

トラブルから学ぶ、監視とデータ分析の重要性

富士ソフト株式会社
安斎 寛之



- Amazon Connect Delivery
- IoT Services Competency
- Managed Service Provider
- Govt. Services Competency
- Oracle Services Competency
- Migration Services Competency
- Well-Architected Partner Program

自己紹介



Anzai Hiroyuki

安齋 寛之

AWS Ambassador

富士ソフト株式会社
ソリューション事業本部インフラ事業部クラウドソリューション部
リーダー

～主な業務内容～
インフラエンジニアとしてヘルプデスク・サポートデスク業務から開始し、データセンターでのサーバ設計・構築・運用経験を経て、現在はパブリッククラウドの設計・構築・運用まで幅広く担当。過去の経験から、お客様の希望に沿ったうえで効率的なシステム構築を行う事を得意としている。
得意分野は自動化。





FUJISOFT
INCORPORATED

富士ソフト 株式会社

AWSプレミアティアサービスパートナー

AWSの最新技術に精通したアプリケーションとインフラ技術者が、安心安全スピーディーなAWSへの移行から、AWSを活用してビジネス変革をご支援するDX（デジタルトランスフォーメーション）を実現しています。

国内のAWSのパートナーで技術認定（コンピテンシー）の取得数で2番目の企業です。（2023年3月現在）



- Amazon Connect Delivery
- IoT Services Competency
- Managed Service Provider
- Govt. Services Competency
- Oracle Services Competency
- Migration Services Competency
- Well-Architected Partner Program

