AWS Well-Architected フレームワーク

コスト最適化の柱

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

コスト最適化の柱: AWS Well-Architected フレームワーク

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon の商標およびトレードドレスはAmazon 以外の製品およびサービスに使用することはできません。また、お客様に誤解を与える可能性がある形式で、または Amazon の信用を損なう形式で使用することもできません。Amazon が所有していないその他のすべての商標は Amazon との提携、関連、支援関係の有無にかかわらず、それら該当する所有者の資産です。

Table of Contents

要約と序章 1
- 要約 1
- 序章 1

コスト最適化 3
- 設計原則 3
- 定義 4

クラウド財務管理を実践する 5
- COST01-BP01 コスト最適化の所有権を設定する 8
  - 実装のガイダンス 8
  - リソース 10
- COST01-BP02 財務とテクノロジーの連携を確立する 11
  - 実装のガイダンス 8
  - リソース 10
- COST01-BP03 クラウドの予算と予測を確立する 16
  - 実装のガイダンス 8
  - リソース 10
- COST01-BP04 組織のプロセスにコスト意識を採り入れる 19
  - 実装のガイダンス 8
  - リソース 10
- COST01-BP05 コスト最適化に関して報告および通知する 21
  - 実装のガイダンス 8
  - リソース 10
- COST01-BP06 コストをプロアクティブにモニタリングする 24
  - 実装のガイダンス 8
  - リソース 10
- COST01-BP07 新しいサービスリリースに関する最新情報を把握しておく 25
  - 実装のガイダンス 8
  - リソース 10
- COST01-BP08 コスト意識を持つ文化を生み出す 27
  - 実装のガイダンス 8
  - リソース 10
- COST01-BP09 コスト最適化によるビジネス価値を数値化する 29
  - 実装のガイダンス 8
  - リソース 10

経費支出と使用量の認識 31
ガバナンス 31
COST02-BP01 組織の要件に基づいてポリシーを策定する 32
COST02-BP02 目標およびターゲットを策定する 36
COST02-BP03 アカウント構造を実装する 40
COST02-BP04 グループとロールを実装する 44
COST02-BP05 コストコントロールを実装する 46
COST02-BP06 プロジェクトのライフサイクルを追跡する 48
コストと使用量のモニタリング 51
COST03-BP01 詳細情報ソースを設定する 51
COST03-BP02 コストと使用状況に組織情報を追加する 54
COST03-BP03 コスト属性カテゴリを特定する 56
COST03-BP04 組織のメトリクスを確立する 59
COST03-BP05 請求およびコスト管理ツールを設定する 60
COST03-BP06 ワークロードメトリクスに基づいてコストを配分する 63
リソースを削除する 65
COST04-BP01 ライフタイム全体にわたってリソースを追跡する 65
COST04-BP02 廃止プロセスを実装する 67
COST04-BP03 リソースを廃止する 70
COST04-BP04 自動的にリソースを廃止する 70
COST04-BP05 データ保持ポリシーを適用する 72
コスト効率を考慮しながらリソースを利用する 74
サービスを選択する際にコストを評価する 74
COST05-BP01 組織のコスト要件を特定する 74
COST05-BP02 ワークロードのすべてのコンポーネントを分析する 76
COST05-BP03 各コンポーネントの詳細な分析を実行する 79
COST05-BP04 コスト効率の高いライセンスを提供するソフトウェアを選択する 81
COST05-BP05 組織の優先順位に従ってコストが最適化されるようにこのワークロードのコンポーネントを選択する 83
COST05-BP06 異なる使用量について経時的なコスト分析を実行する 85
正しいリソースタイプ、リソースサイズ、リソース数を選択する 87
COST06-BP01 コストモデリングを実行する 87
COST06-BP02 データに基づいてリソースタイプ、リソースサイズ、リソース数を選択する 89
COST06-BP03 メトリクスに基づいて自動的にリソースタイプ、リソースサイズ、リソース数を選択する 91

COST06-BP04 共有リソースの使用を検討する 94
最適な料金モデルを選択する 97
COST07-BP01 料金モデルの分析を実行する 102
COST07-BP02 コストに基づいてリージョンを選択する 105
COST07-BP03 費用対効果の高い条件を提供するサードパーティーの契約を選択する 108
COST07-BP04 このワークロードのすべてのコンポーネントに対して料金モデルを実装する 109
COST07-BP05 管理アカウントレベルで料金モデル分析を実行する 111
データ転送を計画する 114
COST08-BP01 データ転送モデリングを実行する 114
COST08-BP02 データ転送コストを最適化するコンポーネントを選択する 116
COST08-BP03 データ転送コストを削減するサービスを実装する 118
需要を管理しリソースを供給する 121
COST09-BP01 ワークロードの需要に関する分析を実行する 121
実装のガイダンス 8
リソース 10
COST09-BP02 需要を管理するためのバッファまたはスロットルを実装する 124
実装のガイダンス 8
リソース 10
COST09-BP03 リソースを動的に供給する 127
実装のガイダンス 8
実装手順 9
リソース 10
継続的最適化 134
レビュープロセスを定義し、ワークロードを定期的に分析 134
COST10-BP01 ワークロードレビュープロセスを開発する 134
COST10-BP02 このワークロードを定期的に見直し、分析する 136
オペレーションの自動化 138
COST11-BP01 運用のオートメーションを実行する 138
結論 143
寄稿者 144
詳細情報 145
ドキュメントの改訂 146
注意 148
AWS 用語集 149

コスト最適化の柱 - AWS Well-Architected フレームワーク

公開日: 2024 年 6 月 27 日 (ドキュメントの改訂)

要約

このホワイトペーパーでは、Amazon Web Services (AWS) Well-Architected フレームワークのコスト最適化の柱に焦点を当てます。本書は、お客様が AWS 環境の設計、提供、メンテナンスを行う際にベストプラクティスを適用できるようにするためのガイダンスを提供します。

コストが最適化されたワークロードでは、すべてのリソースがフル活用され、コストの最小化を達成して、機能要件が満たされます。このホワイトペーパーは、組織内での機能の構築、ワークロードの設計、サービスの選択、サービスの構成と運用、およびコスト最適化手法の適用に関する詳細なガイダンスを提供します。

序章

AWS Well-Architected フレームワークは、AWS でワークロードを構築する際の決定事項を理解するのに役立ちます。このフレームワークは、信頼性が高く、安全かつ効率的で、コスト効率に優れ、持続可能なワークロードをクラウド内で設計および運用するためのアーキテクチャ設計のベストプラクティスを提供します。アーキテクチャをベストプラクティスに照らして評価し、改善すべき領域を特定する一貫した方法を紹介します。私たちは、well-architected ワークロードを設計することで、ビジネスの成功の可能性が大幅に高まると考えています。

このフレームワークは次の 6 つの柱に基づいています。

- 運用上の優秀性
- セキュリティ
- 信頼性
- パフォーマンス効率
- コスト最適化
- 持続可能性

本ホワイトペーパーでは、コスト最適化の柱と、サービスやリソースを最も効果的に活用したワークロードの設計方法、最小限のコストでビジネス成果を達成する方法に焦点を当てています。

コスト最適化の柱のベストプラクティスを組織に適用する方法について学習します。従来のオンプレミスソリューションでは、コストを最適化するためのハードルにぶつかる可能性があります。これは将来のキャパシティやビジネスニーズを予測すると同時に、複雑な調達プロセスを進める必要があるためです。このホワイトペーパーのプラクティスを適用すると、皆さんの組織が以下の目標達成を支援します。

- クラウド財務管理を実践する
- 経費支出と使用量の認識
- コスト効率を考慮しながらリソースを利用する
- 需要を管理しリソースを供給する
- 継続的最適化

本ホワイトペーパーは、最高技術責任者 (CTO)、最高財務責任者 (CFO)、設計者、デベロッパー、財務管理者、ファイナンシャルプランナー、ビジネスアナリスト、オペレーションチームメンバーなどの技術担当者および財務担当者を対象としています。本ホワイトペーパーでは、運用の詳細やアーキテクチャのパターンについては説明しておりませんが、該当するリソースへの参照先が記載されています。

コスト最適化

コスト最適化とは、システムのワークロードのライフサイクル全体にわたって改良、改善する継続的プロセスです。本ホワイトペーパーのプラクティスは、組織がコストを最小限に抑えて投資利益率を最大化すると同時に、ビジネス成果の達成につながるようなコストを意識したワークロードの構築および運用を支援します。

トピック

- 設計原則
- 定義

設計原則

コスト最適化のため、以下の設計原則を検討する:

クラウド財務管理の実装: クラウドで財務上の成功を達成し、ビジネス価値の実現を加速するには、クラウド財務管理に投資する必要があります。組織は、テクノロジーと使用量管理の新たなドメインで機能を構築するために必要な時間とリソースを投入する必要があります。コスト効率の高い組織になるためには、セキュリティまたはオペレーションの能力と同様、知識の積み上げ、プログラム、リソース、プロセスを通じて能力を構築する必要があります。

消費モデルを導入する: コンピューティングリソースの使用分のみを支払い、ビジネス要件に応じて使用量を増減することができます。例えば、通常、1 週間の稼働日に開発環境とテスト環境を使用するのは、1 日あたり 8 時間程度です。未使用時にこのようなリソースを停止することで、コストを 75% 削減できる可能性があります (168 時間から 40 時間に減少)。

全体的な効率を測定する: ワークロードのビジネス成果とその実現に関連するコストを測定します。このデータを利用すると、生産性および機能性の向上とコスト削減から得られるメリットを理解することができます。

差別化につながらない手間のかかる作業にコストをかけるのをやめる: サーバーのラッキング、積み上げ、電力供給などのデータセンターの手間のかかる運用作業は AWS が行います。また、マネージドサービスを使用することで、オペレーティングシステムやアプリケーションの管理に伴う運用上の負担も解消されます。この結果、IT インフラストラクチャよりも顧客やビジネスプロジェクトに集中できるようになります。

コストを分析し帰属関係を明らかにする: クラウドでは、ワークロードのコストと使用状況を正確に確認しやすくなり、IT コストを収益システムと個々のワークロード所有者に透過的に結び付けるこ

とができます。これによって投資収益率 (ROI) を把握できるため、ワークロードの所有者はリソースを最適化してコストを削減する機会が得られます。

定義

クラウドでのコスト最適化には 5 つの重点分野があります。

- クラウド財務管理を実践する
- 経費支出と使用量の認識
- 費用対効果の高いリソース
- 需要の管理とリソースの提供
- 継続的最適化

この柱にも、Well-Architected フレームワーク内の他の柱と同様に、コスト最適化のために検討すべきトレードオフがあります。例えば、市場投入に要する期間と、コストのどちらを優先すべきでしょうか。市場投入に要する期間の短縮、新機能導入、納期順守といったケースでは、前払いコストの投資を最適化するよりも、スピードを重視して最適化することが最善なことがあります。

設計上の決定は、データではなく時間的制約によって下されることがあります。また、長期的にコストを最適化できるデプロイのベンチマークの選定に時間をかけるよりも、「万一の場合」の備えを過度に重視してしまう傾向が常にあります。過大な見積もりの結果、オーバープロビジョニングになり、最適化が不十分なデプロイを行ってしまいます。ただし、オンプレミス環境からクラウド環境に「リフトアンドシフト」式にリソースを移行してから最適化する必要がある場合は、選択肢としては妥当なこともあります。

適切な労力を当初からコスト最適化戦略に投入すると、ベストプラクティスが一貫して適用され、不要なオーバープロビジョニングも回避できるため、クラウドの経済的メリットをより早く実感できます。以下のセクションでは、クラウドの財務管理の初期および継続的な実装と、ワークロードのコスト最適化のための手法とベストプラクティスを提供します。

クラウド財務管理を実践する

クラウド財務を管理するには、コストの透明性、制御、計画、AWS 環境の最適化を確立して運用するために、既存の財務プロセスを進化させる必要があります。

従来の静的なウォーターフォール計画と IT 予算およびコスト評価モデルを動的なクラウドの使用に適用すると、リスクが生じ、計画が不正確になり、可視性が低くなります。最終的には、コストを効果的に最適化、管理し、長期的なビジネス価値を実現する機会を失うことにつながります。これらの落とし穴を回避するために、クラウドでネイティブにアプリケーションを構築する場合でも、ワークロードをクラウドに移行する場合でも、クラウドサービスの採用を拡大する場合でも、クラウドジャーニー全体でコストを積極的に管理する必要があります。

クラウド財務管理 (CFM) は、財務、製品、技術、ビジネス組織が、AWS での利用拡大やスケールの拡張に伴い、コストの管理、最適化、計画を実施できるようにします。CFM の主な目的は、お客様が最もコスト効率の高い方法でビジネス成果を達成し、俊敏性とコントロールの適切なバランスを見つけながら、経済的価値とビジネス価値の創造を加速できるようにすることです。

CFM ソリューションは、コスト透明性、コントロール、予測、最適化により、ビジネスを変革するのに役立ちます。また、これらのソリューションは、すべてのチームと機能にわたる説明責任を促す、コスト意識の高い文化を醸成します。財務チームは、コストの発生源を確認し、予期しない費用を最小限に抑えながら運用を実施し、動的なクラウドの使用を計画して、チームがクラウドの導入を拡大しながらクラウド費用を節約することができます。これをエンジニアリングチームと共有することで、リソースの選択、使用、最適化に必要な財務状況を提供できます。

AWS CFM は、ビジネスの俊敏性を維持しながら、クラウドコストを管理、最適化、および計画するための一連の機能を提供します。CFM は、コストを効果的に管理するだけでなく、投資を期待できるビジネス成果を促進しているかを検証するうえで最も重要です。AWS クラウドにおけるクラウド財務管理フレームワークの 4 つの柱とは、観察、節約、計画、実行です。これらの柱にはそれぞれ、アクティビティと機能が設定されています。

クラウド財務管理の 4 つの柱。

- 観察: 現在、クラウド支出の計算、モニタリング、説明責任の作成をどのような方法で行っていますか。AWS を初めて利用する場合や AWS を利用しようと計画している場合、コストと使用状況の可視性を確立する計画がありますか?

  AWS のコストを理解し、支出を最適化するには、それらのコストがどこから来ているのかを知る必要があります。そのためには、アカウントやリソースを計画的に管理し、財務部門が支出の流れを把握し、各チームが利益に対する責任を果たせるようにすることが必要です。

  AWS サービス: AWS Control Tower、AWS Organizations、コスト配分タグ、タグポリシー、AWS Resource Groups、AWS Cost Categories、AWS Cost Explorer、AWS Cost and Usage Report、RI、SP

  リソース: AWS のタグ付けのベストプラクティス、AWS Cost Categories

- 節約: 現在、支出を最適化するためにどのようなコスト最適化の手段を使用していますか。AWS を使用していない場合、一般的な使用ベースおよび価格モデルベースの最適化についてご存知でしょうか?

  節約の基本方針では、価格やリソースの推奨によりコストを最適化します。コストの最適化は、新しいクラウド運用モデルに対して明確に定義された戦略を持つことから始まります。クラウドジャーニーのできるだけ早い段階から始め、適切なプロセスと動作によって強化されたコスト意識の高い文化の準備を整えるのが理想的です。

  クラウドのコストを最適化するには、さまざまな方法があります。その 1 つは、適切な購入モデル (RI および SP) の選択、または Amazon EC2 スポットインスタンスを採用できるようにワー

クロードが不変かつコンテナ化されているかどうかです。さらに、Amazon EC2 Auto Scaling グループを使用してワークロードをスケールします。

AWS サービス: RI と SP、Amazon EC2 Auto Scaling グループ、スポットインスタンス

リソース: リザーブドインスタンス、Savings Plans、Amazon EC2 の操作に関するベストプラクティス

- 計画: 将来のクラウドの使用と支出について、現在どのように計画していますか。新たな移行に伴う価値創出を定量化する方法論をお持ちですか? クラウドの可変的な利用を採用するために、現在の予算編成と予測のプロセスを発展させましたか?

  計画の基本方針を定めることは、柔軟な予算編成と予測によって計画を改善することを意味します。可視化とコスト管理を確立した後は、クラウドプロジェクトの支出を計画し、期待値を設定したいと思うことでしょう。AWS は、動的な予測と予算編成のプロセスを柔軟に構築できるため、コストが予算の上限を守っているか、あるいは超えているかを常に把握することができます。

  AWSサービス: AWS Cost Explorer、AWS Cost and Usage Report、AWS Budgets

  リソース: 使用状況ベースの予測、AWS 予算レポート、アラート

- 実行: クラウド支出の管理に使用している運用プロセスとツールにはどのようなものがありますか。また、それらの取り組みは誰が主導していますか。AWS を実際に使い始めてから、日々の運用面で工夫されたことはありますか?

  実行の基本方針とは、請求とコスト管理を実際に行うことです。ガードレールを設定し、ガバナンスを確立することで、経費が予算内に収まるようにすることができます。AWS には、開始するにあたり役に立ついくつかのツールが用意されています。

  AWS サービス: AWS Billing and Cost Managementコンソール、AWS Identity and Access Management、サービスコントロールポリシー (SCP)、AWS Service Catalog、AWS Cost Anomaly Detection、AWS Budgets

  リソース: AWS Billing コンソールの使用を開始する

以下は、クラウドの財務管理におけるベストプラクティスです。

ベストプラクティス

- COST01-BP01 コスト最適化の所有権を設定する
- COST01-BP02 財務とテクノロジーの連携を確立する

- COST01-BP03 クラウドの予算と予測を確立する
- COST01-BP04 組織のプロセスにコスト意識を採り入れる
- COST01-BP05 コスト最適化に関して報告および通知する
- COST01-BP06 コストをプロアクティブにモニタリングする
- COST01-BP07 新しいサービスリリースに関する最新情報を把握しておく
- COST01-BP08 コスト意識を持つ文化を生み出す
- COST01-BP09 コスト最適化によるビジネス価値を数値化する

COST01-BP01 コスト最適化の所有権を設定する

組織全体のコスト認識を確立し、維持する責任を持つチーム (クラウドビジネスオフィス、Cloud Center of Excellence または FinOps チーム) を作成します。コスト最適化の所有者には、組織全体およびクラウド財務を理解している個人またはチーム (財務、テクノロジー、およびビジネスチームの人材が必要) を指定することができます。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

これは、クラウドコンピューティングにおけるコスト意識の文化を確立して維持する責任を負う、クラウドビジネスオフィス (CBO) または Cloud Center of Excellence (CCoE) 機能またはチームについての概要です。この機能は、組織内の個人でもチームでも構いません。組織全体から財務、テクノロジーなどの主な関係者を集めてチームを新規編成することもできます。

担当者 (個人またはチーム) は、コスト管理とコスト最適化活動に必要な時間を、優先順序を付けて配分します。小規模な組織の場合、大企業のフルタイムの担当者と比較すると、費やす時間の割合は少ない場合があります。

プロジェクトマネジメント、データサイエンス、財務分析、ソフトウェアやインフラストラクチャの開発など、複合的なアプローチが求められます。担当者は次の 3 つの異なる所有権内でコスト最適化を実行することにより、ワークロードの効率を高めることができます。

- 集中型: FinOps チーム、クラウド財務管理 (CFM) チーム、クラウドビジネスオフィス (CBO)、Cloud Center of Excellence (CCoE) などの指定チームを通じて、お客様はガバナンスの仕組みを設計、導入し、ベストプラクティスを社内全体で推進することができます。
- 分散型: テクノロジーチームに影響を与え、コスト最適化を実行します。

- ハイブリッド: 集中型チームと分散型チームの両方が協力して、コスト最適化を実行することができます。

この担当者は、コスト最適化目標 (ワークロード効率メトリクスなど) に対する実行および提供能力を評価されることになります。

この担当者のためにエグゼクティブスポンサーシップを確保する必要があり、これが重要な成功要因になります。エグゼクティブスポンサーは、クラウド利用のコスト効率を判断する最高責任者として、チームの考え方を上長にエスカレーションし、組織が定める優先事項としてコスト最適化活動が扱われるようにチームをサポートします。これを怠ると、ガイダンスが無視される可能性があり、コスト削減の機会が優先されなくなります。エグゼクティブスポンサーとチームは、協力して組織のクラウド利用を効率化し、ビジネスバリューを実現できるようにします。

ビジネス、Enterprise-On-Ramp、またはエンタープライズサポートプランを利用していて、このチームまたは担当者の構築に支援が必要な場合は、アカウントチームを通じてクラウド財務管理 (CFM) のエキスパートにご連絡ください。

実装手順

- 主要なメンバーを定義する: コスト管理には、組織内のすべての関係部署が貢献し、関心を持つ必要があります。通常、一般的な組織内チームには、財務、アプリケーションまたはプロダクトの所有者、管理、技術チーム (DevOps) が含まれています。一部は専属 (財務、技術) で、その他は必要に応じて定期的に関与します。CFM を実行する個人またはチームには、以下のスキルセットが必要です。
  - ソフトウェア開発: スクリプトと自動化を構築する場合。
  - インフラストラクチャエンジニアリング: スクリプトをデプロイし、プロセスを自動化して、サービスまたはリソースのプロビジョニング方法を理解します。
  - 運用の洞察力: CFM とは、クラウドの効率的な利用を測定、モニタリング、修正、計画、スケーリングすることで、クラウドで効率的に運用することです。
- 目標とメトリクスを定義する: この担当者は、さまざまな方法で組織に価値をもたらす必要があります。これらの目標は定義され、組織が進化するにつれて継続的に進化します。一般的な活動には、組織全体のコスト最適化に関する教育プログラムの作成と実行、コスト最適化のためのモニタリングやレポート作成などの組織全体の標準策定、最適化に関するワークロード目標の設定などがあります。この担当者は、組織のコスト最適化機能について定期的に組織に報告する必要もあります。

価値ベースまたはコストベースの重要業績指標 (KPI) を定義できます。KPI を定義すると、効率性と予想されるビジネス成果の観点から、予想されるコストを計算できます。価値ベースの KPI は、コストおよび使用量のメトリクスをビジネスバリュー要因に結び付け、AWS の費用の変化を合理化するうえで役立ちます。価値ベースの KPI を導き出す最初のステップは、組織横断的に協力し、KPI の標準セットを選択し、合意することです。

- 定期的なミーティングを設定する: グループ (財務、技術、およびビジネスチーム) は定期的にミーティングを行い、目標とメトリクスを確認する必要があります。一般的なミーティングでは、組織の状態の確認、現在実行中のプログラムの確認、全体的な財務および最適化メトリクスの確認を行います。その後、主要なワークロードの詳細を報告します。

  このような定期的なレビューにより、ワークロードの効率性 (コスト) とビジネス成果を確認できます。例えば、ワークロードのコストが 20% 増加した場合、顧客使用量も増加したかもしれません。この場合、この 20% のコスト増加を投資と解釈できます。このような定期的なミーティングにより、チームは組織全体にとって有意義な価値ベースの KPI を特定できます。

リソース

関連ドキュメント:

- AWS CCOE ブログ
- クラウドビジネスオフィスの作成
- CCOE - Cloud Center of Excellence

関連動画:

- Vanguard CCOE 成功事例

関連する例:

- Cloud Center of Excellence (CCoE) を活用した企業全体の変革
- 企業全体を変革する CCOE の構築
- CCOE を構築するときに回避すべき 7 つの落し穴

COST01-BP02 財務とテクノロジーの連携を確立する

クラウドジャーニーのすべての段階で、コストと使用状況に関するディスカッションに財務チームとテクノロジーチームを参加させます。チームは、定期的に集まり、組織の目標やターゲット、コストと使用状況の現状、財務や会計のプラクティスなどのトピックについて話し合います。

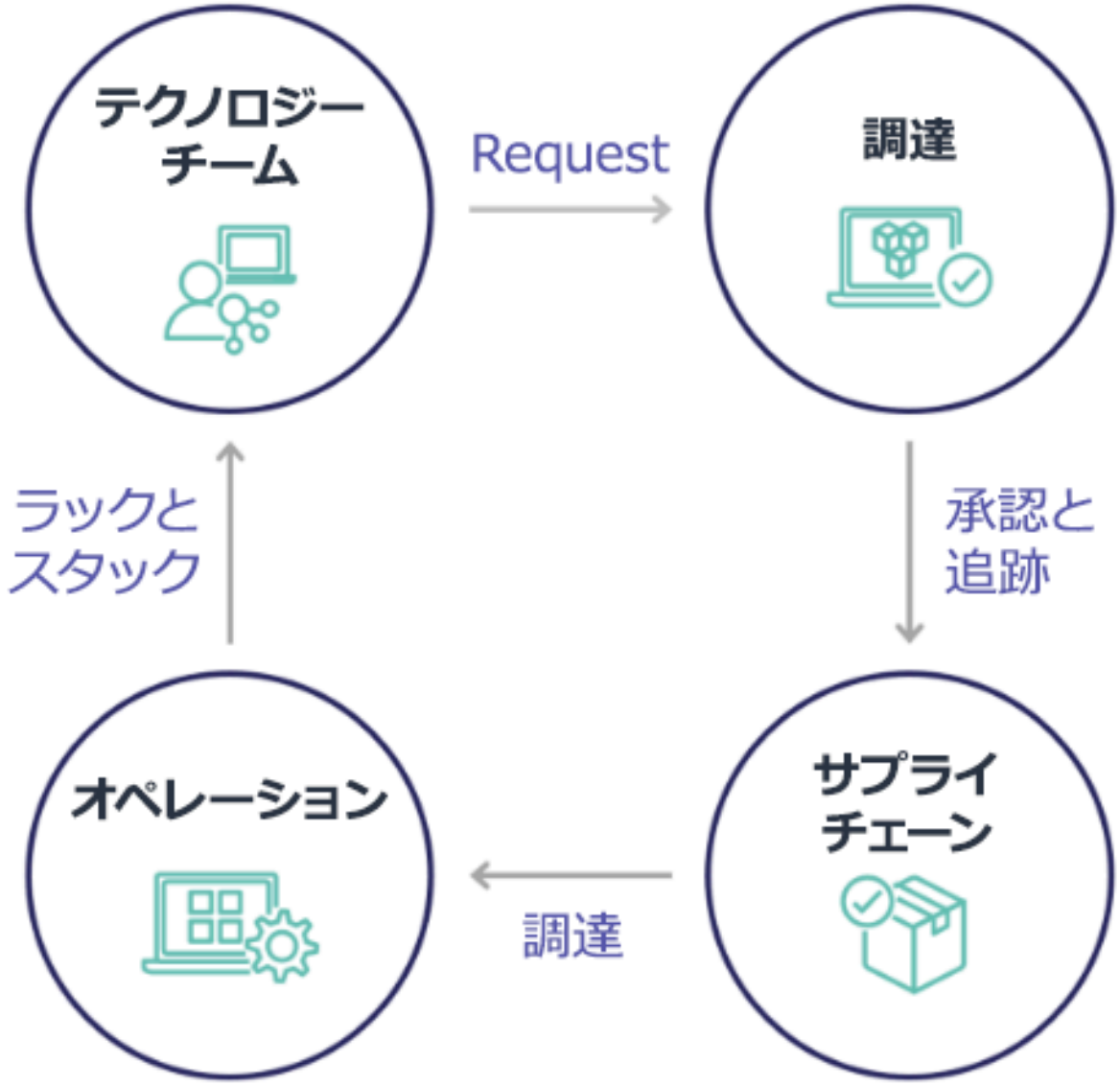
このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

承認、調達、インフラストラクチャのデプロイサイクルが短縮されるため、テクノロジーチームは、クラウドでのイノベーションを迅速化することができます。財務組織はこれまでプロジェクト承認時のデータセンターやオンプレミス環境の調達に大幅に費やしていた時間とリソースを調整することができます。

財務および調達組織の観点から見ると、資本予算、資本要求、承認、調達、物理的インフラストラクチャの設置のプロセスは、何十年にもわたって学習され、標準化されてきたプロセスの 1 つです。

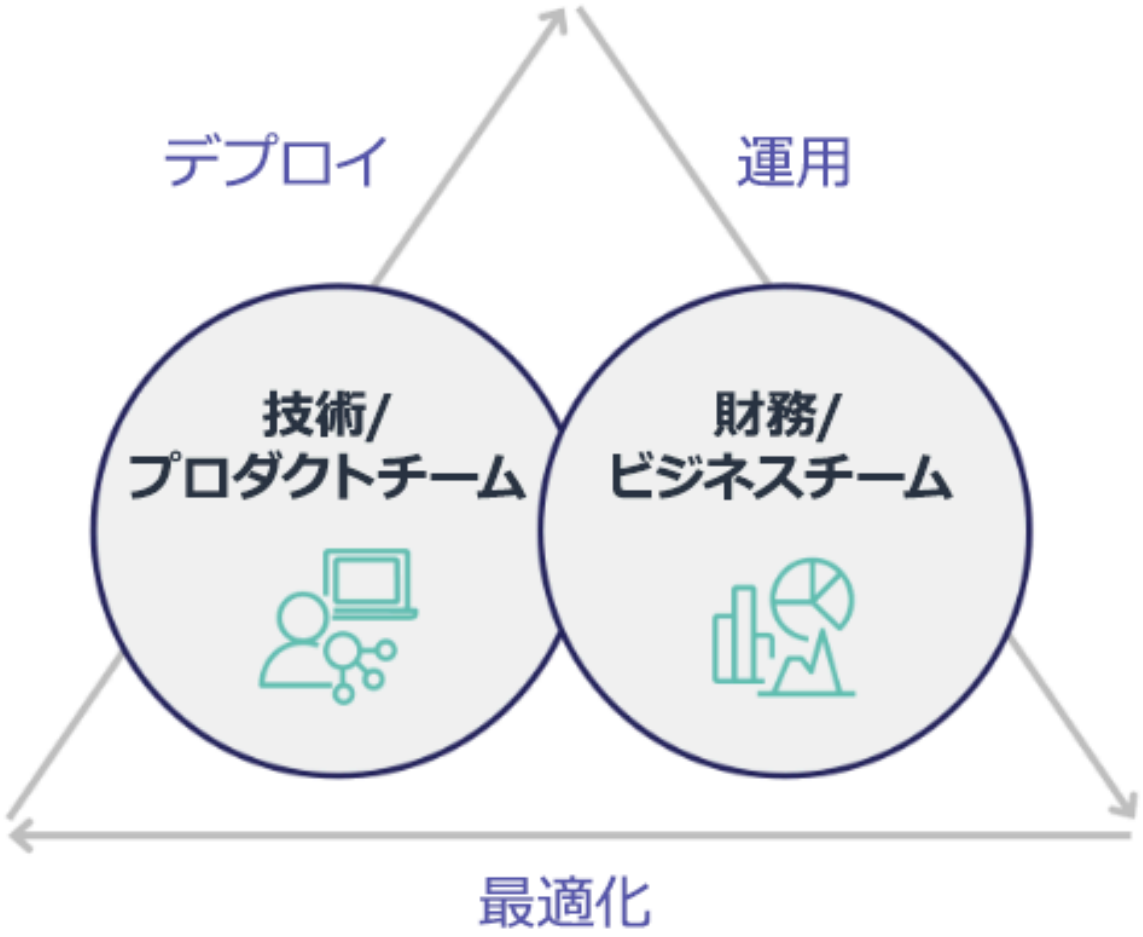
- 通常、エンジニアリングチームや IT チームは依頼主です。
- さまざまな財務チームが承認者、調達者として機能します。
- 運用チームは、すぐに使えるインフラストラクチャのラック、スタック、および引き渡しを行います。



クラウドの導入により、インフラストラクチャの調達と消費は依存関係の連鎖と切り離されます。クラウドモデルでは、テクノロジーチームと製品チームは単なる構築者ではなく、製品の運用者兼所有者となり、調達とデプロイなど、従来は財務および運用チームに割り当てられていた活動のほとんどを担当します。

クラウドリソースのプロビジョニングに必要なものは、アカウントと適切なアクセス許可のセットだけです。これは IT および財務リスクを軽減することにもなります。つまり、チームは常に数回のクリックまたは API コールで、アイドル状態または不要なクラウドリソースを停止することができるのです。また、テクノロジーチームがより迅速にイノベーションを起こすことができるのも、実験を立ち上げては破棄する俊敏性と能力があってこそです。クラウド消費には変動的な性質があるため、資本支出予算と予測の観点から予測可能性に影響することがある一方で、クラウドによって組織は、

オーバープロビジョニングのコストを削減し、控えめなアンダープロビジョニングに伴う機会コストも削減できます。



主要な財務部門とテクノロジー部門のステークホルダーどうしのパートナーシップを確立し、組織の目標の共通理解を深め、クラウドコンピューティングのさまざまな消費モデルにおいて財務上の成功を実現するためのメカニズムを開発します。クラウドジャーニーのすべてのステージにおいて、コストと使用量に関するディスカッションに参加する必要がある組織内の関連するチームは、以下のとおりです。

- ファイナンシャルリード: CFO、財務管理者、フィナンシャルプランナー、ビジネスアナリスト、調達、ソーシング、支払担当は、クラウドの消費モデル、購入オプション、月次請求プロセスを理解する必要があります。財務部門は、テクノロジーチームと連携して、IT バリューの事例を作成およびソーシャル化し、テクノロジー関連の支出がビジネスの成果にどのように関連しているのかをビジネスチームが理解できるようにする必要があります。このように、テクノロジー関連の支出はコストではなく投資とみなされます。クラウド運用にはオンプレミスのオペレーションと比べて

根本的な違い (使用量の変動率、従量制料金やティア別料金、料金モデル、請求明細と使用量情報など) があるため、クラウド利用が調達プロセス、インセンティブ追跡、コスト配分、財務諸表などのビジネス局面に与えるインパクトをファイナンス部門で理解することが不可欠です。

- テクノロジーリード: テクノロジーリード (製品およびアプリケーションの所有者を含む) は、財務要件 (予算の制約など) やビジネス要件 (サービスレベルアグリーメントなど) を認識する必要があります。これにより、組織が目指すビジネス目標を達成するワークロードの導入が可能になります。

財務とテクノロジーのパートナーシップには、以下のような利点があります。

- 財務チームとテクノロジーチームは、コストと使用量をほぼリアルタイムで把握できます。
- 財務チームとテクノロジーチームは、クラウドへの支出の変動に対応するための標準となる運用手順を確立します。
- 財務部門のステークホルダーは、コミットメント割引 (リザーブドインスタンスや AWS Savings Plans など) の購入に資金がどう使用されるか、また組織を拡大するためにクラウドがどのように利用されるかに関して、戦略アドバイザーとして行動します。
- 既存の支払いアカウントと調達プロセスは、クラウドと共に使用されます。
- 財務チームとテクノロジーチームは、協力して将来的な AWS のコストと使用量を予測し、組織の予算を調整および構築します。
- 両者の共通言語により組織間のコミュニケーションが向上し、財務の概念への共通理解が得られます。

コストと使用量のディスカッションについて、組織内でかかわるべきその他のステークホルダーは以下のとおりです。

- 事業部門オーナー: 事業部門オーナーは、事業部門と会社全体の両方に方向性を提供できるように、クラウドのビジネスモデルを理解する必要があります。こうしたクラウド知識は成長とワークロード使用量を予測する際に、またリザーブドインスタンスや Savings Plans などの長期購入オプションを検討する際に重要な役割を果たします。
- エンジニアリングチーム: エンジニアがクラウド財務管理 (CFM) に取り組むよう促す、コストを意識した企業文化の構築には、財務チームとテクノロジーチームのパートナーシップの確立が欠かせません。CFM や財務業務の実務担当者や財務チームが共通して抱える問題の 1 つは、エンジニアにクラウド上のビジネス全体を理解させ、ベストプラクティスに従わせ、推奨されるアクションを取らせることです。

- サードパーティー: サードパーティー (コンサルタントやツールなど) を利用する場合、こうしたサードパーティーが財務目標に適合し、エンゲージメントモデルと投資収益率 (ROI) を通じて、どちらの整合性も実証できるようにします。通常、サードパーティーは自社管理のワークロードのレポーティングと分析を担当したり、自社設計のワークロードのコストを分析したりします。

CFM を導入し、成功させるには、財務、テクノロジー、ビジネスの各チームが協力し、組織全体におけるクラウド費用の伝達と評価の方法を変える必要があります。エンジニアリングチームを巻き込み、あらゆる段階でコストと使用に関する議論に参加させ、ベストプラクティスに従って合意されたアクションを取るよう奨励します。

実装手順

- 主要なメンバーを定義する: 財務チームとテクノロジーチームのすべての関連メンバーがこの連携に関与していることを確認します。関連する財務メンバーは、クラウドの請求書に関する業務に従事するメンバーです。通常は、CFO、財務コントローラー、財務プランナー、ビジネスアナリスト、購買管理です。テクノロジーチームのメンバーは通常、製品およびアプリケーションの所有者、テクニカルマネージャー、およびクラウド上に構築するすべてのチームの代表者です。他のメンバーとしては、製品の使用に影響するマーケティングなどのビジネスユニットの所有者、および貴社の目標やメカニズムとの整合性を確保し、報告をサポートするコンサルタントなどのサードパーティーが含まれることがあります。
- ディスカッションのためのトピックを定義する: チーム間で共通する、または理解を共有する必要があるトピックを定義します。作成から支払いまでにかかるコストを追います。関連するメンバー、および適用する必要がある組織のプロセスを書き留めます。各ステップまたはプロセスのほか、利用可能な料金モデル、階層化された料金、割引モデル、予算、財務要件などの関連情報を理解します。
- 定期的なミーティングを設定する: 財務とテクノロジーのパートナーシップを構築するために、定期的なコミュニケーションの機会を設け、連携を維持します。グループは目標とメトリクスに照らして定期的に集まる必要があります。一般的なミーティングでは、組織の状態の確認、現在実行中のプログラムの確認、全体的な財務および最適化メトリクスの確認を行います。その後、主要なワークロードが詳細に報告されます。

リソース

関連ドキュメント:

- AWS ニュースブログ

COST01-BP03 クラウドの予算と予測を確立する

既存の組織の予算作成および予測プロセスを調整し、非常に変動しやすいクラウドのコストと使用状況の性質に対応できるようにします。プロセスは、トレンドベースまたはビジネスドライバーベースのアルゴリズム、またはそれらの組み合わせを使用して、動的なものにする必要があります。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

従来のオンプレミス IT 環境では、ピーク需要を満たすための新しい IT ハードウェアやサービスの購入など、まれにしか変化しない固定費を計画するという課題に直面することがあります。これとは対照的に、AWS クラウドでは、お客様が実際の IT ニーズやビジネスニーズに応じて、使用するリソースに対して料金を支払うという異なるアプローチを取っています。クラウド環境では、需要が月単位、日単位、さらには時間単位で変動する場合があります。

