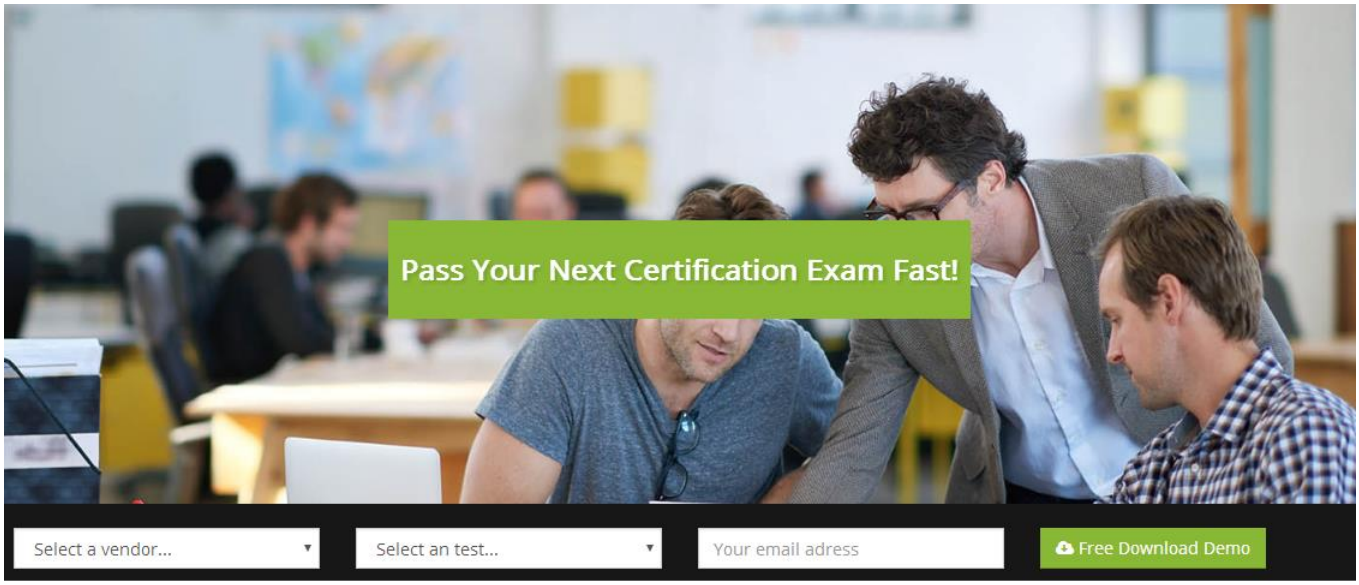
ITPassLeader



Instant Download | 365 Days Free Updates | Money Back Guarantee | Security & Privacy

Choose the version that fits your needs	PDF Version	Desktop Test Engine	Online Test Engine
Latest and Up-to-Date exam dumps with real exam questions answers.	✓	✓	✓
Get 12-Months free updates without any extra charges.	✓	✓	✓
Experience same exam environment before appearing in the certification exam.	✗	✓	✓
100% exam passing guarante in the first attempt.	✓	✓	✓
20% discount on more than one license and 30% discount on 5+ license purchases.	✗	✓	✓
100% secure purchase on SSL.	✓	✓	✓
Completely private purchase without sharing your personal info with anyone.	✓	✓	✓

http://www.itpassleader.com

High-praise Exam Dumps Questions grant you success by high pass rate - ITPassLeader

Exam : **AWS-Solutions-Associate-JP**

Title : AWS Certified Solutions Architect - Associate (SAA-C02) (AWS-Solutions-Associate日本語版)

Vendor : Amazon

Version : DEMO

QUESTION NO: 1
ある企業は、Amazon
EC2インスタンス上でモノリシックアーキテクチャのeコマースプラットフォームを運用しています。このプラットフォームでは、WebサービスとAPIサービスが稼働しています。同社は、アーキテクチャを分離し、スケーラビリティを向上したいと考えています。
同社はまた、注文を追跡し、失敗した注文を再処理する機能も求めています。
これらの要件を満たすソリューションはどれでしょうか?
A. 注文をAmazon Simple Queue Service (Amazon SQS)
キューに送信します。キューを消費して注文を処理するようにAWS
Lambda関数を設定します。SQSデッドレターキューを実装します。
B. Amazon Simple Queue Service (Amazon SQS)
キューに注文を送信します。キューを使用するように Amazon Elastic Container Service (Amazon ECS) タスクを設定します。SQS の可視性タイムアウトを実装します。
C. Amazon Kinesis Data Streams を使用して注文をキューに入れます。AWS Lambda
関数を使用してデータストリームを消費します。Amazon S3
を設定して、失敗した注文を追跡し、再処理します。
D. 注文をAmazon Simple Queue Service (Amazon SQS)
キューに送信します。キューを消費して注文を処理するようにAWS
Lambda関数を設定します。SQSロングポーリングを使用するようにLambda関数を設定します。
Answer: A
Explanation:
* To decouple the monolith and enhance scalability, AWS best practice is to introduce an asynchronous message queue, such as Amazon SQS, between the web/API tier and the order-processing logic.
* AWS Lambda functions consuming from the SQS queue provide serverless, auto-scaling processing without managing servers.
* To track and reprocess failed orders, SQS supports dead-letter queues (DLQs). Messages that cannot be processed successfully after a configurable number of attempts are automatically moved to the DLQ, where operations teams or automated processes can inspect and reprocess them.
Why others are not correct:
* B: ECS tasks can consume an SQS queue, but this requires managing container infrastructure and does not inherently provide as simple reprocessing/visibility as combining Lambda with a DLQ. Visibility timeout is not a tracking or archival mechanism.
* C: Kinesis is a streaming service designed for ordered event streams, not primarily for order-queue semantics and DLQs; SQS is simpler and purpose-built for this pattern.
* D: Long polling reduces empty responses and API calls but does nothing for tracking or reprocessing failed messages; without a DLQ, failed orders are harder to manage

QUESTION NO: 2
ある企業では、単一のAmazon
EC2インスタンス上で動作するアプリケーションを運用しています。このアプリケーションは、同じEC2インスタンス上で動作するMySQLデータベースを使用しています。この企業は、増加するトラフィックに対応するために、高可用性と自動拡張性を備えたソリューショ

ンを必要としています。
これらの要件を満たすソリューションはどれでしょうか?
A. アプリケーションを、Application Load Balancer の背後にある Auto Scaling グループで実行される EC2 インスタンスにデプロイします。複数の MySQL 互換ノードを持つ Amazon Redshift クラスターを作成します。
B. アプリケーションを、Application Load Balancer の背後にあるターゲットグループとして設定された EC2 インスタンスにデプロイします。複数のインスタンスを持つ Amazon RDS for MySQL クラスターを作成します。
C. アプリケーションを、Application Load Balancer の背後にある Auto Scaling グループで実行される EC2 インスタンスにデプロイします。データベース層用に Amazon Aurora Serverless MySQL クラスターを作成します。
D. アプリケーションを、Application Load Balancer の背後にあるターゲットグループとして設定された EC2 インスタンスにデプロイします。MySQL コネクタを使用する Amazon ElastiCache (Redis OSS) クラスターを作成します。
Answer: C
Explanation:
Amazon Aurora Serverless is a fully managed, MySQL-compatible database that automatically scales based on demand and provides high availability. Combining this with EC2 Auto Scaling and an Application Load Balancer achieves both application and database high availability and scalability.
Reference Extract:
"Aurora Serverless automatically starts up, shuts down, and scales capacity based on your application's needs, providing a cost-effective, highly available database solution." Source: AWS Certified Solutions Architect - Official Study Guide, Aurora Serverless and Scaling section.

QUESTION NO: 3
ソリューションアーキテクトは、ある企業向けにマルチリージョン災害復旧（DR）戦略を設計しています。この企業は、Application Load Balancer（ALB）の背後にあるAuto Scalingグループ内のAmazon EC2インスタンス上でアプリケーションを実行しています。このアプリケーションは、社内のプライマリおよびセカンダリAWSリージョンでホストされています。
プライマリリージョンに障害が発生した場合、アプリケーションはセカンダリリージョンからのDNSクエリに応答する必要があります。トラフィックを処理できるのは一度に1つのリージョンのみです。
これらの要件を満たすソリューションはどれでしょうか?
A. Amazon Route 53 Resolver にアウトバウンドエンドポイントを作成します。ネットワーク上の DNS リゾルバーにクエリを転送する方法を決定する転送ルールを作成します。ルールを各リージョンの VPC に関連付けます。
B. Amazon Route 53 にプライマリおよびセカンダリ DNS レコードを作成します。ヘルスチェックとフェイルオーバールーティングポリシーを設定します。

C. Amazon Route 53
でトラフィックポリシーを作成します。地理位置情報ルーティングポリシーと ELB Application Load Balancer の値タイプを使用します。
D. Amazon Route 53 プロファイルを作成します。DNS
リソースをプロファイルに関連付けます。プロファイルを各リージョンの VPC に関連付けます。
Answer: B
Explanation:
Amazon Route 53 supports failover routing policies, which use health checks to route DNS queries to a secondary Region only if the primary endpoint fails. This design ensures only one Region is active for traffic at any given time. This is the recommended architecture for active-passive, multi-Region DR strategies.
AWS Documentation Extract:
"Failover routing lets you route traffic to a primary resource, such as a web server in one Region, and a secondary resource in another Region. If the primary fails, Route 53 can route traffic to the secondary resource automatically." (Source: Amazon Route 53 documentation, Routing Policy Types) A, D: These options do not configure DNS failover for external users.
C: Geolocation routing is for regional distribution, not DR failover.
Reference: AWS Certified Solutions Architect - Official Study Guide, Multi-Region DR and Route 53.

