**AMI**

・リージョン固有であり、コピー先のリージョンにコピーすることで新しいインスタンスを新しいリージョンで起動することができる

**Linux インスタンスの休止**

・インスタンスを休止状態にすると、インスタンスメモリの内容がAmazonEBS ルートボリュームに保存される、インスタンスが再起動すると、インスタンスメモリ（RAM）の内容が再ロードされる

**MFA Delete**

・セキュリティを強化するにはMFADeleteに対応するようにバケット設定する

この設定を行うとバケットの所有者は特定のバージョンを削除したりバケットのバージョニングを削除したりバケットのバージョニング状態を変更したりするために、全てのリクエストに2つの認証形態を含める必要がある

**Elastic Beanstalk**

・AWSクラウド内でのアプリのデプロイと管理をさらに簡単にするサービス

・開発者は単にそのアプリをアップロードするだけで、EBが自動的に容量のプロビジョニング、負荷分散、Auto-Scailng、およびアプリ状態モニタリングといったデプロイ詳細を処理する

**AWS Config**

・リソースを評価する

・AWSリソースの設定を評価、監査、審査できるサービス

・AWSリソースの設定が継続的にモニタリングおよび記録さ、望まれる設定に対する記録された設定の評価を自動的に実行できる

・Configを使用するとAWSリソース間の設定や関連性の変更を確認し、詳細なリソース設定履歴を調べ、社内ガイドラインで指定された設定に対する全体的なコンプライアンスを確認できる、これにより、コンプライアンス監査、セキュリティ分析、変更管理、運用上のトラブルシューティングを簡素化できる

**Lambda**

**・**数秒でスケーリングできるサーバーレスソリューションを提供する、API Gatewayとの統合も可能

・最大実行時間は15分、最大実行時間は頻繁にアップデートされるためLambdaクォータを確認する

<https://docs.aws.amazon.com/ja_jp/lambda/latest/dg/gettingstarted-limits.html>

・LambdaはDynamoDBと統合されている

**EC2インスタンスストア**

・毎秒数百万のトランザクションに対して最小のレイテンシーをサポートする、また、データはすべてのマシンですでに複製されているためデータが失われる可能性はない

EC2 インスタンスタイプ

オンデマンドインスタンス

・標準ではオンデマンドインスタンス

・あらかじめ決められた一定のレートの料金で使用した時間に対して課金されるため、仕様期間（例えば、1ヶ月間など）の時間帯にしか使用しないサーバー群や、キャンペーン時の一時的なWebサイトなどが考えられる

リザーブドインスタンス

・あらかじめ決められた使用期間（一年または三年）分を購入することで最大72％オフの割引価格が適用される

・支払い方法は「金額前払い」「一部前払い」「前払いなし」の３種類のオプションから選択できる、ただし「全額前払い」では途中で利用をキャンセルしても払い戻しされない

スタンダートタイプ

・リージョンやAZを指定してインスタンスを購入

コンバージブルタイプ

・あらかじめ指定したインスタンス構成（ファミリー、OSなど）に依存せず、柔軟で構成変更（ただし、作成時の価格と同等以上）が可能、その分だけ割引率はスタンダートタイプより低く設定されています

スポットインスタンス

・EC2の余剰リソースを入札形式で安価に利用する方法、割引率最も高い（最高で90％オフ）購入オプション

・短時間だったら（数時間程度）最も安いオプション

**VPCエンドポイント**

**NATゲートウェイ**

・ネットワークアドレス変換サービス

・NATゲートウェイを使用すると、プライベートサブネット内のインスタンスはVPC外のサービスに接続できるが外部サービスはそれらのインスタンスとの接続は開始できない

インターネットゲートウェイ

**Application Load Balancer**

**・**二つの異なるターゲットグループで使用し、URLのクエリ文字列に基づいたリスナールールを定義することで、リクエストを適切にルーティングできる

・一つのALBを使用して、二つの異なるだーげっとグループを構成できる、これによりALBは一部のリクエストをAWSでホストされていｒアプリにルーティングし、他のリクエストをオンプレミスでホストされているアプリにルーティングできる

この構成はＵＲＬのクエリ文字列に基づいてリスナールールを設定することで、リクエストを適切なターゲットグループに振り分ける事ができる

**Netwrok Load Balancer**

・毎秒数百万のリクエストを処理できる

・ロードバランサーは接続リクエストを受信すると、デフォルトルールのターゲットグループからターゲットを選択する

リスナー構成で指定されたポート上で選択した、ターゲットへのTCP接続を開こうとする

・転送中のデータの暗号化を提供するためにTLS設定ができる

これによりウェブ層に転送されるデータのセキュリティ性を向上する

**AutoScaling**

・負荷を予測できるならスパイクのインスタンスをスケーリングするようにAuto Scalingスケジュールに基づくスケーリングポリシーを設定する

・急なアクセス増加（スパイク）には対応できない

AMIからEC2を作成し起動するところで時間を要するため

| **スケーリング方法** | **詳細** | | **ユースケース** |
| --- | --- | --- | --- |
| **手動** | グループサイズを手動でスケーリング | | 緊急インスタンスの増減が必要な場合 |
| **スケジュールに基づく** | 指定した日付と時刻に自動でスケーリング | | ピークの時間帯が決まっている場合 |
| **予測** | 過去のメトリクスを分析し、今後を予測した上でスケーリングする | | 朝のみ集中するとパターンが決まっている場合 |
| **動的** | **簡易** | 一つのメトリクスを基準にスケーリングする | その時のサービス需要に合わせて柔軟にスケーリングを行いたい場合 |
| **ステップ** | 一つのメトリクスに複数の閾値を設定しスケーリングする |
| **ターゲット追跡** | 一つのメトリクスに目標値を指定し、スケーリングする |