アジェンダ

本日お話する内容について

実際に体験したトラブルについて

オブザーバビリティについて

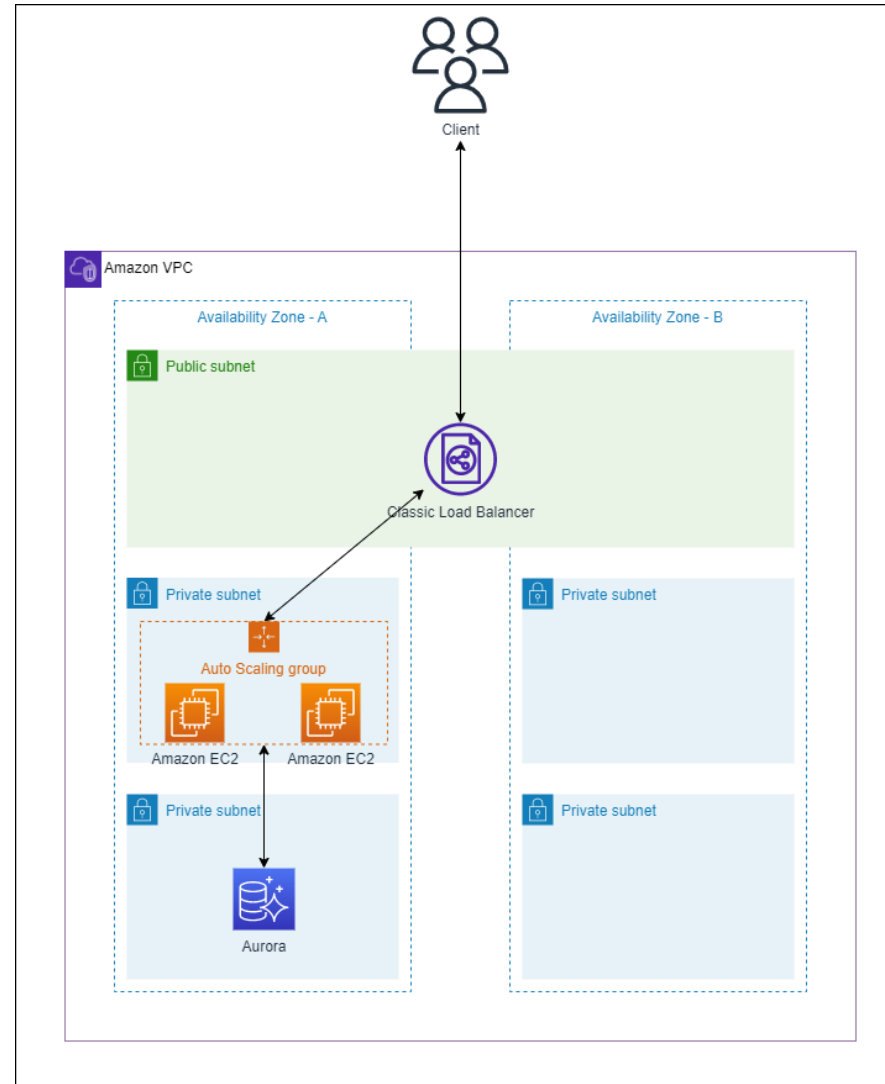
まとめ

本日本話しする内容について

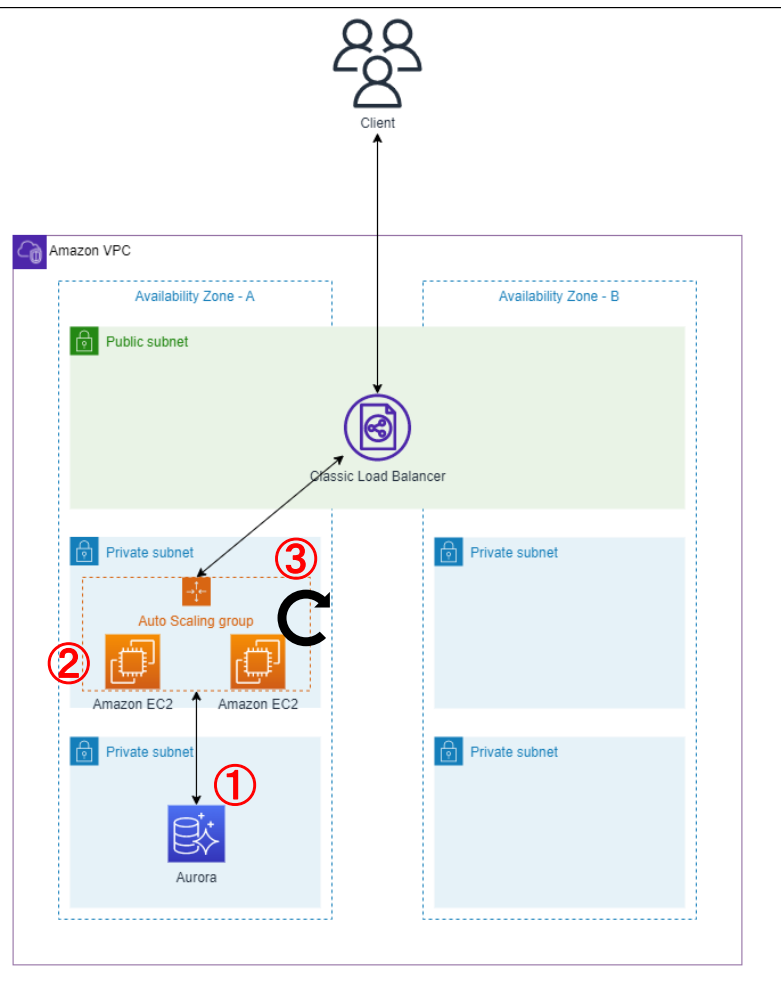
今日お集まりいただいたお客様には、私が実際に体験したトラブル案件を通して、**何を感じ、何を学び、何を今後すべきなのか**を検討した内容をお話したいと思います。



