AWS ホワイトペーパー

Amazon EC2 スポットインスタンスの概要

Amazon EC2 スポットインスタンスの概要: AWS ホワイトペーパー

Table of Contents

Amazon EC2 スポットインスタンスの概要

公開日: 2021 年 3 月 5 日 (ドキュメント履歴と寄稿者)

要約

このホワイトペーパーは、投資から得られる価値を最大化し、予測の精度とコスト予測可能性を向上させ、オーナーシップとコスト透明性の文化を醸成し、最適化の状態を継続的に測定できるようにユーザーを支援することを目的としています。

このホワイトペーパーでは、Amazon EC2 スポットインスタンスの概要と、その効果的な使用に関するベストプラクティスを解説します。

はじめに

オンデマンド、リザーブドインスタンス、Savings Plans に加え、Amazon Elastic Compute Cloud (Amazon EC2) の 4 番目の料金モデルとして、スポットインスタンスがあります。

スポットインスタンスのモデルでは、オンデマンド料金と比較して、最大 90% の割引料金で予備の Amazon EC2 コンピューティング容量を利用できます。これにより、アプリケーションの運用コストを大幅に削減したり、同じ予算でアプリケーションのコンピューティング性能やスループットを増大したりできます。オンデマンドインスタンス と スポットインスタンス の唯一の違いは、EC2 が容量を必要とするときに、EC2 が スポットインスタンス を中断できることです。この中断の際には、2 分前に通知があります。

リザーブドインスタンスや Savings Plans とは異なり、スポットインスタンスでは、コスト削減を実現するためにコミットメントが必要なく、オンデマンド料金よりも有利です。ただし、容量プール (インスタンスタイプとアベイラビリティーゾーンの組み合わせ) に利用可能な容量がない場合、EC2 はスポットインスタンスを終了する可能性があります。そのため、スポットインスタンスは柔軟なワークロードに最適です。

スポットインスタンスの使用が適切なケース

スポットインスタンス は、ステートレスかつフォールトトレラントで、柔軟性の高いさまざまなアプリケーションに適しています。たとえば、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 ハイパフォーマンスおよびハイスループットコンピューティング (HPC/HTC)、レンダリングワークロードなどの柔軟なワークロードに適しています。

スポットインスタンスは、柔軟性がない、ステートフル、フォールトイントレラント、またはインスタンスノード間で緊密に結合されているワークロードには適していません。また、ターゲット容量がフルに利用できない期間が時折あることに耐えられないワークロードにはスポットインスタンスは推奨されません。こうしたワークロードにスポットインスタンスを使用したり、中断に対処するためにオンデマンドインスタンスへのフェールオーバーを試みたりしないように強く警告します。

スポットインスタンスの起動方法

スポットインスタンスを起動するためのサービスには、Amazon EC2 Auto Scaling が最もお勧めです。このサービスを使用すると、必要な容量を起動して維持することができ、中断された、または手動で終了したリソースを置き換えるリソースを自動的にリクエストできます。Auto Scaling グループを設定する場合、アプリケーションのニーズに基づいてインスタンスタイプと希望する容量を指定するだけで済みます。詳細については、Amazon EC2 Auto Scaling グループのユーザーガイドの「Auto Scaling グループ」を参照してください。

さらに柔軟性が必要な場合、独自のインスタンス起動ワークフローを構築している場合、またはインスタンス起動やスケーリングメカニズムの側面を個別に制御したい場合は、EC2 Auto Scaling の代わりに、インスタントモードでの EC2 フリートの使用を検討することをお勧めします。この同期 API を使用すると、インスタンスタイプと起動要件のリストを指定でき、スポットインスタンスやオンデマンドインスタンスを起動するための EC2 RunInstances API コールよりも柔軟な機能が提供されます。

クラウドワークロードの実行に AWS のサービスを使用する場合は、スポットインスタンスの起動にも使用できます。たとえば、Amazon EMR、Amazon EKS、Amazon ECS、AWS Batch、AWS Elastic Beanstalk などがあります。AWS クラウドと統合できるサードパーティーのツールを使用してスポットインスタンスを起動することもできます。