**Route53**

・DNSウェブサービス

主にドメイン名をIPアドレスに変換する役割を果たす

・URLクエリ文字列に基づいてルーティングを行う機能は提供していない

ルーティングポリシー

シンプルルーティングポリシー

・ドメインで特定の機能を実行する単一のリソースがある場合に仕様する

・例えばexample.comウェブサイトのためにコンテンツを提供する１つのウェブサーバーなど

フェイルオーバールーティングポリシー

・アクティブ/パッシブフェイルオーバーを設定する際に使用する

位置情報ルーティングポリシー

・ユーザーの位置情報に基づいてトラフィックをルーティングする際に使用する

地理的近接性ルーティングポリシー

・リソースの場所に基づいてトラフィックをルーティングし

必要に応じてトラフィックのある場所をリソースから別の場所のリソースに移動する際に使用する

レイテンシールーティングポリシー

・複数の場所にリソースがあり、レイテンシーの最も小さいリソースにトラフィックをルーティングする際に使用する

複数回答ルーティングポリシー

・アトランダムに選ばれた最大８つの正常なレコードを持つDNSクエリにRoute53を応答させる場合に使用する

加重ルーティングポリシー

・指定した比率で複数のリソースにトラフィックをルーティングするのに使用する

**AWS Global Accelerator**

・AWSのグローバルネットワークインフラを利用して、ユーザーのトラフィックのパフォーマンスを最大60%向上させるサービス

・アプリへのパスの最適化はインターネットが混雑してる場合にパケット損失、ジッター、レイテンシーを一貫して、低く保つのに役立つ

・複数のリージョンにデプロイされた、アプリに対して、リージョン間の自動フェイルオーバーを実現できる

標準アクセラレーター

**カスタムルーティングアクセラレーター**

・転送速度の向上、高い可用性、セキュリティ向上などの利点を受け取れる

**Transfer Family**

・カスタムルーティングアクセラレーターの一部で、データ転送を高速かつセキュアに行うためのソリューションを提供する

・大容量のデータ転送やセキュアなファイル転送が必要な場合に有効

・FTP FTPS SSL TLS SFTPなどのプロトコルをサポートしている

・データの転送状況の監視、アクセスコントロールの設定、転送ログの取得など、さまざまな管理機能を提供している

・AWSの他のサービスと統合することができる、S3との連携によりファイルの保存、バックアップが用意になる

**TransferEndpoint**

・FTP、SFTPクライアントなどのツールを使用して、データをアップロードやダウンロードすることができる

**Amazon FSx for Windows File Server**

・業界標準のサーバーメッセージブロック（SMB）プロトコルを介してアクセスできる、信頼性が高くスケーラブルな完全マネージド型のファイルストレージを提供する、Windows Serve上に構築され、ユーザークォータ、エンドユーザーファイルの復元、Microsoft Active Directory(AD)統合など幅広い管理機能を提供する

・SSDおよびHDDストレージオプションを使用してワークロードのニーズに合わせて、コストとパフォーマンスを最適化できる

・ストレージを拡張し、ファイルシステムのスループットパフォーマンスをいつでも変更できる、Amazom FSx ファイルストレージにはWindows Linux MacOSのコンピューティングインスタンスとAWSまたはオンプレミスで実行されているデバイスからアクセスできる。

・同じアクセス許可を保持している会社のActiveDirectoryに基づくアクセス制御を備えた共有ファイルシステムを作成できる

**Amazon Elastic File system（EFS）**

・シンプルでサーバーレス、セットエンドフォーゲットできる伸縮自在なファイルシステムを提供する

**※Amazon EFSライフサイクル管理**

・ファイルシステムのコスト効率の高いファイルストレージを自動的に管理する

・有効にすると、ライフサイクル管理はファイルシステムに応じて、一定期間アクセスされていないファイルを、低頻度アクセス（標準IA）または１ゾーン低頻度アクセス（１ゾーンIA）ストレージクラスに移行する、この期間はIAへの移行ライフサイクルポリシーを使用して定義する

