合があります。

```
TokenException
```

```
HTTP Status Code: 400
```

理由 1: 無効なメールが指定されている

```
Message: Invalid Email used
```

使用しているメールアドレスが、AppClient を作成したときに `starterUserEmails` 属性にリストされたアドレスと一致していることを確認します。一致しない場合は、一致する E メールアドレスに変更してから再度試します。使用しているメールを表示するには、GetAppClient API オペレーションを実行します。

理由 2: トークンが指定されていない場合、grant_type が refresh_token として返される。

```
Message: refresh_token must be non-null for Refresh Token Grant-type
```

リクエストで指定した更新トークンが null または空です。Token API コールのレスポンスに、受信したアクティブな `refresh_token` を指定します。

ThrottlingException

API が、許可されているクォータを超えるレートで呼び出されると、以下の例外が発生する可能性があります。

```
ThrottlingException
HTTP Status Code: 429
```

`ListActionableInsights`、`ListMeetingInsights`、`PutFeedback` のエラー

ValidationException

API パラメータのいずれかが、API の仕様で定義されている制限事項を満たしていない場合、以下の例外が発生することがあります。

```
ValidationException
HTTP Status Code: 400
```

ThrottlingException

API が、許可されているクォータを超えるレートで呼び出されると、以下の例外が発生する可能性があります。

```
ThrottlingException
HTTP Status Code: 429
```

エンドユーザー向けの AppFabric for productivity (プレビュー) の使用を開始する

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

このセクションでは、 AWS AppFabric for productivity (プレビュー) を有効にしてタスク管理とワークフロー効率を向上させたい SaaS アプリケーションのエンドユーザーを対象としています。以下の手順に従ってアプリケーションを接続し、AppFabric を承認すれば、クロスアプリケーションインサイトを表示し、好みのアプリケーションのアクション (メール送信や会議のスケジュール設定など) を実行することができます。接続できるアプリケーションは、Asana、Atlassian Jira Suite、Google Workspace、Microsoft 365、Miro、Slack、Smartsheet などです。AppFabric にコンテンツへのアクセスを許可すると、AppFabric からお使いのアプリケーションに、クロスアプリケーションのインサイトやアクションを直接取り込むことができるため、作業効率の向上やワークフローの最新状態の把握に役立ちます。

AppFabric for productivity は、Amazon Bedrockを活用した生成 AI を使用しています。AppFabric は、ユーザーが明示的に許可した場合にのみ、インサイトとアクションを生成します。ユーザーは、各アプリケーションに、使用するコンテンツを完全にコントロールする権限を付与します。AppFabric が、インサイトの生成に使用される、基盤となる大規模言語モデルのトレーニングや改善に、お客様のデータを使用することはありません。詳細については、「Amazon Bedrock よくある質問」を参照してください。

トピック

- 前提条件
- ステップ 1. AppFabric にサインイン
- ステップ 2. インサイトを表示することをアプリに許可する
- ステップ 3. アプリケーションを接続してインサイトとアクションを生成する
- ステップ 4. 自分のアプリケーションでインサイトを確認しクロスアプリケーションアクションを実行する
- IT およびセキュリティ管理者向けの AppFabric for productivity (プレビュー) 機能へのアクセスを管理する

- AppFabric for productivity のエンドユーザーエラーのトラブルシューティング

前提条件

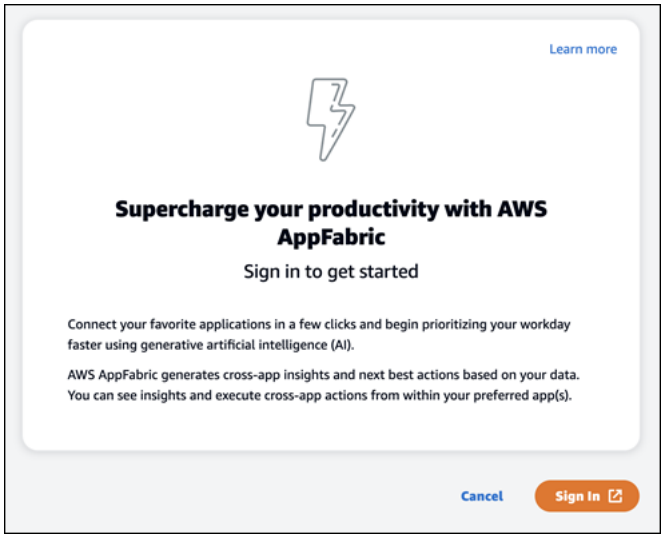
作業を開始する前に、以下の条件がそろっていることを確認します。

- AppFabric にサインインするための認証情報: AppFabric for productivity の使用を開始するには、Asana、Google Workspace、Microsoft 365、Slack のいずれかのプロバイダーのフェデレーションサインイン認証情報 (ユーザー名とパスワード) が必要です。AppFabric にサインインすると、AppFabric for productivity を有効にするそれぞれのアプリケーションで、ユーザーとして認識されます。サインインすると、お使いのアプリケーションを接続してインサイトの生成を開始できます。
- アプリケーションを接続するための認証情報: クロスアプリケーションのインサイトとアクションは、ユーザーが認可するアプリケーションに基づいてのみ生成されます。サインインの認証情報は (ユーザー名とパスワード) は、認可するアプリケーションごとに、必要になります。サポートされているアプリケーションには、Asana、Atlassian Jira Suite、Google Workspace、Microsoft 365、Miro、Slack、Smartsheet などがあります。

ステップ 1. AppFabric にサインイン

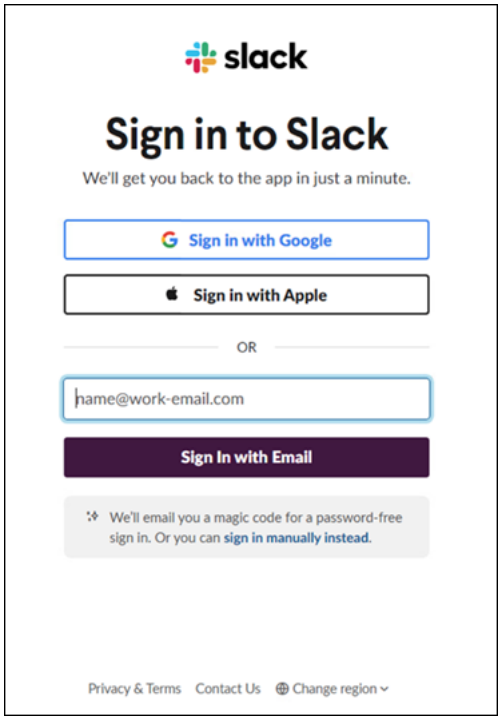
アプリケーションを AppFabric に接続し、お使いのコンテンツやインサイトを好みのアプリケーションに直接取り込みます。

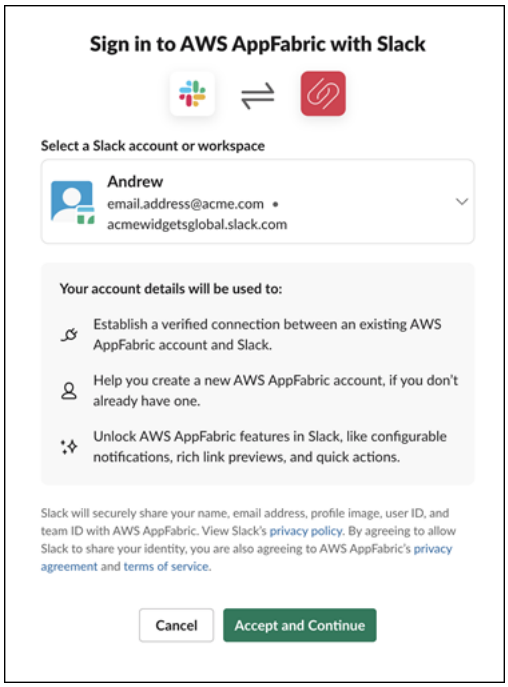
1. AppFabric は、どのアプリケーションでもさまざまな方法で使用することができ、アプリケーションのエクスペリエンスを高めます。そのため、以下の AppFabric for productivity のホームページにアクセスするためのエントリポイントは、アプリケーションごとに異なります。このホームページでは、AppFabric を有効にするプロセスに関するコンテキストが設定されており、ユーザーは最初にサインインを求められます。ユーザーが AppFabric を有効にしようとしているすべてのアプリケーションで、この画面が表示されます。



2. Asana、Google Workspace、Microsoft 365、Slack のいずれかのプロバイダーから自分の認証情報を使ってサインインします。ベストなエクスペリエンスを得るには、AppFabric を有効にしようとしているアプリケーションと同じプロバイダーを使用してサインインします。例えば、App1 で Google Workspace の認証情報を使用した場合は、App2 でも、また再度サインインする必要がある場合にも、Google Workspace を使用することが推奨されます。別のプロバイダーを使ってサインインする場合は、アプリケーションを接続するプロセスを再度実行する必要があります。

3. サインインの認証情報の入力を求められたら入力し、このプロバイダーから AppFabric にサインインすることを承認します。

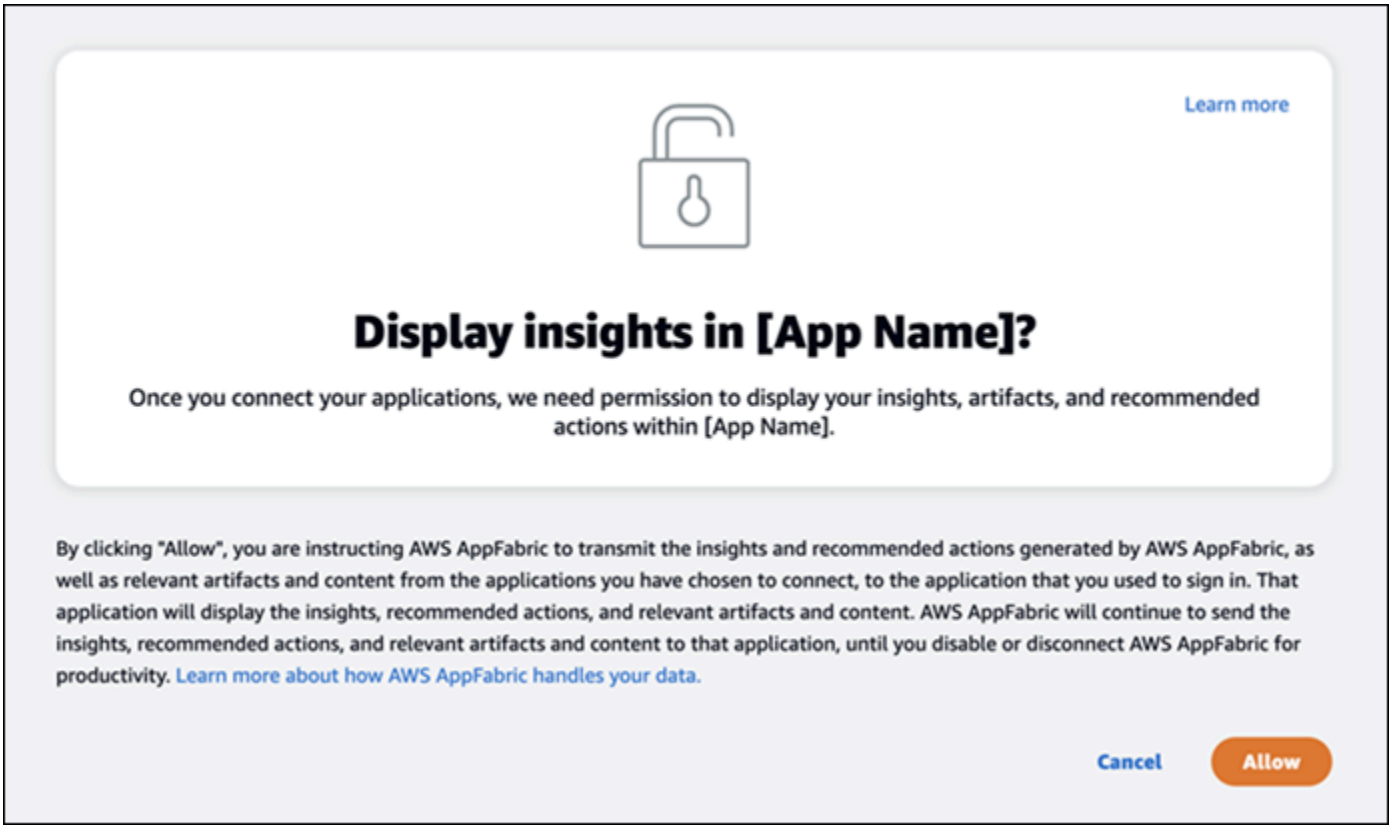




ステップ 2. インサイトを表示することをアプリに許可する

サインインすると、AppFabric for productivity を有効にしようとしているアプリケーション内に、クロスアプリケーションのインサイトとアクションを表示することを、AppFabric に許可するかどうかを尋ねる同意ページが表示されます。例えば、AppFabric に、Google Workspace のメールやカレン

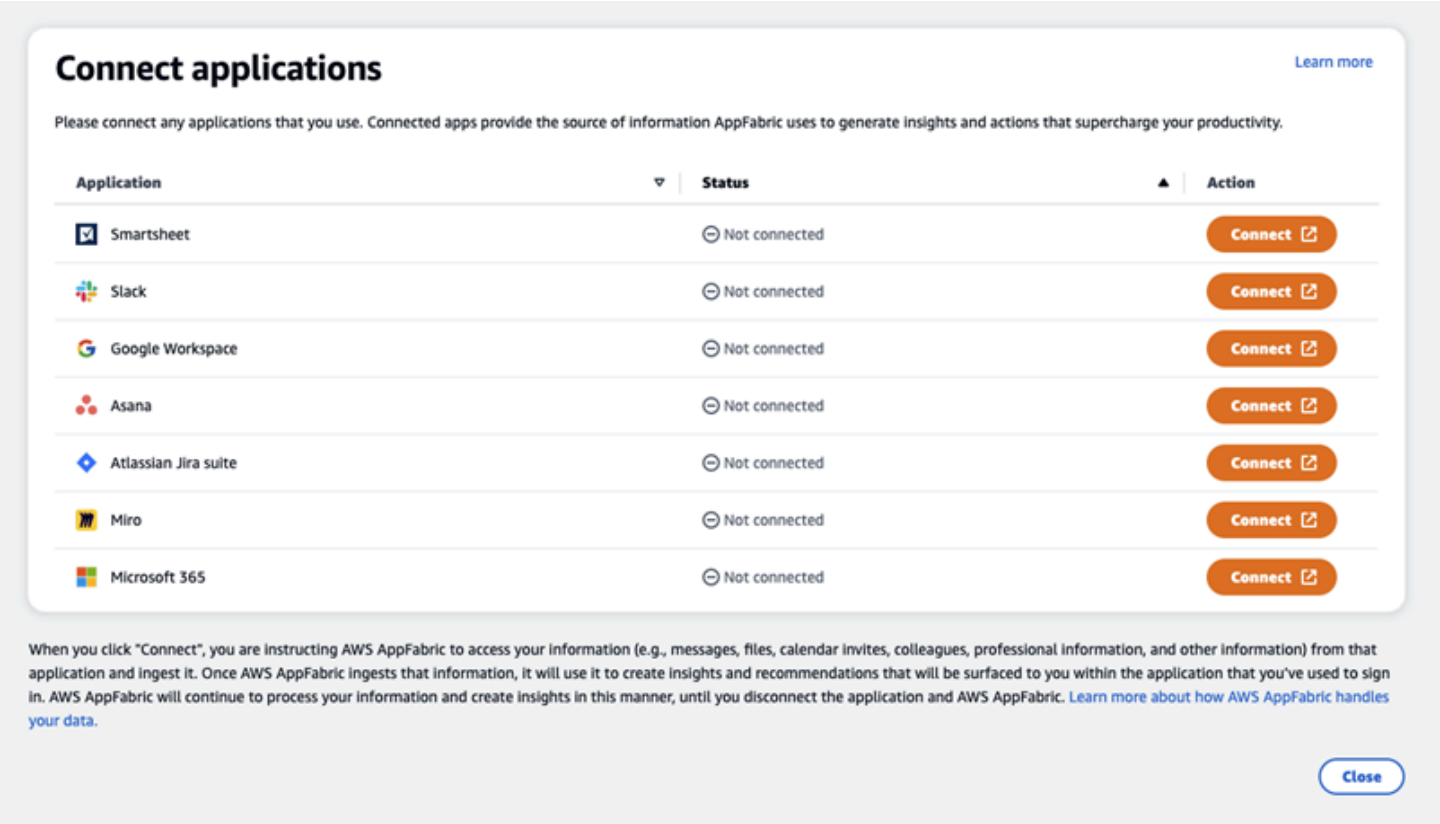
ダーのイベントを取り込み Asana に表示することを許可するかどうか、といったようにです。この同意のステップは、AppFabric を有効にするアプリケーションごとに 1 回実行すれば、その後は必要ありません。



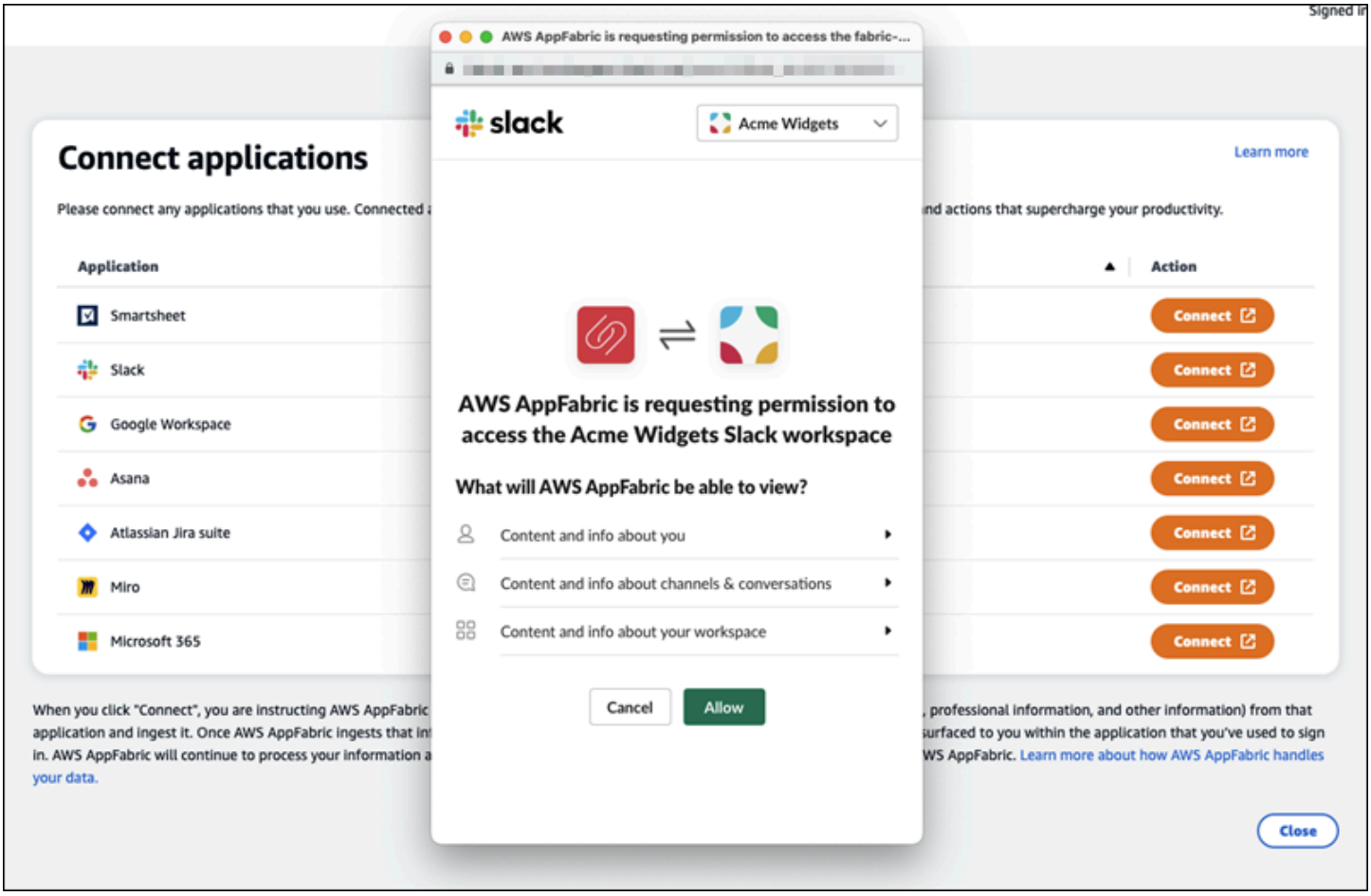
ステップ 3. アプリケーションを接続してインサイトとアクションを生成する

同意のステップを完了すると、[アプリケーションを接続] ページに進みます。ここでは、クロスアプリケーションのインサイトやアクションの生成に使用する個々のアプリケーションの、接続、切断、再接続が行えます。ほとんどの場合、サインインして同意のステップを完了した後は、このページを使って、接続したアプリケーションを管理します。

