

1) ある企業には、個々のビジネスグループが所有する多数の AWS アカウントがあります。そのアカウントのうちの 1 つが最近侵害されました。攻撃者が多数のインスタンスを起動したため、そのアカウントの請求額が高額になりました。

同社はセキュリティ侵害に対処しましたが、ソリューションアーキテクトは、すべてのアカウントで過剰な支出を防ぐソリューションを開発しなければなりません。各ビジネスグループは、各自の AWS アカウントで完全なコントロールを維持したいと考えています。

これらの要件を満たすために、ソリューションアーキテクトが推奨すべきソリューションはどれですか。

- A) AWS Organizations を使用する。各 AWS アカウントを管理アカウントに追加する。ec2:InstanceType 条件キーを使用する SCP を作成して、各アカウントで高コストのインスタンスタイプが起動されないようにする。
- B) カスタマー管理型の新しい IAM ポリシーを、各アカウントの IAM グループに添付する。ec2:InstanceType 条件キーを使用するようにポリシーを設定し、高コストのインスタンスタイプが起動されないようにする。既存の IAM ユーザーをすべて各グループに配置する。
- C) AWS アカウントごとに請求アラートをオンにする。アカウントが指定された支出しきい値を超えると必ず Amazon Simple Notification Service (Amazon SNS) 通知をアカウント管理者に送信する Amazon CloudWatch アラームを作成する。
- D) 各アカウントで AWS Cost Explorer をオンにする。各アカウントの Cost Explorer レポートを定期的に確認して、支出が希望額を超えていないことを確認する。

2) ある企業は、AWS Organizations 内の組織に複数の AWS アカウントを持っています。同社はオンプレミスの Active Directory と AWS Single Sign-On (AWS SSO) を統合し、すべてのアカウントのインフラストラクチャを管理するための最小権限のアクセス許可を Active Directory ユーザーに付与しています。

ソリューションアーキテクトは、すべての AWS アカウントでの読み取り専用アクセスを必要とする、サードパーティーのモニタリングソリューションを統合しなければなりません。モニタリングソリューションは独自の AWS アカウントで実行されます。

モニタリングソリューションに必要なアクセス許可を与えるために、ソリューションアーキテクトは何をすべきですか。

- A) AWS SSO ディレクトリにユーザーを作成する。読み取り専用アクセス許可セットをユーザーに割り当てる。モニタリングが必要なすべての AWS アカウントをユーザーに割り当てる。サードパーティーのモニタリングソリューションに、ユーザー名とパスワードを指定する。
- B) 組織の管理アカウントに IAM ロールを作成する。サードパーティーのモニタリングソリューションの AWS アカウントにロールを引き受けることを許可する。
- C) サードパーティーのモニタリングソリューションの AWS アカウントを組織に招待する。すべての機能を有効にする。

- D) サードパーティーのモニタリングソリューションの新しい IAM ロールを定義する AWS CloudFormation テンプレートを作成する。信頼ポリシーで、サードパーティーのモニタリングソリューションの AWS アカウントを指定する。スタックセットを使用して、リンクされたすべての AWS アカウントで IAM ロールを作成する。

3) あるチームが、パブリック Amazon S3 バケットでホストされる HTML フォームを作成しています。このフォームでは、JavaScript を使用して Amazon API Gateway API エンドポイントにデータをポストします。API エンドポイントは AWS Lambda 関数と統合されています。チームは API Gateway コンソールで各メソッドをテストし、有効な応答を受け取りました。

フォームが API エンドポイントに正常にポストされ、有効な応答を受け取れるように、チームはどのステップの組み合わせを完了する必要がありますか。 (2 つ選択)

- A) クロスオリジンリソース共有 (CORS) を許可するように S3 バケットを設定する。
- B) Amazon S3 ではなく Amazon EC2 でフォームをホストする。
- C) API Gateway のクォータ引き上げをリクエストする。
- D) API Gateway でクロスオリジンリソース共有 (CORS) を有効にする。
- E) S3 バケットをウェブホスティング用に設定する。

4) ある会社が、Amazon API Gateway、AWS Lambda 関数、Amazon Cognito、Amazon DynamoDB を使用するサーバーレスモバイルアプリケーションを実行しています。トラフィックが急増すると、ユーザーから、断続的にシステム障害が発生しているとの報告があります。API Gateway API エンドポイントが、有効なリクエストに対して HTTP ステータスコード 502 (Bad Gateway) エラーを返しています。