**Amazon EBS**

AWSのクラウドストレージサービスであり、EC2インスタンスに永続的なブロックレベルストレージを提供する

|  |  | 【使用ケース】 |
| --- | --- | --- |
| **汎用SSDボリューム** | 中程度のパフォーマンスとコストバランスを提供するストレージオプション | ・汎用SSDボリューム→　ワークロードに適した柔軟な性能を提供  ・Provisioned IOPS SSD→　明示的に指定したIOPSを保障 |
| **Provisioned IOPS SSD ボリューム** | 高速で予測可能なパフォーマンスを提供するために、事前にIOPSを割り当てることができるEBSストレージオプション |
| **スループット最適化 HDD ボリューム** | 大容量のデータストレージと高いデータ転送速度を提供するために最適化されたEBSストレージオプション | ・スループット最適化 HDD ボリューム→　大容量のデータのシーケンシャルアクセスを最適化するために  ・Cold HDD ボリューム→　低コストで大容量のデータを保存、頻繁なアクセスが必要ではないデータ |
| **Cold HDD ボリューム** | 低コストで長期間のデータ保持に最適化されたEBSストレージオプション |

※**IOPS**　ストレージデバイスが1秒間に処理できる入出力操作の数を表す性能指標

**スループット最適化HDD**

・500MiB/秒のスループットをサポートし、大規模なシーケンシャルデータに理想的なストレージソリューションを提供する

**Amazon S3**

・費用対効果の高い方法でファイルを保存するために使用できる

・ファイルはローカル処理のために**EBS**にダウンロードすることができる

**S3 Intelligent-Tiering**

・オーバーヘッドなしで、オブジェクトの可用性、パフォーマンス、または耐久性に影響を与えることなく、ストレージコストを自動的に最適化するのに役立つ

**S3署名付きURL**

・署名付きURLの作成者がそのオブジェクトへのアクセス許可を持っている場合、署名付きURLを使用して、URLで識別されるオブジェクトにアクセスすることができる

つまり、オブジェクトをアップロードするために署名付きURLを受け取った場合、オブジェクトをアップロードすることができるのは

署名付きURLの作成者がそのオブジェクトをアップロードするのに必要な権限を持っている場合のみ

**Amazon S3 Glacier**

・標準取り出しが3~5時間以内に完了する最もコスト効率の高いソリューションを提供する。

**Amazon S3 Glacierアーカイブの取り出しオプション**

**標準取り出し（Standard）**

・数時間以内にすべてのアーカイブオブジェクトにアクセスできる

標準取り出しは取り出しオプションを指定しないでS3 Glacierおよび

S3 Glacier Deep Archiveの取り出しリクエストを行った場合にデフォルトで適用される

アーカイブへのアクセスは通常３～５時間かかる

**大容量取り出し（Bulk）**

・S3 Glacierの最も安価な取り出しオプション、大量のデータ（ペタバイトのデータも含む）を安価に取得できる

大容量の取り出しはS3Glacierストレージクラスに格納されているオブジェクトに対して5～12時間かかる

**迅速取り出し（Expedited）**

・アーカイブのサブセットが緊急に必要になった場合にデータにすばやくアクセスできる

250MBを超えるような最大サイズのアーカイブを除き

Expedited取り出しを使用してアクセスするデータはすべて、1～5分以内に使用可能になる

※迅速取り出しにはオンデマンドとプロビジョニング済みの２種類がある

**オンデマンド**

・１～５分以内に取り出しを完了できる場合に行う

**プロビジョニング**

・必要に応じて迅速取り出しで利用できる取り出し容量を確保する

**S3 レプリケーション**

・レプリケーションでは異なるS3バケット間でオブジェクトを自動的に非同期コピーできる

**クロスリージョンレプリケーション（CRR）**

・異なるAWSリージョンのS3バケット全体にオブジェクトをコピーするために使用される

※レプリケーションの要件

レプリケーション元とレプリケート先の両方のバケットで、バージョニングを有効にする必要がある

**Cross-Origin Resource Sharing(CORS)**

・特定のドメインにロードされたクライアントウェブアプリケーションが異なるドメイン内のリソースと通信する方法を定義するサービス

・CORSのサポートにより、S3を使用して機能豊富なクライアント側ウェブアプリケーションを構築してS3リソースに対するクロスオリジンアクセスを選択的に許可できる

**Storage Gateway**

・Storage GatewayのボリュームゲートウェイはiSCSIプロトコルを使用してアプリケーションにブロックストレージを提供し、ＡＷＳ上にストレージを拡張できる

**Amazon S3 ファイルゲートウェイ**

**・**NFSプロトコルをサポートし、ローカルキャッシュを使用してデータへの低レイテンシーアクセスを提供する

・SMBプロトコルをサポートし、ローカルキャッシュはデータへの即時アクセスを提供できる

・ＳＭＢプロトコルをサポートしていて、オンプレミスのデータベースをローカルファイルサーバーにバックアップするのに適している

・ファイルゲートウェイにはキャッシング機能があり、一週間分のバックアップファイルにすぐにアクセスできる

**Amazon S3 ボリュームゲートウェイ**

**・**オンプレミスのアプリケーションサーバーからiSCSI(Internet Small Computer System Interface) デバイスとしてマウントできる、クラウドベースのストレージボリュームを提供する

**Amazon S3 ファイルゲートウェイ**

・S3サービスと仮想ソフトウェアアプライアンスを組み合わせる、この組み合わせを使用することで、ネットワークファイルシステム（NFS）やサーバーメッセージブロック（SMB）などの業界標準のファイルプロトコルを使用して、AmazonS3でオブジェクトを保存および取得できる