QUESTION NO: 4
ある企業がAmazon Aurora
MySQLを使用してアプリケーションを開発しています。チームは、本番環境に影響を与えずに新機能をテストするために、頻繁にスキーマ変更を行います。テスト後は、ダウンタイムを最小限に抑えながら、変更内容を本番環境に反映させる必要があります。
これらの要件を満たすソリューションはどれですか?
A. 既存のクラスターをベースにステージング Aurora
クラスターを作成します。ステージングクラスターでスキーマの変更をテストします。
B. 読み取りレプリカを作成し、そのスキーマを変更して、プライマリに昇格します。
C. Aurora
MySQLのBlue/Greenデプロイメントを作成します。ステージング環境でスキーマを変更し、テスト後にトラフィックを切り替えます。
D. Aurora データベースを DynamoDB
に複製し、スキーマの変更を適用し、アプリケーションを DynamoDB に切り替えます。
Answer: C
Explanation:
Aurora blue/green deployments are specifically designed for safe schema changes, zero-downtime updates, and production isolation.
The staging (green) environment can receive schema changes without affecting production (blue). After validation, you perform a fast, minimally disruptive switchover that updates production.
Read replicas (Option B) do not allow schema changes. Creating an independent staging cluster (Option A) does not provide automated, low-downtime cutover. DynamoDB (Option D) is not compatible with MySQL schemas.

QUESTION NO: 5
ある企業には、発注書の受信と処理を行うアプリケーションがあります。このアプリケーションはXMLデータのみをサポートしています。JSON形式での注文を受け付けるようにアプリケーションを設定する必要がありますが、アプリケーションに変更を加えるつもりはありません。
ソリューションアーキテクトは、Amazon API Gateway HTTP API
を使用して新しい注文APIを作成しています。ソリューションアーキテクトは、アプリケーションのDNSレコードを新しいHTTP APIを指すように変更する必要があります。
A.
HTTPプロキシ統合を使用して、XMLリクエストをアプリケーションに渡します。JSONリクエストの場合は、API
Gatewayマッピングを使用して注文書をXMLに変換します。アプリケーションを呼び出すには、API Gatewayに統合されたAWS Lambda関数を使用します。
B.
HTTPプロキシ統合を使用して、XMLリクエストをアプリケーションに渡します。JSONリクエストの場合は、API Gatewayに統合されたAWS
Lambda関数を使用して、注文書をJSONからXMLに変換し、アプリケーションを呼び出します。
C.
HTTPカスタム統合を使用して、XMLリクエストをアプリケーションに渡します。JSONリクエストの場合は、API
Gatewayマッピングを使用して注文書をXMLに変換します。アプリケーションを呼び出すには、API Gatewayに統合されたAWS Lambda関数を使用します。
D.
HTTPカスタム統合を使用して、XMLリクエストをアプリケーションに渡します。JSONリクエストの場合は、API Gatewayに統合されたAWS
Lambda関数を使用して、注文書をJSONに変換し、アプリケーションを呼び出します。
Answer: B
Explanation:
Why Option B is Correct:
HTTP Proxy Integration: Passes XML requests directly to the application, which already supports XML.
JSON Conversion: An AWS Lambda function converts JSON requests to XML and calls the application.
API Gateway: Acts as a front end to handle JSON requests and integrates seamlessly with Lambda for the transformation process.
Why Other Options Are Not Ideal:
Option A: Suggests using API Gateway mappings to convert JSON to XML. API Gateway mapping templates are limited in functionality and are not ideal for complex transformations.
Option C and D: Use HTTP custom integration unnecessarily, which adds complexity without additional benefits.
AWS References:
Amazon API Gateway Integration:AWS Documentation - API Gateway Integration AWS Lambda:AWS Documentation - Lambda

QUESTION NO: 6
ある企業が、AWS Lambda関数をAmazon RDS for MySQL
DBインスタンスに接続するアプリケーションを設計しています。このDBインスタンスは多数の接続を管理しており、接続性とリカバリ性を向上させるためにアプリケーションを修正する必要があります。
最も少ない運用オーバーヘッドでこれらの要件を満たすソリューションはどれでしょうか?
A. 接続プールに Amazon RDS Proxy を使用します。DB インスタンスへの接続に RDS Proxy を使用するようにアプリケーションを変更します。
B.
接続プール用の新しいRDSインスタンスを作成します。アプリケーションを変更して、接続に新しいRDSインスタンスを使用するようにします。
C.
DBインスタンスの負荷を分散するためにリードレプリカを作成します。リードレプリカ間で負荷を分散するためにネットワークロードバランサーを作成します。
D. RDS for MySQL DB インスタンスを Amazon Aurora MySQL に移行して、DB インスタンスのパフォーマンスを向上させます。
Answer: A
Explanation:
Amazon RDS Proxy helps manage thousands of concurrent database connections by pooling and reusing them efficiently. It is especially useful for serverless applications like AWS Lambda that can open numerous connections quickly, potentially overwhelming the database. Using RDS Proxy reduces connection management overhead and improves fault tolerance.
Reference: AWS Documentation - Amazon RDS Proxy

QUESTION NO: 7
ある企業が、開発者チームのためにAWS上に開発環境を構築しています。チームはプロジェクトデータを保存するために複数のAmazon
S3バケットにアクセスする必要があります。また、開発インスタンスを実行するためにAmazon EC2を使用する必要があります。
同社は、開発者が特定のAmazon
S3バケットとEC2インスタンスにのみアクセスできるようにする必要があります。アクセス権限は、チーム内の各開発者の役割に応じて割り当てる必要があります。同社は、永続的な認証情報の使用を最小限に抑え、最小権限の原則に従ってアクセスを安全に管理したいと考えています。
これらの要件を満たすソリューションはどれでしょうか?
A. Amazon S3 および Amazon EC2 に対する管理者レベルの権限を持つ IAM ロールを作成します。開発者が Amazon S3 および Amazon EC2 にアクセスするには、Amazon Cognito を使用してサインインする必要があります。
B. Amazon S3 および Amazon EC2 に対するきめ細かな権限を持つ IAM ロールを作成します。開発者の認証情報を管理するために AWS IAM Identity Center を設定します。
C. Amazon S3 および Amazon EC2 へのプログラム的なアクセス権を持つ IAM ユーザーを作成します。開発者ごとに Amazon S3 および Amazon EC2

にアクセスするための個別のアクセスキーを生成します。
D. Amazon S3 用の VPC エンドポイントを作成します。開発者は、要塞ホストを介して Amazon EC2 インスタンスと Amazon S3 バケットにアクセスする必要があります。
Answer: B
Explanation:
The most secure and manageable way to provide developers with temporary, least-privilege access is by using AWS IAM Identity Center (formerly AWS SSO). IAM Identity Center allows assigning IAM roles with scoped permissions based on the developer's team role. This ensures no permanent credentials are required and minimizes risk.
Option B enables role-based access with centralized identity and access management, making it the most secure and scalable solution for managing developer permissions.

