



[AWS Black Belt Online Seminar]

Amazon Elastic Block Store (Amazon EBS)

サービスカットシリーズ

Solutions Architect 池田正一
2019/3/20

AWS 公式 Webinar

<https://amzn.to/JPWebinar>



過去資料

<https://amzn.to/JPArchive>



自己紹介



池田正一

エンタープライズソリューション本部
ソリューションアーキテクト

- 大手のお客様を担当し、情シス部門からLOBまで幅広くお客様のAWSの利用に関して、アーキテクチャをデザインするご支援
- 好きなAWSのサービス：EBS、AWS Certificate Manager



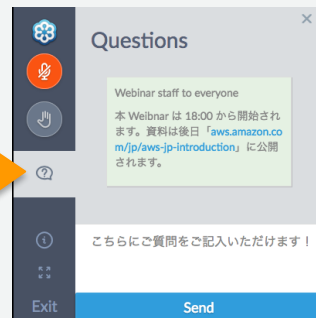
AWS Black Belt Online Seminar とは

「サービス別」「ソリューション別」「業種別」のそれぞれのテーマに分かれて、アマゾンウェブ サービス ジャパン株式会社が主催するオンラインセミナーシリーズです。

質問を投げることができます！

- 書き込んだ質問は、主催者にしか見えません
- 今後のロードマップに関するご質問は
お答えできませんのでご了承下さい

- ① 吹き出しをクリック
- ② 質問を入力
- ③ Sendをクリック



Twitter ハッシュタグは以下をご利用ください
#awsblackbelt

内容についての注意点

- 本資料では2019年3月20日時点のサービス内容および価格についてご説明しています。最新の情報はAWS公式ウェブサイト(<http://aws.amazon.com>)にてご確認ください。
- 資料作成には十分注意しておりますが、資料内の価格とAWS公式ウェブサイト記載の価格に相違があった場合、AWS公式ウェブサイトの価格を優先とさせていただきます。
- 価格は税抜表記となっています。日本居住者のお客様が東京リージョンを使用する場合、別途消費税をご請求させていただきます。
- 公式ドキュメント上で、TBのものはTB、TiBのものはTiBで表記しております。
- AWS does not offer binding price quotes. AWS pricing is publicly available and is subject to change in accordance with the AWS Customer Agreement available at <http://aws.amazon.com/agreement/>. Any pricing information included in this document is provided only as an estimate of usage charges for AWS services based on certain information that you have provided. Monthly charges will be based on your actual use of AWS services, and may vary from the estimates provided.

本セミナーの目的

- Amazon Elastic Block Store(EBS)の概要を理解する
- EBSの環境をより最適に利用する方法を身につける
- EBSの最新アップデートを身につける

本日のアジェンダ

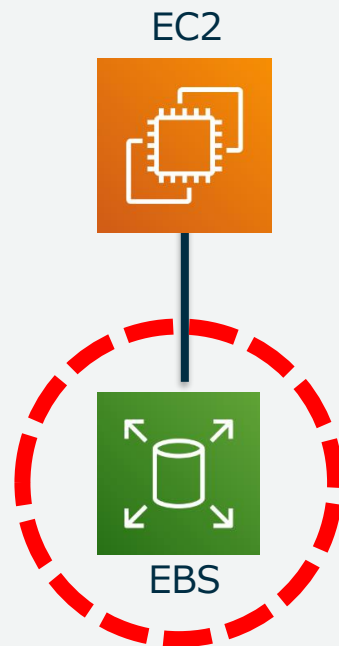
- Amazon Elastic Block Store (EBS) 概要 ～アップデートを含めて～
 - EBSの概要
 - タイプ別の特徴
 - Elastic Compute Cloud (EC2)インスタンスとパフォーマンスの考慮
 - 監視
 - NVMe SSD
- EBSの機能
 - Elastic Volume
 - EBS Snapshot
 - 暗号化
- EBSの価格
- まとめ

本日のアジェンダ

- Amazon Elastic Block Store (EBS) 概要 ～アップデートを含めて～
 - EBSの概要
 - タイプ別の特徴
 - EC2インスタンスとパフォーマンスの考慮
 - 監視
 - NVMe SSD
 - EBSの機能
 - Elastic Volume
 - EBS Snapshot
 - 暗号化
 - EBSの価格
 - まとめ

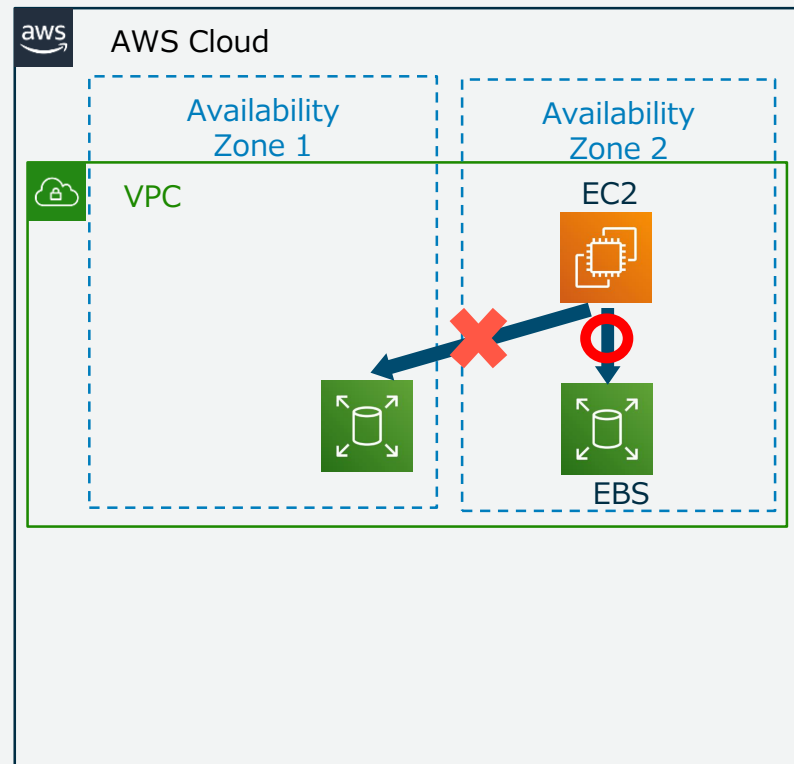
Amazon Elastic Block Store (EBS) の特徴

- Amazon Elastic Compute Cloud (EC2) インスタンスにアタッチして使用するブロックレベルのストレージサービス
- OS やアプリケーション、データの置き場所など様々な用途で利用される
- Snapshot 機能による S3 へのバックアップや、ディスクの暗号化機能を提供
- 99.999%の可用性を備えるように設計されている



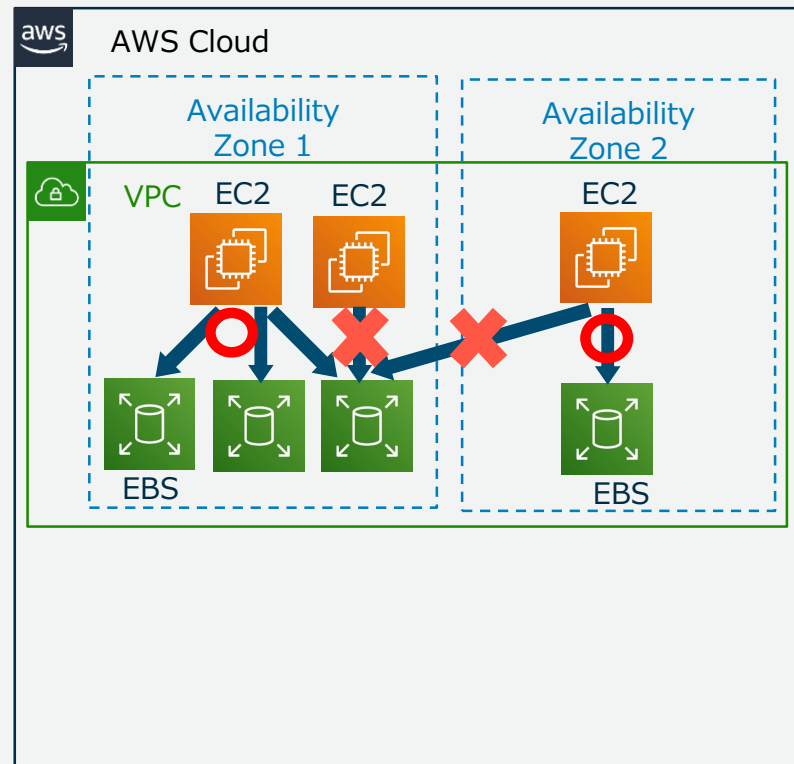
Amazon Elastic Block Store (EBS) の特徴

- 容量は1 GiB 単位で指定できる。最大容量は16 TiB
- アベイラビリティゾーン(AZ)毎に独立しているため、同一 AZ のインスタンスからのみ利用可能



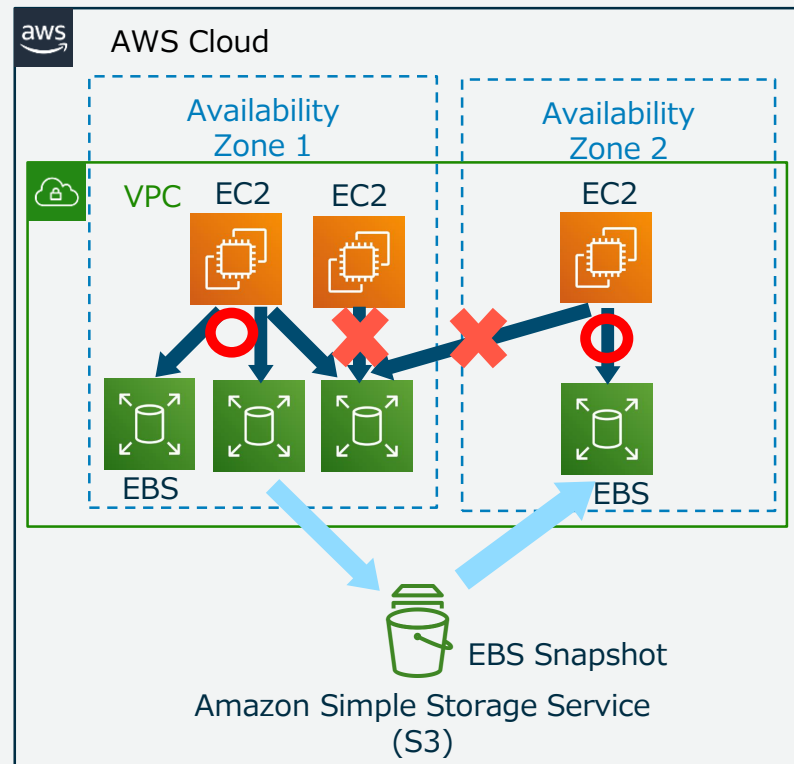
Amazon Elastic Block Store (EBS) の特徴

- 容量は1 GiB 単位で指定できる。最大容量は16 TiB
- アベイラビリティゾーン(AZ)毎に独立しているため、同一 AZ のインスタンスからのみ利用可能
- EC2 インスタンスに複数の EBS を接続することはできるが、EBS を複数のインスタンスで共有することはできない



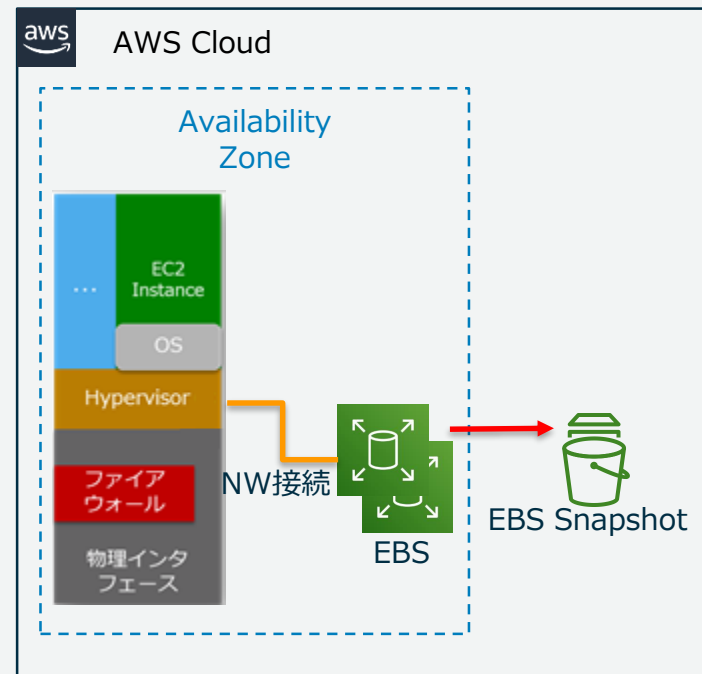
Amazon Elastic Block Store (EBS) の特徴

- 容量は1 GiB 単位で指定できる。最大容量は16 TiB
- アベイラビリティゾーン(AZ)毎に独立しているため、同一 AZ のインスタンスからのみ利用可能
- EC2 インスタンスに複数の EBS を接続することはできるが、EBS を複数のインスタンスで共有することはできない
- EBSのバックアップとして S3 に Snapshot を取得し、任意の AZ に復元できる



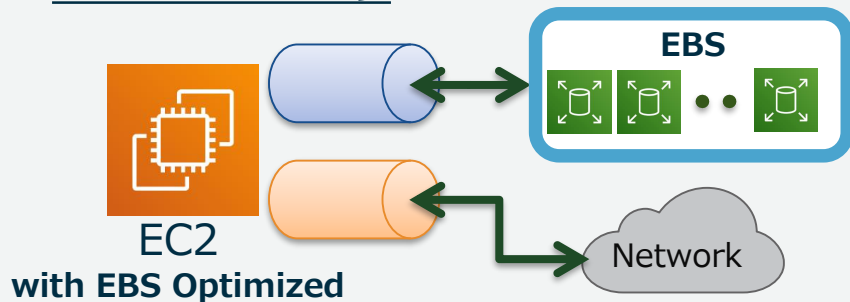
EBS の基本的なアーキテクチャ

- ボリュームのデータは AZ 内で複数の HW にレプリケートされており、一般的にはさらなる冗長化のためのRAID構成は不要
- 実体はネットワーク接続型ストレージだが、ユーザはネットワークを意識する必要はない
- セキュリティグループによる通信制御の対象外。全ポートを閉じてても EBS は利用できる

