記事
Toshihiko Minamoto · 2021年7月26日 5m read

InterSystems API Management で OAuth 2.0 を使用して API を保護する - パート 2

この 3 部構成の記事では、IAM を使って、以前に IRIS にデプロイされた認証されていないサービスに OAuth 2.0 標準に従ったセキュリティを追加する方法を説明します。

パート 1 では、サービスを保護するプロセス全体を理解しやすくするために、IRIS と IAM の基本的な定義と構成を示しながら OAuth 2.0 の背景を説明しました。

このパートでは、着信リクエストに存在するアクセストークンを検証し、検証が成功した場合にはそのリクエストをバックエンドに転送するように IAM を構成する手順について詳しく説明します。

この連載の最後のパートでは、IAM でアクセストークンを生成し（承認サーバーとして機能します）、それを検証するようにするための構成を説明し、重要な最終考慮事項を示します。

IAM をお試しになりたい方は、InterSystems 営業担当者にお問い合わせください。

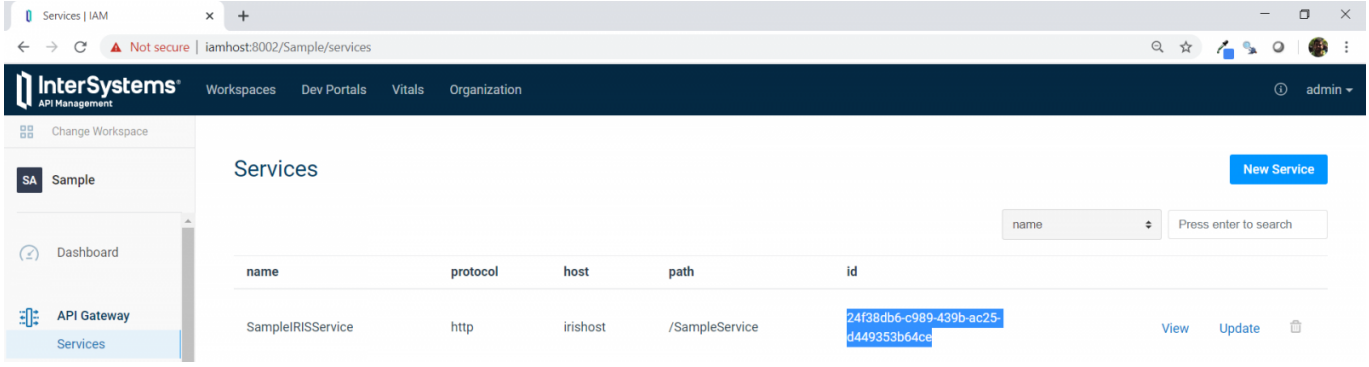
シナリオ 1: アクセストークンのバリデーターとして機能する IAM

このシナリオでは、JWT（JSON Web トークン）形式でアクセストークンを生成する外部承認サーバーを使用します。 この JWT はアルゴリズム RS256 と秘密鍵を使用して署名されています。 JWT 署名を検証するには、ほかのグループ（この場合 IAM）に承認サーバーが提供する秘密鍵が必要です。

外部承認サーバーが生成するこの JWT には、本体に、このトークンの有効期限を示すタイムスタンプを含む「exp」と呼ばれるクレームと、承認サーバーのアドレスを含む「iss」と呼ばれる別のクレームも含まれます。

したがって、IAM はリクエストを IRIS に転送する前に、承認サーバーの秘密鍵と JWT 内の「exp」クレームに含まれる有効期限のタイムスタンプを使用して、この JWT 署名を検証する必要があります。

これを IAM で構成するために、まず、IAM の「SampleIRISService」に「JWT」というプラグインを追加しましょう。 追加するには、IAM のサービスページに移動して「SampleIRISService」の ID をコピーします。これは後で使用します。



コピーしたら、プラグインに移動して［新規プラグイン］ボタンをクリックし、「JWT」プラグインを見つけて

［有効化］をクリックします。

次のページで、［serviceid］フィールドに「SampleIRISService」の ID
を貼り付け、［config.claimstoverify］パラメーターの「exp」ボックスを選択します。

［config.keyclaimname］パラメーターの値が「iss」であることに注意してください。 これは後で使用します。

次に、［作成］ボタンをクリックします。

クリックしたら、左メニューの「コンシューマー」セクションに移動し、前に作成した「ClientApp」をクリックします。 ［資格情報］タブに移動し、［新しい JWT 資格情報］ボタンをクリックします。

次のページで、JWT の署名に使用されるアルゴリズム（この場合
RS256）を選択肢、［rsapublickey］フィールドに公開鍵（PEM
形式で承認サーバーから提供された公開鍵）を貼り付けます。

［鍵］フィールドには、JWT プラグインを追加したときに［config.keyclaimname］フィールドに入力した JWT クレームのコンテンツを挿入する必要があります。 したがって、この場合は、JWT の iss
クレームのコンテンツを挿入する必要があります。私の場合、このコンテンツは承認サーバーのアドレスです。

挿入したら、［作成］ボタンをクリックします。
ヒント: デバッグ用に、JWT をデコードするオンラインツールがあります。それに公開鍵を貼り付けると、クレームとその値を確認して、署名を検証することができます。 このオンラインツールは https://jwt.io/#debugger にあります。

JWT プラグインが追加されたため、認証無しでリクエストを送信することはできなくなりました。
以下に示すように、単純な GET リクエストを認証なしで次の URL に送信する場合、

http://iamhost:8000/event/1

「401 Unauthorized」ステータスコードで不正なメッセージが返されます。

IRIS から結果を取得するには、リクエストに JWT を追加する必要があります。

したがって、最初に承認サーバーに JWT をリクエストする必要があります。
ここで使用しているカスタム承認サーバーは、POST
リクエストが、ユーザーやクライアント情報を含むキー値ペアとともに次の URL に送信された場合に JWT を返します。

https://authorizationserver:5001/auth

このリクエストとそのレスポンスは次のようになります。

次に、レスポンスから取得した JWT

を承認ヘッダーの下にベアラートークンとして追加し、以前に使用したのと同じ URL に GET リクエストを送信することができます。

http://iamhost:8000/event/1

または、クエリ文字列パラメーターとして追加することも可能です。クエリ文字列のキーは、JWT プラグインを追加したときに［config.uriparamnames］フィールドに指定された値（この場合は「jwt」）です。

最後に、［config.cookiename］フィールドに名前が入力されている場合は、JWT を cookie としてリクエストに含めるオプションもあります。

IAM でアクセストークンを生成して検証するために必要な構成と重要な最終考慮事項を理解するには、この連載のパート 3 であり最後となる記事をご覧ください。

#API #OAuth2 #REST API #セキュリティ #InterSystems IRIS

ソースURL:https://jp.community.intersystems.com/post/intersystems-api-management-%E3%81%A7-oauth-20-%E3%82%92%E4%BD%BF%E7%94%A8%E3%81%97%E3%81%A6-api-%E3%82%92%E4%BF%9D%E8%AD%B7%E3%81%99%E3%82%8B-%E3%83%91%E3%83%BC%E3%83%88-2