トラブル案件のシステム構成図



トラブル内容



Trouble

1

Auroraのバグを引く

Auroraのバージョン依存のバグによってトラフィックが増加した時にDB側が処理を捌ききれずにコネクションが拒否される

Trouble

2

Amazon EC2の負荷高騰

バックエンドのDBにアクセスできない事でAmazon EC2の負荷が増し結果としてCPU100%でハングし始める

Trouble

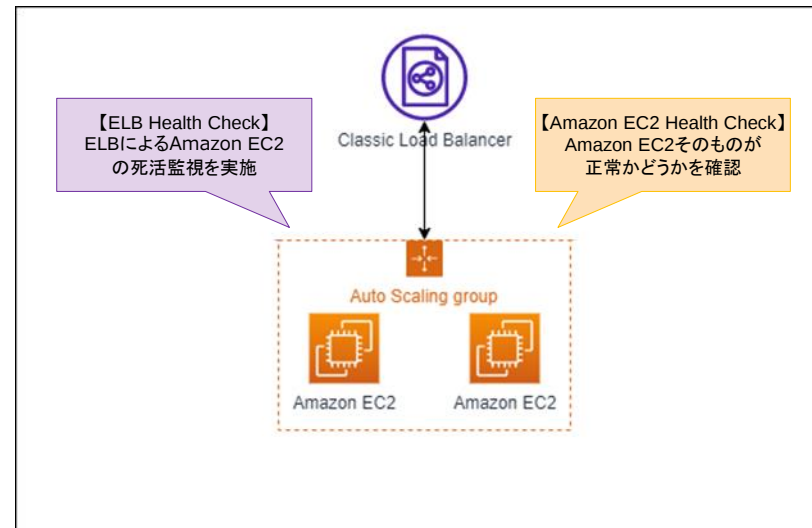
3

Auto Scalingのループ

Auto Scaling GroupのEC2インスタンスがELBからのヘルスチェックに失敗し起動されてはTerminateされるを繰り返す

トラブルの解決

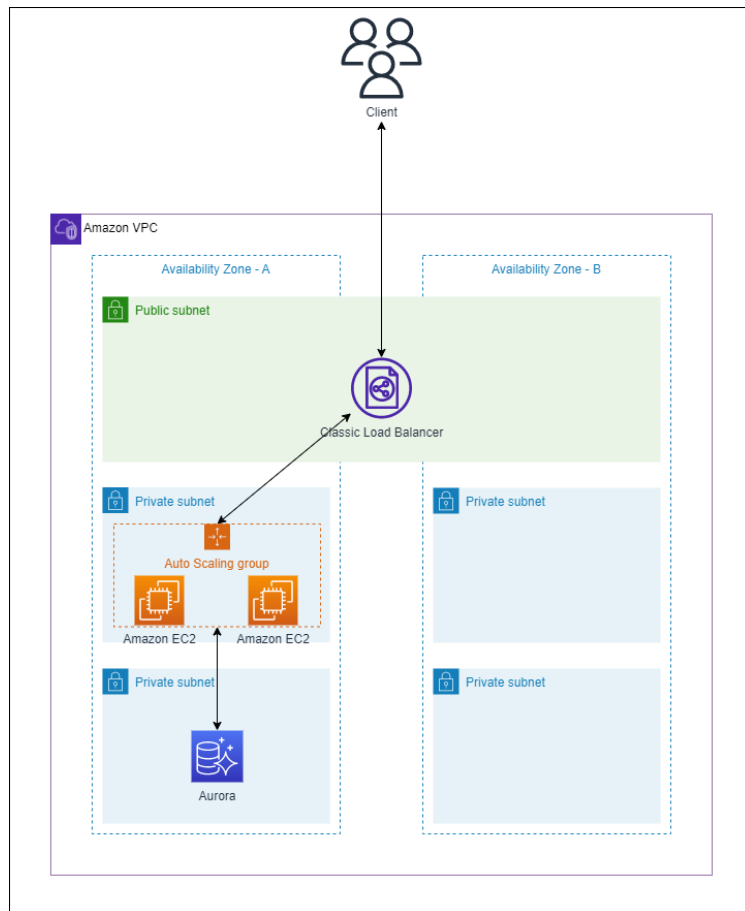
Auto Scaling groupのヘルスチェックからELBを外す事で解決



ヘルスチェックタイプ	チェックする事柄
Amazon EC2 ステータスチェックと予定されているイベント	- インスタンスが実行中であることをチェックします。 - ハードウェアまたはソフトウェアの根本的な問題で、インスタンスの機能を損なう可能性があるものをチェックします。 これは、Auto Scaling グループに対するデフォルトのヘルスチェックタイプです。
Elastic Load Balancing のヘルスチェック	ロードバランサーがインスタンスを正常として報告しているかどうかをチェックして、インスタンスがリクエストを処理できることを確認します。 このヘルスチェックタイプを実行するには、Auto Scaling グループに対してこのタイプを有効にする必要があります。
カスタムヘルスチェック	カスタムヘルスチェックに従って、インスタンスのヘルス問題を示す可能性のあるその他の問題をチェックします。

その後の調査が難航

トラブルは一旦収まったものの何が起因してこの状態が発生したのか原因を探ろうにも原因が突き止められない状況が発生した



1

Amazon EC2のログが取れてなかった

障害発生時のAmazon EC2のログが取得できていなかった為
OS・MW・App含めて当時何が起きていたのか追えない

2

Auroraのログが取れていなかった

かろうじて取得できていたのはerrorlogであった為
errorlogの内容を元にAborted Connectionが大量発生していたのは確認できたが、それ以上もそれ以下も追えない状態が発生

3

スケーリングポリシーがうまく機能していなかった

最大3台までスケールできるように設定していたが
平均値で80%以上のCPU使用率を15分継続する必要がある
結論としてそこまで達していないのでスケールできてなかった

問題

今回のシステムに対して貴方はどのようなアプローチを行いますか？



A person with dark hair, wearing a dark blue t-shirt, is holding black binoculars to their eyes. They are looking out over a vast, arid landscape with rolling orange-brown hills under a hazy, light-colored sky. The image has a soft, slightly blurred quality, emphasizing the concept of observation.

Observability (可観測性)

オブザーバビリティを構成する要素

オブザーバビリティは3つの要素から構成されています。

メトリクス (Metrics)

【説明】
監視対象から取得する
データポイントのセット

【構成要素】
CPU使用率
DISK使用率
メモリ使用率
スループット
etc...



ログ (Log)

【説明】
サーバやデータベース、アプリケーション等の利用状況等の記録

【構成要素】
システムログ
エラーログ
アプリケーションログ
データベースログ
etc...