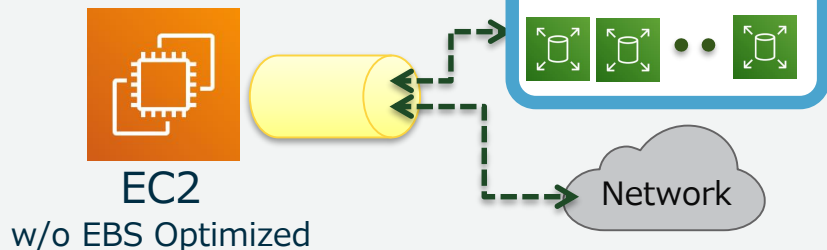


EBS 最適化インスタンス

EBS最適化あり



EBS最適化なし



- EBS最適化インスタンスは、独立した帯域を確保しI/O性能の安定化に繋がる
- c3/m3/r3/t2などの旧世代インスタンスを除いてデフォルトでオンになっている
- EBS との専用の帯域として56.25 MB /s から 1,750 MB /s まで、大きいインスタンスタイプほど使える帯域が広い

https://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/EBSOptimized.html

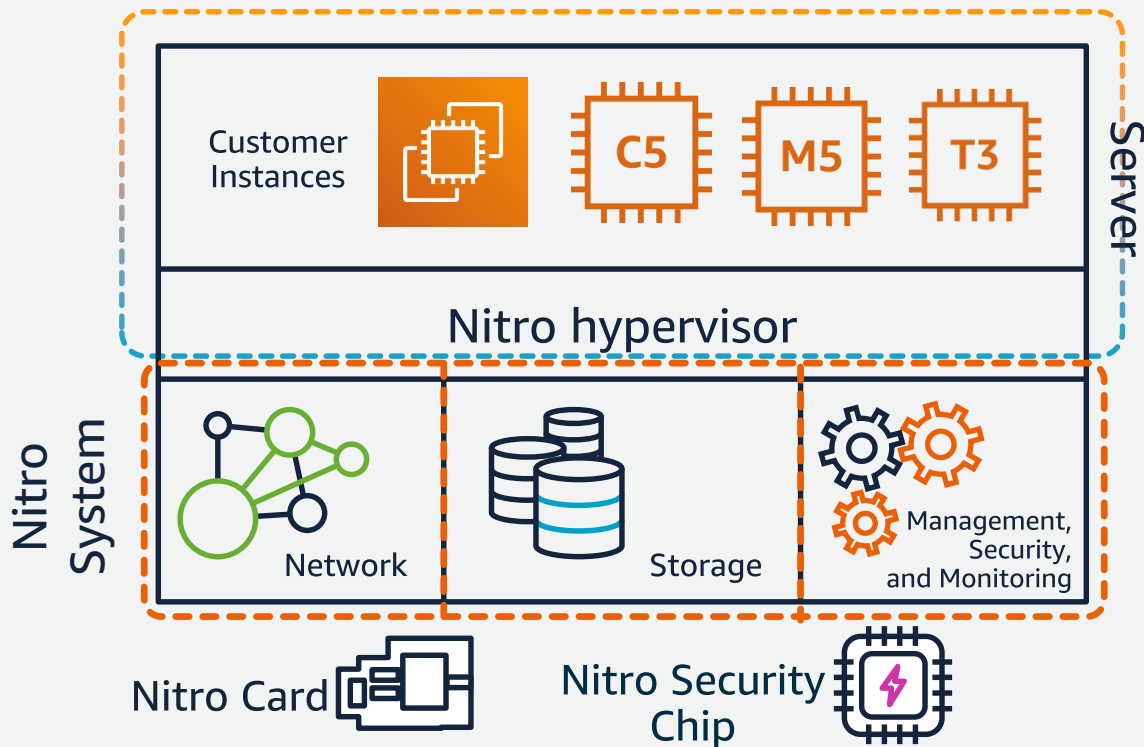
EC2のシステム基盤 (AWS Nitro System)

独自のハードウェア/Hypervisorにより最適化された性能を提供

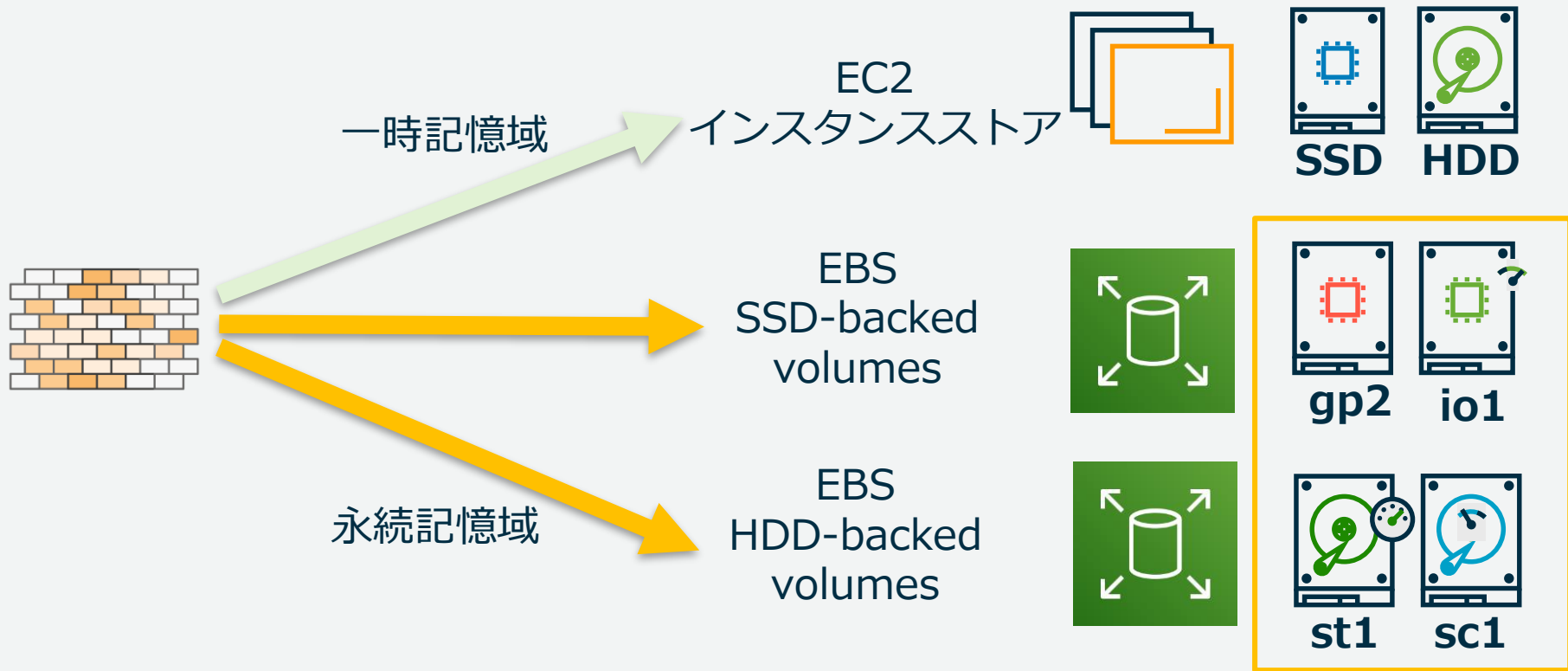
C5、M5など最新のインスタンスは、EC2ソフトウェアスタック全体を専用ハードウェアへオフロード

例えば、EBSへ書き込む処理やEBSの暗号化の処理をオフロード

最適化されたバージョンのLinux KVM をベースにした完全に新しいEC2 Hypervisor (C5,M5より前はXenベースのHypervisorを使用)



ブロックストレージの種類



https://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/InstanceStorage.html

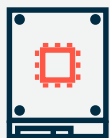
© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



EBSのユースケース

SSD

汎用SSD



gp2

- ブートボリューム
- 負荷が読めないシステム
- 小規模なデータベース
- 開発・テスト環境
- 仮想デスクトップ

プロビジョンド

IOPS



io1

- リレーショナルデータベース
 - MySQL
 - PostgreSQL
 - Oracle
 - Microsoft SQL Server
- NoSQL データベース
 - MongoDB
 - Cassandra
 - ElasticSearch
- 持続的なIOPSパフォーマンスが必要なアプリケーション

HDD

スループット 最適化HDD



st1

- ビックデータ、分析
 - Hadoop
 - Splunk
 - データウェアハウス

コールドHDD





sc1



- ログデータ
- アーカイブ
- 低頻度アクセスの大量データ

EBSのボリュームタイプ (SSDタイプ)



ボリュームタイプ	汎用SSD (gp2) - General Purpose SSD	プロビジョンドIOPS (io1) - Provisioned IOPS(SSD)
ユースケース	<ul style="list-style-type: none">システムブートボリューム仮想デスクトップ小～中規模のデータベース開発環境や検証環境用 	<ul style="list-style-type: none">汎用SSDでは処理しきれない高い IO 性能を要求するアプリケーション1 ボリューム辺り16,000 IOPSや 250 MiB /sを超える性能を要するワークロード大規模なデータベース 
ボリュームサイズ	<ul style="list-style-type: none">1 GiBから16 TiBまで	<ul style="list-style-type: none">4 GiBから16 TiB まで
IOPS	<ul style="list-style-type: none"><u>1 GiB あたり3 IOPSのベースラインパフォーマンス</u>ベースラインパフォーマンスが 3,000 IOPS 以下の場合、3,000 IOPS までバーストが可能最小 100 IOPS (33.33 GiB 以下) から最大 16,000 IOPS (5,334 GiB 以上)	<ul style="list-style-type: none"><u>必要な IOPS 値を指定可能</u>容量(GiB)あたり50 IOPS を指定できる最小 100 IOPS最大 64,000 IOPS (Nitro ベースインスタンス)最大 32,000 IOPS (その他インスタンス)
スループット	<ul style="list-style-type: none">128 MiB /s (170 GiB まで)最大 250 MiB /s(170 GiB から 334 GiB)250 MiB /s (334 GiB 以上)	<ul style="list-style-type: none">最大 1,000 MiB /s (2000 IOPS 以上のときかつ Nitro ベースインスタンス)最大 500 MiB /s (その他のインスタンス)

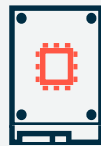
EBSのボリュームタイプ (HDDタイプ)

ボリュームタイプ	スループット最適化HDD (st1) - Throughput Optimized HDD	コールドHDD (sc1) - ColdHDD
ユースケース	<ul style="list-style-type: none">• EMR• データウェアハウス• 大規模なETL処理• 大規模なログ分析 ※起動ボリュームには利用できない 	<ul style="list-style-type: none">• ログデータ保管• バックアップ• アーカイブ ※起動ボリュームには利用できない 
ボリュームサイズ	• 500 GiB から16 TiB まで	• 500 GiB から16 TiB まで
IOPS	• 最大 500 IOPS	• 最大 250 IOPS
スループット	<ul style="list-style-type: none">• ベース値 : 1 TiB あたり 40 MiB /s• バースト値 : 1 TiB あたり 250 MiB /s• バーストクレジット上限 : 1TiB / 1TiB• 最大 500 MiB /s	<ul style="list-style-type: none">• ベース値 : 1 TiB あたり 12 MiB /s• バースト値 : 1 TiBあたり 80 MiB /s• バーストクレジット上限 : 1 TiB /1 TiB• 最大 250 MiB /s

本日のアジェンダ

- Amazon Elastic Block Store (EBS) 概要 ～アップデートを含めて～
 - EBSの概要
 - タイプ別の特徴
 - EC2インスタンスとパフォーマンスの考慮
 - 監視
 - NVMe SSD
- EBSの機能
 - Elastic Volume
 - EBS Snapshot
 - 暗号化
- EBSの価格
- まとめ

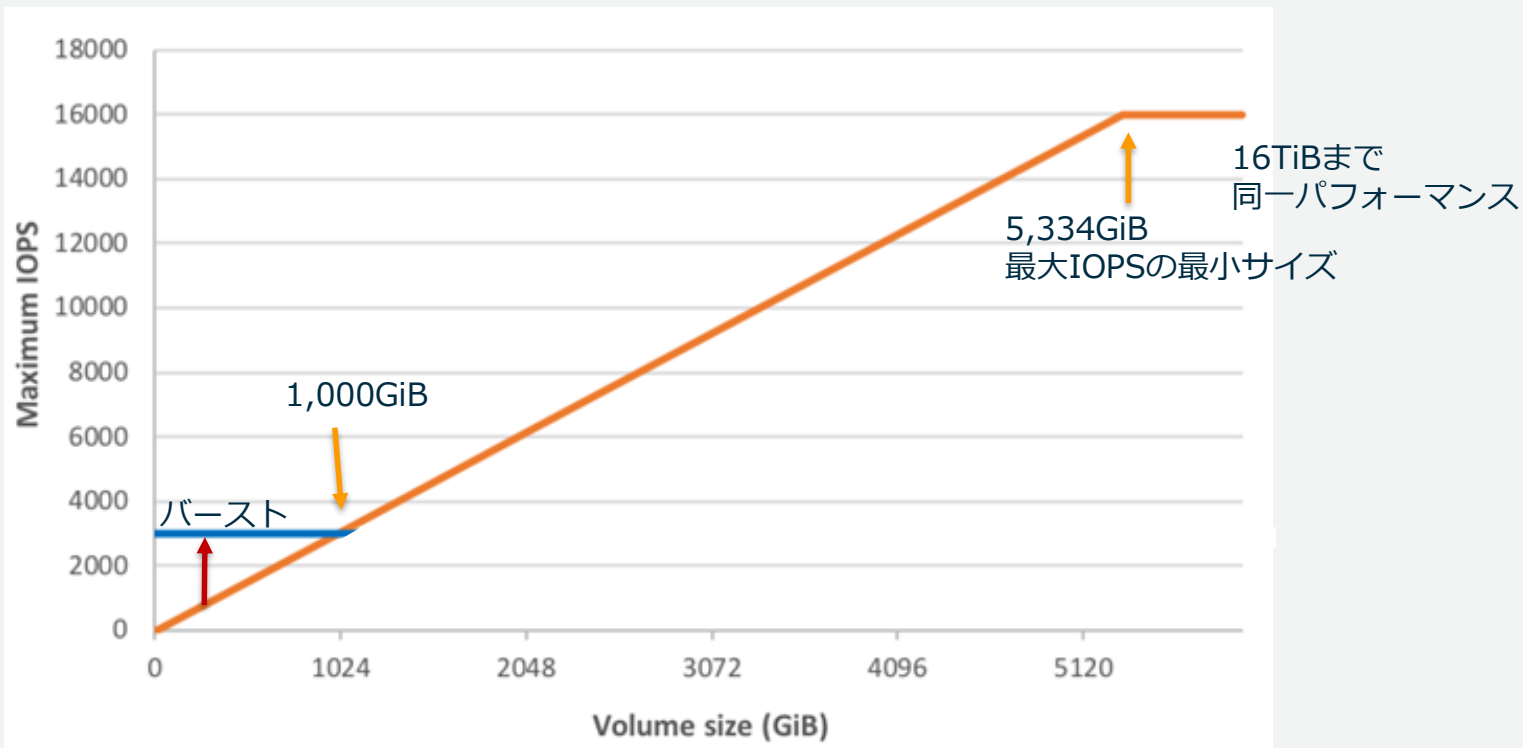
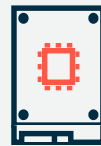
gp2の概要