アプリケーションを接続するには、使用するアプリケーションの横にある [接続] をクリックします。



インサイトを生成しアクションを完了するには、アプリケーションのサインイン認証情報を入力し、AppFabric にデータへのアクセス権限を付与する必要があります。

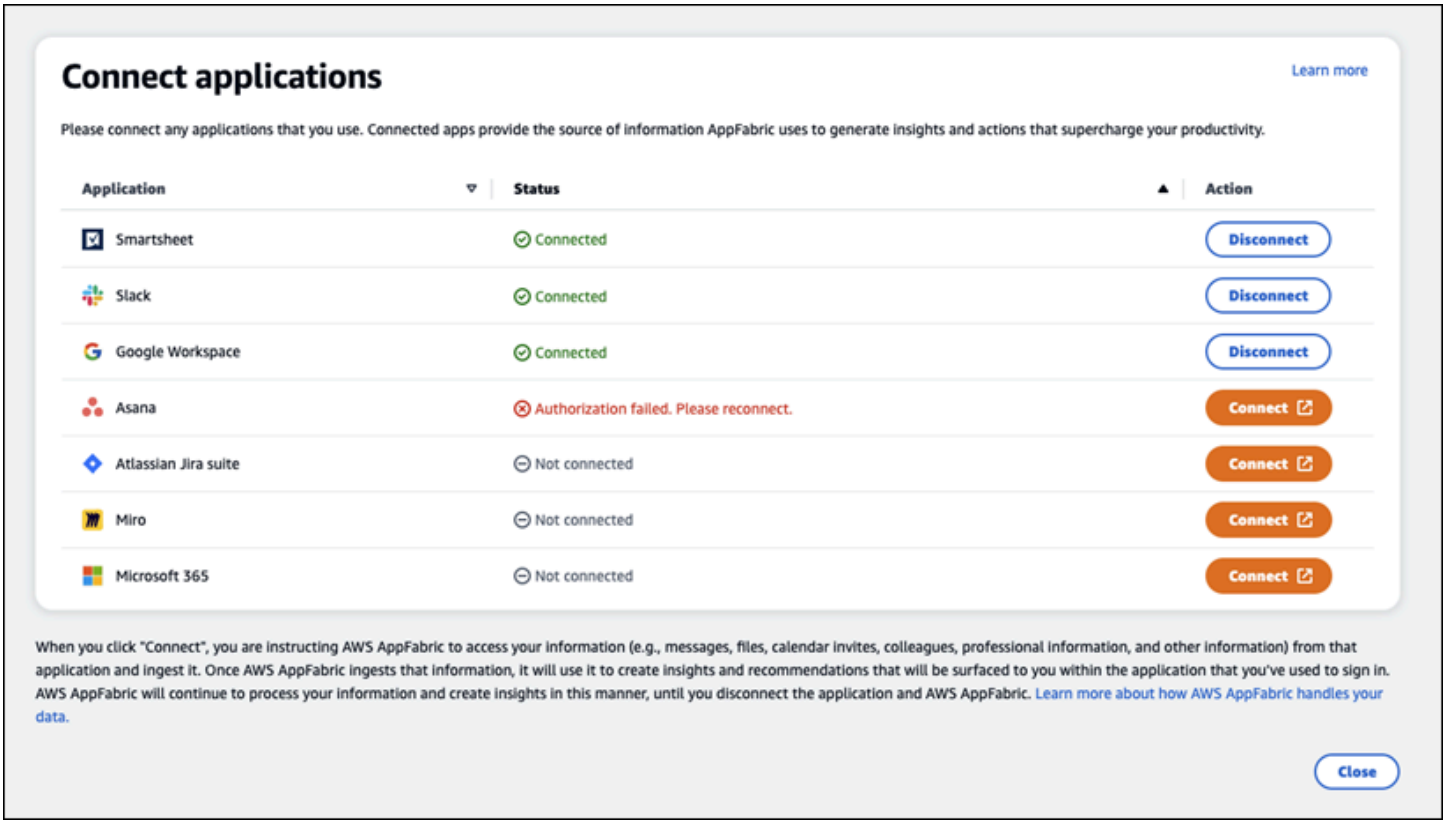


アプリケーションが正常に接続すると、そのアプリケーションのステータスが [未接続] から[接続済み] に変わります。以上の承認ステップは、インサイトやアクションの生成に使用するアプリケーションごとに実行する必要があります。

アプリケーションは、1 度接続すればその状態が永久に続くわけではありません。定期的に接続し直す必要があります。なぜ必要かというと、インサイトを生成するためのアクセス権限が引き続き付与されていることを確認するためです。

アプリケーションのステータスには、以下の状態があります。

- 接続済み - AppFabric は承認済みで、アプリケーションのデータを使ってインサイトを生成しています。
- 未接続 - AppFabric は、アプリケーションのデータを使ってインサイトを生成していません。接続するとインサイトの生成を開始できます。
- 承認に失敗しました。再接続してください。- アプリケーションの認可に失敗した可能性があります。このエラーが表示されたときは、[接続] をクリックして再度アプリケーションを接続します。



セットアップが完了したので、アプリケーションに戻ることができます。アプリケーション内にインサイトが表示されるまで、数時間かかることがあります。

接続済みのアプリケーションを管理するときは、必要に応じてこのページに戻ります。アプリケーションを切断すると、AppFabric は、そのアプリケーションのデータを使用したり、新しいデータを収集して新しいインサイトを生成したりすることを停止します。切断したアプリケーションのデータは、7 日以内に、その間アプリケーションをを再度接続しなければ、自動的に削除されます。

ステップ 4. 自分のアプリケーションでインサイトを確認しクロスアプリケーションアクションを実行する

アプリケーションを AppFabric に接続すると、有用性の高いインサイトにアクセスしたり、好みのアプリケーションからクロスアプリケーションアクションを直接実行したりすることができるようになります。注: この機能は各アプリで確実に使用できるわけではなく、使用できるかどうかは、アプリケーションデベロッパーが AppFabric for productivity のどの機能を有効にしているかに依存します。

クロスアプリケーションインサイト

AppFabric for productivity には次の 2 種類のインサイトがあります。

- 実行可能なインサイト: AppFabric が、接続したアプリケーション間を横断して E メール、カレンダーイベント、タスク、メッセージ等の情報を分析し、ユーザーが優先すべき重要なキーインサイトを生成します。さらに、推奨されるアクション (メールの送信、会議のスケジュール設定、タスクの作成など) が生成されることもあります。これらのアクションは、好みのアプリケーションを開いたまま、編集および実行することが可能です。例えば、対処すべきカスタマーエスカレーションがあります、推奨されるアクションはこちらです、といったインサイトを受け取って、顧客とのミーティングをスケジュールする、といったことが行えます。
- 会議の準備に関するインサイト: この機能を使うと、今後の会議に向けて準備することができます。AppFabric が今後の会議を分析し、会議の目的を簡潔にまとめます。さらに、接続されているアプリケーションから、関連するアーティファクト (E メール、メッセージ、タスクなど) が表示されます。アプリケーションを切り替えてコンテンツを探す手間が省け、会議の準備を効率的に進めることができます。

クロスアプリケーションアクション

特定のインサイトではさらに、AppFabric で、推奨されるアクション (メールの送信、会議のスケジュール設定、タスクの作成など) を生成することもできます。アクションを生成するときに、接続済みアプリケーションのコンテンツとコンテキストに基づいて特定のフィールドを自動入力することができます。例えば、インサイトに基づいて、推奨される返信メールやタスクの名称を生成するなどです。推奨されるアクションをクリックすると、AppFabric が所有するユーザーインターフェイスが表示され、自動入力されたコンテンツは、アクションを実行する前にここで編集することができます。生成 AI と、基盤となる大規模言語モデル (LLM) はハルシネーションを起こすことがあるため、AppFabric は、ユーザーが事前に確認および入力を行っていないアクションは実行しません。

> **Note**
>
> ユーザーは、AppFabric LLM の出力を検証し確定する責任を負います。AppFabric は LLM の出力の精度や品質を保証しません。詳細については、「AWS Responsible AI Policy」を参照してください。

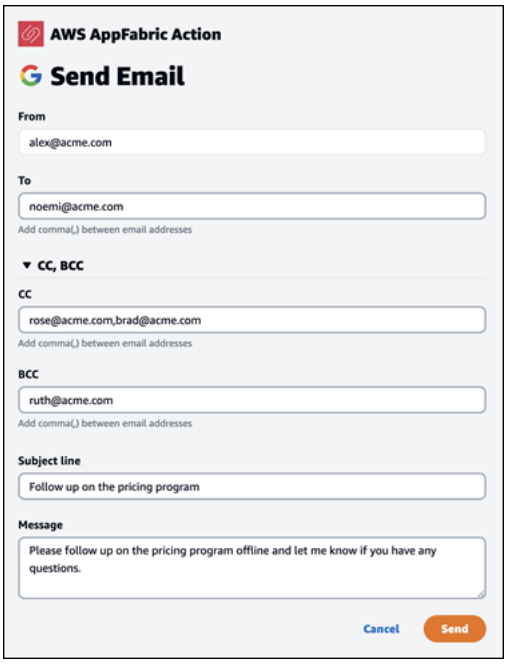
E メールの作成 (Google Workspace、Microsoft 365)

AppFabric では、ユーザーは、好みのアプリケーション内でメールを編集し送信することができます。差出人、宛先、Cc/Bcc、メールの件名、メール本文など、基本のフィールドはサポートされています。AppFabric では、タスクの完了に要する時間を短縮するため、これらのフィールドにコンテ

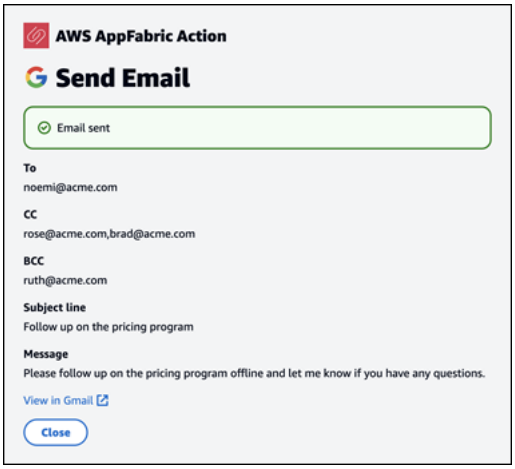
ンツを入力することがあります。E メールの編集が完了したら、[送信] を選択して E メールを送信します。

メールを送信するには、以下のフィールドへの入力は必須です。

- 1 つ以上の受信者の E メールアドレス (To、CC、BCC)。有効な E メールアドレスである必要があります。
- 件名と本文のフィールド。



E メールが送信されると、送信されたことを示す確認画面が表示されます。さらに、指定されたアプリケーションで E メールを表示するためのリンクが表示されます。このリンクをクリックするとアプリケーションにすばやく移動でき、メールが送信されたことを確認できます。

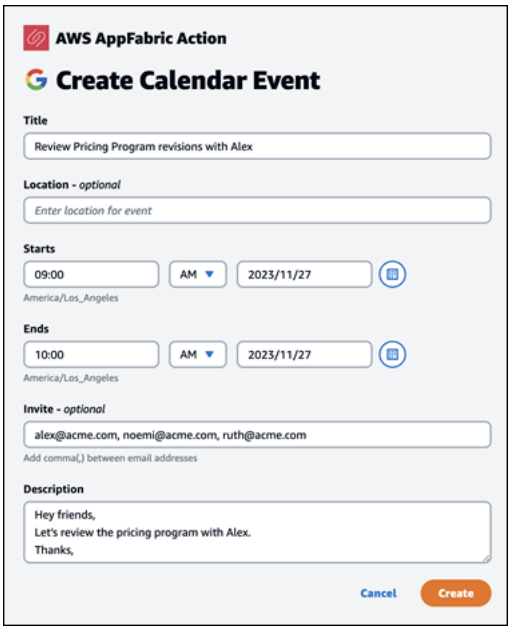


カレンダーイベントの作成 (Google Workspace、Microsoft 365)

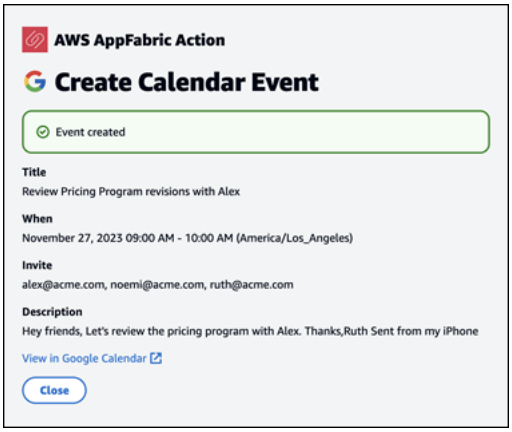
AppFabric では、ユーザーは、好みのアプリケーション内でカレンダーを編集したり作成したりできます。基本のカレンダーイベントフィールド (イベントのタイトル、場所、開始/終了日時、招待者リスト、イベントの内容など) がサポートされています。AppFabric では、タスクの完了に要する時間を短縮するため、これらのフィールドにコンテンツを入力することがあります。カレンダーイベントの編集が完了したら、[作成] を選択してイベントを作成します。

カレンダーイベントを作成するときは、以下のフィールドへの入力は必須です。

- タイトル、開始日時、終了日時、イベントの内容。
- 開始日時は終了日時より前にすることはできません。
- 招待フィールドは任意ですが、入力する場合は有効な E メールアドレスを入力する必要があります。



カレンダーイベントを送信すると、イベントが作成されたことを示す確認画面が表示されます。さらに、指定されたアプリケーションでイベントを表示するためのリンクが表示されます。このリンクをクリックするとアプリケーションにすばやく移動でき、イベントが作成されたことを確認できます。

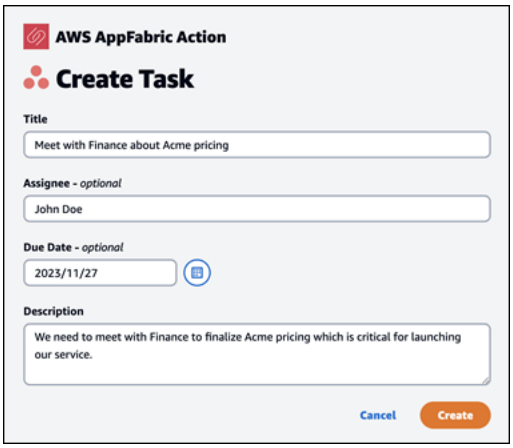


タスクの作成 (Asana)

AppFabric では、ユーザーは、好みのアプリケーション内から Asana のタスクを編集したり作成したりできます。基本のタスクフィールド (タスク名、タスク所有者、期日、タスクの内容など) がサポートされています。AppFabric では、タスクの作成に要する時間を短縮するため、これらのフィールドにコンテンツを入力することがあります。タスクの編集が完了したら、[作成] を選択してタスクを作成します。タスクは、LLM の推奨に従って、該当する Asana ワークスペース、プロジェクト、タスクのいずれかに作成されます。

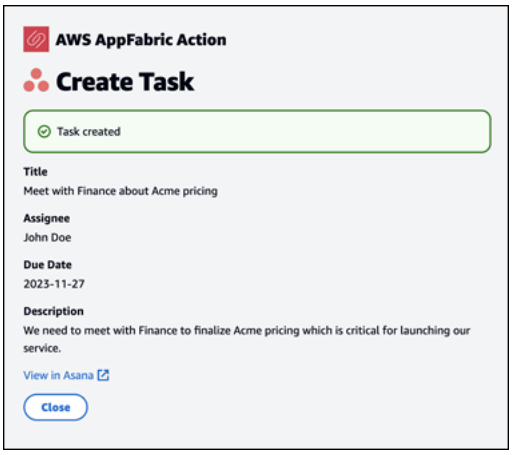
Asana タスクを作成するときは、以下のフィールドへの入力は必須です。

- タイトルと内容。
- 変更された場合、担当者には有効な E メールアドレスを入力しなければなりません。



タスクが作成されると、Asana にタスクが作成されたことを示す確認画面が表示されます。さらに、Asana のタスクを確認できるリンクも表示されます。このリンクをクリックすると、アプリ

ケーションにすばやく移動でき、タスクが作成されたことを確認したり、タスクを該当する Asana ワークスペース、プロジェクト、またはタスクに移動させたりすることができます。

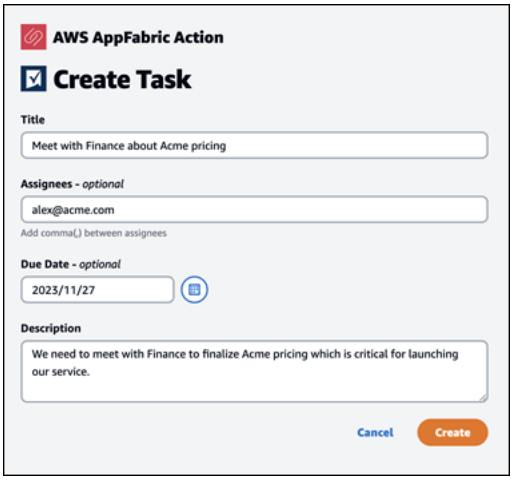


タスクの作成 (Smartsheet)

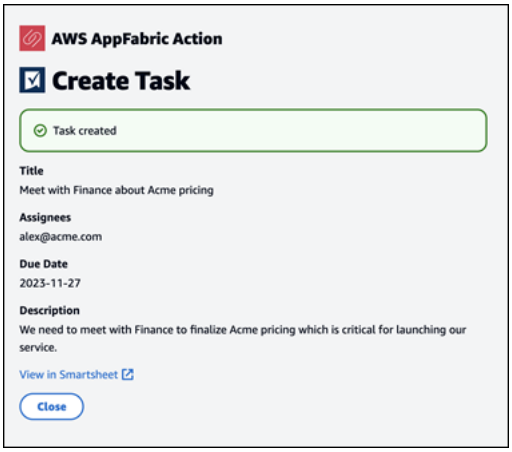
AppFabric では、ユーザーは、好みのアプリケーション内から Smartsheet のタスクを編集したり作成したりできます。基本のタスクフィールド (タスク名、タスク所有者、期日、タスクの内容など) がサポートされています。AppFabric では、タスクの作成に要する時間を短縮するため、これらのフィールドにコンテンツを入力することがあります。タスクの編集が完了したら、[作成] を選択してタスクを作成します。Smartsheet のタスクでは、AppFabric は新しいプライベートの Smartsheet シートを作成し、作成済みのタスクをすべて入力します。これは、AppFabric が生成するアクションを、1 つの場所で体系的に一元化できるようにするために行われます。

Smartsheet タスクを作成するときは、以下のフィールドへの入力は必須です。

- タイトルと内容。
- 入力された場合、担当者には有効な E メールアドレスを入力しなければなりません。



タスクが作成されると、Smartsheet にタスクが作成されたことを示す確認画面が表示されます。さらに、Smartsheet のタスクを確認できるリンクも表示されます。このリンクをクリックすると、アプリケーションにすばやく移動して、作成された Smartsheet シートにあるタスクを確認できます。今後作成される Smartsheet のタスクは、すべてこのシートに入力されます。シートが削除されると、AppFabric が新しいシートを作成します。



IT およびセキュリティ管理者向けの AppFabric for productivity (プレビュー) 機能へのアクセスを管理する

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

AppFabric for productivity のユーザーポータルは、AppFabric for productivity (プレビュー版) 機能と連携している SaaS アプリケーションの、すべてのユーザーに公開されています。こうした生成 AI 機能へのアクセスを組織内で管理したいと考えている IT 管理者の方は、以下の方法を検討できます。

- ID プロバイダー (IdP) ログインを制限する: ID プロバイダー経由のログインアクセスをブロックすることで、生成 AI 機能へのユーザーアクセスを制御できます。
- 特定のアプリケーションの OAuth を無効にする: OAuth を無効にすることで、ダウンストリームで制限を実行できます。このアクションをとると、ユーザーは、OAuth 認証を必要とするアプリケーションを会社のワークスペースに接続できなくなります。

AppFabric for productivity のエンドユーザーエラーのトラブルシューティング

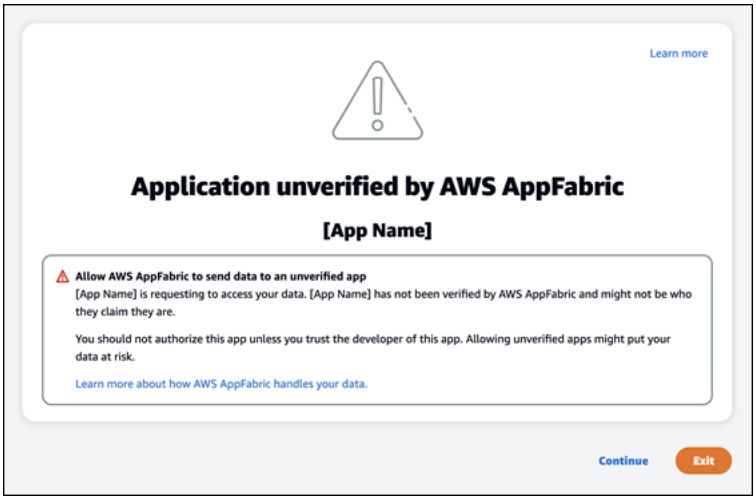
AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