Infrastructure as Code (AWS CloudFormation、AWS CDK など) や、AWS API、CLI、SDK などを使用すると、スポットインスタンスの起動を自動化できます。Spot Blueprints には、スポットのベストプラクティスに準拠する AWS CloudFormation および Hashicorp Terraform の Infrastructure as Code のテンプレートが生成できるガイド付きウィザードが用意されています。

スポットインスタンスのしくみ

スポットインスタンスは、稼動中は他の EC2 インスタンスとまったく同じように動作します。ただし、EC2 が容量を取り戻す必要がある場合、Amazon EC2 により中断される場合があります。

EC2 がスポットインスタンスを中断すると、選択した中断動作に応じて、インスタンスは終了、停止、または休止状態になります。

実行時間が 1 時間未満で EC2 によってスポットインスタンスが中断された場合、使用されたその 1 時間未満分は課金されません。ただし、お客様がスポットインスタンスを停止または終了した場合は、使用時間の端数分についても料金をいただきます (オンデマンドやリザーブドインスタンスと同様です)。異なるオペレーティングシステムで実行中に中断されたスポットインスタンスについての課金方法については、EC2 ユーザーガイドの「中断されたスポットインスタンスの請求」を参照してください。

各アベイラビリティーゾーンにおける各インスタンスタイプのスポット料金は、EC2 予備容量に対する需要と供給の長期トレンドに応じて決定されます。実際に使用した時間を秒単位で四捨五入したスポット料金が課金されます。

オプションで、スポットインスタンスの上限料金を指定できます。上限料金を指定しない場合、デフォルトの上限料金はオンデマンド料金となります。スポットインスタンスの実行時に、使用した分のスポット料金以上が課金されることは決してありません。上限料金は指定せず、上限料金をデフォルトのオンデマンド料金に設定することをお勧めします。上限料金を高く設定しても、スポットインスタンスが起動する可能性が高くなることも、スポットインスタンスが中断される可能性が低くなることもありません。これは、EC2 が容量を必要とした場合であれば、上限料金にかかわらず、お客様のスポットインスタンスを中断することがあるためです。

アベイラビリティーゾーンにおけるインスタンスタイプのスポット料金は随時変更される可能性がありますが、通常、頻繁には変更されません。AWS では DescribeSpotPriceHistory API 経由で、また API からのデータが反映される AWS マネジメントコンソールで、スポットインスタンスの現在のスポット料金と料金履歴を公開しています。これは、スポット料金の変動の度合いとタイミングを経時的に評価するうえで役立ちます。

スポットインスタンスの中断の管理

スポットインスタンスの中断を正常に処理し、パフォーマンスや可用性への影響を最小限に抑えるための最善の方法は、アプリケーションをフォールトトレラントな設計にすることです。これを実現するうえで、EC2 インスタンスの再調整に関する推奨事項とスポットインスタンスの中断通知を利用できます。

EC2 インスタンスの再調整に関する推奨事項とは、スポットインスタンスの中断リスクが高くなると送信される通知です。この通知は、スポットインスタンス中断 2 分前の通知の前にスポットインスタンスをプロアクティブに管理する機会を提供します。ワークロードを、中断のリスクが高くない新しいスポットインスタンスまたは既存のスポットインスタンスに再調整することができます。容量の再調整機能は EC2 Auto Scaling グループで提供されており、この通知機能が簡単に使用できるようになっています。詳細については、「Amazon EC2 Auto Scaling Capacity Rebalancing」を参照してください。

スポットインスタンスの中断通知は、Amazon EC2 によってスポットインスタンスが中断される 2 分前に送信される警告です。ワークロードに「時間の制約がない」場合、スポットインスタンスが中断される際に、終了するのではなく、停止または休止させるようにスポットインスタンスを設定できます。中断が起こるとスポットインスタンスは自動的に停止または休止状態になり、使用可能な容量ができると自動的にインスタンスが再開されます。

EC2 インスタンスの再調整に関する推奨事項やスポットインスタンスの中断通知を使用し、フォールトトレランスを考慮に入れてワークロードを設計すると、通知をキャプチャしてジョブの状態をストレージ (Amazon S3、Amazon EFS、Amazon FSx など) に保存したり、インスタンスからのログファイルを維持したり (またはフォールトトレラントなアプローチをとって連続的にストリーミングしたり)、ロードバランサーからの接続のドレインを実行したりできます。