- デフォルトのボリュームタイプ
- 最小 100 IOPS (33.33 GiB 以下) から最大 16,000 IOPS (5,334 GiB 以上)
 - 1,000 GiBまでは 3,000 IOPSまでバースト
- スループットは、1 IOPS 毎に256 KB /s
 - 128 MiB /s (170 GiBまで)
 - 最大 250 MiB /s (170 GiB から 334 GiB)
 - 250 MiB /s (334 GiB 以上)
- プロビジョニングされたパフォーマンスの90%を、利用時間の 99% で発揮するように設計

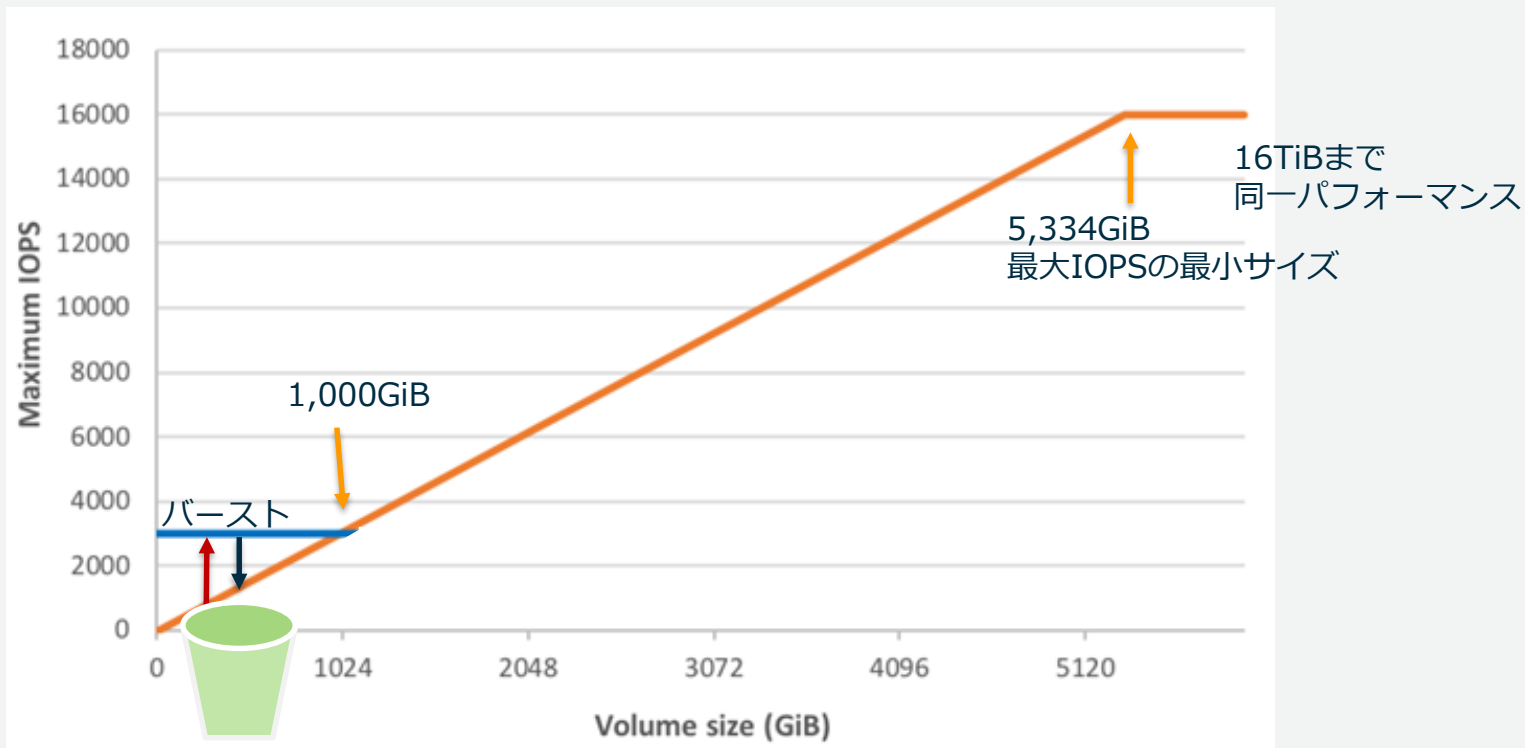
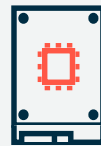
gp2での容量とIOPS

- ベースラインパフォーマンス
- バーストパフォーマンス

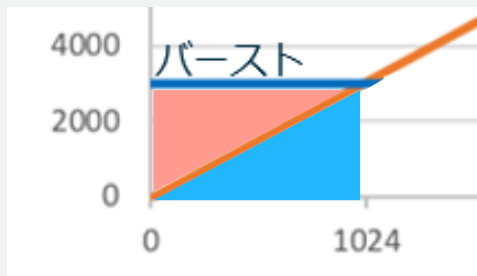
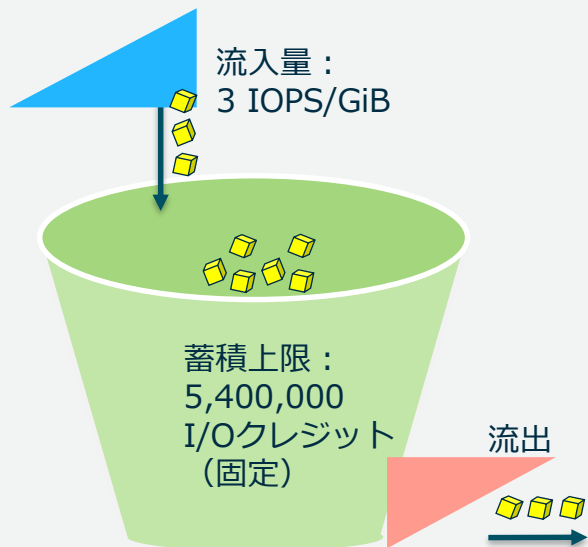
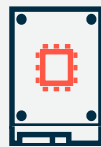


gp2での容量とIOPS

- ベースラインパフォーマンス
- バーストパフォーマンス



gp2のバーストバケットモデル



• 5,400,000 I/Oクレジットまで蓄積できるバーストバケットがボリュームごとに存在する

※ボリューム作成直後は満タン状態でスタート

- 最小のEBSストレージ容量でも30分間で 3,000 IOPSを維持するのに十分なクレジットを提供

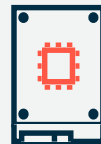
• バケットへの流入量(ベースラインパフォーマンス)は1 GiBあたり 3 IOPSとなる。流出量すなわち実 IOPS がこれを下回ると、バケットの残高が増えていく

• バケットにクレジットが残っていれば、流入量を超えて実 IOPS を 3,000 IOPSまで引き上げるバーストが利用できる

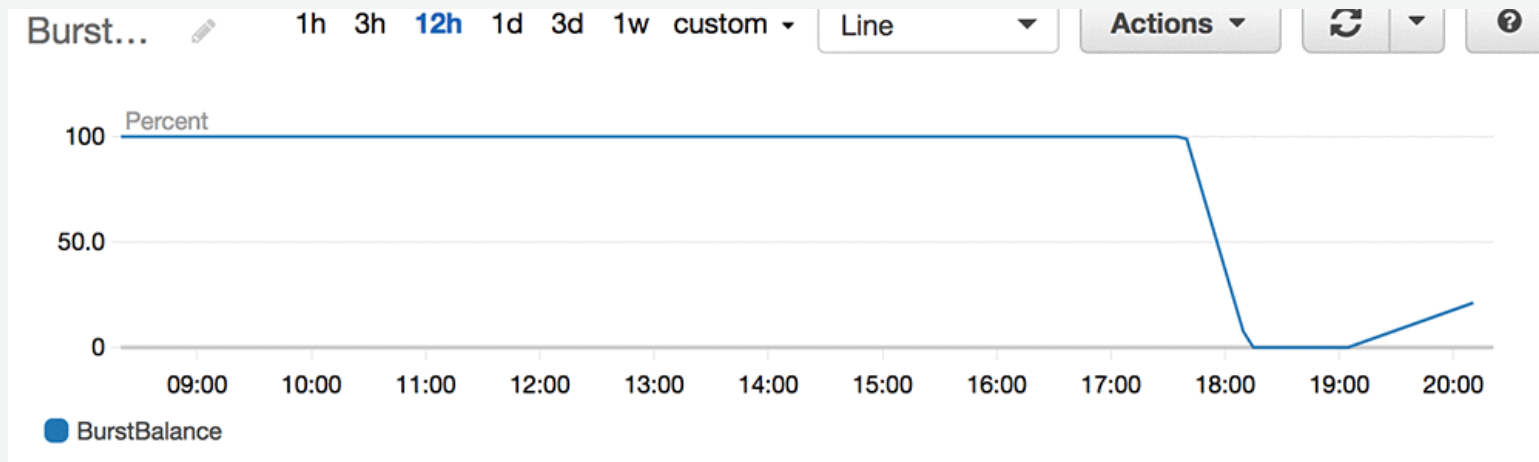
• バケットのクレジットが枯渇すると、新たに流入するクレジット分 (=ベースラインパフォーマンス分) のパフォーマンスのみが利用できることになる

- 空のクレジットバランスを補充する秒数は、100 GiB のボリュームの場合 18,000 秒、500 GiBのボリュームサイズだと 3,600 秒

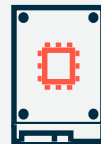
I/O クレジットの監視



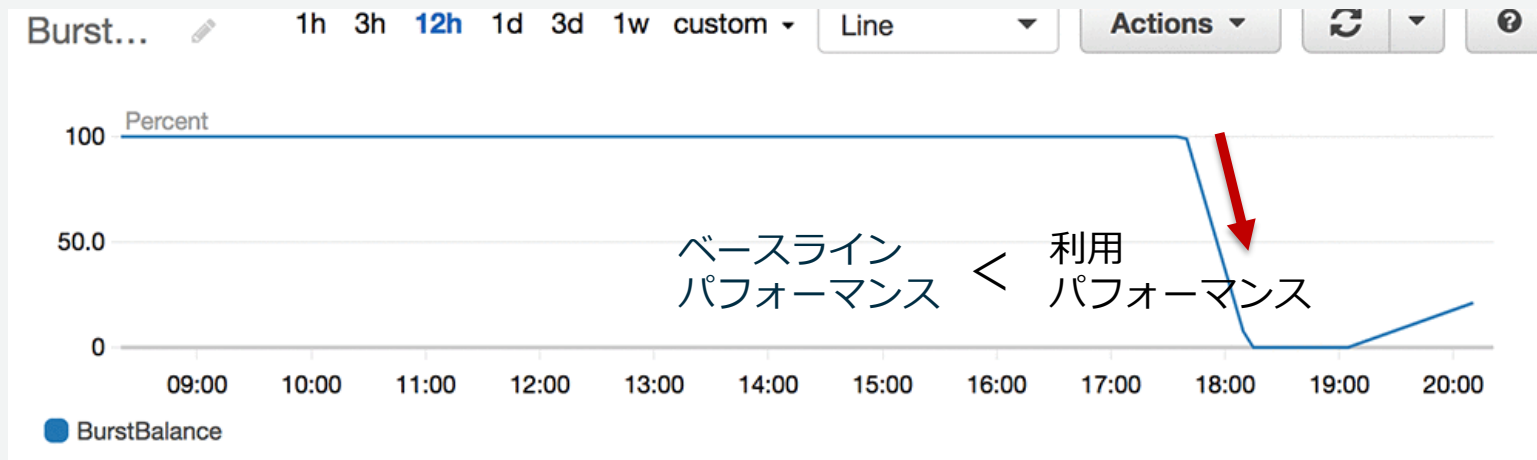
- Amazon CloudWatch (CloudWatch) のメトリクス BurstBalance は残っている I/O クレジット (gp2 用) またはスループットクレジット (st1 および sc1 用) の割合を提供
- gp2のバースト上限の 5,400,000 I/Oクレジットに対する実際の残クレジットの割合
- クレジットが頻繁に 0%になる場合は、gp2の容量追加、io1への変更を検討



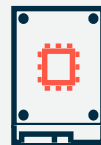
I/O クレジットの監視



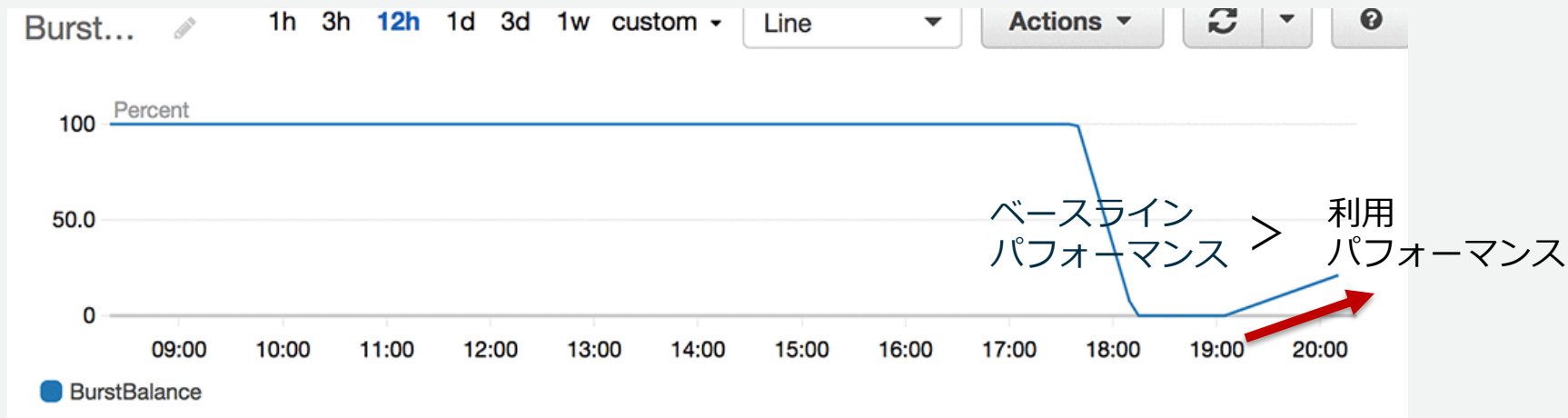
- Amazon CloudWatch (CloudWatch) のメトリクス BurstBalance は残っている I/O クレジット (gp2 用) またはスループットクレジット (st1 および sc1 用) の割合を提供
- gp2のバースト上限の 5,400,000 I/Oクレジットに対する実際の残クレジットの割合
- クレジットが頻繁に 0%になる場合は、gp2の容量追加、io1への変更を検討