このセクションでは、AppFabric for productivity のよくあるエラーとトラブルシューティングについて説明します。

未検証のアプリケーション

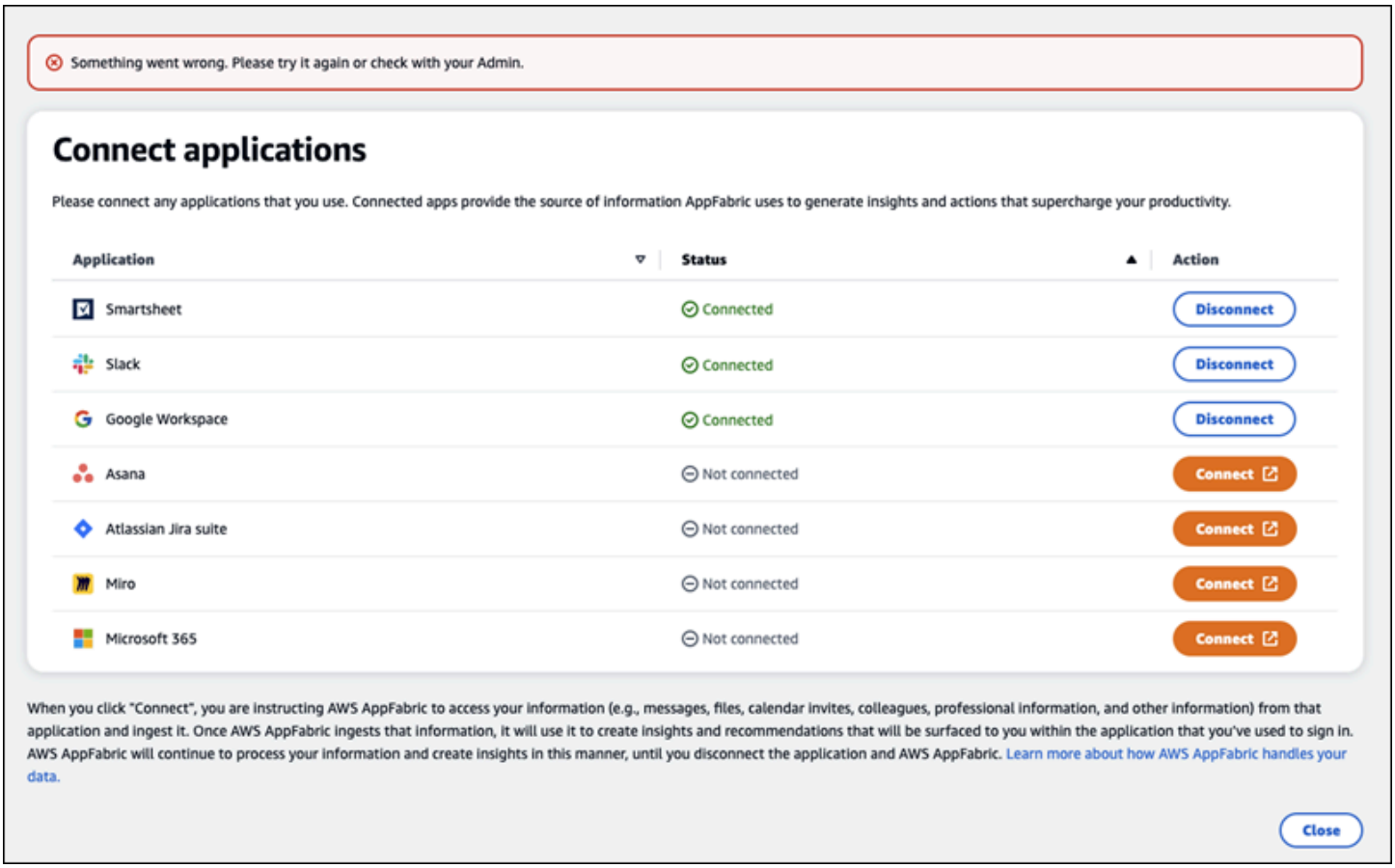
AppFabric for productivity を使用してエクスペリエンスを強化するアプリケーションは、エンドユーザーに機能を公開する前に検証プロセスを経る必要があります。AppFabric にサインインしようとしたときに [未検証] のバナーが表示された場合、そのアプリケーションは、デベロッパーの ID とアプリケーションの登録情報の正確性とを確認する AppFabric の検証プロセスが完了していないことを意味します。アプリケーションはすべて、未検証の状態からスタートし、検証プロセスが完了した場合のみ、検証済みの状態になります。

未検証のアプリケーションを使用するときは注意が必要です。アプリケーションの開発者が不明である場合は、ステータスが検証済みに変わってから使用を開始するようにします。



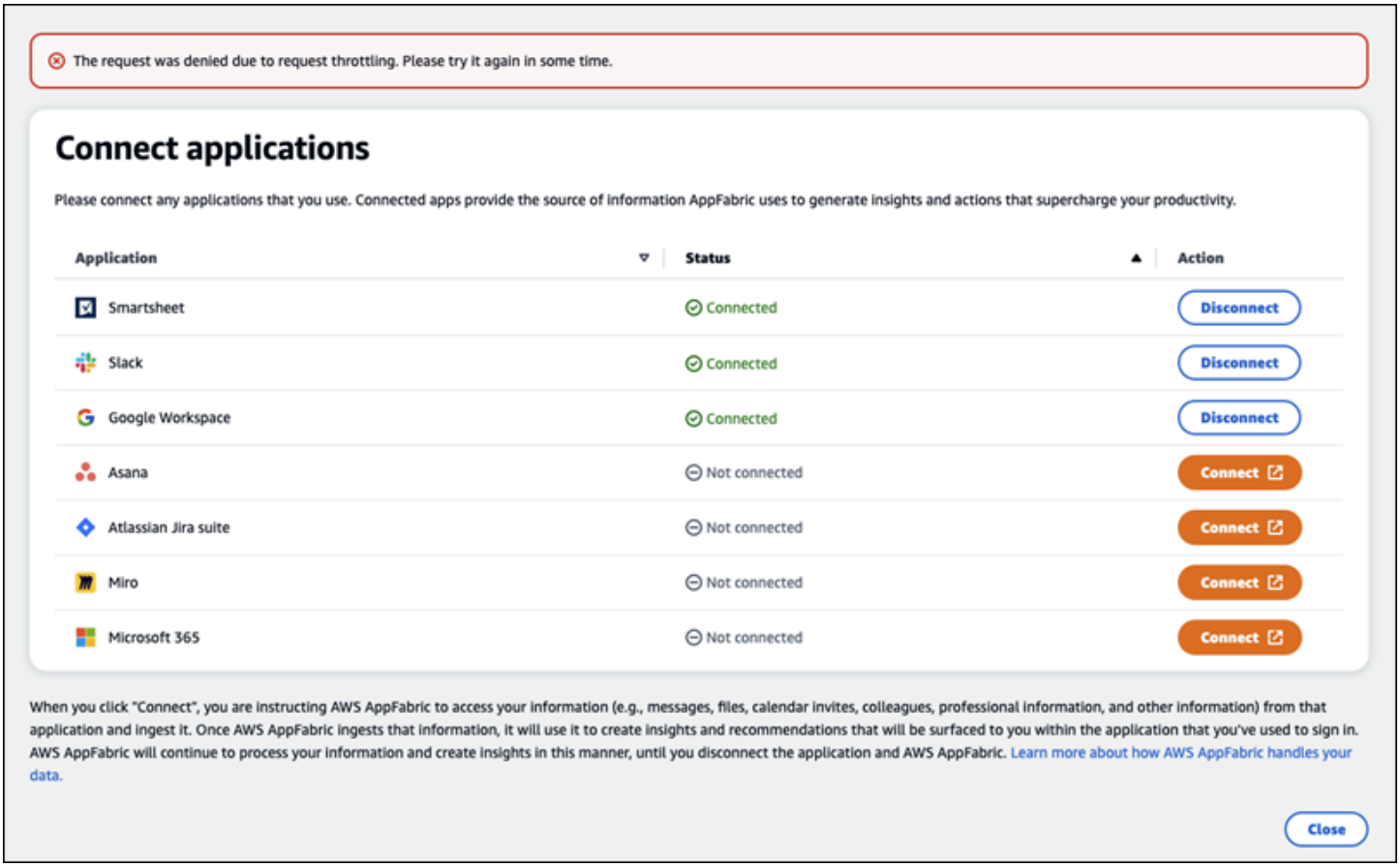
問題が発生しました。もう一度試すか、管理者に確認します (`InternalServerException`)。

このメッセージは、AppFabric のユーザーポータルがアプリケーションをリスト化できなかったり、不明なエラー、例外、障害などが原因でアプリケーションを切断したりした場合に表示されます。後でもう一度お試しください。



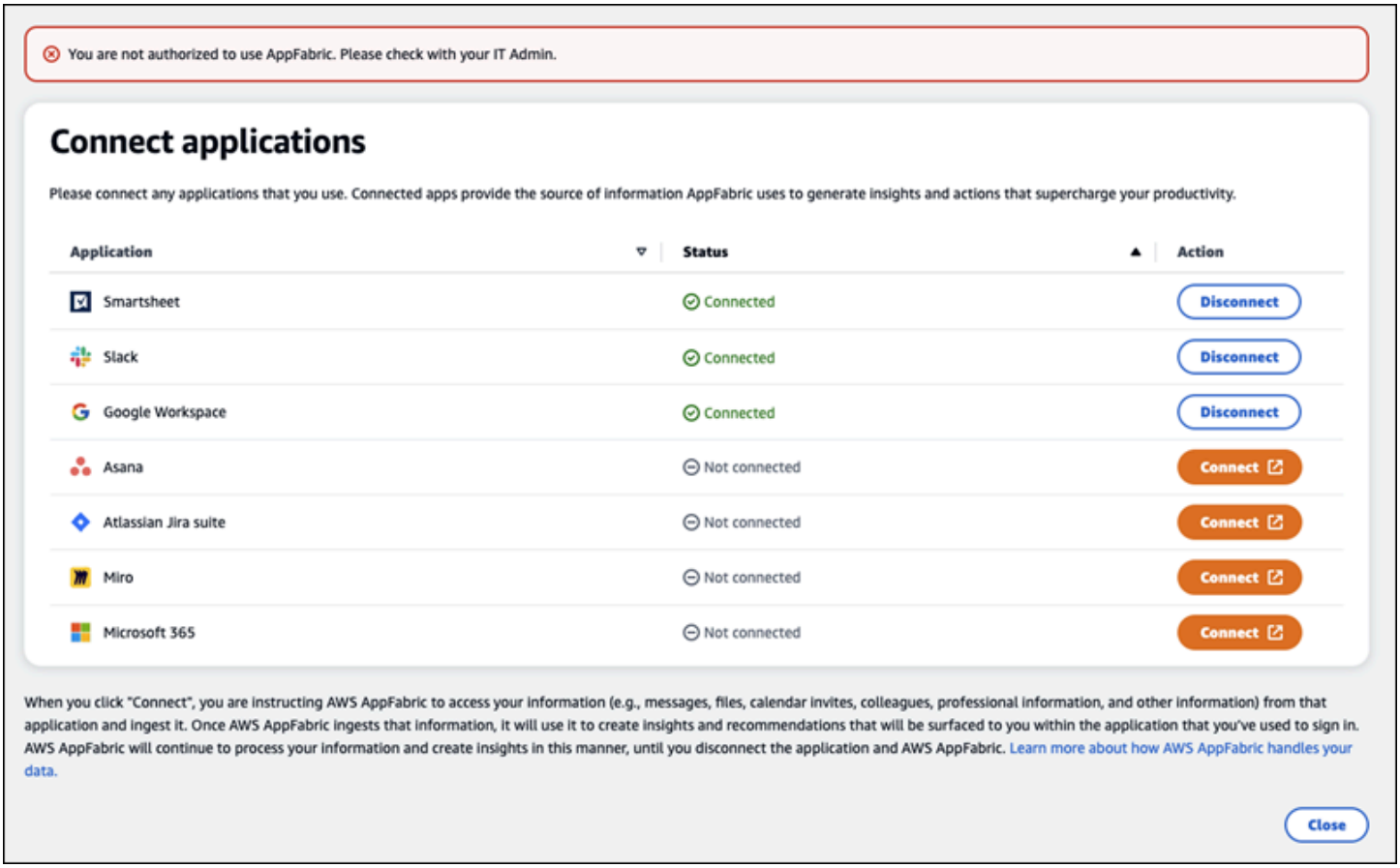
リクエストのスロットリングにより、リクエストが拒否されました。しばらくしてからもう一度試してください (`ThrottlingException`)。

このメッセージは、AppFabric のユーザーポータルがアプリケーションをリスト化できなかったり、スロットリングの問題が原因でアプリケーションを切断したりした場合に表示されます。後でもう一度お試しください。



AppFabric を使用する権限がありません。AppFabric に再度ログインしてください (**`AccessDeniedException`**)。

このメッセージは、AppFabric のユーザーポータルがアプリケーションをリスト化できなかったり、アクセス拒否の例外が原因でアプリケーションを切断したりした場合に表示されます。AppFabric に再度サインインします。



AppFabric for productivity APIsプレビュー)

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

このセクションでは、 AWS AppFabric 生産性向上機能の API オペレーション、データ型、一般的なエラーについて説明します。

> **Note**
>
> その他の AppFabric API については、「AWS AppFabric API Reference」を参照してください。

トピック

- AppFabric for productivity の API アクション (プレビュー)

- AppFabric for productivity の API データ型 (プレビュー)
- AppFabric for productivity の一般的な API エラー (プレビュー)

AppFabric for productivity の API アクション (プレビュー)

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

AppFabric for productivity 機能では、以下のアクションがサポートされています。

AppFabric API のその他のアクションについては、「AWS AppFabric API Actions」を参照してください。

トピック

- 承認
- CreateAppClient
- DeleteAppClient
- GetAppClient
- ListActionableInsights
- ListAppClients
- ListMeetingInsights
- PutFeedback
- トークン
- UpdateAppClient

承認

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

AppClient を認証します。

トピック

- リクエスト本文

リクエスト本文

リクエストは以下の JSON 形式のデータを受け入れます。

パラメータ	説明
app_client_id	承認する AppClient の ID です。
redirect_uri	承認後にエンドユーザーがリダイレクトされる URI です。
state	リクエストとコールバック間の状態を維持するための一意の値です。

CreateAppClient

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

AppClient を作成します。

トピック

- リクエスト本文
- レスポンス要素

リクエスト本文

リクエストは以下の JSON 形式のデータを受け入れます。

パラメータ	説明
appName	アプリケーションの名前。 タイプ: 文字列 長さの制約: 最小長は 1 です。最大長は 255 です。 必須: はい

パラメータ	説明
clientToken	リクエストの冪等性のために割り当てる一意の識別子 (大文字と小文字を区別) を指定します。これにより、同じ操作を誤って 2 度実行することなく、リクエストを安全に再試行できます。操作の後半の呼び出しで同じ値を渡す場合は、他のすべてのパラメータにも同じ値を渡す必要があります。UUID タイプの値を使用することが推奨されます。 この値を指定しない場合、 はランダムな値 AWS を生成します。 同じ `ClientToken` を使って、異なるパラメータで操作を再試行すると、再試行は `IdempotentParameterMismatch` のエラーにより失敗します。 タイプ: 文字列 パターン: `[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}` 必須: いいえ

パラメータ	説明
customerManagedKeyIdentifier	によって カスタマー管理キー 生成された の ARN AWS Key Management Service。このキーはデータの暗号化に使用します。 キーが指定されていない場合 AWS マネージドキー は、 が使用されます。リソースに割り当てるタグの、キーと値のペアのマップ。 AWS 所有のキー およびカスタマーマネージドキーの詳細については、「 AWS Key Management Service デベロッパーガイド」の「カスタマーキーと AWS キー」を参照してください。 タイプ: 文字列 長さの制約: 最小長は 1 です。最大長は 1,011 です。 パターン: `arn:.+$|^[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}` 必須: いいえ
[Description] (説明)	アプリケーションの説明です。 タイプ: 文字列 必須: はい
iconUrl	AppClient のアイコンまたはロゴの URL です。 タイプ: 文字列 必須: いいえ

パラメータ	説明
redirectUrls	承認後にエンドユーザーがリダイレクトされる URI です。redirectUrls は 5 個まで追加できます。例えば、`https://localhost:8080` 。 型: 文字列の配列 配列メンバー: 最小数は 1 項目です。最大数は 5 項目です。 長さの制限：最小長 1、最大長は 2,048 です。 パターン: `(http|https):\/\/[-a-zA-Z0-9_:.\/]+` 必須: はい
starterUserEmails	AppClient が検証されるまでインサイトを受け取ることが許可されている、ユーザーのためのスターターメールアドレス。 型: 文字列の配列 配列メンバー: 1 項目の定数です。 長さの制約: 最小長は 0 です。最大長は 320 です。 パターン: `[a-zA-Z0-9.!#$%&'*+/=?^_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)*` 必須: はい
[タグ]	リソースに割り当てるタグの、キーと値のペアのマップ。 タイプ: タグオブジェクトの配列 配列メンバー: 最小数は 0 項目です。最大数は 50 項目です。 必須: いいえ

レスポンス要素

アクションが成功すると、HTTP 201 レスポンスが返されます。

サービスから以下のデータが JSON 形式で返されます。

パラメータ	説明
appClientSummary	AppClient の概要が含まれています。 タイプ:AppClientSummary オブジェクト

DeleteAppClient

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

アプリケーションクライアントを削除します。

トピック

- リクエスト本文
- レスポンス要素

リクエスト本文

リクエストは以下の JSON 形式のデータを受け入れます。

パラメータ	説明
appClientIdentifier	リクエストに使用する AppClient の Amazon リソースネーム (ARN) または Universal Unique Identifier (UUID) です。 長さの制限：最小長 1、最大長は 1,011 です。 パターン: `arn:.+$|^[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}` 必須: はい

レスポンス要素

アクションが成功した場合、サービスは空の HTTP 本文を持つ HTTP 204 レスポンスを返します。

GetAppClient

> AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

AppClient に関する情報を返します。

トピック

- リクエスト本文
- レスポンス要素

リクエスト本文

リクエストは以下の JSON 形式のデータを受け入れます。

パラメータ	説明
appClientIdentifier	リクエストに使用する AppClient の Amazon リソースネーム (ARN) または Universal Unique Identifier (UUID) です。 長さの制限：最小長 1、最大長は 1,011 です。 パターン: `arn:.+$|^[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}` 必須: はい

レスポンス要素

アクションが成功すると、サービスは HTTP 200 レスポンスを返します。

サービスから以下のデータが JSON 形式で返されます。

パラメータ	説明
appClient	AppClient に関する情報が含まれます。 タイプ: AppClient オブジェクト

ListActionableInsights

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

実行可能な E メールメッセージ、タスク、その他の更新の最も重要なものを一覧表示します。

トピック

- リクエスト本文
- レスポンス要素

リクエスト本文

リクエストは以下の JSON 形式のデータを受け入れます。

パラメータ	説明
nextToken	`nextToken` が返された場合、その他にもまだ結果があります。`nextToken` の値は、各ページに固有のページネーショントークンです。後続ページを取得するには、返されたトークンを使用して再度呼び出します。他の引数をすべて維持します。各ページネーショントークンの有効期間は 24 時間です。期限の切れたページネーショントークンを使用すると、HTTP 400 InvalidToken エラーが返されます。

レスポンス要素

アクションが成功すると、HTTP 201 レスポンスが返されます。

サービスから以下のデータが JSON 形式で返されます。

パラメータ	説明
ActionableInsightsList	件名、説明、アクション、作成済みのタイムスタンプなど実行可能なインサイトを一覧表示します。詳細については、「ActionableInsights」を参照してください。
nextToken	`nextToken` が返された場合、その他にもまだ結果があります。`nextToken` の値は、各ページに固有のページネーショントークンです。後続ページを取得するには、返されたトークンを使用して再度呼び出します。他の引数をすべて維持します。各ページネーショントークンの有効期間は 24 時間です。期限の切れたページネーショントークンを使用すると、HTTP 400 InvalidToken エラーが返されます。 タイプ: 文字列

ListAppClients

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

すべての AppClients のリストを返します。

トピック

- リクエスト本文
- レスポンス要素

リクエスト本文

リクエストは以下の JSON 形式のデータを受け入れます。

パラメータ	説明
maxResults	コールごとに返される結果の最大数です。`nextToken` を使用すると結果ページをさらに取得できます。

パラメータ	説明
	こちらはあくまでも上限です。1 回のコールで返される実際の結果が、指定の最大数より少なくなる場合もあります。 有効範囲: 最小値 1 最大値は 100 です。
nextToken	`nextToken` が返された場合、その他にもまだ結果があります。`nextToken` の値は、各ページに固有のページネーショントークンです。後続ページを取得するには、返されたトークンを使用して再度呼び出します。他の引数をすべて維持します。各ページネーショントークンの有効期間は 24 時間です。期限の切れたページネーショントークンを使用すると、HTTP 400 InvalidToken エラーが返されます。

レスポンス要素

アクションが成功すると、サービスは HTTP 200 レスポンスを返します。

サービスから以下のデータが JSON 形式で返されます。

パラメータ	説明
appClientList	AppClient の結果のリストが含まれています。 タイプ: AppClientSummary オブジェクトの配列。
nextToken	`nextToken` が返された場合、その他にもまだ結果があります。`nextToken` の値は、各ページに固有のページネーショントークンです。後続ページを取得するには、返されたトークンを使用して再度呼び出します。他の引数をすべて維持します。各ページネーショントークンの有効期間は 24 時間です。期限の切れたページネーショントークンを使用すると、HTTP 400 InvalidToken エラーが返されます。 タイプ: 文字列

ListMeetingInsights

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

実行可能なカレンダーイベントの最も重要なものを一覧表示します。

トピック

- リクエスト本文
- レスポンス要素

リクエスト本文

リクエストは以下の JSON 形式のデータを受け入れます。

パラメータ	説明
nextToken	`nextToken` が返された場合、その他にもまだ結果があります。`nextToken` の値は、各ページに固有のページネーショントークンです。後続ページを取得するには、返されたトークンを使用して再度呼び出します。他の引数をすべて維持します。各ページネーショントークンの有効期間は 24 時間です。期限の切れたページネーショントークンを使用すると、HTTP 400 InvalidToken エラーが返されます。

