探索的ビッグデータ解析と再現可能研究 (WS-EBDA-RR-2023)

アマゾンウェブサービスと Daily TAQ データ

小池祐太 (東京大学, CREST JST)

2023 年 8 月 27 日

東京大学大学院数理科学研究科, CREST JST

1. Daily TAQ データとは
2. アマゾンウェブサービス (AWS) とは
3. 実際のアクセス手順
   - (1) IAM ユーザーの作成
   - (2) ユーザーポリシーの更新 (スキップ可能)
   - (3) VPC の作成/設定
   - (4) EC2 インスタンスを作成
   - (5) EC2 インスタンスから S3 バケットにアクセス

Daily TAQ データとは

Daily TAQ データとは

- **ティックデータ**
  - 金融市場で取引される金融資産の価格や取引数, 注文数などの情報を, 約定や注文が発生するごとに記録したデータ
- **Daily TAQ データ** †
  - ニューヨーク証券取引所 (NYSE) が販売しているティックデータセットの 1 つ
  - 米国証券市場の主要取引所で取引されたすべての銘柄の最良気配と約定に関するデータを記録
  - ウェブサイトから注文して購入できる

---

†詳細はウェブサイト `https://www.nyse.com/market-data/historical/daily-taq` を参照

Daily TAQ データとは

- 購入後, データにアクセスしてダウンロードする必要がある
- データへのアクセス方法は何度か変遷してきた (年は推定):
  - 2018 年ごろまで: FTP (File Transfer Protocol) サーバへの接続
  - 2019 年ごろ ∼: SFTP (SSH File Transfer Protocol) による MFT (Managed File Transfer) サイトへのアクセス
    - 2022 年秋に MFT サイトは廃止され, SFTP サーバへのアクセスに切り替わった模様
  - 2021 年 11 月 ∼: Amazon Simple Storage Service (Amazon S3) のバケットへのアクセス

本日の目的

NYSE の S3 バケットに (なんとか) アクセスするための手順を説明

アマゾンウェブサービス (AWS) とは

アマゾンウェブサービス (AWS) とは

- Amazon S3 は, アマゾン社のクラウドコンピューティングサービス「**アマゾンウェブサービス (Amazon Web Services, AWS)**」におけるサービスの１つ
- 個々のサービスの利用は基本的に有料だが, 無料利用枠がある (サービスごとに異なる)
  - Amazon S3 の場合, 最初の１年間は 5GB まで無料
  - ただし, 以下で最低限必要なのは Amazon EC2;
    こちらも最初の１年間は１ヶ月につき 750 時間まで無料
  - 詳細は AWS のサイト参照
- サービス利用のためのアカウントの作成自体は無料でできる

アマゾンウェブサービス (AWS) とは

- ちょっとしたディスクレーマー
  - 私は
    NYSE の S3 バケットにアクセスして Daily TAQ データを取得する
    という目的以外では AWS を一切使わないので, 本日はその方法のみに特化して話します
    - AWS の一般論のようなことは一切話しません (そもそも話せません)
  - そのため, Daily TAQ データの購入を検討している方以外には, あまり益のない話かもしれません
  - 一方で, 上の目的を達成するには,
    他アカウントが所有していて, かつ (公開) インターネットから直接アクセスできない S3 バケットにアクセスする
    ということをする必要があるので, Daily TAQ に限らず類似のケースに対応する上で役立つかもしれない, とも少し期待しています

実際のアクセス手順

実際のアクセス手順

- NYSE によるマニュアル:

  `https://www.nyse.com/publicdocs/nyse/data/NYSE_TAQ_Data_AWS_Cloud_Access_Dev_Instructions.pdf`

- 何が面倒か?
  - アクセスしたい S3 バケットは NYSE アカウント下のものであり, NYSE に付与されたアクセス権 (正確には IAM ロール) を持った状態でないとアクセスできない
  - インターネット経由ではアクセスできず, Amazon Virtual Private Cloud (VPC) 経由でアクセスする必要がある
  - (※ 用語は後ほど順次説明するが, 私の理解は正確でない可能性が高い)

実際のアクセス手順

だいたいの流れ

(1) IAM ユーザーを作成し, NYSE に S3 バケットへのアクセス権 (IAM ロール) を付与してもらう

(2) NYSE から付与された IAM ロールを利用できるようユーザーポリシーを更新する (スキップ可能)