**キャッシュボリュームモード**

・アプリに変更を加えず、Amazon S3にデータを保存し、オンプレミスで頻繁にアクセスするデータのコピーをローカルに保持する事ができる

**Amazon S3 Transfer Acceleration**

**・**異なる地理的な地域からS3への高速データ転送を可能になる

・S3とクライアント間のデータ転送を最適化し、アップロードおよびダウンロード速度を大幅に向上させる

・グローバルなアプリケーションで特に効果的でユーザーが大容量の動画ファイルを迅速にアップロード、ダウンロードするのに役立つ

**Amazon S3 Glacier Deep Archive**

・最も安価なS3ストレージクラス

・１年のうち１回が２回しかアクセスされないようなデータを対象とした長期保存をサポートする

・特に金融サービス、ヘルスケア、パブリックセクターなどの規制が厳しい業界を対象としている、コンプライアンスの要件を満たすために７～１０年以上データセットを保管するように設計されている

AWS Resource Groups

**AWS WAF**

・不正なIPアドレスからの攻撃を防ぐためにセキュアでスケーラブルなソリューションを提供する

オリジンアクセスアイデンティティー（OAI）（S3）

**Snowball**

・データ転送デバイスのこと

・大量のデータを高速、セキュアにAWSクラウド環境に移動するために使用される

**DateSync**

・日付や時刻の同期を行うための技術やツールのこと、異なるシステムやデバイス間で正確な日時情報を共有するために使用される

**Migration Hub**

・アプリやデータを異なる環境やプラットフォームに移行する際に使用されるツールやサービス、移行プロセスを管理し、可視化して効率的に移行を実施するための機能を提供する

**Amazon Elasti Cache for Redis (SQL系)**

AWSが提供するマネージドサービスなRedisデータベースサービス

Redisは高速なキャッシュやデータストアとして使用され、SQL系のデータベースとは異なるデータベースの一種

**Redis AUTH コマンド**

・Redis AUTH コマンドを使用すると、パスワードで保護されたRedisサーバー上でRedisコマンドを実行する権限を付与する前にユーザーにパスワードを入力を要求することにより、データのセキュリティの向上させることができる

・ユーザーがパスワードで保護されたRedisサーバーにパスワードを入力するように要求させるには

レプリケーショングループまたはクラスターを作成するときに、レプリケーションまたはクラスターに対する後続のすべてのコマンドで、パラメーター –auth-token に正しいパスポートを含めるようにする

**サブスクライブ（SQL系）**

・SQL系のデータベースで使用される用語でデータベースの特定のテーブルやクエリ結果を監視し、変更があった場合に通知する仕組み

変更をリアルタイムで追跡するため使用される

**FIFOキュー（SQL系）**

・First-In-First-Out（先入れ先出し）の原則に基づいてデータを処理するキューのこと

データは順番に処理され、最初に追加されたデータが最初に取り出される

**Auroraレプリカ**

・AuroraDBクラスター内の独立したエンドポイント

・読み取りオペレーションのスケーリングと可用性の向上に最適

最大１５個のAuroraレプリカを一つのＡＷＳリージョンの中でＤＢクラスターが処理するＡＺ全体に分散できる

**Amazin Aurora Global Database**

・リージョン間での高可用性を実現するために設計されたデータベースサービス

・低遅延によりアプリはすぐにレポートを作成するために必要なACIDトランザクションを利用できる

・Amazon AuroraはSQLと互換性がある

**ElastiCache for Redis**

・データベース前に追加することで、データベースへの読み取り負荷を軽減できる

これは、キャッシュされたデータを高速に取得することができるため、データベースへのアクセス回数が現象し、パフォーマンスが向上する

・スナップショット、レプリケーション、およびミリ秒未満の応答時間の機能を提供していて、これによりアプリケーションの可用性とパフォーマンスが向上する

Amazon Aurora

・最強

・標準的なMySQLデータベースと比べて最大で５倍、標準的なPostgreSQLデータベースと比べて最大で３倍高速

・AuroraとDynamoDBは3つの異なるAZに格納する

Auroraのストレージシステム

・分散型で耐障害性と自己修復機能を備えている

・データベースインスタンスごとに最大128TBまで自動的にスケールされる

・最大15個の低レイテンシーリードレプリカ、ポイントタイムリカバリ、Ｓ３への継続的なバックアップ、3つのＡＺ間でのレプリケーションにより優れたパフォーマンスと可用性を発揮する

**Amazon Aurora MySQL**

・アプリコードや運用要件の変更を最小限に抑えつつ

一秒未満のレプリカラグを維持してアプリを拡張することができる

・Aurora Auto Scalingを有効にすることで負荷の変化に応じて、自動的にスケーリングされアプリのパフォーマンスが向上する

**Aurora Serverless for Aurora MySQL**

最小限の操作でパフォーマンス、スケーリング、耐久性を向上させる

**サービスコントロールポリシー（SCP）**

組織を管理するために使用できるポリシーのタイプ

SCPは組織内のすべてのアカウントの最大使用アクセス制限を一元的に管理できる機能を提供し、アカウントが組織のアクセスコントールガイドラインに沿って活動することを確実にする