レスポンス要素

アクションが成功すると、HTTP 201 レスポンスが返されます。

サービスから以下のデータが JSON 形式で返されます。

パラメータ	説明
MeetingInsightList	会議に関する実行可能なインサイトを一覧表示します。詳細については、「MeetingInsights」を参照してください。

パラメータ	説明
nextToken	`nextToken` が返された場合、その他にもまだ結果があります。`nextToken` の値は、各ページに固有のページネーショントークンです。後続ページを取得するには、返されたトークンを使用して再度呼び出します。他の引数をすべて維持します。各ページネーショントークンの有効期間は 24 時間です。期限の切れたページネーショントークンを使用すると、HTTP 400 InvalidToken エラーが返されます。 タイプ: 文字列

PutFeedback

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

特定のインサイトまたはアクションに関するフィードバックを送ることをユーザーに許可します。

トピック

- リクエスト本文
- レスポンス要素

リクエスト本文

リクエストは以下の JSON 形式のデータを受け入れます。

パラメータ	説明
id	フィードバックの送信対象となるオブジェクトの ID です。InsightId か ActionId のいずれかになります。
feedbackFor	フィードバックの送信対象となるインサイトのタイプです。 使用できる値: `ACTIONABLE_INSIGHT` | `MEETING_INSIGHT` | `ACTION`

パラメータ	説明
feedbackRating	1 から 5 までの評価です。値が大きいほど評価が高いことを意味します。

レスポンス要素

アクションが成功した場合、サービスは空の HTTP 本文を持つ HTTP 201 レスポンスを返します。

トークン

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

AppClients が認可コードをアクセストークンと交換できるようにする情報が含まれます。

トピック

- リクエスト本文
- レスポンス要素

リクエスト本文

リクエストは以下の JSON 形式のデータを受け入れます。

パラメータ	説明
コード	認可エンドポイントから受信した認可コードです。 タイプ: 文字列 長さの制約: 最小長は 1 です。最大長は 2,048 です。 必須: いいえ
grant_type	トークンの付与のタイプ。`authorization_code` または `refresh_token` である必要があります。 タイプ: 文字列

パラメータ	説明
	必須: はい
app_client_id	AppClient の ID です。 タイプ: 文字列 パターン: `[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}` 必須: はい
redirect_uri	認可エンドポイントに渡されたリダイレクト URI。 タイプ: 文字列 必須: いいえ
refresh_token	最初のトークンリクエストで受け取った更新トークンです。 タイプ: 文字列 長さの制約: 最小長は 1 です。最大長は 4,096 です。 必須: いいえ

レスポンス要素

アクションが成功すると、サービスは HTTP 200 レスポンスを返します。

サービスから以下のデータが JSON 形式で返されます。

パラメータ	説明
appfabric_user_id	トークン用のユーザーの ID。この値は、リクエストが `authorization_code` グラントタイプを使用している場合のみ返されます。 タイプ: 文字列

パラメータ	説明
expires_in	トークンの有効期限が切れるまでの残りの秒数。 型: 長整数
refresh_token	次のリクエストに使用する更新トークンです。 タイプ: 文字列 長さの制約: 最小長は 1 です。最大長は 2,048 です。
token (トークン)	アクセストークンです。 タイプ: 文字列 長さの制約: 最小長は 1 です。最大長は 2,048 です。
token_type	トークンのタイプです。 タイプ: 文字列

UpdateAppClient

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

AppClient を更新します。

トピック

- リクエスト本文
- レスポンス要素

リクエスト本文

リクエストは以下の JSON 形式のデータを受け入れます。

パラメータ	説明
appClientIdentifier	リクエストに使用する AppClient の Amazon リソースネーム (ARN) または Universal Unique Identifier (UUID) です。 長さの制限：最小長 1、最大長は 1,011 です。 パターン: `arn:.+$|^[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}` 必須: はい
redirectUrls	承認後にエンドユーザーがリダイレクトされる URI です。redirectUrls は 5 個まで追加できます。例えば、`https://localhost:8080` 。 型: 文字列の配列 配列メンバー: 最小数は 1 項目です。最大数は 5 項目です。 長さの制限：最小長 1、最大長は 2,048 です。 パターン: `(http|https):\/\/[-a-zA-Z0-9_:.\/]+`

レスポンス要素

アクションが成功すると、サービスは HTTP 200 レスポンスを返します。

サービスから以下のデータが JSON 形式で返されます。

パラメータ	説明
appClient	AppClient に関する情報が含まれます。 タイプ: AppClient オブジェクト

AppFabric for productivity の API データ型 (プレビュー)

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

AppFabric API には、さまざまなアクションに使用される複数のデータ型が含まれています。このセクションでは、AppFabric for productivity 機能のデータ型について詳しく説明します。

AppFabric API のその他のデータ型については、「AWS AppFabric API Data Types」を参照してください。

> **Important**
> データ型構造内の各要素の順序は保証されません。アプリケーションは特定の順序を想定するべきではありません。

トピック

- ActionableInsights
- AppClient
- AppClientSummary
- MeetingInsights
- VerificationDetails

ActionableInsights

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

アプリケーションポートフォリオの E メール、カレンダーの招待、メッセージ、タスクに基づく、ユーザーにとって重要かつ最も適したアクションの概要が含まれます。ユーザーは、すべてのアプリケーションを横断する先を見越したインサイトを確認でき、その日の最適な進め方を確認することができます。これらのインサイトを見れば、ユーザーが、インサイトのサマリーだけでなく、インサイトを生成した、各アプリケーションやアーティファクトなどの参照情報 (埋め込みリンクなど) にも目を向けるべき理由がわかります。

パラメータ	説明
insightId	生成されたインサイトの一意の ID です。
insightContent	インサイトの概要と、インサイトの生成に使用されたアーティファクトへの埋め込みリンクを返します。 こちらは、埋め込みリンク (<a> タグ) を含む HTML コンテンツです。
insightTitle	生成されたインサイトの件名です。
createdAt	インサイトが生成された日時です。
actions	生成されたインサイトで推奨されるアクションのリストです。 アクションオブジェクトには、以下のパラメータが含まれています。 • `actionId` - 生成されたアクションの一意の ID です。 • `actionIconUrl` - アクションの実行が推奨されているアプリケーションのアイコン URL です。 • `actionTitle` - 生成されたアクションの件名です。 • `actionUrl` - エンドユーザーが AppFabric のユーザーポータルでアクションを表示して実行するための一意の URL です。 アクションを実行する場合、ISV のアプリケーションはこの URL を使用してユーザーを AppFabric ユーザーポータル (ポップアップ画面) にリダイレクトします。 • `actionExecutionStatus` - アクションのステータスを示す列挙型です。 指定できる値: `EXECUTED` | `NOT_EXECUTED`

AppClient

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

AppClient に関する情報が含まれます。

パラメータ	説明
appName	アプリケーションの名前。 タイプ: 文字列 必須: はい
arn	AppClient の Amazon リソースネーム (ARN) です。 タイプ: 文字列 長さの制約: 最小長は 1 です。最大長は 1,011 です。 パターン: `arn:.+` 必須: はい
[Description] (説明)	アプリケーションの説明です。 タイプ: 文字列 必須: はい
iconUrl	AppClient のアイコンまたはロゴの URL です。 タイプ: 文字列 必須: いいえ
redirectUrls	AppClient で許可されているリダイレクト URL。 型: 文字列の配列 配列メンバー: 最小数は 1 項目です。最大数は 5 項目です。

パラメータ	説明
	長さの制限：最小長 1、最大長は 2,048 です。 パターン: `(http|https):\/\/[-a-zA-Z0-9_:.\/]+` 必須: はい
starterUserEmails	AppClient が検証されるまでインサイトを受け取ることが許可されている、ユーザーのためのスターターメールアドレス。 型: 文字列の配列 配列メンバー: 1 項目の定数です。 長さの制約: 最小長は 0 です。最大長は 320 です。 パターン: `[a-zA-Z0-9.!#$%&'*+/=?^_`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)*` 必須: はい
verificationDetails	AppClient 検証のステータスと理由が含まれます。 タイプ: VerificationDetails オブジェクト 必須: はい
customerManagedKeyArn	AppClient AWS Key Management Service 用に によって生成された の カスタマー管理キー Amazon リソースネーム (ARN)。 タイプ: 文字列 長さの制約: 最小長は 1 です。最大長は 1,011 です。 パターン: `arn:.+` 必須: いいえ

パラメータ	説明
appClientId	AppClient の ID です。app-client の o-auth フローで使用される手段。 タイプ: 文字列 パターン: `[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}` 必須: いいえ

AppClientSummary

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

AppClient に関する情報が含まれます。

パラメータ	説明
arn	AppClient の Amazon リソースネーム (ARN) です。 タイプ: 文字列 長さの制約: 最小長は 1 です。最大長は 1,011 です。 パターン: `arn:.+` 必須: はい
verificationStatus	AppClient の検証ステータスです。 型: 文字列 有効な値：`pending_verification | verified | rejected` 必須: はい

パラメータ	説明
appClientId	AppClient の ID です。app-client の o-auth フローで使用される手段。 タイプ: 文字列 パターン: `[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}` 必須: いいえ

MeetingInsights

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

上位 3 件の会議の概要と、会議の目的、関連するクロスアプリケーションアーティファクト、また、タスク、E メール、メッセージ、カレンダーイベントのアクティビティが含まれます。

パラメータ	説明
insightId	生成されたインサイトの一意の ID です。
insightContent	インサイトの説明で、詳細が文字列の形式で強調表示されています。例えば、なぜこのインサイトが重要なのか、など。
insightTitle	生成されたインサイトの件名です。
createdAt	インサイトが生成された日時です。
calendarEvent	ユーザーが注意すべき重要なカレンダーイベントまたは会議です。 カレンダーイベントオブジェクト: • `startTime` - イベントの開始時刻です。 • `endTime` - イベントの終了時刻です。

パラメータ	説明
	• `eventUrl` - ISV アプリケーションのカレンダーイベントの URL です。
resources	インサイトの生成に関連する他のリソースを含むリストです。 リソースオブジェクト: • `appName` - リソースが属するアプリケーションの名前です。 • `resourceTitle` - リソースの件名です。 • `resourceType` - リソースのタイプです。 指定できる値は以下のとおりです。`EMAIL | EVENT | MESSAGE | TASK` • `resourceUrl` - アプリケーション内のリソース URL です。 • `appIconUrl` - リソースが属するアプリケーションの画像 URL です。
nextToken	次回のインサイトのセットを取得するためのページネーショントークンです。こちらはオプションのフィールドで、null が返された場合は、ロードするインサイトがそれ以上ないことを意味します。

VerificationDetails

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

AppClient 検証のステータスと理由が含まれます。

パラメータ	説明
verificationStatus	AppClient の検証ステータスです。 型: 文字列

パラメータ	説明
	有効な値：`pending_verification | verified | rejected` 必須: はい
statusReason	AppClient の検証ステータスの理由です。 タイプ: 文字列 長さの制限: 最小長は 1 です。最大長は 1,024 です。 必須: いいえ

AppFabric for productivity の一般的な API エラー (プレビュー)

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

このセクションでは、 AWS AppFabric 生産性向上機能の API アクションに共通するエラーを一覧表示します。

AppFabric のその他の一般的な API エラーについては、「AWS AppFabric API Reference」の AppFabric for productivity の AppClients AppClients のトラブルシューティング および「AWS AppFabric API common errors」を参照してください。

例外名	説明
TokenException	トークンリクエストは無効です。 HTTP ステータスコード: 400

AppFabric でのデータ処理

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

AppFabric では、ユーザーコンテンツを個別に、AppFabric が管理する Amazon S3 バケットにそれぞれ分けて保存する手順を取っています。そうすることで、各ユーザーに固有のインサイトを生成することができます。ユーザーのコンテンツは、保管時および転送中の暗号化を含め、合理的な手段を講じて保護しています。お客様のコンテンツは、取り込まれてから 30 日以内に自動的に削除されるようにシステムで設定されています。AppFabric は、ユーザーがアクセスできなくなったデータアーティファクトを使用して、インサイトを生成することはありません。例えば、ユーザーがデータソース (アプリケーション) を切断した場合、AppFabric はそのアプリケーションからデータを収集することを停止し、切断したアプリケーションに残っているアーティファクトを使ってインサイトを生成することはしません。AppFabric のシステムは、こうしたデータを 30 日以内に削除するように設定されています。

AppFabric が、インサイトの生成に使用される、基盤となる大規模言語モデルのトレーニングや改善に、ユーザーのコンテンツを使用することはありません。AppFabric の生成 AI 機能の詳細については、「Amazon Bedrock よくある質問」を参照してください。

保管中の暗号化

AWS AppFabric は、保管時の暗号化をサポートしています。これは、AppFabric がディスクに保持されているときにユーザーに関連するすべてのデータを透過的に暗号化し、データにアクセスするときに復号するサーバー側の暗号化機能です。

転送中の暗号化

AppFabric は、TLS 1.2 を使用して転送中のすべてのコンテンツを保護し、 AWS 署名バージョン 4 で AWS サービスの API リクエストに署名します。

AppFabric の用語と概念

このトピックでは、使用開始に役立つ AWS AppFabric の主要な用語と概念について説明します。

アプリバンドル

AppFabric のアプリバンドルには、AppFabric アプリのすべての承認と取り込みが保存されます (以下の取り込みの定義を参照してください)。 AWS アカウント ごとに 1 つのアプリバンドルを作成できます AWS リージョン。

AppClient (アプリクライアントとアプリケーションクライアントも含む)

データ受信者アプリ用の OAuth AppClient です。各データ受信者アプリは、AppFabric データにアクセスするために AppClient を登録する必要があります。デベロッパーユーザーには、AppClient を登録するための AWS アカウントが必要です。各 AWS アカウントで登録できる AppClient は 1 つだけです。AppFabric は AppClient に基づいてアクセストークンを配布します。AppClient には、この AppClient を介して AppFabric データにアクセスするデータ受信者アプリに関する情報が含まれています。

アプリ認可

アプリ認可は、アプリケーションに接続して操作するためのアクセス許可を AppFabric に付与します。これにより、OAuth (Open Authorization - アプリケーションにアクセス権を付与するためのアクセス委任のオープン標準) または個人アクセストークン (PAT) 認証情報を使用して、アプリケーションから監査ログを取り込むことができます。アプリバンドルごとに複数のアプリ認証 (最大 50 件) を設定できます。これにより、AppFabric は、アプリケーションの各テナントで必要に応じてアプリ認可の作成手順を繰り返すことで、アプリケーションの複数のテナントから監査ログを取り込むことができます。共有される認証情報は、 AWS Key Management Service （ AWS KMS) の AWS 所有のキー またはカスタマーマネージドキーで暗号化され、AppFabric に保存されます。

取り込み

AppFabric 取り込みでは、アプリケーション認可を使用して、アプリケーションのパブリック APIs を介してアプリケーションから監査ログを取得します。続いて、監査ログが 1 件以上 (最大 5 件) の取り込み先に転送されます。

クライアント ID

OAuth フローを使用するアプリケーションに接続するためのアプリ認可を作成すると、 AppFabric からクライアント ID とクライアントシークレットの入力を求められる場合があります。クライアン

ト ID とクライアントシークレットは、アプリケーションの認証アプリにあります。特定の認証アプリ内のクライアント ID を確認する方法については、「サポートされているアプリケーション」を参照してください。共有されるクライアント ID とクライアントシークレットは、 AWS 所有のキー またはカスタマーマネージド AWS KMS キーで暗号化され、AppFabric に保存されます。

クライアントシークレット

OAuth フローを使用するアプリケーションに接続するためのアプリ認可を作成すると、 AppFabric からクライアント ID とクライアントシークレットの入力を求められる場合があります。クライアント ID とクライアントシークレットは、アプリケーションの認証アプリにあります。特定の認証アプリ内のクライアントシークレットを確認する方法については、「サポートされているアプリケーション」を参照してください。共有されるクライアント ID とクライアントシークレットは、 AWS 所有のキー またはカスタマーマネージド AWS KMS キーで暗号化され、AppFabric に保存されます。

取り込み先

取り込み先は、取り込みから取得した監査ログの保存場所を定義します。各取り込みは、 内の Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose である 1 つ以上の送信先 (最大 5 つ) に監査ログを配信できます AWS アカウント。取り込み先ごとに、ログを raw 形式にするか、オープンサイバーセキュリティスキーマフレームワーク (OCSF) スキーマに正規化するかを定義できます。OCSF スキーマを選択すると、ログの形式 (JSON または Apache Parquet) を定義できます。Apache Parquet 形式は、Amazon S3 が取り込み先として選択されている場合にのみ使用できます。

データ受信者アプリ

AppFabric を呼び出して、生成されたインサイトを AppFabric から取得するアプリケーションです。

OAuth

OAuth はオープンプロトコルで、ウェブ、モバイル、デスクトップアプリケーションのシンプルで標準的な方法による安全な認可を可能にします。AppFabric は OAuth を使用して一部のアプリ認証を作成します。

オープンサイバーセキュリティスキーマフレームワーク (OCSF)

オープンサイバーセキュリティスキーマフレームワーク (OCSF) は、ベンダーに依存しないコアセキュリティスキーマと並んで、スキーマを開発するための拡張可能なフレームワークを提供するオープンソースプロジェクトです。ベンダーやその他のデータプロデューサーは、このスキーマを特定のドメインに採用したり拡張したりできます。その目標は、既存のセキュリティ標準やプロセスを補完

しながら、あらゆる環境、アプリケーション、ソリューションで採用されるオープン標準を提供することです。AppFabric はこのスキーマを拡張して、Software as a Service (SaaS) 志向のイベント構造を作成しました。AppFabric がサポートするすべての SaaS アプリ監査ログはこの構造に基づいて正規化されます。詳細については、「AWS AppFabric 用のオープンサイバーセキュリティスキーマフレームワーク」を参照してください。

個人アクセストークン (PAT)

個人アクセストークン (PAT) は、通常のパスワードの代わりにコンピューターシステムへのアクセスに使用できる文字列です。OAuth フローを使用するアプリケーションに接続するためのアプリ認可を作成すると、 AppFabric から PAT を求められる場合があります。PAT は、アプリケーションの認証アプリにあります。特定の認証アプリで PAT の場所を確認する方法については、「サポートされているアプリケーション」を参照してください。共有されるサービスアカウントトークンは、AWS 所有のキー またはカスタマーマネージド AWS KMS キーで暗号化され、AppFabric に保存されます。

サービスアカウントトークン

アプリケーションに接続するための AppFabric アプリ認可を作成する場合、一部のアプリケーションではアプリケーション認証用のサービスアカウントの作成が必要になります。AppFabric は、アプリケーション認可プロセスの一環としてサービスアカウントトークンを要求する場合があります。特定の認証アプリ内のサービスアカウントトークンの場所については、「サポートされているアプリケーション」を参照してください。共有されるサービスアカウントトークンは、 AWS 所有のキーまたはカスタマーマネージド AWS KMS キーで暗号化され、AppFabric に保存されます。

テナント ID

アプリ認可を作成すると、AppFabric はアプリのテナント ID とテナント名の入力を求める場合があります。テナント ID は、アプリケーションテナントの一意の識別子です。アプリケーションごとに、Slack の Workspace ID 、または Asana のドメインID など、テナントに対して使用する用語が異なる場合があります。特定のアプリケーションのテナント ID の場所を確認する方法については、「サポートされているアプリケーション」を参照してください。

テナント名

アプリ認可を作成すると、AppFabric はアプリのテナント ID とテナント名の入力を求める場合があります。テナント名はテナント ID に与える一意の名前で、アプリバンドル内で使用されます。この値は、アプリ認可とそれに関連するすべての取り込みを示すために使用されます。

AWS AppFabric のセキュリティ

のクラウドセキュリティが最優先事項 AWS です。お客様は AWS 、セキュリティを最も重視する組織の要件を満たすように構築されたデータセンターとネットワークアーキテクチャからメリットを得られます。

セキュリティは、 AWS お客様とお客様の間の責任共有です。責任共有モデルでは、これをクラウドのセキュリティおよびクラウド内のセキュリティと説明しています。

- クラウドのセキュリティ – AWS は、 AWS のサービス で実行されるインフラストラクチャを保護する責任を担います AWS クラウド。 は、お客様が安全に使用できるサービス AWS も提供します。サードパーティーの監査者は、AWS コンプライアンスプログラムコンプライアンスプログラムの一環として、当社のセキュリティの有効性を定期的にテストおよび検証。 AWS AppFabric に適用されるコンプライアンスプログラムの詳細については、「コンプライアンスプログラムAWS による対象範囲内のサービスコンプライアンスプログラム」を参照してください。
- クラウド内のセキュリティ – お客様の責任は、使用する によって決まり AWS のサービス ます。また、ユーザーは、データの機密性、会社の要件、適用される法律や規制など、その他の要因についても責任を負います。

このドキュメントは、AppFabric を使用する際の責任共有モデルの適用方法を理解するのに役立ちます。以下のトピックでは、セキュリティおよびコンプライアンスの目的を達成するために AppFabric を設定する方法を示します。また、AppFabric リソースのモニタリングと保護 AWS のサービス に役立つ他の の使用方法についても説明します。

トピック

- AWS AppFabric でのデータ保護
- AWS AppFabric の Identity and Access Management
- AWS AppFabric のコンプライアンス検証
- AWS AppFabric のセキュリティのベストプラクティス
- AWS AppFabric の耐障害性
- AWS AppFabric のインフラストラクチャセキュリティ
- AWS AppFabric での設定と脆弱性の分析

