AMAZON EC2 スポットインスタンスと SPOT（旧 SPOTINST）について

スポットインスタンスの活用

皆さん、Amazon EC2 を活用されていますか？
Amazon Elastic Compute Cloud(Amazon EC2、以下 EC2)は、利用したいリソースを利用した時間だけ料金を支払うオンデマンドでの料金体系ですが、他にも安く使うための様々な仕組みがあります。
本記事では、その仕組みのご紹介と、特に"スポットインスタンス"について掘り下げていきます。
皆様の EC2 の利用コストの最適化の方法を是非習得してください！

EC2 の料金体系とスポットインスタンス

ここでは、EC2 で利用可能な料金プランの概要を解説します。

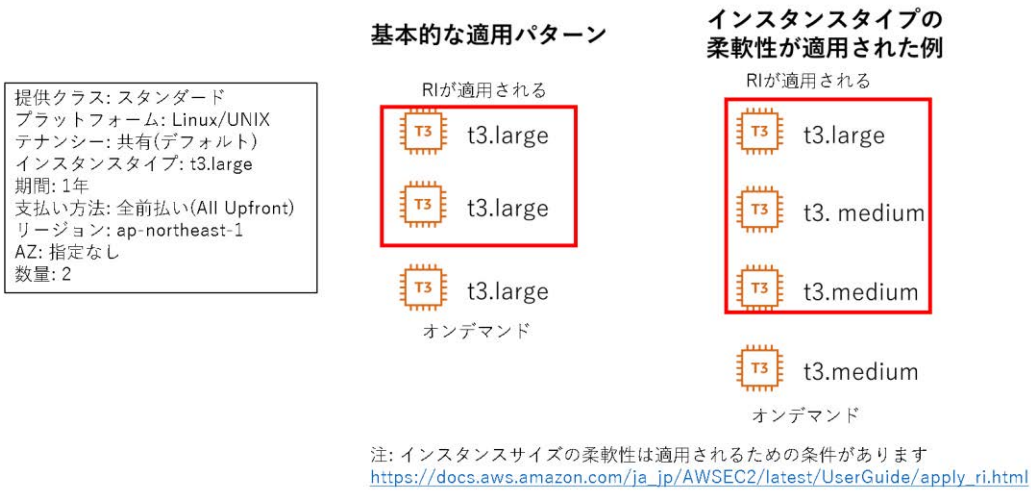
オンデマンド

通常の支払い方法です。リージョン毎に対応するインスタンスタイプの 1 時間あたりの価格が設定されており、利用した分だけ課金されます。現在では、Free 系の Linux については秒単位の課金(最小 60 秒)となります。Windows や RedHatEnterpriseLinux などの商用 OS の場合は 1 時間単位の課金となります。

リザーブドインスタンス

オンデマンドが利用分を都度払う仕組みだったのに対して、リザーブドインスタンスは予め利用するリソースの条件を指定し、1 年または 3 年間利用する契約とすることで割引を得ることができます。契約形態や支払い方法によって割引率が異なるため、利用する予定のユースケースに照らし合わせて選択する必要があります。本記事では詳細は割愛しますが、常時稼働させるようなユースケースで特に有効なコスト削減手段です。ただし、条件指定が一致しないと適用されないので運用に注意が必要なプランでもあります。

リザーブドインスタンス 使用例

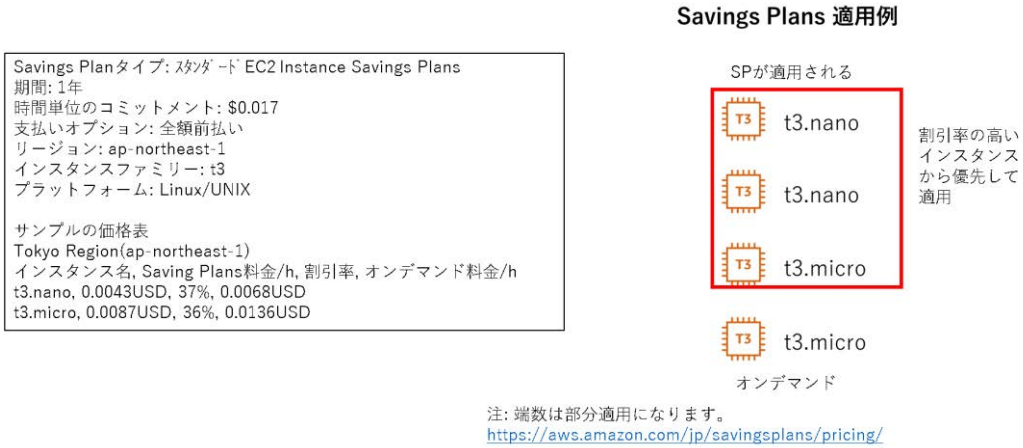


注: インスタンスサイズの柔軟性は適用されるための条件があります
https://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/apply_ri.html