QUESTION NO: 8
ソリューションアーキテクトは、Amazon API Gateway REST API を保護する必要があります。ユーザーは、一般的な外部ソーシャルアイデンティティプロバイダー (IdP) を使用して API にログインできる必要があります。ソーシャル IdP は、SAML や OpenID Connect (OIDC) などの標準認証プロトコルを使用する必要があります。ソリューションアーキテクトは、アプリケーションの脆弱性を悪用する攻撃から API を保護する必要があります。
これらのセキュリティ要件を満たす手順の組み合わせはどれですか? (2 つ選択してください)。
A. REST API に関連付けられた AWS WAF ウェブ ACL を作成します。適切なマネージドルールを ACL に追加します。
B. AWS Shield Advanced をサブスクライブします。DDoS 保護を有効にします。Shield Advanced を REST API に関連付けます。
C. ソーシャル IdP とのフェデレーションを持つ Amazon Cognito ユーザープールを作成します。ユーザープールを REST API と統合します。
D. API GatewayでAPIキーを作成します。APIキーをREST APIに関連付けます。
E. AWS WAF でソーシャル IdP のみを許可する IP アドレスフィルターを作成します。フィルターをウェブ ACL と API に関連付けます。
Answer: A C
Explanation:
Step A:AWS WAF with managed rules protects the API against application-layer attacks, such as SQL injection and cross-site scripting (XSS).
Step C:Amazon Cognito provides secure authentication and supports federation with social IdPs using OIDC or SAML. It integrates seamlessly with API Gateway.
Option B:AWS Shield Advanced provides DDoS protection, which is not explicitly required in this scenario.
Option D:API keys provide identification, not authentication, and are insufficient for this use case.
Option E:IP filters in WAF are overly restrictive for federated authentication scenarios.
AWS Documentation References:
Amazon Cognito Federation
AWS WAF Managed Rules

QUESTION NO: 9
ある小売企業がAWS上でアプリケーションを運用しています。このアプリケーションでは、ウェブサーバーとしてAmazon EC2、データベースサービスとしてAmazon RDS、そしてグローバルコンテンツ配信としてAmazon CloudFrontを使用しています。
同社には DDoS 攻撃を軽減するソリューションが必要です。
どのソリューションがこの要件を満たすでしょうか?
A. クエリリクエストの長さを制限するAWS WAFカスタムルールを実装します。CloudFrontをAWS WAFと連携するように設定します。
B. AWS Shield Advanced を有効にします。CloudFront を Shield Advanced と連携するように設定します。
C. Amazon Inspector を使用して EC2 インスタンスをスキャンします。Amazon GuardDuty を有効にします。
D. Amazon Macie を有効にします。CloudFront Origin Shield を設定します。
Answer: B
Explanation:
AWS Shield Advanced provides advanced DDoS protection for AWS workloads, including EC2, CloudFront, and RDS. When integrated with CloudFront, Shield Advanced offers comprehensive detection and mitigation against large and sophisticated DDoS attacks, along with 24x7 access to the AWS DDoS Response Team (DRT). AWS WAF provides application-level protection, but for complete DDoS mitigation, Shield Advanced is the recommended solution.
Reference Extract from AWS Documentation / Study Guide:
"AWS Shield Advanced provides expanded DDoS attack protection for applications running on AWS. It offers always-on detection and automatic inline mitigations that minimize application downtime and latency." Source: AWS Certified Solutions Architect - Official Study Guide, Security and DDoS Protection section.

QUESTION NO: 10
ある企業が、毎日実行しているMicrosoft Windowsのバッチジョブをオンプレミス環境からAWSに移行しています。現在のバッチジョブは最大1時間実行されています。同社は、クラウド環境向けにバッチジョブのプロセスを最新化したいと考えています。
最も少ない運用オーバーヘッドでこれらの要件を満たすソリューションはどれでしょうか?
A. Windows バッチジョブ処理を処理するために、Auto Scaling グループに Amazon EC2 インスタンスのフリートを作成します。
B. Windows バッチジョブを処理するための AWS Lambda 関数を実装します。Amazon EventBridge ルールを使用して Lambda 関数を呼び出します。
C. AWS Fargate を使用して Windows バッチジョブをコンテナとしてデプロイします。AWS Batch を使用してバッチジョブの処理を管理します。
D. Amazon EC2 インスタンスで Amazon Elastic Kubernetes Service (Amazon EKS) を使用して、バッチジョブ処理用の Windows コンテナをオーケストレーションします。
Answer: C
Explanation:

AWS Batch supports Windows-based jobs and automates provisioning and scaling of compute environments.
Paired with AWS Fargate, it removes the need to manage infrastructure. This solution requires the least operational overhead and is cloud-native, providing flexibility and scalability.
Reference: AWS Documentation - AWS Batch with Fargate for Windows Workloads

QUESTION NO: 11
質問：
機械学習（ML）チームは、Amazon S3
バケット内のデータを利用するアプリケーションを構築しています。MLチームは、AWS上でのモデルトレーニングワークフロー用のストレージソリューションを必要としています。トレーニングデータセットへの頻繁なアクセスをサポートする高性能ストレージが必要です。このストレージソリューションは、Amazon S3
とネイティブに統合できる必要があります。これらの要件を満たし、運用オーバーヘッドが最も少ないソリューションはどれでしょうか？
オプション:
A. Amazon Elastic Block Store (Amazon EBS)
ボリュームを使用して、高性能ストレージを実現します。AWS DataSync を使用して、S3 バケットから EBS ボリュームにデータを移行します。
B. Amazon EC2
MLインスタンスを使用して、高性能ストレージを提供します。トレーニングデータはAmazon EBSボリュームに保存します。S3 Copy
APIを使用して、S3バケットからEBSボリュームにデータをコピーします。
C. 高性能ストレージを提供するには、Amazon FSx for Lustre
を使用します。トレーニングデータセットは Amazon S3 標準ストレージに保存します。
D. 高性能ストレージを提供するために Amazon EMR
を使用します。トレーニングデータセットは Amazon S3 Glacier Instant Retrieval ストレージに保存します。
Answer: C
Explanation:
Amazon FSx for Lustre is a high-performance file system optimized for fast processing of workloads such as machine learning, high-performance computing (HPC), and video processing. It integrates natively with Amazon S3, allowing you to:
Access S3 Data: FSx for Lustre can be linked to an S3 bucket, presenting S3 objects as files in the file system.
High Performance: It provides sub-millisecond latencies, high throughput, and millions of IOPS, which are ideal for ML workloads.Amazon Web Services, Inc.
Minimal Operational Overhead: Being a fully managed service, it reduces the complexity of setting up and managing high-performance file systems.
References:
Amazon FSx for Lustre - High-Performance File System Integrated with S3Amazon Web Services, Inc.
What is Amazon FSx for Lustre?

QUESTION NO: 12
ある企業は、Amazon
S3バケットを使用してバックアップデータをアーカイブする予定です。規制により、バックアップデータは7年間保持する必要があります。
保存期間中、企業は管理者を含むユーザーによるデータの削除を防止する必要があります。
当社は7年後にデータを削除することができます。
これらの要件を満たすソリューションはどれでしょうか?
A.
7年間の削除操作を拒否するS3バケットポリシーを作成します。7年後にデータを削除するS3ライフサイクルポリシーを作成します。
B. ガバナンスモードでデータを 7 年間保持する S3
オブジェクトロックのデフォルトの保持ポリシーを作成します。
7 年後にデータを削除する S3 ライフサイクル ポリシーを作成します。
C. コンプライアンスモードでデータを 7 年間保持する S3
オブジェクトロックのデフォルトの保持ポリシーを作成します。
7 年後にデータを削除する S3 ライフサイクル ポリシーを作成します。
D. S3
バッチオペレーションジョブを作成し、各オブジェクトに7年間のリーガルホールドを設定します。7年後にデータを削除するS3ライフサイクルポリシーを作成します。
Answer: C
Explanation:
Comprehensive and Detailed Step-by-Step Explanation:
The requirement is toprevent data deletion by any user, including administrators, for 7 years while allowing automatic deletion afterward.
S3 Object Lock in Compliance Mode (Correct Choice - C)
Compliance mode ensures that even the root user cannot delete or modify the objects during the retention period.
After 7 years, the S3 Lifecycle policy automatically deletes the objects.
This meets bothimmutability and automatic deletionrequirements.
Governance Mode (Option B - Incorrect)
Governance mode prevents deletion,but administrators can override it.
The requirement explicitly states thateven administrators must not be able to delete the data.
S3 Bucket Policy (Option A - Incorrect)
An S3 bucket policy candeny deletes, but policies can be modified at any time by administrators.
It does not enforce strict retention like Object Lock.
S3 Batch Operations Job (Option D - Incorrect)
A legal hold does not have an automatic expiration.
Legal holds must be manually removed, which is not efficient.
Why Option C is Correct:
S3 Object Lock in Compliance Mode prevents deletion by all users, including administrators.
The S3 Lifecycle policy deletes the data automatically after 7 years, reducing operational overhead.
References:
S3 Object Lock Compliance Mode