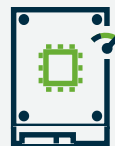
I/O クレジットの監視



- Amazon CloudWatch (CloudWatch) のメトリクス BurstBalance は残っている I/O クレジット (gp2 用) またはスループットクレジット (st1 および sc1 用) の割合を提供
- gp2のバースト上限の 5,400,000 I/Oクレジットに対する実際の残クレジットの割合
- クレジットが頻繁に 0%になる場合は、gp2の容量追加、io1への変更を検討



io1の概要



- 最小 100 IOPSから Nitro ベース インスタンスに対して最大 64,000 IOPS、他のインスタンス向けに対して最大 32,000 IOPSを提供

ボリュームタイプ	プロビジョンド IOPS SSD (io1)	?
サイズ(GiB)	1280	(Min: 4 GiB, Max: 16384 GiB)
IOPS	64000	(Min: 100 IOPS, Max: 64000 IOPS)

- スループットは、32,000 IOPSで 500 MB/s (1 IOPS 毎に256 KiB /s) 、 64,000 IOPS で最大 1,000 MB /s (1 IOPS毎に16 KiB /s) を提供
- 1年間のうち 99.9%の時間について、指定した IOPS 値の $\pm 10\%$ の範囲の性能を発揮する
- 容量と IOPS の最大割合 は 50:1。例えば、100 GiB のボリュームは最大 5,000 IOPS
 - 最適な レイテンシーを実現するためには、容量と IOPS の比率を 2:1 以上に、例えば、2,000 IOPS のボリュームは 1,000 GiB よりも小さくする

スループット最適化HDD – st1



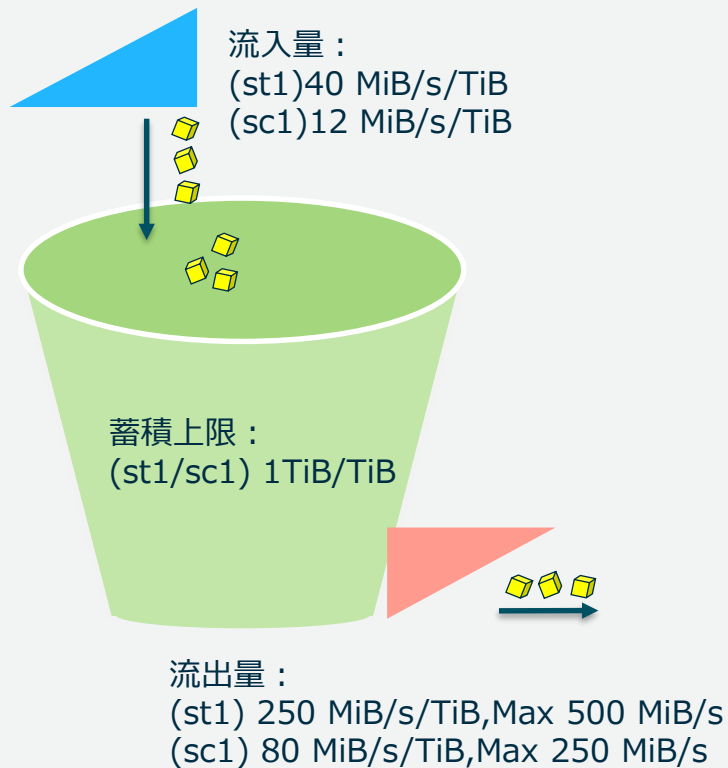
- シーケンシャルアクセス時に高い性能を発揮するタイプ。高いスループットを要求するビッグデータ処理に最適
- 仕様
 - 容量：
500 GiB から16 TiBまで、1 GiB単位で指定可能
 - IOPS：
最大 500 IOPS, 1 MiB の I/O サイズで読み取りと書き込みが処理
 - スループット：
容量 1 TiBあたり40 MiB/s がベースラインパフォーマンス
1 TiB あたり 250 MiB/s まで性能を引き上げるバーストが利用可能
スループットの上限值は 500MiB/s となる

コールドHDD – sc1



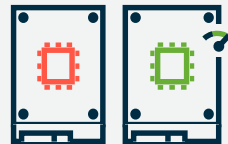
- st1 と同様のユースケースで高性能が不要な場合に。ログやバックアップのアーカイブ先としても利用可能
- 仕様
 - 容量 :
500 GiB から 16 TiB まで、1 GiB 単位で指定可能
 - IOPS :
最大 250 IOPS, 1 MiB の I/O サイズで読み取りと書き込みが処理
 - スループット :
容量 1 TiBあたり 12MiB/s がベースラインパフォーマンス
1 TiB あたり 80 MiB/s まで性能を引き上げるバーストが利用可能
スループットの上限値は 250 MiB/s となる

st1/sc1のバーストバケットモデル



- ボリューム容量 1 TiBあたりにつき、1 TiB まで蓄積できるバーストバケットが存在する
- 流入量(ベースラインパフォーマンス)はボリュームタイプによって異なる
 - ✓ st1: 1 TiB あたり 40 MiB/s、最大 500 MiB/s
 - ✓ sc1: 1 TiB あたり 12 MiB/s、最大 192 MiB/s
- 流出量(バースト時性能)もボリュームタイプに依存
 - st1: 1 TiB あたり 250 MiB/s、最大 500 MiB/s
 - sc1: 1 TiB あたり 80 MiB/s、最大 250 MiB/s
- st1で12,800 GiB (12.5 TiB)以上確保すると常時 500MB /s となりバーストの概念がなくなる
- Snapshot 作成中はバーストが発生せずベースラインパフォーマンスとなる

gp2 と io1 の IOPS カウント



- 256 KiBまでの連続したアクセスを 1 IOPS とカウントする
 - 例① : 32 KiBの連続するアクセス8回は、 I/O命令を1回発行
 - 例② : 32 KiBのランダムなアクセス8回は、 I/O命令を8回発行
- 256 KiBを超える場合は複数回の256 KiBブロックアクセスを行ったものとしてカウントされる
 - 例① : 32 KiBアクセスの1回は、 I/O命令を1回発行
 - 例② : 512 KiBアクセスを1回行くと、 I/O命令を2回発行
 - 大きなブロックサイズのアクセスを行うと、低い IOPS 値でもスループットを稼げるが、EBS ボリューム や EC2 側の最大スループットに注意

本日のアジェンダ

- Amazon Elastic Block Store (EBS) 概要 ～アップデートを含めて～
 - EBSの概要
 - タイプ別の特徴
 - EC2インスタンスとパフォーマンスの考慮
 - 監視
 - NVMe SSD
- EBSの機能
 - Elastic Volume
 - EBS Snapshot
 - 暗号化
- EBSの価格
- まとめ

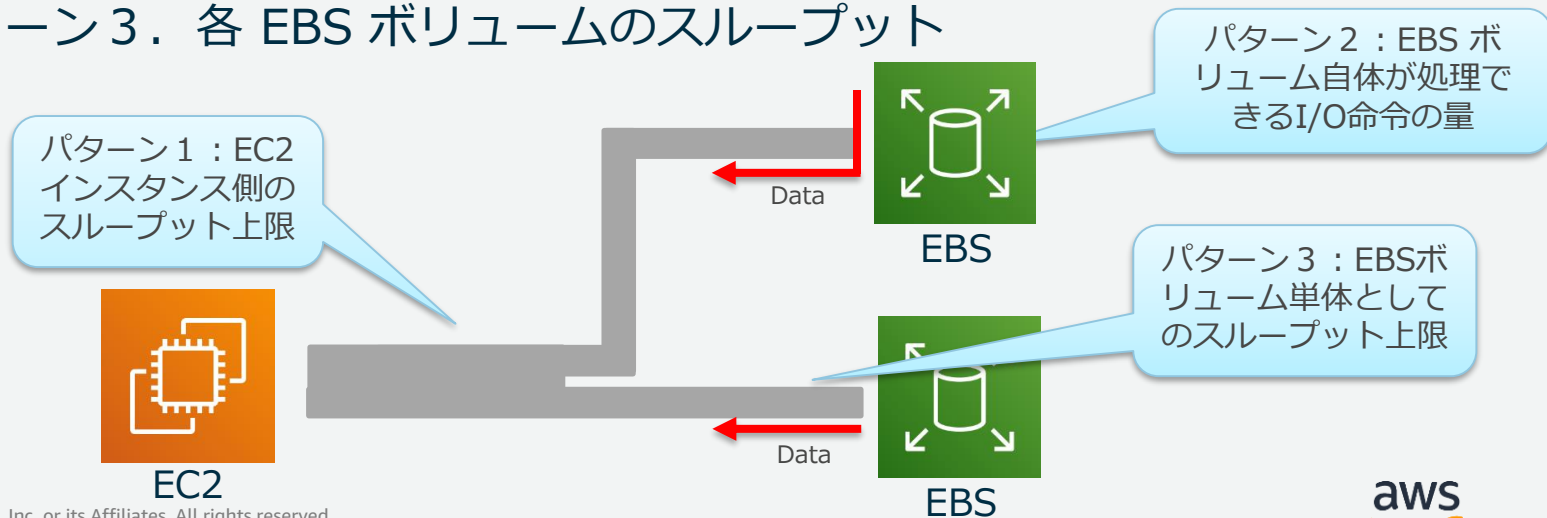
EBSのパフォーマンスを律速する要素

- EBS のパフォーマンスは3つの要素で律速されるので、システム全体としてのボトルネックを把握することが重要

パターン1． EC2 インスタンス側のスループット

パターン2． EBS ボリュームが処理できるI/O命令の回数 (IOPS)

パターン3． 各 EBS ボリュームのスループット



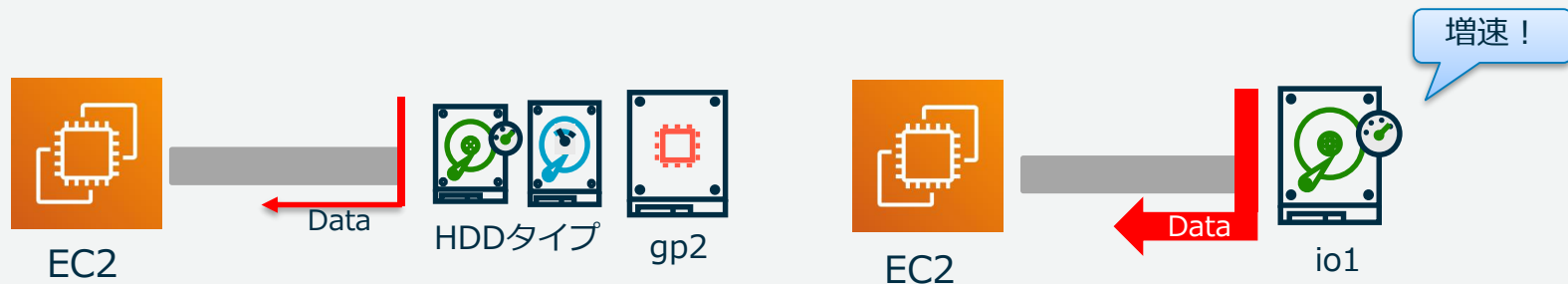
パターン1. EC2インスタンス側のスループットを改善する

- 最新ではないインスタンスタイプの場合は、EBS最適化 (EBS-Optimized) を有効にする
- EC2 インスタンスタイプによって決まる EBS スループットの上限值に到達していないかを確認する
 - CloudWatch のVolume Read / Write Bytesの合計値
 - OS で EBS ボリュームへの総流量を確認 (iostat や perfmon など)
- 上限に到達している場合はインスタンスタイプを大きくすることでスループットを改善する



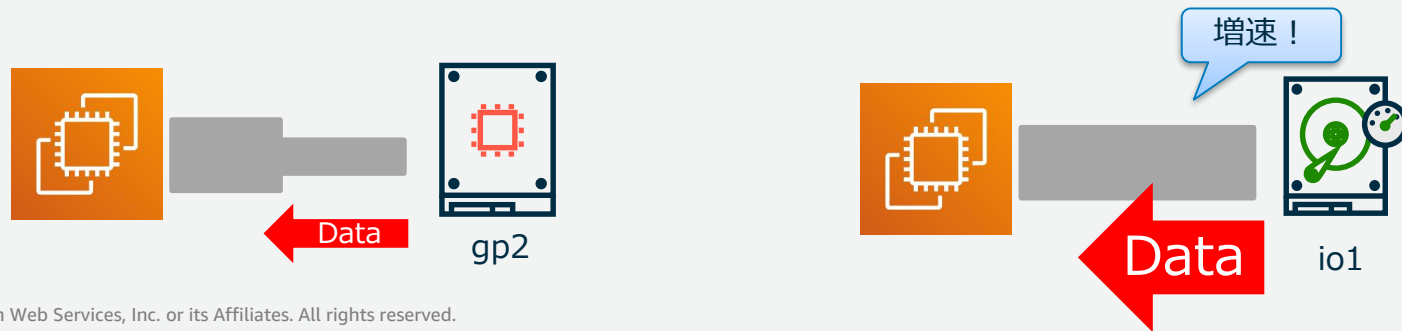
パターン2. EBS ボリューム側のI/O処理性能を改善する

- EBS ボリューム側の実績 IOPS を確認する
 - CloudWatch のVolume Read / Write Opsの合計値
 - OS で EBS ボリュームへの I/O 命令回数を確認（iostat や perfmon など）
- 上限に到達していればボリュームの変更を検討
 - タイプを変更（HDDタイプ→gp2, gp2→io1）
 - スペックを変更（gp2：容量を増加, io1：IOPS 値を増加）



パターン3. EBS側のスループットを改善する

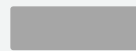
- 個々の EBS ボリュームのスループットを確認する
 - CloudWatch のVolume Read / Write Bytesの合計値
 - OS で EBS ボリュームへの総流量を確認（ iostat や perfmon など）
- 上限に到達していればボリュームの変更を検討
 - タイプを変更（gp2→io1, gp2→st1）
 - gp2からst1に変更する場合はアクセスパターンに注意