クラウドを使用すると効率性、スピード、俊敏性が向上し、その結果、コストと使用パターンの変動性も高まります。コストは、ワークロードの効率性の向上、または新しいワークロードや機能のデプロイによって減少することもあれば、場合によっては増加することもあります。拡大する顧客ベースに対応するためのワークロードのスケールに伴って、リソースのアクセス性が高まるため、クラウドの使用量とコストも相応に増加します。クラウドサービスにおけるこの柔軟性はコストと予測にも及び、一定の伸縮性がもたらされます。

こうしたビジネスニーズや需要ドライバーの変化に合わせて密に調整を行い、可能な限り正確な計画を目指すことが不可欠です。この変動に対応できるように、従来の組織の予算プロセスを適応させる必要があります。

新しいワークロードのコストを予測する際は、コストモデリングを検討します。コストモデリングによって、予想されるクラウドコストのベースラインを把握できるため、これを基に総保有コスト (TCO)、投資収益率 (ROI) およびその他の財務分析を実行し、関係者と共に目標と期待値を設定して、コスト最適化の機会を特定することができます。

組織は、コストの定義と承認されているグループ化を理解しておく必要があります。予測の詳細度の度合いは、組織の構造と社内ワークフローによって異なります。特定の要件と組織環境に適した詳細度を選択してください。どのレベルで予測を実行するかを理解しておくことが重要です。

- 管理アカウントまたは AWS Organizations レベル: 管理アカウントは、AWS Organizations の作成に使用するアカウントです。組織には、デフォルトで 1 つの管理アカウントがあります。
- 連結アカウントまたはメンバーアカウント: 組織のアカウントは標準の AWS アカウントで、AWS リソースと、それらのリソースにアクセスできる ID を含みます。

- 環境: 環境は、アプリケーションバージョンを実行する AWS リソースのコレクションです。環境は、複数の連結アカウントまたはメンバーアカウントで構成できます。
- プロジェクト: プロジェクトは、一定の期間内に達成すべき目標またはタスクの組み合わせです。予測する際は、プロジェクトのライフサイクルを考慮することが重要です。
- AWS サービス: 予測のために AWS サービスをグループ化できるコンピューティングサービスやストレージサービスなどのグループまたはカテゴリ。
- カスタムグループ化: ビジネスユニット、コストセンター、チーム、コスト配分タグ、コストカテゴリ、連結アカウント、これらの組み合わせなど、組織のニーズに基づいてカスタムグループを作成できます。

使用コストに影響を与える可能性のあるビジネスドライバーを特定し、それぞれについて個別に予測して、予想される使用量を事前に計算します。組織内の IT チームや製品チームに関連するドライバーもあります。マーケティングイベント、プロモーション、地理的拡大、合併、買収など、その他のビジネスドライバーについては、営業、マーケティング、ビジネス部門のリーダーが把握しているため、これらの関係者と協力して、これらすべての需要ドライバーについても考慮することが重要です。

AWS Cost Explorer を使用すると、過去の支出に基づいて、定義された将来の時間範囲におけるトレンドベースの予測を行うことができます。AWS Cost Explorer の予測エンジンは、料金タイプ (リザーブドインスタンスなど) に基づいて履歴データをセグメント化し、機械学習とルールベースのモデルを組み合わせて使用し、すべての料金タイプにわたる費用を個別に予測します。

予測プロセスを確立しモデルを構築した後、AWS Budgets を使用して、期間、繰り返し、または金額 (固定費または可変費) を指定し、サービス、AWS リージョン、タグなどのフィルターを追加することで、カスタム予算を詳細レベルで設定します。予算は通常 1 年単位で設定され、固定されるため、関係者全員の厳守が求められます。一方、予測はより柔軟であり、年間を通じて再調整が可能で、1 年、2 年、または 3 年の期間にわたる動的な予測が可能です。予算と予測はどちらも、テクノロジーやビジネスのさまざまなステークホルダーの間で財務上の期待事項を確立するうえで重要な役割を果たします。正確な予測と実装は、第一にコストのプロビジョニングに直接責任を負うステークホルダーに説明責任をもたらし、全体的なコスト意識を高めることにもつながります。

既存予算のパフォーマンスについて常に情報を入手するには、定期的に AWS Budgets レポートを作成して自分とステークホルダーに E メールで送信されるようにスケジュールします。また、実際のコストに基づいて AWS Budgets アラートを作成することもできます (これは反応型です)。または、予測コストに基づいて作成することも可能で、この場合は潜在的なコスト超過に対する緩和策を実施する時間を確保することができます。コストや使用量が一定レベルを実際に超えた場合、または予算額を超えると予測された場合にアラートを受け取ることができます。

トレンドベースのアルゴリズム (コスト履歴を入力値として使用) と、動的で支出が変動する環境に最適なドライバーベースのアルゴリズム (新製品の発売や営業地域の拡大、ワークロードの新しい環境など) を使用して、既存の予算編成と予測のプロセスをより動的なものになるよう調整します。Cost Explorer またはその他のツールを使用してトレンドベースの予測を決定したら、AWS 料金見積りツール を使用し、予想される使用量 (トラフィック、1 秒あたりのリクエスト数、または必要な Amazon EC2 インスタンス) に基づいて AWS ユースケースと将来のコストを見積もります。

予算はこうした予測計算と見積もりに基づいて設定する必要があるため、その予測の正確性を追跡します。統合されたクラウドコスト予測の正確性と有効性を監視します。実際の支出を予測と比較して定期的に見直し、必要に応じて調整して予測の精度を向上させます。予測の差異を追跡し、報告された差異について根本原因分析を実行して、措置を講じ、予測を調整します。

COST01-BP02 財務とテクノロジーの連携を確立する で説明しているように、IT、財務、その他の関係者が一貫性を保つために同じツールやプロセスを使用し、パートナーとなって連携することが重要です。予算を変更する必要がある場合は、ミーティングの回数を増やし、こうした変更により迅速に対応できるようにします。

実装手順

- 組織内のコスト言語を定義する: 複数のディメンションとグループ化を使用して、組織内に共通の AWS コスト言語を作成します。ステークホルダーが予測の詳細度、料金モデル、およびコスト予測のレベルを理解していることを確認します。
- トレンドベースの予測を分析する: AWS Cost Explorer や Amazon Forecast などのトレンドベースの予測ツールを使用します。サービス、アカウント、タグ、コストカテゴリなど、複数のディメンションで使用コストを分析します。
- ドライバーベースの予測を分析する: ビジネスドライバーがクラウドの使用に与える影響を特定し、それぞれについて個別に予測して、予想される使用コストを事前に計算します。ビジネスユニットのオーナーやステークホルダーと緊密に連携して新しいドライバーへの影響を把握し、予想されるコストの変化を計算して正確な予算を決定します。
- 既存の予測および予算編成プロセスを更新する: トレンドベース、ビジネスドライバーベースなど、採用されている予測方法、または両方の予測方法の組み合わせに基づいて、予測および予算編成プロセスを定義します。予算は計算された現実的なもので、予測に基づいている必要があります。
- アラートと通知を設定する: AWS Budgets アラートとコスト異常検出を使用して、アラートと通知を取得します。
- 主要関係者と定期的なレビューを行う: 例えば、IT、財務、プラットフォームの各チーム、およびその他のビジネス分野の関係者と、ビジネスの方向性や使用状況の変化について調整します。

リソース

関連ドキュメント:

- AWS Cost Explorer
- AWS Cost and Usage Report
- Cost Explorer で予測する
- QuickSight 予測
- AWS Budgets

関連動画:

- AWS Budgets を使用して支出と使用量を追跡するにはどうすればよいですか?
- AWS コスト最適化シリーズ: AWS Budgets

関連する例:

- ドライバーベースの予測を理解し構築する
- 予測の文化を醸成する方法
- クラウドコスト予測の改善方法
- 適切なツールによるクラウドコスト予測

COST01-BP04 組織のプロセスにコスト意識を採り入れる

コスト意識を高め、透明性を強化し、使用量に影響する新規または既存のプロセスにコストの説明責任を採り入れ、コスト意識に関する既存のプロセスを活用します。従業員のトレーニングにコスト意識の要素を採り入れます。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

コスト意識は、組織の新規および既存のプロセスに採り入れる必要があります。他のベストプラクティスの基盤となる前提条件の機能の 1 つです。可能な限り既存のプロセスを再利用し、修正することが推奨されます。これにより、俊敏性と速度への影響を最小化することができます。クラウドのコストをテクノロジーチームと、ビジネスチームおよび財務チームの意思決定者に報告して、コスト

意識を高め、効率性についての重要業績評価指標 (KPI) を財務およびビジネス部門関係者向けに確立します。次の推奨事項は、ワークロードにコスト意識を実装するのに役立ちます。

- 変更による財務への影響を数値化するコスト測定が変更管理に含まれていることを確認します。これは、コスト関連の懸念に積極的に対処し、コスト削減を強調するのに役立ちます。
- コスト最適化が、業務能力の中核をなす要素であることを確認します。例えば、既存のインシデント管理プロセスを活用して、コストと使用量に関する異常値 (コスト超過) の根本原因を調査、特定することができます。
- オートメーションやツールにより、コスト削減とビジネス価値の実現を加速します。導入コストを考える場合、時間や費用の投資を正当化するために、投資収益率 (ROI)の要素を含むように話を組み立てます。
- コミットメントベースの購入オプション、共有サービス、マーケットプレイスでの購入を含むクラウド使用に対してショーバックまたはチャージバックを実施し、最もコストを意識したクラウド消費を促進することで、クラウドのコストを配分します。
- 既存のトレーニングおよび開発プログラムを拡張し、コスト意識向上のためのトレーニングを組織全体で実施します。これには継続的なトレーニングと認定を含めることをお勧めします。これにより、コストと使用量を自己管理できる組織が育成されます。
- AWS Cost Anomaly Detection、AWS Budgets、AWS Budgets レポートのような無料の AWS ネイティブツールを利用します。

組織がクラウド財務管理 (CFM) プラクティスを一貫して採用すると、それらの行動は仕事や意思決定の方法に根付いていきます。その結果、新しいクラウドで生まれたアプリケーションを設計する開発者から、これらの新しいクラウド投資の ROI を分析する財務マネージャーに至るまで、よりコストを意識した文化が生まれます。

実装手順

- 関連する組織のプロセスを把握する: 各組織単位は、そのプロセスをレビューし、コストと使用状況に影響を与えるプロセスを特定します。リソースの作成や終了につながるすべてのプロセスをレビュー対象とする必要があります。インシデント管理やトレーニングなど、ビジネスにおけるコスト意識の支援につながるプロセスを探します。
- コストを意識した自立的な企業文化を確立する: すべての関係者がクラウドのコストを理解できるように、変更原因と影響をコストとして認識するようにします。これにより、組織がコストを意識したイノベーションの文化を自立的に確立することができます。
- コストを意識したプロセスに更新する: 各プロセスをコストが意識されるよう変更します。このプロセスでは、コストの影響の評価などの追加の事前チェック、またはコストと使用状況の予想さ

れた変化が発生したかどうかを検証する事後チェックが必要になる場合があります。トレーニングやインシデント管理などのサポートプロセスは、コストと使用状況の項目を含むように拡張できます。

ご不明な点がありましたら、アカウントチームを通じて CFM のエキスパートにお問い合わせいただくか、以下のリソースや関連ドキュメントをご覧ください。

リソース

関連ドキュメント:

- AWS クラウド財務管理

関連する例:

- 効率的なクラウドコスト管理の戦略
- コスト管理ブログシリーズ #3: コストショックの扱い方
- AWS Cost Management 初心者ガイド

COST01-BP05 コスト最適化に関して報告および通知する

クラウド予算を設定して、使用量の異常を検出するメカニズムを設定します。関連ツールで、事前定義済み目標に対するコストと使用量に関連するアラートを設定し、使用量がそれらの目標を超えた場合に通知を受け取るようにします。定期的にミーティングを開催して、ワークロードのコスト効率を分析し、社内にコスト意識を浸透させます。

このベストプラクティスを活用しない場合のリスクレベル: 低

実装のガイダンス

組織内のコストと使用量の最適化について、定期的に報告する必要があります。コストパフォーマンスについて話し合う専用セッションの運用や、ワークロードの定期的な運用レポートサイクルにコスト最適化を盛り込むことも意味があるでしょう。サービスとツールを使用して、コストパフォーマンスを定期的にモニタリングし、コスト節減機会を実現します。

AWS Cost Explorer を使用して、複数のフィルターと詳細度でコストと使用状況を確認できます。これにより、サービス別またはアカウント別のコスト、日次コスト、マーケットプレイスコストなどの

ダッシュボードとレポートが表示されます。設定された予算に対するコストと使用量の進行状況を追跡するには、AWS Budgets レポートを使用します。

AWS Budgets を使用して、コストと使用状況を追跡し、しきい値を超えた場合に E メールまたは Amazon Simple Notification Service (Amazon SNS) 通知から受信したアラートにすばやく対応できるようにカスタム予算を設定します。優先予算期間を日次、月次、四半期、または年次に設定し、特定の予算制限を作成して、実際のコストまたは予測コストと使用状況が予算しきい値に対してどのように推移するかを常に把握します。アラートとそれらのアラートに対するアクションを設定し、自動的に実行することも、予算目標を超えたときに承認プロセスを通じて実行するようにもできます。

コストと使用状況に関する通知を実装して、想定外の場合にコストと使用状況の変化を迅速に処理できるようにします。AWS Cost Anomaly Detection を使用すると、イノベーションを遅らせることなく想定外のコストを減らし、制御を強化できます。AWS Cost Anomaly Detection は異常な支出と根本原因を特定し、想定外の請求のリスクを減らすために役立ちます。3 つのシンプルなステップで、状況に応じて独自のモニタを作成し、異常な支出が検出されたときにアラートを受け取ることができます。

QuickSight と AWS Cost and Usage Report (CUR) データを使用して、高度にカスタマイズされたレポートでより詳細なデータを示すこともできます。QuickSight では、レポートのスケジュールを設定したり、過去のコストと使用状況やコスト削減の機会に関する定期的なコストレポート E メールを受信したりできます。高度な可視性をもたらす、QuickSight 上に構築された Cost Intelligence Dashboard (CID) ソリューションを確認してください。

AWS Trusted Advisor を使用すると、プロビジョニングされたリソースがコスト最適化のための AWS のベストプラクティスに準拠しているかどうかを検証するためのガイダンスが提供されます。

視覚的なグラフで、詳細なコストと使用量に関する Savings Plans のレコメンデーションを確認できます。時間単位のグラフには、オンデマンドの支出と推奨される Savings Plans のコミットメントが表示され、推定削減額、Savings Plans の適用範囲、Savings Plans 使用率に関するインサイトが得られます。これにより、組織は支出分析モデルの構築に時間とリソースを費やすことなく、毎時の支出にどのように Savings Plans が適用されているかを理解できます。

Savings Plans、リザーブドインスタンス、および Amazon EC2 の適切なサイズ設計に関する AWS Cost Explorer からのレコメンデーションを含んだレポートを定期的に作成して、定常状態のワークロード、アイドルおよび使用量の少ないリソースに関するコストの削減を開始します。デプロイされているリソースのうち、クラウドの無駄に関する費用を特定し、回収します。クラウドの無駄は、サイズ設定が正しくないリソースが作成されたときや、使用量のパターンが予想とは異なるときに発生します。AWS のベストプラクティスに従って無駄を削減するか、アカウントチームやパートナーにクラウドコストの最適化と節約の支援を依頼します。

定期的にレポートを作成し、リソースの購入オプションを改善することで、ワークロードの単価を下げることができます。Savings Plans、リザーブドインスタンス、Amazon EC2 スポットインスタンスなどの購入オプションは、耐障害性の高いワークロードのコストを最も低く抑え、関係者 (ビジネス所有者、財務チーム、テクノロジーチーム) がこれらのコミットメントの議論に参加できるようにするものです。

クラウドの総保有コスト (TCO) の削減に役立つ可能性のある機会または新しいリリースの発表を含むレポートを共有します。新しいサービス、リージョン、機能、ソリューション、またはさらにコスト削減を実現する新しい方法を採用します。

実装手順

- AWS Budgets を設定する: ワークロードのすべてのアカウントで AWS Budgets を設定します。タグを使用して、アカウント全体の支出の予算とワークロードの予算を設定します。
  - Well-Architected ラボ: コストと使用に関するガバナンス
- コスト最適化について報告する: ワークロードの効率について話し合い、分析する定期的なミーティングを設定します。確立されたメトリクスを使用して、達成されたメトリクスとそれを達成するためにかかったコストを報告します。好ましくない傾向を特定して修正すると共に、組織全体で推進できるような改善傾向を特定します。報告には、アプリケーションチームと所有者、財務、およびクラウド支出に関する主要な意思決定者の代表者が参加する必要があります。

リソース

関連ドキュメント:

- AWS Cost Explorer
- AWS Trusted Advisor
- AWS Budgets
- AWS Cost and Usage Report
- AWS Budgets のベストプラクティス
- Amazon S3 分析

関連する例:

- AWS クラウドコストの最適化を開始する主な方法

COST01-BP06 コストをプロアクティブにモニタリングする

ツールとダッシュボードを実装して、ワークロードのコストをプロアクティブにモニタリングします。通知を受けたときだけコストやカテゴリを見るのではなく、設定されたツールや既存のツールで定期的にコストを見直しましょう。コストをプロアクティブにモニタリングし、分析することで、ポジティブな傾向を把握し、組織全体で推進することが可能になります。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

例外や異常がある場合に限らず、組織内のコストと使用量を事前にモニタリングすることを推奨します。オフィスや職場環境全体を高度に可視化するダッシュボードにより、主な担当者が必要な情報にアクセスできるようになります。また組織がコスト最適化を重視していることを示すことができます。可視化されたダッシュボードにより、成功事例を積極的に推進し、組織全体で実践することができます。

AWS Cost Explorer または Amazon QuickSight などの他のダッシュボードを使用してコストを確認しプロアクティブに分析するための、日次または頻繁なルーチンを作成します。AWS サービスの使用量とコストを AWS アカウントレベル、ワークロードレベル、または特定の AWS サービスレベルでグループ化とフィルタリングを使用して分析し、予想どおりかどうかを検証します。時間単位、リソース単位の詳細度やタグを使用して、上位リソースの発生コストをフィルタリングし、特定します。AWS ソリューションアーキテクトによって構築された Amazon QuickSight ソリューションである Cost Intelligence Dashboard を使用して独自のレポートを作成し、予算を実際のコストや使用状況と比較することもできます。

実装手順

- コスト最適化について報告する: ワークロードの効率について話し合い、分析する定期的なミーティングを設定します。確立されたメトリクスを使用して、達成されたメトリクスとそれを達成するためにかかったコストを報告します。ネガティブな傾向を特定して修正し、ポジティブな傾向を特定して組織全体に普及させます。報告には、アプリケーションチームと所有者、財務、経営の代表者が参加する必要があります。
- コストと使用量の日次詳細度 AWS Budgets を作成してアクティブ化し、潜在的なコスト超過を防ぐためのアクションをタイムリーに実行する: AWS Budgets ではアラート通知を設定できるため、予算タイプが事前設定されたしきい値から外れても常に把握できます。AWS Budgets を活用する最善の方法は、予想されるコストと使用量を限度として設定することです。それにより、予算を超えたものは使い過ぎとみなすことができます。

- コストモニタとして AWS Cost Anomaly Detection を作成する: AWS Cost Anomaly Detection は、高度な機械学習テクノロジーによって異常な支出と根本原因を特定するため、迅速に対策を講じることができます。評価したい支出セグメント (例えば、個々の AWS サービス、メンバーアカウント、コスト配分タグ、コストカテゴリ) を定義するコストモニタを設定することができ、アラート通知をいつ、どこで、どのように受け取るかを設定することが可能です。各モニタには、ビジネスオーナーやテクノロジーチーム向けの複数のアラートサブスクリプションをアタッチし、各サブスクリプションの名前、コスト影響しきい値、アラート頻度 (個別アラート、日次サマリー、週次サマリー) などを設定します。
- AWS Cost Explorer を使用して、または AWS Cost and Usage Report (CUR) データを Amazon QuickSight ダッシュボードに統合して、組織のコストを可視化する: AWS Cost Explorer には、経時的に AWS コストと使用状況を確認し、理解して管理することを可能にする使いやすいインターフェイスがあります。カスタマイズ可能でアクセスしやすい Cost Intelligence Dashboard が、独自のコスト管理と最適化ツールの基盤作成を支援します。

リソース

関連ドキュメント:

- AWS Budgets
- AWS Cost Explorer
- 日次コストおよび使用量の予算
- AWS Cost Anomaly Detection

関連する例:

- Slack による AWS Cost Anomaly Detection アラート

COST01-BP07 新しいサービスリリースに関する最新情報を把握しておく

エキスパートや AWS パートナーに定期的に相談して、コストの低いサービスと機能を検討します。AWS のブログやその他の情報ソースを確認します。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

AWS は常に新しい機能を追加しているため、最新のテクノロジーを利用して、実験とイノベーションをより迅速に行うことができます。新しい AWS のサービスや機能を実装することで、ワークロードのコスト効率を改善できる場合があります。新しいサービスと機能のリリース情報については、「AWS Cost Management」、「AWS ニュースブログ」、「AWS コスト管理ブログ」、「AWS の最新情報」を定期的に確認してください。「最新情報」では、AWS サービス、機能、リージョン拡大の発表があった際に、その概要をお知らせしています。

実装手順

- ブログをサブスクライブする: AWS ブログのページにアクセスし、最新情報ブログやその他の関連ブログにサブスクライブします。通信設定ページで E メールアドレスを使用してサインアップできます。
- AWS ニュースをサブスクライブする: 新しいサービスと機能のリリース情報については、「AWS ニュースブログ」および「AWS の最新情報」を定期的に参照してください。RSS フィードまたは E メールでお知らせやリリースを購読することができます。
- AWS の値下げをフォローする: すべてのサービスにおいて定期的な値下げを行うことは、AWS がその規模から得られる経済的な効率性をお客様に還元するための標準的な方法です。AWS は 2006 年以降、134 回値下げしています (2023 年 9 月 20 日現在)。料金面の懸念から経営判断を保留しているものがあれば、値下げや新サービス統合後に再度見直すことも可能です。Amazon Elastic Compute Cloud (Amazon EC2) インスタンスを含む過去の値下げの取り組みについては、「AWS ニュースブログ」の料金引き下げカテゴリを参照してください。
- AWS のイベントおよび交流: ローカルの AWS サミットや、地域内の他の組織との交流に参加しましょう。直接参加できない場合は、バーチャルイベントに参加して、AWS のエキスパートや他のお客様のビジネスケースから情報を得るようにしてください。
- アカウントチームとのミーティングを設ける: アカウントチームとの定期的なミーティングを設定し、チームと会い、業界の動向と AWS のサービスについて話し合います。アカウントマネージャー、ソリューションアーキテクト、サポートチームに相談します。

リソース

関連ドキュメント:

- AWS コスト管理
- AWS の最新情報

- AWS ニュースブログ

関連する例:

- Amazon EC2 - IT コストの最適化と削減に取り組んだ 15 年間
- AWS ニュースブログ - 料金引き下げ

COST01-BP08 コスト意識を持つ文化を生み出す

コストを意識した企業文化を醸成するために、組織全体で改革やプログラムを実施しましょう。まず小さく始めて、機能や組織でのクラウド利用の増加に合わせて規模を拡大していき、さまざまなプログラムを運用していくことをお勧めします。

このベストプラクティスを活用しない場合のリスクレベル: 低

実装のガイダンス

コスト意識を持つ文化があると、組織全体で有機的かつ分散的に実行されるベストプラクティスを通じて、コストの最適化とクラウド財務管理 (財務運用、Cloud Center of Excellence、クラウド運用チームなど) の規模を拡大できます。コスト意識を持つことで、トップダウンで集中的に行う厳格なアプローチと比較して、最小限の労力で組織全体に高いレベルの能力を生み出すことができます。

クラウドコンピューティング、特にクラウドコンピューティングの主なコスト要因についてのコスト意識を持つことで、チームは予想される変更の成果をコスト面で理解できます。クラウド環境にアクセスするチームは、料金モデルを意識することと併せ、従来のオンプレミスデータセンターとクラウドコンピューティングの違いを意識する必要があります。

コストを意識する文化の主な利点は、技術チームが必要に応じて消極的なコスト最適化を行うのではなく、積極的かつ継続的にコストを最適化することにあります (例えば、新しいワークロードを設計する場合や既存のワークロードを変更する場合、コストは機能要件でないとみなされます)。

この文化のわずかな変化が、現在および将来のワークロードの効率に大きな影響を与える可能性があります。これには、次のような例があります。

- エンジニアリングチームに可視性を与え、意識を高めることで、自分たちが何をしているのか、コスト面にどのような影響を与えるのかを理解させることができます。
- 組織全体のコストと使用量にゲーム的要素を取り入れる。これは、公開ダッシュボードや、チーム間の標準コストと標準使用量 (ワークロードあたりのコストやトランザクションあたりのコストなど) を比較するレポートによって実行できます。

- コスト効率を認識する。自発的または独断で行なったコスト最適化の成果を公開または非公開で評価して、間違いから学び、今後繰り返さないようにします。
- あらかじめ設定された予算でワークロードを実行するために、トップダウンの組織的要件を作成します。
- 変更のビジネス要件と、アーキテクチャインフラストラクチャまたはワークロード設定に対して要求された変更がコストに及ぼす影響を探求して、必要な分だけを支払うようにします。
- 変更計画者は、想定される変更のコストへの影響を意識し、高いコスト効率でビジネス成果をもたらすように関係者に確認してもらう必要があります。

実装手順

- クラウドコストをテクノロジーチームに報告する: これによりコスト意識を高め、財務およびビジネス関係者にとって効率的な KPI を確立します。
- 予定されている変更を関係者やチームメンバーに通知する: 予定されている変更とワークロードに対する費用対効果を週次の変更ミーティングで協議するための議題を作成します。
- アカウントチームとのミーティングを設ける: アカウントチームとの定期的なミーティングを設定し、業界の動向と AWS のサービスについて話し合います。アカウントマネージャー、アーキテクト、サポートチームと話します。
- 成功事例を共有する: ワークロード、AWS アカウント、または組織のコスト削減に関する成功事例を共有して、コスト最適化に関する前向きな姿勢と励みにします。
- トレーニング: 技術チームやチームメンバーが、AWS クラウドに関するリソースコストを意識するためのトレーニングを受けられるようにします。
- AWS のイベントおよび交流: ローカルの AWS サミットや、地域内の他の組織との交流に参加します。
- ブログをサブスクライブする: AWS ブログページに移動し、最新情報ブログやその他の関連ブログにサブスクライブして、AWS が共有する新しいリリース、実装、例、変更をチェックします。

リソース

関連ドキュメント:

- AWS ブログ
- AWS コスト管理
- AWS ニュースブログ

関連する例:

- AWS クラウド財務管理

COST01-BP09 コスト最適化によるビジネス価値を数値化する

コスト最適化でビジネス価値を数値化することで、組織に対するメリットの全体像を把握できます。コスト最適化は必要な投資であるため、ビジネス価値を数値化することで、各ステークホルダーに投資利益率を説明できます。ビジネス価値の数値化により、将来のコスト最適化投資に対して関係者からより多くの賛同を得ることができます。また、組織のコスト最適化活動の成果を測定するためのフレームワークを取得できます。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

ビジネス価値の数値化とは、企業の行動や決断がもたらす利益を測定することです。ビジネス価値は、有形 (経費の削減や利益の向上など) の場合もあれば、無形 (ブランドの評判や顧客満足度の向上など) の場合もあります。

コスト最適化によるビジネス価値の数値化とは、支出を効率化する取り組みがもたらす価値や利益を判断することです。例えば、ある企業が AWS へのワークロードのデプロイに 100,000 USD を費やし、後日最適化した結果、品質や成果を損なうことなく、わずか 80,000 USD にコストダウンしたとします。この場合、コスト最適化がもたらすビジネス価値を数値化すると、20,000 USD の節約になります。しかし、単純な節約額だけでなく、納期の短縮や顧客満足度の向上、コスト最適化の取り組みの成果が現れたその他の指標を踏まえて、価値を数値化することもできます。関係者は、コスト最適化の潜在的価値、ワークロードの最適化にかかるコスト、投資収益率を判断する必要があります。

コスト最適化によって節約した金額を報告することに加えて、実現した付加価値も数値化することをお勧めします。コスト最適化のメリットは通常、ビジネス成果に対して削減されたコストという観点で数値化されます。例えば、Savings Plans を購入すると、Amazon Elastic Compute Cloud (Amazon EC2) のコスト削減を数値化できます。Savings Plans により、コストを削減し、ワークロードの出力レベルを維持できます。アイドル状態の Amazon EC2 インスタンスを削除した場合や、アタッチされていない Amazon Elastic Block Store (Amazon EBS) ボリュームを削除した場合は、AWS の利用料削減を数値化できます。

コスト最適化のメリットは、コスト削減やコスト回避にとどまりません。効率性向上とビジネス価値を測定するために、その他のデータを追加で取得することを検討してください。

実装手順

- ビジネス上のメリットを評価する: これは、AWS クラウドコストを分析、調整して、支出から得られるメリットを最大化するプロセスです。ビジネス価値を顧みずコスト削減にばかり着目するのではなく、コスト最適化がもたらすビジネス上の利点と投資収益率を検討してください。支出した金額からもっと価値を引き出せるようになります。賢く支出し、最大の収益率を見込める分野に投資および出費することが大切です。
- 予測 AWS コストを分析する: 予測により、財務関係者は、組織内外の他の利害関係者と見通しを立て、組織の財務予測の可能性を向上させることができます。AWS Cost Explorer を使用して、コストと使用量を予測できます。

リソース

関連ドキュメント:

- AWS クラウドエコノミクス
- AWS ブログ
- AWS コスト管理
- AWS ニュースブログ
- Well-Architected 信頼性の柱のホワイトペーパー
- AWS Cost Explorer

関連動画:

- AWS で Windows を使用してビジネス価値を実現する

関連する例:

- CUSTOMER 360 のビジネス価値を測定し最大化する
- Amazon Web Services マネージドデータベースを採用することのビジネス価値
- 独立系ソフトウェアベンダーが Amazon Web Services を採用することのビジネス価値
- クラウドモダナイゼーションのビジネス価値
- Amazon Web Services に移行することのビジネス価値

経費支出と使用量の認識

組織のコストおよびコスト要因を把握することは、コストと使用量を効果的に管理し、コスト削減の機会を特定するうえできわめて重要です。組織では一般に、複数のワークロードが複数のチームによってオペレーションされています。各チームはさまざまな組織単位に属する可能性があり、そのそれぞれに独自の収益の流れがあります。リソースコストの帰属先をワークロード、各組織、製品オーナーのいずれかに割り当てると、リソースを効率的に使用し、無駄を削減できます。コストと使用量を正確にモニタリングすることで、各組織単位や製品の収益性が把握できるようになり、より確かな情報に基づいて組織内のリソース配分を決定できます。使用量が変化するとコストも変動するため、組織内のあらゆるレベルの使用量を認識することは、変化を促進する鍵となります。

使用量と支出を認識するために、多面的なアプローチを取ることを検討してください。チームは、データを収集、分析し、それに続いて報告する必要があります。考慮すべき主な要因は以下のとおりです。

トピック

- ガバナンス
- コストと使用量のモニタリング
- リソースを削除する

ガバナンス

クラウドのコストを管理するには以下のガバナンス領域を通じて使用量を管理する必要があります。

ベストプラクティス

- COST02-BP01 組織の要件に基づいてポリシーを策定する
- COST02-BP02 目標およびターゲットを策定する
- COST02-BP03 アカウント構造を実装する
- COST02-BP04 グループとロールを実装する
- COST02-BP05 コストコントロールを実装する
- COST02-BP06 プロジェクトのライフサイクルを追跡する

COST02-BP01 組織の要件に基づいてポリシーを策定する

組織によるリソースの管理方法を定義するポリシーを策定し、定期的に検査します。ポリシーでは、リソースのライフタイム全体にわたる作成、変更、廃止を含む、リソースとワークロードのコスト面をカバーする必要があります。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

組織のコストおよびコスト要因を把握することは、コストと使用量を効果的に管理して、コスト削減の機会を特定するうえできわめて重要です。組織では一般に、複数のワークロードが複数のチームによってオペレーションされています。各チームはさまざまな組織単位に属する可能性があり、そのそれぞれに独自の収益の流れがあります。リソースのコストをワークロード、それぞれの組織、製品オーナーに帰属させることができると、リソースを効率的に使用し、無駄を削減できます。コストと使用量を正確にモニタリングすることで、ワークロードがどの程度最適化されているか、また組織単位や製品の収益性がどの程度であるかを理解するのに役立ちます。この知識により、組織内のどこにリソースを割り当てるかについて、より多くの情報に基づいた意思決定が可能になります。使用量が変化するとコストも変動するため、組織内のあらゆるレベルの使用量を認識することは、変化を促進する鍵となります。使用量と支出を認識するために、多面的なアプローチを取ることを検討してください。

ガバナンスを実行するための最初のステップは、組織の要件を使用して、クラウド使用に関するポリシーを策定することです。ポリシーでは、組織がクラウドをどのように使用するかや、リソースをどのように管理するかを定義します。ポリシーではコストや使用量に関係するリソースとワークロードのあらゆる局面、つまりリソースのライフタイム全体にわたる作成、変更、廃止をカバーする必要があります。クラウド環境でのあらゆる変更について、ポリシーと手順の遵守と実装を徹底してください。IT の変更管理会議では、質問を提起して、計画された変更によるコストへの影響 (増加または減少)、ビジネスの正当性、期待される結果を確認します。

ポリシーを簡単に理解し、組織全体で効果的に実装するには、シンプルなものにする必要があります。また、ポリシーは、(使用されるように) 遵守と解釈が容易で、(チーム間で誤解が生じないように) 具体的である必要があります。さらに、お客様のビジネス状況や優先順位が変わるとポリシーが古くなるため、(当社のメカニズムと同様に) 定期的に検査し、更新する必要があります。

使用する地理的リージョンやリソースを稼働する時間帯など、大局的な幅広いポリシーから始めます。続いてポリシーを徐々に絞り込み、さまざまな組織単位やワークロードに対応させます。一般的なポリシーの例としては、どのサービスと機能を利用できるか (例えば、テスト環境や開発環境では低パフォーマンスのストレージ)、どのタイプのリソースを各グループで使用できるか (例えば、開発

用アカウントのリソースの最大サイズをミディアムにする)、これらのリソースをどの程度のスパンで使用するか (一時的、短期、特定の期間)、などがあります。

ポリシーの例

次に、コスト最適化に焦点を当てた独自のクラウドガバナンスポリシーを作成するために確認できるポリシーの例を示します。組織の要件と関係者の要求に基づいてポリシーを調整してください。

- ポリシー名: リソースの最適化やコスト削減ポリシーなど、明確なポリシー名を定義します。
- 目的: このポリシーを使用すべき理由と、期待される成果について説明します。このポリシーの目的は、ビジネス要件を満たすために必要なワークロードのデプロイと実行に、必要な最小限のコストがあると確認することです。
- 範囲: このポリシーを誰が使用すべきか、いつ使用すべきかを明確に定義します。例えば、DevOps X Team が X 環境 (本番環境または非本番環境) で us-east のお客様にこのポリシーを使用する、などです。

ポリシーステートメント

1. ワークロードの環境とビジネス要件 (開発、ユーザー受け入れテスト、本番稼働前、または本番稼働) に基づいて、us-east-1 または複数の us-east リージョンを選択します。
2. Amazon EC2 と Amazon RDS インスタンスが、午前 6 時から夜 8 時 (東部標準時 (EST)) に実行されるようにスケジュールを立てます。
3. 8 時間後に未使用の Amazon EC2 インスタンスをすべて停止し、24 時間非アクティブ状態の未使用の Amazon RDS インスタンスをすべて停止します。
4. 非本番環境で非アクティブ状態が 24 時間続いたら、未使用の Amazon EC2 インスタンスをすべて終了します。Amazon EC2 インスタンス所有者に (タグに基づいて) 本番環境で停止した Amazon EC2 インスタンスを確認するように促し、使用していない場合は Amazon EC2 インスタンスが 72 時間以内に終了することを通知します。
5. m5.large などの汎用インスタンスファミリーとサイズを使用し、AWS Compute Optimizer を使用して CPU とメモリの使用率に基づいてインスタンスのサイズを変更します。
6. 自動スケーリングの使用を優先し、トラフィックに基づいて実行中のインスタンスの数を動的に調整します。
7. 重要ではないワークロードにはスポットインスタンスを使用します。
8. キャパシティ要件を確認し、予測可能なワークロードに備えて削減プランやリザーブドインスタンスをコミットして、クラウド財務管理チームに通知します。

9. Amazon S3 ライフサイクルポリシーを使用して、アクセス頻度の低いデータを安価なストレージ階層に移動します。保存ポリシーが定義されていない場合は、Amazon S3 Intelligent-Tiering を使用してオブジェクトをアーカイブ階層に自動的に移動します。
10. Amazon CloudWatch を使用して、リソースの使用状況をモニタリングし、スケーリングイベントをトリガーするアラームを設定します。
11. それぞれの AWS アカウントについて、AWS Budgets を使用して、コストセンターとビジネスユニットに基づいてアカウントのコストと使用量の予算を設定します。
12. AWS Budgets を使用してアカウントのコストと使用量の予算を設定すると、支出を常に把握し、予期しない請求を回避できるため、コストをより適切に制御できます。

手順: このポリシーを実装するための詳細な手順を提供するか、各ポリシーステートメントの実装方法を説明する他のドキュメントを参照してください。このセクションでは、ポリシー要件を実行するためのステップバイステップの手順を説明する必要があります。

このポリシーを実装するために、さまざまなサードパーティー製ツールや AWS Config ルールを使用してポリシーステートメントの遵守を確認したり、AWS Lambda 関数を使用して自動修復アクションをトリガーしたりできます。AWS Organizations を使用してポリシーを適用することもできます。さらに、リソースの使用状況を定期的に見直し、必要に応じてポリシーを調整して、ビジネスニーズが引き続き満たされていることを確認する必要があります。

実装手順

- 関係者とのミーティングを設ける: ポリシーを策定するには、組織内の関係者 (クラウドビジネスオフィス、エンジニア、またはポリシー実施部門の意思決定者) に、要件を明記して文書化するよう依頼します。幅広く開始し、各ステップで最小単位まで継続的に絞り込んでいくという反復型アプローチを採用します。チームメンバーには、組織単位やアプリケーションの所有者など、ワークロードの直接の関係者に加えて、セキュリティチームや財務チームなどのサポートグループを含めます。
- 確認する: 誰が AWS クラウドにアクセスしてデプロイできるかを指定したポリシーに、チームが同意していることを確認します。チームが組織のポリシーに従っているかどうか、同意したポリシーと手順に沿ってチームがリソースを作成しているかどうかを確認します。
- オンボーディングトレーニングセッションを作成する: 新しい組織メンバーに対し、コスト意識を定着させ、組織の要件を特定するために、オンボーディングトレーニングコースを完了するよう求めます。新しいメンバーは、以前の経験から異なるポリシーを想定している場合や、ポリシーについてまったく考えていない場合があります。