S3 Lifecycle Policies

QUESTION NO: 13
ソリューションアーキテクトは、Amazon EMR
クラスター上で実行される大規模なデータ分析ジョブを最適化する必要があります。このジョブは完了までに 13
時間かかります。クラスターには、コンピューティングに最適化された大規模なインスタンス上にデプロイされた複数のコアノードとワーカーノードがあります。
EMRログを確認した結果、ジョブの実行中に複数のノードが5時間以上アイドル状態になっていることが判明しました。ソリューションアーキテクトは、クラスターのパフォーマンスを最適化する必要があります。
この要件を最もコスト効率よく満たすソリューションはどれでしょうか?
A.
コアノードの数を増やして、アイドル時間なしで分析ジョブを処理するのに十分な処理能力を確保します。
B. EMR
マネージドスケーリング機能を使用して、ワークロードに基づいてクラスターのサイズを自動的に変更します。
C. 分析ジョブをAWS
Lambda関数セットに移行します。関数の同時実行予約を設定します。
D.
分析ジョブのコアノードをメモリ最適化インスタンスタイプに移行して、ジョブの合計実行時間を短縮します。
Answer: B
Explanation:
EMR managed scaling dynamically resizes the cluster by adding or removing nodes based on the workload.
This feature helps minimize idle time and reduces costs by scaling the cluster to meet processing demands efficiently.
Option A:Increasing the number of core nodes might increase idle time further, as it does not address the root cause of underutilization.
Option C:Migrating the job to Lambda is infeasible for large analytics jobs due to resource and runtime constraints.
Option D:Changing to memory-optimized instances may not necessarily reduce idle time or optimize costs.
AWS Documentation References:
EMR Managed Scaling

QUESTION NO: 14
ある企業は、AWS DataSync
を使用して、オンプレミスシステムからAWSへ数百万件のファイルを移行しています。ファイルのサイズは平均10KBです。
同社はファイルストレージとしてAmazon
S3を利用したいと考えています。移行後1年間は、ファイルは1～2回アクセスされるだけで、すぐに利用可能である必要があります。1年後は、ファイルは少なくとも1年間アーカイブ

する必要があります。
7年。
これらの要件を最もコスト効率よく満たすソリューションはどれでしょうか?
A.
アーカイブツールを使用してファイルを大きなオブジェクトにグループ化します。DataSyncを使用してオブジェクトを移行します。最初の1年間は、オブジェクトをS3 Glacier Instant Retrievalに保存します。ライフサイクル設定を使用して、1年後にファイルをS3 Glacier Deep Archiveに移行し、7年間の保持期間を設定します。
B.
アーカイブツールを使用してファイルをラージオブジェクトにグループ化します。DataSyncを使用してオブジェクトをS3 Standard-Infrequent Access (S3 Standard-IA) にコピーします。ライフサイクル設定を使用して、1年後にファイルをS3 Glacier Instant Retrievalに移行し、7年間の保持期間を設定します。
C. ファイルの保存先ストレージクラスを S3 Glacier Instant に設定します。取得ライフサイクルポリシーを使用して、1 年後にファイルを S3 Glacier Flexible Retrieval に移行し、保持期間は 7 年に設定します。
D. DataSync タスクを設定して、ファイルを S3 標準-低頻度アクセス (S3 標準-IA) に転送します。ライフサイクル設定を使用して、ファイルを S3 に移行します。1 年後に Deep Archive を実行し、保持期間は 7 年です。
Answer: A

QUESTION NO: 15
ある企業は Amazon Elastic Kubernetes Service (Amazon EKS) クラスターを使用しています。この企業では、サービスアカウント向け IAM ロール (IRSA) を使用して、EKS クラスター内の Kubernetes サービスアカウントが特定の AWS リソースに安全かつきめ細やかにアクセスできるようにする必要があります。
これらの要件を満たすソリューションの組み合わせはどれですか? (2 つ選択してください)
A.
必要な権限を定義するIAMポリシーを作成します。このポリシーをEKSノードのIAMロールに直接アタッチします。
B. EKS クラスター内にネットワークポリシーを実装して、Kubernetes サービスアカウントが特定の AWS サービスにアクセスできないようにします。
C. EKS クラスターの IAM ロールを変更し、各 Kubernetes サービスアカウントの権限を追加します。IAM ロールと Kubernetes ロールが 1 対 1 でマッピングされていることを確認します。
D. 必要な権限を含む IAM ロールを定義します。Kubernetes サービスアカウントに、IAM ロールの Amazon リソースネーム (ARN) をアノテーションとして付与します。
E. サービス アカウントの IAM ロールと OpenID Connect (OIDC) ID プロバイダーの間に信頼関係を設定します。
Answer: D E
Explanation:
IAM Roles for Service Accounts (IRSA): IRSA allows you to associate an IAM role with a Kubernetes service account. This enables pods to assume the IAM role and access AWS resources securely.

Annotating Service Accounts: By annotating Kubernetes service accounts with the ARN of the IAM role, you establish the association required for IRSA.
OIDC Identity Provider: EKS clusters use OpenID Connect (OIDC) to authenticate service accounts. Setting up a trust relationship between the IAM role and the OIDC provider allows the Kubernetes service account to assume the IAM role.

QUESTION NO: 16
ある企業は、世界中の20,000以上の小売店に展開されているクライアントにサービスを提供するアプリケーションを保有しています。このアプリケーションは、ポート443でHTTPS経由で公開されるバックエンドWebサービスで構成されています。アプリケーションは、Application Load Balancer (ALB) の背後にあるAmazon
EC2インスタンスでホストされています。
小売店はパブリックインターネットを介してウェブアプリケーションと通信します。当社は、各小売店が地域ISPから割り当てられたIPアドレスを登録することを許可しています。
同社のセキュリティ チームは、小売店によって登録された IP
アドレスのみにアクセスを制限することで、アプリケーション
エンドポイントのセキュリティを強化することを推奨しています。
これらの要件を満たすためにソリューション アーキテクトは何をすべきでしょうか?
A. AWS WAF Web
ACLをALBに関連付けるALBのIPルールセットを使用してトラフィックをフィルタリングするルール内のIPアドレスを更新して、登録されたIPアドレスを含める
B. ALB を管理するために AWS Firewall Manager をデプロイします。ALB
へのトラフィックを制限するファイアウォールルールを設定します。登録された IP
アドレスを含めるようにファイアウォールルールを変更します。
C. IPアドレスをAmazon DynamoDBテーブルに保存します。ALBにAWS
Lambda認可関数を設定し、受信リクエストが登録済みのIPアドレスからのものであることを検証します。
D. ALB のパブリック インターフェイスを含むサブネット上のネットワーク ACL
を構成します。登録された各 IP アドレスのエントリを使用して、ネットワーク ACL
の入力ルールを更新します。
Answer: A
Explanation:
AWS WAF (Web Application Firewall): AWS WAF allows you to create custom rules to block or allow web requests based on conditions that you specify.
Web ACL (Access Control List):
Create a web ACL and associate it with the ALB.
Use IP rule sets to specify the IP addresses of the retail locations that are allowed to access the application.
Security and Flexibility:
AWS WAF provides a scalable way to manage access control, ensuring that only traffic from registered IP addresses is allowed.
You can dynamically update the IP rule sets to add or remove IP addresses as needed.
Operational Simplicity: Using AWS WAF with a web ACL is straightforward and integrates seamlessly with the ALB, providing an efficient solution for managing access control based on IP addresses.

References:
AWS WAF
How AWS WAF Works

QUESTION NO: 17
ある企業は、Amazon RDS for PostgreSQL
データベースへのアクセスを許可するための安全なソリューションを設計しています。
Amazon EC2
インスタンスで実行されるアプリケーションは、長期的な認証情報を保存せずに、データベースに対して安全に認証できる必要があります。
これらの要件を満たすソリューションはどれでしょうか?
A. RDS IAM 認証を有効にし、データベース認証情報を保存するように AWS Secrets Manager を設定します。
実行時に資格情報を取得するようにアプリケーションを構成します。
B.
データベースにカスタムIAMポリシーを設定し、EC2インスタンスのIPアドレスからのアクセスを許可します。アプリケーションがデータベースへの認証に静的パスワードを使用するように設定します。
C.
アプリケーションごとにIAMユーザーを設定します。アクセスキーIDとシークレットアクセスキーをEC2インスタンスの環境変数に保存します。IAMユーザーにデータベースへの権限を付与します。
D.
IAMロールを使用してEC2インスタンスに権限を割り当てます。アプリケーションがRDSデータベースからトークンを取得し、IAM認証を使用して認証するように設定してください。
Answer: D
Explanation:
For Amazon RDS for PostgreSQL, AWS provides IAM database authentication. With this feature, applications do not use stored long-term usernames and passwords. Instead, they use temporary authentication tokens that are generated by AWS and validated by the RDS database.
The AWS best practice pattern is:
* Attach an IAM role to the EC2 instances (instance profile).
* Grant that role the necessary permissions (for example, rds-db:connect) to the specific RDS database user.
* The application running on the EC2 instance uses the role's temporary credentials to call the RDS token-generation API and obtain a short-lived authentication token.
* The application then uses this token as the password when connecting to RDS for PostgreSQL.
This removes the need to store long-term credentials in the application or on the instance and uses IAM roles with temporary credentials, aligning with the security requirement.
Option A still relies on stored credentials (even if in Secrets Manager), which are long-lived and rotated but not token-based per-connection IAM authentication.
Option B uses static passwords and IP-based access, which does not meet the "no long-term credentials" requirement.