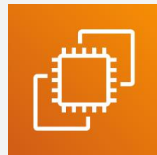
EC2とEBS間のボトルネック

IOPS数 1 I/Oサイズ
スループット MB/ s

帯域



m4.xlarge



6,000 IOPS 16KB I/O
93.75 MB /s

gp2 500 GiB



1,500 IOPS 256 KiB I/O
250 MiB /s

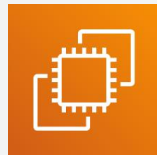
EC2とEBS間のボトルネック

IOPS数 1 I/Oサイズ
スループット MB/ s

帯域



m4.xlarge



6,000 IOPS 16KB I/O
93.75 MB /s

gp2 500 GiB



1,500 IOPS 256 KiB I/O
250 MiB /s

256KBのランダムIO

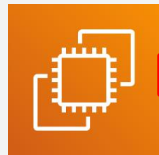
EC2とEBS間のボトルネック

IOPS数 1 I/Oサイズ
スループット MB/ s

帯域



m4.xlarge



gp2 500 GiB



6,000 IOPS 16KB I/O

93.75 MB /s

1,500 IOPS 256 KiB I/O

250 MiB /s

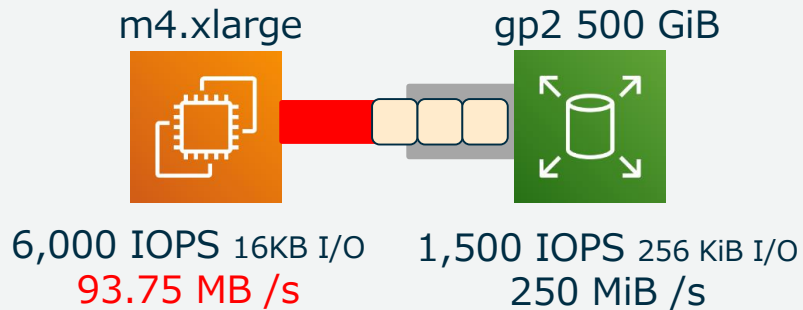
256KBのランダムIO

パターン1 : EC2
インスタンス側の
スループット上限

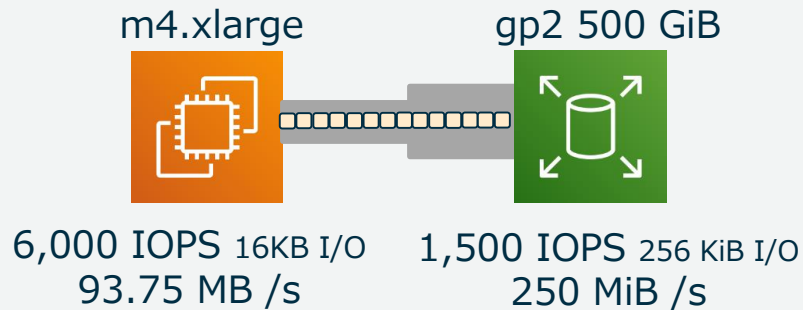
EC2とEBS間のボトルネック

IOPS数 1 I/Oサイズ
スループット MB/ s

帯域



256KBのランダムIO



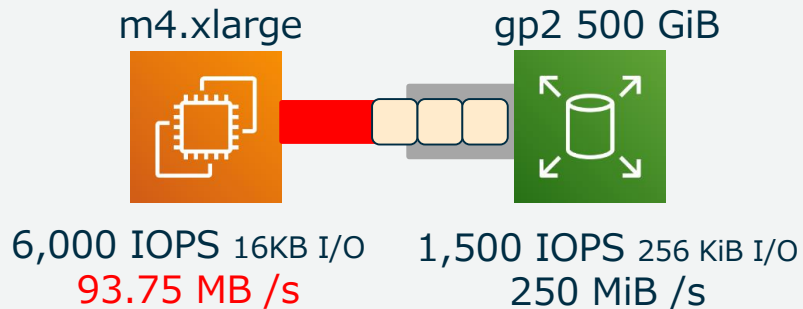
16KBのランダムIO

EC2とEBS間のボトルネック

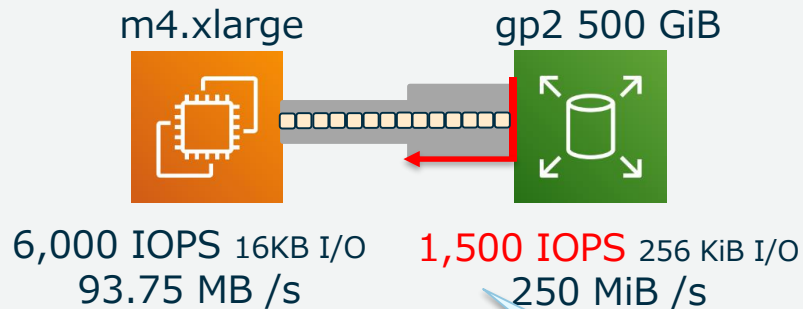
IOPS数 1 I/Oサイズ
スループット MB/ s

帯域

バースト

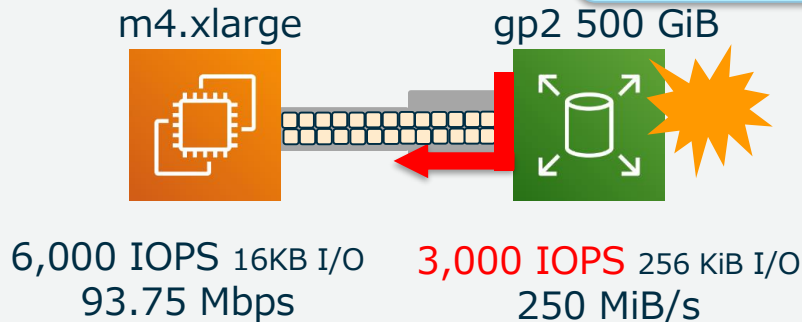


256KBのランダムIO



16KBのランダムIO

パターン2 : EBS ボリューム自体が処理できるI/O命令の量



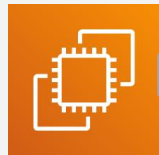
EC2とEBS間のボトルネック

IOPS数 1 I/Oサイズ
スループット MB/ s

帯域



m5.xlarge



6,000 IOPS 16KB I/O
106.25 MB /s

gp2 500 GiB



1,500 IOPS 256 KiB I/O
250 MiB /s

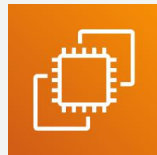
EC2とEBS間のボトルネック

IOPS数 1 I/Oサイズ
スループット MB/ s

帯域
バースト



m5.xlarge



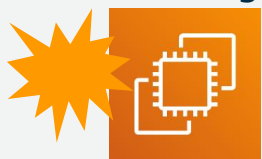
6,000 IOPS 16KB I/O
106.25 MB /s

gp2 500 GiB



1,500 IOPS 256 KiB I/O
250 MiB /s

m5.xlarge



18,750 IOPS 16KB I/O
437.5 MB/ s

m5.large
m5.xlarge
m5.2xlarge
m5.4xlarge
.
.

30分/24時間

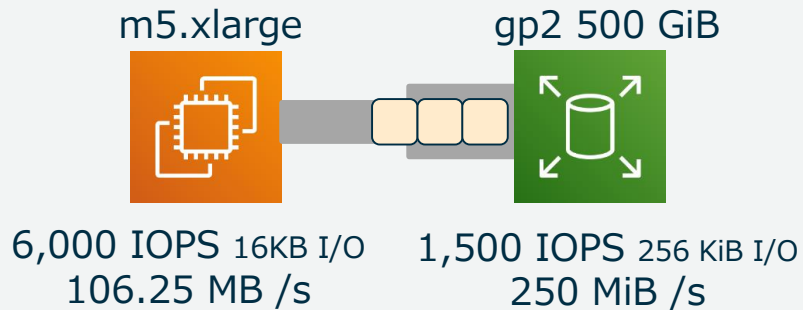


ベースライン
パフォーマンス

EC2とEBS間のボトルネック

IOPS数 1 I/Oサイズ
スループット MB/ s

帯域
バースト



256KBのランダムIO

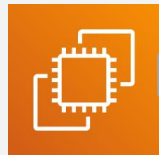
EC2とEBS間のボトルネック

IOPS数 1 I/Oサイズ
スループット MB/ s

帯域
バースト



m5.xlarge



gp2 500 GiB



6,000 IOPS 16KB I/O
106.25 MB /s

1,500 IOPS 256 KiB I/O
250 MiB /s

256KBのランダムIO

30分/24時間



m5.xlarge



gp2 500 GiB



18,750 IOPS 16KB I/O
437.5 MB/ s

3,000 IOPS 256 KiB I/O
250 MiB/s

パターン3 : EBSボ
リューム単体として
のスループット上限

EC2とEBS間のボトルネック

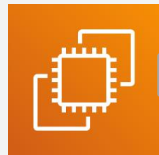
IOPS数 1 I/Oサイズ
スループット MB/ s

帯域

バースト



m5.xlarge



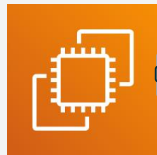
gp2 500 GiB



6,000 IOPS 16KB I/O
106.25 MB /s

1,500 IOPS 256 KiB I/O
250 MiB /s

m5.xlarge



gp2 500 GiB



6,000 IOPS 16KB I/O
106.25 MB /s

1,500 IOPS 256 KiB I/O
250 MiB /s

256KBのランダムIO

30分/24時間



m5.xlarge



gp2 500 GiB



18,750 IOPS 16KB I/O
437.5 MB/ s

3,000 IOPS 256 KiB I/O
250 MiB/s

16KBのランダムIO

EC2とEBS間のボトルネック

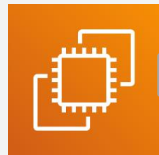
IOPS数 1 I/Oサイズ
スループット MB/ s

帯域

バースト



m5.xlarge



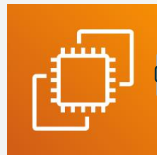
gp2 500 GiB



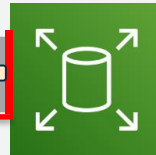
6,000 IOPS 16KB I/O
106.25 MB / s

1,500 IOPS 256 KiB I/O
250 MiB / s

m5.xlarge



gp2 500 GiB



6,000 IOPS 16KB I/O
106.25 MB / s

1,500 IOPS 256 KiB I/O
250 MiB / s

256KBのランダムIO

30分/24時間



m5.xlarge



gp2 500 GiB



18,750 IOPS 16KB I/O
437.5 MB/ s

3,000 IOPS 256 KiB I/O
250 MiB/s

16KBのランダムIO

30分/24時間



m5.xlarge



gp2 500 GiB



18,750 IOPS 16KB I/O
437.5 MB/ s

3,000 IOPS 256 KiB I/O
250 MiB/s

パターン2 : EBS ボリューム自体が処理できるI/O命令の量

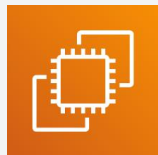
最大性能のEC2とEBS

帯域



m5.24xlarge

io1 16,000 GiB



EC2



EBS

80,000 IOPS	16KB I/O	64,000 IOPS	16 KiB
1,750 MB /s		1,000 MB /s	

16KBのランダムIO

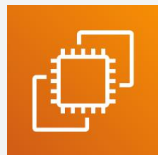
最大性能のEC2とEBS

帯域



m5.24xlarge

io1 16,000 GiB



EC2



EBS

80,000 IOPS	16KB I/O	64,000 IOPS	16 KiB
1,750 MB /s		1,000 MB /s	

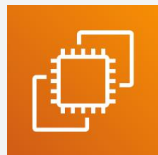
16KBのランダムIO

最大性能のEC2とEBS

帯域



m5.24xlarge



EC2

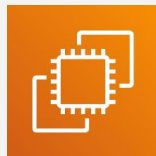
io1 16,000 GiB



EBS

80,000 IOPS 16KB I/O 64,000 IOPS 16 KiB
1,750 MB /s 1,000 MB /s

m5.24xlarge



EC2

80,000 IOPS 16KB I/O
1,750 MB/ s

io1 16,000 GiB



EBS

64,000 IOPS 16 KiB
1,000 MB /s

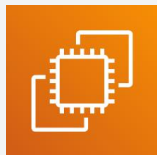
本日のアジェンダ

- Amazon Elastic Block Store (EBS) 概要 ～アップデートを含めて～
 - EBSの概要
 - タイプ別の特徴
 - EC2インスタンスとパフォーマンスの考慮
 - 監視
 - NVMe SSD
- EBSの機能
 - Elastic Volume
 - EBS Snapshot
 - 暗号化
- EBSの価格
- まとめ

監視の考え方

OS

- CPU、メモリ
- ネットワーク
- 性能(スループット,IOPS)
- 容量



EC2



EBS

CloudWatch



- CPU
- ネットワーク
- EBS (Nitroハイパーバイザー)
- 性能 (スループット,IOPS)
- 容量
- バーストクレジット

EBS の監視


	性能	容量	バーストクレジット
方法	CloudWatch 標準メトリクス	CloudWatch カスタムメトリクス	CloudWatch 標準メトリクス
メトリクス	Volume Read / Write Bytes、 Volume Read / Write Ops、 VolumeConsumedReadWriteOps (io1 のみ)	ディスクの使用量、空き容量	BurstBalance
間隔	io1は 1 分間、gp2、st1、sc1は 5 分間のメトリクスを CloudWatch へ送信		

本日のアジェンダ

- Amazon Elastic Block Store (EBS) 概要 ～アップデートを含めて～
 - EBSの概要
 - タイプ別の特徴
 - EC2インスタンスとパフォーマンスの考慮
 - 監視
 - NVMe SSD
- EBSの機能
 - Elastic Volume
 - EBS Snapshot
 - 暗号化
- EBSの価格
- まとめ

NVMe SSD



- Nitro ベースインスタンスでは、NVMe ブロックデバイスとして EBS ボリュームを認識
- OS の NVMe ドライバを利用して PCI バスをスキャンして、アタッチされた EBS を検出
- 旧世代のインスタンスやハイパーバイザーからの移行 
 - 旧世代ハイパーバイザーから Nitro ハイパーバイザーに変更する場合は、NVMe ドライバが必要
 - OS 自体が NVMe デバイスに対応していること確認
 - OS のアップグレードが難しいなど、条件を満たすことができない場合は c4 / m4 / r4 などの旧世代のインスタンスのまま利用することも検討

NVMe SSD



- OSからNVMeデバイスに送信される I/O 操作のタイムアウト値を最大値 (4294967295)にすることを推奨



```
[ec2-user@ ~]$ sudo cat /sys/module/nvme_core/parameters/io_timeout  
4294967295
```