- ワークロードの場所を定義する: ワークロードの運用場所 (国や国内のエリアなど) を定義します。この情報は、AWS リージョンとアベイラビリティーゾーンへのマッピングに使用されます。
- サービスとリソースを定義してグループ化する: ワークロードに必要なサービスを定義します。サービスごとに、タイプ、サイズ、必要なリソースの数を指定します。アプリケーションサーバーやデータベースストレージなどの機能別にリソースのグループを定義します。リソースは複数のグループに属することができます。
- 機能別にユーザーを定義およびグループ化する: ワークロードに関係するユーザーについて、当該ユーザーが誰かまたは組織内での地位に焦点を当てるのではなく、何を行うか、またはどのようにワークロードを使用するかに焦点を当てて定義します。類似するユーザーまたは機能をグループ化します。AWS 管理ポリシーをガイドとして使用できます。
- アクションを定義する: 特定済みの場所、リソース、およびユーザーを使用して、ワークロードのライフタイム (開発、運用、廃止) にわたって成果を得るために、それぞれが必要とするアクションを定義します。各場所で、グループ内の個々の要素ではなく、グループに基づいてアクションを特定します。開始時には読み取りまたは書き込みを幅広く設定し、それぞれのサービスについて、特定のアクションへと絞り込んでいきます。
- レビュー期間を定義する: ワークロードと組織の要件は時間の経過とともに変化する可能性があります。ワークロードのレビュースケジュールを定義して、組織の優先順位に合わせた状態を維持します。
- ポリシーを文書化する: 定義されたポリシーが、必要に応じて組織でアクセス可能であることを確認します。これらのポリシーは、環境へのアクセスを実装、保守、監査するために使用されます。

リソース

関連ドキュメント:

- クラウドにおける変更管理
- ジョブ機能の AWS 管理ポリシー
- AWS 複数アカウントの請求戦略
- AWS のサービスのアクション、リソース、および条件キー
- AWS 管理とガバナンス
- IAM ポリシーを使用して AWS リージョンへのアクセスを制御する
- グローバルインフラストラクチャリージョンと AZ

関連動画:

- AWS での大規模な管理とガバナンス

COST02-BP02 目標およびターゲットを策定する

ワークロードのコストおよび使用量の両方について、目標およびターゲットを策定します。目標は、期待される成果に基づく方向性を組織に示し、ターゲットは、ワークロードについて具体的に達成すべき測定可能な成果を示します。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

組織のコスト、目標使用量、ターゲットを設定します。AWS で成長を続ける組織にとって、コスト最適化の目標を設定して追跡することは重要です。これらの目標や主要業績評価指標 (KPI) には、オンデマンドでの支出の割合や、AWS Graviton インスタンスまたは gp3 EBS ボリュームタイプなどの特定の最適化されたサービスの導入などが含まれます。事業運営において重要な効率向上を測定できる測定可能で達成可能な目標を設定します。目標は、組織に期待される成果に関するガイダンスと方向性をもたらします。

ターゲットは、具体的かつ測定可能な達成すべき成果をもたらします。つまり、目標は進みたい方向を示し、ターゲットはその方向へどこまで進み、その目標をいつ達成する必要があるかを表します。このとき、「SMART」、つまり具体的 (specific)、測定可能 (measurable)、割り当て可能 (assignable)、現実的 (realistic)、タイムリー (timely) であることをガイダンスとします。「プラットフォームの使用量を大幅に増加させ、コストは微増 (非線形) にとどまるようにする」などは目標の例です。「プラットフォームの使用量を 20% 増加させ、コスト増は 5% 未満に抑える」などはターゲットの例です。ワークロードを 6 か月ごとに効率化する必要があるというケースも、目標としてはよくあります。付随する目標は、ビジネスあたりのコストメトリクスを 6 か月ごとに 5% 削減する必要があるというものです。適切なメトリクスを使用し、組織に合わせて計算された KPI を設定してください。基本的な KPI から始めて、後でビジネスニーズに応じて発展させていくことができます。

コスト最適化の目標は、ワークロードの効率を高めることです。つまり、ワークロードのビジネス成果あたりのコストを経時的に削減することです。すべてのワークロードにこの目標を設定するのと併せて、6 か月 ~ 1 年ごとに効率を 5% 高めるなどのターゲットを設定します。クラウドの場合、これは、コスト最適化能力の構築と、新しいサービスや機能のリリースによって達成できます。

ターゲットは、目標達成のために到達を目指す数値化可能なベンチマークであり、このベンチマークによって実際の結果をターゲットと比較します。コンピューティングサービス (スポット導

入、Graviton 導入、最新のインスタンスタイプ、オンデマンドカバレッジなど)、ストレージサービス (EBS GP3 導入、古い EBS スナップショット、Amazon S3 Standard ストレージなど)、またはデータベースサービスの使用量 (RDS オープンソースエンジン、Graviton 導入、オンデマンドカバレッジなど) のユニットあたりのコストについて、KPI を使用してベンチマークを設定します。これらのベンチマークと KPI により、最も費用対効果の高い方法で AWS のサービスを利用していることを確認することができます。

次の表は、参照用の標準の AWS メトリクスの一覧です。これらの KPI の目標値は、組織によって異なります。

カテゴリ	KPI (%)	説明
コンピューティング	EC2 使用量カバレッジ	SP + RI + スポットを使用した EC2 インスタンス (コストまたは時間単位) と EC2 インスタンスの合計 (コストまたは時間単位) の比較
コンピューティング	コンピューティング SP/RI 使用率	利用可能な SP または RI 時間の合計に対する SP または RI の使用時間
コンピューティング	EC2/時間コスト	EC2 コストをその時間に実行されている EC2 インスタンスの数で割った値
コンピューティング	vCPU コスト	すべてのインスタンスの vCPU あたりのコスト
コンピューティング	最新のインスタンス生成	Graviton (またはその他の最新世代のインスタンスタイプ) のインスタンスの割合
データベース	RDS カバレッジ	RI を使用した RDS インスタンス (コストまたは時間単位) と RDS インスタンスの合計 (コストまたは時間単位) の比較

カテゴリ	KPI (%)	説明
データベース	RDS 使用率	利用可能な RI 時間の合計に対する使用された RI 時間
データベース	RDS の稼働時間	RDS コストをその時間に実行されている RDS インスタンスの数で割った値
データベース	最新のインスタンス生成	Graviton (またはその他の最新インスタンスタイプ) のインスタンスの割合
[Storage (ストレージ)]	ストレージの使用率	最適化ストレージコスト (Glacier、ディープアーカイブ、低頻度アクセスなど) を合計ストレージコストで割ったもの
Tagging	タグ付けされていないリソース	Cost Explorer: 1. クレジット、割引、税金、返金、マーケットプレイスを除外し、最新の月額コストをコピーします。 2. Cost Explorer で [タグ付けされていないリソースのみ表示] を選択します。 3. タグ付けされていないリソースの額を月額コストで割ります。

この表を使用して、組織の目標に基づいて算出した目標値またはベンチマーク値を含めます。正確かつ現実的な KPI を定義するには、ビジネスの特定のメトリクスを測定し、そのワークロードのビジネス成果を理解する必要があります。組織内のパフォーマンスメトリクスを評価する場合は、個別の目的を果たすさまざまな種類のメトリクスを識別します。これらのメトリクスは、ビジネス全体への

影響を直接測定するものではなく、技術的なインフラストラクチャのパフォーマンスと効率を主に測定するものです。例えば、サーバーの応答時間、ネットワークレイテンシー、システムの稼働時間などを追跡します。これらのメトリクスは、インフラストラクチャが組織の技術面での運用をどの程度サポートしているかを評価するうえで非常に重要です。ただし、顧客満足度、収益増加、市場シェアなど、より広範なビジネス目標に関する直接的なインサイトは得られません。ビジネスパフォーマンスを包括的に理解するには、ビジネスの成果と直接相関する戦略的なビジネスメトリクスを、これらの効率性メトリクスの補完として使用します。

KPI と関連するコスト削減の機会をほぼリアルタイムで把握し、経時的に進捗状況を追跡します。KPI 目標の定義と追跡を始める際は、クラウドインテリジェンスダッシュボード (CID) の KPI ダッシュボードをお勧めします。KPI ダッシュボードには、コストと使用状況レポート (CUR) から取得したデータに基づいた一連の推奨コスト最適化 KPI が表示され、カスタム目標の設定や経時的な進捗状況の追跡ができます。

KPI 目標を設定して追跡する別のソリューションがある場合は、そのソリューションが組織内のすべてのクラウド財務管理のステークホルダーによって採用されていることを確認してください。

実装手順

- 予想される使用レベルを定義する: まず、使用レベルに焦点を当てます。アプリケーションの所有者、マーケティング、およびより広範のビジネスチームと協力して、ワークロードに対して予想される使用レベルを把握します。顧客の需要が経時的にどのように変化するか、季節的な要因による増加やマーケティングキャンペーンによって何が変化する可能性があるかなどを考慮します。
- ワークロードのリソースとコストを定義する: 使用レベルを定義したうえで、これらの使用レベルを満たすために必要なワークロードリソースの変化を数値化します。ワークロードコンポーネントのサイズまたはリソースの数を増やすこと、データ転送を増やすこと、または特定のレベルでワークロードコンポーネントを別のサービスに変更することが必要な場合があります。こうした主要ポイントごとにコストを特定し、使用量の変化に伴うコストの変化を予測します。
- ビジネス目標を定義する: 予想される使用量とコストの変化から結果を取得し、これを、予想されるテクノロジーや実行中のプログラムの変化と組み合わせて、ワークロードの目標を策定します。目標は、使用量とコスト、および使用量とコストの関係を考慮したものにする必要があります。目標はシンプルかつ大局的なものにして、そのビジネスで求められる成果を従業員が理解できるような内容にする必要があります (未使用のリソースを一定のコストレベル以下に抑えるなど)。未使用リソースのタイプごとに目標を定義したり、目標とターゲットでの損失原因となりうるコストを具体的に定義したりする必要はありません。使用量に変化がない状態でコストの変化が予想される場合は、組織的なプログラム (トレーニングや教育による能力向上など) を用意しておきます。
- ターゲットを定義する: 定義された目標ごとに、測定可能なターゲットを指定します。ワークロードの効率改善が目標である場合、ターゲットでは、改善の量 (通常は 1 USD あたりのビジネス成

果) と、その改善をいつ達成すべきかを数値化します。例えば、過剰プロビジョニングによる無駄を最小限に抑えるという目標を設定したとします。この場合、ターゲットとしては、実稼働ワークロードの最初の階層でコンピューティングの過剰プロビジョニングによる無駄を階層コンピューティングコストの 10% 以下に抑える、さらに 2 つ目のターゲットとして、実稼働ワークロードの 2 つ目の階層でコンピューティングの過剰プロビジョニングによる無駄を階層コンピューティングコストの 5% 以下に抑える、といったような内容が考えられます。

リソース

関連ドキュメント:

- ジョブ機能の AWS 管理ポリシー
- AWS 複数アカウントの請求戦略
- IAM ポリシーを使用して AWS リージョンへのアクセスを制御する
- S.M.A.R.T. 目標
- CID KPI ダッシュボードでコスト最適化 KPI を追跡する方法

関連動画:

- Well-Architected ラボ: 目標とターゲット (レベル 100)

関連する例:

- 単位メトリクスとは
- ビジネスをサポートする単位メトリクスの選択
- 実際の単位メトリクス: 得た教訓
- 単位メトリクスがビジネス機能間の調整にどのように役立つか

COST02-BP03 アカウント構造を実装する

組織にマッピングされるアカウントの構造を実装します。これは、組織全体でのコストの割り当てと管理に役立ちます。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

AWS Organizations では、ワークロードを AWS でスケールする際に環境を一元管理するのに役立つ、複数の AWS アカウントを作成できます。組織単位 (OU) 構造で AWS アカウントをグループ化し、各 OU の下に複数の AWS アカウントを作成することで、組織階層をモデル化できます。アカウント構造を作成するには、まず、どの AWS アカウントを管理アカウントにするかを決定する必要があります。その後、「管理アカウントに関するベストプラクティス」と「メンバーアカウントのベストプラクティス」に従って、設計したアカウント構造に基づき、メンバーアカウントとして新しい AWS アカウントを作成したり、既存のアカウントを選択したりできます。

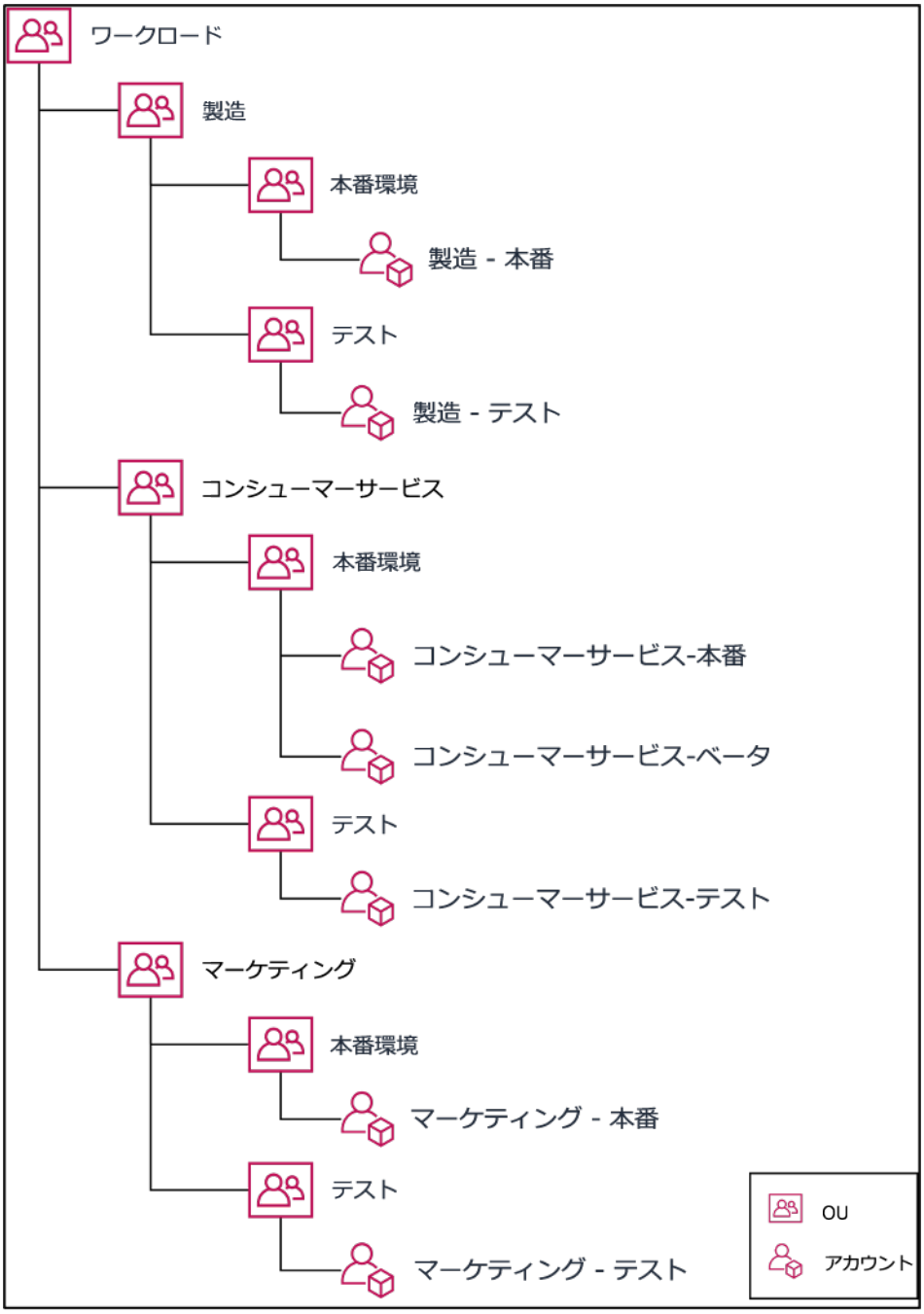
組織の規模や使用状況にかかわらず、少なくとも 1 つの管理アカウントとそれに紐づく 1 つのメンバーアカウントを常に持つことをお勧めします。すべてのワークロードリソースはメンバーアカウント内にのみ存在する必要があります。管理アカウントにはリソースを作成しないでください。AWS アカウントをいくつ持つべきかについて、一律の答えはありません。現在と将来の運用モデルとコストモデルを評価し、AWS アカウントの構造が組織の目標を反映するようにします。ビジネス上の理由から複数の AWS アカウントを作成する企業もあります。次に例を示します。

- 組織単位、コストセンター、特定のワークロード間で、管理、会計、請求の職務機能を切り離す必要がある場合。
- AWS のサービスの制限が特定のワークロードのみに設定される場合。
- ワークロードとリソース間の隔離と分離には要件があります。

AWS Organizations 内では、一括請求により、1 つ以上のメンバーアカウントと管理アカウントとの間に構造が作成されます。メンバーアカウントを使用すると、コストと使用量をグループ別に分離し、区別できます。一般的には、各組織単位 (財務、マーケティング、営業など)、各環境ライフサイクル (開発、テスト、本番など)、各ワークロード (ワークロード a、b、c) にメンバーアカウントをいったん分離したうえで、一括請求を使用してこれらの連結アカウントを集約します。

一括請求機能により、複数のメンバー AWS アカウントの支払いを単一の管理アカウントにまとめつつ、リンクされた各アカウントのアクティビティを可視化することができます。コストと使用量が管理アカウントに集計されると、サービスの従量制割引とコミットメント割引 (Savings Plans とリザーブドインスタンス) を最大限に活用し、割引額を最大化できます。

次の図は、AWS Organizations を組織単位 (OU) で使用して複数のアカウントをグループ化し、各 OU の下に複数の AWS アカウントを配置する方法を示しています。アカウントを整理するためのパターンを提供するために、さまざまなユースケースやワークロードに OU を使用することをお勧めします。



組織単位の下に複数の AWS アカウントをグループ化する例。

AWS Control Tower では、複数の AWS アカウントのセットアップと構成をすばやく行い、ガバナンスが組織の要件に適合していることを確認できます。

実装手順

- 分離要件を定義する: 分離の要件は、セキュリティ、信頼性、財務構造など、複数の要因の組み合わせです。各要因を順番に確認し、ワークロードまたはワークロード環境を他のワークロードから

分離するかどうかを指定します。セキュリティは、アクセス要件とデータ要件への準拠を促進します。信頼性は、環境やワークロードが他の環境に影響を与えないように制限を管理します。Well-Architected フレームワークのセキュリティと信頼性の柱を定期的に見直し、提供されるベストプラクティスに従います。財務構造により、厳格な財務分離 (異なるコストセンター、ワークロードのオーナーシップ、説明責任) が実現します。分離の一般的な例としては、実稼働ワークロードとテストワークロードを別々のアカウントで実行することや、組織内の個々の事業部門や部署、またはアカウントを所有する関係者に請求書と請求データを提供できるように別のアカウントを使用することなどが挙げられます。

- グループ化要件を定義する: グループ化要件は分離要件を上書きしませんが、管理を支援するために使用されます。分離を必要としない同様の環境またはワークロードをグループ化します。例として、1 つ以上のワークロードから複数のテスト環境または開発環境をグループ化することが挙げられます。
- アカウント構造を定義する: これらの分離およびグループ化を使用して、各グループのアカウントを指定し、分離要件が維持されるようにします。これらのアカウントは、メンバーアカウントまたは連結アカウントです。これらのメンバーアカウントを単一の管理アカウントまたは支払者アカウントでグループ化することで使用量が合算されるため、すべてのアカウントでの従量制割引がより大きくなり、すべてのアカウントに対して単一の請求書が発行されます。請求データを分離し、各メンバーアカウントに請求データの個別のビューを表示することができます。メンバーアカウントが使用量や請求データを他のアカウントに表示してはならない場合、または AWS から別々の請求書を必要とする場合は、複数の管理アカウントまたは支払者アカウントを定義します。この場合、各メンバーアカウントは独自の管理アカウントまたは支払者アカウントを持つことになります。リソースは常にメンバーアカウントまたは連結アカウントに配置する必要があります。管理アカウントまたは支払者アカウントは、管理のためにのみ使用してください。

リソース

関連ドキュメント:

- コスト配分タグの使用
- ジョブ機能の AWS 管理ポリシー
- AWS 複数アカウントの請求戦略
- IAM ポリシーを使用して AWS リージョンへのアクセスを制御する
- AWS Control Tower
- AWS Organizations
- 管理アカウントとメンバーアカウントのベストプラクティス

- 複数のアカウントで AWS 環境を構成する
- 共有リザーブドインスタンスと Savings Plans の割引の有効化
- 一括請求
- 一括請求

関連する例:

- CUR の分割とアクセスの共有

関連動画:

- AWS Organizations のご紹介
- AWS Organizations のベストプラクティスを使用するマルチアカウント AWS 環境を設定する

関連する例:

- 通信会社の AWS マルチアカウント戦略の定義
- AWS アカウントを最適化するためのベストプラクティス
- AWS Organizations での組織単位のベストプラクティス

COST02-BP04 グループとロールを実装する

ポリシーに沿ったグループおよびロールを実装し、各グループのインスタンスおよびリソースを作成、変更、廃止できるユーザーを管理します。例えば、開発、テスト、本番グループを実装します。これは、AWS のサービスやサードパーティーのソリューションに適用されます。

このベストプラクティスを活用しない場合のリスクレベル: 低

実装のガイダンス

ユーザーのロールとグループは、安全で効率的なシステムを設計および実装するうえで基本となる構成要素です。ロールとグループは、組織が統制の必要性と柔軟性や生産性の要件とのバランスを図るうえで役立ち、最終的には組織の目標とユーザーのニーズの実現を助けます。AWS Well-Architected フレームワークの「セキュリティの柱」の「ID とアクセス管理」セクションで推奨されているように、適切な条件下で適切なリソースへのアクセスを提供するために、堅牢な ID 管理とアクセス許可

が必要です。ユーザーには、それぞれの業務を完遂するために必要なアクセス権のみを与えます。そうすることで、不正アクセスや誤用に伴うリスクが最小限に抑えられます。

ポリシーを作成した後で、組織内の論理グループとユーザーロールを作成できます。アクセス許可を割り当て、使用量を制御できるようになり、強固なアクセス制御メカニズムの実装を助け、機密情報への不正アクセスも防止できます。人材のおおまかなグループ化から始めます。通常これは、組織単位と役職 (IT 部門のシステム管理者、会計監査担当者、ビジネスアナリストなど) と合致します。グループによって、類似したタスクに従事し、類似したアクセス権を必要とするユーザーを分類します。ロールとは、グループとして義務付けられた仕事の定義を指します。個々のユーザー単位ではなく、グループやロール単位でアクセス許可を管理する方が簡単です。ロールとグループを通じてユーザー全員に一貫して体系的にアクセス許可を割り当てることで、ミスや不整合を防ぐことができます。

ユーザーのロールが変更された場合、管理者は個々のユーザーアカウントを設定し直さなくても、ロールまたはグループのレベルでアクセス権を調整できます。例えば、IT のシステム管理者はすべてのリソースを作成するためのアクセスが必要ですが、分析チームのメンバーは分析リソースを作成するアクセスのみで十分です。

実装手順

- グループを実装する: 必要に応じて、組織のポリシーで定義されているユーザーのグループを使用して、対応するグループを実装します。ユーザー、グループ、認証のベストプラクティスについては、AWS Well-Architected フレームワークの「セキュリティの柱」を参照してください。
- ロールとポリシーを実装する: 組織のポリシーで定義されているアクションを使用して、必要なロールとアクセスポリシーを作成します。ロールとポリシーのベストプラクティスについては、AWS Well-Architected フレームワークの「セキュリティの柱」を参照してください。

リソース

関連ドキュメント:

- ジョブ機能の AWS 管理ポリシー
- AWS 複数アカウントの請求戦略
- セキュリティの柱 - AWS Well-Architected Framework
- AWS Identity and Access Management (IAM)
- AWS Identity and Access Management ポリシー

関連動画:

- アイデンティティ管理とアクセス管理を使用する理由

関連する例:

- IAM ポリシーを使用して AWS リージョンへのアクセスを制御する
- クラウド財務管理ジャーニーの開始: クラウドコストオペレーション

COST02-BP05 コストコントロールを実装する

組織のポリシーと定義済みのグループおよびロールに基づいてコントロールを実装します。これらは、リージョンやリソースタイプへのアクセスコントロールなど、組織の要件によって定義されたコストのみが発生することを保証するものです。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

コスト管理を導入する際の一般的な最初のステップは、ポリシー外のコストまたは使用状況イベントが発生した場合に通知するように設定することです。ワークロードや新しいアクティビティを制限したり悪影響を与えたりすることなく、迅速に行動し、是正措置の必要性の有無を確認できます。ワークロードと環境の制限を理解したら、ガバナンスを適用できます。AWS Budgets では、AWS コスト、使用量、コミットメント割引 (Savings Plans とリザーブドインスタンス) の通知を設定し、月額予算を定義できます。予算は、集計コストのレベル (例えば、全コスト)、またはリンクアカウント、サービス、タグ、アベイラビリティーゾーンなどの特定のディメンションのみを含む詳細レベルで作成できます。

AWS Budgets で予算制限を設定したら、AWS Cost Anomaly Detection を使用して予期しないコストを削減します。AWS Cost Anomaly Detection は、機械学習を使用してコストと使用状況を継続的にモニタリングし、異常な支出を検出するコスト管理サービスです。これにより、異常な支出と根本原因を特定するため、迅速に対策を講じることができます。まず、AWS Cost Anomaly Detection でコストモニタを作成し、ドルのしきい値 (影響が 1,000 USD を超える異常に対してアラートを出すなど) を設定し、アラートの設定を選択します。アラートを受信したら、異常の背後にある根本原因とコストへの影響を分析できます。また、AWS Cost Explorer では独自の異常解析を監視および実行することもできます。

AWS Identity and Access Management および AWS Organizations サービスコントロールポリシー (SCP) を使用して AWS にガバナンスポリシーを適用します。IAM により、AWS のサービスとリ

ソースへのアクセスを安全に管理できます。IAM を使用すると、AWS のリソースを作成または管理できるユーザー、作成できるリソースのタイプ、リソースを作成できる場所を制御できます。これにより、定義されたポリシーの範囲を超えてリソースが作成される可能性が最小限に抑えられます。以前に作成したロールとグループを使用し、IAM ポリシーを割り当てて正しい使用法を適用します。SCP は、組織内のすべてのアカウントで利用可能な最大権限を一元管理し、アカウントをアクセス制御ガイドラインの範囲内に維持することができます。SCP はすべての機能が有効になっている組織でのみ使用可能で、デフォルトで SCP によるメンバーアカウントのアクションの可否を設定できます。アクセス管理の実装の詳細については、Well-Architected のセキュリティの柱についてのホワイトペーパーを参照してください。

AWS Service Quotas を管理することで、ガバナンスを導入することもできます。Service Quotas を最小限のオーバーヘッドで設定し、正確に維持することで、組織の要件以外のリソースの作成を最小限に抑えることができます。これを実現するには、要件がどれだけ速く変化するかを理解し、進行中のプロジェクト (リソースの作成と廃止の両方) を理解し、クォータ変更をどれだけすばやく実装できるかを考慮する必要があります。Service Quotas を使用して、必要に応じてクォータを増加させることができます。

実装手順

- 支出に関する通知を実装する: 定義した組織のポリシーを使用して、AWS Budgets を作成し、支出がポリシーを外れた場合に通知を提供するようにします。アカウントごとに複数のコスト予算を設定し、アカウント全体の支出を通知します。アカウント内のより小さな単位について、各アカウント内にコスト予算を追加で設定します。これらの単位は、アカウント構造によって異なります。一般的な例としては、AWS リージョン、ワークロード (タグを使用)、または AWS のサービスがあります。個人の E メールアカウントではなく、E メール配信リストを通知の受信者として設定します。金額を超えたときの実際の予算を設定するか、予測された使用量が通知されたときの予測された予算を使用します。特定の IAM または SCP のポリシーを適用したり、ターゲットの Amazon EC2 または Amazon RDS インスタンスを停止したりできる AWS Budget アクションを事前設定することもできます。予算に関するアクションは、自動的に開始するか、ワークフローの承認を得るようにすることができます。
- 異常な支出に関する通知を実装する: AWS Cost Anomaly Detection を使用して、組織内の予想外のコストを削減し、異常とみられる支出の根本原因を分析します。指定した粒度で異常な支出を特定するコストモニタを作成し、AWS Cost Anomaly Detection で通知を設定すると、異常な支出が検出された際にアラートが送信されます。これにより、異常の背後にある根本的な原因を分析し、コストへの影響を理解できます。AWS Cost Anomaly Detection を設定する際に AWS Cost Categories を使用して、予想外のコストの根本原因を分析し、必要なアクションをタイムリーに実行できるプロジェクトチームまたはビジネスユニットチームを特定します。

- 使用量のコントロールを実装する: 定義した組織のポリシーを使用して、IAM ポリシーとロールを実装し、ユーザーが実行できるアクションと実行できないアクションを指定します。AWS ポリシーには、複数の組織ポリシーを含めることができます。ポリシーを定義するのと同じ方法で、幅広く開始し、各ステップでより詳細なコントロールを適用します。サービスの制限も、使用量に対する効果的なコントロールです。すべてのアカウントに正しいサービス制限を実装します。

リソース

関連ドキュメント:

- ジョブ機能の AWS 管理ポリシー
- AWS 複数アカウントの請求戦略
- IAM ポリシーを使用して AWS リージョンへのアクセスを制御する
- AWS Budgets
- AWS Cost Anomaly Detection
- AWS コストを管理する

関連動画:

- AWS Budgets を使用して支出と使用量を追跡するにはどうすればよいですか?

関連する例:

- IAM アクセス管理ポリシーの例
- サービスコントロールポリシーの例
- AWS 予算アクション
- タグを使用して Amazon EC2 リソースへのアクセスを制御する IAM ポリシーを作成する方法を教えてください。
- IAM アイデンティティのアクセスを特定の Amazon EC2 リソースに制限することはできますか?
- Slack でのコスト異常検出のための Amazon Q Developer in chat applications との統合

COST02-BP06 プロジェクトのライフサイクルを追跡する

プロジェクト、チーム、環境のライフサイクルを追跡、計測、監査して、不要なリソースの使用やそれに伴う支払いを回避できます。

このベストプラクティスを活用しない場合のリスクレベル: 低

実装のガイダンス

プロジェクトのライフサイクルを効果的に追跡することで、組織は計画、管理、リソースの最適化を改善し、コスト管理を強化できます。追跡で得られたインサイトは、意思決定に役立つ貴重な情報となり、コスト効率性やプロジェクト全体の成功に寄与します。

ワークロードのライフサイクル全体を追跡すれば、ワークロードやワークロードコンポーネントが不要になった時点でわかります。既存のワークロードとコンポーネントが使用中のように見える場合がありますが、AWS が新しいサービスや機能をリリースした時点で、廃止または刷新される可能性があります。ワークロードの以前のステージに注目してください。ワークロードが本番稼働状態になったら、以前の環境は廃止するか、再び必要になるまでキャパシティを大幅に削減することができます。

リソースに期間やリマインダーのタグを付けて、ワークロードがレビューされた時点をマークしておくことができます。例えば、開発環境の前回のレビューから数か月経っている場合は、再度レビューを行って、新しいサービスを導入できるか、環境が使用中かを調査する適切なタイミングである可能性があります。アプリケーションを AWS の myApplications でグループ化してタグ付けし、重要度、環境、最終レビュー、コストセンターなどのメタデータを管理および追跡できます。ワークロードのライフサイクルを追跡すると共に、アプリケーションのコスト、状態、セキュリティ体制、パフォーマンスをモニタリングおよび管理できます。

AWS には、エンティティのライフサイクル追跡に使用できるさまざまな管理およびガバナンスサービスが用意されています。AWS Config または AWS Systems Manager を使用して、AWS リソースと設定の詳細なインベントリを入手できます。プロジェクトやアセットを管理する既存のシステムを統合して、組織内のアクティブなプロジェクトや製品を追跡することが推奨されます。現在のシステムを AWS が提供する豊富なイベントやメトリクスと組み合わせることにより、重要なライフサイクルイベントのビューを作成し、前もってリソースを管理し、不要なコストを削減できます。

アプリケーションライフサイクル管理 (ALM) と同様に、プロジェクトのライフサイクルを追跡するには、設計と開発、テスト、本番稼働、サポート、ワークロードの冗長性など、複数のプロセス、ツール、チームが連携する必要があります。

プロジェクトのライフサイクルの各段階を注意深く監視することで、組織は重要なインサイトを得て管理を強化し、プロジェクトを計画から実施、完遂に至るまで円滑に進めることができます。入念な監視下で、プロジェクトは品質基準を満たすだけでなく、納期どおりに予算内で完了し、全体的なコスト効率が向上します。

エンティティライフサイクル追跡の実装の詳細については、AWS Well-Architected 運用上の優秀性の柱についてのホワイトペーパーを参照してください。

実装手順

- プロジェクトのライフサイクルモニタリングプロセスを確立する: Cloud Center of Excellence チームは、プロジェクトのライフサイクルモニタリングプロセスを確立する必要があります。ワークロードを監視するための構造的かつ体系的なアプローチを確立し、プロジェクトの管理、可視性、パフォーマンスを高めます。監視プロセスの効果と価値を最大限に引き出すために、プロセスの透明性と協調性を高め、継続的に改善していきます。
- ワークロードレビューを実行する: 組織のポリシーに従い、定期的に既存のプロジェクトを監査し、ワークロードレビューを実施します。監査に費やされる労力の量は、組織のおおよそのリスク、価値、またはコストに比例する必要があります。監査に含めるべき主な領域は、インシデントまたは機能停止の組織に対するリスク、価値、組織への寄与 (収益またはブランドに対する評価で測定)、ワークロードのコスト (リソースおよび運用の合計コストとして測定)、およびワークロードの使用量 (時間単位ごとの組織の成果の数で測定) です。これらの領域がライフサイクルを通じて変化する場合、完全または部分的な廃止など、ワークロードの調整が必要です。

リソース

関連ドキュメント:

- AWS でのタグ付けのガイダンス
- ALM (アプリケーションライフサイクル管理) とは何ですか?
- ジョブ機能の AWS 管理ポリシー

関連する例:

- IAM ポリシーを使用して AWS リージョンへのアクセスを制御する

関連ツール

- AWS Config
- AWS Systems Manager
- AWS Budgets
- AWS Organizations

- AWS CloudFormation

コストと使用量のモニタリング

ワークロードに詳細な可視化を導入し、チームがコストと使用量に対しアクションを実行できるようにします。コストの最適化は、コストと使用状況の内訳、将来の支出、使用状況、機能をモデル化して予測する機能、コストと使用量を組織の目標に合わせて調整するためのメカニズムの実装をきめ細かく理解することから始まります。コストと使用量をモニタリングするために必要な領域は次のとおりです。

ベストプラクティス

- COST03-BP01 詳細情報ソースを設定する
- COST03-BP02 コストと使用状況に組織情報を追加する
- COST03-BP03 コスト属性カテゴリを特定する
- COST03-BP04 組織のメトリクスを確立する
- COST03-BP05 請求およびコスト管理ツールを設定する
- COST03-BP06 ワークロードメトリクスに基づいてコストを配分する

COST03-BP01 詳細情報ソースを設定する

コスト管理ツールとレポートツールを設定して、コストと使用状況に関するデータの分析と透明性を改善します。コストと使用量の追跡と区別を容易にするログエントリを作成するようにワークロードを設定します。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

時間単位の粒度など、コスト管理ツールの詳細な請求情報により、組織は消費量をさらに詳細に追跡でき、コスト増加の原因を特定する手助けとなります。これらのデータソースは、組織全体のコストと使用量の最も正確なビューを提供します。

AWS Data Exports を使用して AWS Cost and Usage Report (CUR) 2.0 のエクスポートを作成できます。これは、AWS から詳細なコストと使用状況の詳細を取得するための新しい推奨方法です。これにより、課金されるすべての AWS のサービスについて、日単位または時間単位の精度による使用量、レート、コスト、使用属性 (CUR と同じ情報) が提供されます。また、いくつかの改善点も提示

されます。CUR では、タグ付け、場所、リソース属性、アカウント ID など想定可能なあらゆる側面からレポートを作成できます。

作成するエクスポートのタイプとして、標準データエクスポート、QuickSight 統合によるコストと使用状況ダッシュボードへのエクスポート、レガシーデータエクスポートの 3 種類のエクスポートタイプがあります。

- [標準データエクスポート]: Amazon S3 に定期的に配信されるテーブルのカスタマイズされたエクスポート。
- [コストと使用状況ダッシュボード]: QuickSight へのエクスポートと統合により、事前構築済みのコストと使用状況ダッシュボードをデプロイします。
- [レガシーデータのエクスポート]: レガシー AWS Cost and Usage Report (CUR) のエクスポートです。

次のカスタマイズを行ったデータエクスポートを作成できます。

- リソース ID の包含
- 分割コスト配分データ
- 時間単位の詳細
- バージョニング
- 圧縮タイプとファイル形式

Amazon ECS または Amazon EKS でコンテナを実行するワークロードの場合、分割コスト配分データを有効にすると、コンテナワークロードによる共有コンピューティングリソースとメモリリソースの消費状況に基づいて、個々のビジネスユニットやチームにコンテナコストを配分できます。分割コスト配分データにより、新しいコンテナレベルのリソースのコストと使用状況データが AWS Cost and Usage Report に導入されます。分割コスト配分データは、クラスターで実行されている個々の ECS サービスとタスクのコストを計算することで算出されます。

コストと使用状況ダッシュボードは、コストと使用状況ダッシュボードテーブルを定期的に S3 バケットにエクスポートし、事前構築済みのコストと使用状況ダッシュボードを QuickSight にデプロイします。コストと使用状況データのダッシュボードをすぐにデプロイしたい場合は、このオプションを使用します (カスタマイズはできません)。

必要に応じて、レガシーモードで CUR をエクスポートできます。AWS Glue など他の処理サービスを統合して分析用にデータを準備して、SQL でデータをクエリして Amazon Athena でデータを分析したりできます。