※組織が一括請求機能のみを有効にしている場合、SCPは使用できない

**DynamoDB**

・規模に関係なく数ミリ秒台のパフォーマンスを実現する

・世界最大のアプリの一部サポートしている

・事実上無制限のスループットとストレージでアプリを構築できる

**DynamoDB Accelerator（DAX）**

・フルマネージド型高可用性インメモリキャッシュでDynamoDB用に特化している

・一秒あたりのリクエスト数が数百万件になる場合でも、ミリセカンドからマイクロセカンドへの最大10倍のパフォーマンスで向上を実現する

**Amazon Athena**

・インタラクティブなクエリサービスでS3内のデータを標準SQLを使用して簡単に分析できる

・Athenaはサーバーレスなのでインフラストラクチャの管理は不要

・操作が簡単

・Athenaは初期状態でAWS Glueデータカタログと統合されている

**Amazon RDS**

・最小限の労力でAWSに移行できる、マルチAZはディザスタリカバリをサポートをし、リードレプリカは読み取りトラフィックの処理に役立つ

RDSリードレプリカ

・リードレプリカによってRDSデータベースインスタンスのパフォーマンスと耐久性が向上する、これを使用すれば読み取り頻度の高いデータベースのワークロードに対して単一ＤＢインスタンスキャパシティーの制約を超えて伸縮自在にスケールアウトできる

・アプリの大容量読み取りトラフィックをデータの複数のコピーから提供することにより全体の読み込みスループットを向上させることができる

・リードレプリカはRDSforMySQL MariaDB PostgreSQL Oracle SQL Server Amazon Auroraで利用できる

RDS forPostgreSQL

※MySQLおよびPostgreSQLはIAMデータベース認証

・IAMデータベース認証をサポートしている、これにより一時的な認証トークンを使用してデータベースを認証できる

**Amazon Redshift**

・ワークロード管理機能を提供していて、インタラクティブで長時間実行されるジョブの優先順位付けに役立つ

また、データを単一のストレージサービスに保存することで、コストを最小限に抑えることができる

**リードレプリカの使用**

・MariaDB、MySQL、Oracle、PostgreSQL、Microsoft SQL Serverの

DBエンジンに組み込まれたレプリケーション機能を使用して、リードレプリカと呼ばれる特殊なタイプのDBインスタンスをソースDBから作成する

・ソースDBインスタンスがプライマリDBインスタンスになる

・プライマリDBインスタンスに対して行った更新はリードレプリカに非同期的にコピーされる、読み込みクエリをアプリからリードレプリカにルーティングすることにより、プライマリDBインスタンスの負荷を低減できる

**TTL属性**

データの有効期限やキャッシュ時間の管理を容易にするために使用される

※DynamoDBではデータの自動削除を、Route 53ではDNSレコードのキャッシュ時間を制御するためにTTL属性が活用される

**AWS Key Management Service（KMS）**

・KMS Keys（KMS キー）とデータキーを生成する機能を提供する

キーポリシーはキーへのアクセスの制御に役立つ

・KMSを使用することで暗号化キーを簡単に作成して管理し、幅広いＡＷＳのサービスやアプリケーションでの使用を制御できるようになる

・ＫＭＳはセキュアで弾力性の高いサービスで、キーを保護するためにFIPS140-2の検証済みまたは検証段階のハードウェアセキュリティモジュールを使用する

・KMSでのキーポリシーはKMSキーにのみ適用される

・KMSからKMSキーをエクスポートできない

**Transit Gateway**

・複数のVPCやオンプレミスネットワークを効率的に接続し、一元的に管理することができる

・ECMPルーティングと組み合わせることで、VPNトンネルの数を増やしスループットをスケーリングすることが可能

**CloudTrail**

・AWSアカウントのガバナンス、コンプライアンス、および運用とリスクの監査を行えるように支援するAWSのサービス

・ユーザ、ロール、またはAWSサービスによって実行されたアクションはCloudTrailにイベントとして記録される

**CloudFront**

・コンテンツ配信ネットワークは大規模にスケールされグローバルに展開している、CloudFrontネットワークは216のPOPを有しているほか、弾力性に優れたAmazonバックボーンネットワークを活用しており、エンドユーザーに優れたパフォーマンスと可用性を提供する

