記事
Tomohiro Iwamoto · 2020年8月13日 7m read

AWS上にて稼働中のInterSystems IRISの管理ポータルとの通信を暗号化(https)する方法

本記事について

InterSystems IRISは、管理ポータルへのアクセス方法がデフォルトではhttpとなっており、クライアントが社内、サーバがクラウドという配置の場合、なんらかの方法でトラフィックを暗号化したいと考える方も多いかと思います。
そこで、AWS上にて稼働中のIRISの管理ポータル(あるいは各種RESTサービス)との通信を暗号化する方法をいくつかご紹介したいと思います。

> 本記事では、アクセスにIRIS組み込みのapacheサーバを使用しています。ベンチマーク目的や本番環境のアプリケーションからのアクセス方法としては使用しないでください。
> 短期間・少人数での開発・動作検証・管理目的でのアクセスを暗号化する事を想定しています。

ドメイン名とメジャーな認証局発行のSSLサーバ証明書を用意できればベストなのですが、上記のような用途の場合、コスト面でなかなか難しいと思います。
ですので、下記の証明書の使用を想定しています。
- 自己署名(いわゆるオレオレ証明書)
- 自分で建てた認証局で署名した証明書(いわゆるオレオレ認証局)

また、下記のような実行環境を想定しています。

手元のPC環境	O/S	Windows10
	ブラウザ	Chrome/FireFox/E
	IDE	vscode+ObjectScri
	ローカルPCで未使用のポート番号	8888
	EC2インスタンス作成時に使用したキーペアの秘密鍵	aws-secret.pem

AWS環境	IRISホストの公開ホスト名	ec2-54-250-169-xxx.ap
	IRISのウェブサーバポート番号	52773
	O/S	Ubuntu 18.04LTS
	O/Sユーザ名	ubuntu

直接アクセス

1) ポートフォワーディングを使う

最もお手軽な方法です。
セキュリティグループで、SSH用のポート(22)をインターネットからのインバウンド許可する必要があります。

```
C:\Users\xxxx>ssh -i aws-secret.pem -L 8888:localhost:52773 ubuntu@ec2-54-250-169-xxx
.ap-northeast-1.compute.amazonaws.com
```

このコマンドを実行中は、SSHを実行したPCからは、IRISホストに

http://localhost:8888/csp/sys/%25CSP.Portal.Home.zen

でアクセスできます。
sshでログインした状態になっているので、開発時に行うような端末操作(IRISの起動・停止、IRISセッション開始など)に利用できます。
この方法は、スーパーサーバ・ポート(51773)にも有効ですので、Studioによる通信の暗号化にも使用できます。

注意)Windowsの場合、ssh用の秘密鍵(aws-secret.pem)を%USERPROFILE%\に配置しないと、エラーになります。

```
C:\Users\xxxx>dir %USERPROFILE%\aws-secret.pem
2020/07/14  17:10             1,692 aws-secret.pem
               1 ??????               1,692 ???
               0 ????????  108,323,176,448 ????????
```

vscodeの設定(settings.json)は下記のようになります。

```
{
    "objectscript.conn": {
        "host": "localhost",
        "https": false,
        "port": 8888,
        "ns": "USER",
        "username": "xxx",
        "password": "xxx",
        "active": true
    }
}
```

2) SSL設定を施したReverse Proxyを使う

IRISホストに自己証明書を適用したapacheやnginxをReverseProxy設定で配置します。
セキュリティグループで、HTTPS用のポート(443)をインターネットからのインバウンド許可する必要があります。

apache及びnginxの設定を行うスクリプトをこちらに掲載しています。これにより、ブラウザからIRISホストに

https://ec2-54-250-169-xxx.ap-northeast-1.compute.amazonaws.com/csp/sys/...

でアクセスできます。

vscodeの設定(settings.json)は下記のようになります。

```
{
```

```
    "objectscript.conn": {
        "host": "ec2-54-250-169-xxx.ap-northeast-1.compute.amazonaws.com",
        "https": true,
        "port": 443,
        "ns": "USER",
        "username": "xxx",
        "password": "xxx",
        "active": true
    }
}
```

踏み台ホスト経由

検証用とはいえ、ユーザデータやコードを含むEC2インスタンスのSSHやHTTPSポートをインターネット解放するのは不安がある場合は、踏み台ホスト(Bastion Host)を使用しますが上記の方法は、踏み台ホストがある場合でも同様です。
セキュリティグループで、踏み台ホストとIRISホスト間のTCPトラフィックをインバウンド許可する必要があります。