(3) VPC を作成/設定

(4) EC2 インスタンスを作成

(5) EC2 インスタンスに接続して, AWS CLI コマンドを用いて NYSE の S3 バケットにアクセス

(1) IAM ユーザーの作成

- IAM (Identity and Access Management の略) は, AWS のサービスの 1 つで, AWS の各サービスやリソースへのアクセス権を管理するためのサービス
- **IAM ユーザー**: アカウント内で作成できるユーザーで, 個別にユーザー名と PW をもつ
  - 複数作成可能
  - PC を使う際に作成するユーザーのようなもの (アカウントが PC に対応)
- アカウント作成段階ではルートユーザーしかおらず, 普段使いには IAM ユーザーとしてサインインすることが推奨されている

(1) IAM ユーザーの作成

- IAM ユーザーの作成は特に難しくない (ネットで検索すればわかる)
  - AWS マネジメントコンソールにサインインし, IAM ダッシュボードにアクセス (上部の検索窓に「IAM」と入力すればよい)
  - 左部の「アクセス管理 > ユーザー」にアクセスし, 「ユーザーの作成」を実行
  - 許可の設定のところで, 管理者権限にあたる「AdministratorAccess」ポリシーをアタッチしておくと, サービスごとのアクセス許可の設定をいじらずに済むので後の作業が楽 (セキュリティ的にはよくないかも)
- IAM ユーザー作成後, 「AWS のアカウント ID」と「IAM ユーザーのリソース名 (ARN)」を NYSE に知らせる (マイページのようなところから登録できる)

(1) IAM ユーザーの作成

- アカウント ID は, コンソール右上のユーザー名のところから確認できる
- IAM ユーザーの ARN は, IAM ダッシュボード左部の「ユーザー」を選択し, リストから当該ユーザー名を選べば確認できるが, 以下のような形式である:

  ```
  arn:aws:iam::「アカウントID」:user/「IAMユーザー名」
  ```

  - 「アカウント ID」と「IAM ユーザー名」の箇所は適切なものに置き換える

(2) ユーザーポリシーの更新 (スキップ可能)

- IAM ユーザーの各サービス/リソースへのアクセス権限は, **IAM ポリシー**によって管理されている
- 「NYSE の S3 バケットへのアクセス権限」に対応する **IAM ロール**もリソースの 1 つであるから, 利用にはアクセス権限が必要
- 従って, 先ほど作成した IAM ユーザーに, NYSE から付与された IAM ロールへのアクセス権限 (**AssumeRole** と呼ばれる) を与える必要がある
- しかし, この IAM ユーザーが管理者権限を持っていれば IAM ロールへのアクセス権限もあるので, このステップはスキップ可能

(3) VPC の作成/設定

- NYSE の S3 バケットはセキュリティ上の問題からインターネットからアクセスできないため, **Amazon Virtual Private Cloud (VPC)** 経由でアクセスする必要がある
- VPC: AWS クラウド上の仮想ネットワーク
- 次ステップで VPC 内に **EC2 インスタンス**という仮想サーバを設置して, そこ経由で S3 にアクセスする
- しかし, S3 自体は VPC 外に存在するサービスのため, VPC から S3 にプライベート接続するために**エンドポイント**というものを作成する必要がある

(3) VPC の作成/設定

- エンドポイントには 2 種類ある
  (a) ゲートウェイエンドポイント
  (b) インターフェイスエンドポイント
- ここでは (a) を使う
  - (b) を使ってどうやるかわかっていないため
  - そもそも (b) は時間あたり利用料がかかるので, 無料の (a) を使う
- 以上をまとめると, このステップで行うことは以下の 2 つ

(i) VPC の作成 (実はスキップ可能)

(ii) ゲートウェイエンドポイントの作成

(3) VPC の作成/設定

- NYSE の S3 バケットが us-east-1 リージョンにあるため, VPC も同じリージョンに作成する必要がある
  → AWS コンソール右上部のリージョンを us-east-1 (バージニア北部) に変更しておく
- 実は, VPC はリージョンごとにデフォルトのものが 1 つ用意されているので, それを使えば新たに作成する必要はない
  - デフォルト VPC はインターネットからのアクセス制限に関するデフォルト設定がないので, プロダクション環境での使用はセキュリティ的に非推奨らしい
  - 新たに作成する場合, デフォルトではインターネットからのアクセスができないので, **インターネットゲートウェイ**を追加する必要がある