・ユーザーへの静的、動的ウェブコンテンツ（.html .css .js イメージファイルなど）の配信を高速化するウェブサービス

・エッジロケーションと呼ばれるデータセンターの世界規模のネットワークを通じてコンテンツが配信される

・エッジロケーションでは最も低いレイテンシーが提供されるのでコンテンツは最高のパフォーマンスで配信される

**CloudFrondの料金**

・企業やウェブアプリケーション開発者に、少ない待ち時間と高いデータ転送速度でコンテンツを配信できる

・簡単で費用対効果の高い方法を提供するウェブサービス

・予約容量を用いると割引が適用される、予約容量とはお客様でに12ヶ月以上の期間、最低使用量レベルを約束してもらうオプション

その代わりに大幅な割引料金を提供するもの

・予約容量契約は単一の領域から毎月のデータ転送量10TBが最低ライン、使用量が多ければ追加の割引が適用される

**CloudWatch**

・モニタリングおよび管理サービス

・AWSリソースの監視、パフォーマンスの分析、ログの収集、アラームの設定などを行うことができる

・リソースのモニタリングにはEC2インスタンス、RDSデータベース、ELBロードバランサーなど、さまざまなAWSサービスを監視できる

・異常が発生した場合やリソースの状態が一定のしきい値を超えた場合、ログで特定のキーワードを検出した場合などにアラートを通知するなど状態に応じたアクションをとることができる

**CloudWatch Logs**

・CloudWatchLogsでサブスクリプションを作成することによって、KinesisDataFirehoseに送信し、そのあとOpenSearch Serviceにロードできる。

**Secrets Manager**

・アプリ、サービス、ITリソース、アクセスに必要なシークレットの保護を支援する

・データベースの認証情報、APIキー、この他のシークレットをそのライフサイクルを通して容易にローテーション、管理、取得できるようにする

・ユーザーとアプリはSecrtsManagerAPIを呼び出してシークレットを取得するので秘密情報をプレーンテキストでコードに書き込む必要がなくなる

・SecretsManagerはRDS、Redshift、DocumentDBへの統合を組み込むことでシークレットのローテーションを提供する

APIキーやOAuthトークンなどの他のタイプのシークレットにも使える。

**CloudFormation**

・AWSが提供するインフラストラクチャ自動化サービス、テンプレートを使用してAWSリソースを定義し、効率的にインフラストラクチャを作成、管理、更新することができる

**Amazon Kinsis**

・リアルタイムにログを取得して処理するためのスケーラブルなストレージを提供している

・ストリーミングデータをリアルタイムで収集、処理、分析することが簡単になるため、インサイトを適時に取得して新しい情報に迅速に対応できる

**Kinesis Data Streams**

・リアルタイムのストリーミングデータを簡単に収集、保存、処理、分析できるため、タイムリーな洞察を得て、新しい情報にすばやく対応できる

・データレコードの大量のストリームをリアルタイムで収集し、処理するにはKinesis Data Streamsを使用する

・ストリーミングデータ処理を支援し、パーティション内の順序を維持する

Kinesis Data streamはKinesis Data Firehoseと統合してデータをS3に保存し、コスト効率の高いソリューションを提供する

**Kinesis Data Firehose**

・リアルタイム配信用の完全マネージド型サービス

・複数のプロデューサーからのデータを取得して、S3にプッシュできる

・リアルタイムのストリーミングデータなどの目的地へS3、Redshift

ESなどサポート対象のサードパーティサービスプロバイダーが所有するカスタムHTTPエンドポイントまたはHTTPエンドポイント

Kinesis Data Firehoseを使用すると、アプリを記述したりリソースを管理したりする必要はない

**AWS Fargate**

サーバーレスで従量制料金のコンピューティングエンジンであり、サーバーを管理することなくアプリケーションの構築に集中することができる

**AWS Directory Service**

・Directory Serviceを使用してマネージド型Active Directoryを作成し、現在のインスタンスをアンインストールすることができる

また、マネージドサービスを利用することでIT部門の管理者の負担を軽減することができる

**プロトコル**

**UDP**

・ネット上でデータを送受信するためのプロトコル

・非接続型で信頼性、順序性の制約が少ない特徴がある

・データを小さなパケットに分割し、宛先に送信する、パケットが確実に届くことを保証しないため、高速なデータ転送が可能だが、一部のパケット損失や順序の入れ替わりが起こる場合がある

・主にリアルタイム性が求められるアプリで使用される、音声ビデオのストリーミングなど

**TCP**

・ネット上でデータを送受信するためのプロトコル

・接続型で信頼性や順序性の制約がある

・データの送信前に接続を確率し、パケットの到着確認や再送制御を行う、そのため

データの完全性、順序を確保することができるが、通信のオーバーヘッドが発生し、UDPよりも遅延が生じる場合がある

・主にファイル転送、ウェブページの表示など、データの完全性、順序が必要なアプリで使用される

**HTTP**

・ウェブブラウザとウェブサーバー間で、情報をやり取りするためのプロトコル

・HTTPはテキストベースのプロトコルで、ハイパーテキストの形式でデータを送信する

・クライアントがウェブサーバーにリクエストを送信し、ウェブサーバーがそれに応答を返す、リクエストとレスポンスのやり取りに基づいて、ウェブページの表示やデータの送受信が行われる