AWS AppFabric でのデータ保護

責任 AWS 共有モデル、 AWS AppFabric でのデータ保護に適用されます。このモデルで説明されているように、 AWS はすべての を実行するグローバルインフラストラクチャを保護する責任があります AWS クラウド。ユーザーは、このインフラストラクチャでホストされるコンテンツに対する管理を維持する責任があります。また、使用する「 AWS のサービス 」のセキュリティ設定と管理タスクもユーザーの責任となります。データプライバシーの詳細については、データプライバシーに関するよくある質問を参照してください。欧州でのデータ保護の詳細については、AWS セキュリティブログに投稿された AWS 責任共有モデルおよび GDPR のブログ記事を参照してください。

データ保護の目的で、認証情報を保護し AWS アカウント 、 AWS IAM Identity Center または AWS Identity and Access Management (IAM) を使用して個々のユーザーを設定することをお勧めします。この方法により、それぞれのジョブを遂行するために必要な権限のみが各ユーザーに付与されます。また、次の方法でデータを保護することもお勧めします:

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。TLS 1.2 が必須で、TLS 1.3 をお勧めします。
- を使用して API とユーザーアクティビティのログ記録を設定します AWS CloudTrail。CloudTrail 証跡を使用して AWS アクティビティをキャプチャする方法については、「 AWS CloudTrail ユーザーガイド」のCloudTrail 証跡の使用」を参照してください。
- AWS 暗号化ソリューションと、その中のすべてのデフォルトのセキュリティコントロールを使用します AWS のサービス。
- Amazon Macie などの高度な管理されたセキュリティサービスを使用します。これらは、Amazon S3 に保存されている機密データの検出と保護を支援します。
- コマンドラインインターフェイスまたは API AWS を介して にアクセスするときに FIPS 140-3 検証済み暗号化モジュールが必要な場合は、FIPS エンドポイントを使用します。利用可能な FIPS エンドポイントの詳細については、「連邦情報処理規格 (FIPS) 140-3」を参照してください。

お客様の E メールアドレスなどの極秘または機密情報を、タグ、または [名前] フィールドなどの自由形式のテキストフィールドに含めないことを強くお勧めします。これは、コンソール、API、または SDK を使用して AppFabric AWS CLIまたは他の AWS のサービス を使用する場合も同様です。AWS SDKs タグ、または名前に使用される自由記述のテキストフィールドに入力したデータは、請求または診断ログに使用される場合があります。外部サーバーに URL を提供する場合、そのサーバーへのリクエストを検証できるように、認証情報を URL に含めないことを強くお勧めします。

> **Note**
>
> AppFabric for security に適用されるデータ保護の詳細については、「AppFabric でのデータ処理」を参照してください。

保管中の暗号化

AWS AppFabric は、保管時の暗号化をサポートしています。これは、AppFabric がアプリケーションバンドルに関連するすべてのデータをディスクに保持するときに透過的に暗号化し、データにアクセスするときに復号するサーバー側の暗号化機能です。デフォルトでは、AppFabric は AWS 所有のキー from AWS Key Management Service () を使用してデータを暗号化しますAWS KMS。独自のカスタマーマネージドキーを使用してデータを暗号化することもできます AWS KMS。

ユーザーを削除すると、そのユーザーのメタデータはすべて、完全に削除されます。

転送中の暗号化

アプリバンドルを設定するときは、 AWS 所有のキー またはカスタマーマネージドキーを選択できます。監査ログの取り込み用にデータを収集して正規化する際、AppFabric はデータを中間の Amazon Simple Storage Service (Amazon S3) バケットに一時的に保存し、このキーを使用して暗号化します。この中間バケットは、バケットライフサイクルポリシーを使用して 30 日後に削除されます。

AppFabric は、TLS 1.2 を使用して転送中のすべてのデータを保護し、 AWS 署名 V4 を使用して の AWS のサービス API リクエストに署名します。

キー管理

AppFabric は、 AWS 所有のキー またはカスタマーマネージドキーによるデータの暗号化をサポートしています。暗号化されたデータを完全に管理できるので、カスタマーマネージドキーを使用することをお勧めします。カスタマーマネージドキーを選択すると、AppFabricは、カスタマーマネージドキーへのアクセスを許可するリソースポリシーをカスタマーマネージドキーにアタッチします。

カスタマーマネージドキー

カスタマーマネージドキーを作成するには、「AWS KMS デベロッパーガイド」の「暗号化KMS キーを作成する」のステップに従います。

キーポリシー

キーポリシーは、カスタマーマネージドキーへのアクセスを制御します。すべてのカスタマーマネージドキーには、キーポリシーが 1 つだけ必要です。このポリシーには、そのキーを使用できるユーザーとその使用方法を決定するステートメントが含まれています。キーポリシーは、カスタマーマネージドキーの作成時に指定できます。キーポリシーの作成について詳細については、[AWS KMS デベロッパーガイド] の[キーの作成] のポリシー を参照してください。

AppFabric でカスタマーマネージドキーを使用するには、AppFabric リソースを作成する AWS Identity and Access Management (IAM) ユーザーまたはロールに、カスタマーマネージドキーを使用するアクセス許可が必要です。AppFabric でのみ使用するキーを作成し、AppFabric ユーザーをそのキーのユーザーとして追加することをお勧めします。この方法では、データへのアクセス範囲が制限されます。ユーザーが必要とする権限は次のとおりです。

- `kms:DescribeKey`
- `kms:CreateGrant`
- `kms:GenerateDataKey`
- `kms:Decrypt`

AWS KMS コンソールでは、適切なキーポリシーを使用してキーを作成する手順を説明します。キーポリシーの詳細については、「AWS KMS デベロッパーガイド」の「 AWS KMSのキーポリシー」を参照してください。

以下は、それを許可するキーポリシーの例です。

- キーの完全な AWS アカウントのルートユーザー 制御。
- カスタマーマネージドキーまたはカスタマーマネージドキーを使用することをお勧めします。
- アプリバンドルのキーポリシーは、でセットアップされます。`us-east-1`

```
{
    "Id": "key-consolepolicy-3",
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow access for key administrators",
            "Effect": "Allow",
            "Principal": {"AWS": "arn:aws:iam::111122223333:root"},
            "Action": ["kms:*"],
```

```
            "Resource": "arn:aws:kms:us-east-1:111122223333:key/key_ID"
        },
        {
            "Sid": "Allow read-only access to key metadata to the account",
            "Effect": "Allow",
            "Principal": {"AWS": "arn:aws:iam::111122223333:root"},
            "Action": [
                "kms:Describe*",
                "kms:Get*",
                "kms:List*",
                "kms:RevokeGrant"
            ],
            "Resource": "*"
        },
        {
            "Sid": "Allow access to principals authorized to use AWS AppFabric",
            "Effect": "Allow",
            "Principal": {"AWS": "IAM-role/user-creating-appfabric-resources"},
            "Action": [
                "kms:Decrypt",
                "kms:GenerateDataKey",
                "kms:DescribeKey",
                "kms:CreateGrant",
                "kms:ListAliases"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "appfabric.us-east-1.amazonaws.com",
                    "kms:CallerAccount": "111122223333"
                }
            }
        }
    ]
}
```

AppFabric が で許可を使用する方法 AWS KMS

AppFabricでカスタマーマネージドキーを使用するには許可が必要です。詳細については、「AWS KMS デベロッパーガイド」の「許可 AWS KMS」を参照してください。

App Bundle を作成すると、AppFabric は`CreateGrant`リクエストを送信してユーザーに代わって許可を作成します AWS KMS。の許可 AWS KMS は、AppFabric に顧客アカウントの AWS KMS キー

へのアクセスを許可するために使用されます。AppFabricは、このグラントが、以下の内部オペレーションでカスタマー管理キーを使用することを求めます。

- カスタマーマネージドキーによって暗号化されたデータキーを生成する AWS KMS リクエストGenerateDataKeyを に送信します。
- にDecryptリクエストを送信 AWS KMS して暗号化されたデータキーを復号し、データの暗号化と転送中のアプリケーションアクセストークンの復号化に使用できます。
- 転送中のアプリケーションアクセストークンを暗号化 AWS KMS するEncryptリクエストを に送信します。

グラントの例を以下に示します。

```
{
  "KeyId": "arn:aws:kms:us-east-1:111122223333:key/ff000af-00eb-00ce-0e00-
ea000fb0fba0SAMPLE",
  "GrantId": "0ab0ac0d0b000f00ea00cc0a0e00fc00bce000c000f0000000c0bc0a0000aaafSAMPLE",
  "Name": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
  "CreationDate": "2022-10-11T20:35:39+00:00",
  "GranteePrincipal": "appfabric.us-east-1.amazonaws.com",
  "RetiringPrincipal": "appfabric.us-east-1.amazonaws.com",
  "IssuingAccount": "arn:aws:iam::111122223333:root",
  "Operations": [
    "Decrypt",
    "Encrypt",
    "GenerateDataKey"
  ],
  "Constraints": {
    "EncryptionContextSubset": {
      "appBundleArn": "arn:aws:fabric:us-east-1:111122223333:appbundle/
ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE"
    }
  }
},
```

アプリバンドルを削除すると、AppFabric はカスタマー管理キーに対して発行されたグラントを廃止します。

AppFabricの暗号化キーのモニタリング

AppFabric で AWS KMS カスタマーマネージドキーを使用すると、 AWS CloudTrail ログを使用して AppFabric が送信するリクエストを追跡できます AWS KMS。

以下は、AppFabric `CreateGrant`がカスタマーマネージドキー に使用したときに記録される CloudTrail イベントの例です。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAIGDTESTANDEXAMPLE:SampleUser",
        "arn": "arn:aws:sts::111122223333:assumed-role/AssumedRole/SampleUser",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAIGDTESTANDEXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/AssumedRole",
                "accountId": "111122223333",
                "userName": "SampleUser"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-04-28T14:01:33Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-04-28T14:05:48Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateGrant",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "appfabric.amazonaws.com",
    "userAgent": "appfabric.amazonaws.com",
    "requestParameters": {
        "granteePrincipal": "appfabric.us-east-1.amazonaws.com",
        "constraints": {
            "encryptionContextSubset": {
                "appBundleArn": "arn:aws:appfabric:us-east-1:111122223333:appbundle/
ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE"
```

```
            }
        },
        "keyId": "arn:aws:kms:us-east-1:111122223333:key/EXAMPLEID",
        "retiringPrincipal": "appfabric.us-east-1.amazonaws.com",
        "operations": [
            "Encrypt",
            "Decrypt",
            "GenerateDataKey"
        ]
    },
    "responseElements": {
        "grantId": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
        "keyId": "arn:aws:kms:us-east-1:111122223333:key/KEY_ID"
    },
    "additionalEventData": {
        "grantId":
 "0ab0ac0d0b000f00ea00cc0a0e00fc00bce000c000f0000000c0bc0a0000aaafSAMPLE"
    },
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": false,
    "resources": [
        {
            "accountId": "AWS Internal",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-east-1:111122223333:key/key_ID"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "sharedEventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventCategory": "Management",
    "tlsDetails": {
        "tlsVersion": "TLSv1.3",
        "cipherSuite": "TLS_AES_256_GCM_SHA384",
        "clientProvidedHostHeader": "kms.us-east-1.amazonaws.com"
    }
}
```

AWS AppFabric 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制御 AWS のサービス するのに役立つ です。IAM 管理者は、AppFabric リソースを使用するための 認証 (サインイン)、認可 (アクセス許可を持つ) できるユーザーを制御します。IAM は、追加料金なしで使用できる AWS のサービス です。

トピック

- 対象者
- アイデンティティを使用した認証
- ポリシーを使用したアクセスの管理
- AWS AppFabric と IAM の連携方法
- AWS AppFabric のアイデンティティベースのポリシーの例
- AppFabric のサービスリンクロールの使用
- AWSAWS AppFabric の マネージドポリシー
- AWS AppFabric アイデンティティとアクセスのトラブルシューティング

対象者

AWS Identity and Access Management (IAM) の使用方法は、AppFabric で行う作業によって異なります。

サービスユーザー – ジョブを実行するために AppFabric サービスを使用する場合は、管理者から必要なアクセス許可と認証情報が与えられます。作業を実行するためにさらに多くの AppFabric の特徴を使用するとき、追加の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、管理者に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。AppFabric の機能にアクセスできない場合は、「AWS AppFabric アイデンティティとアクセスのトラブルシューティング」を参照してください。

[サービス管理者] - 社内の AppFabric リソースを担当している場合は、通常、AppFabric へのフルアクセスがあります。サービスのユーザーがどの AppFabric機能やリソースにアクセスするかを決めるのは管理者の仕事です。その後、IAM 管理者にリクエストを送信して、サービスユーザーの権限を変更する必要があります。このページの情報を点検して、IAM の基本概念を理解してください。貴社が AppFabric で IAM を利用する方法の詳細については、「AWS AppFabric と IAM の連携方法」を参照してください。

IAM 管理者 - IAM 管理者は、AppFabric 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 AppFabric 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 AppFabric のアイデンティティベースのポリシーの例」を参照してください。

アイデンティティを使用した認証

認証は、ID 認証情報 AWS を使用して にサインインする方法です。として、IAM ユーザーとして AWS アカウントのルートユーザー、または IAM ロールを引き受けることによって、認証 (にサインイン AWS) される必要があります。

ID ソースを介して提供された認証情報を使用して、フェデレーティッド ID AWS として にサインインできます。 AWS IAM Identity Center （ IAM Identity Center) ユーザー、会社のシングルサインオン認証、Google または Facebook 認証情報は、フェデレーション ID の例です。フェデレーティッド ID としてサインインする場合、IAM ロールを使用して、前もって管理者により ID フェデレーションが設定されています。フェデレーション AWS を使用して にアクセスすると、間接的にロールを引き受けることになります。

ユーザーのタイプに応じて、 AWS Management Console または AWS アクセスポータルにサインインできます。へのサインインの詳細については AWS、「 AWS サインイン ユーザーガイド」の「 へのサインイン AWS アカウント方法」を参照してください。

AWS プログラムで にアクセスする場合、 はソフトウェア開発キット (SDK) とコマンドラインインターフェイス (CLI) AWS を提供し、認証情報を使用してリクエストを暗号化して署名します。AWS ツールを使用しない場合は、自分でリクエストに署名する必要があります。リクエストに自分で署名する推奨方法の使用については、「IAM ユーザーガイド」の「API リクエストに対するAWS Signature Version 4」を参照してください。

使用する認証方法を問わず、追加セキュリティ情報の提供をリクエストされる場合もあります。たとえば、 では、アカウントのセキュリティを高めるために多要素認証 (MFA) を使用する AWS ことをお勧めします。詳細については、「AWS IAM Identity Center ユーザーガイド」の「多要素認証」および「IAM ユーザーガイド」の「IAM のAWS 多要素認証」を参照してください。

AWS アカウント ルートユーザー

を作成するときは AWS アカウント、アカウント内のすべての およびリソースへの AWS のサービス完全なアクセス権を持つ 1 つのサインインアイデンティティから始めます。この ID は AWS アカウント ルートユーザーと呼ばれ、アカウントの作成に使用した E メールアドレスとパスワードでサインインすることでアクセスできます。日常的なタスクには、ルートユーザーを使用しないことを強く

お勧めします。ルートユーザーの認証情報は保護し、ルートユーザーでしか実行できないタスクを実行するときに使用します。ルートユーザーとしてサインインする必要があるタスクの完全なリストについては、「IAM ユーザーガイド」の「ルートユーザー認証情報が必要なタスク」を参照してください。

フェデレーティッドアイデンティティ

ベストプラクティスとして、管理者アクセスを必要とするユーザーを含む人間のユーザーに、ID プロバイダーとのフェデレーションを使用して一時的な認証情報 AWS のサービス を使用して にアクセスすることを要求します。

フェデレーティッド ID は、エンタープライズユーザーディレクトリ、ウェブ ID プロバイダー、AWS Directory Service、アイデンティティセンターディレクトリ、または ID ソースを介して提供された認証情報 AWS のサービス を使用して にアクセスするユーザーです。フェデレーティッド ID がアクセスすると AWS アカウント、ロールを引き受け、ロールは一時的な認証情報を提供します。

アクセスを一元管理する場合は、 AWS IAM Identity Centerを使用することをお勧めします。IAM Identity Center でユーザーとグループを作成するか、独自の ID ソースのユーザーとグループのセットに接続して同期し、すべての AWS アカウント とアプリケーションで使用できます。IAM Identity Center の詳細については、「AWS IAM Identity Center ユーザーガイド」の「What is IAM Identity Center?」(IAM Identity Center とは) を参照してください。

IAM ユーザーとグループ

IAM ユーザーは、単一のユーザーまたはアプリケーションに対して特定のアクセス許可 AWS アカウント を持つ 内の ID です。可能であれば、パスワードやアクセスキーなどの長期的な認証情報を保有する IAM ユーザーを作成する代わりに、一時的な認証情報を使用することをお勧めします。ただし、IAM ユーザーでの長期的な認証情報が必要な特定のユースケースがある場合は、アクセスキーをローテーションすることをお勧めします。詳細については、「IAM ユーザーガイド」の「長期的な認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションする」を参照してください。

IAM グループは、IAM ユーザーの集団を指定するアイデンティティです。グループとしてサインインすることはできません。グループを使用して、複数のユーザーに対して一度に権限を指定できます。多数のユーザーグループがある場合、グループを使用することで権限の管理が容易になります。例えば、IAMAdmins という名前のグループを設定して、そのグループに IAM リソースを管理する許可を与えることができます。

ユーザーは、ロールとは異なります。ユーザーは 1 人の人または 1 つのアプリケーションに一意に関連付けられますが、ロールはそれを必要とする任意の人が引き受けるようになっています。ユー

ザーには永続的な長期の認証情報がありますが、ロールでは一時認証情報が提供されます。詳細については、「IAM ユーザーガイド」の「IAM ユーザーに関するユースケース」を参照してください。

IAM ロール

IAM ロールは、特定のアクセス許可 AWS アカウント を持つ 内の ID 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。で IAM ロールを一時的に引き受けるには AWS Management Console、ユーザーから IAM ロール (コンソール) に切り替えることができます。ロールを引き受けるには、 または AWS API オペレーションを AWS CLI 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「ロールを引き受けるための各種方法」を参照してください。

IAM ロールと一時的な認証情報は、次の状況で役立ちます:

- フェデレーションユーザーアクセス – フェデレーティッド ID に許可を割り当てるには、ロールを作成してそのロールの許可を定義します。フェデレーティッド ID が認証されると、その ID はロールに関連付けられ、ロールで定義されている許可が付与されます。フェデレーションのロールについては、「IAM ユーザーガイド」の「サードパーティー ID プロバイダー (フェデレーション) 用のロールを作成する」を参照してください。IAM Identity Center を使用する場合は、許可セットを設定します。アイデンティティが認証後にアクセスできるものを制御するため、IAM Identity Center は、権限セットを IAM のロールに関連付けます。アクセス許可セットの詳細については、「AWS IAM Identity Center User Guide」の「Permission sets」を参照してください。
- 一時的な IAM ユーザー権限 - IAM ユーザーまたはロールは、特定のタスクに対して複数の異なる権限を一時的に IAM ロールで引き受けることができます。
- クロスアカウントアクセス - IAM ロールを使用して、自分のアカウントのリソースにアクセスすることを、別のアカウントの人物 (信頼済みプリンシパル) に許可できます。クロスアカウントアクセス権を付与する主な方法は、ロールを使用することです。ただし、一部の では AWS のサービス、 (プロキシとしてロールを使用する代わりに) リソースに直接ポリシーをアタッチできます。クロスアカウントアクセスにおけるロールとリソースベースのポリシーの違いについては、「IAM ユーザーガイド」の「IAM でのクロスアカウントのリソースへのアクセス」を参照してください。
- クロスサービスアクセス — 一部の は他の の機能 AWS のサービス を使用します AWS のサービス。例えば、あるサービスで呼び出しを行うと、通常そのサービスによって Amazon EC2 でアプリケーションが実行されたり、Amazon S3 にオブジェクトが保存されたりします。サービスでは、呼び出し元プリンシパルの許可、サービスロール、またはサービスリンクロールを使用してこれを行う場合があります。

- 転送アクセスセッション (FAS) – IAM ユーザーまたはロールを使用してアクションを実行すると AWS、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストをリクエストする と組み合わせて使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。FAS リクエストを行う際のポリシーの詳細については、「転送アクセスセッション」を参照してください。
- サービスロール - サービスがユーザーに代わってアクションを実行するために引き受ける IAM ロールです。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除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「AWS のサービスに許可を委任するロールを作成する」を参照してください。
- サービスにリンクされたロール – サービスにリンクされたロールは、 にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント 、サービスによって所有されます。IAM 管理者は、サービスリンクロールの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 – IAM ロールを使用して、EC2 インスタンスで実行され、 AWS CLI または AWS API リクエストを行うアプリケーションの一時的な認証情報を管理できます。これは、EC2 インスタンス内でのアクセスキーの保存に推奨されます。EC2 インスタンスに AWS ロールを割り当て、そのすべてのアプリケーションで使用できるようにするには、インスタンスにアタッチされたインスタンスプロファイルを作成します。インスタンスプロファイルにはロールが含まれ、EC2 インスタンスで実行されるプログラムは一時的な認証情報を取得できます。詳細については、「IAM ユーザーガイド」の「Amazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用して許可を付与する」を参照してください。

ポリシーを使用したアクセスの管理

でアクセスを制御する AWS には、ポリシーを作成し、ID AWS またはリソースにアタッチします。ポリシーは AWS 、アイデンティティまたはリソースに関連付けられているときにアクセス許可を定義する のオブジェクトです。 は、プリンシパル (ユーザー、ルートユーザー、またはロールセッション) がリクエストを行うときに、これらのポリシー AWS を評価します。ポリシーでの権限により、リクエストが許可されるか拒否されるかが決まります。ほとんどのポリシーは JSON ドキュメ

ント AWS として に保存されます。JSON ポリシードキュメントの構造と内容の詳細については、「IAM ユーザーガイド」の「JSON ポリシー概要」を参照してください。

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

デフォルトでは、ユーザーやロールに権限はありません。IAM 管理者は、リソースで必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き受けることができます。

IAM ポリシーは、オペレーションの実行方法を問わず、アクションの許可を定義します。例えば、`iam:GetRole` アクションを許可するポリシーがあるとします。そのポリシーを持つユーザーは、 AWS Management Console、、 AWS CLIまたは AWS API からロール情報を取得できます。

アイデンティティベースのポリシー

アイデンティティベースポリシーは、IAM ユーザーグループ、ユーザーのグループ、ロールなど、アイデンティティにアタッチできる JSON 許可ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件をコントロールします。アイデンティティベースポリシーの作成方法については、「IAM ユーザーガイド」の「カスタマー管理ポリシーでカスタム IAM アクセス許可を定義する」を参照してください。

アイデンティティベースのポリシーは、さらにインラインポリシーまたはマネージドポリシーに分類できます。インラインポリシーは、単一のユーザー、グループ、またはロールに直接埋め込まれています。管理ポリシーは、 内の複数のユーザー、グループ、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。マネージドポリシーまたはインラインポリシーのいずれかを選択する方法については、「IAM ユーザーガイド」の「管理ポリシーとインラインポリシーのいずれかを選択する」を参照してください。

リソースベースのポリシー

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあげられます。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーでは、プリンシパルを指定する必要があります。プ

リンシパルには、アカウント、ユーザー、ロール、フェデレーティッドユーザー、または を含めることができます AWS のサービス。

リソースベースのポリシーは、そのサービス内にあるインラインポリシーです。リソースベースのポリシーでは、IAM の AWS マネージドポリシーを使用できません。

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、および Amazon VPC は AWS WAF、ACLs。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 は、一般的でない追加のポリシータイプをサポートします。これらのポリシータイプでは、より一般的なポリシータイプで付与された最大の権限を設定できます。

- アクセス許可の境界 - アクセス許可の境界は、アイデンティティベースポリシーによって IAM エンティティ (IAM ユーザーまたはロール) に付与できる権限の上限を設定する高度な機能です。エンティティにアクセス許可の境界を設定できます。結果として得られる権限は、エンティティのアイデンティティベースポリシーとそのアクセス許可の境界の共通部分になります。`Principal` フィールドでユーザーまたはロールを指定するリソースベースのポリシーでは、アクセス許可の境界は制限されません。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。アクセス許可の境界の詳細については、「IAM ユーザーガイド」の「IAM エンティティのアクセス許可の境界」を参照してください。
- サービスコントロールポリシー (SCPs) – SCPsは、 の組織または組織単位 (OU) の最大アクセス許可を指定する JSON ポリシーです AWS Organizations。 AWS Organizations は、ビジネスが所有する複数の をグループ化して一元管理するためのサービス AWS アカウント です。組織内のすべての機能を有効にすると、サービスコントロールポリシー (SCP) を一部またはすべてのアカウントに適用できます。SCP は、各 を含むメンバーアカウントのエンティティのアクセス許可を制限します AWS アカウントのルートユーザー。Organizations と SCP の詳細については、「AWS Organizations ユーザーガイド」の「サービスコントロールポリシー (SCP)」を参照してください。
- リソースコントロールポリシー (RCP) – RCP は、所有する各リソースにアタッチされた IAM ポリシーを更新することなく、アカウント内のリソースに利用可能な最大数のアクセス許可を設定する

ために使用できる JSON ポリシーです。RCP は、メンバーアカウントのリソースのアクセス許可を制限し、組織に属しているかどうかにかかわらず AWS アカウントのルートユーザー、 を含む ID の有効なアクセス許可に影響を与える可能性があります。RCP をサポートする のリストを含む Organizations と RCP の詳細については、AWS Organizations RCPs「リソースコントロールポリシー (RCPs」を参照してください。 AWS のサービス

- セッションポリシー - セッションポリシーは、ロールまたはフェデレーションユーザーの一時的なセッションをプログラムで作成する際にパラメータとして渡す高度なポリシーです。結果としてセッションの権限は、ユーザーまたはロールのアイデンティティベースポリシーとセッションポリシーの共通部分になります。また、リソースベースのポリシーから権限が派生する場合もあります。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。詳細については、「IAM ユーザーガイド」の「セッションポリシー」を参照してください。

複数のポリシータイプ

1 つのリクエストに複数のタイプのポリシーが適用されると、結果として作成される権限を理解するのがさらに難しくなります。が複数のポリシータイプが関与する場合にリクエストを許可するかどうか AWS を決定する方法については、「IAM ユーザーガイド」の「ポリシー評価ロジック」を参照してください。

AWS AppFabric と IAM の連携方法

IAM を使用して AppFabric へのアクセスを管理する前に、AppFabric で利用できる IAM の機能について学びます。

AWS AppFabric で使用できる IAM 機能

IAM 機能	AppFabricサポート
アイデンティティベースポリシー	はい
リソースベースのポリシー	いいえ
ポリシーアクション	はい
ポリシーリソース	あり
ポリシー条件キー	いいえ

IAM 機能	AppFabricサポート
ACL	いいえ
ABAC (ポリシー内のタグ)	あり
一時的な認証情報	いいえ
プリンシパル権限	はい
サービスロール	いいえ
サービスリンクロール	はい

AppFabric およびその他の がほとんどの IAM 機能と AWS のサービス 連携する方法の概要については、「IAM ユーザーガイド」のAWS 「IAM と連携する のサービス」を参照してください。

AppFabric の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のサポート: あり

アイデンティティベースポリシーは、IAM ユーザーグループ、ユーザーのグループ、ロールなど、アイデンティティにアタッチできる JSON 許可ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件をコントロールします。ID ベースのポリシーの作成方法については、「IAM ユーザーガイド」の「カスタマー管理ポリシーでカスタム IAM アクセス許可を定義する」を参照してください。

IAM アイデンティティベースのポリシーでは、許可または拒否するアクションとリソース、およびアクションを許可または拒否する条件を指定できます。プリンシパルは、それが添付されているユーザーまたはロールに適用されるため、アイデンティティベースのポリシーでは指定できません。JSON ポリシーで使用できるすべての要素について学ぶには、「IAM ユーザーガイド」の「IAM JSON ポリシーの要素のリファレンス」を参照してください。

AppFabric のアイデンティティベースのポリシーの例

AppFabric 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 AppFabric のアイデンティティベースのポリシーの例」を参照してください。

AppFabric 内のリソースベースのポリシー

リソースベースのポリシーのサポート: なし

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあげられます。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーでは、プリンシパルを指定する必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーティッドユーザー、または を含めることができます AWS のサービス。

クロスアカウントアクセスを有効にするには、アカウント全体、または別のアカウントの IAM エンティティをリソースベースのポリシーのプリンシパルとして指定します。リソースベースのポリシーにクロスアカウントのプリンシパルを追加しても、信頼関係は半分しか確立されない点に注意してください。プリンシパルとリソースが異なる場合 AWS アカウント、信頼されたアカウントの IAM 管理者は、プリンシパルエンティティ (ユーザーまたはロール) にリソースへのアクセス許可も付与する必要があります。IAM 管理者は、アイデンティティベースのポリシーをエンティティにアタッチすることで権限を付与します。ただし、リソースベースのポリシーで、同じアカウントのプリンシパルへのアクセス権が付与されている場合は、アイデンティティベースのポリシーをさらに付与する必要はありません。詳細については、「IAM ユーザーガイド」の「IAM でのクロスアカウントリソースアクセス」を参照してください。

AppFabric のポリシーアクション

ポリシーアクションのサポート:あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

JSON ポリシーの `Action` 要素にはポリシー内のアクセスを許可または拒否するために使用できるアクションが記述されます。ポリシーアクションの名前は通常、関連付けられた AWS API オペレーションと同じです。一致する API オペレーションのない許可のみのアクションなど、いくつかの例外があります。また、ポリシーに複数のアクションが必要なオペレーションもあります。これらの追加アクションは依存アクションと呼ばれます。

このアクションは関連付けられたオペレーションを実行するためのアクセス許可を付与するポリシーで使用されます。

AppFabric アクションのリストを確認するには、「サービス認可リファレンス」の[AWS AppFabric で定義されるアクション]」を参照してください。

AppFabric のポリシーアクションは、アクションの前に以下のプレフィックスを使用します。