実装手順

- データエクスポートを作成する: 必要なデータを使用してカスタマイズされたエクスポートを作成し、エクスポートのスキーマを制御します。基本的な SQL を使用して請求とコスト管理データのエクスポートを作成し、QuickSight と連携して請求とコスト管理データを可視化します。また、標準モードでデータをエクスポートして、Amazon Athena などの他の処理ツールでデータを分析することもできます。
- コストと使用状況レポートを設定する: 請求コンソールを使用して、少なくとも 1 つのコストと使用状況レポートを設定します。すべての識別子とリソース ID を含む時間単位の粒度でレポートを設定します。粒度が異なる他のレポートを作成して、概要情報を提供することもできます。
- Cost Explorer で時間単位の詳細度を設定する: 過去 14 日間の時間単位の粒度でコストと使用状況データにアクセスするには、請求コンソールで時間単位とリソースレベルのデータを有効にすることを検討してください。
- アプリケーションログ記録を有効にする: アプリケーションがもたらすビジネスの各成果がログに記録され、追跡および測定が可能であることを確認します。このデータの粒度が少なくとも 1 時間単位であることを確認し、コストと使用状況のデータと一致するようにします。ログ記録とモニタリングの詳細については、Well-Architected 運用上の優秀性の柱についてのホワイトペーパーを参照してください。

リソース

関連ドキュメント:

- AWS Data Exports
- AWS Glue
- QuickSight
- AWS コスト管理の料金
- AWS リソースのタグ付け
- Cost Explorer によるコストの分析
- AWS Cost and Usage Reportの管理
- Well-Architected 運用上の優秀性の柱

関連する例:

- AWS アカウントのセットアップ

- AWS Billing and Cost Management のデータエクスポート
- AWS Cost Explorer の一般的なユースケース

COST03-BP02 コストと使用状況に組織情報を追加する

組織、ワークロード属性、およびコスト配分カテゴリに基づいてタグ付けスキーマを定義します。これによりコスト管理ツールで、フィルター処理によるリソースの検索や、コストおよび使用状況のモニタリングを行うことができます。目的、チーム、環境、またはビジネスに関連するその他の基準によって、可能な限りすべてのリソースに一貫したタグ付けを実装します。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

AWS でタグ付けを実装して、リソースに組織の情報を追加します。追加した情報は、コストと使用状況の情報に追加されます。タグはキーと値のペアです。キーは組織全体で一意になるように定義されている必要があります。値はリソースのグループに対して一意になります。キーと値のペアの一例としては、キーが `Environment` で、値は `Production` となります。本稼働環境のすべてのリソースには、キーと値のペアがあります。タグ付けにより、関連性の高い組織情報を使用して、コストを分類、追跡できます。組織のカテゴリ (コストセンター、アプリケーション名、プロジェクト、オーナーなど) を表すタグを適用し、ワークロードやワークロードの特性 (テストや本番など) を識別して、組織全体のコストと使用状況の帰属先を付与できます。

AWS リソース (Amazon Elastic Compute Cloud インスタンスや Amazon Simple Storage Service バケットなど) にタグを付け、そのタグをアクティブ化すると、AWS はこの情報をコストと使用状況レポートに追加します。タグ付けされたリソースとタグ付けされていないリソースに関するレポート作成および分析を実行することで、社内のコスト管理ポリシーへの準拠を強化し、正確に帰属を特定できます。

組織のアカウント全体に AWS タグ付け基準を作成、導入することで、AWS 環境を一貫性のある統一された方法で管理することができます。タグポリシーを AWS Organizations で使用して、ルールを定義します。ルールは、AWS Organizations のアカウントの AWS リソースに対してタグをどのように使用できるかを定めたものです。タグポリシーを使用すると、AWS リソースにタグを付ける標準アプローチを簡単に導入できます。

AWS タグエディタでは、複数のリソースのタグを追加、削除、管理できます。タグエディタを使用してタグ付けするリソースを検索し、検索結果からそのリソースのタグを管理します。

AWS Cost Categories を使用すると、リソースにタグを付けることなく組織としての意味をコストに割り当てることができます。コストと使用量に関する情報を、一意の内部組織構造にマッピングできます。アカウントやタグなどの請求ディメンションを使用して、コストをマッピングおよび分類するカテゴリルールを定義します。これにより、タグ付けに加えて、より高いレベルの管理機能が提供されます。また、特定のアカウントとタグを複数のプロジェクトにマッピングすることもできます。

実装手順

- タグスキーマを定義する: すべての利害関係者をビジネス全体から集めて、スキーマを定義します。これには通常、技術、財務、および管理ロールの担当者が含まれます。すべてのリソースに必要なタグのリストと、リソースに必要なタグのリストを定義します。タグの名前と値が組織全体で一貫していることを確認します。
- リソースをタグ付けする: 定義したコスト帰属カテゴリを使用して、カテゴリに従ってワークロードのすべてのリソースにタグを付けます。効率を高めるには、CLI、タグエディタ、AWS Systems Manager などのツールを使用します。
- AWS Cost Categories を実装する: タグ付けを実装しなくても Cost Categories を作成できます。Cost Categories では、既存のコストと使用量ディメンションを使用します。スキーマからカテゴリルールを作成し、それをコストカテゴリに実装します。
- タグ付けを自動化する: すべてのリソースにわたってタグ付けの高いレベルを維持していることを確認するには、タグ付けを自動化して、リソースの作成時に自動的にタグ付けされるようにします。AWS CloudFormation などのサービスを使用して、リソースの作成時にタグ付けされていることを確認します。Lambda 関数を使用して自動的にタグ付けするカスタムソリューションや、ワークロードを定期的にスキャンし、タグ付けされていないリソースをすべて削除するマイクロサービスを作成することもできます。これは、テスト環境および開発環境に最適です。
- タグ付けをモニタリング、レポートする: 組織全体でタグ付けの高いレベルを維持していることを確認するには、ワークロード全体でタグをレポートおよびモニタリングします。AWS Cost Explorer を使用して、タグ付けされたリソースとタグ付けされていないリソースのコストを表示したり、タグエディタなどのサービスを使用したりできます。タグ付けされていないリソースの数を定期的に確認し、必要なレベルのタグ付けになるまでタグを追加するアクションを実行します。

リソース

関連ドキュメント:

- タグ付けのベストプラクティス
- AWS CloudFormation リソースタグ

- AWS Cost Categories
- AWS リソースのタグ付け
- AWS Budgets によるコストの分析
- Cost Explorer によるコストの分析
- AWS コストと使用状況レポートの管理

関連動画:

- コストセンターまたはプロジェクトによる請求を分割するための AWS リソースをどのようにタグ付けすればよいか教えてください
- AWS リソースのタグ付け

COST03-BP03 コスト属性カテゴリを特定する

組織内のコストを内部消費エンティティに配分するために使用できるビジネスユニット、部門、プロジェクトなどの組織カテゴリを特定します。こうしたカテゴリを活用して、支出の説明責任の徹底、コスト意識の向上、効果的な消費行動の促進を図ります。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

コストを分類するプロセスは、予算編成、会計、財務報告、意思決定、ベンチマーキング、およびプロジェクト管理においてきわめて重要です。費用を分類してカテゴリ化することで、チームはクラウドジャーニーで発生するコストの種類をよりよく理解でき、情報に基づいた意思決定を行い、予算を効果的に管理できるようになります。

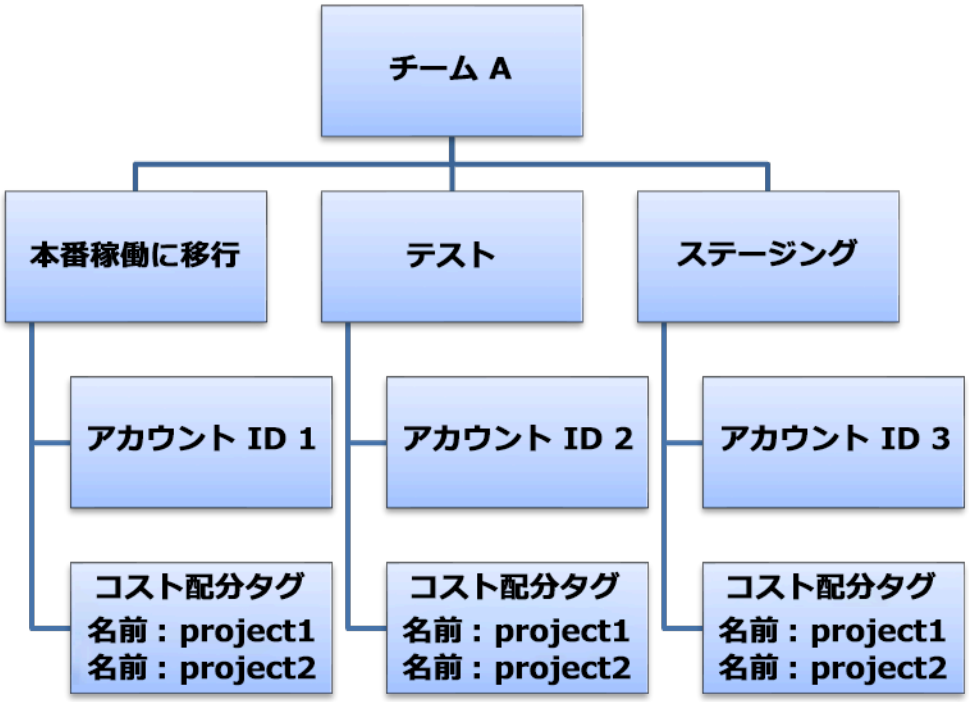
クラウド支出の説明責任は、統制の取れた需要とコスト管理に対する強力なインセンティブを確立します。その結果、クラウド支出の大部分を消費するビジネスユニットやチームに割り当てている組織では、クラウドコストを大幅に節約できます。また、クラウド支出を配分することで、組織は一元化されたクラウドガバナンスのベストプラクティスをさらに採用できるようになります。

定期的なミーティングで、財務チームやその他の関係者と協力し、組織内でコストを配分する方法の要件を理解します。ワークロードのコストは、開発、テスト、本稼働、廃止などライフサイクル全体にわたって配分する必要があります。学習、スタッフ育成、アイデア創出に要したコストが、どのように組織に帰属するかを理解します。これは、この目的で使用される金額を、一般的な IT コスト予算ではなく、トレーニング予算や開発の予算に正しく割り当てるうえで役立ちます。

組織内のステークホルダーとコスト属性カテゴリを定義したら、AWS Cost Categories を使用して、コストと使用状況の情報を特定のプロジェクトのコストや部門やビジネスユニットの AWS アカウントなど、AWS クラウドの有意義なカテゴリにグループ化します。カスタムカテゴリを作成して、アカウント、タグ、サービス、料金タイプなどのさまざまなディメンションを使用して定義したルールに基づき、コストと使用状況の情報をカスタムカテゴリにマッピングすることもできます。コストカテゴリを設定すると、カテゴリごとにコストと使用状況の情報を確認できるようになり、組織の戦略や購入に関する決定をより適切に行うことができます。これらのカテゴリは、AWS Cost Explorer、AWS Budgets、および AWS Cost and Usage Report にも表示されます。

例えば、ビジネスユニット (DevOps チーム) のコストカテゴリを作成し、各カテゴリの下に、複数のルール (各サブカテゴリのルール) を作成します。各ルールでは、定義したグループに基づいて、複数のディメンション (AWS アカウント、コスト配分タグ、サービス、料金タイプ) を使用します。Cost Categories を使うと、ルールベースのエンジンを使用してコストを分類できます。ルールを設定することで、コストをカテゴリ別に分類します。ルール内では、特定の AWS アカウント、AWS サービス、料金タイプなどの各カテゴリについて、複数のディメンションを使用してフィルター処理を行うことができます。これらのカテゴリは、AWS Billing and Cost Management コンソールで使用できます。これには AWS Cost Explorer、AWS Budgets、AWS Cost and Usage Report、および AWS Cost Anomaly Detection があります。

例として、次の図は、組織でコストと使用状況の情報をグループ化する方法を示しています。例えば、複数のチーム (コストカテゴリ)、複数の環境 (ルール)、そして複数のリソースまたはアセットを持つ各環境 (ディメンション) にグループができます。



コストと使用状況の組織図

コストカテゴリを使用して、コストのグループを作成することもできます。コストカテゴリの作成後 (使用状況レコードの値が更新されるまでに最長で 24 時間かかります)、作成したコストカテゴリは、AWS Cost Explorer、AWS Budgets、AWS Cost and Usage Report、および AWS Cost Anomaly Detection に表示されます。AWS Cost Explorer および AWS Budgets では、コストカテゴリが追加の請求ディメンションとして表示されます。これを使用して、特定のコストカテゴリ値でフィルタリングしたり、コストカテゴリ別にグループ化したりできます。

実装手順

- 組織のカテゴリを定義する: 社内の関係者およびビジネスユニットとミーティングを行い、組織の構造と要件を反映したカテゴリを定義します。これらのカテゴリは、ビジネスユニット、予算、コストセンター、部門など、既存の財務カテゴリの構造に直接マッピングされます。トレーニングや教育など、クラウドがもたらすビジネスの成果を確認します。これらは組織のカテゴリでもあります。
- 機能を反映したカテゴリを定義する: 社内の関係者およびビジネスユニットとミーティングを行い、企業内の機能を反映したカテゴリを定義します。これは、ワークロードまたはアプリケーション名、および実稼働、テスト、開発などの環境のタイプである場合があります。
- AWS Cost Categories を定義する: AWS Cost Categories の使用に対してコストと使用状況情報を整理するコストカテゴリを作成して、AWS コストと使用状況を有意義なカテゴリにマッピングします。同じリソースに複数のカテゴリを割り当てることも、同じリソースを複数の異なるカテゴリに含めることもできるため、必要な数のカテゴリを定義します。これにより、AWS Cost Categories を使用したカテゴリ化された構造内でコストを管理できるようになります。

リソース

関連ドキュメント:

- AWS リソースのタグ付け
- コスト配分タグの使用
- AWS Budgets によるコストの分析
- Cost Explorer によるコストの分析
- AWS Cost and Usage Report の管理
- AWS Cost Categories

- AWS Cost Categories を用いてコストを管理する
- コストカテゴリを作成する
- コストカテゴリのタグ付け
- コストカテゴリ内で料金を分割する
- AWS Cost Categories の機能

関連する例:

- AWS Cost Categories でコストと使用状況のデータを整理する
- AWS Cost Categories を用いてコストを管理する

COST03-BP04 組織のメトリクスを確立する

このワークロード用のメトリクスを組織内で定めます。ワークロードのメトリクスの例として、作成された顧客レポートや顧客に提供されるウェブページが挙げられます。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

ワークロードのアウトプットがビジネスの成功に対してどのように測定されるかを理解します。通常、各ワークロードには、パフォーマンスを示す主な成果の小さな組み合わせがあります。多数のコンポーネントを含む高度なワークロードがある場合は、リストに優先順位を付けるか、各コンポーネントのメトリクスを定義して追跡できます。チームと協力して、どのメトリクスを使用するか理解します。この単位は、ワークロードの効率または各ビジネス成果のコストを把握するために使用されます。

実装手順

- ワークロードの成果を定義する: ビジネスの利害関係者とミーティングをして、ワークロードの成果を定義します。これらは顧客の使用状況の主要な測定指標であり、技術的メトリクスではなく、ビジネスメトリクスである必要があります。ワークロードごとに少数の概要的なメトリクス (5 つ未満) が存在する必要があります。ワークロードが異なるユースケースで複数の成果を生成する場合は、それらを単一のメトリクスにグループ化してください。
- ワークロードコンポーネントの成果を定義する: 必要に応じて、大規模で複雑なワークロードがある場合、または明確に定義された入出力を使用してワークロードをコンポーネント (マイクロサー

ビスなど) に簡単に分割できる場合は、各コンポーネントのメトリクスを定義します。この作業では、コンポーネントの価値とコストを反映する必要があります。最大のコンポーネントから開始し、大きさ順で、最小のコンポーネントまで作業します。

リソース

関連ドキュメント:

- AWS リソースのタグ付け
- AWS Budgets によるコストの分析
- Cost Explorer によるコストの分析
- AWS コストと使用状況レポートの管理

COST03-BP05 請求およびコスト管理ツールを設定する

クラウド支出を管理および最適化するには、組織のポリシーに合ったコスト管理ツールを設定します。これには、コストと使用状況のデータを整理して追跡し、統合された請求とアクセス許可での制御の強化、予算編成と予測を通じた計画の改善、通知またはアラートの受信、リソースと価格の最適化によるコスト削減を行うサービス、ツール、リソースが含まれます。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

確固たる説明責任を確立するには、まずアカウント戦略をコスト配分戦略の一部として検討します。これを正しく行えば、それ以上先に進む必要はないかもしれません。正しく行えない場合、認識の欠如が発生し、さらに問題点が増える可能性があります。

クラウド支出の説明責任を推進するには、コストと使用状況の可視化が可能なツールへのアクセスをユーザーに許可します。AWS では、以下の目的に合わせてすべてのワークロードとチームを設定することをお勧めします。

- 整理: 独自のタグ付け戦略と分類を使用して、コスト配分とガバナンスのベースラインを確立します。AWS Control Tower や AWS Organizations などのツールを使用して複数の AWS アカウントを作成します。サポートされている AWS リソースにタグを付け、組織構造 (ビジネスユニット、部門、プロジェクト) に基づいてわかりやすく分類します。特定のコストセンターのアカウント名にタグ付けし、それを AWS Cost Categories とマッピングして、ビジネスユニットのアカウント

をコストセンターのグループにまとめることで、ビジネスユニットの所有者が複数のアカウントの消費を 1 か所で確認できるようにします。

- アクセス: 組織全体の請求情報を一括請求で追跡します。適切なステークホルダーとビジネスオーナーがアクセスできることを確認します。
- 制御: 適切なガードレールを使用して、効果的なガバナンスメカニズムを構築し、サービスコントロールポリシー (SCP)、タグポリシー、IAM ポリシー、予算アラートを使用する際の想定外のシナリオを回避します。例えば、チームが効果的な制御メカニズムを使用する場合のみ目的のリージョンで推奨リソースを作成できるようにしたり、特定のタグ (cost-center など) がないとリソースを作成できないようにしたりすることができます。
- 現状確認: 現在のコストと使用量を示すダッシュボードを設定します。ダッシュボードはオペレーションダッシュボードと同様に、作業環境内の目に付きやすい場所で使用できるようにする必要があります。データをエクスポートし、AWS Cost Optimization Hub のコストと使用状況ダッシュボードまたは任意のサポート対象製品を使用することで、このような可視性が可能になります。ペルソナごとに別々のダッシュボードを作成しなければならない場合があります。例えば、マネージャーのダッシュボードはエンジニアリングのダッシュボードとは異なる場合があります。
- 通知: コストまたは使用量が定義された制限を超え、AWS Budgets または AWS コスト異常検出で異常が発生した場合に通知します。
- レポート: すべてのコストと使用量の情報を要約します。詳細で帰属先が特定可能なコストデータを使用して、クラウド支出の認識と説明責任の意識を高めます。レポートを使用するチームと関連性があり、推奨事項を含めたレポートを作成します。
- 追跡: 設定された目標またはターゲットに対する現在のコストと使用量を表示します。
- 分析: チームメンバーは、さまざまなフィルター (リソース、アカウント、タグなど) を使用して、時間単位、日単位、または月単位でカスタム分析とディープ分析を実行できます。
- 検査: リソースのデプロイとコスト最適化の機会を最新の状態に保ちます。Amazon CloudWatch、Amazon SNS、または Amazon SES を使用して、組織レベルでのリソースデプロイに関する通知を受け取ります。AWS Trusted Advisor または AWS Compute Optimizer を使用してコスト最適化の推奨事項を確認します。
- トレンドレポート: 指定した期間のコストと使用量の変動を、指定の詳細度で示します。
- 予測: 作成した予測ダッシュボードで、将来の推定コストを示し、リソースの使用量と支出を見積もります。

AWS Cost Optimization Hub を使用して、統合された潜在的なコスト削減の機会を一元的な場所から理解し、Amazon Athena と統合するためのデータエクスポートを作成できます。また、AWS Cost Optimization Hub を使用してコストと使用状況ダッシュボードをデプロイすることもできます。この

ダッシュボードでは、QuickSight を使用してインタラクティブなコスト分析を行ったり、コストに関するインサイトを安全に共有したりできます。

組織に必須のスキルや処理能力がない場合、AWS ProServ、AWS Managed Services (AMS)、または AWS パートナーを利用できます。サードパーティーのツールを利用することもできますが、利用に際しては必ず価値提案を検証するようにしてください。

実装手順

- ツールへのチームベースのアクセスを許可する: アカウントを設定してグループを作成し、必要なコストと使用状況レポート (グループの使用状況に関するもの) へのアクセスを許可します。また、AWS Identity and Access Management を使用して AWS Cost Explorer などのツールへのアクセスを制御します。これらのグループには、アプリケーションを所有または管理するすべてのチームの代表者を含める必要があります。これにより、すべてのチームがコストと使用状況の情報にアクセスして、各自の使用を追跡できるようになります。
- コストタグとカテゴリを整理する: チーム、ビジネスユニット、アプリケーション、環境、プロジェクト全体でコストを整理します。リソースタグを使用して、コスト配分タグごとにコストを整理します。タグ、アカウント、サービスなどを使用してディメンションに基づいて Cost Categories を作成し、コストをマッピングします。
- AWS Budgets を設定する: ワークロードのすべてのアカウントでAWS Budgets を設定します。タグとコストカテゴリを使用して、アカウント全体の支出に対する予算とワークロードに対する予算を設定します。予算額を超えたときや、推定コストが予算を超えるときにアラートを受信するよう、AWS Budgets の通知を設定します。
- AWS コスト異常検出を設定する: AWS コスト異常検出を使用することにより、コストと使用状況をモニタリングし、通常と異なる支出を検出できます。集計レポートでアラートを個別に受信したり、E メールまたは Amazon SNS トピックでアラートを受信したりすることで、異常の根本原因を分析および特定し、コストの増加を引き起こしている要因を特定できます。
- コスト分析ツールを使用する: AWS Cost Explorer をワークロードとアカウントについて設定し、さらに分析を行うためにコストデータを視覚化します。ワークロードのダッシュボードを作成することにより、全体的な支出、ワークロードの主要な使用状況メトリクス、過去のコストデータに基づく将来のコストの予測を追跡できます。
- コスト削減分析ツールを使用する: AWS Cost Optimization Hub を使用して、未使用リソースの削除、適切なサイズ設定、Savings Plans、予約、Compute Optimizer の推奨事項など、カスタマイズされた推奨事項でコスト削減の機会を特定します。
- 高度なツールを設定する: 任意でビジュアルを作成して、インタラクティブな分析やコストインサイトの共有を支援できます。AWS Cost Optimization Hub でデータエクスポートを使用する

と、QuickSight を活用したコストと使用状況ダッシュボードを組織に合わせて作成し、さらなる詳細と粒度が得られます。また、Amazon Athena でデータエクスポートを使用して高度な分析機能を実装することで高度なクエリを実施したり、QuickSight でダッシュボードを作成したりできます。AWS パートナーと協力して、統合されたクラウド請求書のモニタリングと最適化のためのクラウド管理ソリューションを導入できます。

リソース

関連ドキュメント:

- AWS Billing and Cost Management とは
- ベストプラクティスの AWS 環境を確立する
- AWS リソースのタグ付けのベストプラクティス
- AWS リソースのタグ付け
- AWS Cost Categories
- AWS Budgets によるコストの分析
- AWS Cost Explorer によるコストの分析
- AWS データエクスポートとは

関連動画:

- クラウドインテリジェンスダッシュボードのデプロイ
- FinOps またはコスト最適化のメトリクスまたは KPI に関するアラートを受け取る

関連する例:

- QuickSight を活用したコストと使用状況ダッシュボード
- AWS コストと使用状況ガバナンスワークショップ

COST03-BP06 ワークロードメトリクスに基づいてコストを配分する

使用量メトリクスや業績に基づいてワークロードのコストを配分し、ワークロードのコスト効率を測定します。インサイトとチャージバック機能が利用できる分析サービスにより、コストと使用状況データを分析するプロセスを実装します。

このベストプラクティスを活用しない場合のリスクレベル: 低

実装のガイダンス

コスト最適化とは、最低の価格点でビジネス成果を達成するということです。ワークロードメトリクス (ワークロードの効率で測定) に基づいてワークロードのコストを配分することによってのみ達成できます。定義されたワークロードメトリクスを、ログファイルまたは他のアプリケーションのモニタリングを使用してモニタリングします。このデータをワークロードのコストと組み合わせます。ワークロードのコストは、特定のタグ値またはアカウント ID のコストを確認することで取得できます。この分析を時間単位で実行します。静的なコストコンポーネント (恒久的に実行されるバックエンドデータベースなど) でリクエストレートが変化する (使用量のピークが午前 9 時から午後 5 時で、夜間のリクエストはほとんどない、など) 場合、通常、効率性は変化します。静的コストと変動コストの関係を理解しておくと、最適化アクティビティの焦点を絞ることができます。

共有リソースのワークロードメトリクスの作成は、Amazon Elastic Container Service (Amazon ECS) や Amazon API Gateway のコンテナ化されたアプリケーションのようなリソースに比べて難しい場合があります。ただし、使用量を分類してコストを追跡する方法はあります。Amazon ECS および AWS Batch の共有リソースを追跡する必要がある場合は、AWS Cost Explorer で分割コスト配分データを有効にできます。分割コスト配分データを使用すると、コンテナ化されたアプリケーションのコストと使用状況を把握して最適化し、共有コンピューティングリソースとメモリリソースの消費状況に基づいてアプリケーションコストを個々のエンティティに配分できます。

実装手順

- ワークロードメトリクスにコストを割り当てる: 定義されたメトリクスと設定されたタグを使用して、ワークロードの出力とワークロードのコストを組み合わせたメトリクスを作成します。Amazon Athena や Amazon QuickSight などの分析サービスを使用して、ワークロード全体やコンポーネントに対する効率性ダッシュボードを作成します。

リソース

関連ドキュメント:

- AWS リソースのタグ付け
- AWS Budgets によるコストの分析
- Cost Explorer によるコストの分析
- AWS コストと使用状況レポートの管理

関連する例:

- AWS 分割コスト配分データにより Amazon ECS および AWS Batch のコストの可視性を向上する

リソースを削除する

プロジェクト、従業員、テクノロジーリソースのリストを管理するようになると、使用されなくなったリソースや所有者のいなくなったプロジェクトを特定できるようになります。

ベストプラクティス

- COST04-BP01 ライフタイム全体にわたってリソースを追跡する
- COST04-BP02 廃止プロセスを実装する
- COST04-BP03 リソースを廃止する
- COST04-BP04 自動的にリソースを廃止する
- COST04-BP05 データ保持ポリシーを適用する

COST04-BP01 ライフタイム全体にわたってリソースを追跡する

ライフタイム全体にわたって、リソースや、リソースとシステムとの関係を追跡するメソッドを定義し、実装します。タグ付けにより、リソースのワークロードまたは機能を特定できます。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

不要になったワークロードリソースを廃止します。一般的な例としては、テスト用途のリソースがあります。テストが完了したら、リソースは削除できます。タグを使用してリソースを追跡する (およびそれらのタグに関するレポートを実行する) ことで、使用されなくなったり、ライセンスの有効期限が切れたりした場合に、廃止する資産を特定するのに役立ちます。リソース追跡には、タグの使用が効果的な方法です。リソースにその機能か、または廃止可能になる既知の日付をラベリングできます。そうすると、これらのタグでレポートを作成できます。機能タグを付ける場合の例として、`feature-X testing` という値であれば、ワークロードのライフサイクルの観点からリソースの目的を識別できます。もう 1 つの例は、削除されるタグキーの名前や値などのリソースに `LifeSpan` または `TTL` を使用して、廃止の期間や特定の時間を定義するものです。

実装手順

- タグ付けスキームを実装する: リソースが属するワークロードを識別するタグ付けスキームを実装し、ワークロード内のすべてのリソースが適切にタグ付けされることを確認します。タグ付けにより、目的、チーム、環境など、ビジネスに関連した基準でリソースを分類することができます。タグ付けのユースケース、戦略、テクニックの詳細については、「AWS のタグ付けのベストプラクティス」を参照してください。
- ワークロードのスループットまたは出力モニタリングを実装する: 入力リクエストまたは出力完了に対してワークロードスループットモニタリングまたはアラームを実装します。ワークロードのリクエストまたは出力がゼロになったときに、ワークロードのリソースが使用されなくなったことを示す通知を提供するように設定します。ワークロードが通常の条件下で定期的にゼロまで下がる場合は、時間要因を組み込みます。未使用または十分に活用されていないリソースの詳細については、「AWS Trusted Advisor コスト最適化チェック」を参照してください。
- AWS リソースをグループ化する: AWS リソースのグループを作成します。AWS Resource Groups を使用すると、同じ AWS リージョンにある AWS リソースを整理し管理することができます。ほとんどのリソースにタグを追加して、組織内のリソースを識別および並べ替えることができます。サポートされているリソースに一括でタグを追加するときはタグエディタを使用します。承認済み製品のポートフォリオを作成、管理し、エンドユーザーに配布して、製品ライフサイクルを管理するときは、AWS Service Catalog の使用を検討してください。

リソース

関連ドキュメント:

- AWS Auto Scaling
- AWS Trusted Advisor
- AWS Trusted Advisor コスト最適化チェック
- AWS リソースのタグ付け
- カスタムメトリクスをパブリッシュする

関連動画:

- AWS Trusted Advisor を使用してコストを最適化する方法

関連する例:

- AWS リソースを整理するにはどうすればよいですか?

- AWS Trusted Advisor を使用してコストを最適化する方法を教えてください。

COST04-BP02 廃止プロセスを実装する

未使用のリソースを特定して廃止するためのプロセスを実装します。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

組織全体で標準化されたプロセスを導入し、未使用のリソースを特定し、排除します。このプロセスでは、組織のすべての要件が満たされていることを検証するために、検索を実行する頻度と、リソースを削除するプロセスを定義する必要があります。

実装手順

- 廃止プロセスを作成、実装する: ワークロードの開発者や所有者と協力して、ワークロードとそのリソースの廃止プロセスを構築します。このプロセスでは、ワークロードが使用中であるかどうか、およびワークロードの各リソースが使用中であるかどうかを検証する方法を網羅する必要があります。リソースを廃止するために必要なステップを詳述し、サービスから削除すると同時に、規制要件の遵守を確保します。ライセンスやアタッチされたストレージなど、関連するリソースも含める必要があります。廃止プロセスが開始されたことをワークロードの所有者に通知します。

  プロセスの一部として何を確認する必要があるかについては、以下の廃止手順を使用してください。

  - 廃止されるリソースを特定する: AWS クラウドで廃止の対象となるリソースを特定します。必要な情報をすべて記録し、廃止スケジュールを設定します。タイムラインでは、プロセス中に予期せぬ問題が発生した場合 (およびそのタイミング) を考慮してください。
  - 調整とコミュニケーションをする: ワークロードの所有者と協力して、廃止されるリソースを確認します。
  - メタデータを記録してバックアップを作成する: メタデータ (パブリック IP、リージョン、AZ、VPC、サブネット、セキュリティグループなど) を記録し、本番環境のリソースに必要な場合、または重要なリソースである場合はバックアップ (Amazon Elastic Block Store スナップショットの作成、AMI の取得、キーのエクスポート、証明書のエクスポートなど) を作成します。
  - Infrastructure as Code の検証をする: リソースが AWS CloudFormation、Terraform、AWS Cloud Development Kit (AWS CDK)、またはその他の Infrastructure as Code デプロイツールでデプロイされたかどうかを判断し、必要に応じて再デプロイできるようにします。

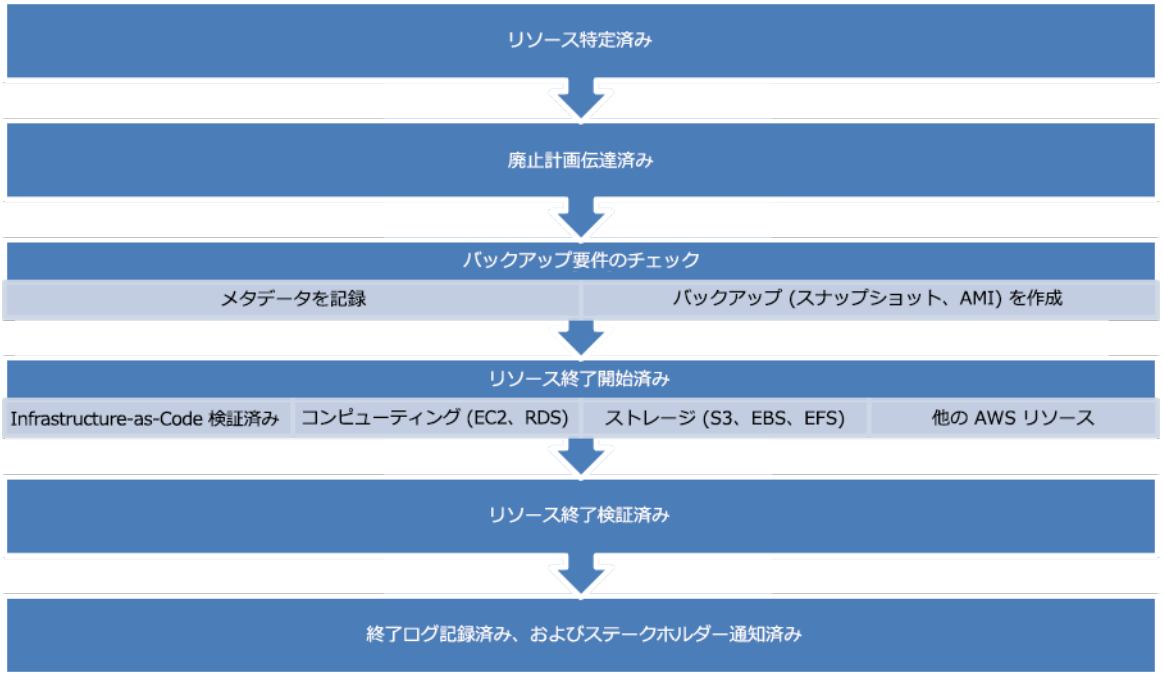
- アクセスの防止をする: リソースが必要かどうかを判断する間にリソースの使用を防ぐため、制限付きコントロールを一定期間適用します。必要に応じて、リソース環境を元の状態に戻せることを確認します。
- 内部廃止プロセスを遵守する: 組織ドメインからのリソースの削除、DNS レコードの廃止、または設定管理ツール、モニタリングツール、自動化ツール、およびセキュリティツールからのリソース削除など、組織の管理タスクと廃止プロセスに従います。

リソースが Amazon EC2 インスタンスである場合は、次のリストを参照してください。詳細については、「Amazon EC2 リソースを削除するか、終了する方法を教えてください。」を参照してください。

- すべての Amazon EC2 インスタンスとロードバランサーを停止または終了します。Amazon EC2 インスタンスは、終了後しばらくの間コンソールに表示されます。実行状態にないインスタンスには課金されません。
- Auto Scaling インフラストラクチャを削除する
- すべての専有ホストを解放します。
- Amazon EBS ボリュームと Amazon EBS スナップショットをすべて削除します。
- すべての Elastic IP アドレスを解放します。
- すべての Amazon マシンイメージ (AMI) の登録を解除します。
- すべての AWS Elastic Beanstalk 環境を終了します。

リソースが Amazon S3 Glacier ストレージ内のオブジェクトであり、最小保存期間を満たす前にアーカイブを削除した場合、日割り計算による早期削除料が課金されます。Amazon S3 Glacier の最小保存期間は使用するストレージクラスによって異なります。各ストレージクラスの最小保存期間の概要については、Amazon S3 ストレージクラスのパフォーマンスを参照してください。早期削除料金の計算方法の詳細については、「Amazon S3 の料金」を参照してください。

次の簡単な廃止プロセスのフローチャートは、廃止手順を概説しています。リソースを廃止する前に、廃止対象として特定したリソースが組織で使用されていないことを確認します。



リソース廃止フローです。

リソース

関連ドキュメント:

- AWS Auto Scaling
- AWS Trusted Advisor
- AWS CloudTrail

関連動画:

- CloudFormation スタックを削除するがいくつかのリソースを保持する
- Amazon EC2 インスタンスを起動したユーザーを確認するには

関連する例:

- Amazon EC2 リソースを削除するか、終了する方法を教えてください。
- 自分のアカウントで EC2 インスタンスを起動したユーザーを確認するにはどうすればよいですか?