SAVINGS PLANS

購入する期間はリザーブドインスタンスと同様に 1 年または 3 年間利用する契約です。ただし、リザーブドインスタンスと比べて適用基準が柔軟になります。1 時間あたりに使用する金額を確定することで、その金額に達成するまでのオンデマンド利用分が割引されます。EC2 の特定インスタンスファミリーに適用するプランと、より広範囲にコンピュータリソースを対象とするパターンを選択出来ます。

Savings Plans 使用例



注: 端数は部分適用になります。
https://aws.amazon.com/jp/savingsplans/pricing/

スポットインスタンス

Amazon は EC2 をはじめとするコンピュータリソースサービスをホストするために、データセンターに多数のサーバーリソースを保持しています。利用されていないサーバーリソースについて、オンデマンド価格と比べて安価に提供しています。ただし、SLA が保証されず、AWS 側の都合で停止されてしまうなど、通常の利用方法と比べていくつか制限があるため、利用にあたって注意が必要です。詳細について以降で述べます。利用方法はオンデマンドでの起動の場合に似ていて、スポットリクエストによる実行が一般的です。オンデマンドと同様必要事項を入力しますが、リクエストの条件を指定し、その指定条件ないで起動可能なリソースを確保してインスタンスを起動します。

スポットインスタンス 起動例

ジョブタイプ: Flexible workloads
最小コンピュートユニット: インスタンスタイプ t3.micro
アベイラビリティーゾーン: 指定なし
合計ターゲット容量: 2
フリートリクエスト: t3.micro, t3a.micro
上記以外デフォルト値または説明で関係ないため割愛

スポットインスタンスとは？

ここまででご説明した料金プランの中で価格的には最も安くなる可能性があるスポットインスタンスですが、その特性は他の料金プランと大きく異なります。

メリット

利用料金は非常に安価です。オンデマンドと比較して最大 90%割引されます。価格はスポットインスタンス用の市場価格が適用されます。需要と供給により決まります。現在の価格は、Web サイトで確認することが出来ます。

https://aws.amazon.com/jp/ec2/spot/pricing/
https://aws.amazon.com/jp/ec2/spot/instance-advisor/

制限事項

スポットインスタンスは格安である一方、様々な制限事項があるため、利用用途を注意する必要があります。

1. 有休リソースがある場合にのみ利用可能
スポットインスタンスは有休リソースを活用しています。多くのユーザが他の料金プランやコンピュートリソースを利用している場合に、スポットインスタンス用のキャパシティが確保できない場合があり、その際はそもそもインスタンスを起動することが出来ません。また、後述する AWS の都合による停止が発生する可能性があります。

2. AWS 側の都合で停止される
前述のように、スポットインスタンスは有休リソースを活用していますが、キャパシティ予約済みのリクエストや、オンデマンド利用などの他の料金プランのリクエストが増えて該当するアベイラビリティーゾーンのキャパシティが枯渇した時に、稼働しているスポットインスタンスを終了してキャパシティを回復しようとします。その結果、終了対象と判断された場合、メタデータや CloudWatch Event で停止を通知した上で、2 分後に終了処理を実行します。

終了処理から 1 分以内に終了できない場合は、最終的に強制停止が実行されるため、時間内に OS のシャットダウン処理や必要な場合データの保存などをしておく必要があります。
なお、インスタンス起動時の設定に応じて終了の代わりに停止を選択して EBS を再利用することは可能です。
停止される順序は、スポットインスタンスのリクエスト実施時に設定した入札金額によります。現在のスポットインスタンス市場価格が入札金額より高くなった場合は、すぐに停止対象となります。市場価格より入札価格が高い場合は、該当インスタンスタイプのうち入札価格の低い順番で停止対象となります。

3. SLA が保証されない
スポットインスタンスは AWS の都合で停止する可能性があるため、SLA の対象外となります。

スポットインスタンスの活用方法

先にあげたような特性があるため、スポットインスタンスを利用する場合、ユースケースを考える必要があります。

1. ステートレス
EBS を保持可能とはいえ、状態を保持するユースケースにはあまり向いていません。上で稼働させるアプリケーションのアーキテクチャをステートレス前提でするなどの対応が必要です。

2. 短時間稼働
短時間で利用して終了出来るような処理に向いています。たとえばバッチ処理等です。6 時間以内であれば、短時間のスポットインスタンスの利用も可能で、この場合は指定時間以内に強制停止することはありません。

また、システムとしてリソースを維持できるようにするための仕組みが用意されているので、それを前提としたシステムアーキテクチャを考えることも視野に入れて下さい。

SPOTFLEET によるキャパシティ維持

必要なインスタンス数を確保するための仕組みとして、Spot Fleet があります。スポットリクエスト作成時に、必要なインスタンス数を指定しておき、強制停止によってインスタンス数が減った場合に別のインスタンスを起動させることでインスタンス数を維持するようにできます。AWS 側のキャパシティの都合でスポットリクエストが実行できない場合にそなえて、オンデマンドインスタンスを併用するように設定することもできます。この仕組みによって、常に指定したインスタンス数を確保することが出来るようになります。ただしこの仕組みでは指定した台数にすることのみに特化しているため、Web システムなどのスケールアウト/スケールインは行えません。