```
appfabric
```

単一のステートメントで複数のアクションを指定するには、アクションをカンマで区切ります。

```
"Action": [
      "appfabric:action1",
      "appfabric:action2"
         ]
```

ワイルドカード文字 (*) を使用すると、複数のアクションを指定することができます。例えば、`List` 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": "appfabric:List*"
```

AppFabric 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 AppFabric のアイデンティティベースのポリシーの例]」を参照してください。

AppFabric のポリシーリソース

ポリシーリソースのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

`Resource` JSON ポリシー要素はアクションが適用されるオブジェクトを指定します。ステートメントには`Resource` または `NotResource` 要素を含める必要があります。ベストプラクティスとして、[Amazon リソースネーム (ARN)] を使用してリソースを指定します。これは、リソースレベルの許可と呼ばれる特定のリソースタイプをサポートするアクションに対して実行できます。

オペレーションのリスト化など、リソースレベルの権限をサポートしないアクションの場合は、ステートメントがすべてのリソースに適用されることを示すために、ワイルドカード (*) を使用します。

```
"Resource": "*"
```

AppFabric リソースタイプとその ARNsAWS AppFabric で定義されるリソースタイプ」を参照してください。各リソースの ARN を指定できるアクションについては、AWS AppFabric で定義されるアクション」を参照してください。

AppFabric 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 AppFabric のアイデンティティベースのポリシーの例」を参照してください。

AppFabric のポリシー条件キー

サービス固有のポリシー条件キーへのサポート: なし

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

`Condition` 要素 (または `Condition` ブロック) を使用すると、ステートメントが有効な条件を指定できます。`Condition` 要素はオプションです。イコールや未満などの 条件演算子 を使用して条件式を作成して、ポリシーの条件とリクエスト内の値を一致させることができます。

1 つのステートメントに複数の `Condition` 要素を指定する場合、または 1 つの `Condition` 要素に複数のキーを指定する場合、 AWS では `AND` 論理演算子を使用してそれらを評価します。1 つの条件キーに複数の値を指定すると、 は論理`OR`オペレーションを使用して条件 AWS を評価します。ステートメントの権限が付与される前にすべての条件が満たされる必要があります。

条件を指定する際にプレースホルダー変数も使用できます。例えば IAM ユーザーに、IAM ユーザー名がタグ付けされている場合のみリソースにアクセスできる権限を付与することができます。詳細については、「IAM ユーザーガイド」の「IAM ポリシーの要素: 変数およびタグ」を参照してください。

AWS は、グローバル条件キーとサービス固有の条件キーをサポートしています。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。

AppFabric 条件キーのリストを確認するには、「サービス認可リファレンス」のAWS AppFabric の条件キー」を参照してください。条件キーを使用できるアクションとリソースについては、AWS AppFabric で定義されるアクション」を参照してください。

AppFabric 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 AppFabric のアイデンティティベースのポリシーの例」を参照してください。

AppFabricの ACL

ACL のサポート: なし

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするための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。

AppFabricでの ABAC

ABAC (ポリシー内のタグ) のサポート: あり

属性ベースのアクセス制御 (ABAC) は、属性に基づいてアクセス許可を定義する認可戦略です。では AWS、これらの属性はタグと呼ばれます。タグは、IAM エンティティ (ユーザーまたはロール) および多くの AWS リソースにアタッチできます。エンティティとリソースのタグ付けは、ABAC の最初の手順です。その後、プリンシパルのタグがアクセスしようとしているリソースのタグと一致した場合にオペレーションを許可するように ABAC ポリシーをします。

ABAC は、急成長する環境やポリシー管理が煩雑になる状況で役立ちます。

タグに基づいてアクセスを管理するには、`aws:ResourceTag/key-name`、`aws:RequestTag/key-name`、または `aws:TagKeys` の条件キーを使用して、ポリシーの条件要素でタグ情報を提供します。

サービスがすべてのリソースタイプに対して 3 つの条件キーすべてをサポートする場合、そのサービスの値はありです。サービスが一部のリソースタイプに対してのみ 3 つの条件キーのすべてをサポートする場合、値は「部分的」になります。

ABAC の詳細については、「IAM ユーザーガイド」の「ABAC 認可でアクセス許可を定義する」を参照してください。ABAC をセットアップする手順を説明するチュートリアルについては、「IAM ユーザーガイド」の「属性ベースのアクセスコントロール (ABAC) を使用する」を参照してください。

AppFabric での一時的な認証情報の使用

一時的な認証情報のサポート: なし

一部の AWS のサービス は、一時的な認証情報を使用してサインインすると機能しません。一時的な認証情報 AWS のサービス を使用する場合などの詳細については、IAM ユーザーガイドの「IAM AWS のサービス と連携する 」を参照してください。

ユーザー名とパスワード以外の AWS Management Console 方法で にサインインする場合、一時的な認証情報を使用します。たとえば、会社のシングルサインオン (SSO) リンク AWS を使用して にアクセスすると、そのプロセスによって一時的な認証情報が自動的に作成されます。また、ユーザーとしてコンソールにサインインしてからロールを切り替える場合も、一時的な認証情報が自動的に作成されます。ロールの切り替えに関する詳細については、「IAM ユーザーガイド」の「ユーザーから IAM ロールに切り替える (コンソール)」を参照してください。

一時的な認証情報は、 AWS CLI または AWS API を使用して手動で作成できます。その後、これらの一時的な認証情報を使用して アクセスできます AWS。長期的なアクセスキーを使用する代わりに、一時的な認証情報を動的に生成 AWS することをお勧めします。詳細については、「IAM の一時的セキュリティ認証情報」を参照してください。

AppFabric のクロスサービスプリンシパル許可

転送アクセスセッション (FAS) のサポート: あり

IAM ユーザーまたはロールを使用して でアクションを実行すると AWS、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストをリクエストする と組み合わせて使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。FAS リクエストを行う際のポリシーの詳細については、「転送アクセスセッション」を参照してください。

AppFabric のサービスロール

サービスロールのサポート: なし

サービスロールとは、サービスがユーザーに代わってアクションを実行するために引き受ける IAM ロールです。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、「IAM ユーザーガイド」の「AWS のサービスに許可を委任するロールを作成する」を参照してください。

> **Warning**
> サービスロールの許可を変更すると、AppFabric の機能性が損なわれる可能性があります。AppFabric が指示する場合以外は、サービスロールを編集しないでください。

AppFabric のサービスにリンクされたロール

サービスリンクロールのサポート: あり

サービスにリンクされたロールは、 にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント 、サービスによって所有されます。IAM 管理者は、サービスリンクロールの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。

AppFabric サービスリンクロールの作成または管理の詳細については、「AppFabric のサービスリンクロールの使用」を参照してください。

AWS AppFabric のアイデンティティベースのポリシーの例

デフォルトでは、 ユーザーおよびロールには、 AppFabric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 Command Line Interface （ AWS CLI ） AWS Management Console、または AWS API を使用してタスクを実行することはできません。IAM 管理者は、リソースで必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き継ぐことができます。

これらサンプルの JSON ポリシードキュメントを使用して、IAM アイデンティティベースのポリシーを作成する方法については、「IAM ユーザーガイド」の「IAM ポリシーを作成する (コンソール)」を参照してください。

各リソースタイプの ARN の形式など、AppFabric で定義されるアクションとリソースタイプの詳細については、「サービス認可リファレンス」のAWS AppFabric 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。 ARNs

目次

- ポリシーに関するベストプラクティス
- AppFabricコンソールの使用
- AppFabric for security の IAM ポリシーの例
  - アプリケーションバンドルへのアクセスを許可する

- コンテンツに対するアクセス制限
- 取り込みの削除または停止を制限する
- AppFabric for productivity IAM ポリシーの例
  - productivity の機能への読み取り専用アクセスを許可する
  - productivity の機能への完全なアクセスを許可する
  - AppClients を作成するためのアクセスを許可する
  - AppClients の詳細を得るためのアクセスを許可する
  - AppClients を一覧表示するためのアクセスを許可する
  - AppClients を更新するためのアクセスを許可する
  - AppClients を削除するためのアクセスを許可する
  - アプリケーションを承認するためのアクセスを許可する
- その他 IAM ポリシーの例
  - 自分の権限の表示をユーザーに許可する

ポリシーに関するベストプラクティス

ID ベースのポリシーは、ユーザーのアカウントで誰かが AppFabricリソースを作成、アクセス、または削除できるどうかを決定します。これらのアクションを実行すると、 AWS アカウントに料金が発生する可能性があります。アイデンティティベースポリシーを作成したり編集したりする際には、以下のガイドラインと推奨事項に従ってください:

- AWS 管理ポリシーを開始し、最小特権のアクセス許可に移行 – ユーザーとワークロードにアクセス許可の付与を開始するには、多くの一般的なユースケースにアクセス許可を付与するAWS 管理ポリシーを使用します。これらは で使用できます AWS アカウント。ユースケースに固有の AWS カスタマー管理ポリシーを定義することで、アクセス許可をさらに減らすことをお勧めします。詳細については、「IAM ユーザーガイド」の「AWS マネージドポリシー」または「ジョブ機能の AWS マネージドポリシー」を参照してください。
- 最小特権を適用する – IAM ポリシーで許可を設定する場合は、タスクの実行に必要な許可のみを付与します。これを行うには、特定の条件下で特定のリソースに対して実行できるアクションを定義します。これは、最小特権アクセス許可とも呼ばれています。IAM を使用して許可を適用する方法の詳細については、「IAM ユーザーガイド」の「IAM でのポリシーとアクセス許可」を参照してください。
- IAM ポリシーで条件を使用してアクセスをさらに制限する - ポリシーに条件を追加して、アクションやリソースへのアクセスを制限できます。例えば、ポリシー条件を記述して、すべてのリク工

ストを SSL を使用して送信するように指定できます。条件を使用して、サービスアクションが などの特定の を通じて使用されている場合に AWS のサービス、サービスアクションへのアクセスを許可することもできます AWS CloudFormation。詳細については、「IAM ユーザーガイド」の「IAM JSON ポリシー要素:条件」を参照してください。

- IAM Access Analyzer を使用して IAM ポリシーを検証し、安全で機能的な権限を確保する - IAM Access Analyzer は、新規および既存のポリシーを検証して、ポリシーが IAM ポリシー言語 (JSON) および IAM のベストプラクティスに準拠するようにします。IAM アクセスアナライザーは 100 を超えるポリシーチェックと実用的な推奨事項を提供し、安全で機能的なポリシーの作成をサポートします。詳細については、「IAM ユーザーガイド」の「IAM Access Analyzer でポリシーを検証する」を参照してください。
- 多要素認証 (MFA) を要求する – で IAM ユーザーまたはルートユーザーを必要とするシナリオがある場合は AWS アカウント、MFA をオンにしてセキュリティを強化します。API オペレーションが呼び出されるときに MFA を必須にするには、ポリシーに MFA 条件を追加します。詳細については、「IAM ユーザーガイド」の「MFA を使用した安全な API アクセス」を参照してください。

IAM でのベストプラクティスの詳細については、「IAM ユーザーガイド」の「IAM でのセキュリティのベストプラクティス」を参照してください。

AppFabricコンソールの使用

`AWSAppFabricReadOnlyAccess` AWS マネージドポリシーを IAM ID にアタッチして、 の AppFabric コンソールを含む AppFabric サービスへの読み取り専用アクセス許可を付与します AWS Management Console。または、`AWSAppFabricFullAccess` AWS 管理ポリシーを IAM ID にアタッチして、AppFabric サービスへの完全な管理アクセス許可を付与することもできます。詳細については、「AWSAWS AppFabric の マネージドポリシー」を参照してください。

AppFabric for security の IAM ポリシーの例

以下のポリシー例は、AppFabric for security の機能に適用されます。

アプリケーションバンドルへのアクセスを許可する

次のポリシー例では、AppFabric サービスのアプリケーションバンドルへのアクセスを許可しています。

```
{
    "Statement": [
        {
            "Effect": "Allow",
```

```
            "Action": [
                "appfabric:StartUserAccessTasks",
                "appfabric:BatchGetUserAccessTasks"
            ],
            "Resource": ["arn:aws:appfabric:*:*:appbundle/*"]
        }
    ],
    "Version": "2012-10-17"
}
```

コンテンツに対するアクセス制限

次のポリシー例では、AppFabric サービスのアプリケーションバンドルへのアクセスを制限しています。

```
{
    "Statement": [
        {
            "Action": ["appfabric:*"],
            "Effect": "Allow",
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "appfabric:StartUserAccessTasks",
                "appfabric:BatchGetUserAccessTasks"
            ],
            "Resource": ["arn:aws:appfabric:*:*:appbundle/*"]
        }
    ],
    "Version": "2012-10-17"
}
```

取り込みの削除または停止を制限する

次のポリシー例では、AppFabric サービスの取り込みの削除または停止を制限しています。

```
{
    "Statement": [
        {
            "Action": ["appfabric:*"],
            "Effect": "Allow",
```

```
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "appfabric:StopIngestion",
                "appfabric:DeleteIngestion",
                "appfabric:DeleteIngestionDestination"
            ],
            "Resource": ["arn:aws:appfabric:*:*:appbundle/*"]
        }
    ],
    "Version": "2012-10-17"
}
```