COST04-BP03 リソースを廃止する

定期監査や使用状況の変化などのイベントを契機としてリソースを廃止します。通常、廃止は定期的に行われ、手動または自動で実行できます。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

使用していないリソースを検索する場合は節減額の程度によって検索頻度と投入する労力を決定する必要があるため、コスト発生額の小さいアカウントの分析は、コスト発生額が高額のアカウントよりも頻度を下げるべきです。イベントの検索および廃止は、製品が寿命を迎えた場合や交換する場合など、ワークロードの状態の変化によって開始されます。イベントの検索および廃止は、市況の変化や製品終了などの外部イベントによって開始される場合もあります。

実装手順

- リソースを廃止する: これは、不要になった AWS リソースの廃止段階またはライセンス契約の終了段階です。スナップショットやバックアップの取得などの不要な中断を防ぐために、廃止段階に移行してリソースを廃止する前に完了したすべての最終チェックを完了します。廃止プロセスを使用して、未使用と識別された各リソースを廃止します。

リソース

関連ドキュメント:

- AWS Auto Scaling
- AWS Trusted Advisor

COST04-BP04 自動的にリソースを廃止する

重要度が低いリソース、不要なリソース、使用率が低いリソースを特定して廃止する作業を適切に行えるようにワークロードを設計します。

このベストプラクティスを活用しない場合のリスクレベル: 低

実装のガイダンス

オートメーションを使用して、廃止プロセスの関連コストを削減または削除します。自動廃止するようにワークロードを設計すると、そのライフタイム全体にわたるワークロードコストを削減できま

す。Amazon EC2 Auto Scaling または Application Auto Scaling を使用して、廃止プロセスを実行できます。API または SDK でカスタムコードを実装し、ワークロードリソースを自動的に廃止することもできます。

モダンアプリケーションはサーバーレスファースト、つまりサーバーレスサービスの採用を優先するように構築されています。AWS は、コンピューティング、インテグレーション、データストアというスタックの 3 つのレイヤーすべてに対応するサーバーレスサービスを開発しました。サーバーレスアーキテクチャを使用すると、トラフィックの少ない間、自動的にスケールアップおよびスケールダウンしてコストを節約できます。

実装手順

- Amazon EC2 Auto Scaling または Application Auto Scaling を実装する: サポートされているリソースは Amazon EC2 Auto Scaling または Application Auto Scaling で設定します。これらのサービスは、AWS サービス利用時の使用率とコスト効率の最適化に役立ちます。これらのサービスは、需要が低下すると余分なリソースを自動的に削除するため、過剰な支出を避けることができます。
- インスタンスを終了するように CloudWatch を設定する: CloudWatch アラーム を使用してインスタンスを終了するように設定できます。廃止プロセスのメトリクスを使用して、Amazon Elastic Compute Cloud アクションでアラームを実装します。ロールアウトする前に、非本番環境でオペレーションを検証します。
- ワークロード内にコードを実装する: AWS SDK または AWS CLI を使用してワークロードリソースを廃止できます。AWS と統合し、使用されなくなったリソースを終了または削除するコードをアプリケーション内に実装します。
- サーバーレスサービスを使用する: アプリケーションを構築および実行する際はサーバーレスアーキテクチャとイベント駆動型アーキテクチャを AWS で構築することを優先します。AWS は、自動的に最適化されたリソース使用率と自動廃止 (スケールインとスケールアウト) を提供するサーバーレステクノロジーサービスを提供します。サーバーレスアプリケーションでは、リソース使用率が自動的に最適化され、過剰プロビジョニングの費用が発生しません。

リソース

関連ドキュメント:

- Amazon EC2 Auto Scaling
- Amazon EC2 Auto Scaling の使用を開始する
- Application Auto Scaling
- AWS Trusted Advisor

- AWS でのサーバーレス
- EC2 インスタンスを停止、終了、再起動、または復旧するアラームを作成する
- Amazon CloudWatch アラームへの終了アクションの追加

関連する例:

- AWS CloudFormation スタックの自動削除のスケジュール

COST04-BP05 データ保持ポリシーを適用する

サポートされるリソースでデータ保持ポリシーを定義し、組織の要件に従ってオブジェクトの削除を処理します。不要または孤立したリソースや不要になったオブジェクトを、特定して削除します。

このベストプラクティスを活用しない場合のリスクレベル: 中

データ保持ポリシーとライフサイクルポリシーを使用して、特定されたリソースの廃止プロセスに関連するコストとストレージコストを削減します。データ保持ポリシーとライフサイクルポリシーを定義してストレージクラスの移行や削除を自動で実行すると、ライフタイム期間の全体的なストレージコストが削減されます。Amazon Data Lifecycle Manager を使用して、Amazon Elastic Block Store スナップショットおよび Amazon EBS-backed Amazon マシンイメージ (AMI) の作成と削除を自動化できます。また、Amazon S3 Intelligent-Tiering または Amazon S3 ライフサイクル設定を使用して、Amazon S3 オブジェクトのライフサイクルを管理できます。また、API または SDK を使用してカスタムコードを実装することで、オブジェクトを自動的に削除するライフサイクルポリシーとポリシールールを作成することもできます。

実装手順

- Amazon Data Lifecycle Manager を使用する: Amazon Data Lifecycle Manager のライフサイクルポリシーを使用して、Amazon EBS スナップショットと Amazon EBS-backed AMI の削除を自動化します。
- バケットにライフサイクル設定をセットアップする: バケットで Amazon S3 ライフサイクル設定を使用して、ビジネス要件に基づいて Amazon S3 がオブジェクトのライフサイクル中に実行するアクションおよびオブジェクトライフサイクルの終了時の削除アクションを定義します。

リソース

関連ドキュメント:

- AWS Trusted Advisor
- Amazon Data Lifecycle Manager
- Amazon S3 のバケットのライフサイクル設定を行う方法

関連動画:

- Amazon Data Lifecycle Manager を使用した Amazon EBS スナップショット管理の自動化
- ライフサイクル設定ルールを使用して Amazon S3 バケットを空にするにはどうすればよいですか?

関連する例:

- ライフサイクル設定ルールを使用して Amazon S3 バケットを空にするにはどうすればよいですか?

コスト効率を考慮しながらリソースを利用する

ワークロードに適したサービス、リソース、設定の使用は、コスト節減の鍵を握ります。費用対効果の高いリソースを作成する場合、以下の点を考慮してください。

AWS ソリューションアーキテクト、AWS ソリューション、AWS リファレンスアーキテクチャ、APN パートナーを利用すると、学習内容に基づいてアーキテクチャを選択できます。

トピック

- サービスを選択する際にコストを評価する
- 正しいリソースタイプ、リソースサイズ、リソース数を選択する
- 最適な料金モデルを選択する
- データ転送を計画する

サービスを選択する際にコストを評価する

ベストプラクティス

- COST05-BP01 組織のコスト要件を特定する
- COST05-BP02 ワークロードのすべてのコンポーネントを分析する
- COST05-BP03 各コンポーネントの詳細な分析を実行する
- COST05-BP04 コスト効率の高いライセンスを提供するソフトウェアを選択する
- COST05-BP05 組織の優先順位に従ってコストが最適化されるようにこのワークロードのコンポーネントを選択する
- COST05-BP06 異なる使用量について経時的なコスト分析を実行する

COST05-BP01 組織のコスト要件を特定する

チームメンバーと協力して、コストの最適化とこのワークロードのその他の柱とのバランス (パフォーマンスや信頼性など) を定義します。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

ほとんどの組織で、情報技術 (IT) 部門は複数の小さなチームから構成されており、抱える議題や注力分野は、所属メンバーの専門分野とスキルを反映してそれぞれに異なります。組織全体の目標、優先事項、目的を把握し、各部門や各プロジェクトがそうした目標にどのように貢献しているかを理解する必要があります。人員、設備、技術、資材、外部サービスなど、すべての重要なリソースを分類しておくことは、組織の目標を達成し、包括的な予算計画を立てるうえで不可欠です。こうした体系的なアプローチでコストを特定し、把握することが、組織にとって現実的で健全なコスト計画を立てるための基本です。

ワークロードのサービスを選択する場合は、組織の優先順位を理解することが重要です。コスト最適化と、AWS Well-Architected フレームワークの他の柱 (パフォーマンスや信頼性など) とのバランスを図ります。このプロセスは、組織の目標、市況、業務のダイナミクスの変化を反映するために、体系的かつ定期的に実施する必要があります。十分にコスト最適化されたワークロードとは組織の要件に最も適合するソリューションであって、必ずしも最低コストのソリューションとは限りません。製品、ビジネス、技術、財務など、組織内のすべてのチームと会合し、情報を収集します。競合する利益または代替アプローチ間のトレードオフの影響を評価し、重点領域を決定するか、一連のアクションを選択する際に十分な情報に基づいて意思決定を下せるようにします。

例えば、新しい機能の市場投入までの時間を短縮することは、コストの最適化よりも重視されることがあります。または、非リレーショナルデータ用にリレーショナルデータベースを選択すれば、データ型に合わせて最適化されたデータベースに移行してアプリケーションを更新するよりも、システムの移行が簡素化されます。

実装手順

- 組織のコスト要件を特定する: 製品管理、アプリケーション所有者、開発および運用チーム、管理、財務ロールのメンバーなど、組織のチームメンバーとミーティングを行います。このワークロードとそのコンポーネントに対して、Well-Architected の柱に優先順位を付けます。柱を順番に並べたリストを作成してください。また、それぞれの柱に重みを付け、その柱が他の柱よりどの程度重視されているかや、2 つの柱の重点度がどの程度類似しているかを示すことができます。
- 技術的負債に対処し、文書化する: ワークロードのレビュー中に、技術的負債に対処します。バックログ項目を文書化して、後日、リファクタリングやリアーキテクティングで最適化を進めることを目標に、ワークロードを保持します。生じたトレードオフを他のステークホルダーに明確に伝えることが大切です。

リソース

関連するベストプラクティス:

- REL11-BP07 可用性の目標と稼働時間のサービスレベルアグリーメント (SLA) を満たす製品を設計する
- OPS01-BP06 トレードオフを評価する

関連ドキュメント:

- AWS 総保有コスト (TCO) 計算ツール
- Amazon S3 ストレージクラス
- AWS クラウド製品

COST05-BP02 ワークロードのすべてのコンポーネントを分析する

現在のサイズやコストに関係なく、すべてのワークロードが分析されることを確認します。見直しを行う際には、現在のコストや予想コストなどの潜在的利益を織り込む必要があります。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

組織にビジネス価値をもたらすべく設計されたワークロードコンポーネントには、さまざまなサービスが含まれる場合があります。コンポーネントごとに、ビジネスニーズに対応する特定の AWS クラウドサービスを選択できます。何が選択されるかは、それらのサービスに関する知識や使用経験などの要因によって違ってきます。

「COST05-BP01 組織のコスト要件を特定する」で説明されているように組織の要件を特定したら、ワークロード内のすべてのコンポーネントを徹底的に分析します。現時点と今後予測されるコストとサイズを考慮して、各コンポーネントを分析します。分析のコストを、ワークロードのライフサイクル全体で削減が見込まれる額と比較検討してください。対象となるワークロードのコンポーネントをすべて分析するための労力が、そのコンポーネントの最適化により見込まれる節約額や改善に見合っていなければなりません。例えば、提案されたリソースのコストが月額 10 USD で、予測負荷が月額 15 USD を超えない場合に、コストを 50% (月額 5 USD) 削減するために 1 日分の労力を費やすようでは、システムの寿命全体にわたって得られると考えられる利益を超えることになるかもしれません。データに基づくより高速でより効率的な予測を使用すると、このコンポーネントの全体的な成果を最善のものにできます。

ワークロードは時間の経過とともに変化する可能性があり、ワークロードのアーキテクチャや使用方法が変化すると、適切だったサービスの組み合わせが最適ではなくなってしまうことがあります。サービスの選択に関する分析には、現在および将来のワークロードの状態と使用量レベルが組み込まれる必要があります。将来のワークロードの状態や使用量に合わせてサービスを運用すると、今後の変更に必要な労力を軽減または削除できることになり、全体的なコストを削減できます。例えば、最初は EMR Serverless の使用が適しているかもしれません。ただし、そのサービスの使用量が増えてきたら、EC2 の EMR に移行することで、ワークロードの該当コンポーネントのコストを削減できる可能性があります。

AWS Cost Explorer および AWS Cost and Usage Report (CUR) では、概念実証 (PoC) または実行中の環境のコストを分析できます。AWS 料金見積りツール を使用してワークロードのコストを見積もることもできます。

技術チームがワークロードを見直すためのワークフローを作成します。このワークフローはシンプルなものにし、必要なステップをすべて網羅することで、チームがワークロードの各コンポーネントとその料金を理解できるようにします。組織はこのワークフローに従い、またそれを各チームの特定のニーズに基づいてカスタマイズできます。

1. ワークロードに使用されている各サービスを一覧表示する: これは良い出発点です。現在使用されているすべてのサービスと、コストの発生源を特定します。
2. これらのサービスの料金体系を理解する: 各サービスの料金モデルを理解します。AWS の各サービスには、使用量、データ転送、機能に固有の料金などの要因に基づくさまざまな料金モデルがあります。
3. 予期しないワークロードコストがあり、予想される使用量やビジネス成果と一致しないサービスに着目する: AWS Cost Explorer または AWS Cost and Usage Report の使用量と価値に対してコストが比例しない例外やサービスを特定します。最適化にかける労力に優先順位を付けるには、コストとビジネス成果を相関付けることが重要です。
4. AWS Cost Explorer、CloudWatch Logs、VPC フローログ、Amazon S3 ストレージレンズを使用して、これらの高コストの根本原因を把握する: これらのツールは、高コストの診断に役立ちます。各サービスは、使用量とコストの確認と分析に役立つ異なる観点を提供します。例えば、Cost Explorer は全体的なコスト傾向の特定に役立ち、CloudWatch Logs は運用に関するインサイトを提供します。また、VPC フローログは IP トラフィックを表示し、Amazon S3 ストレージレンズはストレージ分析に有用です。
5. AWS Budgets を使用して、サービスまたはアカウントの特定金額の予算を設定する: 予算の設定は、積極的なコスト管理方法です。AWS Budgets を使用して、カスタム予算しきい値を設定し、コストがそのしきい値を超えたときにアラートを受け取ります。

6. 請求および使用状況アラートを送信するように Amazon CloudWatch アラームを設定する: コストと使用状況メトリクスのモニタリングとアラートを設定します。CloudWatch アラームを使用すると特定のしきい値を超えたときに通知できるため、介入の応答時間を短縮できます。

現在の属性に関係なく、すべてのワークロードコンポーネントを戦略的に見直すことで、時間の経過に伴う着実な強化とコスト削減を促進します。このレビュープロセスに費やす労力は、それに見合う効果が得られるか慎重に検討したうえで、決める必要があります。

実装手順

- ワークロードコンポーネントを一覧する: ワークロードのコンポーネントのリストを作成します。このリストを使用して、各コンポーネントが分析されたことを確認します。費やされる労力は、組織の優先順位によって定義されたワークロードの重要度に見合うものにする必要があります。効率向上のために、リソースを機能別にグループ化します (複数のデータベースがある場合は本番データベースストレージなど)。
- コンポーネントリストを優先順位付けする: コンポーネントリストを取得して、労力をかける順で優先順位を付けます。これは通常、コンポーネントのコストが最も高価なものから最も安価なものへ、または組織の優先順位で定義されている重要度の順に並べられます。
- 分析を実行する: リストの各コンポーネントについて、使用可能なオプションとサービスを確認し、組織の優先順位に最適なオプションを選択します。

リソース

関連ドキュメント:

- AWS 料金見積りツール
- AWS Cost Explorer
- Amazon S3 ストレージクラス
- AWS クラウド 製品

関連動画:

- AWS コスト最適化シリーズ: CloudWatch

COST05-BP03 各コンポーネントの詳細な分析を実行する

各コンポーネントの、組織にかかる全体的なコストを調べます。運用および管理のコスト、特にクラウドプロバイダーが提供するマネージドサービスを使用するコストを考慮して、総保有コストを計算します。レビューを行う際には、潜在的利益 (分析に費やされた時間がコンポーネントのコストに比例しているなど) を織り込む必要があります。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

時間短縮を検討して、チームが技術的な負債の返済、イノベーション、付加価値機能、ビジネスの差別化要素の構築に集中できるようにします。例えば、データベースをできるだけ迅速にオンプレミス環境からクラウドにリフトアンドシフト (リホストともいいます) して、後で最適化する必要がある場合です。AWS でマネージドサービスを利用して、ライセンスコストの排除または削減を模索することには、時間をかけるだけの価値があります。AWS のマネージドサービスによって、OS のパッチ適用やアップグレードなど、サービス維持に伴う運用上および管理上の負担が軽減されるため、イノベーションとビジネスに集中できます。

マネージドサービスはクラウド規模で運用されるため、トランザクションまたはサービス単位でコストを削減できます。アプリケーションのコアアーキテクチャを変更せずに、具体的なメリットを生み出すための潜在的最適化作業を行うことができます。例えば、Amazon Relational Database Service (Amazon RDS) などの Database as a Service プラットフォームに移行するか、アプリケーションを AWS Elastic Beanstalk などのフルマネージドプラットフォームに移行することで、データベースインスタンスの管理に要する時間を短縮したいと考えているとします。

通常、マネージドサービスは、十分なキャパシティを確保するために設定できる属性を備えています。この属性を設定およびモニタリングして、余剰キャパシティを最小限に抑え、パフォーマンスを最大化する必要があります。AWS Management Console や AWS API および SDK を使用して AWS Managed Services の属性を変更し、需要の変化に合わせてリソースのニーズを調整できます。例えば、Amazon EMR クラスター (または Amazon Redshift クラスター) のノード数を増減して、規模をスケールアウトまたはスケールインできます。

また、AWS リソースの複数のインスタンスを圧縮して、高密度での使用を有効にすることもできます。例えば、単一の Amazon Relational Database Service (Amazon RDS) データベースインスタンスで、複数の小さなデータベースをプロビジョニングできます。使用量が増えたら、スナップショットや復元プロセスを使用して、そのデータベースの 1 つを専用の Amazon RDS データベースインスタンスに移行できます。

マネージドサービスでワークロードをプロビジョニングする際は、サービスキャパシティの調整要件を理解する必要があります。主な要件としては、時間、労力、通常のワークロードオペレーションへの影響などが一般には考えられます。プロビジョニングされたリソースでは変更が発生するまでの時間が許容され、このために必要なオーバーヘッドをプロビジョニングする必要があります。サービス変更に必要な継続的労力は、システムに統合する API と SDK や、Amazon CloudWatch などのモニタリングツールを使用することで、実質ゼロまで減らすことができます。

Amazon RDS、Amazon Redshift、Amazon ElastiCache はマネージドデータベースサービスを提供しています。Amazon Athena、Amazon EMR、および Amazon OpenSearch Service は、マネージド分析サービスを提供します。

AMS は、エンタープライズのお客様やパートナーに代わって AWS インフラストラクチャを運用するサービスです。コンプライアンスに準拠したセキュアな環境で、ワークロードをデプロイできます。AMS では、エンタープライズクラウド運用モデルとオートメーションを使用して、組織の要件を満たし、クラウド移行を高速化し、オンゴーイングの管理コストを削減できます。

実装手順

- 徹底分析を実行する: コンポーネントリストを使用して、各コンポーネントを優先度が高いものから処理します。優先度がより高く、より多くのコストがかかるコンポーネントについては、追加の分析を実行し、利用可能なすべてのオプションとその長期的な影響を評価します。優先度の低いコンポーネントの場合、使用状況の変化によってコンポーネントの優先度が変更するかどうかを評価し、かける労力の適切性の分析を実行します。
- マネージドリソースと非マネージドリソースを比較する: 管理するリソースの運用コストを考慮して、AWS マネージドリソースと比較します。例えば、Amazon EC2 インスタンスで実行しているデータベースをレビューし、Amazon RDS (AWS マネージドサービス) を使用した場合と比較したり、Amazon EMR を、Amazon EC2 で Apache Spark を実行する場合と比較したりします。セルフマネージドワークロードから AWS のフルマネージドワークロードに移行する際は、オプションを慎重に研究してください。考慮すべき最も重要な 3 つの要因は、使用するマネージドサービスのタイプ、データの移行に使用するプロセス、AWS 責任共有モデルの理解です。

リソース

関連ドキュメント:

- AWS 総保有コスト (TCO) 計算ツール
- Amazon S3 ストレージクラス
- AWS クラウド 製品

- AWS 責任共有モデル

関連動画:

- Why move to a managed database?
- What is Amazon EMR and how can I use it for processing data?

関連する例:

- マネージドデータベースに移行すべき理由
- Consolidate data from identical SQL Server databases into a single Amazon RDS for SQL Server database using AWS DMS
- Amazon Managed Streaming for Apache Kafka (Amazon MSK) にデータを大規模に提供する
- Migrate an ASP.NET web application to AWS Elastic Beanstalk

COST05-BP04 コスト効率の高いライセンスを提供するソフトウェアを選択する

オープンソースソフトウェアは、ワークロードに多大なコストをもたらすソフトウェアライセンスコストを排除することができます。ライセンスされたソフトウェアが必要な場合は、CPU などの任意の属性に結びついたライセンスは避け、出力または結果に結びついたライセンスを探します。これらのライセンスのコストは、提供するメリットに応じてより密にスケールされます。

このベストプラクティスを活用しない場合のリスクレベル: 低

実装のガイダンス

「オープンソース」は、ソフトウェア開発の文脈で生まれた用語であり、ソフトウェアが特定の無料配布基準に準拠しているという意味です。オープンソースソフトウェアは、誰でも検査、変更、拡張できるソースコードで構成されています。組織はビジネス要件、エンジニアのスキル、予測される使用状況、その他の技術的な依存関係を踏まえて、ライセンスコストを最小限に抑えるため、AWS でオープンソースソフトウェアを使用することを検討できます。言い換えれば、ソフトウェアライセンスのコストは、オープンソースソフトウェアを使用することで削減できます。オープンソースソフトウェアへの変更は、ワークロードサイズが拡大するにつれ、ワークロードコストに大きな影響を与える可能性があります。

ライセンスを取得したソフトウェアで得られる効果を、ワークロードの最適化にかかる総コストに照らして測定してください。ライセンス変更とその変更がワークロードコストに与える影響をモデリングします。あるベンダーがデータベースライセンスのコストを変更したなら、それがワークロードの全体的な効率にどのような影響を与えるかを調査します。ベンダーの過去の価格アナウンスを検討して、ベンダー製品全体のライセンス変更の傾向を検討してください。ライセンスコストは、ハードウェアごとにスケールするライセンス (CPU バウンドライセンス) など、スループットや使用量とは関係なくスケールされる場合があります。こうしたライセンスは、それに伴う成果が見られないままコストが急増する可能性があるため、避けてください。

例えば、Linux オペレーティングシステムを搭載した Amazon EC2 インスタンスを us-east-1 で運用する場合、Windows で実行する別の Amazon EC2 インスタンスを運用する場合と比較して約 45% のコスト削減になります。

AWS 料金見積りツール では、Amazon RDS インスタンスや各種データベースエンジンなど、ライセンスオプションが異なるさまざまなリソースのコストを包括的に比較できます。さらに、AWS Cost Explorer では、既存のワークロード (特にライセンスがさまざまに異なるワークロード) のコストを有益な視点で検討できます。ライセンス管理のために、AWS License Manager はソフトウェアライセンスを監督および処理するための合理化された方法を提供します。お客様は、AWS クラウドでお好みのオープンソースソフトウェアをデプロイして運用できます。

実装手順

- ライセンスオプションを検討する: 利用可能なソフトウェアのライセンス条項を確認します。必要な機能を備えたオープンソースバージョンを探し、ライセンスされたソフトウェアの利点がコストを上回っているかどうかを調べます。好条件があれば、ソフトウェアのコストに見合う利点が得られます。
- ソフトウェアプロバイダーを分析する: ベンダーからの料金またはライセンスの変更履歴を確認します。特定のベンダーのハードウェアまたはプラットフォームで実行することについての懲罰的な条件など、結果に見合わない変更を調べます。また、監査の実行方法や課される可能性のある罰則についても確認します。

リソース

関連ドキュメント:

- Open Source at AWS
- AWS 総保有コスト (TCO) 計算ツール
- Amazon S3 ストレージクラス

- AWS クラウド製品

関連する例:

- オープンソースブログ
- AWS オープンソースブログ
- 最適化とライセンス評価

COST05-BP05 組織の優先順位に従ってコストが最適化されるようにこのワークロードのコンポーネントを選択する

ワークロードのすべてのコンポーネントを選択したときのコストを考慮します。これには、アプリケーションレベルのサービスとマネージドサービス、またはサーバーレス、コンテナ、イベント駆動型アーキテクチャを使用して、全体のコストを削減することが含まれます。オープンソースソフトウェアやライセンス料金がかからないソフトウェア、または代替品を使用して、支出を最小限に抑えます。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

すべてのコンポーネントを選択する際は、サービスのコストとオプションを考慮します。これには、アプリケーションレベルのサービスとマネージドサービスである、Amazon Relational Database Service (Amazon RDS)、Amazon DynamoDB、Amazon Simple Notification Service (Amazon SNS)、Amazon Simple Email Service (Amazon SES) を使用して組織の全体的なコストを削減することが含まれます。

コンピューティングにはサーバーレスやコンテナを使用します。AWS Lambda や静的ウェブサイト用の Amazon Simple Storage Service (Amazon S3) などです。可能であればアプリケーションをコンテナ化し、Amazon Elastic Container Service (Amazon ECS) や Amazon Elastic Kubernetes Service (Amazon EKS) などの AWS マネージドコンテナサービスを使用します。

オープンソースソフトウェア、またはライセンス料金のないソフトウェア (コンピューティングワークロード用の Amazon Linux、データベースを Amazon Aurora に移行するなど) を使用して、ライセンスコストを最小限に抑えます。

Lambda、Amazon Simple Queue Service (Amazon SQS)、Amazon SNS、Amazon SES などのサーバーレスまたはアプリケーションレベルのサービスを使用できます。これらのサービスではリソース

を管理する必要がなく、コード実行、キューサービス、メッセージ配信の機能を利用できます。もう 1 つの利点は、使用量に応じてパフォーマンスとコストをスケールインするため、コスト配分とコストの帰属が効率的になることです。

イベント駆動型アーキテクチャは、サーバーレスサービスで使用することもできます。イベント駆動型アーキテクチャはプッシュベースであるため、イベントはルーターで発生してもオンデマンドで取得されます。この方法では、イベントをチェックするために定期的にポーリングする費用が発生しません。つまり、ネットワーク帯域幅の消費を抑え、CPU 使用率は低く、アイドルなフリートキャパシティは少なくなり、SSL/TLS ハンドシェイクも減ります。

サーバーレスの詳細については、Well-Architected サーバーレスアプリケーションレンズのホワイトペーパーを参照してください。

実装手順

- 各サービスを選択してコストを最適化する: 優先順位リストと分析を使用して、組織の優先順位に最も合致する各オプションを選択します。需要に合わせてキャパシティを増やすのではなく、より低いコストでより優れたパフォーマンスを得られる可能性がある他のオプションを検討します。例えば、AWS 上のデータベースに対する予想されるトラフィックを見直す必要がある場合、インスタンスサイズを増やす、または Amazon ElastiCache サービス (Redis または Memcached) を使用してデータベースにキャッシュメカニズムを提供することを検討します。
- イベント駆動型アーキテクチャを評価する: サーバーレスアーキテクチャを使用すると、分散マイクロサービスベースのアプリケーション向けにイベント駆動型アーキテクチャを構築することもできます。これを利用すると、スケーラブルで回復性が高く、迅速かつコスト効果の高いソリューションを構築できます。

リソース

関連ドキュメント:

- AWS 総保有コスト (TCO) 計算ツール
- AWSサーバーレス
- イベント駆動型アーキテクチャとは
- Amazon S3 ストレージクラス
- AWS クラウド製品
- Amazon ElastiCache (Redis OSS)

関連する例:

- イベント駆動型アーキテクチャの導入
- イベント駆動型アーキテクチャ
- Amazon ElastiCache (Redis OSS) を使用して 100 倍効率よく Statsig を実行する方法
- AWS Lambda 関数を使用するためのベストプラクティス

COST05-BP06 異なる使用量について経時的なコスト分析を実行する

ワークロードは時間の経過とともに変化することがあります。それぞれのサービスまたは機能のコスト効率は、使用レベルによって異なります。各コンポーネントについて予想使用量に基づく経時的な分析を実行することで、ワークロードのコスト効率性がそのライフタイム全体にわたって維持されます。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

AWS で新しいサービスや機能がリリースされると、ワークロードに最適なサービスが変化する可能性があります。求められる労力は、潜在的な利点が反映されたものである必要があります。ワークロードレビューの頻度は、組織の要件によって異なります。ワークロードにかなりのコストがかかっている場合、新しいサービスの運用が早いほどコスト削減が最大になるため、レビュー頻度が高い方が有利です。レビューの開始要因には、使用パターンの変化も挙げられます。使用量が大幅に変化した場合は、別のサービスを使った方がよい場合もあります。

データを AWS クラウドに移動する必要がある場合、AWS が提供するバリエーション豊かな製品やパートナーツールを選択して、データセットを移行できます。データセットは、ファイル、データベース、マシンイメージ、ブロックボリューム、あるいはテープバックアップであっても構いません。例えば、大量のデータを AWS に対して入出力する場合や、エッジでデータを処理する場合、AWS の目的別デバイスのいずれかを使用して、コスト効果が高い方法でペタバイト規模のデータをオフラインで移動できます。別の例としては、より速いデータ転送速度が必要な場合、VPN よりも、ビジネスに必要な安定した接続性能を提供する直接接続サービスの方が安価な場合があります。

さまざまな使用状況において繰り返したコスト分析を基にして、スケーリングアクティビティをレビューします。結果を分析して、複数のインスタンスタイプと購入オプションを使用したインスタンスの追加に合わせてスケーリングポリシーを調整できるかを確認します。設定をレビューして、最

小限を削減してもユーザーリクエストを処理できる (ただしより小さなフリートサイズで) かを確認し、予想される高需要を満たすためにリソースを追加します。

組織のステークホルダーと話し合い、AWS Cost Explorer の予測機能を使用してサービス変更の潜在的な影響を予測し、時間の経過とともにさまざまな使用状況のコスト分析を行います。AWS Budgets、CloudWatch 請求アラーム、AWS Cost Anomaly Detection を使用して使用状況レベルのトリガーをモニタリングし、最もコスト効果が高いサービスをなるべく迅速に特定して実装します。

実装手順

- 予測された使用パターンを定義する: マーケティングや製品所有者などの組織と協力して、ワークロードに対して期待および予測される使用パターンを文書化します。これまでと今後両方のコストと使用量の増加についてビジネス上の関係者と話し合い、増加がビジネス要件に沿ったものであることを確認します。自社の AWS リソースを使用するユーザーが増える日、週、月を特定します。そのタイミングで既存のリソースのキャパシティを増やすか追加サービスを導入して、コストを削減しパフォーマンスを向上させる必要があります。
- 予測された使用量に基づきコスト分析を実行する: 定義された使用パターンを使用して、それらの各ポイントで分析を実行します。分析作業は、潜在的な結果を反映する必要があります。例えば、使用量の変化が大きい場合は、コストと変化を確認するために詳細な分析を実行する必要があります。つまり、コストが増えていれば、ビジネスにおける使用量も同様に増えているはずです。

リソース

関連ドキュメント:

- AWS 総保有コスト (TCO) 計算ツール
- Amazon S3 ストレージクラス
- AWS クラウド製品
- Amazon EC2 Auto Scaling
- クラウドへのデータ移行
- AWS Snow Family

関連動画:

- AWS OpsHub for Snow Family

正しいリソースタイプ、リソースサイズ、リソース数を選択する

最適なリソースタイプ、リソースサイズ、リソース数を選択することで、最低限のリソースコストで技術要件を満たすことができます。適切なサイズ変更アクティビティは、ワークロードのすべてのリソース、各リソースのすべての属性、および適切なサイジング操作に関連する作業が考慮されます。適切なサイズ変更は、使用パターンや、AWS の値下げや 新しい AWS リソースタイプなどの外部要因の変化により開始される反復プロセスになることがあります。適切なサイズ設定の作業コストが、ワークロードの寿命全体にわたって可能と考えられる削減額を上回る場合は、1 回だけ設定することも可能です。

AWS には以下のような多様なアプローチがあります。

ベストプラクティス

- COST06-BP01 コストモデリングを実行する
- COST06-BP02 データに基づいてリソースタイプ、リソースサイズ、リソース数を選択する
- COST06-BP03 メトリクスに基づいて自動的にリソースタイプ、リソースサイズ、リソース数を選択する
- COST06-BP04 共有リソースの使用を検討する

COST06-BP01 コストモデリングを実行する

組織の要件 (ビジネスニーズや既存のコミットメントなど) を特定して、ワークロードとその各コンポーネントのコストモデリング (全体コスト) を実行します。予測されたさまざまな負荷のワークロードに対してベンチマークアクティビティを実行し、コストを比較します。モデリングの際には、潜在的な利点を織り込む必要があります。例えば、費やされた時間がコンポーネントのコストに釣り合っているなどです。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

ワークロードと各コンポーネントのコストモデリングを実行してリソース間のバランスを把握し、特定のパフォーマンスレベルに応じてワークロード内の各リソースの適切なサイズを見つけます。コストの考慮事項を理解すると、計画されたワークロードのデプロイにおける価値実現の成果評価時に、組織のビジネスケースや意思決定プロセスがわかります。

予測されたさまざまな負荷のワークロードに対してベンチマークアクティビティを実行し、コストを比較します。モデリングの際には、費やした時間がコンポーネントのコストまたは予想される削減額

に比例しているといった潜在的な利点を織り込む必要があります。このプロセスのベストプラクティスについては、AWS Well-Architected フレームワークのパフォーマンス効率の柱に関するレビューセクションを参照してください。

例えば、コンピューティングリソースで構成されているワークロードのコストモデリングを作成するときは、AWS Compute Optimizer がワークロード実行のためのコストモデリングに役立ちます。使用履歴に基づき、コンピューティングリソースの正しいサイズ設定に関するレコメンデーションを提供します。CloudWatch エージェントが Amazon EC2 インスタンスにデプロイされていることを確認し、AWS Compute Optimizer 内でより正確な推奨事項を得ることができるメモリメトリクスを収集します。リスクレベルに応じて複数のレコメンデーションを作成できる機械学習が使われている無料サービスであるため、コンピューティングリソースにとって理想的なデータソースです。

他のサービスやワークロードコンポーネントのサイズ適正化のために、カスタムログをデータソースとして使用できるサービスは、AWS Trusted Advisor、Amazon CloudWatch、Amazon CloudWatch Logs など複数あります。リソースをチェックして使用率が低いリソースにフラグを立てる AWS Trusted Advisor は、リソースのサイズを適正化しコストモデリングを作成するのに役立ちます。

コストモデリングのデータとメトリクスに関する推奨事項は以下のとおりです。

- モニタリングはユーザーエクスペリエンスを正確に反映する必要があります。対象期間に適切な間隔を選択して、平均の変わりに最大値や 99 パーセンタイル値をじっくり見極めます。
- すべてのワークロードのサイクルをカバーするために必要な分析期間の適切な間隔を選択します。例えば、分析を 2 週間間隔で実行する場合、1 か月サイクルで使用率が高くても見逃す場合があり、過小プロビジョニングにつながる可能性があります。
- 既存のコミットメント、他のワークロード用に選択された料金モデル、イノベーションを迅速化しコアビジネスバリューに集中する能力を考慮し、計画されたワークロードに対して適切な AWS サービスを選択します。

実装手順

- リソースのコストモデリングを実行する: ワークロードまたは概念実証を、テストする特定のリソースタイプとサイズを持つ別のアカウントにデプロイします。テストデータを使用してワークロードを実行し、出力結果のほか、テスト実行時のコストデータを記録します。その後、ワークロードを再デプロイするか、リソースタイプとサイズを変更して、テストをもう一度実行します。コストモデリングの際には、これらのリソースで使用する可能性のある製品のライセンス料と、これらのリソースをデプロイおよび管理する推定運用 (作業またはエンジニア) コストを含めます。一定期間 (時間単位、日次、月次、年次、三年次) のコストモデリングを考慮します。

リソース

関連ドキュメント:

- AWS Auto Scaling
- 適切なサイジングのための機会の特定
- Amazon CloudWatch の特徴
- コスト最適化: Amazon EC2 の適切なサイジング
- AWS Compute Optimizer
- AWS Pricing Calculator

関連する例:

- Perform a Data-Driven Cost Modelling
- 計画している AWS リソース構成のコストを見積もる方法を教えてください。
- Choose the right AWS tools

COST06-BP02 データに基づいてリソースタイプ、リソースサイズ、リソース数を選択する

ワークロードとリソースの特性に関するデータに基づいて、リソースのサイズやタイプを選択します。例えば、コンピューティング、メモリ、スループット、書き込み頻度などです。この選択は通常、以前の (オンプレミス) バージョンのワークロード、ドキュメント、ワークロードに関する他の情報ソースを用いて行います。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

Amazon EC2 では、さまざまなユースケースに合わせて、CPU、メモリ、ストレージ、ネットワークキャパシティのレベルが異なる、幅広いインスタンスタイプを選択肢として用意しています。インスタンスタイプごとに CPU、メモリ、ストレージ、ネットワーク機能の組み合わせが異なるため、プロジェクトに適したリソースの組み合わせを柔軟に選択できます。どのインスタンスタイプにもサイズが複数用意されており、ワークロードの需要に基づいてリソースを調整できます。必要なインスタンスタイプを判断するには、インスタンスで実行する予定のアプリケーションまたはソフトウェア

のシステム要件に関する詳細情報を収集する必要があります。これらの詳細には、次の内容を含める必要があります。

- オペレーティングシステム
- CPU コア数
- GPU コア
- システムメモリ (RAM) の容量
- ストレージタイプとスペース
- ネットワーク帯域幅要件

コンピューティング要件の目的と必要なインスタンスを明らかにしたうえで、さまざまな Amazon EC2 インスタンスファミリーを検討します。次のインスタンスタイプファミリーが提供されています。

- 汎用
- コンピューティング最適化
- メモリを最適化
- ストレージの最適化
- 高速コンピューティング
- HPC 最適化

特定の Amazon EC2 インスタンスファミリーが達成できる特定の目的とユースケースの詳細については、「AWS インスタンスタイプ」を参照してください。

お客様のニーズに最適な特定のインスタンスファミリーとインスタンスタイプを選択するには、システム要件の収集が不可欠です。インスタンスタイプ名は、ファミリー名とインスタンスサイズで構成されます。例えば、t2.micro インスタンスは T2 ファミリーに属するマイクロサイズです。

ワークロードとリソースの特性 (例えば、コンピューティング、メモリ、スループット、書き込み頻度) に基づいて、リソースのサイズやタイプを選択します。この選択は通常、コストモデリング、以前のバージョンのワークロード (オンプレミスバージョンなど)、ドキュメント、ワークロードに関する他の情報ソース (ホワイトペーパー、公開ソリューション) を用いて行います。AWS 料金見積りツールやコスト管理ツールを使用すれば、十分な判断材料を基にインスタンスのタイプ、サイズ、構成を決定できます。

実装手順

- データに基づいてリソースを選択する: コストモデリングのデータを使用して、予測されるワークロードの使用レベルを選択し、指定されたリソースタイプとサイズを選択します。コストモデリングデータに基づいて、インスタンスに求められるデータ転送速度を考慮しつつ、仮想 CPU の数、総メモリ (GiB)、ローカルインスタンスストアボリューム (GB)、Amazon EBS ボリューム、ネットワークパフォーマンスレベルを決定します。常に詳細な分析と正確なデータに裏付けられた選択を行い、パフォーマンスの最適化とコスト管理の効率化を両立させましょう。

リソース

関連ドキュメント:

- AWS インスタンスタイプ
- AWS Auto Scaling
- Amazon CloudWatch の特徴
- EC2 Right Sizing によるコスト最適化

関連動画:

- Selecting the right Amazon EC2 instance for your workloads
- Right size your service

関連する例:

- It just got easier to discover and compare Amazon EC2 instance types

COST06-BP03 メトリクスに基づいて自動的にリソースタイプ、リソースサイズ、リソース数を選択する

現在実行しているワークロードからのメトリクスを用いて、コストを最適化する適切なサイズやタイプを選択します。コンピューティング、ストレージ、データ、ネットワーキングなどのサービスに対して、適切なスループット、サイジング、ストレージのプロビジョニングを行います。これは、自動スケーリングなどのフィードバックループまたはワークロードのカスタムコードで行うことができます。

このベストプラクティスを活用しない場合のリスクレベル: 低

実装のガイダンス

ワークロード内に、実行中のワークロードのアクティブなメトリクスを使用してそのワークロードを変更するフィードバックループを作成します。適切なサイジング操作を実行するように設定した AWS Auto Scaling などのマネージドサービスを使用できます。AWS には、API、SDK、最小限の労力でリソースを変更することができる機能も用意されています。Amazon EC2 インスタンスの停止と起動のワークロードをプログラムして、インスタンスサイズやインスタンスタイプを変更できます。これにより、適切なサイジングによる利点が得られるだけでなく、変更に必要なほぼすべての運用コストを削減することもできます。