AUTO SCALING によるスケールイン/スケールアウト

AWS では、様々な条件によるインスタンス数の増減手段として Auto Scaling による自動的な増減処理が実行可能です。その際にスポットインスタンスを利用することが可能です。スポットインスタンスの強制停止が発生したとしても、Auto Scaling Group 側で Spot Fleet の機能を活用してインスタンス数の維持を行うことが出来ます。さらには負荷増大などのイベントに対してスポットインスタンスによるインスタンス数を増やしたり、不要になった追加分のインスタンスを減らしたりすることも可能です。

ユースケースとしては、最低限維持が必要なインスタンス数についてはオンデマンドによるキャパシティ予約を含めた構成、負荷による一時的な増加分はスポットインスタンスで賄う、などの柔軟な構成が可能です。

SPOT（旧 SPOTINST）の活用

上記のように、AWS ではスポットインスタンスを活用してのシステムを作りこむ為の仕組みはいろいろ用意されていますが、いくつか課題があります。

1. ステートフルなユースケースへの対応
EBS の維持はできますが、完全なインスタンスの維持はできません。EBS のディアタッチから別のインスタンスで再アタッチするため、インスタンスとしては別物になります。

2. 停止時間の制限
停止対象になってから停止処理開始まで 2 分しか時間がないため、通常のシャットダウンに時間がかかるようなサーバーの場合正常終了が出来なくなってしまいます。

3. より積極的なコスト最適化
Spot Fleet により複数のゾーンやインスタンスタイプから必要な台数分のインスタンスを確保することはできますが、そこには戦略的な視点は反映されません。コストを重視したい、あるいは安定して稼働させたい、といったニーズに細かく答えることは難しいです。

これらの課題を解決するソリューションとして、Spot という SaaS サービスがあります。

Spot では、AI による予測アルゴリズムに基づき、最適なスポットインスタンスの活用を行うことが出来ます。AWS の Auto Scaling や Spot Fleet の仕組みは、指定したインスタンス台数を維持するために、スポットインスタンスが停止した際に、新たなスポットインスタンスを立ち上げるという動きをします。
それに対して、Spot のサービスでは、Elastigroup という機能により、AI でスポットインスタンスの停止が発生する予兆を 10 分程度前に予測し、あらかじめ代替えとなるスポットインスタンスをあらかじめ起動して停止予定のものと入れ替えることで、必要なインスタンス数を下回ることなく必要なインスタンス数を維持することが可能です。

Elastigroup によるインスタンスの入れ替え

また、Elastigroup のステートフルを維持する機能により、利用していた IP を引継ぎ、ディスクデータを維持して入れ替えることでステートフルな状態維持を可能としています。

Elastigroup では AWS のさまざまなサービスを使ったユースケースに対応しているため、AWS でよく使うシステムアーキテクチャに柔軟に適用することが可能です。

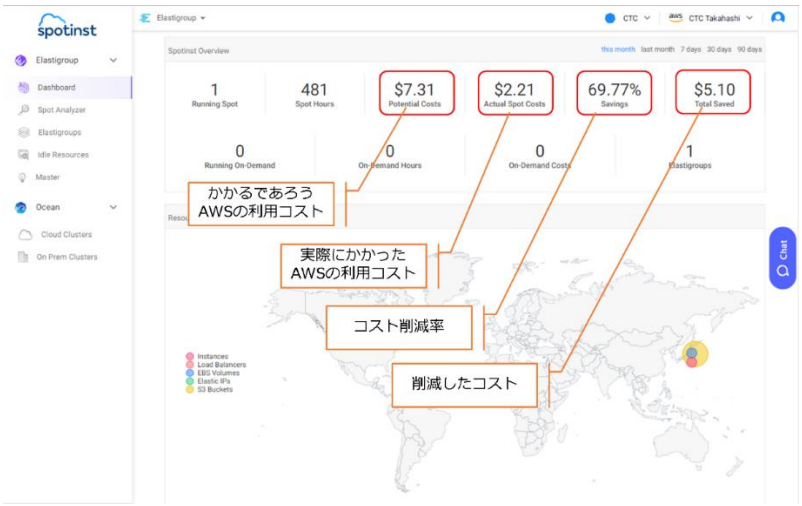
AWS ユースケース

Spot では、Elastigroup の活用により、SLA を保証することもできます。

また、Spot の利用価格はオンデマンド利用料金とスポットインストの差額金額の一部となっており、差額が発生しない場合は利用料金も発生しないことになるため、導入により金銭的に不利になることはありません。

ダッシュボードによるコストに関する可視化も用意なため、コスト可視化ツールとしての利用やレポーティング用の素材としての利用も可能となっています。

Spot ダッシュボード



Spot では 14 日間の無償トライアルを提供しています。
ご興味ありましたら気軽にお試しください。