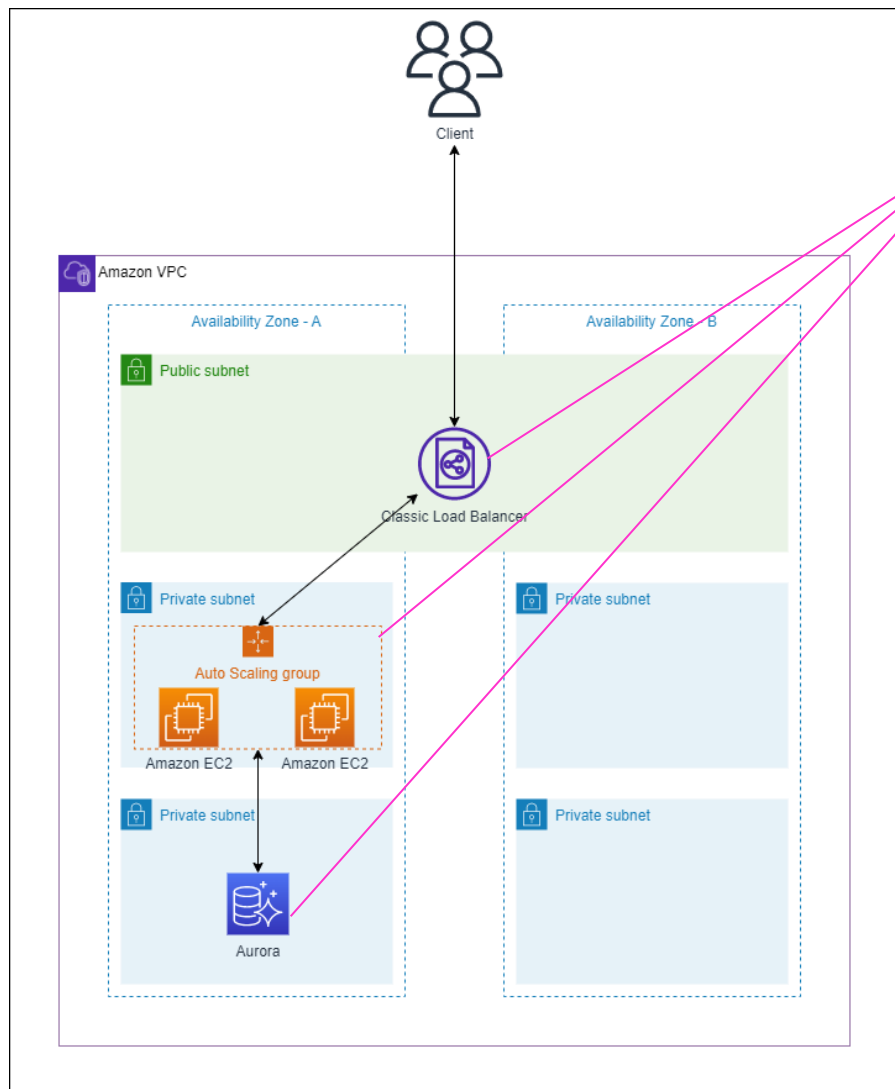
トレース (Trace)

【説明】
トランザクションやリクエストがアプリケーションを流れる際のアクティビティや各コンポーネント間の接続

【構成要素】
コンテキスト
メタデータ
トポロジーマップ
etc...