下記のような実行環境であると想定します。

AWS環境	IRISホストの公開ホスト名	なし
	IRISホストの内部IPアドレス	10.0.1.81
	踏み台ホストの公開ホスト名	ec2-54-250-169-yyy.ap-

1) ポートフォワーディングを使う

セキュリティグループで、SSH用のポート(22)をインターネットからのインバウンド許可する必要があります。

下記のコマンドを、踏み台ホストに対して実行します。

```
C:\Users\xxxx>ssh -i aws-secret.pem -L 8888:10.0.1.81:52773 ubuntu@ec2-54-250-169-yyy
.ap-northeast-1.compute.amazonaws.co
```

以下、同様です。

2) SSL設定を施したReverse Proxyを使う

セキュリティグループで、HTTPS用のポート(443)をインターネットからのインバウンド許可する必要があります。
同じ作業を、踏み台ホストで実行します。
転送先のURLをIRISホストの内部IPアドレス:10.0.1.81に変更します。

```
ProxyRequests Off
ProxyPass / http://10.0.1.81:52773/
ProxyPassReverse / http://10.0.1.81:52773/
```

以下、ホスト名が踏み台ホスト名に変わる事以外は、同様です。

AWS/ALB経由

あまり一般的ではないかもしれませんが、AWS/ALBに自己証明書を適用することができます。この場合、ALBにてSSL終端させるので、別途SSL有効化したapacheを用意する手間が省けます。

ALBの作成には、最低2つのAZの指定が必要なので、ここではICM(InterSystems Cloud Manager)にて、DM(2台)をMIRROR:trueで作成した環境を使用しました。
(ICMについては、icmを利用してIRISクラスターを構成する方法をご覧ください)

default.json (抜粋)

```
{
  "Zone": "ap-northeast-1a,ap-northeast-1c",
  "Mirror": "true",
}
```

definitions.json

```
[
    {
        "Role": "DM",
        "Count": "2",
        "MirrorMap": "primary,backup",
        "ZoneMap": "0,1"
    }
]
```

setup.shで作成した証明書ファイルを使用します。
これらのファイルをACM(Certificate Manager)にインポートします。
証明書本文:server.crtの内容
証明書のプライベートキー:server.keyの内容
証明書チェーン:inca.pemの内容

補足)awscliを使える環境であれば、setup.shの最後の行をアンコメントすれば自動登録されます。

下記設定でALBを新規作成します。

手順 1: ロードバランサーの設定
名前:任意
スキーム:インターネット向け
IP アドレスタイプ:ipv4
リスナー:https(port:443)
アベイラビリティーゾーン:ap-northeast-1a,ap-northeast-1c

手順 2: セキュリティ設定の構成
デフォルトの証明書の選択:ACM から証明書を選択する
証明書の名前:(先ほどACMにインポートした証明書を選択)

手順 3: セキュリティグループの設定
セキュリティグループの設定:新しいセキュリティグループを作成する→https(port:443)のみを許可。

手順 4: ルーティングの設定
ターゲットグループ:新しいターゲットグループ
名前:任意
ターゲットの種類:インスタンス
プトロコル:http
ポート:52773

ヘルスチェック
プトロコル:http
パス:/csp/bin/mirrorstatus.cxw
ヘルスチェックの詳細設定
ポート:上書き→52773

手順 5: ターゲットの登録
先ほど作成したEC2インスタンスを登録済みに追加

作成したALBの状態がactiveになれば、ALBのDNS名を使ってhttpsアクセスできるようになります。

https://[DNS名]/csp/sys/exp/%25CSP.UI.Portal.SQL.Home.zen

#AWS #セキュリティ #管理ポータル #開発環境 #InterSystems IRIS #InterSystems IRIS for Health

ソースURL:
https://jp.community.intersystems.com/post/aws%E4%B8%8A%E3%81%AB%E3%81%A6%E7%A8%BC%E5%83%8D%E4%B8%AD%E3%81%AEintersystems-iris%E3%81%AE%E7%AE%A1%E7%90%86%E3%83%9D%E3%83%BC%E3%82%BF%E3%83%AB%E3%81%A8%E3%81%AE%E9%80%9A%E4%BF%A1%E3%82%92%E6%9A%97%E5%8F%B7%E5%8C%96https%E3%81%99%E3%82%8B%E6%96%B9%E6%B3%95