この問題を解決するソリューションはどれですか。

- A) Lambda 関数の同時実行クォータを増やす。ConcurrentExecutions メトリクスがクォータに近づいたときに通知アラートを送信するように Amazon CloudWatch を設定する。
- B) API Gateway API エンドポイントの 1 秒あたりのトランザクションクォータに関する通知アラートを設定する。クォータに達したときにクォータを増やす Lambda 関数を作成する。
- C) 複数の AWS リージョンの Amazon Cognito ユーザープールにユーザーをシャーディングして、ユーザー認証のレイテンシーを低減する。
- D) DynamoDB の強力な整合性のある読み込みを使用して、クライアントアプリケーションが常に最新のデータを受信できるようにする。

5) ある企業が Amazon Elastic Container Service (Amazon ECS) クラスターで新しいウェブサービスを立ち上げています。クラスターは 100 個の Amazon EC2 インスタンスで構成されています。会社のポリシーでは、クラスターインスタンスのセキュリティグループが HTTPS (ポート 443) を除くすべてのインバウンドトラフィックをブロックすることを義務付けています。

これらの要件を満たすソリューションはどれですか。

- A) ユーザーデータスクリプトを使用して、クラスターインスタンスの SSH ポートを 2222 に変更する。ポート 2222 経由で SSH を使用して各インスタンスにログインする。
- B) ユーザーデータスクリプトを使用して、クラスターインスタンスの SSH ポートを 2222 に変更する。AWS Trusted Advisor を使用して、ポート 2222 経由でクラスターインスタンスをリモート管理する。
- C) SSH キーペアなしでクラスターインスタンスを起動する。AWS Systems Manager Run Command を使用して、クラスターインスタンスをリモート管理する。
- D) SSH キーペアなしでクラスターインスタンスを起動する。AWS Trusted Advisor を使用して、クラスターインスタンスをリモートで管理する。

6) ある会社には 2 つの AWS アカウントがあります。1 つのアカウントは本番環境のワークロード用、もう 1 つのアカウントは開発ワークロード用です。開発チームと運用チームが、これらのワークロードを作成して管理しています。同社は、次の要件を満たすセキュリティ戦略を必要としています。

- ・ 開発チームのメンバーは、開発アプリケーションインフラストラクチャを作成および削除する必要がある。
- ・ 運用チームのメンバーは、開発および本番環境のアプリケーションインフラストラクチャを作成および削除する必要がある。
- ・ 開発チームのメンバーは本番インフラストラクチャにアクセスできなくてはならない。
- ・ すべてのユーザーが 1 セットの AWS 認証情報を持っている必要がある。

これらの要件を満たす戦略はどれですか。

- A) 本番稼働用アカウントで以下を実行します。
 - ・ アプリケーションインフラストラクチャを作成および削除できる運用 IAM グループを作成する。
 - ・ 運用チームのメンバーごとに IAM ユーザーを作成する。これらのユーザーを運用グループに割り当てる。開発アカウントで以下を実行します。
 - ・ アプリケーションインフラストラクチャを作成および削除できる開発 IAM グループを作成する。
 - ・ 運用チームのメンバーと開発チームのメンバーごとに IAM ユーザーを作成する。これらのユーザーを開発グループに割り当てる。
- B) 本番稼働用アカウントで以下を実行します。
 - ・ アプリケーションインフラストラクチャを作成および削除できる 運用 IAM グループを作成する。開発アカウントで以下を実行します。
 - ・ アプリケーションインフラストラクチャを作成および削除できる開発 IAM グループを作成する。
 - ・ 開発チームのメンバーごとに IAM ユーザーを作成する。これらのユーザーを開発グループに割り当てる。

- 運用チームのメンバーごとに IAM ユーザーを作成する。これらのユーザーを開発グループと本番稼働用アカウントの運用グループに割り当てる。
- C) 開発アカウントで以下を実行します。
- 本番稼働用アカウントでアプリケーションインフラストラクチャを作成および削除できる共有 IAM ロールを作成する。
 - アプリケーションインフラストラクチャを作成および削除できる開発 IAM グループを作成する。
 - 共有ロールを引き受けることができる運用 IAM グループを作成する。
 - 開発チームのメンバーごとに IAM ユーザーを作成する。これらのユーザーを開発グループに割り当てる。
 - 運用チームのメンバーごとに IAM ユーザーを作成する。これらのユーザーを開発グループと運用グループに割り当てる。
- D) 本番稼働用アカウントで以下を実行します。
- アプリケーションインフラストラクチャを作成および削除できる共有 IAM ロールを作成する。
 - 開発用アカウントを共有ロールの信頼ポリシーに追加する。
- 開発アカウントで以下を実行します。
- アプリケーションインフラストラクチャを作成および削除できる開発 IAM グループを作成する。
 - 本番稼働用アカウントで共有ロールを引き受けることができる運用 IAM グループを作成する。
 - 開発チームのメンバーごとに IAM ユーザーを作成する。これらのユーザーを開発グループに割り当てる。
 - 運用チームのメンバーごとに IAM ユーザーを作成する。これらのユーザーを開発グループと運用グループに割り当てる。