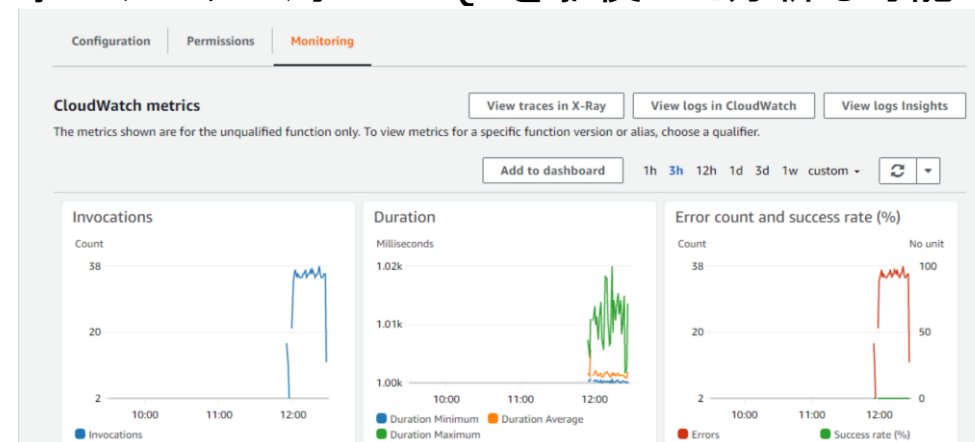
オブザーバビリティ（メトリクス）



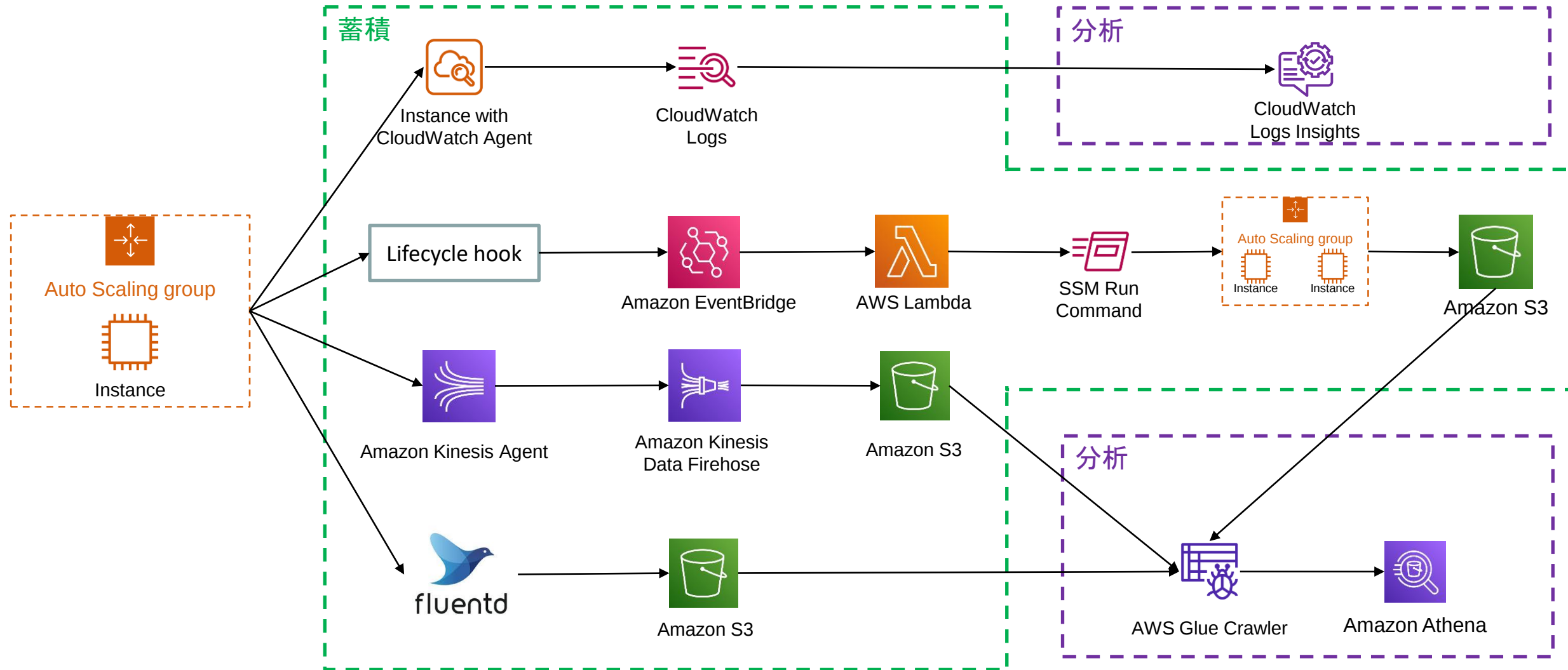
Amazon CloudWatch Metrics

【説明】

システムパフォーマンスに関するデータを取得可能
 詳細モニタリングの有効によってより短いデータポイントの取得も可能
 カスタムでメトリクスを発行する事もできる
 Metrics MathやMetrics Insightsと組み合わせる
 事でデータに対してSQLを駆使した分析も可能



オブザーバビリティ(ログ/AutoScaling)



オブザーバビリティ (ログ/Aurora)