Option C stores long-term IAM user keys on the instances, which is explicitly against best practices and does not directly integrate with RDS authentication.

QUESTION NO: 18
DevOpsチームのリーダーメンバーがAWSアカウントを作成します。DevOpsエンジニアは、パスワードマネージャーアプリケーションを通じて、ソリューションアーキテクトとアカウント認証情報を共有します。
ソリューション アーキテクトは、新しいアカウントのルート
ユーザーを保護する必要があります。
この要件を満たすアクションはどれですか? (2 つ選択してください)。
A. ルート ユーザーのパスワードを新しい強力なパスワードに更新します。
B. 仮想多要素認証 (MFA) デバイスを使用して、ルート ユーザー アカウントを保護します。
C. DevOps チームの各メンバーに IAM ユーザーを作成します。各 IAM ユーザーに AWS 管理ポリシー「AdministratorAccess」を割り当てます。
D. ルートユーザーアクセスキーを作成します。キーをAWS Systems Managerパラメータストアに新しいパラメータとして保存します。
E. ルートユーザーが承認されたサービスのみを使用できるように、ルートユーザーの IAM ロールを更新します。
Answer: A B
Explanation:
Securing the root user account requires setting a strong password and enabling multi-factor authentication (MFA). AWS recommends never sharing the root user credentials, setting up individual IAM users for everyday operations, and always protecting the root user with MFA for maximum security.
Reference Extract:
"AWS recommends securing the root user with a strong password and enabling multi-factor authentication (MFA). Do not use or share root credentials for everyday tasks." Source: AWS Certified Solutions Architect - Official Study Guide, IAM and Security Best Practices section.

QUESTION NO: 19
ある企業が、オンラインゲームのリアルタイム分析をユーザーが操作できるサーバーレスWebアプリケーションを開発しています。ゲームからのデータはリアルタイムでストリーミング配信する必要があります。ユーザーデータ用のデータベースとして、耐久性が高く、レイテンシの低いものが必要です。このアプリケーションを利用するユーザー数は不明です。設計上の考慮事項としては、アプリケーションのスケールに応じて1桁ミリ秒の応答時間を実現する必要があります。
これらの要件を満たす AWS サービスの組み合わせはどれですか? (2 つ選択してください)。
A. Amazon CloudFront
B. Amazon DynamoDB
C. Amazon Kinesis
D. Amazon RDS
E. AWS グローバルアクセラレーター
Answer: B C
Explanation:

Amazon Kinesis allows real-time ingestion of game events at scale, while Amazon DynamoDB provides millisecond-latency access to user data, automatically scaling with demand. This combination ensures real- time processing and fast data retrieval without managing infrastructure.
Reference: AWS Documentation - Real-Time Processing with Kinesis and Low-Latency Databases with DynamoDB

QUESTION NO: 20
ある企業は、AWS Organizations 内の同一組織内の複数の AWS アカウントで複数のアプリケーションを実行しています。コンテンツ管理システム (CMS) は、VPC 内の Amazon EC2 インスタンスで実行されています。CMS は、別の AWS アカウントにデプロイされた Amazon Elastic File System (Amazon EFS) ファイルシステムの共有ファイルにアクセスする必要があります。EFS アカウントは別の VPC にあります。
どのソリューションがこの要件を満たすでしょうか?
A. EFS Elastic IP アドレスを使用して、EFS ファイルシステムを EC2 インスタンスにマウントします。
B.
2つのアカウント間でVPC共有を有効にします。EFSマウントヘルパーを使用して、ファイルシステムをEC2インスタンスにマウントします。EFSファイルシステムを共有サブネットに再デプロイします。
C. AWS Systems Manager Run Command を設定して、EFS ファイルシステムを EC2 インスタンスにマウントします。
D. EC2インスタンスにamazon-efs-utilsパッケージをインストールします。efs-configファイルにマウントターゲットを追加します。
EFS アクセス ポイントを使用して EFS ファイル システムをマウントします。
Answer: D
Explanation:
To access an EFS file system across accounts and VPCs, the EFS must be mounted using VPC peering or AWS Transit Gateway, and the EC2 instances must use the amazon-efs-utils package with the correct mount target or access point.
Using an EFS access point simplifies access management, especially across accounts, by providing a POSIX identity and access policy layer.
VPC sharing doesn't support EFS directly unless the subnet and resources are shared properly, which requires redeployment. Therefore, option D is the most complete and correct.

QUESTION NO: 21
ある企業は、重要なストレージアプリケーションをAWSクラウドで運用しています。このアプリケーションは、2つのAWSリージョンでAmazon S3を使用しています。この企業は、リモートユーザーのデータを、パブリックネットワークの混雑を避け、最も近いS3バケットに送信することをアプリケーションに求めています。また、Amazon S3の管理を最小限に抑えながら、アプリケーションのフェイルオーバーを実現したいと考えています。
これらの要件を満たすソリューションはどれでしょうか?

A.
2つのリージョン間でアクティブ/アクティブ設計を実装します。ユーザーに最も近いリージョンのS3エンドポイントを使用するようにアプリケーションを設定します。
B.
S3マルチリージョンアクセスポイントを使用したアクティブ/パッシブ構成を使用します。各リージョンにグローバルエンドポイントを作成します。
C.
ユーザーデータを、ユーザーに最も近いリージョンのS3エンドポイントに送信します。S3バケットの同期を維持するために、S3クロスアカウントレプリケーションルールを設定します。
D.
単一のグローバルエンドポイントを持つアクティブ/アクティブ構成でマルチリージョンアクセスポイントを使用するように Amazon S3 を設定します。S3 クロスリージョンレプリケーションを設定します。
Answer: D
Explanation:
AWS S3 Multi-Region Access Points enable customers to use a single global endpoint for S3 bucket access across multiple AWS Regions, providing automatic routing to the nearest Region. This reduces public network congestion by directing user data to the closest S3 bucket and supports high availability with active- active configuration.
Cross-Region Replication ensures data is replicated between buckets in different Regions, meeting the failover and resilience requirements with minimal management overhead.
Option D aligns best with AWS's recommended approach to resilient, low-latency, and simplified multi- Region S3 access.
Option A lacks the global endpoint and automatic failover. Option B incorrectly describes Multi-Region Access Points configuration and suggests global endpoints per Region, which is contradictory. Option C's cross-account replication adds complexity and does not provide a single global endpoint.
References:
AWS Well-Architected Framework - Reliability Pillar
(https://d1.awsstatic.com/whitepapers/architecture
/AWS_Well-Architected_Framework.pdf)
Amazon S3 Multi-Region Access Points
(https://docs.aws.amazon.com/AmazonS3/latest/userguide
/MultiRegionAccessPoints.html)
S3 Cross-Region Replication
(https://docs.aws.amazon.com/AmazonS3/latest/userguide/replication.html)

QUESTION NO: 22
ある会社は、TCP および UDP マルチプレイヤー ゲーム機能を備えたオンライン ゲーム アプリケーションを持っています。同社は、Amazon Route 53 を使用して、アプリケーション トラフィックを異なる AWS リージョンにある複数のネットワーク ロード バランサー (NLB) に向けます。同社は、ユーザーの増加に備えて、アプリケーションのパフォーマンスを向上させ、オンライン ゲームの遅延を短縮する必要があります。

これらの要件を満たすソリューションはどれですか?
A. NLB の前に Amazon CloudFront ディストリビューションを追加します。 Cache-Control: max-age パラメータを増やします。
B. NLB を Application Load Balancer (ALB) に置き換えます。レイテンシベースのルーティングを使用するように Route 53 を設定します。
C. NLB の前に AWS Global Accelerator を追加します。正しいリスナーポートを使用するように Global Accelerator エンドポイントを構成します。
D. 'NLB の背後に Amazon API Gateway エンドポイントを追加します。 API キャッシュを有効にします。さまざまなステージのメソッドキャッシュをオーバーライドします。
Answer: C
Explanation:
AWS Global Accelerator is designed to improve the availability and performance of applications with global users by using the AWS global network. It provides static anycast IP addresses and routes user traffic over the AWS edge network to the optimal AWS Region and endpoint based on health, geography, and routing policies. Global Accelerator supports both TCP and UDP traffic and can have Network Load Balancers as endpoints.
For latency-sensitive workloads such as multiplayer gaming, Global Accelerator reduces latency and jitter compared to internet-based routing and handles Regional failover quickly.
CloudFront (Option A) is optimized for HTTP/HTTPS content caching and is not appropriate for arbitrary TCP/UDP gaming traffic. Application Load Balancers (Option B) do not support UDP traffic. API Gateway (Option D) is for HTTP APIs and is not suitable for raw TCP/UDP game traffic.