7) あるソリューションアーキテクトが、ビッグデータアプリケーションのコストを削減しなければならなくなりました。アプリケーション環境は、Amazon Kinesis Data Streams にイベントを送信する数百ものデバイスで構成されています。デバイス ID はパーティションキーとして使用されるため、各デバイスは個別のシャードを取得します。各デバイスは、毎秒 50 KB から 450 KB のデータを送信します。AWS Lambda 関数はシャードをポーリングしてデータを処理し、その結果を Amazon S3 に保存します。

別の Lambda 関数が 1 時間ごとに結果データに対して Amazon Athena クエリを実行し、外れ値を特定します。この Lambda 関数は、外れ値を Amazon Simple Queue Service (Amazon SQS) キューに配置します。2 つの EC2 インスタンスで構成される Amazon EC2 Auto Scaling グループは、キューをモニタリングし、外れ値に対処するために 30 秒のプロセスを実行します。デバイスは 1 時間ごとに平均 10 個の外れ値を送信します。

アプリケーションに対する変更の組み合わせで、最もコストが削減されるのはどれですか。(2 つ選択)

- A) Auto Scaling グループの起動設定を変更して、同じインスタンスファミリーでより小さいインスタンスタイプを使用するようにする。

- B) Auto Scaling グループを、メッセージがキューに到着したときに呼び出される Lambda 関数に置き換える。
- C) デバイスとデータストリームを再構成して、1 つのデータストリームシャードに対して 10 台のデバイスの比率を設定する。
- D) デバイスとデータストリームを再構成して、1 つのデータストリームシャードに対して 2 台のデバイスの比率を設定する。
- E) Auto Scaling グループのターゲット容量を EC2 インスタンス 1 つに変更する。

8) ある企業は、Application Load Balancer の背後で Amazon EC2 インスタンス上に e コ머스アプリケーションを運用しています。インスタンスは、複数のアベイラビリティゾーンにまたがる Amazon EC2 Auto Scaling グループ内で実行されます。注文が正常に処理されると、アプリケーションは注文データをサードパーティーのアフィリエイトの外部追跡システムに即座にポストし、そのシステムから注文の紹介に対して売上手数料が支払われます。

マーケティングプロモーションが成功すると、EC2 インスタンスの数が 2 から 20 に増加しました。この間、アプリケーションは正常に動作し続けました。しかし、リクエスト率の上昇でサードパーティーのアフィリエイトが過負荷になり、リクエストが失敗する結果となりました。

負荷がかかってもプロセス全体が正しく機能するように、ソリューションアーキテクトはどのアーキテクチャの変更を組み合わせるべきですか。(2 つ選択)

- A) アフィリエイトを呼び出すコードを新しい AWS Lambda 関数に移動する。Lambda 関数を非同期で呼び出すようにアプリケーションを変更する。
- B) アフィリエイトを呼び出すコードを新しい AWS Lambda 関数に移動する。注文データを Amazon Simple Queue Service (Amazon SQS) キューに配置するようにアプリケーションを変更する。キューから Lambda 関数を呼び出す。
- C) 新しい AWS Lambda 関数のタイムアウトを増やす。
- D) 新しい AWS Lambda 関数の予約済み同時実行数を減らす。
- E) 新しい AWS Lambda 関数のメモリを増やす。

9) ある企業が AWS でオンラインチケット発行ウェブアプリケーションを構築しました。このアプリケーションは AWS App Runner でホストされ、Amazon Elastic Container Registry (Amazon ECR) リポジトリに保存されているイメージを使用します。アプリケーションは Amazon Aurora MySQL DB クラスターにデータを格納します。同社は Amazon Route 53 にドメイン名を設定しています。

アプリケーションは、Active-Active 構成の 2 つの AWS リージョンにデプロイする必要があります。

アーキテクチャの変更を最小限に抑えながら、これらの要件を満たすステップの組み合わせはどれですか。(3 つ選択)