AWS およびサードパーティーのサービスには、アプリケーションへの影響を低減するために、スポット中断の処理機能をすでに提供しているものもあります。たとえば、スポットインスタンスを使用したマネージド型ノードグループを実行する Amazon EKS の場合、既存のノードについての再調整に関する推奨事項や中断通知を受け取ると、自動的に代わりの Kubernetes ノードを起動します。

スポットインスタンスの制限

実行およびリクエストされるスポットインスタンスの数には、リージョンごとに AWS アカウントあたりの制限があります。スポットインスタンスの上限は、実行中のスポットインスタンスが使用中であるか、未処理のスポットインスタンスリクエストが受理されると使用することになる仮想中央演算装置 (vCPU) の数をもとに管理されます。ユーザーがスポットインスタンスを終了したが、スポットインスタンスリクエストをキャンセルしなかった場合、Amazon EC2 がスポットインスタンスの終了を検出してリクエストを閉じるまで、リクエストはスポットインスタンスの vCPU 上限に対してカウントされます。

スポットインスタンスには、以下の 6 種類の制限があります。

- すべての標準 (A、C、D、H、I、M、R、T、Z) スポットインスタンスリクエスト
- すべての F スポットインスタンスリクエスト
- すべての G スポットインスタンスリクエスト
- すべての Inf スポットインスタンスリクエスト
- すべての P スポットインスタンスリクエスト
- すべての X スポットインスタンスリクエスト

各制限は、1 つ以上のインスタンスファミリーの vCPU 制限を指定します。さまざまなインスタンスファミリー、世代、およびサイズの詳細については、「Amazon EC2 インスタンスタイプ」を参照してください。

vCPU の制限には、変化するアプリケーションのニーズに合わせて、任意の組み合わせのインスタンスタイプを起動するために必要な vCPU の数に関して、お客様の制限を使用できます。例えば、オールスタンダードスポットインスタンスリクエストの制限が 256 vCPU の場合であれば、32 個の `m5.2xlarge` スポットインスタンス (32 x 8 vCPU) または 16 個の`c5.4xlarge` スポットインスタンス (16 x 16 vCPU) をリクエストできます。あるいは、合計 256 個の vCPU までであれば、スタンダードスポットインスタンスのタイプとサイズを任意に組み合わせてリクエストできます。

詳細については、Linux インスタンス用 Amazon EC2 ユーザーガイドの「スポットインスタンスの上限と使用量のモニタリング」と「スポットインスタンス制限の引き上げをリクエストする」を参照してください。

スポットインスタンスのベストプラクティス

お客様のアプリケーションにベストプラクティスをどのように適用するかは、お客様のインスタンスタイプ要件、予算要件、アプリケーション設計によって決まります。

- インスタンスタイプに関して柔軟に対応する。 スポットインスタンスプールは、同じインスタンスタイプ (m5.large など) とアベイラビリティーゾーン (us-east-1a など) を持つ、未使用の EC2 インスタンスのセットです。どのインスタンスタイプをリクエストし、どのアベイラビリティーゾーンでワークロードをデプロイするか柔軟に対応することで、スポットが必要な量のコンピューティング容量を見つけ、割り当てられる可能性が高くなります。たとえば、c4、m5、m4 ファミリのラージを使用してもよいのであれば、「c5.large」を指定する必要はありません。
- 容量を最適化する割り当て戦略を使用する。 EC2 Auto Scaling グループの割り当て戦略を使えば、予備容量を持つスポットインスタンスプールを手動で探す必要なく、ターゲット容量をプロビジョニングできます。最も可用性の高いスポットインスタンスプールからインスタンスが自動的にプロビジョニングされる、容量最適化戦略を採用することをお勧めします。最適な容量を持つプールからスポットインスタンスの容量が供給されるため、スポットインスタンスが中断される可能性が低減します。割り当て戦略の詳細については、Amazon EC2 Auto Scaling ユーザーガイドの「スポットインスタンス」を参照してください。
- プロアクティブな容量再調整機能を使用する。 容量の再調整機能を使用すると、実行中のスポットインスタンスが 2 分間の中断通知を受け取る前に、新しいスポットインスタンスを使用して Auto Scaling グループをプロアクティブに拡張することにより、ワークロードの可用性を維持できます。容量の再調整が有効になっている場合、Auto Scaling は、再調整に関する推奨事項を受け取ったスポットインスタンスを積極的に置き換えることを試みます。これにより、中断のリスクが高くない新しいスポットインスタンスにワークロードを再調整することができます。
- 統合された AWS のサービスを使用してスポットインスタンスを管理する。 他の AWS のサービスは、個々のインスタンスやフリートを管理する必要なく、全体的なコンピューティングコストを削減できるよう、スポットと統合されています。該当するワークロードについて、Amazon EMR、Amazon ECS サービス、AWS Batch、Amazon EKS、SageMaker、AWS Elastic Beanstalk、Amazon GameLift の採用を検討することをお勧めします。これらのサービスに関するスポットのベストプラクティスの詳細については、「Amazon EC2 スポットインスタンスのワークショップ」のサイトを参照してください。
- スポットインスタンス向けの最新かつ適正な起動ツールを選択する。 AWS に統合されたサービスのいずれかがワークロードに適さず、スポットインスタンスの起動を制御する機能を持つアプリケーションを構築する必要がある場合は、適切なツールを使用してください。ほとんどのワークロードでは、EC2 Auto Scaling が適切です。ELB ベースのアプリケーション、コンテナ化された