- NVMe デバイス名 (/dev/nvme[0-26]n1)を利用
 - EC2の起動ごとにデバイス名が変更される可能性があるので、デバイスファイル名ではなくUUIDで /etc/fstab を指定



- xfsの場合 :

```
[ec2-user@ ~]$ cat /etc/fstab  
#  
UUID=f5bd1ae0-8 d8deb328c92 / xfs defaults,noatime 1 1  
UUID=2a0efcd5-6 e6467074410 /data xfs defaults,nofail 0 2
```

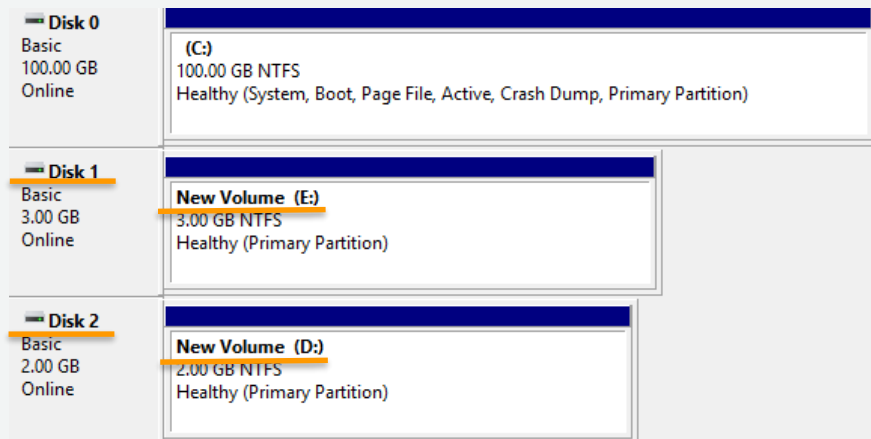
- UUIDではなくLABELで指定することも可能

https://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/ebs-using-volumes.html#ebs-mount-after-reboot

NVMe SSD




- WindowsのNVMe デバイス名
 - OSが認識する Disk Number (Diskpart コマンドで指定する番号) が変更される可能性はあるが、OS側でドライブレターは変更されない

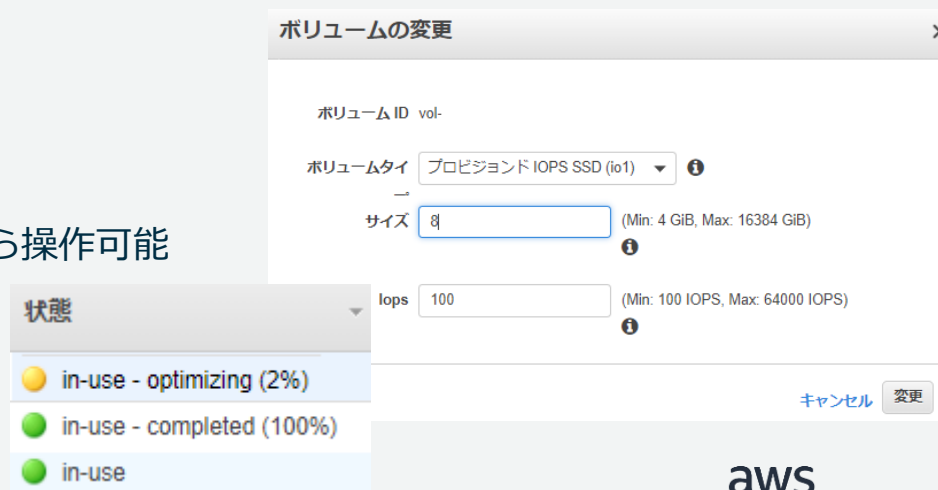


本日のアジェンダ

- Amazon Elastic Block Store (EBS) 概要 ～アップデートを含めて～
 - EBSの概要
 - タイプ別の特徴
 - EC2インスタンスとパフォーマンスの考慮
 - 監視
 - NVMe SSD
- EBSの機能
 - Elastic Volume
 - EBS Snapshot
 - 暗号化
- EBSの価格
- まとめ

Elastic Volume

- EBS ボリュームを EC2 インスタンスにアタッチ中でもサイズや IOPSを変更可能
 - OSから見て空き容量がなくなってしまった場合でもサイズ変更（拡張）可能
 - サイズだけでなく、EBS ボリュームタイプの変更が可能（gp2 → io1 など）
 - gp2からst1、sc1への変更の際には、500 GiB以下ではないこと、またルートボリュームではないでないことを確認 
 - 縮小することはできない
 - io1 は、サイズとIOPSの両方が変更可能
 - API、CLI、マネージメントコンソールから操作可能
 - 変更の処理は、modifying、optimizing、completedと遷移



ボリュームの変更

ボリューム ID vol-

ボリュームタイプ プロビジョンド IOPS SSD (io1) ⓘ

サイズ 8 (Min: 4 GiB, Max: 16384 GiB) ⓘ



IOPS 100 (Min: 100 IOPS, Max: 64000 IOPS) ⓘ

状態

- in-use - optimizing (2%)
- in-use - completed (100%)
- in-use

キャンセル 変更

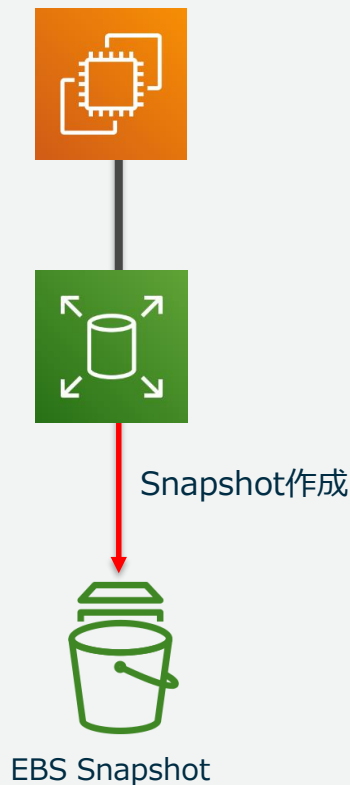
Elastic Volume

- 容量拡張後は、OS側でファイルシステムの拡張を実施
- IOPS の設定は、徐々に反映される 
 - 1 TiB ボリュームが新しい設定になるまで6時間程度必要
 - 反映中（optimizing 状態）は、設定前の IOPS と設定後の IOPS の間の IOPSを提供
- 1度変更すると6時間は変更不可 
- 変更自体は無料。変更後のボリューム設定に応じて、optimizing 状態から課金

本日のアジェンダ

- Amazon Elastic Block Store (EBS) 概要 ～アップデートを含めて～
 - EBSの概要
 - タイプ別の特徴
 - EC2インスタンスとパフォーマンスの考慮
 - 監視
 - NVMe SSD
- EBSの機能
 - Elastic Volume
 - EBS Snapshot
 - 暗号化
- EBSの価格
- まとめ

EBSのSnapshot



- 定期的にEBSの Snapshot を作成することによりバックアップを取得する
- Snapshot 作成時はデータ整合性を保つため静止点を設ける事を推奨
 - ソフトウェアの機能 (例 : RDBMSのバックアップモード)
 - ファイルシステムの機能 (例 : Linuxのxfs_freeze)
 - バックアップソフトウェアの機能
 - (アプリケーションの停止)
 - (ファイルシステムのアンマウント)
- 保存期間や世代数は無制限。世代管理が必要な場合は、AWS CLIやAPI等で自動化したり、他のサービスを利用

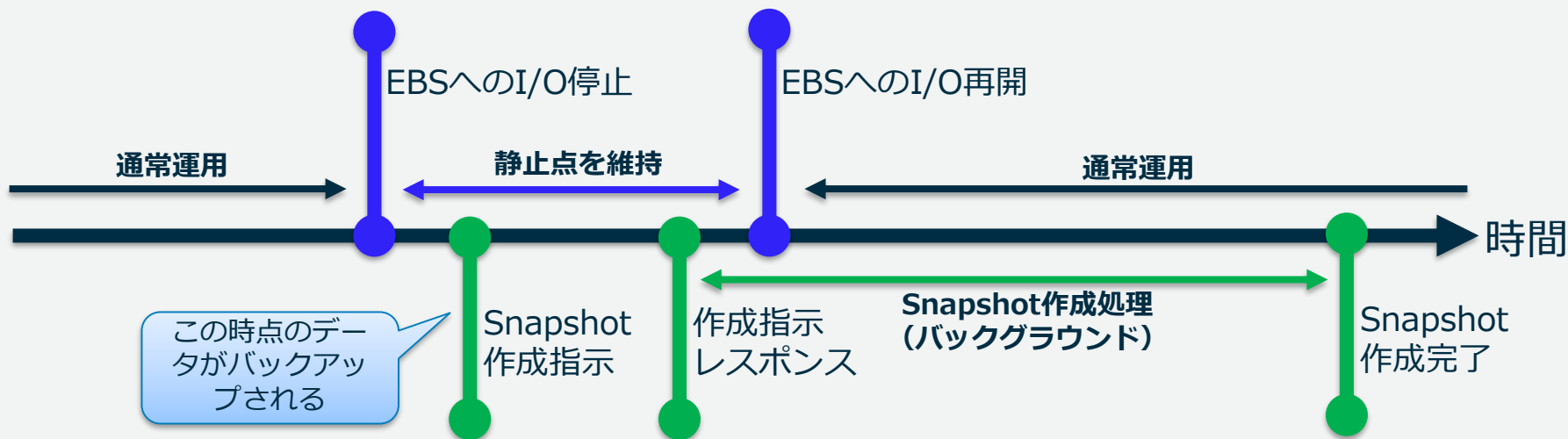
バックアップと静止点

- Snapshotの作成を指示しレスポンスが返ってきたら、その時点のデータのバックアップが開始されている。



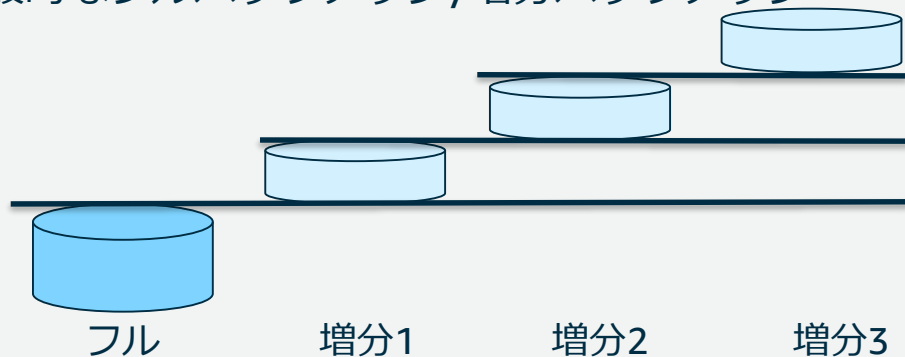
バックアップと静止点

- Snapshotの作成を指示しレスポンスが返ってきたら、その時点のデータのバックアップが開始されている。
- レスポンスが返ってきた時点でI/Oを再開して良いので、静止点を維持するのは短時間で済む



