CloudWatchでエラーログの確認方法

ローカル環境では開発サーバー用のターミナル、ブラウザのコンソールで確認できます。

AWS上ではCloudWatch Logsで以下の通り確認できます。

CloudWatch Logs Insightsとは

AWS CloudWatch に保存されたログを効率的に検索・分析するためのクエリベースのツールです。

AWSにデプロイしたWebサービスで、ログの確認がなかなかしづらいなと思う経験はないでしょうか。

個人的にはログが UTC 時間で区切られており、さらに同じ日の中でも複数の実行環境があればさらに、細かく分割されるため、エラーを見つけるのが大

そこで、「CloudWatch Logs Insights」を使用した下記の手順で対応しましたので、ご紹介いたします。

今後もより良い方法やノウハウが見つかれば随時更新していきます。（現状は備忘録的な内容となっておりご容赦ください）

今回私が実際に対応したのは Amplify コンソールでの SSR アプリケーション ですが、他のケースにも応用できると思います。

CloudWatch Logsのログのインサイトの利用手順

1. CloudWatchで「ログのインサイト」をクリック

2. クエリ言語を選択

クエリ言語	説明	選択の判断材料
ログのインサイト QL	CloudWatch Logs を直接検索・集計する専用クエリ言語	CloudWatch 上のログをそのまま確認・分析したいとき
OpenSearch PPL	OpenSearch に保存したログをパイプ形式で処理	OpenSearch でログを直感的に時系列処理・可視化したいとき
OpenSearch SQL	OpenSearch のログを SQL ライクに分析	OpenSearch でSQLに慣れた方法やBIツール連携をしたいとき

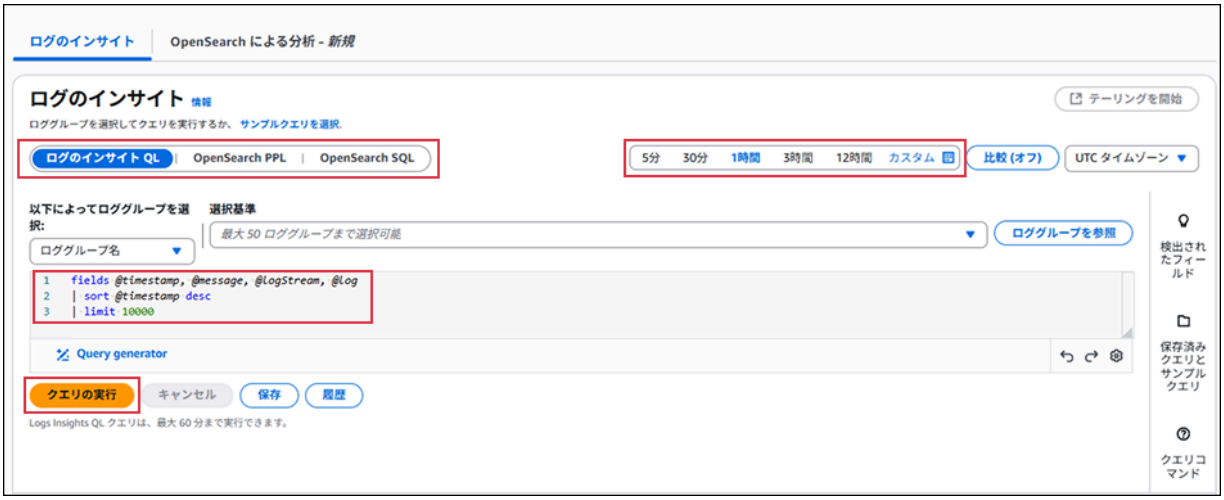
OpenSearchとは

Amazonが提供する オープンソースの検索・分析エンジン です。
大量データ、高度な検索向け

今回は軽量なログ検索なのでOpenSearchを経由せずに
「ログのインサイト QL」を使用

サポートされているクエリ言語
https://docs.aws.amazon.com/ja_jp/AmazonCloudWatch/latest/logs/CWL_AnalyzeLogData_Languages.html

3. 検索対象期間を指定（任意）



4. クエリの入力

```
fields @timestamp, @logStream, @message
| filter @logStream like /dev\//
| filter @message like /ERROR|Error|Exception|Unhandled|Task timed out|FAILED|失敗|エラー|NOT_FOUND/
| sort @timestamp desc
| limit 200
```

クエリの内容の説明

- fieldsで出力するフィールドを指定しています。
  - @timestamp … ログの時刻
  - @logStream … ログストリーム名（どの環境/実行単位か識別できる）
  - @message … 実際のログ本文
- ログストリーム名にdevが含まれるものを対象
- エラーらしきメッセージ（ERROR|Error|Exception|Unhandled|Task timed out|FAILED|失敗|エラー|NOT_FOUND）を含むログを対象
- タイムスタンプで降順