ワークロード、キュー処理ジョブなど、幅広いワークロード向けの非常に包括的な機能セットを提供しているためです。個別のリクエストをより細かく制御する必要があり、「起動専用」ツールが求められる場合は、RunInstances に対する当座の代用として、インスタントモードで EC2 フリートを使用します。EC2 フリートは、インスタンスタイプの分散化や配分戦略などの幅広い機能セットを備えています。

スポットとその他の AWS のサービスとの統合

Amazon EC2 スポットインスタンスは、いくつかの AWS のサービスと統合されています。

Amazon EMR との統合

Amazon EMR クラスターをスポットインスタンスで実行すると、分析ワークロードの膨大な量のデータを処理するコストを大幅に削減できます。EMR インスタンスフリート機能を使用すると、スポットインスタンスをオンデマンドインスタンスやリザーブドインスタンスと簡単に組み合わせて、EMR クラスターを実行できます。EMR 割り当て戦略を使用すると、最も可用性の高い容量プールからスポットインスタンスを起動できます。

EC2 Auto Scaling との統合

Amazon EC2 Auto Scaling グループを使用すると、スポットインスタンスの起動と管理、アプリケーションの可用性の維持、インスタンスタイプと購入オプション (オンデマンドまたはスポット) の選択の多様化、動的でスケジュールできる予測スケーリングポリシーを採用した Amazon EC2 容量のスケーリングを実現できます。詳細については、Amazon EC2 Auto Scaling ユーザーガイドの「Requesting Spot Instances for fault-tolerant and flexible applications」を参照してください。

Amazon EKS との統合

Amazon EKS を使用すると、EKS マネージド型ノードグループでスポットインスタンスを起動して、Kubernetes ベースのワークロードのコストを最適化できます。EKS マネージド型ノードグループは、中断される直前のスポットインスタンスを新しく起動したインスタンスに置き換えることで、スポットインスタンスのライフサイクル全体を管理して、スポットインスタンスの中断時 (EC2 が容量を取り戻す必要がある場合) に、アプリケーションのパフォーマンスや可用性に影響がおよぶ可能性を低減します。詳細については、Amazon EKS ユーザーガイドの「マネージド型ノードグループ」を参照してください。

Amazon ECS との統合

Amazon ECS クラスターをスポットインスタンスで実行すると、コンテナ化されたアプリケーションを実行する運用コストを削減できます。Amazon ECS は、中断される直前のスポットインスタンスの自動ドレイン機能をサポートしています。詳細については、Amazon Elastic Container Service デベロッパーガイドの「スポットインスタンスの使用」を参照してください。

Amazon ECS と AWS Fargate Spot の統合

コンテナ化されたタスクが中断でき、柔軟性がある場合は、AWS Fargate Spot キャパシティプロバイダーを使用して ECS タスクを実行できます。タスクはサーバーレスコンテナプラットフォームである AWS Fargate で実行され、Fargate Spot によりコスト削減のメリットが得られます。詳細については、Amazon Elastic Container Service デベロッパーガイドの「AWS Fargate キャパシティプロバイダー」を参照してください。