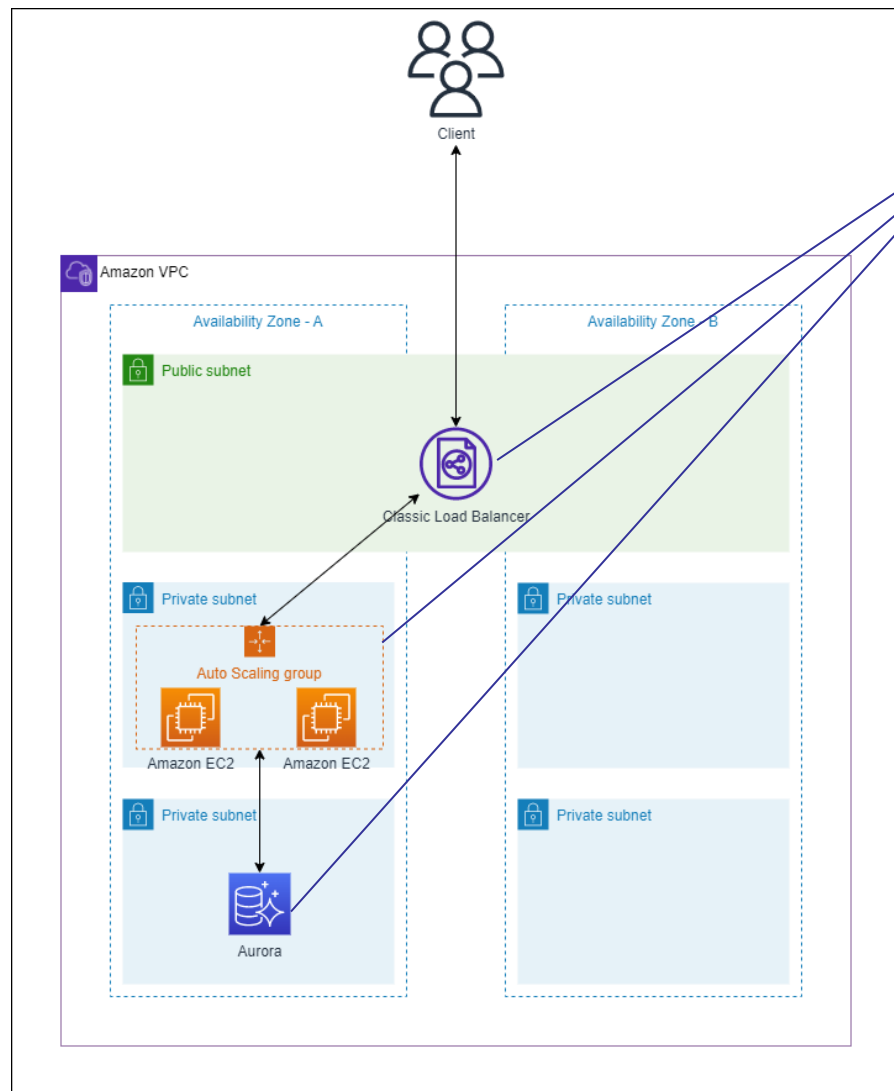
Amazon Aurora

ログの種類	確認できる内容	障害調査において必須か	その他
監査ログ	いつ誰がどこで何を実施したか	▲	有効化するとDBのパフォーマンスに影響がある
エラーログ	DBで発生したエラー	○	デフォルトで生成される
全般ログ	DBのプロセスが吐くログ	×	クライアントから受信したSQL全般が記録される
スロークエリーログ	実行に数秒以上かかったクエリー	▲	パフォーマンスチューニングの観点では使用した方がいい
拡張モニタリング	DBインスタンスのOSレイヤーの情報	▲	CPU,メモリ,ファイルシステム,ディスクI/O etc..
DevOps Guru	MLを駆使しRDSのパフォーマンスや運用上の問題を検出し推奨事項を提案	×	利用にあたりPerformance Insightsの有効化が必要
Performance Insights	既存のDBのモニタリング機能を拡張して、データベースのパフォーマンスを明確にする	○	一部エンジンではサポートしていないインスタタイプがある