- A) ECR イメージの 2 つ目のリージョンへのクロスリージョンレプリケーションを設定する。

- B) 2 つ目のリージョンの ECR リポジトリから VPC エンドポイントを作成する。
- C) 2 つ目のデプロイターゲットを 2 つ目のリージョンに追加して、App Runner 設定を編集する。
- D) App Runner を 2 番目のリージョンにデプロイする。Route 53 レイテンシールーティングポリシーを設定する。
- E) 目的の 2 つのリージョンで Amazon DynamoDB グローバルテーブルを使用して、データベースを変更する。
- F) 2 つ目のリージョンで書き込み転送が有効になっている Aurora グローバルデータベースを使用する。

10) ある企業が AWS クラウドに多層ウェブアプリケーションをデプロイしました。このアプリケーションは、次の階層で構成されています。

- Elastic IP アドレスを持つ Amazon EC2 インスタンスでホストされている Windows ベースのウェブ層
- パスベースのルーティングを使用する Application Load Balancer (ALB) の背後で実行される EC2 インスタンスでホストされる Linux ベースのアプリケーション層
- Linux EC2 インスタンスで実行される MySQL データベース

すべての EC2 インスタンスは Intel ベースの x86 CPU を使用しています。ソリューションアーキテクトは、パフォーマンスを向上させるためにインフラストラクチャをモダナイズする必要があります。このソリューションでは、アプリケーションの運用上の必要コストを最小限に抑える必要があります。

これらの要件を満たすために、ソリューションアーキテクトが取るべきアクションの組み合わせはどれですか。
(2 つ選択)

- A) MySQL データベースを複数の EC2 インスタンスで実行する。
- B) ウェブ層インスタンスを ALB の背後に配置する。
- C) MySQL データベースを Amazon Aurora Serverless に移行する。
- D) すべての EC2 インスタンスタイプを Graviton2 に移行する。
- E) アプリケーション層インスタンスの ALB を企業が管理するロードバランサーに置き換える。

解答

- 1) C – [請求アラーム](#)は、どのビジネスグループからもコントロールを奪うことなく、過剰な支出に関するアラートを提供します。各ビジネスグループが各自のアカウントのコントロールを維持したいと考えているため、オプション A と B は正しくありません。これらのオプションによって、大量のインスタンスの起動が妨げられることはありません。オプション D は手動のプロセスで、過剰な支出に関するアラートはすぐには提供されません。
- 2) D – [AWS CloudFormation StackSets](#) は、1 回のオペレーションで複数のアカウントに IAM ロールをデプロイできます。AWS Single Sign-On (AWS SSO) によって提供される認証情報は一時的なものであるため、オプション A は正しくありません。アプリケーションはアクセス許可を失い、再度ログインが必要になります。オプション B では、管理アカウントにのみアクセス権が付与されます。オプション C は正しくありません。これは、あるアカウントが組織に加入すると、そのアカウントには組織内の他のアカウントへのアクセス許可が付与されないためです。
- 3) D、E – [クロスオリジンリソース共有 \(CORS\)](#) は、ブラウザで実行されるスクリプトから開始される HTTP リクエストを制限するブラウザセキュリティ機能です。CORS は通常、別のドメインまたはオリジンでホストされている API にアクセスするウェブアプリケーションを構築するために必要です。CORS を有効にして、別のドメインでホストされているウェブアプリケーションからの API へのリクエストを許可できます。例えば、API が `https://[api_id].execute-api.[region].amazonaws.com/` でホストされており、`[bucketname].s3.website-[region]` でホストされているウェブアプリケーションから API を呼び出すには、API が CORS をサポートしている必要があります。[ウェブサイトエンドポイント](#)を介して HTML フォームを提供するには、オプション E が必要です。
- API エンドポイントからの動的応答によって返されるように CORS ヘッダーを設定する必要があるため、オプション A は正しくありません。S3 バケットの CORS の設定は役に立ちません。S3 バケットからではなく Amazon EC2 で実行されるウェブサーバーから静的ウェブページを配信する利点がないため、オプション B は正しくありません。API Gateway には、[AWS リージョンごとに 1 秒あたり 10,000 リクエストというデフォルトのクォータ](#)があるため、オプション C は正しくありません。このクォータは、必要に応じて増やすことができます。
- 4) A – Amazon API Gateway は、AWS Lambda 関数が同時実行クォータを超えると、[HTTP ステータスコード 502 \(Bad Gateway\) エラー](#)を断続的に返します。この場合、API Gateway [はリクエストが多すぎるとステータスコード 429 エラーを返す](#)ため、オプション B は正しくありません。エラーは認証プロセス中ではなく、API Gateway API エンドポイントの呼び出し中に発生するため、オプション C は正しくありません。古いデータによって Bad Gateway エラーは発生しないため、オプション D は正しくありません。
- 5) C – [AWS Systems Manager Run Command](#) では、インバウンドポートを開く必要はありません。Run Command は、セキュリティグループに対してデフォルトで開かれているアウトバウンド HTTPS 上で完全に動作します。オプション A と B は正しくありません。開いておくべき受信ポートは 443 だけで