**HTTPS**

・HTTPのセキュアバージョンで、データの暗号化とセキュアな通信を提供する

・通信を暗号化し、データの盗聴や改ざんを防ぐ

・オンラインショッピングや個人情報の送信など、セキュリティが重要な情報のやり取りに使用される

**SSH**

・リモートコンピューターに安全なリモートアクセスを提供するためのプロトコル

・ネットワーク上で安全な暗号化された通信を確率し、遠隔操作やファイル転送などを行うことができる

・セキュアなリモート管理やデータ転送など、セキュリティが求められる場面でよく使われる

**単語**

**セキュア**

・安全や保護を意味する

**レイテンシー**

・通信の遅延時間や応答時間を意味する

**トラフィック**

・ネットワークを通じて送受信されるデータの流れや量

**ジッター**

データの到着間隔の不規則さや変動を意味する

**オーバーヘッド**

・通信や処理において余分に必要なリソースや情報の事を意味する

**ポート４４３**

HTTPS 通信で使用される安全な通信プロトコル、SSL/TTLに対応している

**ポート１４３３**

・ネットワーク上でデータの受け渡しを行うための通信経路のひとつ

**キャッシュ**

・データや情報を一時的に保存するメモリやストレージ領域、再利用や高速なアクセスに役立つ

**ディストリビューション**

・システムやデータを複数の場所に分散配置すること、負荷分散や高可用性を実現するために使用される

**ディザスタリカバリ**

**フォールトトレランス**

**エッジロケーション**

・ネットワーク上のエンドユーザーに近い場所に配置されたサーバーやリソース、高速なアクセスや低レイテンシーを提供する

**スナップショット**

・特定の瞬間のデータや状態を保存すること、後で元の状態に戻すためのバックアップや復元に使用される

**ルーティング**

・ネットワーク上でデータパケットを目的地まで正しい経路で転送すること、効率的な通信やインターネット接続を実現する

**リードレプリカ**

・データベースなどのシステムにおいて、読み取り専用の複製を作成すること、読み取り処理の負荷分散や高可用性を提供する

**ユーザーエクスペリエンス**

・ユーザーがサービスや製品を利用する際の全体的な体験や満足度のこと、使いやすさや快適さが含まれる

**レプリケーション**

・データや情報を複数の場所に同期させること、データの可用性や耐障害性を高めるために使用される

**プレフィックス**

・単語や用語の前におかれる接頭辞

・例えばun- re- pre-などが一般的なプレフィックス

・単語の意味を変えたり、新しい意味や用途を付加する役割

**フェイルオーバー**

・システムやサービスが障害や故障した場合に、別のバックアップや予備のリソースに切り替わることを指す

・主な目的はシステムの可用性や信頼度を向上させること

・自動的に行われる場合もあるけど手動で行われることもある

**エンドポイント**

**・**ネットワーク通信において、通信の出口や目的地となる特定のアドレスや場所を指す

**多要素認証（MFA）**

MFAを有効にするとログインの際パスワードだけでなく多くの情報を入力する必要がある複数ステップのアカウントログインプロセス

※ パスワードに加え、E メールで送信されたコードの入力、秘密の質問への回答、または指紋のスキャンなど

**ソリューション**

・問題や課題を解決するための方法や手段の事

**SFPT**

・ファイル転送プロトコルの一種で、セキュアなファイル転送を行うためのプロトコル

**SFPTソリューション**

・SFPTを使用した問題解決や実装の方法や手段のこと

**FTPS**

・ファイル転送プロトコルの一種でSSL、TLSを使用してファイルをセキュアな転送を行うためのプロトコル

**RTPソリューション**

・リアルタイム通信プロトコルを使用した問題解決や実装の方法や手段のこと

**データレイク**

スマホアプリ、IoT デバイス、Web サイト、SNS など、さまざまソースから収集したデータを、非構造化データか構造化データかを問わず、まとめて入れておことができる