AWS サービスの中には、Amazon Simple Storage Service Intelligent-Tiering のようにタイプやサイズを自動で選べる機能が組み込まれているものもあります。Amazon S3 Intelligent-Tiering では、使用パターンに基づいて、高頻度アクセスと低頻度アクセスの 2 つのアクセスティア間でデータが自動的に移動します。

実装手順

- ワークロードのメトリクスを設定してオブザーバビリティを高める: ワークロードの主要なメトリクスを取得します。これらのメトリクスは、ワークロード出力などのカスタマーエクスペリエンスに関する示唆を提供し、CPU やメモリの使用状況などのリソースのタイプとサイズの違いに合わせて調整されます。コンピューティングリソースの場合、パフォーマンスデータを分析して Amazon EC2 インスタンスのサイズを適切に設定します。アイドル状態のインスタンスと利用率の低いインスタンスを特定します。検索する主なメトリクスは、CPU 使用率とメモリ使用率です (例えば、「Rightsizing with AWS Compute Optimizer and Memory Utilization Enabled」で説明されているように、90% の時間で 40% の CPU 使用率)。4 週間の最大 CPU 使用率およびメモリ使用率が 40%未満のインスタンスを特定します。これらのインスタンスは、コスト削減のために適切なサイズを設定する必要があります。Amazon S3 などのストレージリソースでは、Amazon S3 ストレージレンズを使用できます。これにより、さまざまなカテゴリにわたる 28 のメトリクスをバケットレベルで表示でき、デフォルトでダッシュボードに 14 日間の履歴データを表示できます。Amazon S3 ストレージレンズのダッシュボードは、要約、コスト最適化、またはイベントごとにフィルタリングして特定のメトリクスを分析できます。
- 適切なサイジングの推奨事項を表示する: AWS Compute Optimizer の適切なサイジングの推奨事項を使用するか、コスト管理コンソールで Amazon EC2 の適切なサイジングツールを使用するか、リソースの適切なサイジングでワークロードを調整する AWS Trusted Advisor を確認します。さまざまなリソースの適切なサイジングを行うときは、適切なツールを使用して、Amazon EC2 インスタンス、AWS ストレージクラス、Amazon RDS インスタンスタイプのいずれであれ、適切

なサイジングのガイドラインに従うことが重要です。ストレージリソースには、Amazon S3 ストレージレンズを使用できます。これにより、オブジェクトストレージの使用状況、アクティビティの傾向を可視化し、コストを最適化してデータ保護のベストプラクティスを適用するための実用的な推奨事項を作成できます。Amazon S3 ストレージレンズが組織全体のメトリクスの分析から取得した、状況に応じた推奨事項を使用することで、ストレージを最適化する手順をすぐに実行することができます。

- メトリクスに基づいて自動的にリソースタイプとサイズを選択する: ワークロードメトリクスを使用して、ワークロードリソースを手動でまたは自動で選択します。コンピューティングリソースの場合、AWS Auto Scaling を設定したり、アプリケーション内でコードを実装したりすると、頻繁な変更が要求される場合に必要となる労力を減らすことができるほか、手動プロセスより早く変更を実装できる可能性もあります。1 つの Auto Scaling グループ内で、オンデマンドインスタンスとスポットインスタンスのフリートを起動してオートスケールできます。スポットインスタンスの使用で割引を受けるだけでなく、リザーブドインスタンスまたは Savings Plan を使用して、通常のオンデマンドインスタンス コストの割引料金を受け取ることができます。これらの要素をすべて組み合わせることで、Amazon EC2 インスタンスのコスト削減を最適化し、アプリケーションに必要なスケールとパフォーマンスを判断できます。また、Auto Scaling グループ (ASG) で属性ベースのインスタンスタイプ選択 (ABS) 戦略を使用すると、vCPU、メモリ、ストレージなどの属性セットとしてインスタンスの要件を表現できます。新世代のインスタンスタイプがリリースされたら自動的にこれを使用し、Amazon EC2 スポットインスタンスにより、これまでよりも広い範囲のキャパシティにアクセスすることができます。Amazon EC2 Fleet と Amazon EC2 Auto Scaling が指定した属性に適合するインスタンスを選択して起動するため、手動でインスタンスタイプを選択する必要がなくなります。ストレージリソースでは、Amazon S3 Intelligent-Tiering および Amazon EFS Infrequent Access 機能を使用できます。この機能を使用すると、データアクセスパターンが変更されても、パフォーマンスに影響を与えたり、運用オーバーヘッドを発生させたりすることなく、ストレージクラスを自動的に選択してストレージコストを自動的に削減できます。

リソース

関連ドキュメント:

- AWS Auto Scaling
- AWS での適切なサイジング
- AWS Compute Optimizer
- Amazon CloudWatch の特徴
- CloudWatch のセットアップ

- CloudWatch カスタムメトリクスをパブリッシュする
- Amazon EC2 Auto Scaling の使用を開始する
- Amazon S3 Storage Lens
- Amazon S3 の新しいストレージクラス、S3 Intelligent-Tiering を発表
- Amazon EFS 低頻度アクセス
- SDK で Amazon EC2 インスタンスを起動する

関連動画:

- Right Size Your Services

関連する例:

- Amazon EC2 Fleet 用自動スケーリングでの属性ベースのインスタンスタイプ選択
- スケジュールされたスケーリングを使用して Amazon Elastic Container Service を最適化しコストを削減する
- Amazon EC2 Auto Scaling での予測スケーリング
- Amazon S3 ストレージレンズでコストを最適化し、使用状況を可視化する

COST06-BP04 共有リソースの使用を検討する

複数のビジネスユニットに組織レベルでデプロイ済みのサービスについては、リソースの使用率を高め、総保有コスト (TCO) を削減するために共有リソースの使用を検討してください。共有リソースの使用は、既存のソリューションを使用するか、コンポーネントを共有する、あるいはその両方を行うことで管理とコストを一元化できるコスト効率の高いオプションです。アカウント境界の内側、または専用のアカウントでモニタリング、バックアップ、接続性などの一般的な機能を管理します。また、標準化の実装、重複の削減、複雑さの軽減もコストの削減につながります。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

複数のワークロードが同じ機能を果たす場合は、既存のソリューションと共有コンポーネントを使用して管理を改善し、コストを最適化します。セキュリティのベストプラクティスと組織の規制に従うことでクラウドコストを軽減できるように、本稼働の対象外となるデータベースサーバーやディレクトリサービスなどの既存のリソース (特に共有リソース) を使用することを検討してください。最適

な価値実現と効率化を実現するには、消費が活発なビジネスの適正分野にコストを配分することが重要です (ショーバックとチャージバックを使用する) 。

ショーバックとは、消費者、ビジネスユニット、総勘定元帳勘定、責任を負うその他のエンティティなどの帰属カテゴリにクラウドコストを分類するレポートを指します。ショーバックの目的は、チーム、ビジネスユニット、または個人に、各自が消費したクラウドリソースのコストを示すことです。

チャージバックとは、特定の財務管理プロセスに適した戦略に基づいて、中央のサービスの支出を各コストユニットに割り当てることです。お客様の場合、チャージバックは、1 つの共有サービスアカウントで発生したコストを、お客様の報告プロセスに適したさまざまな財務コストカテゴリに請求します。チャージバックメカニズムを確立することで、さまざまなビジネスユニット、製品、チームで発生したコストを報告できます。

ワークロードは、重大または重大ではないに分類できます。この分類に基づいて、重大度の低いワークロードには一般的な設定の共有リソースを使用します。コストをさらに最適化するには、専有サーバーを重大なワークロード専用に確保します。複数のアカウント間でリソースの共有やプロビジョニングを行うことで、リソースを効率的に管理できます。開発環境、テスト環境、本番環境が分かれていても、安全に共有を行うことが可能であり、組織構造を損なうことはありません。

コンテナ化されたアプリケーションのコストと使用状況の理解を深めて最適化するには、分割コスト配分データを使用します。これにより、アプリケーションによる共有コンピューティングリソースとメモリリソースの消費状況に基づいてアプリケーションコストを個々のエンティティに配分できます。分割コスト配分データを使用することで、Amazon Elastic Container Service (Amazon ECS) または Amazon Elastic Kubernetes Service (Amazon EKS) で実行されているコンテナワークロードでタスクレベルのショーバックとチャージバックを実現できます。

分散型アーキテクチャの場合は、共有サービス VPC を構築します。これにより、各 VPC のワークロードで必要な共有サービスに一元的にアクセスできます。これらの共有サービスには、ディレクトリサービスや VPC エンドポイントなどのリソースを含めることができます。管理オーバーヘッドとコストを削減するには、各 VPC にリソースを構築する代わりに、一元的な場所からリソースを共有します。

共有リソースを使用すると、運用コストの節約、リソース使用率の最大化、一貫性の向上につながります。マルチアカウント設計では、一部の AWS のサービスを一元的にホストし、1 つのハブ内で複数のアプリケーションとアカウントを使用してアクセスすることでコストを節約できます。AWS Resource Access Manager (AWS RAM) を使用することで、VPC サブネットや AWS Transit Gateway アタッチメント、AWS Network Firewall、Amazon SageMaker AI Pipelines など、他の一般的なリソースを共有することができます。マルチアカウント環境の場合、AWS RAM でリソースを作成したら、それを他のアカウントと共有します。

組織は、共有コストに効果的にタグ付けし、コストの大部分がタグ付けされていない、または配分されていないといったことがないよう確認する必要があります。共有コストが効果的に配分されず、共有コストの管理責任を誰も負わない場合、共有クラウドのコストは悪循環に陥る可能性があります。リソース、ワークロード、チーム、または組織レベルのどこでコストが発生しているのかを把握しておく必要があります。これを把握することで、コスト発生場所での実際の価値を達成したビジネス成果と比較して理解することができます。最終的に、組織はクラウドインフラストラクチャの共有によってコスト削減のメリットを享受します。クラウド支出を最適化するために、共有クラウドリソースのコスト配分を奨励してください。

実装手順

- 既存のリソースを評価する: ワークロードに類似のサービスを使用する既存のワークロードを確認します。ワークロードのコンポーネントに応じて、ビジネスロジックや技術的要件で許容される場合は、既存のプラットフォームを検討します。
- AWS RAM でリソース共有を使用し、適宜制限を適用する: AWS RAM を使用して、組織内の他の AWS アカウントとリソースを共有します。リソースを共有する場合、複数のアカウントでリソースを重複する必要がないため、リソースメンテナンスの運用負担を最小限に抑えることができます。またこのプロセスにより、作成したリソースをアカウント内のロールやユーザーの他、他の AWS アカウントとも安全に共有することができます。
- リソースにタグを付ける: コストレポートの対象候補となるリソースにタグを付け、コストカテゴリ内で分類します。コスト配分用にこうしたコスト関連のリソースタグを有効にすると、AWS リソースの使用状況を可視化できます。コストと使用状況の可視性に関して適切な度合いの詳細度を定めることに重点を置き、コスト配分レポートと KPI 追跡によってクラウドの消費行動に影響を与えます。

リソース

関連するベストプラクティス:

- SEC03-BP08 組織内でリソースを安全に共有する

関連ドキュメント:

- What is AWS Resource Access Manager?
- AWS Organizations で使用できる AWS サービス
- 共有可能な AWS リソース
- AWS コストと使用状況レポート (CUR) クエリ

関連動画:

- AWS Resource Access Manager - granular access control with managed permissions
- How to design your AWS cost allocation strategy
- AWS Cost Categories

関連する例:

- 共有サービスのチャージバック方法: An AWS Transit Gateway の例
- How to build a chargeback/showback model for Savings Plans using the CUR
- Using VPC Sharing for a Cost-Effective Multi-Account Microservice Architecture
- Improve cost visibility of Amazon EKS with AWS Split Cost Allocation Data
- AWS 分割コスト配分データにより Amazon ECS および AWS Batch のコストの可視性を向上する

最適な料金モデルを選択する

ワークロードコストモデルを実行する: ワークロードコンポーネントの要件を考慮し、料金のポテンシャルモデルを理解します。コンポーネントの可用性要件を定義します。ワークロードで関数を実行する複数の独立したリソースの有無、経時的に必要となるワークロード要件を確認します。デフォルトのオンデマンド料金モデルと他の適用可能なモデルを使用して、リソースのコストを比較します。リソースまたはワークロードコンポーネントで変更可能なものはすべて考慮します。

アカウントレベルの分析を定期的に実施する: コストモデリングを定期的に実行すると、複数のワークロードにまたがって最適化する機会が確実に得られます。例えば、複数のワークロードでオンデマンドを使用している場合、集計レベルでは変更リスクが低くなり、コミットメントベースの割引を運用すると全体的なコストが低くなります。2 週間から 1 か月の定期的なサイクルで分析を実行することを推奨します。この分析により、調整のための小口購入が可能になり、ワークロードやコンポーネントの変更に合わせて料金モデルの調整を続けることができます。

AWS Cost Explorer のレコメンデーションツールを使用して、コミットメント割引を適用する機会を見つけます。

スポットのワークロードを実行する機会を見つけるには、使用量全体の 1 時間ごとのビューを使用して、定期的に生じる使用量や伸縮性の変化を探します。

料金モデル: AWS には複数の料金モデルがあり、組織のニーズに合った最も費用対効果の高い方法でリソース料金を支払うことができます。次のセクションでは、各購入モデルについて説明します。

- オンデマンドインスタンス
- スポットインスタンス
- コミットメント割引 - Savings Plans
- コミットメント割引 - リザーブドインスタンス/キャパシティ
- 地理的選択
- サードパーティーの契約と料金

オンデマンドインスタンス: こちらはデフォルトの従量制料金モデルです。リソース (EC2 インスタンスや、オンデマンドの DynamoDB などのサービス) を利用する際は、定額料金をお支払いいただくだけで、長期のコミットメントはありません。アプリケーションの需要に応じて、リソースまたはサービスの容量を増減することができます。オンデマンドには時間単位の料金がありますが、サービスによっては 1 秒単位での利用も可能です (Amazon RDS、Linux EC2 インスタンスなど)。オンデマンドは、短期間のワークロード (4 か月間のプロジェクトなど)、定期的に急増するワークロード、中断できない予測不可能なワークロードなどを持つアプリケーションに推奨されます。ほかにもオンデマンドが適しているのは、中断のないランタイムを必要としていて、本番稼働前の環境のように実行時間がコミットメント割引 (Savings Plans またはリザーブドインスタンス) を受けられるほど長くないワークロードです。

スポットインスタンス: スポットインスタンスは、長期のコミットメントを必要とせず、オンデマンド料金の最大 90% 割引で利用できる予備の Amazon EC2 コンピューティングキャパシティです。スポットインスタンスを使用すると、アプリケーションの実行コストを大幅に削減したり、アプリケーションの計算性能を同じ予算でスケールしたりできます。スポットインスタンスはオンデマンドとは異なり、Amazon EC2 で容量を戻す必要がある場合、またはスポットインスタンスの料金が設定された料金を超えた場合に、2 分間の警告により中断されることがあります。スポットインスタンスが中断する時間は平均 5% 未満です。

スポットインスタンスは、キューまたはバッファがある場合や、独立した動作でリクエスト処理をする複数のリソース (Hadoop データ処理など) がある場合に最適です。通常、これらのワークロードは、バッチ処理、ビッグデータと分析、コンテナ化された環境、ハイパフォーマンスコンピューティング (HPC) などの耐障害性、ステートレス性、柔軟性を備えています。テスト環境や開発環境などの重要性の低いワークロードもスポットの候補です。

スポットインスタンスは、Amazon EC2 Auto Scaling グループ、Amazon EMR、Amazon Elastic Container Service (Amazon ECS)、AWS Batch など、複数の AWS サービスにも統合されています。

スポットインスタンスを回収する必要がある場合、Amazon EC2 から、CloudWatch Events を介して、およびインスタンスメタデータで、スポットインスタンスの中断通知により 2 分前の警告が送信されます。この 2 分間を使って、アプリケーションの状態を保存したり、実行中のコンテナを空にしたり、最終ログファイルをアップロードしたり、ロードバランサーからインスタンスを削除したりできます。2 分経つと、スポットインスタンスに休止、停止、終了のいずれかが行えます。

ワークロードにスポットインスタンスを採用する際は、以下のベストプラクティスに従ってください。

- 可能な限り多くのインスタンスタイプに柔軟に対応: インスタンスタイプのファミリーとサイズに柔軟に対応することで、目標とするキャパシティ要件を満たす可能性を高め、コストを可能な限り削減し、中断の影響を最小限に抑えます。
- ワークロードの実行場所に柔軟に対応: 使用できる容量はアベイラビリティーゾーンに応じて異なります。それにより、複数の予備容量プールを利用して目標容量を満たせる可能性が増し、コストを最小限に抑えることができます。
- 継続性に配慮した設計: ステートレスかつ耐障害性のあるワークロードを設計し、EC2 キャパシティの一部が中断されても、ワークロードの可用性やパフォーマンスに影響がないようにします。
- パフォーマンスに応じてワークロードのコスト最適化を最大化するため、オンデマンドおよび Savings Plans/リザーブドインスタンスと組み合わせてスポットインスタンスを使用することが推奨されます。

コミットメント割引 - Savings Plans: AWS で特定量のリソースの使用を予約またはコミットすると、さまざまな方法でコストを削減でき、リソース料金の割引を受けることもできます。Savings Plans を使用すると、1 年または 3 年の間、時間単位の支出コミットメントを行い、リソース全体で割引料金を利用することができます。Savings Plans では、Amazon EC2、AWS Fargate、AWS Lambda などの AWS コンピューティングサービス向けに割引をご用意しています。コミットメントを行った場合、お客様はそのコミットメント額を 1 時間ごとに支払い、その金額がオンデマンド使用量から割引料金で引かれます。例えば、1 時間あたり 50 USD をコミットしていて、オンデマンド使用料が 1 時間あたり 150 USD であるとします。Savings Plans の料金を考慮すると、特定の使用量の割引率は 50% になります。したがって、50 USD のコミットメントで 100 USD のオンデマンド使用料をカバーできます。お客様が支払うのは、50 USD (コミットメント) と、オンデマンド使用料の残りの 50 USD です。

Compute Savings Plans は最も柔軟性のあるプランで、利用料金が最大 66% 割引されます。アベイラビリティーゾーン、インスタンスサイズ、インスタンスファミリー、オペレーティングシステム、テナンシー、リージョン、コンピューティングサービスの全体にわたって、割引が自動適用されます。

Instance Savings Plans の場合、柔軟性は低くなりますが割引率は高くなります (最大 72%)。アベイラビリティーゾーン、インスタンスサイズ、オペレーティングシステム、テナンシーの全体にわたって、割引が自動適用されます。

お支払い方法は 3 つあります。

- 前払いなし: 前払いがなく、時間単位の割引が適用された合計利用時間分を毎月支払います。
- 一部前払い: 前払いなしに比べ割引率が高くなります。利用料の一部を前払いで支払います。その後、時間単位の割引が適用された合計利用時間分を毎月支払います。
- 全額前払い: 全期間の使用料を前払いで支払います。コミットメントでカバーされる使用料については、残りの期間にその他のコストは発生しません。

この 3 つの購入オプションを任意に組み合わせてワークロードに適用できます。

Savings Plans は、まず、割引率の高いものから低いものの順に、購入に使用したアカウントの使用量に適用されます。次に、割引率の高いものから低いものの順に、すべての連結アカウントの使用量に適用されます。

すべての Savings Plans は、管理アカウントのような、使用量やリソースのないアカウントで購入することが推奨されます。そうすれば、Savings Plan がすべての使用の中で最も高い割引料金に適用され、割引額を最大化することができます。

ワークロードと使用量は通常、時間の経過とともに変化します。長期間にわたって、Savings Plans のコミットメントを少額ずつ、継続的に購入することが推奨されます。これにより、割引を最大化するための高いカバレッジレベルを維持できるうえに、コスト削減計画をワークロードや組織で求められる要件と常に一致させることができます。

割引の変動があるため、アカウントにターゲットカバレッジを設定しないでください。カバレッジが低くても、コスト削減の可能性が高くなるとは限りません。アカウントのカバレッジが低い場合でも、ライセンスを取得したオペレーティングシステムで、スモールインスタンスで構成して使用する場合は、割引率がわずか数パーセントにしかならないこともあります。代わりに、Savings Planレコメンデーションツールで利用できる潜在的な削減を追跡してモニタリングします。Cost Explorer で Savings Plans のレコメンデーションを頻繁に確認して (定期的な分析を実行)、削減率の見積りが組織で必要な割引率を下回るまで、コミットメントを引き続き購入します。例えば、潜在的な割引が 20% 未満の状態であることを追跡、モニタリングし、それを上回った場合に購入します。

使用率とカバレッジをモニタリングしますが、変更の検出のみを行います。特定の使用率 (カバレッジ率) を目指さないでください。これは削減額に応じてスケールするとは限らないからで

す。Savings Plans を購入してカバレッジが増加していることを確認し、カバレッジや使用率が減少した場合は、それを定量化して明確に示すようにします。例えば、ワークロードリソースを新しいインスタンスタイプに移行すると、既存のプランの使用率は低下しますが、パフォーマンス上の利点は削減率を上回ります。

コミットメント割引 – リザーブドインスタンス/コミットメント: Savings Plans と同様に、リザーブドインスタンス (RI) は、最小限のリソースを実行するコミットメントに対して最大 72% の割引を提供します。リザーブドインスタンスは、Amazon RDS、Amazon OpenSearch Service、Amazon ElastiCache、Amazon Redshift、DynamoDB で使用できます。Amazon CloudFront と AWS Elemental MediaConvert では、最低利用料金のコミットメントに対して割引が適用されます。現在、リザーブドインスタンスは Amazon EC2 で利用できますが、Savings Plans では柔軟性が向上し、管理オーバーヘッドなしで利用できる同じレベルの割引を提供しています。

リザーブドインスタンスでも、前払いなし、一部前払い、全額前払い、1 年または 3 年の期間といった同様の料金オプションをご用意しています。

リザーブドインスタンスは、リージョンまたは特定のアベイラビリティーゾーンで購入できます。アベイラビリティーゾーンで購入すると、キャパシティ予約が提供されます。

Amazon EC2 はコンバーティブル RI を備えていますが、柔軟性の向上と運用コストの削減のため、すべての EC2 インスタンスで Savings Plans を使用する必要があります。

リザーブドインスタンスの追跡と購入には、同じプロセスとメトリクスを使用する必要があります。アカウント全体の RI のカバレッジは追跡しないことが推奨されます。また、使用率のモニタリングや追跡はせず、Cost Explorer で使用状況レポートを表示して、その表の「純削減額」列を使用することが推奨されます。純削減額が著しく大きい場合は、未使用の RI を修正するためのアクションを実行する必要があります。

EC2 フリート: EC2 フリートは、ターゲットとするコンピューティングキャパシティを定義し、インスタンスタイプやフリートのオンデマンドとスポットのバランスを指定できる機能です。EC2 フリートによって、リソースの最低料金の組み合わせが自動的に起動され、定義されたキャパシティが達成されます。

地域の選択: ソリューションを設計する際、ユーザーに近いコンピューティングリソースの場所を探して、レイテンシー低下とデータ主権の強化を図ることが推奨されます。グローバルな利用者のニーズに応えるためには、複数のロケーションを使用する必要があります。コストを最低限に抑えられる地理的ロケーションを選択する必要もあります。

AWS クラウドインフラストラクチャはリージョンとアベイラビリティーゾーンを中心に構築されます。リージョンは世界各国にある物理的な場所であり、その中に複数のアベイラビリティーゾーンが

あります。アベイラビリティーゾーンは、1 つ、または複数の独立したデータセンターで構成されています。各データセンターには冗長電源、ネットワーキング、および接続が装備されており、個別の施設に格納されています。

各 AWS リージョンは、地域の市場の条件下で運用されており、リソースの料金はリージョンごとに異なります。世界的に最小料金で稼働できるように、ソリューションのコンポーネントまたは全体を運用する特定のリージョンを選択します。AWS Simple Monthly Calculator を使用すると、さまざまな リージョンのワークロードにかかるコストを見積もることができます。

サードパーティー契約と料金: クラウドでサードパーティー製のソリューションまたはサービスを利用する場合、料金構造とコスト最適化の結果を連動させることが重要です。料金は、コスト最適化の結果とサービスの価値に合わせてスケールする必要があります。この一例として、削減率を使用したソフトウェアがあります。削減率 (結果) が高くなるほど請求額も上がるというものです。請求額に合わせてスケールする契約は、特定の請求書のあらゆる部分で結果が得られない限り、ほとんどの場合コストの最適化と連動していません。例えば、 Amazon EC2 の推奨事項を提供し、請求全体のある割合が課金されるソリューションでは、メリットを提供しない他のサービスを利用すると、その分増加します。もう 1 つの例は、管理するリソースのコストの割合に応じて課金されるマネージドサービスです。インスタンスサイズを大きくすると常に管理作業が増えるわけではありませんが、請求額が高くなります。これらのサービス料金設定に、コスト最適化プログラムまたはサービス機能が含まれていることを承知のうえで効率性を向上させます。

ベストプラクティス

- COST07-BP01 料金モデルの分析を実行する
- COST07-BP02 コストに基づいてリージョンを選択する
- COST07-BP03 費用対効果の高い条件を提供するサードパーティーの契約を選択する
- COST07-BP04 このワークロードのすべてのコンポーネントに対して料金モデルを実装する
- COST07-BP05 管理アカウントレベルで料金モデル分析を実行する

COST07-BP01 料金モデルの分析を実行する

ワークロードの各コンポーネントを分析します。コンポーネントとリソースを長期間実行するか (コミットメント割引)、動的に短期間実行するか (スポットまたはオンデマンド) を決定します。コスト管理ツールのレコメンデーションを使用して、ワークロードに対して分析を行います。これらのレコメンデーションにビジネスルールを適用して、高いリターンを実現します。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

AWS には複数の料金モデルがあり、組織のニーズに合い、製品に応じた最も費用対効果の高い方法でリソース料金を支払うことができます。チームと協力して最適な料金モデルを決定します。可用性に基づいて決定すると、複数のオプションを組み合わせた料金モデルになることもよくあります

オンデマンドインスタンスでは、実行しているインスタンスに応じて、コンピューティングまたはデータベース容量に対して時間単位または秒単位 (最小 60 秒) で支払うことができます。長期契約や前払いは不要です。

Savings Plans は、1 年または 3 年の期間で一定の使用量 (USD/時間、で計算) をコミットメントする代わりに、Amazon EC2、Lambda、AWS Fargate を低料金で利用できる柔軟な料金モデルです。

スポットインスタンスは、予備のコンピューティング容量を時間単価の割引価格 (オンデマンド価格の最大 90% オフ) でリクエストできる Amazon EC2 の料金の仕組みです。前払いのコミットメントはありません。

リザーブドインスタンスでは、容量に対する前払いにより、最大 75% の割引を受けることができます。詳細については、「予約によるコストの最適化」を参照してください。

本稼働、品質、開発の各環境に関連するリソースに Savings Plans を含めることもできます。また、サンドボックスリソースは必要なときにのみ電源が入るため、その環境内のリソースにオンデマンドモデルを選択することもできます。Amazon スポットインスタンスを使用して Amazon EC2 のコストを削減したり、Compute Savings Plans を使用して Amazon EC2、Fargate、Lambda のコストを削減したりします。AWS Cost Explorer レコメンデーションツールは、Saving Plans によるコミットメント割引の機会を提供します。

過去に Amazon EC2 のリザーブドインスタンスを購入している場合、あるいは組織内でコスト配分を実施している場合、今後も当面の間は Amazon EC2 リザーブドインスタンスをご利用いただけます。ただし、より柔軟にコストを節約するため、いずれは Savings Plans を使用する戦略に切り替えることが推奨されます。AWS Cost Management の Savings Plans (SP) レコメンデーションは、更新することでいつでも新しい Savings Plans レコメンデーションを作成できます。リザーブドインスタンス (RI) を使用すると、Amazon RDS、Amazon Redshift、Amazon ElastiCache、Amazon OpenSearch Service のコストを削減できます。Savings Plans とリザーブドインスタンスには、全額前払い、一部前払い、前払いなしの 3 つのオプションがあります。AWS Cost Explorer RI および SP 購入レコメンデーションで提供されたレコメンデーションを使用します。

スポットのワークロードを実行する機会を見つけるには、使用量全体の 1 時間ごとのビューを使用して、定期的に生じる使用量や伸縮性の変化を探します。スポットインスタンスは、フォールトトレ

ラントで柔軟性があるさまざまなアプリケーションに使用できます。これには、ステートレスウェブサーバー、API エンドポイント、ビッグデータアプリケーションや分析アプリケーション、コンテナ化されたワークロード、CI/CD、その他柔軟性の高いワークロードなどがあります。

Amazon EC2 および Amazon RDS インスタンスを、使用していないとき (就業後や週末) にオフにできるかを分析します。このアプローチによって、24 時間 365 日使用する場合と比較して、70% 以上のコストを削減できます。特定の時間にのみ使用できるようにする必要がある Amazon Redshift クラスターがある場合は、そのクラスターを一時停止して、後で再開できます。Amazon Redshift クラスターや Amazon EC2 および Amazon RDS インスタンスが停止すると、コンピューティングに対する請求が停止され、ストレージ料金のみが適用されます。

オンデマンドキャパシティ予約 (ODCR) は料金割引ではないことに注意してください。インスタンスをリザーブドキャパシティで実行しているかどうかにかかわらず、オンデマンドの場合と同等の料金がキャパシティ予約に課金されます。キャパシティ予約は、実行する予定のリソースに対して十分な容量を提供する必要がある場合に、検討します。ODCR は不要になればキャンセルできるため、長期コミットメントと結びつける必要はありませんが、Savings Plans またはリザーブドインスタンスが提供する割引のメリットを受けることもできます。

実装手順

- ワークロードの伸縮性を分析する: Cost Explorer の時間単位の粒度またはカスタムダッシュボードを使用して、ワークロードの伸縮性を分析します。実行中のインスタンス数の定期的な変化を調べます。短期間のインスタンスはスポットインスタンスまたはスポットフリートの候補です。
  - Well-Architected ラボ: Cost Explorer
  - Well-Architected ラボ: コストの可視化
- 既存の料金契約を見直す: 現在の契約やコミットメントを長期間のニーズの観点から見直します。現在締結しているものと、それらのコミットメントをどの程度使用しているかを分析します。既存の契約による割引やエンタープライズ契約を活用します。エンタープライズ契約では、ニーズに最適な契約を調整できます。長期コミットメントの場合は、リザーブド料金割引、特定のインスタンスタイプに対するリザーブドインスタンスまたは Savings Plans、インスタンスファミリー、AWS リージョン、アベイラビリティーゾーンを検討します。
- コミットメント割引分析を実行する: アカウントで Cost Explorer を使用して Savings Plans とリザーブドインスタンスのレコメンデーションを確認します。必要な割引を適用し、リスクを認識したうえで、正しいレコメンデーションを実装していることを確認するには、Well-Architected ラボに従ってください。

リソース

関連ドキュメント:

- リザーブドインスタンスのレコメンデーションへのアクセス
- インスタンス購入オプション
- AWS エンタープライズ

関連動画:

- Save up to 90% and run production workloads on Spot

関連する例:

- Well-Architected ラボ: Cost Explorer
- Well-Architected ラボ: コストの可視化
- Well-Architected ラボ: 料金モデル

COST07-BP02 コストに基づいてリージョンを選択する

リソースの料金は各リージョンで異なる場合があります。リージョンによるコストの差異を特定し、レイテンシー、データレジデンシー、データ主権に関する要件を満たす場合にのみ、よりコストの高いリージョンにデプロイします。リージョンコストを織り込むことで、このワークロードに対して支払う料金の合計を最低限に抑えることができます。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

AWS クラウドインフラストラクチャはグローバルで、世界中の複数の場所でホストされ、AWS リージョン、アベイラビリティーゾーン、ローカルゾーン、AWS Outposts、Wavelength Zone を中心に構築されています。リージョンとは世界中の物理的な場所であり、各リージョンは、AWS が複数のアベイラビリティーゾーンを設置している地理的に離れた地域です。各リージョン内の複数の独立した場所であるアベイラビリティーゾーンは、1 つ以上の独立したデータセンターで構成されています。各データセンターは、冗長性のある電源、ネットワーク、接続を備えています。

各 AWS リージョンは現地マーケットの条件内で運用されており、土地代、回線代、電気代、税金などのコストが異なるため、リソース料金は各リージョンで異なります。世界的に最小料金で稼働で

きるように、ソリューションのコンポーネントまたは全体を運用する特定のリージョンを選択します。AWS 見積りツールを使用して、ロケーションタイプ (リージョン、Wavelength ゾーン、ローカルゾーン) とリージョンごとにサービスを検索し、さまざまなリージョンでワークロードのコストを見積もります。

ソリューションを設計する際、ユーザーに近いコンピューティングリソースの場所を探して、レイテンシー低下とデータ主権の強化を図ることが推奨されます。ビジネス、データプライバシー、パフォーマンス、セキュリティの要件に基づいて、地理的場所を選択します。エンドユーザーが世界中にいるアプリケーションの場合は、複数の場所を使用します。

データプライバシー、セキュリティ、ビジネス要件に義務がない場合は、AWS のサービスの料金がより安価なリージョンを使用して、ワークロードをデプロイします。例えば、デフォルトのリージョンが アジアパシフィック (シドニー)(`ap-southwest-2`) であり、他のリージョンを使用するにあたっての制約 (データプライバシー、セキュリティなど) がない場合、重要ではない (開発とテスト) Amazon EC2 インスタンスを米国東部 (バージニア北部)(`us-east-1`) リージョンにデプロイすると、コストを抑えることができます。

	コンプライアンス	レイテンシー	コスト	サービス/機能
リージョン 1	✓	15 MS	$$	✓
リージョン 2	✓	20 MS	$$$	X
リージョン 3	✓	80 MS	$	✓
リージョン 4	✓	15 MS	$$	✓
リージョン 5	✓	20 MS	$$$	X
リージョン 6	✓	15 MS	$	✓
リージョン 7	✓	80 MS	$	✓
リージョン 8	✓	15 MS	$	X

リージョン機能マトリックス表

前述のマトリックス表から、他のリージョンに比べてレイテンシーが低く、サービスが利用可能で、コストが最も低いリージョンであるため、このシナリオではリージョン 6 が最適なオプションであることがわかります。

実装手順

- AWS リージョンの料金を確認する: 現在のリージョンのワークロードコストを分析します。サービスおよび使用タイプ別の最も高いコストから、利用可能な他のリージョンのコストを計算します。予測される費用削減効果がコンポーネントまたはワークロードの移動コストを上回っている場合は、新しいリージョンに移行します。
- 複数のリージョンにデプロイする場合の要件を確認する: ビジネス要件と義務 (データプライバシー、セキュリティ、パフォーマンス) を分析して、複数リージョンを使用すべきでない制約があるかどうかを確認します。単一リージョンを使用するよう制限する義務がない場合は、複数のリージョンを使用します。
- 必要なデータ転送を分析する: リージョンを選択するときは、データ転送コストを考慮します。データは顧客とリソースの近くに置いてください。データ転送が最小限でデータの流れがよい、よりコストの低い AWS リージョンを選択します。データ転送のビジネス要件に応じて、Amazon CloudFront、AWS PrivateLink、AWS Direct Connect、AWS Virtual Private Network を使用することで、ネットワークコストの削減、パフォーマンスの向上、セキュリティの強化を実現できます。

リソース

関連ドキュメント:

- リザーブドインスタンスのレコメンデーションへのアクセス
- Amazon EC2 の料金
- インスタンス購入オプション
- リージョン表

関連動画:

- Save up to 90% and run production workloads on Spot

関連する例:

- 一般的なアーキテクチャでのデータ転送コストの概要
- グローバルデプロイにおけるコストの考慮事項
- ワークロードに応じたリージョンを選択する際の注意点

COST07-BP03 費用対効果の高い条件を提供するサードパーティーの契約を選択する

コスト効率に優れた契約と条件により、これらのサービスのコストが、提供されるメリットに見合ったものとなります。組織に追加のメリットを提供するときに、それに合わせてスケールする契約と料金を選択します。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

クラウド環境のコスト管理に役立つさまざまな製品が流通しています。こうした製品は、ターゲットとなる顧客の要件に応じて、コストガバナンスやコスト可視化を重視したものや、コスト最適化を重視したものなど、機能面で違いが見られる場合があります。効果的なコスト最適化とガバナンスの鍵を握る要因の 1 つは、料金モデルが適正で、必要な機能が揃った適切なツールを使用することです。製品ごとに料金モデルは異なります。毎月の請求総額の一定の割合を支払うものもあれば、実際の節約額の一定の割合を支払うものもあります。必要な分だけ支払う従量課金制が理想的です。

クラウドでサードパーティーのソリューションやサービスを利用する場合は、期待する成果に合わせて料金体系を選ぶことが重要です。料金は、コスト最適化の結果とサービスの価値に合わせてスケールする必要があります。例えば、実際に節約できたコストの一部を支払う成功報酬型ソフトウェアの場合、節約率 (成果) が上がるほど、請求額も高くなります。支出負担が増えるに従い、支払う金額も増えるライセンス契約は、コスト最適化の点で必ずしも最適解とは限りません。ただし、請求書のあらゆる項目で優遇を受けられる場合、こうした変動料金が妥当なケースもあるかもしれません。

例えば、Amazon EC2 のレコメンデーションを提供し、請求総額の一定割合を課金するソリューションでは、優遇なしの他のサービスを利用すると、割高になる場合があります。もう 1 つの例は、管理対象となるリソースのコストを一定の割合で支払うマネージドサービスです。インスタンスサイズが大きくなっても必ずしも管理の負担が増えるわけではありませんが、請求額は高くなります。こうしたサービス料金設定に、コスト最適化のプログラムや、効率を向上するサービス機能が含まれていることを確認してください。

顧客は市場に流通しているこれらの製品を比較的高度である、または使いやすいと感じるかもしれません。こうした製品のコストを考慮し、長期的に見てコスト最適化の可能性があるかどうかを考える必要があります。

実装手順

- サードパーティーの契約と条件を分析する: サードパーティーの契約における料金を確認します。さまざまな使用レベルに応じたモデリングを行い、新しいサービスの使用や、ワークロードの増加

による現在のサービスの増加など、新たなコストを考慮します。追加コストによってビジネスに必要なメリットが得られるかどうかを判断します。

リソース

関連ドキュメント:

- リザーブドインスタンスのレコメンデーションへのアクセス
- インスタンス購入オプション

関連動画:

- Save up to 90% and run production workloads on Spot

COST07-BP04 このワークロードのすべてのコンポーネントに対して料金モデルを実装する

永続的に実行されるリソースでは、Savings Plans やリザーブドインスタンスなどのリザーブドキャパシティを利用する必要があります。短期的な使用には、スポットインスタンスまたはスポットフリートを使用するように設定します。オンデマンドインスタンスは、リザーブドキャパシティに対して長時間稼働しない、中断することのできない短期ワークロードに対してのみ使用します (リソースタイプに応じて、期間の 25% から 75%)。