※Aurora MySQLについて記載しています

※赤字は障害発生時に役立つログになります

オブザーバビリティ（トレース）

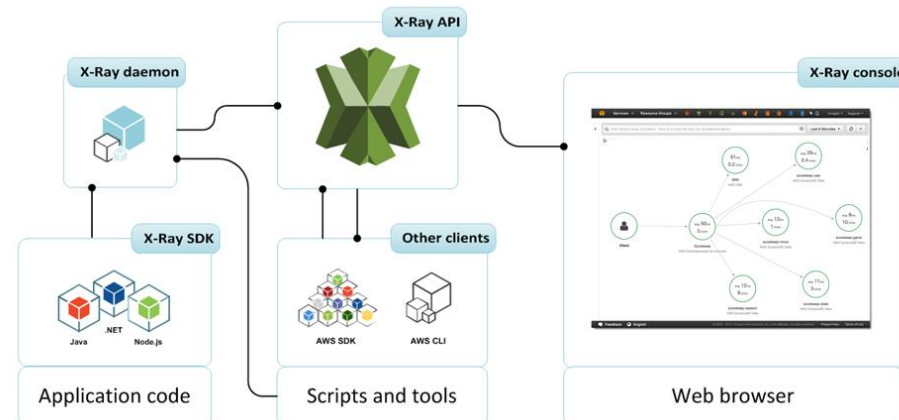


AWS X-Ray

【説明】

アプリケーションやサーバに組み込む事でパフォーマンスの
トラッキングが可能

アプリケーションの実行環境にはデーモンをインストールする
アプリケーションはSDKをソースコードに入れる事で利用可能



https://docs.aws.amazon.com/ja_jp/xray/latest/devguide/aws-xray.html



まとめ



今回このセッションでお伝えしたかったこと

障害は発生してからが肝心

- あとで何が原因でどんな障害が発生したかをトレースする為にも障害ポイントと原因を突き止める仕組みを作っておく事が大事

最低限ログは取得しておこう

- ログがないと何も調査できない状況が発生します

まずはリリースノートを確認しよう

- 今回の障害の原因がDBのバージョンに依存するバグに起因している事が発覚するのに約2週間サポートとやり取りしました