(3) VPC の作成/設定

- ゲートウェイエンドポイントの作成手順:
  1. VPC ダッシュボード左部の「エンドポイント」を選択
  2. 右上部の「エンドポイントを作成」を選択
  3. エンドポイントの設定
     - 名前は適当に設定. 「サービスカテゴリ」は「AWS のサービス」
     - 「サービス」の箇所は「com.amazonaws.us-east-1.s3」を選択 (S3 に接続するためのゲートウェイエンドポイント)
     - 「VPC」はエンドポイントを作成する VPC を選択. ここではデフォルト VPC を選択
     - 「ルートテーブル」はメインが「はい」のものをチェック (そもそもデフォルト VPC は 1 つしかルートテーブルがないはず)
     - あとはそのままで「エンドポイントを作成」を実行

(4) EC2 インスタンスを作成

- EC2 インスタンスの作成手順:
  1. EC2 ダッシュボード左部の「インスタンス」を選択
  2. 右上部の「インスタンスを起動」を選択
  3. インスタンスの設定エンドポイントの設定
     - 「キーペア」の箇所でログイン用のキーペアを指定. なければ「新しいキーペアの作成」を実行して作成すればよい (名前だけ適当につけて作成すればよい)
     - 他の設定は特にいじる必要はない. ネットワーク設定では前ステップで作成・設定した VPC を指定する必要があるが, デフォルト VPC がすでに指定されているはず
     - 「インスタンスを起動」を実行することで, EC2 インスタンスが作成して起動する
  4. 起動後は「インスタンスに接続」を選択することで, 接続方法が表示される (次ステップで説明)

(4) EC2 インスタンスを作成

注意事項

- EC2 インスタンスは起動時間に応じて使用料が発生するので, 使っていないときは停止しておく方がよい
- 停止するには, EC2 ダッシュボード左部の「インスタンス」を選択し, リストから停止したいインスタンスをチェックして, 「インスタンスの状態」から「インスタンスを停止」を選択
- 「インスタンスを終了」を選ぶとインスタンス自体が完全削除されてしまうので注意
- 再び起動したい場合は, 起動したいインスタンスを上と同じ手順で選んで, 「インスタンスの状態」から「インスタンスを起動」を選択
- なお, インスタンスのストレージも使用量がかかるので, 不要なファイルは適宜削除するのがよい

(5) EC2 インスタンスから S3 バケットにアクセス

- EC2 インスタンスへの接続方法は 4 種類あることになっているが, 私はそのうちの 2 つしか使ったことがない (そもそも他が使えない):
  - ブラウザベースの接続 (EC2 Instance Connect)
  - SSH による接続
- どちらがよいということもないが, ブラウザベースの方が説明が楽なので, ここではこちらを使う

(5) EC2 インスタンスから S3 バケットにアクセス

- EC2 インスタンスへの接続手順 (ブラウザベース):
  1. EC2 ダッシュボードの「インスタンス」から, 起動状態のインスタンスを選択
  2. 上部の「接続」を選択
  3. 「EC2 Instance Connect」タブを選択し, 下部の「接続」を実行
- 通常のコマンドラインと同じ感覚で操作できる
- また, **AWS コマンドラインインターフェイス (AWS CLI)** というツールがインストール済みで, このツールのコマンドを用いることで AWS の各種サービスにアクセスできる
- しかし, 実際に利用するには認証情報等の設定が必要なので, それを説明する

(5) EC2 インスタンスから S3 バケットにアクセス

- EC2 インスタンスから AWS のリソースにアクセスするには, そのリソースを所有するアカウントの認証情報を設定する必要がある
- 認証情報ファイルは `aws configure` というコマンドで作成できる
- このコマンドを実行すると次の 4 つの情報の入力を要求される:
  1. **AWS Access Key ID**:
     認証に使う IAM ユーザーのアクセスキーの ID
  2. **AWS Secret Access Key**:
     認証に使う IAM ユーザーのシークレットアクセスキー
  3. **Default region name**: 「us-east-1」と設定すればよい
  4. **Default output format**: 「json」と設定すればよい

(5) EC2 インスタンスから S3 バケットにアクセス

- アクセスキーは IAM ダッシュボードから作成可能:
  1. IAM ダッシュボード左部の「ユーザー」を選択し, リストからキーを発行したいユーザー名を選ぶ
  2. 「アクセスキーを作成」を選択
  3. 「ユースケース」は「コマンドラインインターフェース」を選択
     - 代替案が推奨されるが VPC との併用はできないので無視
  4. 「確認」をチェックし次に進み, 説明タグを適当に入力し「アクセスキーを作成」を実行
  5. 作成されたアクセスキーの情報は後で使うので, CSV ファイルをダウンロードしておくと便利 (セキュリティ的にはよくないが)