Amazon Batch との統合

AWS Batch は、AWS 上のバッチコンピューティングのワークロードを計画、スケジュール、実行します。AWS Batch は、ユーザーに代わってスポットインスタンスを動的にリクエストできるため、バッチジョブの運用コストを削減できます。

Amazon SageMaker との統合

Amazon SageMaker では、マネージド型のスポットインスタンスを使用して、機械学習モデルを簡単にトレーニングできます。マネージドスポットトレーニングを採用すると、オンデマンドインスタンスと比較して、トレーニングモデルのコストが最大 90% 最適化されます。SageMaker は、ユーザーに代わってスポットの中断を管理します。詳細については、Amazon SageMaker デベロッパーガイドの「Amazon SageMaker のマネージドスポットトレーニング」を参照してください。

Amazon Gamelift との統合

Amazon GameLift は、マルチプレイヤーゲーム向けにクラウドサーバーをデプロイ、運用、スケーリングする、ゲームサーバーホスティングソリューションです。Amazon Gamelift は、スポットインスタンスをサポートしており、ホスティングコストを大幅に削減できます。ホスティングリソースのフリートを作成する際、オンデマンドインスタンスとスポットインスタンスのいずれかを選択できます。スポットインスタンスは 2 分前の通知の後に中断される可能性がありますが、Amazon GameLift の FleetIQ により、中断の可能性は最小限に抑えられます。詳細については、Amazon GameLift デベロッパーガイドの「GameLift でのスポットインスタンスの使用」を参照してください。

AWS Elastic Beanstalk との統合

AWS Elastic Beanstalk は、Java、.NET、PHP、Node.js、Python、Ruby、Go、Docker を使用して開発されたウェブアプリケーションやサービスを、Apache、Nginx、Passenger、IIS などの使い慣

れたサーバーでデプロイおよびスケーリングするための、使いやすいサービスです。コードをアップロードするだけで、キャパシティーのプロビジョニング、ロードバランシング、オートスケーリングからアプリケーションの状態モニタリングまで、Elastic Beanstalk がデプロイを自動的に処理します。Elastic Beanstalk 環境でスポットインスタンスを使用すると、ウェブアプリケーションの基盤となるインフラストラクチャのコストを最適化できます。Elastic Beanstalk でスポットインスタンスを使用する方法については、AWS Elastic Beanstalk デベロッパーガイドの「スポットインスタンスのサポート」を参照してください。

まとめ

コンピューティングのニーズが柔軟な場合でも、予算を増やさず容量の増強を望んでいる場合でも、スポットインスタンスは AWS のコストの最適化、拡張性を考慮した構築のための優れた方法です。ワークロードのアーキテクチャを適切に設計することにより、スポットインスタンスを活用して幅広いニーズに対応できます。詳細については、「Amazon EC2 スポットインスタンス」を参照してください。

リソース

- AWS アーキテクチャセンター
- AWS ホワイトペーパー
- AWS 月間アーキテクチャ
- AWS アーキテクチャブログ
- This is My Architecture の動画
- AWS ドキュメント

ドキュメント履歴と寄稿者

ドキュメント履歴

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 フィードをサブスクライブしてください。

update-history-change	update-history-description	update-history-date
マイナーな更新	ページレイアウトを調整。	2021 年 4 月 30 日
マイナーな更新	最新のベストプラクティスを反映するようにコンテンツを更新。内容をより適切に反映するように、ホワイトペーパーの名称を「Amazon EC2 スポットインスタンスの大規模な活用」から「Amazon EC2 スポットインスタンスの概要」に変更。	2021 年 3 月 5 日
マイナーな更新	スポットインスタンスの制限を更新。	2021 年 2 月 3 日
初版公開	Amazon EC2 スポットインスタンスの大規模な活用を公開。	2018 年 3 月 1 日

> **Note**
>
> RSS 更新を購読するには、使用しているブラウザで RSS プラグインを有効にする必要があります。

寄稿者

本書の執筆に当たり、次の人物および組織が寄稿しました。

- AWS、シニアプロダクトマーケティングマネージャー、Amilcar Alfaro
- AWS、マーケティングマネージャー、Erin Carlson
- AWS ビジネス開発、WW BD リード - コスト最適化担当、Keith Jarrett
- AWS、プリンシパルソリューションアーキテクト、Ran Sheinberg