FUJISOFT

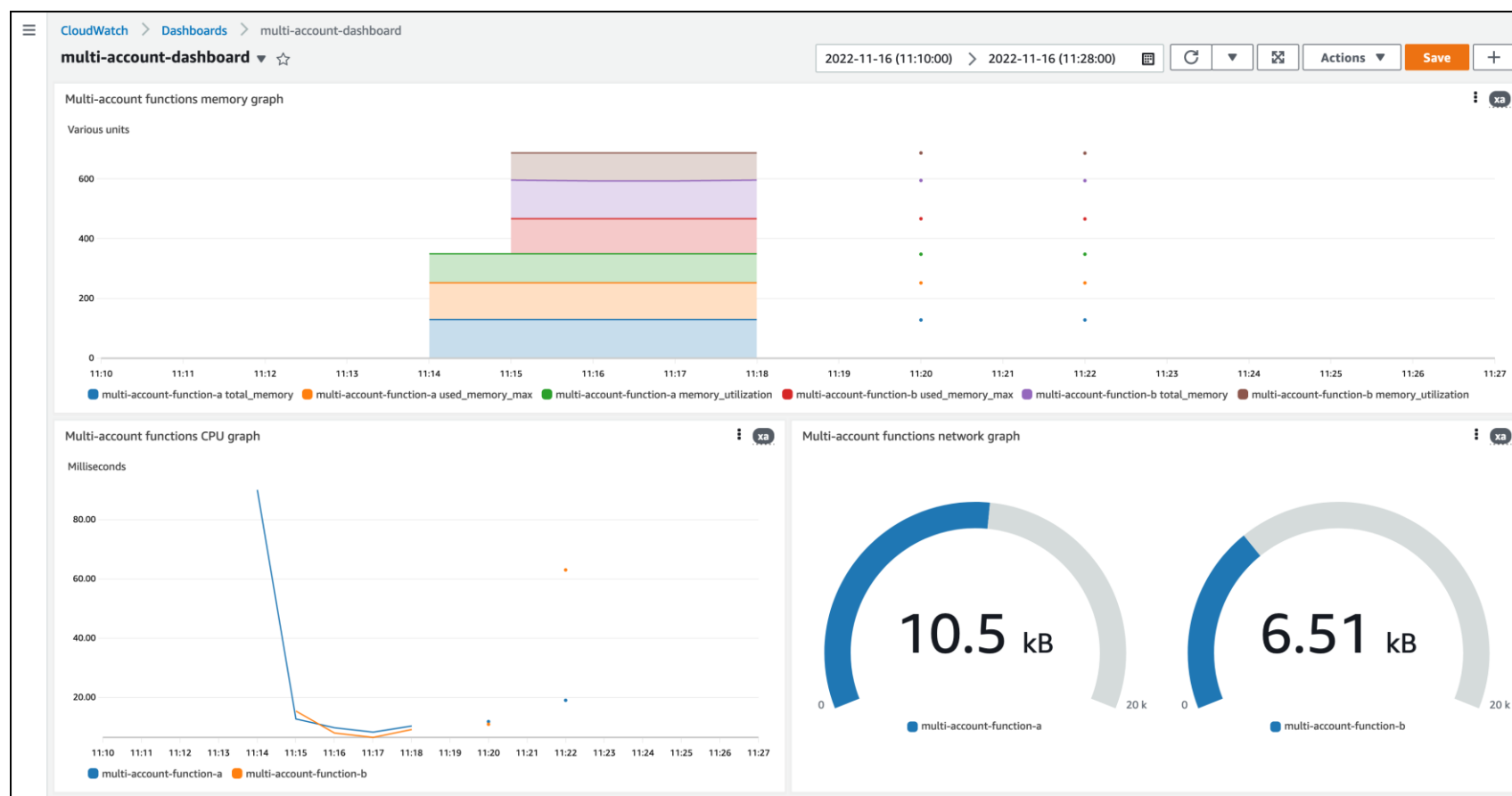
ご清聴ありがとうございました。



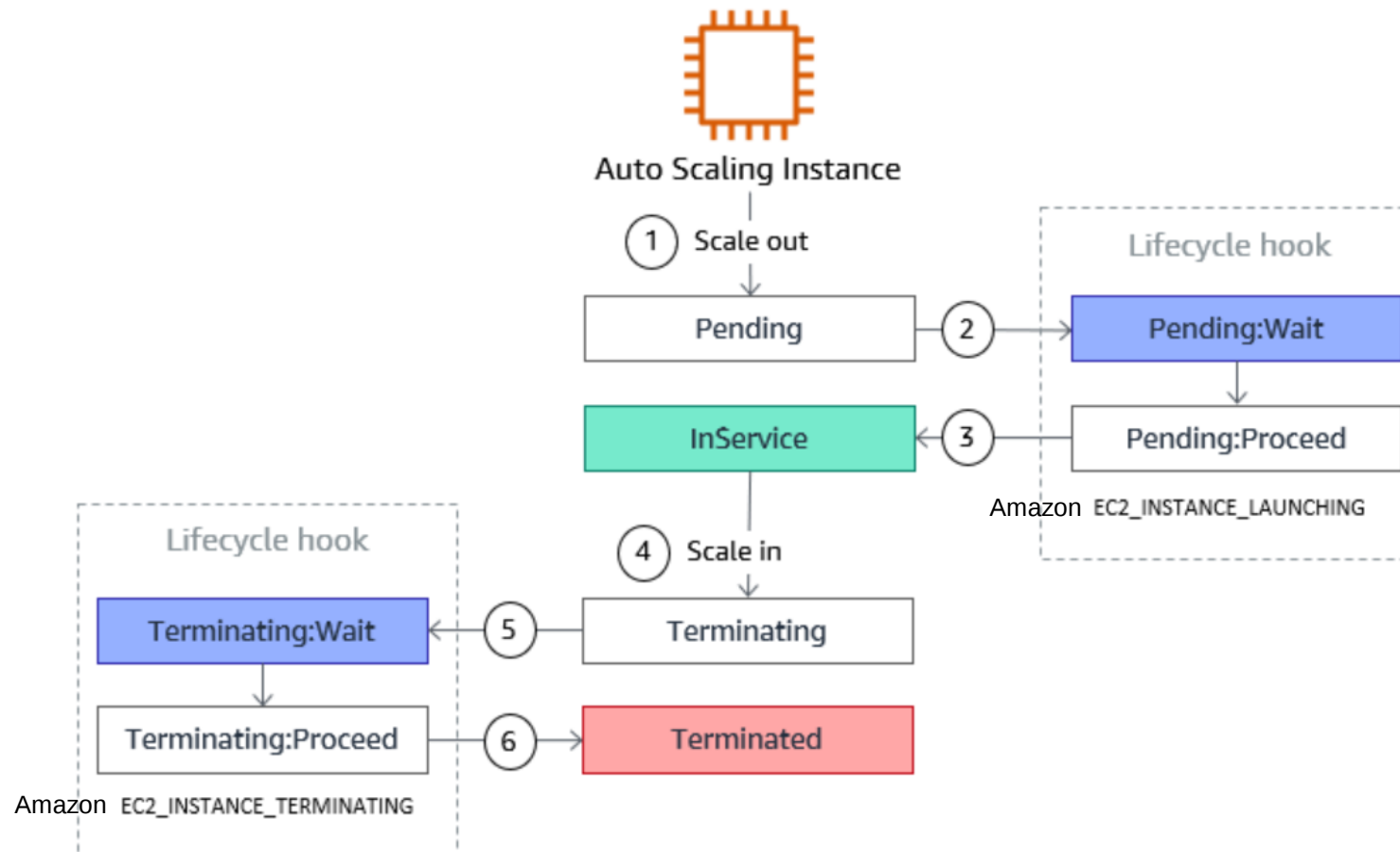
Appendix

CloudWatchクロスアカウントオブザーバビリティ

re:Invent 2022で発表された機能でマルチアカウントのCloudWatchのデータを共有して監視できるようになりました。



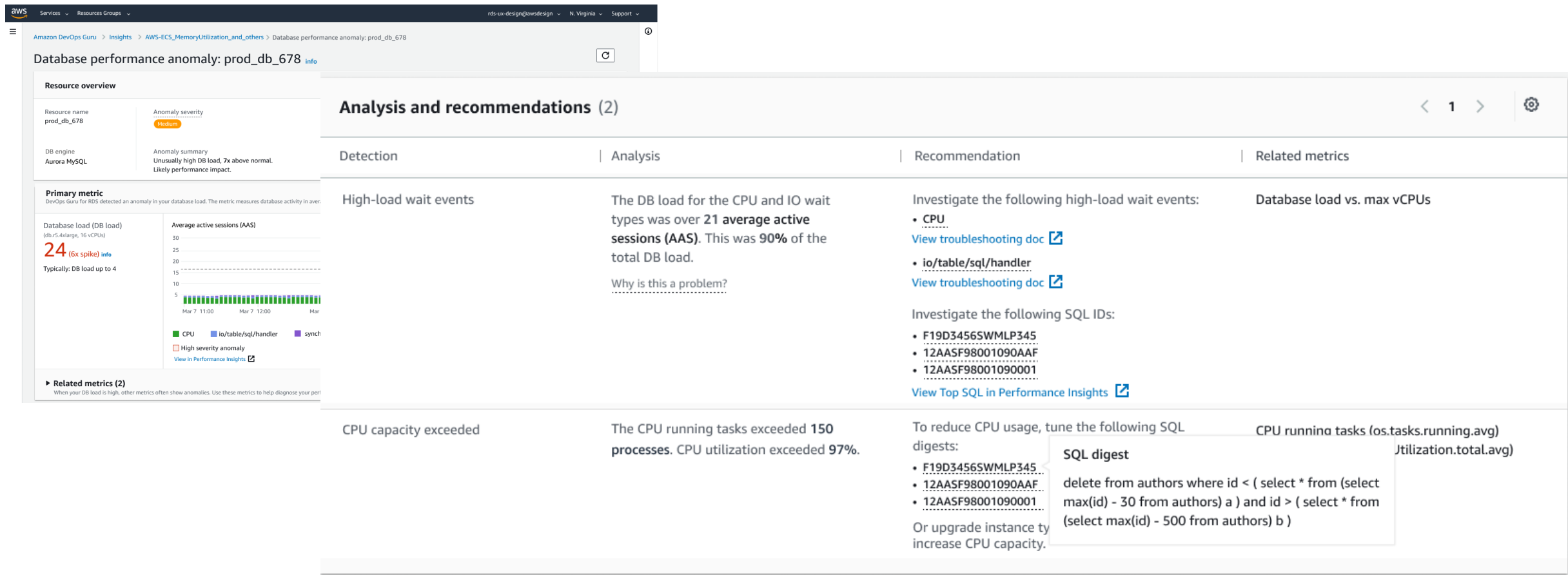
Auto Scaling Lifecycle hook



https://docs.aws.amazon.com/ja_jp/autoscaling/ec2/userguide/lifecycle-hooks-overview.html

Amazon DevOps Guru for Amazon RDS

Amazon RDS データベースパフォーマンスのボトルネックとオペレーション問題を自動検出、診断してくれるサービスです



AWSにおけるDR戦略