AppFabric for productivity IAM ポリシーの例

AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

以下のポリシー例は、AppFabric for productivity の機能に適用されます。

productivity の機能への読み取り専用アクセスを許可する

次のポリシー例では、AppFabric for productivity の機能への、読み取り専用アクセスを許可しています。

> **Important**
>
> このポリシーを IAM コンソールの JSON ポリシーエディターに追加すると、無効なアクションのエラーが表示される場合があります。これは、AppFabric for productivity の機能が現時点ではプレビュー版であるためです。このエラーは無視し、ポリシーの作成を続けていただいて構いません。

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
```

```
                "appfabric:GetAppClient",
                "appfabric:ListActionableInsights",
                "appfabric:ListAppClients",
                "appfabric:ListMeetingInsights"
            ],
            "Resource": "*"
        }
    ],
    "Version": "2012-10-17"
}
```

productivity の機能への完全なアクセスを許可する

次のポリシー例では、AppFabric for productivity の機能への、完全なアクセスを許可しています。

> **Important**
>
> このポリシーを IAM コンソールの JSON ポリシーエディターに追加すると、無効なアクションのエラーが表示される場合があります。これは、AppFabric for productivity の機能が現時点ではプレビュー版であるためです。このエラーは無視し、ポリシーの作成を続けていただいて構いません。

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "appfabric:CreateAppClient",
                "appfabric:DeleteAppClient",
                "appfabric:GetAppClient",
                "appfabric:ListActionableInsights",
                "appfabric:ListAppClients",
                "appfabric:ListMeetingInsights",
                "appfabric:PutFeedback",
                "appfabric:Token"
                "appfabric:UpdateAppClient"
            ],
            "Resource": "*"
        }
    ],
    "Version": "2012-10-17"
```

```
}
```

AppClients を作成するためのアクセスを許可する

次のポリシー例では、AppClients を作成するためのアクセスを許可しています。詳細については、「AppFabric for productivity の AppClient を作成する」を参照してください。

> **Important**
>
> このポリシーを IAM コンソールの JSON ポリシーエディターに追加すると、無効なアクションのエラーが表示される場合があります。これは、AppFabric for productivity の機能が現時点ではプレビュー版であるためです。このエラーは無視し、ポリシーの作成を続けていただいて構いません。

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "appfabric:CreateAppClient"
            ],
            "Resource": ["arn:aws:appfabric:*:*:appclient/*"]
        }
    ],
    "Version": "2012-10-17"
}
```

AppClients の詳細を得るためのアクセスを許可する

次のポリシー例では、AppClients の詳細を得るためのアクセスを許可しています。詳細については、「AppClient の詳細を取得する」を参照してください。

> **Important**
>
> このポリシーを IAM コンソールの JSON ポリシーエディターに追加すると、無効なアクションのエラーが表示される場合があります。これは、AppFabric for productivity の機能が現時点ではプレビュー版であるためです。このエラーは無視し、ポリシーの作成を続けていただいて構いません。

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "appfabric:GetAppClient",
            ],
            "Resource": ["arn:aws:appfabric:*:*:appclient/*"]
        }
    ],
    "Version": "2012-10-17"
}
```

AppClients を一覧表示するためのアクセスを許可する

次のポリシー例では、AppClients を一覧表示するためのアクセスを許可しています。詳細については、「AppClient の詳細を取得する」を参照してください。

> **Important**
>
> このポリシーを IAM コンソールの JSON ポリシーエディターに追加すると、無効なアクションのエラーが表示される場合があります。これは、AppFabric for productivity の機能が現時点ではプレビュー版であるためです。このエラーは無視し、ポリシーの作成を続けていただいて構いません。

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "appfabric:ListAppClients"
            ],
            "Resource": ["arn:aws:appfabric:*:*:appclient/*"]
        }
    ],
    "Version": "2012-10-17"
}
```

AppClients を更新するためのアクセスを許可する

次のポリシー例では、AppClients を更新するためのアクセスを許可しています。詳細については、「AppClient を更新する」を参照してください。

> **Important**
>
> このポリシーを IAM コンソールの JSON ポリシーエディターに追加すると、無効なアクションのエラーが表示される場合があります。これは、AppFabric for productivity の機能が現時点ではプレビュー版であるためです。このエラーは無視し、ポリシーの作成を続けていただいて構いません。

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "appfabric:UpdateAppClient"
            ],
            "Resource": ["arn:aws:appfabric:*:*:appclient/*"]
        }
    ],
    "Version": "2012-10-17"
}
```

AppClients を削除するためのアクセスを許可する

次のポリシー例では、AppClients を削除するためのアクセスを許可しています。詳細については、「AppClient を更新する」を参照してください。

> **Important**
>
> このポリシーを IAM コンソールの JSON ポリシーエディターに追加すると、無効なアクションのエラーが表示される場合があります。これは、AppFabric for productivity の機能が現時点ではプレビュー版であるためです。このエラーは無視し、ポリシーの作成を続けていただいて構いません。

```
{
```

```
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "appfabric:DeleteAppClient"
            ],
            "Resource": ["arn:aws:appfabric:*:*:appclient/*"]
        }
    ],
    "Version": "2012-10-17"
}
```

アプリケーションを承認するためのアクセスを許可する

次のポリシー例では、Token API を使用してアプリケーションを承認するためのアクセスを許可しています。詳細については、「アプリケーションの認証と認可」を参照してください。

> **Important**
>
> このポリシーを IAM コンソールの JSON ポリシーエディターに追加すると、無効なアクションのエラーが表示される場合があります。これは、AppFabric for productivity の機能が現時点ではプレビュー版であるためです。このエラーは無視し、ポリシーの作成を続けていただいて構いません。

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "appfabric:Token"
            ],
            "Resource": ["arn:aws:appfabric:*:*:appclient/*"]
        }
    ],
    "Version": "2012-10-17"
}
```

その他 IAM ポリシーの例

自分の権限の表示をユーザーに許可する

この例では、ユーザーアイデンティティにアタッチされたインラインおよびマネージドポリシーの表示を IAM ユーザーに許可するポリシーの作成方法を示します。このポリシーには、コンソールで、または AWS CLI または AWS API を使用してプログラムでこのアクションを実行するアクセス許可が含まれています。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

AppFabric のサービスリンクロールの使用

AWS AppFabric は AWS Identity and Access Management (IAM) サービスにリンクされたロールを使用します。サービスリンクロールは、 AppFabric 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 AppFabricによって事前定義されており、サービスがユーザーに代わって他の AWS のサービス を呼び出すために必要なすべてのアクセス許可が含まれています。

サービスにリンクされたロールを使用することで、必要なアクセス権限を手動で追加する必要がなくなるため、 AppFabric の設定が簡単になります。AppFabric は、サービスリンクロールのアクセス許可を定義します。特に定義されている場合を除き、 AppFabric のみがそのロールを引き受けることができます。定義された許可には信頼ポリシーと許可ポリシーが含まれ、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、最初に関連リソースを削除する必要があります。これにより、リソースへのアクセス許可を不用意に削除することができないため、 AppFabric のリソースを保護することができます。

サービスにリンクされたロールをサポートする他のサービスの詳細については、AWS 「IAM と連携するサービス」を参照し、「サービスにリンクされたロール」列で「はい」があるサービスを探します。サービスリンクロールに関するドキュメントをサービスで表示するには、リンクで [はい] を選択します。

AppFabric のサービスリンクロールにおけるアクセス許可

AppFabric は、 という名前のサービスにリンクされたロールを使用します。`AWSServiceRoleForAppFabric`これにより、AppFabric は Amazon S3 バケットや Amazon Data Firehose 配信ストリームなどの取り込み先リソースにデータを配置できます。また、これにより AppFabric は CloudWatch メトリクスデータを`AWS/AppFabric`ネームスペースに配置できるようになります。

`AWSServiceRoleForAppFabric` サービスリンクロールは、以下のサービスを信頼してロールを引き受けます。

- `appfabric.amazonaws.com`

`AWSAppFabricServiceRolePolicy`という名前のロールのアクセス許可ポリシーは、指定したリソースに対して以下のアクションを実行することを AppFabric に許可します。

- アクション:`AWS/AppFabric`ネームスペース内で`cloudwatch:PutMetricData`。このアクションは、AppFabric に Amazon CloudWatch `AWS/AppFabric` ネームスペースにメトリックスデータを入力するアクセス権限を付与します。Amazon CloudWatch に保存されたAppFabricメトリクスの詳細については、「Amazon CloudWatch による AWS AppFabric のモニタリング」を参照してください。
- アクション: Amazon S3 バケットでの `s3:PutObject`。このアクションにより、AppFabricが許可され、指定した Amazon S3 バケットに統合データが入力されます。
- アクション: Amazon Data Firehose 配信ストリーム`firehose:PutRecordBatch`内。このアクションは、AppFabric が、ユーザーが指定した Amazon Data Firehose 配信ストリームに取り込まれたデータを配置するアクセス許可を付与します。

詳細については、「AWS AppFabric のマネージドポリシー」を参照してください。

ユーザー、グループ、ロールなどがサービスにリンクされたロールを作成、編集、削除できるようにするには、アクセス権限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アクセス許可」を参照してください。

AppFabric のサービスリンクロールの作成

サービスリンクロールを手動で作成する必要はありません。 AWS Management Console、、 AWS CLIまたは AWS API で AppFabric アプリバンドルを作成すると、AppFabric はサービスにリンクされたロールを作成します。

AppFabric のサービスリンクロールの編集

AppFabric では、 `AWSServiceRoleForAppFabric`のサービスにリンクされたロールを編集することはできません。サービスリンクロールを作成すると、多くのエンティティによってロールが参照される可能性があるため、ロール名を変更することはできません。ただし、IAM を使用したロール記述の編集はできます。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編集 」を参照してください。

AppFabric でのサービスリンクロールの削除

サービスリンクロールを必要とする機能やサービスが不要になった場合は、ロールを削除することをお勧めします。そうすることで、モニタリングや保守が積極的に行わ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、サービスにリンクされたロールを削除する前に、すべての AppFabric アプリケーションバンドルを削除する必要があります。

サービスリンク役割のクリーンアップ

IAM を使用してサービスにリンクされた役割を削除するには最初に、その役割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。AppFabric で作成したアプリバンドルはロールによって使用されます。詳細については、「セキュリティリソースの Delete AWS AppFabric 」を参照してください。

> **Note**
> リソースを削除しようとしたときに AppFabric サービスがロールを使用している場合、削除が失敗する可能性があります。失敗した場合は数分待ってから操作を再試行してください。

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、 AWS CLI、または AWS API を使用して、`AWSServiceRoleForAppFabric`サービスにリンクされたロールを削除します。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削除」を参照してください。

AppFabric サービスリンクロールをサポートするリージョン

AppFabric では、このサービスが利用可能なすべての AWS リージョン で、サービスリンクロールの使用をサポートしています。詳細については、「AWS 全般のリファレンス」の「 AppFabricエンドポイントとクォータ」を参照してください。

AWSAWS AppFabric の マネージドポリシー

ユーザー、グループ、ロールにアクセス許可を追加するには、自分でポリシーを記述するよりも AWS 管理ポリシーを使用する方が簡単です。チームに必要な権限のみを提供する IAM カスタマーマネージドポリシーを作成するには時間と専門知識が必要です。すぐに開始するには、 AWS マネージドポリシーを使用できます。これらのポリシーは、一般的なユースケースをターゲット範囲に含めており、 AWS アカウントで利用できます。 AWS 管理ポリシーの詳細については、IAM ユーザーガイドの「 AWS 管理ポリシー」を参照してください。

AWS のサービス AWS 管理ポリシーを維持および更新します。 AWS 管理ポリシーのアクセス許可は変更できません。サービスでは新しい機能を利用できるようにするために、 AWS マネージドポリシーに権限が追加されることがあります。この種類の更新はポリシーがアタッチされている、すべてのアイデンティティ (ユーザー、グループおよびロール) に影響を与えます。新しい機能が立ち上げられた場合や、新しいオペレーションが使用可能になった場合に、各サービスが AWS マネージドポ

リシーを更新する可能性が最も高くなります。サービスは AWS マネージドポリシーからアクセス許可を削除しないため、ポリシーの更新によって既存のアクセス許可が損なわれることはありません。

さらに、 は、複数のサービスにまたがるジョブ関数の マネージドポリシー AWS をサポートしています。例えば、ReadOnlyAccess AWS 管理ポリシーは、すべての AWS のサービス および リソースへの読み取り専用アクセスを提供します。サービスが新機能を起動すると、 は新しいオペレーションとリソースの読み取り専用アクセス許可 AWS を追加します。ジョブ機能のポリシーの一覧および詳細については、「IAM ユーザーガイド」の「AWS のジョブ機能のマネージドポリシー」を参照してください。

AWS 管理ポリシー: AWSAppFabricReadOnlyAccess

`AWSAppFabricReadOnlyAccess` ポリシーを IAM アイデンティティにアタッチできます。このポリシーでは、 AppFabricサービスに対する読み取り専用アクセスを付与します。

> **Note**
>
> `AWSAppFabricReadOnlyAccess` ポリシーは、AppFabric for productivity の機能への読み取り専用アクセスを許可しません。

アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- `appfabric`— アプリバンドルの取得、アプリバンドルの一覧表示、アプリ承認の取得、アプリ承認のリスト、取り込みの取得、取り込みの一覧表示、取り込み先の取得、取り込み先の取得、取り込み先のリスト、およびリソースタグの一覧表示を行う権限を付与します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "appfabric:GetAppAuthorization",
                "appfabric:GetAppBundle",
                "appfabric:GetIngestion",
                "appfabric:GetIngestionDestination",
                "appfabric:ListAppAuthorizations",
```

```
                "appfabric:ListAppBundles",
                "appfabric:ListIngestionDestinations",
                "appfabric:ListIngestions",
                "appfabric:ListTagsForResource"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS 管理ポリシー: AWSAppFabricFullAccess

`AWSAppFabricFullAccess` ポリシーを IAM アイデンティティにアタッチできます。このポリシーは、 AppFabricサービスへの管理権限を付与します。

> **Important**
>
> AppFabric for productivity の機能が現時点ではプレビュー版であるため、`AWSAppFabricFullAccess` ポリシーはこの機能へのアクセスを許可していません。AppFabric for productivity の機能へのアクセスを許可する方法の詳細については、「AppFabric for productivity IAM ポリシーの例」を参照してください。

アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- `appfabric`— AppFabric に完全な管理権限を付与します。
- `kms`— エイリアスを一覧表示するためのアクセス許可を付与します。
- `s3`すべての Amazon S3 バケットの権限を付与して、バケット位置を獲得します。
- `firehose` – Amazon Data Firehose 配信ストリームを一覧表示し、配信ストリームを記述するアクセス許可を付与します。
- `iam`— AppFabric `AWSServiceRoleForAppFabric` のサービスにリンクされたロールを作成する権限を付与します。詳細については、「AppFabric のサービスリンクロールの使用」を参照してください。

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

```
        {
            "Effect": "Allow",
            "Action": ["appfabric:*"],
            "Resource": "*"
        },
        {
            "Sid": "KMSListAccess",
            "Effect": "Allow",
            "Action": ["kms:ListAliases"],
            "Resource": "*"
        },
        {
            "Sid": "S3ReadAccess",
            "Effect": "Allow",
            "Action": [
                "s3:GetBucketLocation",
                "s3:ListAllMyBuckets"
            ],
            "Resource": "*"
        },
        {
            "Sid": "FirehoseReadAccess",
            "Effect": "Allow",
            "Action": [
                "firehose:DescribeDeliveryStream",
                "firehose:ListDeliveryStreams"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowUseOfServiceLinkedRole",
            "Effect": "Allow",
            "Action": ["iam:CreateServiceLinkedRole"],
            "Condition": {
                "StringEquals": {"iam:AWSServiceName": "appfabric.amazonaws.com"}
            },
            "Resource": "arn:aws:iam::*:role/aws-service-role/appfabric.amazonaws.com/
AWSServiceRoleForAppFabric"
        }
    ]
}
```

AWS マネージドポリシー: AWSAppFabricServiceRolePolicy

IAM エンティティに `AWSAppFabricServiceRolePolicy` ポリシーをアタッチすることはできません。このポリシーは、ユーザーに代わって AppFabricがアクションを実行することを許可する、サービスにリンクされたロールにアタッチされます。詳細については、「AppFabric のサービスリンクロールの使用」を参照してください。

アクセス許可の詳細

このポリシーには、以下のアクセス許可が含まれています。

- `cloudwatch`— AppFabric に Amazon CloudWatch `AWS/AppFabric` 名前空間にメトリックスデータを入力するアクセス権限を付与します。Amazon CloudWatch に保存されたAppFabricメトリクスの詳細については、「Amazon CloudWatch による AWS AppFabric のモニタリング」を参照してください。
- `s3`-AppFabricに、指定した Amazon S3 バケットに統合データの入力権限を付与します。
- `firehose` – AppFabric が、指定した Amazon Data Firehose 配信ストリームに取り込まれたデータを配置するためのアクセス許可を付与します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CloudWatchEmitMetric",
            "Effect": "Allow",
            "Action": ["cloudwatch:PutMetricData"],
            "Resource": "*",
            "Condition": {
                "StringEquals": {"cloudwatch:namespace": "AWS/AppFabric"}
            }
        },
        {
            "Sid": "S3PutObject",
            "Effect": "Allow",
            "Action": ["s3:PutObject"],
            "Resource": "arn:aws:s3:::*/AWSAppFabric/*",
            "Condition": {
                "StringEquals": {"s3:ResourceAccount": "${aws:PrincipalAccount}"}
            }
        },
```

```
        {
            "Sid": "FirehosePutRecord",
            "Effect": "Allow",
            "Action": ["firehose:PutRecordBatch"],
            "Resource": "arn:aws:firehose:*:*:deliverystream/*",
            "Condition": {
                "StringEqualsIgnoreCase": {"aws:ResourceTag/AWSAppFabricManaged":
 "true"}
            }
        }
    ]
}
```

AWS 管理ポリシーに対する AppFabric の更新

このサービスがこれらの変更の追跡を開始してからの AppFabric の AWS マネージドポリシーの更新に関する詳細を表示します。このページの変更に関する自動通知については、 [AppFabric Document history] ページの RSS フィードをご覧ください。

変更	説明	日付
AWSAppFabricReadOnlyAccess - 新しいポリシー	AppFabricは、AppFabricサービスへの読み取り専用権限付与の新しいポリシーを追加しました。	2023 年 6 月 27 日
AWSAppFabricFullAccess - 新しいポリシー	AppFabric は、AppFabric サービスに管理権限を付与する新しいポリシーを追加しました。	2023 年 6 月 27 日
AWSAppFabricServiceRolePolicy - 新しいポリシー	AppFabric は、`AWSServiceRoleForAppFabric` サービスリンクしたロールに対する新ポリシーを追加しました。	2023 年 6 月 27 日

変更	説明	日付
AppFabricが変更の追跡を開始しました	AppFabric は、 AWS 管理ポリシーの変更の追跡を開始しました。	2023 年 6 月 27 日

AWS AppFabric アイデンティティとアクセスのトラブルシューティング

次の情報は、AppFabric と IAM の使用に伴って発生する可能性がある一般的な問題の診断や修復に役立ちます。

トピック

- AppFabric でアクションを実行する権限がない
- iam:PassRole を実行する権限がない
- 自分の 以外のユーザーに AppFabric リソース AWS アカウント へのアクセスを許可したい