このベストプラクティスを活用しない場合のリスクレベル: 低

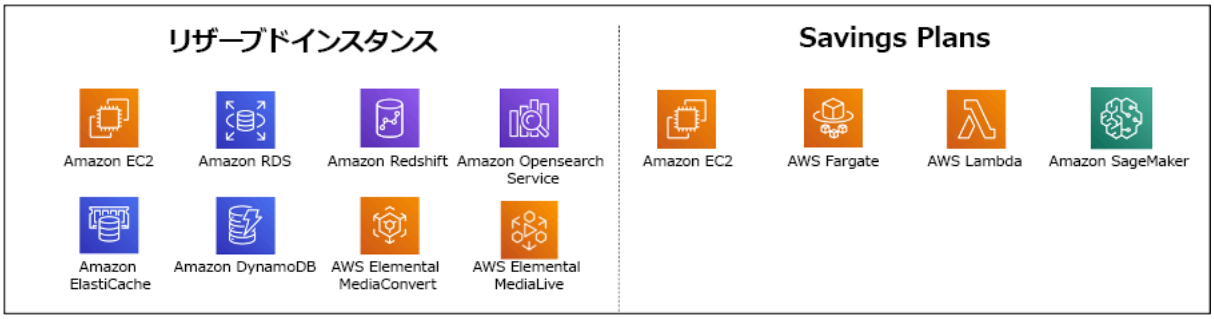
実装のガイダンス

コスト効率を上げるために、AWS は過去の使用状況に基づいて確約利用 (コミットメント) のレコメンデーションをいくつか提示します。これらのレコメンデーションを参考にして、実際に節約できるコストと、そのコミットメントの活用法を理解できます。これらのサービスをオンデマンドまたはスポットで利用することも、一定期間の利用を確約してリザーブドインスタンス (RI) や Savings Plans (SP) でオンデマンドコストを削減することもできます。ワークロードを最適化するには、各ワークロードコンポーネントや複数の AWS サービスだけでなく、該当するサービスのコミットメント割引、購入オプション、スポットインスタンスについても理解する必要があります。

ワークロードのコンポーネントの要件を検討し、これらのサービスのさまざまな料金モデルを理解しましょう。これらのコンポーネントの可用性要件を定義します。ワークロードで関数を実行する複数

の独立したリソースの有無、経時的に必要となるワークロード要件を確認します。デフォルトのオンデマンド料金モデルと他の適用可能なモデルを使用して、リソースのコストを比較します。リソースまたはワークロードコンポーネントで変更可能なものはすべて考慮します。

例えば、AWS のこのウェブアプリケーションアーキテクチャを検討してみましょう。このワークロードのサンプルは、Amazon Route 53、AWS WAF、Amazon CloudFront、Amazon EC2 インスタンス、Amazon RDS インスタンス、ロードバランサー、Amazon S3 ストレージ、Amazon Elastic File System (Amazon EFS) など、複数の AWS サービスで構成されています。これらのサービスをそれぞれ見直し、さまざまな料金モデルでコストをどれくらい削減できるのかを確認する必要があります。RI または SP を利用できるものもあれば、オンデマンドでしか利用できないものもあります。次の図からわかるように、AWS の一部のサービスは利用を確約し、RI または SP を使用できます。



リザーブドインスタンスと Savings Plans を使用してコミットされた AWS サービス

実装手順

- 料金モデルを実装する: 分析結果を使用して、Savings Plans またはリザーブドインスタンスの購入、スポットインスタンスの実装を行います。コミットメントの初回購入時には、リストの上位 5 件または 10 件のレコメンデーションを選択し、翌月または翌々月までの結果をモニタリングして分析します。このプロセスは AWS Cost Management Console が案内してくれます。コンソールから RI または SP のレコメンデーションを確認し、その内容 (タイプ、支払い、期間) をカスタマイズし、時間単位の確約利用料 (1 時間あたり 20 USD など) を確認して、カートに追加します。割引は、対象となる使用量に自動的に適用されます。コミットメント割引で定期的に少量を購入します (例: 2 週間ごとまたは 1 か月ごと)。中断可能またはステートレスなワークロードにスポットインスタンスを実装します。最後に、Amazon EC2 オンデマンドインスタンスを選択し、残りの要件にリソースを割り当てます。
- ワークロードレビューサイクル: 特に料金モデルカバレッジを分析するワークロードのレビューサイクルを実装します。ワークロードが必要なカバレッジを達成したら、部分的に (数か月ごと)、または組織の使用状況の変化に応じて、追加のコミットメント割引を購入します。

リソース

関連ドキュメント:

- Understanding your Savings Plans recommendations
- リザーブドインスタンスのレコメンデーションへのアクセス
- リザーブドインスタンスの購入方法
- インスタンス購入オプション
- スポットインスタンス
- AWS の他のサービスの予約モデル
- Savings Plans Supported Services

関連動画:

- Save up to 90% and run production workloads on Spot

関連する例:

- Savings Plans を購入する前に考慮すべきことは何ですか?
- Cost Explorer で使用率とコストを分析する方法を教えてください。

COST07-BP05 管理アカウントレベルで料金モデル分析を実行する

請求やコスト管理ツールをチェックして、コミットメントや予約を利用した推奨割引を確認し、管理アカウントレベルで定期的に分析を実行します。

このベストプラクティスを活用しない場合のリスクレベル: 低

実装のガイダンス

コストモデリングを定期的に実行すると、複数のワークロード全体を最適化する機会が得られます。例えば、複数のワークロードでオンデマンドインスタンスを使用している場合、集計レベルでは変更リスクが低くなり、コミットメントベースの割引を運用すると全体的なコストが低くなる場合があります。2 週間から 1 か月の定期的なサイクルで分析を実行することを推奨します。これにより、調整のための小口購入が可能になり、ワークロードやコンポーネントの変更に合わせて料金モデルの調整を続けることができます。

AWS Cost Explorer のレコメンデーションツールを使用して、管理アカウントでコミットメント割引を適用する機会を見つけます。管理アカウントレベルでのレコメンデーションは、リザーブドインスタンス (RI) または Savings Plans (SP) を持つ AWS 組織内のすべてのアカウントの使用量を考慮して計算されます。また、割引共有が有効になったときに計算され、アカウント全体で節約を最大化できるコミットメントを推奨します。

管理アカウントレベルでの購入は、多くの場合、最大限の節約を目指して最適化されますが、特定の連結アカウントでの利用に最初に割引を適用したい場合など、連結アカウントレベルで SP を購入することを検討する状況もあります。メンバーアカウントの推奨事項は、独立した各アカウントでの削減を最大化するために、個人アカウントレベルで計算されます。アカウントが RI と SP の両方のコミットメントを所有している場合、それらは次の順序で適用されます。

1. ゾーン RI
2. 標準 RI
3. コンバーティブル RI
4. Instance Savings Plan
5. Compute Savings Plan

管理アカウントレベルで SP を購入した場合、割引率の高い順に節約が適用されます。管理アカウントレベルの SP は、すべての連結アカウントを調べて、割引が最も高いところに節約を適用します。節約が適用される場所を制限したい場合は、連結アカウント単位で Savings Plan を購入すると、そのアカウントが対象となるコンピューティングサービスを実行しているときはいつでも、割引が最初に適用されます。アカウントが対象となるコンピューティングサービスを実行していない場合、割引は同じ管理アカウントにある他の連結アカウント間で分配されます。割引共有はデフォルトでオンになっていますが、必要に応じてオフにできます。

一括請求ファミリーでは、Savings Plans は、まず所有者アカウントの使用に適用され、次に他のアカウントの使用に適用されます。これは共有が有効になっている場合にのみ発生します。Savings Plans は、削減率が最も高いものがまず適用されます。削減率が等しい使用が複数ある場合、Savings Plans は、Savings Plans の割合が最も低い使用にまず適用されます。Savings Plans は、残りの使用量がなくなるか、コミットメントが使い果たされるまで引き続き適用されます。残りの使用はオンデマンド価格で課金されます。AWS コスト管理の Savings Plans レコメンデーションを更新して、いつでも新しい Savings Plans レコメンデーションを作成できます。

インスタンスの柔軟性を分析すると、レコメンデーションに沿ってコミットできます。コストモデリングを作成するにあたって、さまざまなリソースオプションの可能性を含めたワークロードの短期

コストを分析し、AWS 料金モデルの分析を行い、それらをビジネス要件に合わせて、総保有コストとコスト最適化の機会を明らかにします。

実装手順

コミットメント割引分析を実行する: アカウントで Cost Explorer を使用して、Savings Plans とリザーブドインスタンスのレコメンデーションを確認します。Savings Plan のレコメンデーションを理解し、月次費用の見積もりと月次節約額の見積もりを行っていることを確認します。管理アカウントレベルでのレコメンデーションを確認します。レコメンデーションは、アカウント間で最大限の節約を実現するために、RI または Savings Plans の割引共有が有効になっている AWS 組織内におけるすべてのメンバーアカウントの使用量を考慮して計算されます。Well-Architected ラボに従って、必要な割引を適用し、リスクを認識したうえで、正しいレコメンデーションを実装していることを確認します。

リソース

関連ドキュメント:

- AWS の料金のしくみはどのようになっていますか?
- インスタンス購入オプション
- Savings Plans の概要
- Saving Plan recommendations
- リザーブドインスタンスのレコメンデーションへのアクセス
- Saving Plans 推奨事項を理解する
- AWS の使用に Savings Plans が適用される仕組み
- 組織の一括請求 (コンソリデーティッドビリング) 全体に適用される Savings Plan の料金には、どのようなメリットがありますか?
- 共有リザーブドインスタンスと Savings Plans の割引の有効化

関連動画:

- Save up to 90% and run production workloads on Spot

関連する例:

- Savings Plans を購入する前に考慮すべきことは何ですか?

- ローリング Savings Plans を使用してコミットメントのリスクを軽減する方法
- スポットインスタンスの使用が適切なケース

データ転送を計画する

クラウドの利点は、マネージド型のネットワークサービスであることです。スイッチ、ルーター、その他の関連するネットワーク機器などのフリートの管理や運用は不要になります。クラウド内のネットワーキングリソースは CPU とストレージと同じように消費され、同じように実際に使用した分だけを支払うことになります。クラウドでコストを最適化するには、ネットワーキングリソースを効率的に使用する必要があります。

ベストプラクティス

- COST08-BP01 データ転送モデリングを実行する
- COST08-BP02 データ転送コストを最適化するコンポーネントを選択する
- COST08-BP03 データ転送コストを削減するサービスを実装する

COST08-BP01 データ転送モデリングを実行する

組織の要件を取りまとめ、ワークロードとその各コンポーネントのデータ転送モデリングを実行します。これにより、現在のデータ転送要件に対する最低コストを特定できます。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

クラウドでソリューションを設計する際、習慣的にオンプレミスのデータセンターを使用してアーキテクチャを設計してしまったり、知識が不足していたりするせいで、データ転送料金を見落としがちです。AWS のデータ転送料金は、送信元、送信先、トラフィック量によって決まります。設計段階からこれらの料金を織り込めば、コスト削減につながる可能性があります。総保有コスト (TCO) を正確に見積もるには、ワークロードにおけるデータ転送の発生箇所、転送コスト、関連するメリットを把握することがきわめて重要です。これにより、十分な情報に基づいてアーキテクチャ設計上の変更や承諾の決定ができます。例えば、アベイラビリティーゾーン間でデータをレプリケートするマルチアベイラビリティーゾーンを設定したとします。

ワークロードでデータを転送するサービスコンポーネントをモデリングし、これが、求められる信頼性と耐障害性を実現するために許容されるコストであるか (両方のアベイラビリティーゾーンのコンピューティングとストレージに支払うのと同様であるか) を判断します。さまざまな使用量レベルで

コストをモデリングします。ワークロード使用量は経時的に変化します。また、サービスの種類ごとに異なるレベルで費用対効果が向上する場合があります。

データ転送をモデリングする際は、取り込まれるデータの量と転送元を考慮します。また、処理されるデータ量と、必要なストレージやコンピューティングのキャパシティについても検討してください。モデリング中は、ワークロードのアーキテクチャに則したネットワークのベストプラクティスに従い、見込まれるデータ転送コストを最適化します。

AWS 料金見積りツール を使用して、特定の AWS サービスのコストの見積りと、予想されるデータ転送を確認できます。ワークロードを (テスト目的で、または実稼働前の環境で) 既に実行している場合は、AWS Cost Explorer または AWS Cost and Usage Report (CUR) を使用してデータ転送コストを把握し、モデル化します。PoC (概念実証) を設定するか、またはワークロードをテストして、現実的な条件でシミュレートされた負荷を用いてテストを実行します。ワークロードのさまざまな需要に応じてコストをモデルリングできます。

実装手順

- 要件を特定する: 送信元と送信先の間で予定されているデータ転送の、主な目標とビジネス要件は何ですか? 最終的にどのようなビジネス成果を期待していますか? ビジネス要件を収集し、期待される成果を定義します。
- 送信元と送信先を特定する: データ転送の送信元と送信先はどこですか (AWS リージョン内の転送、AWS サービスへの転送、インターネットへの転送など)?
  - AWS リージョン内のデータ転送
  - AWS リージョン 間のデータ転送
  - インターネットへのデータ転送
- データ分類を特定する: 転送されるデータはどのように分類されますか? データの種類は? データの大きさは? データ転送の頻度は? 機密データですか?
- 使用する AWS サービスまたはツールを特定する: このデータ転送にはどの AWS サービスを使用しますか? プロビジョニング済みのサービスを別のワークロードに使用できますか?
- データ転送コストを計算する: 以前に作成したデータ転送モデリングの AWS 料金を使用して、ワークロードのデータ転送コストを計算します。ワークロードの使用量が増減した場合の、使用量別のデータ転送コストを計算します。ワークロードアーキテクチャに複数のオプションがある場合は、比較のために各オプションのコストを計算します。
- コストを結果にリンクする: 発生したデータ転送コストごとに、ワークロードで達成した結果を指定します。コンポーネント間の転送であればデカップリングのため、アベイラビリティーゾーン間の転送であれば冗長性のためかもしれません。

- データ転送モデルを作成する: すべての情報を収集したら、複数のユースケースやさまざまなワークロードの基準となる、データ転送の概念モデルを作成します。

リソース

関連ドキュメント:

- AWS キャッシュソリューション
- AWS の料金
- 「Amazon EC2 の料金」
- Amazon VPC の料金
- Understanding data transfer charges

関連動画:

- Monitoring and Optimizing Your Data Transfer Costs
- S3 Transfer Acceleration

関連する例:

- 一般的なアーキテクチャでのデータ転送コストの概要
- AWS 規範ガイダンス

COST08-BP02 データ転送コストを最適化するコンポーネントを選択する

すべてのコンポーネントを選択し、データ転送コストを低減するようにアーキテクチャを設計します。これには、ワイドエリアネットワーク (WAN) 最適化やマルチアベイラビリティーゾーン (AZ) 設定などのコンポーネントの使用が含まれます。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

データ転送を念頭に置いたアーキテクチャでは、データ転送コストを最小限に抑えることができます。このアーキテクチャでは、コンテンツ配信ネットワークを使用してユーザーに近いデータを特定したり、お客様のプレミスと AWS をつなぐ専用ネットワーク接続を使用したりする場合がありま

す。WAN の最適化やアプリケーションの最適化によって、コンポーネント間で転送されるデータ量を減らすこともできます。

AWS クラウドとの間やその内部でデータを転送する場合、データ転送を最適化する適切な AWS サービスを選択するために、さまざまなユースケース、データの性質、利用可能なネットワークリソースに基づいて転送先を把握することが不可欠です。AWS は、多様なデータ移行要件に対応する幅広いデータ転送サービスを提供しています。組織内のビジネスニーズに基づいて、適切なデータストレージとデータ転送オプションを選択します。

ワークロードアーキテクチャを計画または確認するときは、次の点を考慮してください。

- AWS 内の VPC エンドポイントを使用する: VPCエンドポイントにより、VPC とサポートされている AWS サービス間のプライベート接続が可能になります。これにより、データ転送コストが発生する可能性のある公開インターネットの使用を回避できます。
- NAT ゲートウェイを使用する: NAT ゲートウェイを使用して、プライベートサブネット内のインスタンスがインターネットまたは VPC 外のサービスに接続できるようにします。NAT ゲートウェイの背後にあるリソースのうち、最大量のトラフィックを送信しているリソースのアベイラビリティーゾーンが NAT ゲートウェイと同じかどうかを確認します。違う場合は、そのリソースと同じアベイラビリティーゾーンに新しい NAT ゲートウェイを作成し、AZ 間のデータ転送料金を削減します。
- AWS Direct Connect を使用する: AWS Direct Connect は公開インターネットをバイパスして、オンプレミスネットワークと AWS との間にプライベート接続を直接確立します。インターネットを介して大量のデータを転送するよりも、この方が高コスト効率で確実です。
- リージョンの境界をまたぐデータ転送は回避する: 通常、(特定のリージョンから別のリージョンへの) AWS リージョン 間のデータ転送には、料金が発生します。複数のリージョンをまたいで転送する場合は、慎重に検討したうえで決断してください。詳細については、「複数リージョンのシナリオ」を参照してください。
- データ転送をモニタリングする: Amazon CloudWatch と VPC フローログを使用して、データ転送とネットワークの使用状況に関する詳細情報をキャプチャします。VPC 内でネットワークインターフェイスとの間を行き来するネットワークトラフィックについて、IP アドレスや範囲などの、キャプチャされた情報を分析します。
- ネットワーク使用状況を分析する: AWS Cost Explorer、CUDOS Dashboard、CloudWatch など、測定とレポートのためのツールを使用して、ワークロードのデータ転送コストを把握します。

実装手順

- データ転送用のコンポーネントを選択する: COST08-BP01 データ転送モデリングを実行する で説明したデータ転送モデリングを使用して、データ転送コストが最も大きい場所や、ワークロードの使用状況が変わった場合の場所に焦点を当てます。データ転送の必要性を排除または削減 (またはコストを削減) する代替アーキテクチャや追加のコンポーネントを探します。

リソース

関連するベストプラクティス:

- COST08-BP01 データ転送モデリングを実行する
- COST08-BP03 データ転送コストを削減するサービスを実装する

関連ドキュメント:

- クラウドへのデータ移行
- AWS キャッシュソリューション
- Amazon CloudFront でコンテンツ提供を高速化する

関連する例:

- 一般的なアーキテクチャでのデータ転送コストの概要
- AWS Network Optimization Tips
- Optimize performance and reduce costs for network analytics with VPC Flow Logs in Apache Parquet format

COST08-BP03 データ転送コストを削減するサービスを実装する

データ転送コストを削減するサービスを実装します。例えば、エッジロケーションやコンテンツ配信ネットワーク (CDN) を使用してエンドユーザーにコンテンツを配信する、アプリケーションサーバーまたはデータベースの前にキャッシュレイヤーを構築する、クラウドへの接続に VPN ではなく専用ネットワーク接続を使用するなどです。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

ネットワークデータ転送の使用量を最適化するのに役立つさまざまな AWS サービスがあります。ワークロードのコンポーネント、種類、クラウドアーキテクチャにもよりますが、これらのサービスはクラウド上でのトラフィックの圧縮、キャッシュ、共有、分散に役立ちます。

- Amazon CloudFront は、低レイテンシーかつ高速の転送速度でデータを転送する、グローバルなコンテンツ配信ネットワークです。世界中のエッジロケーションでデータをキャッシュすることで、お客様のリソースの負荷を軽減します。CloudFront を使用することで、レイテンシーを最低限に抑え、世界中の多数のユーザーにコンテンツを配信するための管理労力を軽減できます。経時的に使用量を増やす予定がある場合、Security Savings Bundle を使用すると、CloudFront の使用量を最大 30% 節約できます。
- AWS Direct Connect により、AWS への専用ネットワーク接続を確立できます。このサービスにより、ネットワークコストの削減、帯域幅の増加、インターネット経由の接続よりも安定したネットワーク接続が実現します。
- AWS VPN を使用すると、プライベートネットワークと AWS グローバルネットワークとの間に安全なプライベート接続を確立できます。シンプルな接続とフルマネージド型の伸縮自在なサービスは、小規模なオフィスやビジネスパートナーに最適です。
- VPC エンドポイントにより、プライベートネットワークを利用した AWS サービス間の接続が可能になり、パブリックデータ転送と NAT ゲートウェイのコストを削減できます。ゲートウェイ VPC エンドポイントでは時間単位の料金は発生せず、Amazon S3 と Amazon DynamoDB がサポートされています。インターフェイス VPC エンドポイントは AWS PrivateLink により提供され、時間単位の料金と GB あたりの使用料が発生します。
- NAT ゲートウェイにはスケーリングと管理機能が組み込まれており、スタンドアロンの NAT インスタンスとは異なりコストを節約できます。NAT ゲートウェイはトラフィックの多いインスタンスと同じアベイラビリティーゾーンに配置し、Amazon DynamoDB または Amazon S3 にアクセスが必要なインスタンスでは VPC エンドポイントを使用して、データ転送とデータ処理コストを削減することを検討してください。
- エッジ AWS Snow Family デバイス (Snowball Edge、Snowball Edge、および Snowmobile) で、データを収集し処理するコンピューティングリソースを持つ AWS Snow Family デバイスを使用すると、ペタバイト規模のデータをコスト効率よく、オフラインで AWS クラウド に移動できます。

実装手順

- サービスを実装する: データ転送モデリングを使用し、VPC フローログを確認して、サービスとワークロードタイプに基づいて適切な AWS ネットワークサービスを選択します。最大のコストと最大のボリュームフローがどこにあるかを調べます。AWS のサービスを確認し、転送を減らすか排除するサービス (特にネットワークとコンテンツ配信) があるかどうかを評価します。また、データへの繰り返しのアクセス、または大量のデータがあるキャッシュサービスを探します。

リソース

関連ドキュメント:

- AWS Direct Connect
- AWS の製品を見る
- AWS キャッシュソリューション
- Amazon CloudFront
- AWS Snow Family
- Amazon CloudFront Security Savings Bundle

関連動画:

- Monitoring and Optimizing Your Data Transfer Costs
- AWS コスト最適化シリーズ: CloudFront
- NAT ゲートウェイのデータ転送料金を削減するにはどうすればよいですか?

関連する例:

- 共有サービスのチャージバック方法: An AWS Transit Gateway の例
- Athena クエリと QuickSight を使用してコストと使用状況レポートから AWS データ転送の詳細を深く理解する
- 一般的なアーキテクチャでのデータ転送コストの概要
- AWS Cost Explorer でデータ転送コストを分析する
- Amazon CloudFront の各種機能で AWS アーキテクチャのコストを最適化する
- NAT ゲートウェイのデータ転送料金を削減するにはどうすればよいですか?

需要を管理しリソースを供給する

クラウドに移行すると、お支払いは必要な分のみになります。必要な時にワークロードの需要に合わせたリソースを供給できるため、コストがかかる無駄なオーバープロビジョニングを排除できます。また、スロットル、バッファ、キューを使用して需要を変更すると、少ないリソースで需要を円滑に処理できます。

ジャストインタイム供給による経済的利点を得るには、リソース障害、高可用性、プロビジョニング時間を考慮したプロビジョニングの必要性とのバランスを保つことが必要です。需要が固定か変動的かに応じて、最小限の手間で環境を管理できるようにメトリクスと自動化の作成計画を立てます。スケーリング中でも同様のことが言えます。需要を変更するときは、ワークロードが許容できる最大遅延を把握しておく必要があります。

AWS では、需要の管理とリソースの供給にさまざまなアプローチを使用できます。以下の各セクションでは、こうしたアプローチのベストプラクティスを説明します。

ベストプラクティス

- COST09-BP01 ワークロードの需要に関する分析を実行する
- COST09-BP02 需要を管理するためのバッファまたはスロットルを実装する
- COST09-BP03 リソースを動的に供給する

COST09-BP01 ワークロードの需要に関する分析を実行する

ワークロードの需要を経時的に分析します。分析が季節的傾向を考慮し、ワークロードのライフタイム全体にわたる動作条件を正確に反映したものであることを確認します。分析を行う際には、費やされた時間がワークロードのコストに比例しているなどの潜在的利益を織り込む必要があります。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

クラウドコンピューティングのワークロード需要を分析するには、クラウド環境で開始されるコンピューティングタスクのパターンと特性を理解する必要があります。この分析により、ユーザーはリソース割り当てを最適化し、コストを管理し、パフォーマンスが必要なレベルを満たしていることを確認することができます。

ワークロードの要件を把握します。組織の要件に、リクエストに対するワークロードの応答時間を含める必要があります。応答時間は、需要が管理されているかどうか、または需要を満たすためにリソースの供給を変更する必要があるかどうかを判断するために使用できます。

分析には、需要の予測可能性と再現性、需要の変化率、需要の変化量を含める必要があります。分析は、月末処理や休日のピークなどの時季的な変動が組み込まれるように、十分な期間にわたって実行します。

分析作業では、スケーリングの実装による潜在的な利点が反映されるようにします。コンポーネントの予想される合計コスト、ワークロードのライフタイムにおける使用量の増減およびコストの増減に注目します。

クラウドコンピューティングのワークロード需要分析を行う際に考慮すべき重要な点は、次のとおりです。

1. リソースの使用状況とパフォーマンメトリクス: AWS リソースの使用状況を経時的に分析します。ピーク時とオフピーク時の使用パターンを特定して、リソースの割り当てとスケーリング戦略を最適化します。応答時間、レイテンシー、スループット、エラー率などのパフォーマンスメトリクスをモニタリングします。これらのメトリクスは、クラウドインフラストラクチャの全体的な状態と効率を評価するのに役立ちます。
2. ユーザーとアプリケーションのスケーリング動作: ユーザーの行動と、その行動がワークロードの需要にどのように影響するかを理解します。ユーザートラフィックのパターンを調べることは、コンテンツの配信とアプリケーションの応答性を向上させるうえで役立ちます。需要の増加に伴ってワークロードがどのようにスケールするかを分析します。ロードの変動に対応するために、自動スケーリングパラメータが適切かつ効果的に設定されているかどうかを判断します。
3. ワークロードのタイプ: バッチ処理、リアルタイムデータ処理、ウェブアプリケーション、データベース、機械学習など、クラウドで実行されているさまざまなタイプのワークロードを特定します。ワークロードのタイプごとに、リソース要件とパフォーマンスプロファイルが異なる場合があります。
4. サービスレベルアグリーメント (SLA): 実際のパフォーマンスを SLA と比較して、コンプライアンスを確保し、改善が必要な領域を特定します。

Amazon CloudWatch を使用して、メトリクスの収集とトラッキング、ログファイルの収集とモニタリング、アラームの設定、AWS リソースの変更への自動対応を行うことができます。Amazon CloudWatch を使用して、リソースの使用率、アプリケーションのパフォーマンス、運用の状況をシステム全体で把握できます。

AWS Trusted Advisor では、ベストプラクティスに従ってリソースをプロビジョニングすることで、システムのパフォーマンスと信頼性を向上させ、セキュリティを強化し、コスト節減の機会を探すことができます。また、本番環境以外のインスタンスをオフにして、需要の増減に合わせて Amazon CloudWatch や自動スケーリングを使用することもできます。

最後に、AWS Cost Explorer または QuickSight を AWS Cost and Usage Report (CUR) ファイルやアプリケーションログと共に使用して、ワークロード需要の高度な分析を実行できます。

全体的には、包括的なワークロード需要分析により、組織はリソースのプロビジョニング、スケーリング、最適化について情報に基づいた意思決定が可能になり、パフォーマンス、コスト効率、ユーザー満足度の向上につながります。

実装手順

- 既存のワークロードデータを分析する: 既存のワークロード、以前のバージョンのワークロード、または予測された使用パターンのデータを分析します。Amazon CloudWatch、ログファイルとモニタリングデータを使用して、ワークロードの使用状況についてインサイトを得ます。ワークロードの全サイクルを分析し、月末や年末のイベントなどの季節的な変化のデータを収集します。分析に反映される労力は、ワークロードの特性を反映する必要があります。最大の労力は、需要に最も大きな変化がある価値の高いワークロードに割り当てられる必要があります。需要の変化が最小である低価値のワークロードには、最小の労力を割り当てる必要があります。
- 外部の影響を予測する: 組織全体において、ワークロードの需要に影響を与え、または変化させる可能性のあるチームメンバーとミーティングを行います。一般的なチームは販売、マーケティング、ビジネス開発です。当該メンバーと協力して、業務のサイクルや、ワークロードの需要を変化させるイベントがあるかどうかを把握します。このデータを使用してワークロードの需要を予測します。

リソース

関連ドキュメント:

- Amazon CloudWatch
- AWS Trusted Advisor
- AWS X-Ray
- AWS Auto Scaling
- AWS での Instance Scheduler
- Amazon SQS の開始方法

- AWS Cost Explorer
- QuickSight

関連する例:

- コスト最適化のためのモニタリング、追跡、分析
- CloudWatch での口グの検索および分析

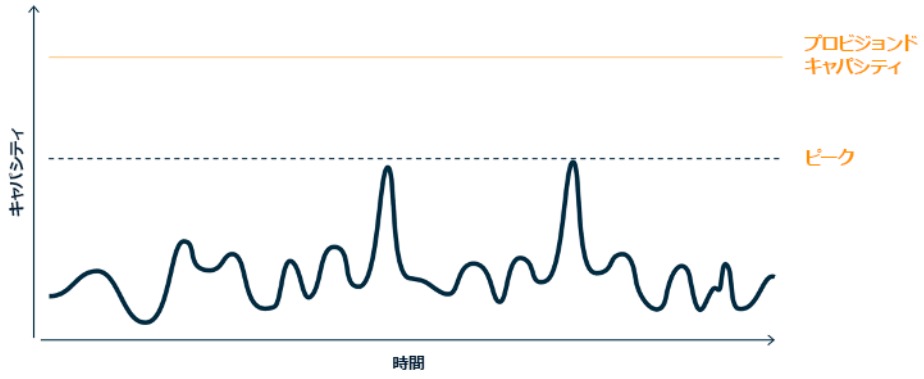
COST09-BP02 需要を管理するためのバッファまたはスロットルを実装する

バッファリングとスロットリングは、ワークロードの需要を修正し、ピークを滑らかにします。クライアントが再試行を実行するときにスロットリングを実行します。バッファリングを実装して、リクエストを保存し、処理を延期できます。スロットルとバッファが、クライアントが要求された時間内にレスポンスを受け取るように設計されていることを確認します。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

クラウドコンピューティングでは、需要を管理し、ワークロードに必要なプロビジョンドキャパシティを削減するために、バッファまたはスロットリングの実装が不可欠です。パフォーマンスを最適化するには、ピークを含む総需要、リクエストの変化のペース、必要な応答時間を測定することが重要です。クライアントにリクエストの再送機能がある場合は、スロットリングの適用が現実的です。逆に、クライアントに再試行の機能がなければ、バッファソリューションの実装が理想的なアプローチです。バッファは、入ってくるリクエストの交通整理を行い、動作速度がさまざまに異なるアプリケーションとの通信を最適化します。



高いプロビジョンドキャパシティを必要とする 2 つの異なるピークの需要曲線

上の図に示す需要曲線を持つワークロードがあるとします。このワークロードには 2 つのピークがあり、これらのピークを処理するために、オレンジの線で示されるリソース容量がプロビジョニングされます。このワークロードで使用されるリソースとエネルギーは需要曲線の下の領域ではなく、プロビジョンドキャパシティのラインの下の領域で示されます。これら 2 つのピークを処理するには、プロビジョンドキャパシティが必要であるためです。ワークロードの需要曲線を平坦化することで、ワークロードに必要なプロビジョンドキャパシティを削減し、環境への影響を減らすことができます。ピークをならすには、スロットリングまたはバッファリングのソリューションの実装を検討してください。

理解を深めるために、スロットリングとバッファリングについて見ていきましょう。

スロットリング: 需要元のソースに再試行機能がある場合は、スロットリングを実装できます。スロットリングでは、その時点でリクエストを処理できない場合は、後で再試行する必要があることが需要側に通知されます。需要側は一定時間待ってから、リクエストを再試行します。スロットリングの運用には、リソースの最大量およびワークロードのコストを制限できるという利点があります。AWS では、Amazon API Gateway を使用してスロットリングを実装できます。

バッファベース: バッファベースのアプローチでは、プロデューサー (キューにメッセージを送信するコンポーネント)、コンシューマー (キューからメッセージを受信するコンポーネント)、およびキュー (メッセージを保持) を使用してメッセージを保存します。メッセージはコンシューマーによって読み取られ、処理されるため、コンシューマーのビジネス要件を満たせる動作速度でメッセージを実行できます。バッファを中心にした方法を採用することで、プロデューサーが送信したメッセージはキューまたはストリームに蓄えられ、コンシューマーがそれぞれの運用上の需要に応じたペースでアクセスできるようになります。

AWS でバッファベースのアプローチを実装する際は、複数のサービスから選択できます。Amazon Simple Queue Service(Amazon SQS) は、単独のコンシューマーが個別のメッセージを読むことができるキューを提供するマネージドサービスです。Amazon Kinesis は、多数のコンシューマーが同じメッセージを読み取ることができるストリームを提供します。

バッファリングとスロットリングは、ワークロードの需要を変化させ、ピークを滑らかにします。クライアントがアクションを再試行する場合はスロットリングを使用し、リクエストを保留して後で処理する場合はバッファリングを使用します。バッファベースのアプローチを採用する場合は、必要な時間内にリクエストを処理するようにワークロードを設計し、作業の重複リクエストを処理できるようにします。全体的な需要、変化率、および要求される応答時間を分析して、必要なスロットルまたはバッファのサイズを適正化します。

実装手順

- クライアント要件を分析する: クライアントリクエストを分析して、再試行を実行できるかどうかを判断します。再試行を実行できないクライアントの場合、バッファを実装する必要があります。全体的な需要、変化率、および要求される応答時間を分析して、必要なスロットルまたはバッファのサイズを決定します。
- バッファまたはスロットルを実装する: ワークロードにバッファまたはスロットルを実装します。Amazon Simple Queue Service (Amazon SQS) などのキューは、ワークロードコンポーネントにバッファを提供できます。Amazon API Gateway は、ワークロードコンポーネントのスロットリングを提供できます。

リソース

関連するベストプラクティス:

- SUS02-BP06 需要曲線を平坦化するためにバッファリングまたはスロットリングを実装する
- REL05-BP02 リクエストのスロットル

関連ドキュメント:

- AWS Auto Scaling
- AWS での Instance Scheduler
- Amazon API Gateway
- Amazon Simple Queue Service
- Amazon SQS の開始方法
- Amazon Kinesis

関連動画:

- 分散アプリに適したメッセージングサービスの選択

関連する例:

- ワークロードでの API スロットリングの管理とモニタリング
- API Gateway を使用した階層型マルチテナント REST API を大規模にスロットリングする

- Amazon API Gateway を使用したマルチテナント Amazon EKS SaaS ソリューションで階層化とスロットリングを有効にする
- キューとメッセージを使用したアプリケーション統合

COST09-BP03 リソースを動的に供給する

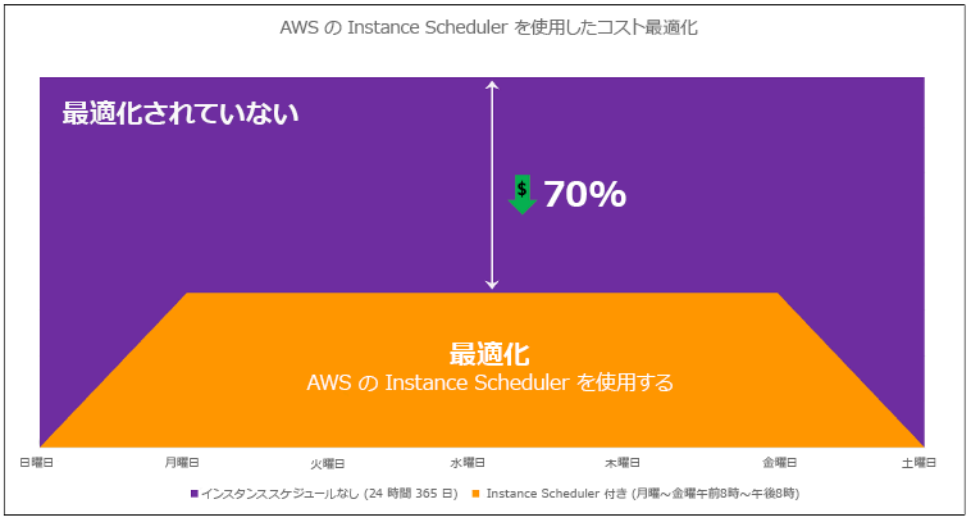
リソースを計画的にプロビジョニングします。これは、自動スケーリングなどの需要ベース、または需要が予測可能でリソースが時間に基づいて提供される時間ベースで行います。これらの手法を使用すると、過剰プロビジョニングやプロビジョニング不足を最小限に抑えることができます。

このベストプラクティスを活用しない場合のリスクレベル: 低

実装のガイダンス

AWS のお客様がアプリケーションに利用できるリソースを増やし、需要に合わせてリソースを供給する方法はいくつかあります。その 1 つが AWS Instance Scheduler を使用した方法です。これにより、Amazon Elastic Compute Cloud (Amazon EC2) と Amazon Relational Database Service (Amazon RDS) インスタンスの起動および停止を自動化します。もう 1 つの方法は AWS Auto Scaling を使用することです。この方法では、アプリケーションやサービスの需要に基づいてコンピューティングリソースを自動的にスケールできます。需要に応じてリソースを供給することで、使用したリソースに対してのみ支払いを行い、必要なときにリソースを起動してコストを削減し、必要でないときにリソースを終了することができます。

AWS Instance Scheduler を使用すると、Amazon EC2 インスタンスや Amazon RDS インスタンスを決まった時間に停止および開始するように設定できます。これにより、例えばユーザーが毎朝 8 時に Amazon EC2 インスタンスにアクセスし、夜 6 時以降は必要としないなど、一貫した時間パターンがある同一リソースの需要に応えることができます。この解決方法では、リソースを使用しないときは停止し、必要なときに開始することで、運用コストを削減できます。



AWS Instance Scheduler によるコストの最適化。

また、AWS Systems Manager Quick Setup を使用してシンプルなユーザーインターフェイス (UI) を使用して、アカウントやリージョン全体で Amazon EC2 インスタンスのスケジュールを簡単に設定できます。AWS Instance Scheduler を使用して Amazon EC2 または Amazon RDS インスタンスをスケジュールでき、既存のインスタンスを停止および起動できます。ただし、Auto Scaling グループ (ASG) の一部であるインスタンス、または Amazon Redshift や Amazon OpenSearch Service などのサービスを管理するインスタンスを停止/開始することはできません。Auto Scaling グループには、グループ内のインスタンスに対して独自のスケジューリングがあり、このスケジュールに基づいてインスタンスが作成されます。