QUESTION NO: 23
ある企業は、静的ウェブサイトをAmazon S3でホスティングしています。この企業は、ウェブページにお問い合わせフォームを追加したいと考えています。お問い合わせフォームには、ユーザーが名前、メールアドレス、電話番号、メッセージを入力するための動的なサーバーサイドコンポーネントが含まれます。同社は、毎月のサイト訪問数を100件未満と見込んでいます。顧客がフォームに入力すると、問い合わせフォームからメールで通知される必要があります。
これらの要件を最もコスト効率よく満たすソリューションはどれでしょうか?
A. Amazon Elastic Container Service (Amazon ECS) で動的お問い合わせフォームをホストします。サードパーティのメールプロバイダーに接続するために、Amazon Simple Email Service (Amazon SES) を設定します。
B. AWS Lambda 関数から問い合わせフォームを返す Amazon API Gateway エンドポイントを作成します。
API Gateway で別の Lambda 関数を設定して、Amazon Simple Notification Service (Amazon SNS) トピックにメッセージを公開します。
C. 静的コンテンツと動的コンテンツの両方をAWS Amplify Hostingでホスティングし、ウェブサイトをホストします。サーバーサイドスクリプトを使用してお問い合わせフォームを構築します。Amazon Simple Queue Service (Amazon SQS) を設定して、メッセージを企業に配信します。

D. ウェブサイトを Amazon S3 から Windows Server を実行する Amazon EC2 インスタンスに移行します。Windows Server 用のインターネット インフォメーション サービス (IIS)
を使用してウェブページをホストします。クライアント側スクリプトを使用してお問い合わせフォームを構築します。フォームを Amazon WorkMail と統合します。

Answer: B

Explanation:
Using API Gateway and Lambda enables serverless handling of form submissions with minimal cost and infrastructure. When coupled with Amazon SNS, it allows instant email notifications without running servers, making it ideal for low-traffic workloads.
Reference: AWS Documentation - Serverless Contact Form with API Gateway, Lambda, and SNS

QUESTION NO: 24

ある企業は、オンプレミスのOracleリレーショナルデータベースにデータを保存しています。この企業は、Amazon Aurora PostgreSQLでデータを分析用に利用できるようにする必要があります。この企業は、AWS サイト間VPN接続を使用して、オンプレミスネットワークをAWSに接続しています。
企業は、Aurora PostgreSQL への移行中にソース
データベースに発生する変更をキャプチャする必要があります。
これらの要件を満たすソリューションはどれでしょうか?

A. AWS Schema Conversion Tool (AWS SCT) を使用して、Oracle スキーマを Aurora PostgreSQL スキーマに変換します。AWS Database Migration Service (AWS DMS) のフルロード移行タスクを使用してデータを移行します。

B. AWS DataSync を使用してデータを Amazon S3 バケットに移行します。Aurora PostgreSQL の aws_s3 拡張機能を使用して、S3 データを Aurora PostgreSQL にインポートします。

C. AWS Schema Conversion Tool (AWS SCT) を使用して、Oracle スキーマを Aurora PostgreSQL スキーマに変換します。AWS Database Migration Service (AWS DMS) を使用して、既存のデータを移行し、進行中の変更をレプリケートします。

D. AWS Snowball デバイスを使用してデータを Amazon S3 バケットに移行します。Aurora PostgreSQL の aws_s3 拡張機能を使用して、S3 データを Aurora PostgreSQL にインポートします。

Answer: C

Explanation:
For the migration of data from an on-premises Oracle database to Amazon Aurora PostgreSQL, this solution effectively handles schema conversion, data migration, and ongoing data replication.
AWS Schema Conversion Tool (SCT): SCT is used to convert the Oracle database schema to a format compatible with Aurora PostgreSQL. This tool automatically converts the database schema and code objects, like stored procedures, to the target database engine.
AWS Database Migration Service (DMS): DMS is employed to perform the data migration. It supports both full-load migrations (for initial data transfer) and continuous replication of ongoing changes (Change Data Capture, or CDC). This ensures that any updates to the Oracle database during the migration are captured and applied to the Aurora PostgreSQL

database, minimizing downtime.
Why Not Other Options?:
Option A (SCT + DMS full-load only): This option does not capture ongoing changes, which is crucial for a live database migration to ensure data consistency.
Option B (DataSync + S3): AWS DataSync is more suited for file transfers rather than database migrations, and it doesn't support ongoing change replication.
Option D (Snowball + S3): Snowball is typically used for large-scale data transfers that don't require continuous synchronization, making it less suitable for this scenario where ongoing changes must be captured.
AWS References:
AWS Schema Conversion Tool- Guidance on using SCT for database schema conversions.
AWS Database Migration Service- Detailed documentation on using DMS for data migrations and ongoing replication.

QUESTION NO: 25
AWS Organizations を使用する企業は、30 の異なる AWS アカウントで 150 のアプリケーションを実行しています。この企業は、AWS コストと使用状況レポートを使用して、管理アカウントに新しいレポートを作成しました。レポートは、データ収集アカウントのバケットに複製される Amazon S3 バケットに配信されます。
同社の上級管理職は、今月初めからの毎日の NAT ゲートウェイ コストを示すカスタム ダッシュボードを表示したいと考えています。
これらの要件を満たすソリューションはどれでしょうか?
A. リクエストされたテーブルビジュアルを含むAmazon QuickSightダッシュボードを共有します。AWS DataSyncを使用して新しいレポートをクエリするようにQuickSightを設定します。
B. リクエストされたテーブルビジュアルを含む Amazon QuickSight ダッシュボードを共有します。新しいレポートをクエリするために Amazon Athena を使用するように QuickSight を設定します。
C. リクエストされたテーブルビジュアルを含む Amazon CloudWatch ダッシュボードを共有する CloudWatch を設定して、AWS DataSync を使用して新しいレポートをクエリします。
D. リクエストされたテーブルビジュアルを含むAmazon CloudWatchダッシュボードを共有します。Amazon Athenaを使用して新しいレポートをクエリするようにCloudWatchを設定します。
Answer: B
Explanation:
The AWS Cost and Usage Report (CUR) delivers detailed, line-item billing data to Amazon S3. AWS recommends querying CUR with Amazon Athena by creating external tables over the CUR S3 location (partitioned by time) to produce daily cost aggregations such as NAT Gateway (EC2:NatGateway) usage and cost. Amazon QuickSight natively connects to Athena as a data source to build and share dashboards with visuals (tables, time series) filtered from the start of the current month. DataSync (A, C) is a file transfer service and cannot query data. CloudWatch dashboards (C, D) visualize metrics/logs, not CUR datasets. Therefore, using Athena to query CUR and QuickSight to present a daily NAT gateway cost

dashboard is the most direct and operationally efficient approach.
References: CUR - querying with Amazon Athena; QuickSight - Athena data source; Cos t categories and service/usage type fields for NAT Gateway; AWS Cost Management best practices.

QUESTION NO: 26
ある企業がAmazon Aurora PostgreSQLデータベースを使用した新しいウェブアプリケーションを立ち上げました。同社は、このアプリケーションにAIを活用した新機能を追加したいと考えています。AIツールを使用するには、ベクトルストレージ機能が必要です。
この要件を最もコスト効率よく満たすソリューションはどれでしょうか?
A. Amazon OpenSearch Service を使用して OpenSearch サービスを作成します。アプリケーションを設定して、ベクター埋め込みをベクターインデックスに書き込むようにします。
B. Amazon DocumentDB クラスターを作成します。ベクトル埋め込みをベクトルインデックスに書き込むようにアプリケーションを設定します。
C. Amazon Neptune ML クラスターを作成します。ベクトルグラフにベクトル埋め込みを書き込むようにアプリケーションを設定します。
D. Aurora PostgreSQLデータベースにpgvector拡張機能をインストールします。アプリケーションがベクター埋め込みをベクターテーブルに書き込むように設定します。
Answer: D
Explanation:
Aurora PostgreSQL supports the pgvector extension, which allows storage and querying of vector embeddings directly inside the database. This eliminates the need for external vector databases and provides cost-effective and performant integration for AI workloads.
Reference: AWS Documentation - Amazon Aurora PostgreSQL and pgvector Support

QUESTION NO: 27
ある企業は、AWS Lake Formation を使用して S3 データレイクを管理しています。S3 データと Aurora MySQL の運用データを結合し、QuickSight でデータを可視化したいと考えています。マーケティングチームは特定の列のみを表示する必要があります。
最も少ない運用オーバーヘッドで列レベルの承認を提供するソリューションはどれですか?
A. EMR を使用して、必要な列のみを含むデータベース データを SPICE に取り込みます。
B. AWS Glue Studio を使用してデータベースデータを S3 に取り込み、IAM ポリシーを使用して列を制御します。
C. AWS Glue Elastic Views を使用して、列制限のあるマテリアライズド S3 ビューを作成します。
D. Lake Formation ブループリントを使用して、データベースデータを S3 に取り込みます。列レベルのアクセス制御には Lake Formation を使用します。QuickSight データソースには Athena を使用します。
Answer: D

Explanation:
AWS Lake Formation provides fine-grained (column-level) access control for data stored in S3. Using a Lake Formation blueprint ensures database ingestion is automated and governed.
QuickSight can query Athena, and Athena honors Lake Formation permissions, enforcing column-level controls automatically.
Options A, B, and C rely on manual filtering or IAM policies, which cannot enforce column-level authorization for SQL queries.