あるという要件があるためです。AWS Trusted Advisor にこの管理機能はないため、オプション D は正しくありません。

6) D – 正解は、管理する 2 つのアカウント間でクロスアカウントアクセスを許可するための[標準的なガイドライン](#)に従っています。オプション A では、運用チームのメンバーに 2 セットの認証情報が必要なため、要件を満たしていません。IAM ユーザーを別のアカウントの IAM グループに追加できないため、オプション B は正しくありません。ロールは別のアカウントのリソースへのアクセス権を付与できないため、オプション C は正しくありません。共有ロールは、その共有ロールが管理するリソースと同じアカウントに属している必要があります。

7) B、D – 外れ値を処理するコンピューティングの平均量は 1 時間あたり 300 秒 (30 秒ごとに 10 個のイベント) です。[AWS Lambda](#) では、外れ値の処理に必要なわずかなコンピューティング時間に対してのみ支払いが発生するため、オプション B が正解です。オプション A と E はコストを削減しますが、いずれの場合も 1 時間あたり 3,300 秒間使用されない Amazon EC2 インスタンス 1 つ以上の料金を支払う必要があります。オプション C と D は、Kinesis Data Streams のシャード時間のコストを削減します。ただし、データ量が単一のシャードの [1 MB/秒のクォータ](#)を超えるため、オプション C は正しくありません。

8) B、D – オプション B では、[Amazon Simple Queue Service \(Amazon SQS\) キュー](#)を使用すると、メインアプリケーションがアフィリエイトへの呼び出しから切り離されます。この変更により、メインアプリケーションがアフィリエイトの容量低下から保護されます。また、失敗したリクエストは自動的にキューに戻ることができます。オプション D では、同時呼び出しの数を減らすことで、アフィリエイトアプリケーションが過負荷になるのを防ぐことができます。

オプション A は Amazon EC2 インスタンスの負荷を軽減しますが、このソリューションではアフィリエイトアプリケーションへのリクエスト数は減りません。オプション C では、外部呼び出しが返されるまでの AWS Lambda 関数の待機時間が長くなりますが、このソリューションではアフィリエイトアプリケーションの負荷は軽減されません。メモリの増加は Lambda 関数とアフィリエイト追跡システム間の相互作用には影響しないため、オプション E は正しくありません。

9) A、D、F – [AWS App Runner](#) は、Amazon Elastic Container Registry (Amazon ECR) リポジトリに保存されているイメージを使用して、コンテナ化されたウェブアプリケーションを迅速にデプロイするために使用できるフルマネージドサービスです。[クロスリージョンレプリケーション](#)では 2 番目の AWS リージョンにリポジトリのコピーが作成されるため、オプション A は正解です。[Route 53](#) を使用してカスタムドメイン名をホストし、複数の AWS リージョンのリソースにトラフィックをルーティングできるため、オプション D は正解です。[Amazon Aurora Global Database](#) は複数のリージョンにまたがり、グローバルに分散されたアプリケーション向けに設計されているため、オプション F は正解です。

VPC エンドポイントは別のリージョンに保存されているイメージへのアクセスを提供しないため、オプション B は正しくありません。オプション C では、App Runner にそのような設定は存在しません。オプション E は機能しますが、Amazon DynamoDB の導入では、Aurora グローバルデータベースを使用する場合よりも

アーキテクチャに多くの変更が必要になります。この問題では、アーキテクチャの変更を最小限に抑える必要があります。

10) B、C – オプション B では、ウェブ層を [Application Load Balancer \(ALB\)](#) の背後に配置することで、ウェブ層の可用性とスケーラビリティを向上させることができます。ALB はクライアントの単一窓口として機能し、受信するアプリケーショントラフィックを Amazon EC2 インスタンスに分散します。

[Amazon Aurora Serverless](#) は高パフォーマンスと高可用性を実現し、運用の複雑さを軽減するため、オプション C が正解です。

EC2 インスタンスを追加しても運用上の必要コストが最小化されないため、オプション A は正しくありません。マネージドサービスを選択した方が良いでしょう。Graviton2 では使用できない Windows インスタンスがアプリケーションに含まれているため、オプション D は正しくありません。企業が管理するロードバランサーでは運用上のコストが最小化されないため、オプション E は正しくありません。