AWS Auto Scaling により、変化する需要に対応するためにキャパシティを調整して、最低限のコストで安定かつ予測可能なパフォーマンスを維持できます。これは、Amazon EC2 インスタンスおよびスポットフリート、Amazon ECS、Amazon DynamoDB、Amazon Aurora と統合するアプリケーションの容量をスケールするためのフルマネージドで無料のサービスです。自動スケーリングでは、リソースの自動検出によってワークロード内の設定可能なリソースを検出できます。また、パフォーマンス、コスト、または両者のバランスを最適化するためのスケーリング戦略が組み込まれており、予測スケーリングによって定期的に発生する急増に対応することができます。

Auto Scaling グループをスケーリングするには、複数のスケーリングオプションを使用できます。

- 現在のインスタンスレベルの常時維持
- 手動でスケールする
- スケジュールに基づくスケーリング
- 需要に基づくスケーリング

- 予測スケーリングの使用

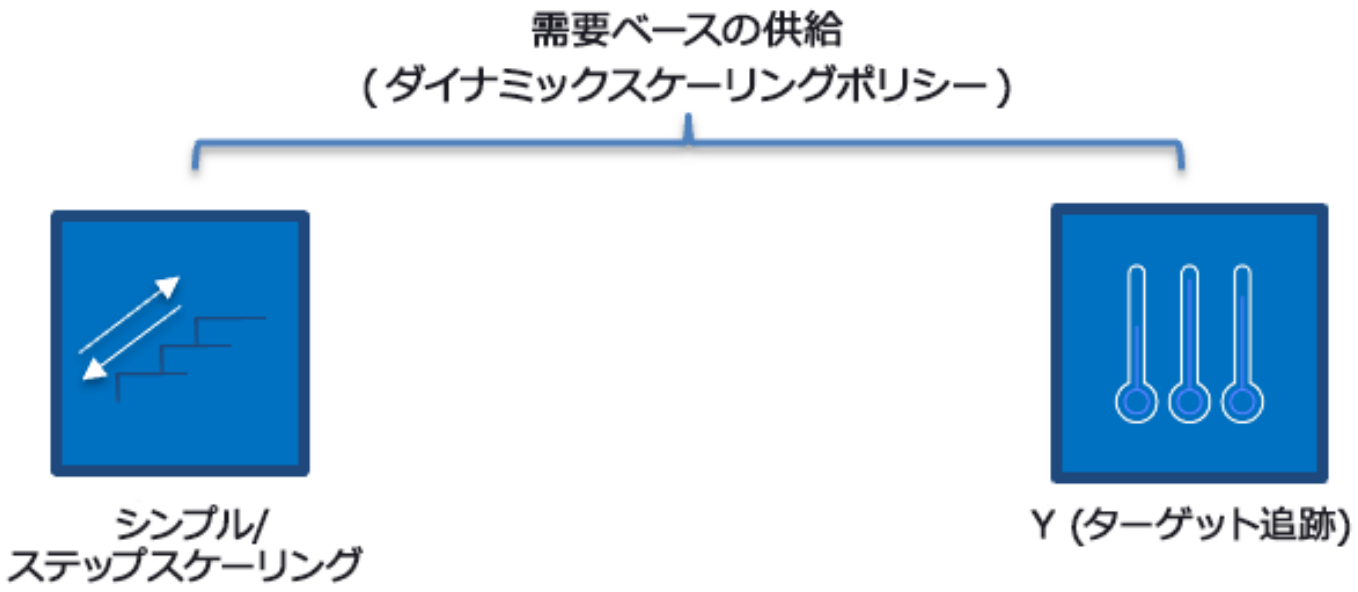
自動スケーリングポリシーは異なり、動的スケーリングポリシーとスケジュールスケーリングポリシーに分類できます。動的ポリシーには、手動または動的スケーリング、スケジュールスケーリングまたは予測スケーリングがあります。スケーリングポリシーは、動的スケーリング、スケジュールスケーリング、予測スケーリングに使用できます。Amazon CloudWatch のメトリクスとアラームを使用して、ワークロードのスケーリングイベントをトリガーすることもできます。最新の機能や改善点にアクセスできるように、起動テンプレートを使用することをお勧めします。起動設定を使用する場合、すべての自動スケーリング機能を使用できるわけではありません。たとえば、スポットインスタンスとオンデマンドインスタンスの両方を起動する Auto Scaling グループや、複数のインスタンスタイプを指定する Auto Scaling グループを作成することはできません。これらの機能を設定するには、起動テンプレートを使用する必要があります。起動テンプレートを使用するときは、それぞれバージョンを作成することをお勧めします。起動テンプレートのバージョン管理では、パラメータのフルセットのサブセットを作成できます。その後、再使用して、同じ起動テンプレートの他のバージョンを作成できます。

AWS Auto Scaling を使用するか、AWS API または SDK でコードにスケーリングを実装できます。これにより、環境を手動変更していた運用コストがなくなり、その結果、全体的なワークロードコストが削減され、変更をより迅速に実行できるようになります。またこれにより、いつでもワークロードのリソースを需要に合わせて調達できます。ベストプラクティスに従って組織に動的にリソースを供給するには、AWS クラウドの水平スケーリングおよび垂直スケーリングと、Amazon EC2 インスタンスで実行されるアプリケーションの特性を理解する必要があります。このベストプラクティスに従うには、クラウド財務管理チームとテクニカルチームが協働することをお勧めします。

Elastic Load Balancing (Elastic Load Balancing) は、複数のリソースに需要を分散させることでスケーリングに役立ちます。ASG と Elastic Load Balancing を使用して、トラフィックを最適にルーティングして受信リクエストを管理し、Auto Scaling グループ内の 1 つのインスタンスに負荷がかかりすぎないようにすることができます。リクエストは、キャパシティや使用率を考慮せずに、ラウンドロビン方式でターゲットグループのすべてのターゲットに分散されます。

一般的な Amazon EC2 メトリクスは、CPU 使用率、ネットワークスループット、Elastic Load Balancing で確認されたリクエストとレスポンスのレイテンシーなどの標準メトリクスです。可能な場合は、カスタマーエクスペリエンスの指標となるメトリクスを使用する必要があります。このメトリクスは一般には、ワークロード内のアプリケーションコードから生成されるカスタムメトリクスです。このドキュメントでは、需要を動的に満たす方法を詳しく説明するために、自動スケーリングを需要ベースの供給モデルと時間ベースの供給モデルの 2 つのカテゴリに分類し、それぞれについて詳しく説明します。

需要ベースの供給: クラウドの伸縮性を活用して、ほぼリアルタイムの需要状況に応じて、変化する需要に対応するリソースを供給できます。需要ベースの供給の場合、API やサービス機能を活用すると、アーキテクチャ内のクラウドリソースの量をプログラムで変更できます。これにより、アーキテクチャ内のコンポーネントをスケールしたり、需要が急増したときにリソースの数を増加させてパフォーマンスを維持したり、需要が後退したときにキャパシティを減少させてコストを節減したりできます。



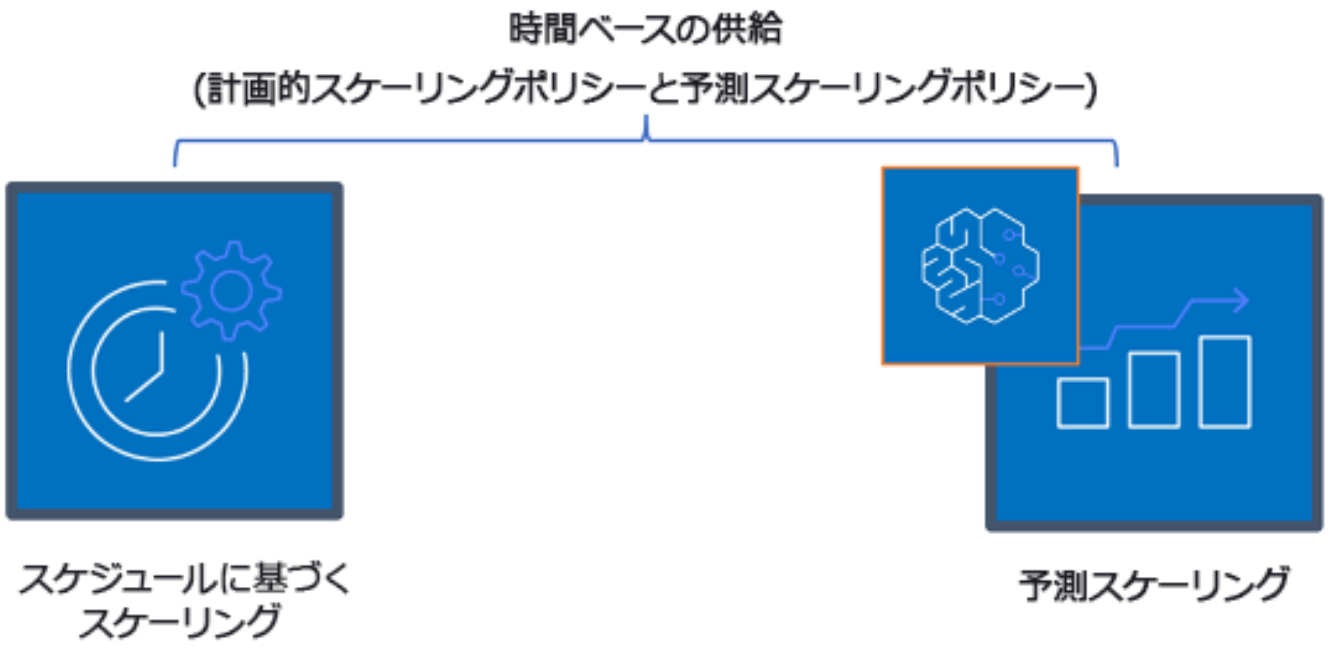
需要ベースの動的スケーリングポリシー

- シンプル/ステップスケーリング: メトリクスをモニタリングし、ユーザーが手動で定義したステップに従ってインスタンスを追加/削除します。
- ターゲット追跡: サーモスタットのような制御メカニズムで、インスタンスを自動的に追加または削除して、メトリクスをユーザー定義の目標に維持します。

需要ベースのアプローチで設計する場合、主に 2 つの点を考慮する必要があります。第 1 に、新しいリソースをどれだけ早くプロビジョニングする必要があるかを理解することです。第 2 に、需要と供給の差異が変動することを理解することです。需要の変動ペースに対処できるようにしておくだけでなく、リソースの不具合にも備えておく必要があります。

時間ベースの供給: 時間ベースのアプローチでは、リソースのキャパシティを予測可能な需要、または時間ごとに明確に定義された需要に合わせます。このアプローチは、通常、リソースの使用率に依存せず、リソースが必要な特定の時間にそのリソースを確保します。また、起動手順、およびシステ

ムや一貫性のチェックにより、遅延なくリソースを提供できます。時間ベースのアプローチでは、繁忙期に追加のリソースを投入したり、キャパシティを拡大したりできます。



時間ベースのスケーリングポリシー

スケジュールされた自動スケーリングまたは予測自動スケーリングを使用して、時間ベースのアプローチを実装できます。営業開始時など、特定の時間にワークロードをスケールアウトまたはスケールインするようにスケジュールできるため、ユーザーがアクセスしたときや需要が増加したときにリソースを利用可能にしておくことができます。予測スケーリングでは、パターンを使用してスケールアウトします。一方スケジュールに基づくスケーリングでは、事前に定義された時間を使用してスケールアウトします。また、Auto Scaling グループで属性ベースのインスタンスタイプ選択 (ABS) 戦略を使用すると、vCPU、メモリ、ストレージなどの属性セットとしてインスタンスの要件を表現できます。これにより、新しい世代のインスタンスタイプがリリースされると自動的に使用し、さらに Amazon EC2 スポットインスタンスでより広い範囲のキャパシティにアクセスできます。Amazon EC2 Fleet と Amazon EC2 Auto Scaling が指定した属性に適合するインスタンスを選択して起動するため、手動でインスタンスタイプを選択する必要がなくなります。

AWS API と SDK と AWS CloudFormation を活用して、必要に応じて環境全体を自動でプロビジョニングおよび廃止することもできます。このアプローチは、所定の営業時間や一定期間にのみ実行される開発環境またはテスト環境に適しています。API を使用した環境内のリソースサイズのスケーリング (垂直スケーリング) にも対応しています。例えば、インスタンスのサイズやクラスを変更し

て、本番稼働ワークロードをスケールアップできます。これを行うには、インスタンスを停止・起動して、別のインスタンスのサイズやクラスを選択します。この手法は、使用中にサイズの拡大、パフォーマンス (IOPS) の調整、ボリュームタイプの変更が可能な Amazon EBS Elastic Volumes などのリソースにも適用できます。

時間ベースのアプローチを設計する際は、主に 2 つの点を考慮する必要があります。第 1 に、使用パターンの一貫性について、第 2 に、パターンを変更した場合の影響です。予測精度は、ワークロードをモニタリングし、ビジネスインテリジェンスを使用することで高めることができます。使用パターンに大幅な変更がある場合は、時間を調整して予測対象範囲に収まるようにします。

実装手順

- スケジュール済みのスケーリングを設定する: 需要の変化を予測できるため、時間ベースのスケーリングは適切な数のリソースを適時に提供できます。また、リソースの作成と設定が、需要の変化に対応するのに十分ではない場合にも役立ちます。ワークロード分析を活用して、AWS Auto Scaling を使用してスケジュールに基づくスケーリングを設定します。時間ベースのスケジューリングを設定するには、予測スケーリングまたはスケジュールに基づくスケーリングを使用し、予想される、または予測可能な負荷の変化に合わせて、事前に Auto Scaling グループの Amazon EC2 インスタンス数を増やすことができます。
- 予測スケーリングの設定: 予測スケーリングを使用して、トラフィックフローの日次および週次のパターンに先立って Auto Scaling グループ内の Amazon EC2 インスタンスの数を増やします。定期的にトラフィックのスパイクがあり、アプリケーションの起動に時間がかかる場合は、予測スケーリングの使用を考慮すべきです。予測スケーリングを使用すると、見積もられた負荷の前にキャパシティを初期化できるため、性質上後手に回る動的スケーリング単体と比較して、より迅速にスケールできます。例えば、ユーザーが始業時間とともにワークロードの仕様を開始し、終業時間後は使用しない場合、予測スケーリングを使用すれば、始業時間前にキャパシティを追加できるため、トラフィックの変化に反応する動的スケーリングで生じる遅延を排除できます。
- 動的自動スケーリングの設定: アクティブなワークロードメトリクスに基づいてスケーリングを設定するには、自動スケーリングを使用します。分析を使用して、正しいリソースレベルで起動するように自動スケーリングを設定し、ワークロードが要求された時間内にスケールすることを検証します。1 つの Auto Scaling グループ内で、オンデマンドインスタンスとスポットインスタンスのフリートを起動してオートスケールできます。スポットインスタンスの使用で割引を受けるだけでなく、リザーブドインスタンスまたは Savings Plan を使用して、通常のオンデマンドインスタンス コストの割引料金を受け取ることができます。これらのすべての要素を組み合わせることで、Amazon EC2 インスタンスのコスト削減を最適化しつつ、アプリケーションに必要なスケールとパフォーマンスを得ることができます。

リソース

関連ドキュメント:

- AWS Auto Scaling
- AWS での Instance Scheduler
- Auto Scaling グループのサイズをスケールする
- Amazon EC2 Auto Scaling の使用を開始する
- Amazon SQS の開始方法
- Amazon EC2 Auto Scaling のスケジュールされたスケーリング
- Amazon EC2 Auto Scaling の予測スケーリング

関連動画:

- 自動スケーリングのターゲットトラッキングスケーリングポリシー
- AWS での Instance Scheduler

関連する例:

- Amazon EC2 Fleet 用自動スケーリングでの属性ベースのインスタンスタイプ選択
- スケジュールされたスケーリングを使用して Amazon Elastic Container Service を最適化しコストを削減する
- Amazon EC2 Auto Scaling での予測スケーリング
- AWS CloudFormation で Instance Scheduler を使用して Amazon EC2 インスタンスをスケジュールするにはどうすればよいですか?

継続的最適化

AWS では、新しいサービスを確認してワークロードに実装することで、経時的に最適化していきます。

AWS で新しいサービスと機能がリリースされた際は、ベストプラクティスとして、既存のアーキテクチャの決定事項を確認し、費用対効果を維持することが推奨されます。要件の変化に応じて、不要になったリソース、コンポーネント、ワークロードを積極的に排除します。経時的に最適化を行うには、次のベストプラクティスを考慮します。

ワークロードを経時的に最適化し組織内の CFM 文化を改善しながら、クラウドの運用にかかる労力のコストを評価し、時間のかかるクラウド運用を見直し、それらを自動化して、関連する AWS サービスやサードパーティー製品、カスタムツール (AWS CLI や AWS SDK など) を採用することにより人的労力およびコストを削減します。

トピック

- レビュープロセスを定義し、ワークロードを定期的に分析
- オペレーションの自動化

レビュープロセスを定義し、ワークロードを定期的に分析

ベストプラクティス

- COST10-BP01 ワークロードレビュープロセスを開発する
- COST10-BP02 このワークロードを定期的に見直し、分析する

COST10-BP01 ワークロードレビュープロセスを開発する

ワークロードレビューの基準とプロセスを定義するプロセスを開発します。レビューを行う際には、潜在的利益を織り込む必要があります。例えば、コアワークロードや、請求の 10% 超に値するワークロードは四半期または 6 か月ごとにレビューし、10% 以下のワークロードは年に 1 回レビューするなどです。

このベストプラクティスを活用しない場合のリスクレベル: 高

実装のガイダンス

ワークロードの費用対効果を最大にするには、ワークロードを定期的にレビューし、新しいサービス、機能、コンポーネントを実装する機会があるかどうかを把握する必要があります。全体的なコスト削減を達成するには、潜在的なコスト削減量に比例したプロセスを行う必要があります。例えば、支出全体の 50% を占めるワークロードは、支出全体の 5% を占めるワークロードよりも定期的かつ徹底的にレビューする必要があります。外部要因または変動性を考慮します。ワークロードにより特定の地域、特定の市場セグメントにサービスが提供されていて、その領域での変化が予測される場合、レビュー頻度を高くすることでコスト削減につながる可能性があります。レビューで考慮すべきもう 1 つの要因は、変更を運用する労力です。変更のテストおよび検証に多大なコストがかかる場合は、レビューの頻度を下げる必要があります。

古くなったレガシーコンポーネントやリソースには維持するための長期的なコストがかかることや、新しい機能を実装できないことを考慮します。テストと検証にかかる現在のコストが、提案されている利益を上回っている場合があります。しかし、ワークロードと現在のテクノロジーとのギャップが時間の経過とともに大きくなるにつれて、変更にかかるコストが増加し、結果として巨額のコストになることがあります。例えば、新しいプログラミング言語に移行するときの費用対効果は現時点で低いとします。しかし、5 年後には、その言語に精通した人材のコストが増加する可能性があります。ワークロードが増加すると、さらに大規模なシステムを新しい言語に移行することになり、結果的にこれまでよりもさらに多大な労力を要します。

ワークロードをコンポーネントに分割し、コンポーネントのコストを割り当て (コストの見積りで可)、各コンポーネントの横に要因 (労力や外部市場など) を一覧表示します。この指標を使用して、各ワークロードのレビュー頻度を決定します。例えば、ウェブサーバーが高コストで、変更の労力が低く、外部要因が高い場合は、レビュー頻度が高くなります。中央データベースが中程度のコストで、変更の労力が高く、外部要因が低い場合は、レビューの頻度は中程度になります。

新しいサービス、設計パターン、リソースの種類、設定が利用できるようになった時点で、これらを評価するプロセスを定義し、ワークロードコストを最適化します。パフォーマンスの柱のレビューと信頼性の柱のレビュープロセスと同様に、最適化および改善のアクティビティを特定、検証、優先順位付けし、これをバックログに組み込みます。

実装手順

- レビュー頻度を定義する: ワークロードとそのコンポーネントを確認する頻度を定義します。継続的な改善とレビューの周期のための時間とリソースを割り当て、ワークロードの効率性と最適化を向上させます。これは要因の組み合わせであり、組織内のワークロード、またワークロード内のコンポーネントによって、異なる場合があります。一般的な要因には、収益またはブランドの観点から評価された組織にとっての重要性、ワークロードの実行にかかる総コスト (運用コストとリソー

スコストを含む)、ワークロードの複雑さ、変更の実装の容易性、ソフトウェアライセンス契約、ある変更がライセンス違反によるライセンス費用の重大な増加を生じさせるかどうかなどが含まれます。コンポーネントは、ウェブサーバーやデータベース、コンピューティングリソースやストレージリソースなど、機能的または技術的に定義できます。それに応じて要因のバランスをとり、ワークロードとそのコンポーネントのための期間を設定します。例えば、ワークロード全体は 18 か月ごとに、ウェブサーバーは 6 か月ごとに、データベースは 12 か月ごとに、コンピューティングおよび短期ストレージは 6 か月ごとに、長期ストレージは 12 か月ごとに、それぞれ確認することができます。

- レビューの十分性を定義する: ワークロードまたはワークロードコンポーネントのレビューに費やされる労力を定義します。レビュー頻度と同様に、これは複数の要因のバランスです。最も大きな利益をもたらす取り組みに集中できるように、改善の機会を定期的に評価し、優先順位を設定します。同時に、それらの活動に必要な作業量を見積もります。予想される結果が目標に達しておらず、作業コストがさらにかかる場合は、代わりの一連のアクションを使用して作業を繰り返します。レビュープロセスには、漸進的な継続的改善を可能にする時間とリソースを含める必要があります。例えば、データベースコンポーネントの分析に 1 週間、コンピューティングリソースの分析に 1 週間、ストレージのレビューに 4 時間を、それぞれ費やすように決めます。

リソース

関連ドキュメント:

- AWS ニュースブログ
- クラウドコンピューティングのタイプ
- AWS の最新情報

関連する例:

- AWS サポートのプロアクティブサービス
- SAP ワークロードの定期的なレビューを計画する

COST10-BP02 このワークロードを定期的に見直し、分析する

既存のワークロードは、それぞれ定義されたプロレスに基づいて定期的に見直され、新しいサービスを導入できるか、既存のサービスを置き換えることができるか、またはワークロードをリアーキテクトできるかを確認します。

このベストプラクティスを活用しない場合のリスクレベル: 中

実装のガイダンス

AWS は定期的に新しい機能を追加しているため、最新のテクノロジーを利用して、より迅速に実験やイノベーションできます。「AWS の最新情報」では、AWS の新機能について詳しく説明し、リリースされた AWS サービス、機能、リージョン拡大に関して概説します。発表されたリリースの詳細を確認して、既存のワークロードの見直しや分析にそれらを使用できます。新しい AWS のサービスと機能の利点を得るには、ワークロードでレビューを行い、必要に応じて新しいサービスや機能を実装する必要があります。つまり、場合によっては、ワークロードに使用している既存のサービスを置き換えたり、ワークロードをモダナイズして新しい AWS のサービスを導入したりする必要があるということです。例えば、ワークロードを見直して、メッセージングコンポーネントを Amazon Simple Email Service に置き換えることができます。これにより、すべての機能を低コストで提供しながら、インスタンスのフリートの運用と維持にかかるコストを削減できます。

ワークロードを分析して潜在的な機会を見出すには、新しいサービスだけではなく、ソリューション構築における新しい方法も考慮する必要があります。他のお客様のアーキテクチャ設計、課題、ソリューションについては、AWS の「This is My Architecture」ビデオをご覧ください。「All-In series」で、実際の AWS サービスとカスタマーストーリーをご覧ください。また、基本的なクラウドアーキテクチャパターンのベストプラクティスを説明、調査、および分類する「Back to Basics」ビデオシリーズを視聴することもできます。もう 1 つのソースは、「How to Build This」動画です。この動画は、AWS サービスを使用して実用最小限の製品 (MVP) を実現する方法について、大きな構想を持つユーザーを支援するように設計されています。確固たるアイデアを持った世界中の構築者が、経験豊富な AWS のソリューションアーキテクトからのアーキテクチャに関するガイダンスを得ることができます。最後に、「入門ガイド」の資料を参照できます。ステップバイステップのチュートリアルが含まれています。

レビュープロセスを開始する前に、合意されたレビュープロセスに従いながら、ワークロードにおけるビジネスの要件、特定のサービスまたはリージョンを使用するためのセキュリティおよびデータのプライバシー要件、パフォーマンス要件に従います。

実装手順

- ワークロードを定期的に見直す: 定義したプロセスを使用して、指定した頻度でレビューを実行します。各コンポーネントに適正な労力を費やしていることを確認します。このプロセスは、コスト最適化のためにサービスを選択した最初の設計プロセスに似ています。サービスとこのサービスがもたらすメリットを分析します。今回は、長期的なメリットだけでなく、変更を行うコストも考慮します。

- 新しいサービスを実装する: 分析の結果、変更を実施する場合は、まずワークロードのベースラインを実行し、各アウトプットの現在のコストを把握します。変更を実施し、分析を実行して、各アウトプットの新しいコストを確認します。

リソース

関連ドキュメント:

- AWS ニュースブログ
- AWS の最新情報
- AWS ドキュメント
- AWS 使用開始
- AWS 一般的なリソース

関連動画:

- AWS - This is My Architecture
- AWS - Back to Basics
- AWS - All-In シリーズ
- 構築する方法

オペレーションの自動化

ベストプラクティス

- COST11-BP01 運用のオートメーションを実行する

COST11-BP01 運用のオートメーションを実行する

管理タスク、デプロイ、人的エラーのリスク低減、コンプライアンス、運用において、オートメーションによって達成可能な時間と労力の節約を数値化することに重点を置いて、クラウドの運用コストを評価します。運用作業に必要な時間と関連コストを評価し、管理タスクを自動化することで、可能な限り手作業を最小限に抑えます。

このベストプラクティスを活用しない場合のリスクレベル: 低

実装のガイダンス

運用を自動化することで、手動タスクの頻度が減り、効率が高まると共に、ワークロードのデプロイ、管理、運用において安定した信頼性の高いエクスペリエンスを提供できるため顧客にメリットをもたらします。インフラストラクチャのリソースを手動の運用タスクから解放し、リソースをより価値の高いタスクやイノベーションに使用できるため、ビジネス成果が向上します。企業は、クラウドでワークロードを管理するための、実績がありテスト済みの方法を求めています。そのソリューションは、安全で、高速で、費用対効果が高く、リスクを最小限に抑え、最大限の信頼性を備えている必要があります。

運用コスト全体を確認して、必要な労力に基づいて運用作業に優先順位を付けることから始めます。例えば、クラウドに新しいリソースをデプロイするのにかかる時間、既存のものを最適化する変更にかかる時間、必要な構成を設定するのにかかる時間はどのくらいでしょうか。オペレーションと管理のコストを考慮に入れて、人間による操作の合計コストを確認します。管理タスクのオートメーションに優先順位を付けて、人間の手作業を減らします。

レビューには潜在的利益を織り込む必要があります。例えば、タスクを手動で実行する場合にかかる時間を、自動で実行する場合と比較します。反復的で価値が高く、時間のかかる複雑なアクティビティの自動化を優先します。通常、高価値で、人的エラーのリスクが高いアクティビティから自動化するのが良い方法です。このようなリスクは望ましくない追加の運用コスト (運用チームの追加作業時間など) の発生につながることが多いためです。

AWS Systems Manager または AWS Config などのオートメーションツールを使用して、運用、コンプライアンス、モニタリング、ライフサイクル、終了のプロセスを合理化します。AWS のサービス、ツール、およびサードパーティー製品を使用して、実装するオートメーションを特定の要件に合わせてカスタマイズできます。次の表は、AWS のサービスを使用して管理と運用を自動化することで実現できる主な運用の機能の一部を示しています。

- AWS Audit Manager: AWS の使用状況を継続的に監査して、リスクとコンプライアンスの評価を簡素化する
- AWS Backup: データ保護を一元的に管理し自動化します。
- AWS Config: コンピューティングリソース、評価、監査、設定の評価、リソースインベントリを設定します。
- AWS CloudFormation: Infrastructure as Code を使用して高可用性リソースを起動します。
- AWS CloudTrail: IT の変更管理、コンプライアンス、制御。
- Amazon EventBridge は、イベントをスケジュールし AWS Lambda をトリガーしてアクションを実行します。

- AWS Lambda: イベントによりトリガーするか、AWS EventBridge で固定スケジュールにより実行して、反復的なプロセスを自動化します。
- AWS Systems Manager: ワークロードの開始と停止、オペレーティングシステムへのパッチ適用、設定の自動化、継続的な管理。
- AWS Step Functions: ジョブをスケジュールしワークフローを自動化します。
- AWS Service Catalog: テンプレートの使用、コンプライアンスと制御を備えた Infrastructure as Code。

AWS の製品やサービスを使用してすぐにオートメーションを導入したいが組織にそのスキルがない場合は、AWS Managed Services (AMS)、AWS プロフェッショナルサービス、AWS パートナーのいずれかにご連絡いただければ、オートメーションの導入数を増やしクラウドでのオペレーショナルエクセレンスを高めることができます。

AWS Managed Services (AMS) は、エンタープライズのお客様やパートナーに代わって AWS インフラストラクチャを運用するサービスです。コンプライアンスに準拠したセキュアな環境で、ワークロードをデプロイできます。AMS では、エンタープライズクラウド運用モデルとオートメーションを使用して、組織の要件を満たし、クラウド移行を高速化し、オンゴーイングの管理コストを削減できます。

また、AWS プロフェッショナルサービスは、AWS を使用して目的のビジネス成果を達成し、運用を自動化できるようサポートします。自動化された堅牢かつ俊敏な IT 運用と、クラウドに最適化されたガバナンス機能のデプロイについてお客様を支援します。モニタリング例の詳細と推奨されるベストプラクティスについては、運用上の優秀性の柱についてのホワイトペーパーを参照してください。

実装手順

- 1 回の構築で多数のデプロイ: CloudFormation、AWS SDK、AWS CLI などの Infrastructure-as-code を使用して、1 回のデプロイで、同様の環境やディザスタリカバリシナリオ向けに何回も使用します。デプロイ中にタグを付け、他のベストプラクティスで定義されている消費を追跡します。AWS Launch Wizard を使用して、多数の一般的なエンタープライズワークロードをデプロイする回数を削減します。AWS Launch Wizard は、AWS のベストプラクティスに従ってエンタープライズワークロードのサイズ変更、設定、デプロイの方法をガイドします。Service Catalog を使用することもできます。こちらを使用すると、承認済みの Infrastructure as Code テンプレートを作成し管理して AWS で使用でき、承認済みのセルフサービス型クラウドリソースを誰でも見つけることができます。

- 継続的なコンプライアンスを自動化する: 記録済みの設定を、事前定義された基準に照らして自動的に評価および修正することを検討します。AWS Organizations を AWS Config および AWS CloudFormation の機能と組み合わせることで、多数のメンバーアカウントの、設定コンプライアンスの大規模な管理および自動化を効率的に実行できます。設定の変更や、AWS リソース間の関係を確認して、リソース設定の履歴を詳しく調べることができます。
- モニタリングタスクを自動化する: AWS には、サービスのモニタリングに使用できるさまざまなツールが用意されています。これらのツールを設定して、モニタリングタスクを自動化できます。ワークロードのすべての部分からモニタリングデータを収集するモニタリング計画を作成して実装すると、マルチポイント障害が発生した場合のデバッグがより簡単になります。例えば、自動モニタリングツールを使用して、Amazon EC2 を観察し、システムステータスチェック、インスタンスステータスチェック、および Amazon CloudWatch アラームで問題が検出された場合に報告を受けることができます。
- メンテナンスとオペレーションを自動化する: 日常的なオペレーションを自動化して人による介入をなくします。AWS サービスとツールを使用して、実装する AWS オートメーションを選択し、特定の要件に合わせてカスタマイズできます。例えば、EC2 Image Builder を使用して仮想マシンやコンテナイメージを構築、テスト、デプロイし、AWS またはオンプレミスで使用できるようにしたり、AWS SSM を使用して EC2 インスタンスにパッチを適用したりするなどです。必要なアクションを AWS のサービスで実行できない場合、またはリソースのフィルタリングを行う、より複雑なアクションを必要とする場合は、AWS Command Line Interface (AWS CLI) または AWS SDK ツールを使用してオペレーションを自動化します。AWS CLI では、AWS のサービスの制御および管理プロセス全体を、スクリプトを使用して、AWS Management Consoleを使用せずに自動化することができます。AWS のサービスとやり取りする AWS SDK を選択します。その他のコード例については、「AWS SDK Code Examples Repository」を参照してください。
- 自動化により継続的なライフサイクルを構築する: 規制や冗長性のためだけでなく、コスト最適化を実現するためにも、確固たるライフサイクルポリシーを確立してこれを維持することが重要です。AWS Backup を使用して、バケット、ボリューム、データベース、ファイルシステムなどのデータストアのデータ保護を一元的に管理および自動化できます。Amazon Data Lifecycle Manager を使用して、EBS スナップショットと EBS-backed AMI の作成、保持、削除を自動化することもできます。
- 不要なリソースを削除する: 未使用のリソースが、サンドボックスや開発用の AWS アカウントに溜まることがよくあります。開発者は通常の開発サイクルの中でさまざまなサービスやリソースを構築して実験し、不要になってもこうしたリソースを削除しないためです。未使用のリソースは、組織にとって不要なコスト、場合によっては高額なコストをもたらす場合があります。これらのリソースを削除することで、これらの環境の運用コストを削減できます。データが不要であることを確認し、不要かどうか不明な場合はバックアップ済みであることを確認します。AWS CloudFormation を使用してデプロイ済みのスタックをクリーンアップできます。これにより、テ

ンプレートで定義されているリソースの大部分が自動的に削除されます。あるいは、aws-nuke のようなツールを使用すると、AWS リソースを削除するための自動化を作成することができます。

リソース

関連ドキュメント:

- AWS クラウドでのオペレーションのモダナイズ
- オートメーション向けの AWS サービス
- インフラストラクチャとオートメーション
- AWS Systems Manager Automation
- 自動モニタリングと手動モニタリング
- SAP 管理および運営のための AWS オートメーション
- AWS Managed Services
- AWS プロフェッショナルサービス

関連動画:

- AWS での大規模な継続的コンプライアンスの自動化
- AWS Backup Demo: Cross-Account & Cross-Region Backup
- Patching for your Amazon EC2 Instances

関連する例:

- 自動運用の改革 (パート I)
- 自動運用の改革 (パート II)
- aws-nuke による AWS リソースの削除の自動化
- AWS Config および AWS SSM を使用して未使用の Amazon EBS ボリュームを削除
- AWS での大規模な継続的コンプライアンスの自動化
- AWS Lambda による IT オートメーション

結論

コストの最適化とクラウドの財務管理は、継続的な取り組みです。財務チームやテクノロジーチームと定期的に協力し、アーキテクチャのアプローチをレビューし、コンポーネントの選択をアップデートしていく必要があります。

AWS が目指しているのは、お客様がコストの最小化と、弾力性、応答性、順応性に優れたデプロイを両立する後押しをすることです。デプロイコストを最適化するために、本書でご紹介したツール、テクニック、ベストプラクティスをぜひご活用ください。

寄稿者

本ドキュメントの寄稿者は次のとおりです。

- Amazon Web Services、Well-Architected コスト最適化の柱リーダー、Fatih (Ben) Mergen
- Amazon Web Services、コスト最適化部門ビジネス開発リーダー、Keith Jarrett
- Amazon Web Services、ビジネスデベロッパーマネージャー、Arthur Basbaum
- Amazon Web Services、コマーシャルアーキテクト、Jarman Hauser

詳細情報

詳細については、次を参照してください。

- AWS Well-Architected フレームワーク
- AWS アーキテクチャセンター

ドキュメントの改訂

このホワイトペーパーの更新に関する通知を受け取るには、RSS フィードにサブスクライブしてください。

変更	説明	日付
ベストプラクティスガイダンスの更新	複数のベストプラクティスの更新。新しいベストプラクティス COST06-BP04。	2024 年 6 月 27 日
ベストプラクティスガイダンスの更新	全体的なベストプラクティスのマイナー更新。	2023 年 12 月 6 日
ベストプラクティスガイダンスの更新	ベストプラクティスを更新し、柱全体における新しいガイダンスを追加。	2023 年 10 月 3 日
ベストプラクティスガイダンスの更新	ベストプラクティスがガバナンス、コストと使用状況のモニタリング、最適な料金モデルの選択、需要と供給のリソースの管理の分野で新しいガイダンスと共に更新されました。	2023 年 7 月 13 日
新しいフレームワークの更新	規範ガイダンスを使用してベストプラクティスを更新、および新しいベストプラクティスを追加。質問 COST 11 とベストプラクティス COST11-BP01 を追加。	2023 年 4 月 10 日
マイナーな更新	料金モデルのセクションに復元されたガイダンスがありません。	2023 年 1 月 13 日

ホワイトペーパーの更新	新しい実装ガイダンスを使用してベストプラクティスを更新。	2022 年 12 月 15 日
ホワイトペーパーの更新	ベストプラクティスに加筆し、改善計画を追加。	2022 年 10 月 20 日
マイナーな更新	イントロダクションに持続可能性の柱を追加。	2021 年 12 月 2 日
マイナーな更新	リンクを更新しました。	2021 年 4 月 25 日
マイナーな更新	リンクを更新しました。	2021 年 3 月 10 日
新しいフレームワークの更新	CFM、新しいサービス、Well-Architected との統合を追加。	2020 年 7 月 8 日
ホワイトペーパーの更新	AWS の変更内容を反映し、カスタマーレビューから学習した内容を追加。	2018 年 7 月 1 日
ホワイトペーパーの更新	AWS の変更内容を反映し、カスタマーレビューから学習した内容を追加。	2017 年 11 月 1 日
初版発行	コスト最適化の柱 - AWS Well-Architected フレームワークを公開。	2016 年 11 月 1 日

注意

お客様は、本書に記載されている情報を独自に評価する責任を負うものとします。本書は、(a) 情報提供のみを目的とし、(b) AWS の現行製品と慣行について説明しており、これらは予告なしに変更されることがあり、(c) AWS およびその関連会社、サプライヤー、またはライセンサーからの契約上の義務や保証をもたらすものではありません。AWS の製品やサービスは、明示または黙示を問わず、一切の保証、表明、条件なしに「現状のまま」提供されます。お客様に対する AWS の責任は AWS 契約によって規定されます。本書は、AWS とお客様との間で締結されるいかなる契約の一部でもなく、その内容を修正するものでもありません。

© 2023 Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS 用語集

AWS の最新の用語については、「AWS の用語集リファレンス」の「AWS 用語集」を参照してください。