**スループット**

一定の時間で、どれだけの量のデータを転送できるかを示す処理能力のこと

（毎秒何GB転送できるか）

**アーカイブ**

・データや情報を保存、保管することを指す

**最小特権の原則**

セキュリティー的観点であらゆるユーザ、デバイスは、意図する機能を実行するために、必要最小限の特権のみを持つべきだとする概念

**ダッシュボード**

・グラフやチャートなどの視覚的な要素を使って、情報やデータを一元化して表示するためのインターフェースのこと

**メトリクス**

・測定や評価のために使用される数値や指標のこと、システムやプロセスの状態や性能を把握するために利用される

**踏み台サーバー**

・インターネットやネットワーク上のセキュリティを強化するために使用される中継サーバーのこと

外部からのアクセスを制限し、内部ネットワークへの安全な接続を提供する

**RDP認証情報**

・リモートデスクトッププロトコルを使用してリモート接続する際に必要な認証情報のこと

**フリート**

・複数のコンピューターやデバイスを一括管理するために使用される概念や仕組みのこと

同じ構成や設定を持つ複数のリソースをまとめて管理することができる

**ステートレス**

・状態を持たないことを指す、特定のリクエストや操作に対して事前の状態や情報の保存が必要なく、それぞれの処理が独立して行われる特性

**NFS**

・異なるコンピューター間でファイル共有を行うためのプロトコル

ネットワーク上でのファイル読み書きを可能にし、複数のクライアントが同じファイルにアクセスできるようにする

**SMB**

Windowsベースのネットワークファイル共有プロトコル

ファイルやプリンターの共有を行い、クライアントとサーバー間でデータのやり取りやアクセスを提供する

**iSCSI**

TCP/IPネットワークを介してストレージデバイスにアクセスするためのプロトコル

ネットワーク上のストレージリソースをブロックベースで提供し、サーバーからストレージデバイスに直接アクセスできるようにする

**iSCSI-VTL**

iSCSIプロトコルを使用してバーチャルテープライブラリにアクセスするための仕組み

バックアップやアーカイブなどのデータ管理を効率的に行うために使用される

**各レイヤー**

**レイヤー１　物理層**

・ネットワーク上の物理的な接続とデータの伝送を担当

・電気信号や光信号などの物理媒体を介してデータを送受信する

**レイヤー２　データリンク層**

・ネットワーク上の隣接するノード間での直接的な通信を制御する

・物理アドレス（MACアドレス）を使用してノードを特定し、フレームを呼ばれるデータの単位で通信を行う

**レイヤー３　ネットワーク層**

・異なるネットワークを結ぶためのパケットの転送とルーティングを担当する

・ＩＰアドレスを使用してノードを特定し、パケットと呼ばれるデータ単位で通信を行う

**レイヤー４　トランスポート層**

ネットワーク上のホスト間でのデータ転送や制御する

信頼性や順序性を保証するＴＣＯや非信頼性のデータ転送を行うＵＤＰなどのプロトコルがある

**レイヤー５　セッション層**

・通信のセッションの確率。終了を管理する

セッションの制御やで王基、エラー復旧などを行う

**レイヤー６　プレゼンテーション層**

・データの表現形式や圧縮などのデーターの変換や処理を行う

・データの表現形式に閑するプロトコルをしているする

**レイヤー７　アプリケーション層**

・ユーザーが直接提供するアプリとの通信を担当

・ウェブブラウジング（HTTP）メール転送（SMTP）ファイル転送（FTP)などのプロトコルが含まれる

YAML

・人間にとって読みやすく書きやすいデータシリアライゼーション形式の一つ、YAMLではデータを階層構造で表現することができる

YAMLのパラメータ

・階層構造の中で特定のデータを表すためのキーと値の組み合わせ、パラメータはデータの属性や設定、オプションなどを表現するために使われる

・パラメーターはインデントによって階層を表現し、階層の深さに応じて親子関係の表現

例

database:　←親パラメータ

host: localhost　←子パラメータ

port: 3306 ←子パラメータ

username: admin ←子パラメータ

password: password123 ←子パラメータ

databaseというパラメーターが親で、その下にhost port username passwordという子のパラメーターが存在していて

各パラメーターはキーと値のペアで構成されていて、データの接続設定に関する情報が表現されている

YAMLのパラメータは構造化されたデータを表現するための重要な要素であり、設定ファイル、データの定義、構成情報の管理など様々な場面で使用されている

**メモ**

・ポート443はSSL用で、データベース層ではなくウェブ層で必要とされる

ウェブ層ではインターネットからのアクセスを受け付けるので、セキュアな通信が通信が必要、データベース層はウェブ層からのみアクセスが必要で、ポート443を使用する必要はない。

・EC2インスタンス上で動作するMySQLは一秒未満のレプリカラグを保証しない

・EC2インスタンス上で稼働するMySQLはRDS For MySQLと異なり

運用管理が手動が行われるので多くの労力が必用になる

非常に大きなコンピューティング最適化インスタンスを選択したとしてもこの方法はレプリケーションの遅延を根本的に解決できな

・Elastic Cache For Redis クラスターをデータベースの前にデプロイする場合、ウェブサイトのコードを変更してキャッシュを確認するようにしなければならない、これによりアプリコードへの変更が最小限でなくなり、キャッシュの管理や運用も増えるので運用要件も増加する

・保管型ボリュームで設定されたStorage Gatewayボリュームゲートウェイでは、頻繁にアクセスするデータのローカルキャッシュを維持しないためレイテンシーは低くならない

・RDSインスタンスは直接暗号化することはできない、スナップショットを取得し、スナップショットを暗号化し、暗号化されたスナップショットを復元することで、暗号化されたRDSインスタンスを作成することができる

・S3バケットポリシーはアプリへのアクセスを制限することはできない

・データをS3に移行することを計画、現段階で一つのリージョンで単一のS３バケットを使用している、すべてのデータを複数のＡＷＳリージョンで使用可能でならなければならない

→別のリージョンにバージョニングを有効にして、追加のS3バケットを作成し、クロスリージョンレプリケーションを設定する

・S3はセキュリティ、耐久性、拡張性の高いクラウドストレージを提供する、ウェアページとクライアント側スクリプトを配置できる静的ウェブサイトをホストするようS3バケットに設定できる

ドメイントラフィックにルーティングするには、Route53を使用してバケットをポイントするエイリアスレコードを作成する

エイリアスレコードはＤＮＳへのRoute53拡張、ルートドメイン、サブドメインの両方にエイリアスレコードを作成できることを除いて、CNAMEレコードに似ている、サブドメインにのみCNAMEレコードを作成できる

・コールドデータストレージ用のS3

・高性能並列ストレージ用のFSx for Lustre

＃１５９　ActiveMQ