(5) EC2 インスタンスから S3 バケットにアクセス

- `aws configure` を実行すると, `~/.aws/` 下に `config` (設定ファイル) と `credentials` (認証情報ファイル) が作成される
- 後者にアクセスキーの情報, 前者にその他の設定が保持される
- 認証情報を設定すると, (アクセスキー発行元ユーザーの権限が及ぶ限り) **自アカウントのリソース**には自由にアクセス可能になる
- 例: 自アカウントの S3 に abcbucket という名前のバケットがある場合,

  ```
  aws s3 ls s3://abcbucket
  ```

  というコマンドでバケットに含まれるオブジェクトをリストアップできる

(5) EC2 インスタンスから S3 バケットにアクセス

- NYSE の S3 バケット名は nyse.taq.prod.rawfiles なので

```
aws s3 ls s3://nyse.taq.prod.rawfiles
```

  というコマンドでバケットの中身をリストアップできる...

- と言いたいところだが, 他アカウントのリソース (今の場合 NYSE の S3 バケット) にアクセスするには, そのアカウントから付与された権限 (IAM ロール) を使って AWS CLI を実行する必要がある
  → あとひと作業必要

(5) EC2 インスタンスから S3 バケットにアクセス

- NYSE のマニュアルには, `config` ファイルを以下のように書くよう記載がある:

```
[default]
region = us-east-1
output = json
role_arn =「NYSEから付与されたIAMロール名」
source_profile=default
sts_regional_endpoints = regional
```

- しかし, これは動かない

(5) EC2 インスタンスから S3 バケットにアクセス

- 原因 ソースプロファイルの設定
  - プロファイルとは認証情報を含む設定の総称で, 指定しなければ `default` プロファイルが使われる (default プロファイルは先ほどの `aws configure` コマンドで作成されたもの)
  - 特定の IAM ロールを使って AWS CLI を実行するには, `role_arn` にその IAM ロール名を設定する必要がある
  - さらに, IAM ロールを使用するには利用権限を持つユーザーの認証情報を持つプロファイル (ここでは `default`) を `source_profile` に設定しておく必要がある
  - しかし, 自分自身をソースプロファイルに設定することはできない

(5) EC2 インスタンスから S3 バケットにアクセス

- 解決策 別のプロファイルを新たに作成し上の設定を書き込む
- 例えば, nyse というプロファイルを新たに作成して設定を書き込む場合は, config ファイルを直接編集して以下の内容を追加すればよい:

```
[profile nyse]
role_arn =「NYSEから付与されたIAMロール名」
source_profile=default
sts_regional_endpoints = regional
```

(5) EC2 インスタンスから S3 バケットにアクセス

- あとは, `aws s3` コマンドを`--profile nyse` オプションをつけて実行すれば, NYSE の S3 バケットにアクセスできる
  - オプションは「`nyse` プロファイルの設定を用いて `aws` を実行する」という意味
- 例えば以下のコマンドでバケット内のファイル・ディレクトリをリストアップできる

```
aws s3 ls s3://nyse.taq.prod.rawfiles --profile nyse
```

(5) EC2 インスタンスから S3 バケットにアクセス

- Daily TAQ データを EC2 インスタンスにコピーするコマンドの例:

```
aws s3 cp s3://nyse.taq.prod.rawfiles/3_DAILY_TAQ/EQY_US_ALL_REF_MASTER/
    EQY_US_ALL_REF_MASTER_2023/EQY_US_ALL_REF_MASTER_202307/
    EQY_US_ALL_REF_MASTER_20230703.gz ~/ --profile nyse
```

- EC2 インスタンス上のファイルをローカルにコピーするには, **ローカル側で** `scp` コマンドを使うとよい: 以下のような感じのコマンドになる

```
scp -i キーペア名.pem ec2-user@インスタンスのパブリックIPアドレス:コピー元パス コピー先パス
```

(「キーペア名.pem」が (ローカル側の) 作業ディレクトリにないといけない)

(5) EC2 インスタンスから S3 バケットにアクセス

- 「NYSE の S3 バケットのデータ」を「自アカウントの S3 バケット」にコピーすることも可能なはずであるが, これはまだできていない
- コマンドは `aws s3 cp` でコピー先を「自アカウントの S3 バケットのディレクトリ」に変更すればよいだけだが, NYSE から付与された IAM ロールでは後者にアクセスする権限がないため, コピーが失敗してしまう
- 解決するには上記 IAM ロールに自アカウントの S3 バケットにアクセスする権限を付与すればいいはずだが, これをどうすればよいかわからずスタックしている (自アカウントの S3 バケットを使うとそこで料金が発生するため実行する動機もさほどないという理由もある)