QUESTION NO: 28
ある企業がAWS上のVPCでアプリケーションを実行しています。同社のオンプレミスデータセンターにはDNSサーバーが設置されています。
データセンターは、プライベート仮想インターフェース（VIF）を使用したAWS Direct Connect接続を介してAWSに接続されます。オンプレミスのDNSサーバーは、VPC内のアプリケーションのDNS名を解決する必要があります。
A. VPC に AWS Verified Access エンドポイントを設定します。Verified Access で DNS 転送ルールを設定します。オンプレミス DNS サーバーを設定して、Verified Access エンドポイント経由で DNS クエリを転送します。
B. オンプレミスの DNS サーバーと VPC 内のアプリケーション間の DNS 解決を有効にするために、Direct Connect 接続を設定します。
C. VPC 内に Amazon Route 53 Resolver のアウトバウンドエンドポイントと Resolver ルールを作成します。オンプレミス DNS サーバーを設定して、アプリケーションのリクエストをアウトバウンドエンドポイントに送信します。
D. VPC 内に Amazon Route 53 Resolver のインバウンドエンドポイントを作成します。オンプレミス DNS サーバーを設定して、アプリケーションのリクエストをインバウンドエンドポイントに送信します。
Answer: D
Explanation:
When on-premises DNS servers need to resolve private DNS names in a VPC, the correct pattern is to create a Route 53 Resolver inbound endpoint. The inbound endpoint allows DNS queries to flow from the on-premises environment into the VPC, where Route 53 can resolve VPC-specific names (such as private hosted zones or private resource records). Outbound endpoints (C) are for sending VPC DNS queries to on-premises, not the reverse. Verified Access (A) is unrelated to DNS resolution. Direct Connect (B) provides network connectivity but does not provide DNS forwarding capabilities. Therefore, option D is the correct design.
References:* Amazon Route 53 Resolver Developer Guide - Inbound and outbound endpoints* AWS Well- Architected Framework - Security Pillar: Hybrid DNS integration

QUESTION NO: 29
グローバルなeコマース企業がAWS上で3層アプリケーションを設計しています。このアプリケーションは、静的コンテンツを提供するウェブ層、ビジネスロジックを処理するアプリケーション層、そして商品情報とユーザーデータを保存するデータベース層で構成されています。このアプリケーションはリレーショナルデータベースと連携します。

同社では、運用オーバーヘッドを最小限に抑えながら、低遅延で世界中のユーザーにサービスを提供できる、可用性の高いアプリケーション アーキテクチャを必要としています。
これらの要件を満たすソリューションはどれでしょうか?
A. 単一のAWSリージョン内のアプリケーション層とウェブ層に、Auto Scalingグループ内のAmazon EC2インスタンスをデプロイします。Application Load Balancerを使用してウェブトラフィックを分散します。データベース層にはAmazon RDSデータベースとマルチAZ配置を使用します。
B. Amazon S3 バケットをオリジンとする Amazon CloudFront ディストリビューションを設定します。AWS Fargate 上の Amazon Elastic Container Service (Amazon ECS) コンテナを使用して、企業が事業を展開する各 AWS リージョンにアプリケーション層をデプロイします。データベース層には Amazon Aurora グローバルデータベースを使用します。
C. 静的ウェブコンテンツの保存にはAmazon S3バケットを使用します。アプリケーション層にはAmazon EC2 Auto ScalingとEC2スポットインスタンスを使用します。データベース層にはリードレプリカを備えたAmazon RDS for MySQLを使用します。AWS Database Migration Service (AWS DMS) を使用して、セカンダリAWSリージョンにデータを複製します。
D. 静的ウェブコンテンツの保存にはAmazon S3バケットを使用します。アプリケーション層では、AWS Lambda関数を使用してサーバーレスバックエンドロジックを処理します。ウェブリクエストに対してLambda関数を呼び出すには、Amazon API Gatewayを使用します。データベース層にはAmazon DynamoDBデータベースを使用します。DynamoDBデータベースを複数のAWSリージョンにデプロイします。
Answer: B
Explanation:
AWS guidance for global, highly available, low-latency applications using a relational database recommends:
* Amazon CloudFront with Amazon S3 for static content to cache data at edge locations globally and minimize latency for static assets.
* A regional application tier deployed close to users using managed container services such as Amazon ECS on AWS Fargate, which removes the need to manage servers and scales automatically, reducing operational overhead.
* A relational database tier using Amazon Aurora global database, which is purpose-built for globally distributed applications. Aurora global database provides a primary cluster in one Region with low- latency read replicas in secondary Regions and fast cross-Region replication, enabling low-latency reads and high availability for global users.
Option A uses only a single Region, which does not meet the "global users with low latency" requirement.
Option C uses RDS and DMS-based replication, which requires more management and does not provide Aurora's integrated global database features.
Option D replaces the relational database with DynamoDB, which violates the requirement that the application interacts with a relational database.

QUESTION NO: 30

ある企業はハイブリッド ネットワーク アーキテクチャを設計する必要があります。
現在、この企業のワークロードは AWS
クラウドとオンプレミスのデータセンターに保存されています。 ワークロードの通信には 1 桁台のレイテンシーが必要です。 この企業は複数の VPC を接続するために AWS Transit Gateway トランジットゲートウェイを使用しています。
どのステップの組み合わせが、これらの要件を最もコスト効率よく満たしますか? (2 つ選択してください。)

A. 各 VPC への AWS サイト間 VPN 接続を確立します。

B. AWS Direct Connect ゲートウェイを、VPC
に接続されているトランジットゲートウェイに関連付けます。

C. AWS Direct Connect ゲートウェイへの AWS サイト間 VPN 接続を確立します。

D. AWS Direct Connect 接続を確立します。Direct Connect
ゲートウェイへのトランジット仮想インターフェイス (VIF) を作成します。

E. AWS サイト間 VPN 接続を、VPC
にアタッチされているトランジットゲートウェイに関連付けます。

Answer: B D

Explanation:
AWS Direct Connect: Provides a dedicated network connection from your on-premises data center to AWS, ensuring low latency and consistent network performance.
Direct Connect Gateway Association:
Direct Connect Gateway: Acts as a global network transit hub to connect VPCs across different AWS regions.
Association with Transit Gateway: Enables communication between on-premises data centers and multiple VPCs connected to the transit gateway.
Transit Virtual Interface (VIF):
Create Transit VIF: To connect Direct Connect with a transit gateway.
Setup Steps:
Establish a Direct Connect connection.
Create a transit VIF to the Direct Connect gateway.
Associate the Direct Connect gateway with the transit gateway attached to the VPCs.
Cost Efficiency: This combination avoids the recurring costs and potential performance variability of VPN connections, providing a robust, low-latency hybrid network solution.
References:
AWS Direct Connect
Transit Gateway and Direct Connect Gateway

QUESTION NO: 31
ある企業は、高い IOPS を必要とする HPC ワークロードを実行しています。
これらの要件を満たす手順の組み合わせはどれですか? (2 つ選択してください)

A. Amazon EFS を高性能ファイルシステムとして使用します。

B. Amazon FSx for Lustre を高性能ファイルシステムとして使用します。

C. EC2インスタンスのAuto
Scalingグループを作成します。リザーブドインスタンスを使用します。スプレッドプレイスメントグループを設定します。分析にはAWS Batchを使用します。

D. Amazon S3 のマウントポイントを高性能ファイルシステムとして使用します。
E. EC2インスタンスのAuto Scalingグループを作成します。混合インスタンスタイプとクラスタープレイスメントグループを使用します。分析にはAmazon EMRを使用します。
Answer: B E
Explanation:
Option B: FSx for Lustre is designed for HPC workloads with high IOPS.
Option E: A cluster placement group ensures low-latency networking for HPC analytics workloads.
Option A: Amazon EFS is not optimized for HPC.
Option D: Mountpoint for S3 does not meet high IOPS needs.