EBS Snapshot の世代管理の仕組み

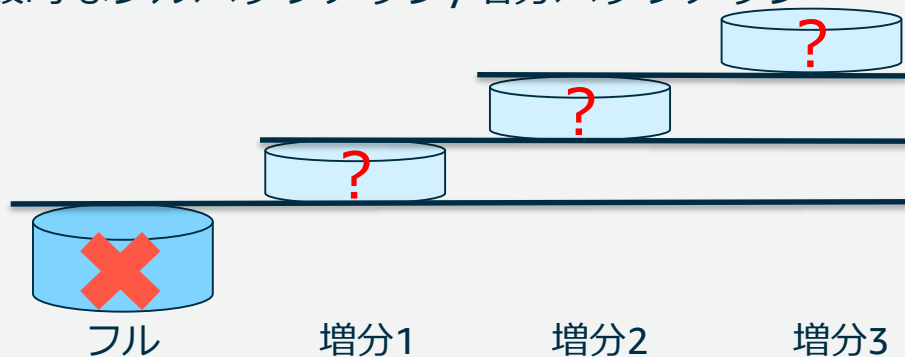
一般的なフルバックアップ/増分バックアップ



- フルバックアップ後、増分バックアップを取得

EBS Snapshot の世代管理の仕組み

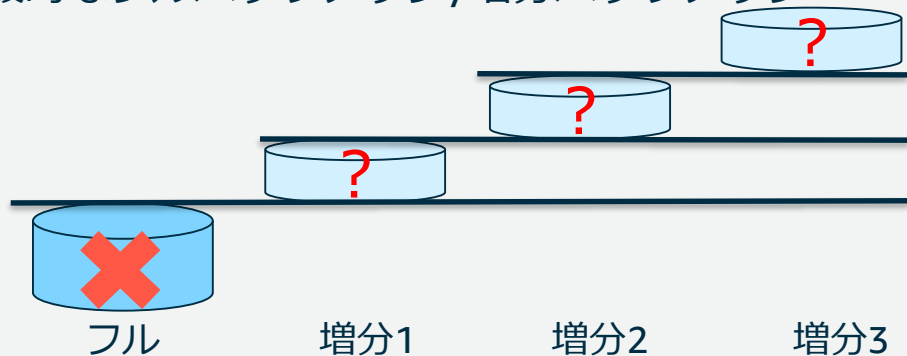
一般的なフルバックアップ/増分バックアップ



- フルバックアップ後、増分バックアップを取得
- フルバックアップが削除されると増分バックアップは意味をなさない

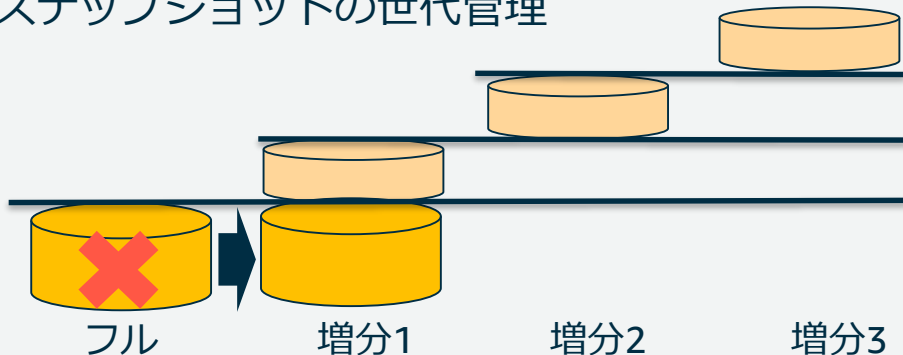
EBS Snapshot の世代管理の仕組み

一般的なフルバックアップ/増分バックアップ



- フルバックアップ後、増分バックアップを取得
- フルバックアップが削除されると増分バックアップは意味をなさない

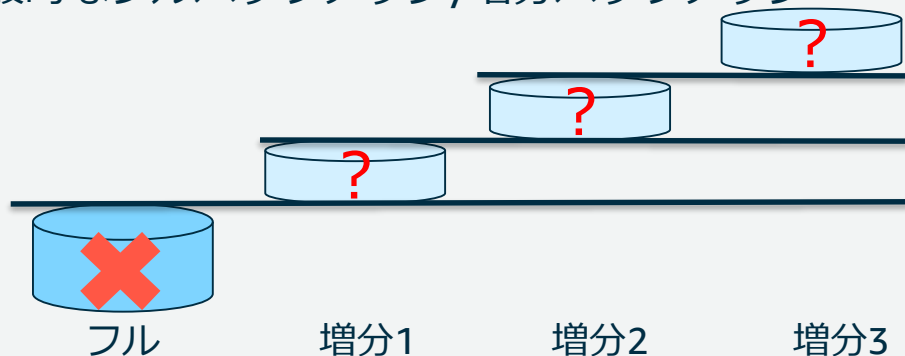
EBSスナップショットの世代管理



- フルバックアップを削除しても、増分1のバックアップがフルバックアップの情報を持続

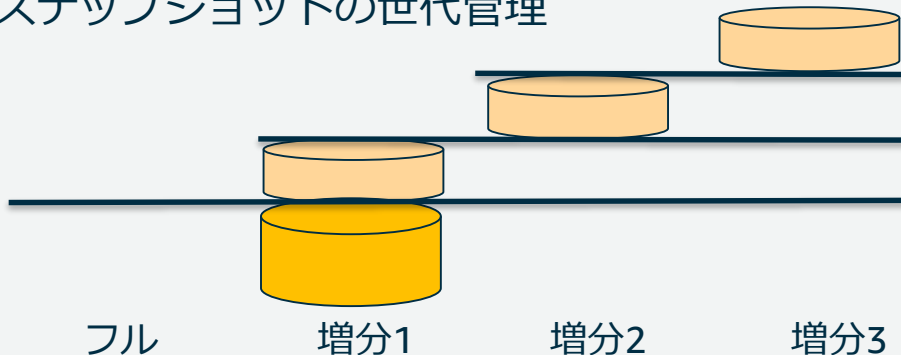
EBS Snapshot の世代管理の仕組み

一般的なフルバックアップ/増分バックアップ



- フルバックアップ後、増分バックアップを取得
- フルバックアップが削除されると増分バックアップは意味をなさない

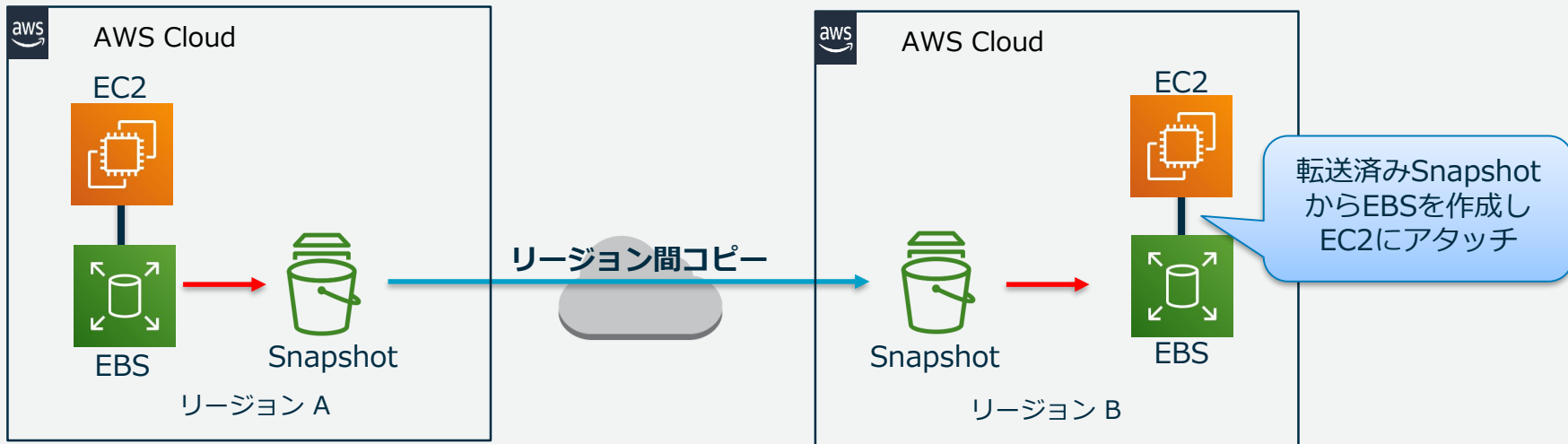
EBSスナップショットの世代管理



- フルバックアップを削除しても、増分1のバックアップがフルバックアップの情報を保持
- フルバックアップ削除後、増分2は増分1を参照

リージョン間コピー

- リージョン間でのSnapshotコピーをサポート
- コピーを指示しておけば非同期で処理が行われるため、バックアップデータを他リージョンに転送しておけばDRを実現できる



リージョン間コピー



- Snapshot 取得の完了をCloudWatch Events と連携することでリージョン間コピーを自動化

– 取得した Snapshot を Lambda 関数で別リージョンにコピー

イベントソース

イベントパターンを構築またはカスタマイズするか、スケジュールを設定してターゲットを呼び出します。

☒ イベントパターン ⓘ ☐ スケジュール ⓘ

サービス別のイベントに一致するイベントパターンの構築

サービス名 EC2

イベントタイプ EBS Snapshot Notification

☐ Any event ☒ Specific event(s)

× createSnapshot

☐ Any result ☒ Specific result(s)

× succeeded

ターゲット

イベントがイベントパターンに一致するか、スケジュールがトリガーされたときに呼び出すターゲットを選択します。

Lambda 関数

機能*

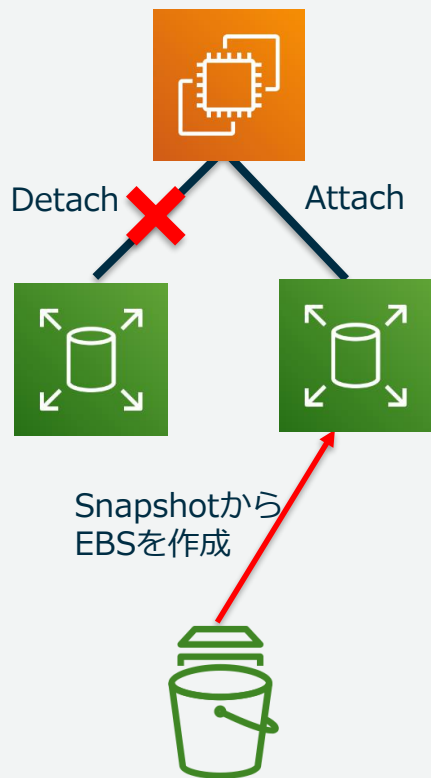
関数の選択

▶ バージョン/エイリアスの設定

▶ 入力の設定

+ ターゲットの追加*

Snapshotからのリストア



- Snapshotから新規EBSを作成し、EC2インスタンスにアタッチされていたものと置き換える
- 古いEBSは不要であれば削除する。障害分析等の目的で他のインスタンスにアタッチしてもOK
- EBSを別AZに移動したい場合は、Snapshot経由で行う

EBSのSnapshotの作成方法

New!

New!

New!

手法	CLI/SDK/ マネージメントコン ソール	Systems Manager / CloudWatch Events	Amazon Data Lifecycle Manager (Amazon DLM)	AWS Backup
概要	CreateSnapshot CLI やマネージドコンソール を利用してSnapshot を取得	Windows環境とLinux環境 が混在の場合に有用	タグ付けしたEBSを定期的 にSnapshot取得	EBSだけでなくEFS、RDS、 DynamoDB、Storage Gatewayをサポート、東 京リージョン未サポート
自動化	CLI/SDKを別途呼び出 すことで実施	実行日時はCronで指定、 Snapshotの削除は別途必 要	2,3,4,6,8,12,24時の間隔 で、最大1000世代までの 保持可能	世代管理を含めて自動化可 能
利用価 格	無料 ストレージの価格とし て、\$0.05/GB/月	カスタムイベント 100 万 件あたり 1.00 USD ストレージの価格として、 \$0.05/GB/月	無料 ストレージの価格として、 \$0.05/GB/月	ストレージの価格として、 \$0.05/GB/月
利用 シーン	既存の運用方法、スク リプトが存在する場合	Linux環境だけでなく Windowsのワークロード が含まれている場合	EBS スナップショットの 管理、世代管理を自動化 したい場合	EBSだけでなく他のサービ スも含めて一元的に管理し たい場合

Systems ManagerのRun Commandを利用したSnapshot



- Windows 環境の EC2 インスタンスでVolume Shadow Copy Service(VSS)と連携すると一貫性のあるSnapshotが取得できる環境で有用
 - VSSと連携することで、SQLServerなどのマイクロソフト製品に対して、アプリケーションの一貫性を保ったEBSボリュームのSnapshotが取得可能
- WindowsインスタンスのEC2にアタッチされたすべてのEBSが対象
 - ブートボリュームを除外することは可能
- Systems Manager (SSM)のRun Commandで使えるSSMドキュメントの1つであるAWSEC2-CreateVssSnapshot からWindowsインスタンスとVolume Shadow Copy Service(VSS)を使用して、VSS対応アプリケーションのイメージレベルバックアップを取得

https://docs.aws.amazon.com/ja_jp/systems-manager/latest/userguide/integration-vss.html

Systems ManagerのRun Commandを利用したSnapshot



- 定期的なSnapshotの取得はCloudWatch Eventsのルールを利用
 - マネージドコンソールから設定する場合、スケジュールにCron式、ターゲットにSSM Run Commandを設定

イベントソース

イベントパターンを構築またはカスタマイズするか、スケジュールを設定してターゲットを呼び出します。

☐ イベントパターン ⓘ ☒ スケジュール ⓘ

☐ 一定の速度 5 分

☒ Cron 式 0 17 * * * *

以後 10 回のトリガー日

1. Mon, 11 Mar 2019 17:00:00 GMT
2. Tue, 12 Mar 2019 17:00:00 GMT
3. Wed, 13 Mar 2019 17:00:00 GMT
4. Thu, 14 Mar 2019 17:00:00 GMT
5. Fri, 15 Mar 2019 17:00:00 GMT
6. Sat, 16 Mar 2019 17:00:00 GMT
7. Sun, 17 Mar 2019 17:00:00 GMT
8. Mon, 18 Mar 2019 17:00:00 GMT
9. Tue, 19 Mar 2019 17:00:00 GMT
10. Wed, 20 Mar 2019 17:00:00 GMT

CloudWatch イベントスケジュールの詳細については、[こちらをご覧ください](#)。

▶ サンプルイベントの表示

ターゲット

イベントがイベントパターンに一致するか、スケジュールがトリガーされたときに呼び出すターゲットを選択します。

SSM Run Command ⓘ

ドキュメント* AWSEC2-CreateVssSnapshot (Windows) ⓘ

ターゲットキー* InstanceIds ⓘ

ターゲット値* ⓘ

Run Command ターゲットは、SSM Run Command を呼び出す先の EC2 インスタンスを指定します。 [詳細はこちら](#)

▼ パラメータの設定

☒ パラメータなし ⓘ

☐ 定数 ⓘ

☐ インプットトランスフォーマー ⓘ

CloudWatch Events は、EC2 インスタンスで EC2 Run Command を呼び出すためのアクセス許可が必要です。続行すると、許可したことになります。

☒ この特定のリソースに対して新しいルールを作成する

AWS_Events_Invoke_Run_Command_1482248579 ⓘ

☐ 既存のロールの使用 ⓘ

CloudWatch Events のアイデンティティベースのポリシーの詳細については、[こちらを参照してください](#)。

🔍 ターゲットの追加*

https://docs.aws.amazon.com/ja_jp/systems-manager/latest/userguide/integration-vss.html

CloudWatch Eventsを利用したSnapshot

- CloudWatch Eventsのターゲットである EC2 Create Snapshot API を利用し、対象のボリュームIDを指定

イベントソース

イベントパターンを構築またはカスタマイズするか、スケジュールを設定してターゲットを呼び出します。

☐ イベントパターン ⓘ ☒ スケジュール ⓘ

☐ 一定の速度 選択

☒ Cron 式

以後 10 回のトリガー日

1. Mon, 11 Mar 2019 17:30:00 GMT
2. Tue, 12 Mar 2019 17:30:00 GMT
3. Wed, 13 Mar 2019 17:30:00 GMT
4. Thu, 14 Mar 2019 17:30:00 GMT
5. Fri, 15 Mar 2019 17:30:00 GMT
6. Sat, 16 Mar 2019 17:30:00 GMT
7. Sun, 17 Mar 2019 17:30:00 GMT
8. Mon, 18 Mar 2019 17:30:00 GMT
9. Tue, 19 Mar 2019 17:30:00 GMT
10. Wed, 20 Mar 2019 17:30:00 GMT

CloudWatch イベントスケジュールの詳細については、[こちらをご覧ください](#)。

ターゲット

イベントがイベントパターンに一致するか、スケジュールがトリガーされたときに呼び出すターゲットを選択します。

EC2 CreateSnapshot API 呼び出し

ボリューム ID*

CloudWatch Events needs permission to create snapshots of EBS Volumes. By continuing, you are allowing us to do so.