AppFabric でアクションを実行する権限がない

アクションを実行する権限がないというエラーが表示された場合は、そのアクションを実行できるようにポリシーを更新する必要があります。

次のエラー例は、mateojackson IAM ユーザーがコンソールを使用して、ある *my-example-widget* リソースに関する詳細情報を表示しようとしたことを想定して、その際に必要なappfabric:*GetWidget* アクセス許可を持っていない場合に発生するものです。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 appfabric:GetWidget on resource: my-example-widget
```

この場合、appfabric:*GetWidget* アクションを使用して *my-example-widget*リソースへのアクセスを許可するように、mateojackson ユーザーのポリシーを更新する必要があります。

サポートが必要な場合は、 AWS 管理者にお問い合わせください。サインイン資格情報を提供した担当者が管理者です。

iam:PassRole を実行する権限がない

iam:PassRole アクションを実行する権限がないというエラーが表示された場合は、ポリシーを更新して AppFabricにロールを渡すことができるようにする必要があります。

一部の AWS のサービス では、新しいサービスロールまたはサービスにリンクされたロールを作成する代わりに、そのサービスに既存のロールを渡すことができます。そのためには、サービスにロールを渡す権限が必要です。

以下の例のエラーは、marymajor という IAM ユーザーがコンソールを使用して AppFabric でアクションを実行しようとする場合に発生します。ただし、このアクションをサービスが実行するには、サービスロールから付与された権限が必要です。メアリーには、ロールをサービスに渡す許可がありません。

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

この場合、Mary のポリシーを更新してメアリーに iam:PassRole アクションの実行を許可する必要があります。

サポートが必要な場合は、 AWS 管理者にお問い合わせください。サインイン資格情報を提供した担当者が管理者です。

自分の 以外のユーザーに AppFabric リソース AWS アカウント へのアクセスを許可したい

他のアカウントのユーザーや組織外の人が、リソースにアクセスするために使用できるロールを作成できます。ロールの引き受けを委託するユーザーを指定できます。リソースベースのポリシーまたはアクセスコントロールリスト (ACL) をサポートするサービスの場合、それらのポリシーを使用して、リソースへのアクセスを付与できます。

詳細については、以下を参照してください:

- AppFabric でこれらの機能がサポートされているかどうかを確認するには、「AWS AppFabric と IAM の連携方法」を参照してください。
- 所有 AWS アカウント している のリソースへのアクセスを提供する方法については、IAM ユーザーガイドの「所有 AWS アカウント している別の の IAM ユーザーへのアクセスを提供する」を参照してください。
- リソースへのアクセスをサードパーティーに提供する方法については AWS アカウント、IAM ユーザーガイドの「サードパーティー AWS アカウント が所有する へのアクセスを提供する」を参照してください。
- ID フェデレーションを介してアクセスを提供する方法については、「IAM ユーザーガイド」の「外部で認証されたユーザー (ID フェデレーション) へのアクセスの許可」を参照してください。

- クロスアカウントアクセスにおけるロールとリソースベースのポリシーの使用方法の違いについては、「IAM ユーザーガイド」の「IAM でのクロスアカウントのリソースへのアクセス」を参照してください。

AWS AppFabric のコンプライアンス検証

AWS のサービス が特定のコンプライアンスプログラムの範囲内にあるかどうかを確認するには、AWS のサービス 「コンプライアンスプログラムによる対象範囲内」を参照して、関心のあるコンプライアンスプログラムを選択します。一般的な情報については、AWS 「コンプライアンスプログラム」を参照してください。

を使用して、サードパーティーの監査レポートをダウンロードできます AWS Artifact。詳細については、「Downloading Reports in AWS Artifact」を参照してください。

を使用する際のお客様のコンプライアンス責任 AWS のサービス は、お客様のデータの機密性、貴社のコンプライアンス目的、適用される法律および規制によって決まります。 は、コンプライアンスに役立つ以下のリソース AWS を提供します。

- セキュリティのコンプライアンスとガバナンス – これらのソリューション実装ガイドでは、アーキテクチャ上の考慮事項について説明し、セキュリティとコンプライアンスの機能をデプロイする手順を示します。
- HIPAA 対応サービスのリファレンス – HIPAA 対応サービスの一覧が提供されています。すべての AWS のサービス が HIPAA の対象となるわけではありません。
- AWS コンプライアンスリソース – このワークブックとガイドのコレクションは、お客様の業界や地域に適用される場合があります。
- AWS カスタマーコンプライアンスガイド – コンプライアンスの観点から責任共有モデルを理解します。このガイドは、複数のフレームワーク (米国国立標準技術研究所 (NIST)、Payment Card Industry Security Standards Council (PCI)、国際標準化機構 (ISO) を含む) のセキュリティコントロールを保護し、そのガイダンスに AWS のサービス マッピングするためのベストプラクティスをまとめたものです。
- 「 デベロッパーガイド」の「ルールによるリソースの評価」 – この AWS Config サービスは、リソース設定が内部プラクティス、業界ガイドライン、および規制にどの程度準拠しているかを評価します。 AWS Config
- AWS Security Hub – これにより AWS のサービス 、セキュリティ状態を包括的に把握できます AWS。Security Hub では、セキュリティコントロールを使用して AWS リソースを評価し、セキュリティ業界標準とベストプラクティスに対するコンプライアンスをチェックします。サポー

トされているサービスとコントロールの一覧については、Security Hub のコントロールリファレンスを参照してください。

- Amazon GuardDuty – 不審なアクティビティや悪意のあるアクティビティがないか環境をモニタリングすることで AWS アカウント、、ワークロード、コンテナ、データに対する潜在的な脅威 AWS のサービス を検出します。GuardDuty を使用すると、特定のコンプライアンスフレームワークで義務付けられている侵入検知要件を満たすことで、PCI DSS などのさまざまなコンプライアンス要件に対応できます。
- AWS Audit Manager – これにより AWS のサービス 、 AWS 使用状況を継続的に監査し、リスクの管理方法と規制や業界標準への準拠を簡素化できます。

AWS AppFabric のセキュリティのベストプラクティス

AWS AppFabric には、独自のセキュリティポリシーを開発および実装する際に考慮すべきいくつかのセキュリティ機能が用意されています。以下のベストプラクティスは一般的なガイドラインであり、完全なセキュリティソリューションを提供するものではありません。これらのベストプラクティスはお客様の環境に必ずしも適切または十分でない可能性があるため、処方箋ではなく、あくまで有用な検討事項とお考えください。

管理者アクセスなしでアプリケーションを監視する

読み取り専用 AWS Identity and Access Management (IAM) アクセス許可を使用すると、誰でも AppFabric を Amazon QuickSight や、 などの他のセキュリティ情報およびイベント管理 (SIEM) ツールと統合できますSplunk。アプリケーションセキュリティをモニタリングするために、データは Amazon Simple Storage Service (Amazon S3) バケットまたは Amazon Data Firehose 配信ストリームに配信されます。

AppFabric イベントを監視する

Amazon CloudWatch metrics を使用して、AppFabricをモニタリングできます。CloudWatch は AppFabric から毎分データを収集し、メトリクスに処理します。メトリクスが指定したしきい値に一致したときに通知をオフにするアラームを設定できます。詳細については、「Amazon CloudWatch による AWS AppFabric のモニタリング」を参照してください。

AWS AppFabric の耐障害性

AWS グローバルインフラストラクチャは、 AWS リージョン およびアベイラビリティーゾーンを中心に構築されています。複数の物理的に分離および分離されたアベイラビリティーゾーン AWS リー

ジョン は、低レイテンシー、高スループット、高度に冗長なネットワークで接続されています。アベイラビリティーゾーンでは、ゾーン間で中断することなく自動的にフェイルオーバーするアプリケーションとデータベースを設計および運用することができます。アベイラビリティーゾーンは、従来の単一または複数のデータセンターインフラストラクチャよりも可用性が高く、フォールトトレラントで、スケーラブルです。

AWS リージョン およびアベイラビリティーゾーンの詳細については、AWS 「 グローバルインフラストラクチャ」を参照してください。

AWS AppFabric のインフラストラクチャセキュリティ

マネージドサービスである AWS AppFabric は、ホワイトペーパー「Amazon Web Services: セキュリティプロセスの概要」に記載されている AWS グローバルネットワークセキュリティ手順で保護されています。

AWS 公開された API コールを使用して、ネットワーク経由で AppFabric にアクセスします。クライアントは、TLS 1.0 以降をサポートしている必要があります。TLS 1.2 以降が推奨されます。また、DHE (Ephemeral Diffie-Hellman) や ECDHE (Elliptic Curve Ephemeral Diffie-Hellman) などの Perfect Forward Secrecy (PFS) を使用した暗号スイートもクライアントでサポートされている必要があります。これらのモードはJava 7 以降など、ほとんどの最新システムでサポートされています。

また、リクエストにはアクセスキー ID と、IAM プリンシパルに関連付けられているシークレットアクセスキーを使用して署名する必要があります。または、リクエストに署名するための一時的なセキュリティ認証情報を生成するには、AWS Security Token Service (AWS STS) を使用することもできます。

AWS AppFabric での設定と脆弱性の分析

設定と IT コントロールは、 AWS とお客様の間の責任共有です。詳細については、 AWS 「 責任共有モデル」を参照してください。

モニタリング AWS AppFabric

モニタリングは、 AWS AppFabric およびその他の AWS ソリューションの信頼性、可用性、パフォーマンスを維持する上で重要な部分です。 は、AppFabric をモニタリングし、問題が発生したときに報告し、必要に応じて自動アクションを実行するために、以下のモニタリングツール AWS を提供しています。

- Amazon CloudWatch は、 AWS リソースと AWS で実行されるアプリケーションをリアルタイムでモニタリングします。メトリクスの収集と追跡、カスタマイズしたダッシュボードの作成、および指定したメトリクスが指定したしきい値に達したときに通知またはアクションを実行するアラームの設定を行うことができます。例えば、CloudWatch で Amazon EC2 インスタンスの CPU 使用率などのメトリクスを追跡し、必要に応じて新しいインスタンスを自動的に起動できます。詳細については、「Amazon CloudWatch ユーザーガイド」を参照してください。
- Amazon CloudWatch Logs を使用すると、Amazon EC2 インスタンスやその他のソースからログファイルをモニタリング AWS CloudTrail、保存、アクセスできます。CloudWatch Logs は、ログファイル内の情報をモニタリングし、特定のしきい値が満たされたときに通知します。高い耐久性を備えたストレージにログデータをアーカイブすることも可能です。詳細については、「Amazon CloudWatch Logs ユーザーガイド」を参照してください。
- AWS CloudTrail は、 によって、または に代わって行われた API コールおよび関連イベントをキャプチャ AWS アカウント し、指定した Amazon S3 バケットにログファイルを配信します。が呼び出したユーザーとアカウント AWS、呼び出し元のソース IP アドレス、および呼び出しの発生日時を特定できます。詳細については、AWS CloudTrail ユーザーガイドをご参照ください。

Amazon CloudWatch による AWS AppFabric のモニタリング

CloudWatch を使用して AWS AppFabric をモニタリングすることで、raw データを収集し、読み取り可能なほぼリアルタイムのメトリクスに加工できます。これらの統計は 15 か月間保持されるため、履歴情報にアクセスし、ウェブアプリケーションまたはサービスの動作をより的確に把握できます。また、特定のしきい値を監視するアラームを設定し、これらのしきい値に達したときに通知を送信したりアクションを実行したりできます。詳細については、「Amazon CloudWatch ユーザーガイド」を参照してください。

AppFabricサービスは、`AWS/AppFabric` 名前空間の以下のメトリクスをレポートします。

メトリクス	説明
AppFabric アプリ認可ステータス	アプリ認可のステータス (1接続の場合は 、その他の0場合は)。
AppFabric データ配信のレイテンシー	AppFabric が SaaS アプリケーションから監査ログを収集し、設定された送信先 (Amazon S3 または Amazon Data Firehose) に配信するのにかかった時間 (秒単位) 。
取り込み先のステータス	取り込み先のステータス (アクティブな場合は 1、その他の場合は 0)。
全体的なデータ遅延	SaaS アプリケーションでイベントが発生したときと、対応する監査ログが AppFabric によって設定された送信先 (Amazon S3 または Amazon Data Firehose) に配信されたときの時間差 (秒単位) 。
取り込まれるデータ量	Amazon Simple Storage Service (Amazon S3) または Amazon Data Firehose に配信されるデータのサイズ。

AppFabric メトリクスでは、次のディメンションがサポートされています。

ディメンション	説明
取り込み先Arn	取り込み先の Amazon リソースネーム (ARN)。

を使用した Logging AWS AppFabric API コール AWS CloudTrail

AWS AppFabric は AWS CloudTrail、AppFabric のユーザー、ロール、または によって実行されたアクションを記録するサービスAppFabricと統合 AWS のサービス されています。CloudTrail は、AppFabric に対するすべての APIコールをイベントとしてキャプチャします。キャプチャされたコールには、AppFabric コンソールからの呼び出しと、AppFabric API オペレーションへのコード

呼び出しが含まれます。証跡を作成する場合は、AppFabric のイベントなど、Amazon S3 バケットへの CloudTrail イベントの継続的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 [イベント履歴] で最新のイベントを表示できます。CloudTrailによって収集された情報を使用して、AppFabricに対して行われた要求、要求が行われたIPアドレス、要求を行った人、行われた時期、および追加の詳細を判別できます。

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

CloudTrail での AppFabric 情報

CloudTrail は、アカウントの作成 AWS アカウント 時に で有効になります。AppFabric でアクティビティが発生すると、そのアクティビティはイベント履歴の他の AWS のサービス イベントとともに CloudTrail イベントに記録されます。 最近のイベントは、 AWS アカウントで表示、検索、ダウンロードできます。詳細については、「AWS CloudTrail ユーザーガイド 」の「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。

AppFabric のイベントなど AWS アカウント、 のイベントの継続的な記録については、証跡を作成します。証跡により、CloudTrail はログファイルを Amazon S3 バケットに配信できます。デフォルトでは、コンソールで証跡を作成するときに、証跡がすべての AWS リージョンに適用されます。証跡は、 AWS パーティション内のすべてのリージョンからのイベントをログに記録し、指定した Amazon S3 バケットにログファイルを配信します。さらに、CloudTrail ログで収集されたイベントデータをより詳細に分析し、それを基にアクションを取るために他の AWS のサービス を設定できます。詳細については、「AWS CloudTrail ユーザーガイド:」の以下のトピックを参照してください。

- 証跡作成の概要
- 「CloudTrail がサポートされているサービスと統合」
- 「CloudTrail の Amazon SNS 通知の設定」
- CloudTrail ログファイルを複数のリージョンから受け取ると複数のアカウントから CloudTrail ログファイルを受け取る

すべての AppFabricアクションは CloudTrail によってログに記録され、AWS API リファレンスに記録されます。例えば、`CreateAppBundle`、`UpdateAppBundle`、`GetAppBundle` 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。アイデンティティ情報は、以下を判別するのに役立ちます。

- リクエストがルートまたは AWS Identity and Access Management (IAM) ユーザー認証情報を使用して行われたかどうか。
- リクエストがロールまたはフェデレーションユーザーのテンポラリなセキュリティ認証情報を使用して行われたかどうか。
- リクエストが、別の AWS のサービスによって送信されたかどうか。

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

AppFabric のログ ファイルエントリーの理解

証跡では、指定した Simple Storage Service (Amazon S3) バケットにイベントをログファイルとして配信するように設定できます。CloudTrail のログファイルは、単一か複数のログエントリを含みます。イベントは任意ソースからの単一リクエストを表し、リクエストされたアクション、アクションの日時、リクエストパラメータなどの情報を含みます。CloudTrail ログファイルは、パブリック API コールの順序付けられたスタックトレースではないため、特定の順序では表示されません。

以下の例は、`CreateAppBundle` アクションを示す CloudTrail ログエントリです。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAIGDTESTANDEXAMPLE:SampleUser",
        "arn": "arn:aws:sts::111122223333:assumed-role/AssumedRole/SampleUser",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAXUFER33B4FVC2GCYR",
                "arn": "arn:aws:iam::111122223333:role/AssumedRole",
                "accountId": "111122223333",
                "userName": "SampleUser"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-05-31T21:11:15Z",
                "mfaAuthenticated": "false"
            }
```

```
        }
    },
    "eventTime": "2023-05-31T21:22:16Z",
    "eventSource": "appfabric.amazonaws.com",
    "eventName": "CreateAppBundle",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "3.90.81.91",
    "userAgent": "Coral/Apache-HttpClient5",
    "requestParameters": {
        "clientToken": "64d9069f-e565-49a4-9374-6dc8631142e2"
    },
    "responseElements": {
        "appBundle": {
            "arn": "arn:aws:appfabric:us-
east-1:111122223333:appbundle/6aa92da0-5eeb-4ff4-aabf-4db7fd022ad1",
            "idpClientConfiguration": {
                "samlAudience": "urn:amazon:cognito:sp:us-east-1_GEdGiavzr",
                "samlRedirect": "https://6aa92da0-5eeb-4ff4-aabf-4db7fd022ad1.auth.us-
east-1.amazoncognito.com/saml2/idpresponse",
                "oidcRedirect": "https://6aa92da0-5eeb-4ff4-aabf-4db7fd022ad1.auth.us-
east-1.amazoncognito.com/oauth2/idpresponse"
            }
        }
    },
    "requestID": "17e15a5d-8c66-46c7-ad5b-f521004fa9c2",
    "eventID": "ba1dd847-86f6-4386-85be-0398e844a358",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management",
    "tlsDetails": {
        "clientProvidedHostHeader": "frontend.fabric.us-east-1.amazonaws.com"
    }
}
```

AppFabric のクォータ

AWS アカウント には、 ごとに、以前は制限と呼ばれていたデフォルトのクォータがあります AWS のサービス。特に明記されていない限り、クォータは地域固有です。一部のクォータについては引き上げをリクエストできますが、その他のクォータについては引き上げることはできません。

AppFabric のクォータを表示するには、[Service Quotas コンソール] を開きます。ナビゲーションペインで、[AWS サービス] を選択し、[AppFabric] を選択します。

クォータの引き上げをリクエストするには、「Service Quotas ユーザーガイド」の「Requesting a quota increase」(クォータ引き上げリクエスト) を参照してください。Service Quotas でクォータがまだ利用できない場合は、[上限引き上げ] フォームを使用してください。

の AppFabric に関連するクォータを次の表 AWS アカウント に示します。

名前	デフォルト	引き上げ可能	説明
アプリケーションバンドル	サポートされている各リージョン: 1	[いいえ]	現在の AWS リージョンのアカウントで作成できるアプリケーションバンドルの最大数。
アプリケーション権限	サポートされている各リージョン: 50	いいえ	現在の AWS リージョンのアカウントで作成できるアプリケーション認可の最大数。
取り込み	サポートされている各リージョン: 50	いいえ	現在の AWS リージョンのアカウントで作成できる取り込みの最大数。

名前	デフォルト	引き上げ可能	説明
取り込み先	サポートされている各リージョン：5	いいえ	アカウントで現在の AWS リージョンに作成できる取り込み先の取り込みあたりの最大数。
AppClient	サポートされている各リージョン: 1	[いいえ]	現在の AWS リージョンのアカウントで作成できる AppClientsの最大数。 AWS AppFabric for productivity 機能はプレビュー版であり、変更される可能性があります。

AppFabric 管理ガイドのドキュメント履歴

次の表に、 AWS AppFabric のドキュメントリリースを示します。

変更	説明	日付
サポートされている新しいアプリケーション	サポートされているアプリケーションJumpCloudとしてを追加しました。詳細については、AWS AppFabric でサポートされているアプリケーション」を参照してください。	2024 年 6 月 5 日
サポートされている新しいアプリケーションとセキュリティツール	サポートされているアプリケーションGoogle Analyticsとして Azure Monitorと を追加しました。詳細については、AWS AppFabric でサポートされているアプリケーション」を参照してください。サポートされるセキュリティツールSingularity Cloudとしてを追加しました。詳細については、「互換性のあるセキュリティツール」を参照してください。	2024 年 4 月 30 日
サポートされている新しいアプリケーション	サポートされているアプリケーションSentinelOneとして を追加しました。詳細については、AWS AppFabric でサポートされているアプリケーション」を参照してください。	2024 年 4 月 25 日

サポートされている新しいアプリケーション	サポートされているアプリケーション1Passwordとしてを追加しました。詳細については、AWS AppFabric でサポートされているアプリケーション」を参照してください。	2024 年 4 月 23 日
サポートされている新しいセキュリティツール	互換性のあるセキュリティツールDynatraceとして を追加しました。詳細については、「互換性のあるセキュリティツール」を参照してください。	2024 年 3 月 26 日
新しいメトリクス	AppFabric App Authorization Status メトリクスを追加しました。詳細については、「Monitoring AWS AppFabric with Amazon CloudWatch Logs」を参照してください。	2024 年 3 月 8 日
サポートされている新しいアプリケーション	サポートされているアプリケーションIBM Security® Verifyとして を追加しました。詳細については、AWS AppFabric でサポートされているアプリケーション」を参照してください。	2024 年 3 月 6 日
サポートされている新しいアプリケーション	サポートされているアプリケーションBoxとして を追加しました。詳細については、AWS AppFabric でサポートされているアプリケーション」を参照してください。	2024 年 2 月 1 日

サポートされている新しいアプリケーションとメトリクス	サポートされているアプリケーションTerraform Cloud として、Cisco Duo、Sales force、 を追加しました。詳細については、AWS AppFabric でサポートされているアプリケーション」を参照してください。AppFabric Data Delivery Latency メトリクスと全体的なデータ遅延メトリクスを追加しました。詳細については、「Monitoring AWS AppFabric with Amazon CloudWatch Logs」を参照してください。	2024 年 2 月 1 日
サポート対象のアプリケーションとして、Atlassian Confluence、Genesys Cloud、HubSpot、OneLogin by One Identity、PagerDuty、Ping Identity を、互換性のあるセキュリティツールとして Barracuda XDR を追加しました。	サポートされている新しいアプリケーションの詳細については、AWS AppFabric および互換性のあるセキュリティツールでサポートされているアプリケーション」を参照してください。 https://docs.aws.amazon.com/appfabric/latest/adminguide/security-tools.html	2023 年 12 月 15 日
サポート対象のアプリケーションとして、Atlassian Confluence、Genesys Cloud、HubSpot、OneLogin by One Identity、PagerDuty、Ping Identity を、互換性のあるセキュリティツールとして Barracuda XDR を追加しました。	サポートされている新しいアプリケーションの詳細については、AWS AppFabric および互換性のあるセキュリティツールでサポートされているアプリケーション」を参照してください。 https://docs.aws.amazon.com/appfabric/latest/adminguide/security-tools.html	2023 年 12 月 15 日

AWS AppFabric for productivity プレビュードキュメントを追加しました	AppFabric for productivity の詳細については、「What is AWS AppFabric for productivity?」を参照してください。	2023 年 11 月 27 日
サポートされているアプリケーションとして GitHub および ServiceNow を追加しました。	新たにサポートされるアプリケーションの詳細については、「サポートされているアプリケーション」を参照してください。	2023 年 10 月 31 日
AWS AppFabric AWS の管理ポリシーの追跡を開始しました	AppFabric の AWS 管理ポリシーの詳細については、AWSAWS AppFabric の管理ポリシー」を参照してください。	2023 年 6 月 27 日
初回リリース	AWS AppFabric 管理ガイドの初回リリース。	2023 年 6 月 27 日

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。