QUESTION NO: 32
ある金融会社には、顧客向けの信用レポートを生成するウェブアプリケーションがあります。この会社は、ウェブアプリケーションのフロントエンドを、Application Load Balancer (ALB) に関連付けられた Amazon EC2 インスタンス群でホストしています。このアプリケーションは、Amazon RDS for SQL Server データベースに対してクエリを実行することでレポートを生成します。
同社は最近、世界中から悪意のあるトラフィックが不要なリクエストを送信することでアプリケーションを悪用していることを発見しました。この悪意のあるトラフィックは、膨大なコンピューティングリソースを消費しています。同社は、この悪意のあるトラフィックに対処する必要があります。
どのソリューションがこの要件を満たすでしょうか?
A. AWS WAF を使用してウェブ ACL を作成します。ウェブ ACL を ALB に関連付けます。ウェブ ACL を更新して、悪意のあるトラフィックに関連付けられた IP アドレスをブロックします。
B. AWS WAF を使用してウェブ ACL を作成します。ウェブ ACL を ALB に関連付けます。AWS WAF Bot Control マネージドルール機能を使用します。
C. ALB とデータベースを保護するために AWS Shield を設定します。
D. AWS WAF を使用してウェブ ACL を作成します。ウェブ ACL を ALB に関連付けます。AWS WAF IP レピュテーションルールを設定します。
Answer: B
Explanation:
TheAWS WAF Bot Control managed ruleis designed to automatically detect and mitigate bot traffic. This feature is particularly useful for addressing malicious traffic and conserving compute resources by filtering unnecessary requests at the ALB level.
Option A:Blocking IP addresses manually introduces significant operational overhead and is not scalable against dynamic, worldwide malicious traffic.
Option C:AWS Shield provides DDoS protection, but the scenario does not describe a DDoS attack. WAF is better suited for managing application-layer threats like bot traffic.
Option D:The AWS WAF IP reputation rule helps block traffic from known bad IPs but may not address bot traffic effectively.
AWS Documentation References:
AWS WAF Bot Control
AWS WAF Managed Rules

QUESTION NO: 33
ある企業が一部のアプリケーションをAWSに移行しています。ネットワークとセキュリティ戦略を確定した後、アプリケーションの迅速な移行と最新化を目指しています。同社は、中央ネットワークアカウントにAWS Direct Connect接続を設定しました。
同社は近い将来、数百のAWSアカウントとVPCを保有する見込みです。企業ネットワークはAWS上のリソースにシームレスにアクセスでき、すべてのVPCと通信できる必要があります。また、クラウドリソースをオンプレミスのデータセンター経由でインターネットに接続することも検討しています。
これらの要件を満たす手順の組み合わせはどれですか? (3 つ選択してください)。
A. 中央アカウントに Direct Connect ゲートウェイを作成します。各アカウントで、Direct Connect ゲートウェイと各仮想プライベートゲートウェイのアカウント ID を使用して関連付け提案を作成します。
B. 中央ネットワークアカウントに Direct Connect ゲートウェイとトランジットゲートウェイを作成します。トランジット VIF を使用して、トランジットゲートウェイを Direct Connect ゲートウェイに接続します。
C.
インターネットゲートウェイをプロビジョニングします。インターネットゲートウェイをサブネットに接続します。ゲートウェイを通過するインターネットトラフィックを許可します。
D.
トランジットゲートウェイを他のアカウントと共有します。トランジットゲートウェイにVPCを接続します。
E. 必要に応じて VPC ピアリングをプロビジョニングします。
F.
プライベートサブネットのみをプロビジョニングします。トランジットゲートウェイとカスタマーゲートウェイで必要なルートを開き、AWS からのアウトバウンドインターネットトラフィックがデータセンターで実行される NAT サービスに流れるようにします。
Answer: B D F
Explanation:
For a large-scale multi-account AWS environment with many VPCs and centralized Direct Connect, AWS recommends using a Transit Gateway (TGW) architecture combined with a Direct Connect gateway (DXGW). This setup allows scalable, centralized connectivity between on-premises and multiple VPCs across accounts.
Step B: Creating a Direct Connect gateway and Transit Gateway in a central network account and connecting them via a transit VIF enables the on-premises network to access all connected VPCs.
Step D: Sharing the transit gateway with other accounts via AWS Resource Access Manager (RAM) allows the central TGW to attach VPCs in multiple accounts, simplifying multi-account connectivity.
Step F: To route cloud resources' internet traffic back through the on-premises data center (for centralized egress), provisioning only private subnets and routing outbound internet traffic through NAT or firewall services in the data center is necessary. This requires configuring transit gateway and customer gateway routes appropriately.

Option A is partially correct in the use of Direct Connect gateway but association proposals are not scalable for hundreds of VPCs and accounts compared to transit gateway. Option C (internet gateway) is irrelevant here as traffic egress is required via on-premises data center, not directly to the internet. Option E (VPC peering) is not scalable for hundreds of VPCs.
References:
AWS Transit Gateway Overview (https://docs.aws.amazon.com/vpc/latest/tgw/what-is-transit-gateway.html) AWS Direct Connect Gateway (https://docs.aws.amazon.com/directconnect/latest/UserGuide/direct-connect- gateways.html) Centralized Egress Architecture with Transit Gateway (https://aws.amazon.com/blogs/networking-and- content-delivery/how-to-set-up-centralized-egress-with-transit-gateway/) AWS Well-Architected Framework - Reliability Pillar (https://d1.awsstatic.com/whitepapers/architecture/AWS_Well-Architected_Framework.pdf)

QUESTION NO: 34
ある企業は、AWS Organizations を使用して複数の AWS アカウントを管理しています。この企業では、アカウント A の特定の Amazon SNS トピックからアカウント B の特定の Amazon SQS キューにメッセージをパブリッシュできる、安全なイベントドリブンアーキテクチャを必要としています。
最小権限を維持しながらこれらの要件を満たすソリューションはどれですか?

A.
アカウントAに、任意のSQSキューにパブリッシュできる新しいIAMロールを作成します。このロールのARNをアカウントBと共有します。

B.
アカウントBのSQSキューポリシーにSNSトピックARNを追加します。SNSトピックが任意のキューにパブリッシュされるように設定します。キューをAWS KMSキーで暗号化します。

C.
アカウントBのSQSキューポリシーを変更し、アカウントAの特定のSNSトピックARNのみがメッセージをパブリッシュできるようにします。SNSトピックに、特定のキューARNに対するパブリッシュ権限が付与されていることを確認してください。

D.
両方のアカウント間で共有されるIAMロールを作成し、すべてのSQSキューへのパブリッシュ権限を付与します。クロスアカウントアクセスを有効にします。

Answer: C
Explanation:
AWS documentation states that the correct and least-privilege method for cross-account SNS-to-SQS integration is:
* Add specific SNS topic ARNs to the SQS queue policy.
* Allow only those topics to publish messages to the queue.
* Ensure SNS has permission to publish to the specific queue ARN.
This ensures strict scoping and adheres to least privilege.
Options A and D grant overly broad permissions. Option B allows publishing to any queue, which violates least privilege.

QUESTION NO: 35
ある企業は、単一のAWSリージョン内のAmazon EC2インスタンス上でエンタープライズリソースプランニング（ERP）システムを運用しています。ユーザーは、EC2インスタンスでホストされているパブリックAPIを使用してERPシステムに接続します。
海外のユーザーからは、データセンターからの API 応答時間が遅いという報告があります。ソリューション アーキテクトは、国際ユーザーに対する API 応答時間を改善する必要があります。
これらの要件を最もコスト効率よく満たすソリューションはどれでしょうか?
A.
各ユーザーのデータセンターをEC2インスタンスに接続するためのパブリック仮想インターフェース（VIF）を持つAWS Direct Connect接続を設定します。ERPシステムAPI用のDirect Connectゲートウェイを作成し、ユーザーのAPIリクエストをルーティングします。
B. Amazon API Gateway エンドポイントを複数のリージョンにデプロイします。Amazon Route 53 のレイテンシーベースルーティングを使用して、リクエストを最も近いエンドポイントにルーティングします。ERP システムに接続するために、リージョン間の VPC ピアリング接続を設定します。
C. AWS Global Accelerator をセットアップします。必要なポートのリスナーを設定します。トラフィックを分散するために、適切なリージョンのエンドポイントグループを設定します。各グループに API 用のエンドポイントを作成します。
D. AWS サイト間VPN を使用して、複数のリージョンとユーザーネットワーク間に専用のVPNトンネルを確立します。VPN接続を介してAPIへのトラフィックをルーティングします。
Answer: C
Explanation:
AWS Global Accelerator improves the performance and availability of applications by directing user traffic through the AWS global network of edge locations using anycast IP addresses. It reduces latency and jitter for global users accessing applications in a single Region.
Why this works:
Global Accelerator routes user requests to the nearest AWS edge location using AWS's high-performance backbone network.
It then forwards traffic to the optimal endpoint - in this case, the public API hosted on EC2.
This is much more cost-effective and requires less operational complexity than deploying and maintaining multiple API Gateway endpoints across regions (Option B), or setting up Direct Connect links for every international location (Option A).
Option C requires no application change and is designed specifically for latency improvement and high availability.
References:
AWS Global Accelerator Documentation
Use Cases for Global Accelerator
Performance Improvements for Global Users