☒ この特定のリソースに対して新しいロールを作成する


☐ 既存のロールの使用 ⓘ

CloudWatch Events のアイデンティティベースのポリシーの詳細については、[こちらを参照してください](#)。

ターゲットの追加*

DLMを利用したSnapshot



- EC2の台数が多く、夜間、休日などに一時的に停止できる環境で有用
- EBSのSnapshotの取得から削除までを一貫して管理
- ボリュームのタグごとにポリシーを設定可能
- 2,3,4,6,8,12,24時間の間隔でSnapshotを取得し、最小1世代から最大1,000世代まで保持可能
- 指定した開始時間から1時間以内で開始 
 - EC2インスタンスの状態は考慮せずにSnapshotを開始
- リージョンごとに最大 100 個までのライフサイクルポリシーを作成可能

AWS Backupを利用したSnapshot



- EBSだけでなく、Amazon EFS、Amazon RDS、DynamoDB、AWS Storage Gatewayを含めたバックアップ運用を一元管理
- バックアップポリシーにて EBS の Snapshot の取得から削除までを管理
- バックアップ時の鍵をKMSで管理しながら、監視やログを含めてAWS Backupで一元化
- EC2インスタンスの状態は考慮せずにSnapshotを開始 
- 2019年3月20日時点で東京リージョンは未サポート

本日のアジェンダ

- Amazon Elastic Block Store (EBS) 概要 ～アップデートを含めて～
 - EBSの概要
 - タイプ別の特徴
 - EC2インスタンスとパフォーマンスの考慮
 - 監視
 - NVMe SSD
- EBSの機能
 - Elastic Volume
 - EBS Snapshot
 - 暗号化
- EBSの価格
- まとめ

暗号化

- ボリュームを暗号化すると、ボリューム内の保存データ、ボリュームとインスタンスの間で移動されるすべてのデータ、Snapshotがすべて暗号化される
 - 暗号化が有効であったとしても、利用者やアプリケーションから特に意識する必要はない。
- 暗号化/復号化の処理はハードウェア機能を使って実施するため、パフォーマンスへの影響は極めて小さい
- 暗号化されたSnapshotを復元すると暗号化されたボリュームが作成される

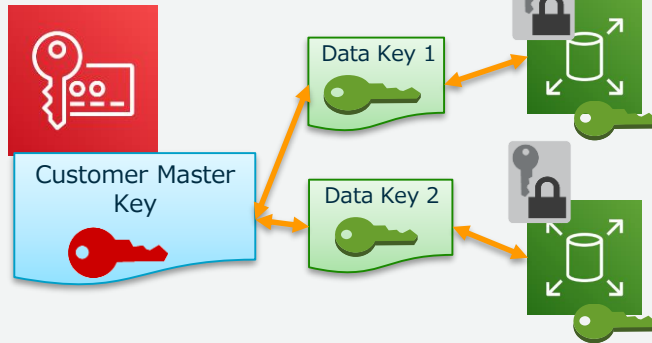
https://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/EBSEncryption.html



暗号化キー

- AES-256を使用してData KeyでEBS ボリュームを暗号化
- Data Keyは、暗号化する各ボリューム毎に一意のキーを生成し、暗号化されたデータと共にボリューム上に保存
- Data Keyの生成には、AWS Key Management Service (AWS KMS) カスタマーマスターキー(CMK)およびカスタマー管理の CMK の両方が利用可能

AWS KMS



エイリアスと説明の作成

このキーのエイリアスと説明を示します。キーのこれらのプロパティは、後で変更できます。

エイリアス (必須)

説明

▼ 詳細オプション

キーマテリアルオリジン ☒ KMS ☐ 外部

[選択のヘルプ](#)

対称 256 ビットキーをキー管理インフラストラクチャから KMS にインポートし、他のカスタマーマスターキーと同じように使用できます。詳細はこちら。

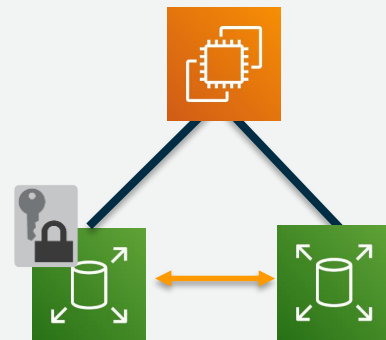
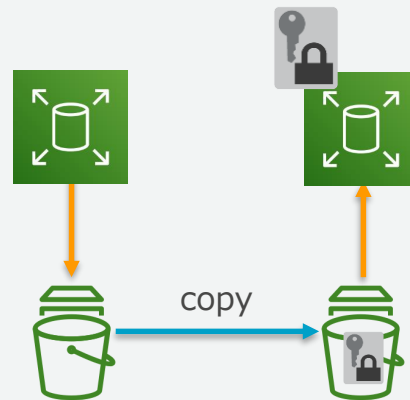
☐ インポートされたキーの使用によるセキュリティ、可用性、および耐久性への影響について理解しています。

Update!

https://docs.aws.amazon.com/ja_jp/kms/latest/developerguide/importing-keys.html

暗号化の有効・無効化

- EBS ボリューム作成後に暗号化を施したい場合は、Snapshot経由で暗号化を有効にできる
 1. Snapshotを取得する
 2. 暗号化を有効にしてSnapshotをコピー
 3. コピーされたSnapshotからEBS ボリュームを作成
 4. 新ボリュームをインスタンスにアタッチ
- 暗号化の解除を行う場合は新規ボリュームを作成してOS側でデータコピーを行う
 - Linux: rsyncコマンドなど
 - Windows: robocopyコマンドなど



起動ボリュームの暗号化

- 起動ボリュームの暗号化もサポート。ただし暗号化を有効にする際は、AMIのコピー機能を利用する

- 稼働中のインスタンスからAMIを作成する
- コンソールやCLI等でAMIコピーを実行。その際にSnapshotの暗号化を有効に設定する
- コピーされたAMIからインスタンスを起動する

AMI のコピー

AMI ami-c12c7bc0 は、新しい AMI へコピーされます。以下に新しいAMIの設定を行ってください。

送信先リージョン

名前

説明

暗号化 ☒ ターゲット EBS スナップショットの暗号化 ^①

マスターキー ^①

キーの詳細

説明	Default master key that protects my EBS volumes when no other key is defined
アカウント	このアカウント (7627)
KMS キー ID	ec46e951-
KMS キー ARN	arn:aws:kms:ap-northeast-1:7627:

キャンセル AMI のコピー

本日のアジェンダ

- Amazon Elastic Block Store (EBS) 概要 ～アップデートを含めて～
 - EBSの概要
 - タイプ別の特徴
 - EC2インスタンスとパフォーマンスの考慮
 - 監視
 - NVMe SSD
- EBSの機能
 - Elastic Volume
 - EBS Snapshot
 - 暗号化
- EBSの価格
- まとめ

EBSの価格

コストの要素	汎用SSD(gp2)	プロビジョンド IOPS(io1)	スループット 最適化HDD(st1)	コールドHDD(sc1)
容量	\$0.12/GB/月	\$0.142/GB/月	\$0.054/GB/月	\$0.03/GB/月
指定IOPS値	(対象外)	\$0.074/IOPS/月	(対象外)	(対象外)
I/Oリクエスト数	(対象外)	(対象外)	(対象外)	(対象外)
Snapshotの容量	\$0.05/GB/月	\$0.05/GB/月	\$0.05/GB/月	\$0.05/GB/月

- ・ 2019年3月20日時点の東京リージョンおよび大阪ローカルリージョンにおける価格
- ・ Snapshotを取得すると対象ボリュームの実データのみを圧縮して保存する
- ・ すべての EBS ボリュームは、EC2 からアタッチされていない場合でも課金される

<https://aws.amazon.com/jp/ebs/pricing/>

本日のアジェンダ

- Amazon Elastic Block Store (EBS) 概要 ～アップデートを含めて～
 - EBSの概要
 - タイプ別の特徴
 - EC2インスタンスとパフォーマンスの考慮
 - 監視
 - NVMe SSD
- EBSの機能
 - Elastic Volume
 - EBS Snapshot
 - 暗号化
- EBSの価格
- まとめ

まとめ

- EBS はバックアップや暗号化の機能を備えたセキュアに利用できる永続化ストレージ
- 4つのボリュームタイプからパフォーマンスやコストに応じて最適な EBS ボリュームを選択して利用。EBS タイプやパフォーマンスは EC2 にアタッチ中も変更可能
- アクセスパターンが読めない場合は gp2 を利用。HDDのボリュームタイプはシーケンシャルアクセスに最適
- 高いパフォーマンスが必要な場合は、io1 を利用し、EBS ボリュームだけでなくEC2 側のスループットにも注意
- 利用シーンに合わせて、Snapshot の取得方法を選択

参考資料

- Amazon Elastic Block Store(EBS)
<http://aws.amazon.com/jp/ebs/>
- ドキュメント：EBSの概要
http://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/AmazonEBS.html
- ドキュメント：EBS最適化インスタンス
http://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/ebs-ec2-config.html
- ドキュメント：EBS APIおよびコマンド概要
http://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/ebs-api-cli-overview.html
- ドキュメント：EBS よくある質問
<https://aws.amazon.com/jp/ebs/faqs/>

Q&A

お答えできなかったご質問については

AWS Japan Blog 「<https://aws.amazon.com/jp/blogs/news/>」にて
資料公開と併せて、後日掲載します。

AWS の日本語資料の場所「AWS 資料」で検索

AWS クラウドサービス活用資料集トップ

アマゾン ウェブ サービス (AWS) は安全なクラウドサービスプラットフォームで、ビジネスのスケールと成長をサポートする処理能力、データベースストレージ、およびその他多種多様な機能を提供します。お客様は必要なサービスを選択し、必要な分だけご利用いただけます。それらを活用するために役立つ日本語資料、動画コンテンツを多数ご提供しております。(本サイトは主に、AWS Webinar で使用した資料およびオンデマンドセミナー情報を掲載しています。)

AWS Webinar お申込 »

AWS 初心者向け »

サービス別資料 »

<https://amzn.to/JPArchive>

参考) st1/sc1 のパラメータチューニング(1)

- 高スループットで読み込みが主体となるワークロードにおいては、性能を最大限引き出すために先読み(Read Ahead)のサイズを 1 MiB に設定することを推奨

1. 現状の設定を確認する

```
$ sudo blockdev --report /dev/(device)
```

2. Read aheadの値を変更する

```
$ sudo blockdev --setra 2048 /dev/(device)
```

3. 設定変更結果を確認する

```
$ sudo blockdev --report /dev/(device)
```

```
[ec2-user@ip-172-31-31-63 ~]$ sudo blockdev --report /dev/xvde
RO  RA  SSZ  BSZ  StartSec      Size  Device
rw 2048  512  512        0 10995116277760 /dev/xvde
```



参考) st1/sc1 のパラメータチューニング(2)

- Linux Kernel のバージョン3.8以上を利用している場合は、先の設定に加えて `xen_blkfront.max` (Linuxカーネルバージョン4.6未満) または `xen_blkfront.max_indirect_segments` (Linux カーネルバージョン 4.6 以降)の値を 256 に設定することを推奨
- この値はカーネルモジュールパラメータとして指定を行う。Amazon Linux の場合は下記の手順で設定変更が可能
 - `/boot/grub/menu.lst`をviで開く
`$ sudo vi /boot/grub/menu.lst`
 - kernel行を以下の通り追記してOSを再起動
(Linuxカーネルバージョン4.6未満) `kernel /boot/vmlinuz-4.4.5-15.26.amzn1.x86_64
root=LABEL=/ console=ttyS0 xen_blkfront.max=256`
(Linuxカーネルバージョン4.6以降) `kernel /boot/vmlinuz-4.9.20-11.31.amzn1.x86_64
root=LABEL=/ console=tty1 console=ttyS0 xen_blkfront.max_indirect_segments=256`

参考) 設計上のパフォーマンス特性



- st1/sc1 は HDD の特性を活かし、高いスループットを低コストに実現することに最適化されている
 - ✓ 一般に HDD はシーケンシャルアクセスは高速だが、小さいデータへのランダムアクセスではヘッドの移動がオーバーヘッドとなるためパフォーマンスが出ない
- st1/sc1で小さいデータブロックへのアクセスを行った場合、シーケンシャルなら可能な限りI/O命令がマージされ効率的だが、ランダムな場合は非効率
 - ✓ (例)連続した 10 個の 128 KiB ブロックへのアクセスではクレジットの消費は2 MiB となる
 - ✓ (例)16 KiB ブロックにランダムで10回アクセスすると、クレジットは 10 MiB 消費される
- 小さいデータへのランダムアクセスになりがちなトランザクショナルな処理やデータベース、ファイルサーバ等への利用は非推奨

参考) st1/sc1の使いどころ Do's



Do's

- st1/sc1は安定したスループットを低コストで得られるよう設計されているので、ETL・DWH・ログ処理・EMRなどのシーケンシャルアクセス用途で利用する。
- st1 はディスクを高速(250 MiB /s以上)にスキャンする用途や、日次のバッチ処理などでボリュームをフルスキャンする用途に適している。
- sc1 はアクセス頻度が低いデータで250 MiB /s以下のスキャン速度でよいものを低コストに保管することができるので、st1 までのパフォーマンスは不要な場合を選択する。

参考) st1/sc1の使いどころ Don'ts



Don'ts

- トランザクション処理やランダムI/Oを多数発行する処理、起動ボリュームには向いていない。
 - ✓ gp2 や io1 の利用を推奨
- 起動ボリュームなど小容量で低コストを追求する場合はマグネティックの利用を考えてもよい。st1/sc1は起動ボリュームには非対応。
- 非常に高いスループットを得るために D2 インスタンスを利用している場合は、現状のままとすることが良いケースが多い。

参考) 事前ウォーミング(Pre-Warming)

- Snapshotから復元したボリュームに限り、各データブロックへの初回アクセス時、S3からのデータ取得が発生するためI/O命令のレイテンシが増加することがある
- 全領域からの読み込み処理の実行による事前ウォーミング(Pre-Warming)を行うことで、Snapshotから復元したボリュームに対する初回アクセス時のペナルティを回避できる
- 実運用時は事前ウォーミングが不可能なケースもあるため、運用要件から判断して実行可能であれば取り込む程度でOK
 - 例) Auto Scalingで起動したインスタンス

参考) ボリュームのコスト例

- 1 TiB の gp2 を1ヶ月間利用した場合
 - 容量分 : $0.12 \times 1024 = \text{約}122.9$ ドル (約13,500円)
- 1 TiB の io1 を 5000 IOPS で1ヶ月間利用した場合
 - 容量分 : $0.142 \times 1024 = \text{約}145.4$ ドル (約16,000円)
 - IOPS 分 : $0.074 \times 5000 = 370$ ドル (約40,700円)
 - 合計 : 約516ドル (約56,900円)

※読み書きブロック数=1TB/8KB=134217728 (約1億3千4百万リクエスト)

参考) ボリュームのコスト例

- 4 TiB の st1 を1ヶ月間利用した場合
 - 容量分 : $0.054 \times 4096 = \text{約}221.2$ ドル (約24,300円)
- 8 TiB の sc1 を1ヶ月間利用した場合
 - 容量分 : $0.03 \times 8196 = \text{約}245.9$ ドル (約27,000円)
- 50%使用済みの gp2 1 TiBのボリュームから Snapshot を作成し、1ヶ月間にわたり保持した場合 (圧縮率は25%と仮定)
 - Snapshot実容量 : $500 \times 0.25 = 125$ GB
 - 容量分 : $0.05 \times 125 = 6.25$ ドル (約690円)

ご視聴ありがとうございました

AWS 公式 Webinar

<https://amzn.to/JPWebinar>



過去資料

<https://amzn.to/JPArchive>

