AWS Certified Solutions Architect Associate Practice Test 6 - Results

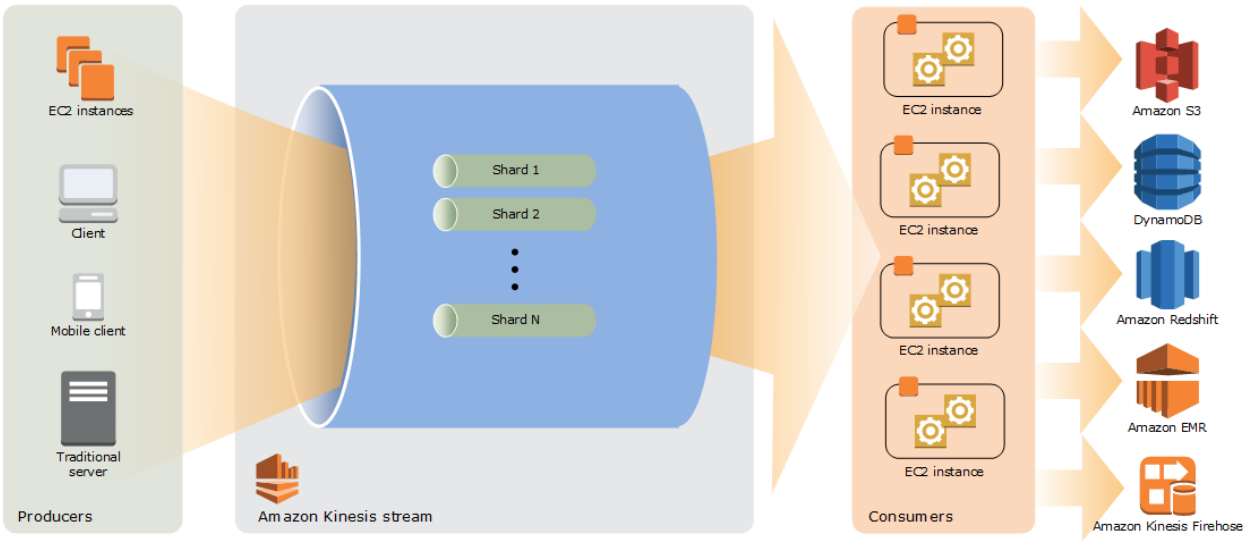
Return to review
Chart
Pie chart with 4 slices.
End of interactive chart.
Attempt 2
All knowledge areas
All questions
Question 1: **Correct**
A startup is building IoT devices and monitoring applications. They are using IoT sensors to monitor the traffic in real-time by using an Amazon Kinesis Stream that is configured with default settings. It then sends the data to an Amazon S3 bucket every 3 days. When you checked the data in S3 on the 3rd day, only the data for the last day is present and no data is present from 2 days ago.

Which of the following is the MOST likely cause of this issue?

- **Amazon S3 bucket has encountered a data loss.**
- **Someone has manually deleted the record in Amazon S3.**
- **By default, data records in Kinesis are only accessible for 24 hours from the time they are added to a stream.**

  (Correct)
- **The access of the Kinesis stream to the S3 bucket is insufficient.**

Explanation
By default, records of a stream in Amazon Kinesis are accessible for up to 24 hours from the time they are added to the stream. You can raise this limit to up to 7 days by enabling extended data retention.



Hence, the correct answer is: **By default, data records in Kinesis are only accessible for 24 hours from the time they are added to a stream.**

The option that says: **Amazon S3 bucket has encountered a data loss** is incorrect because Amazon S3 rarely experiences data loss. Amazon has an SLA for S3 that it commits to its customers. Amazon S3 Standard, S3 Standard–IA, S3 One Zone-IA, and S3 Glacier are all designed to provide 99.999999999% durability of objects over a given year. This durability level corresponds to an average annual expected loss of 0.000000001% of objects. Hence, Amazon S3 bucket data loss is highly unlikely.

The option that says: **Someone has manually deleted the record in Amazon S3** is incorrect because if someone has deleted the data, this should have been visible in CloudTrail. Also, deleting that much data manually shouldn't have occurred in the first place if you have put in the appropriate security measures.

The option that says: **The access of the Kinesis stream to the S3 bucket is insufficient** is incorrect because having insufficient access is highly unlikely since you are able to access the bucket and view the contents of the previous day's data collected by Kinesis.

Reference:

https://aws.amazon.com/kinesis/data-streams/faqs/

https://docs.aws.amazon.com/AmazonS3/latest/dev/DataDurability.html

Check out this Amazon Kinesis Cheat Sheet:

https://tutorialsdojo.com/amazon-kinesis/

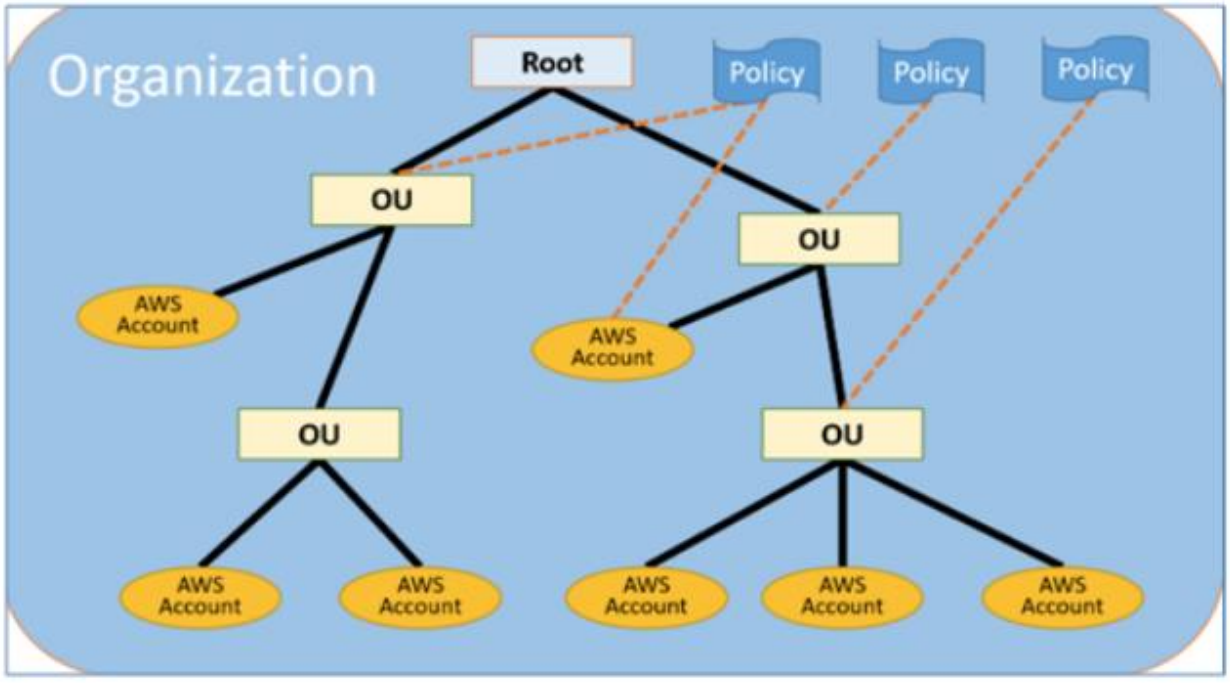
Question 2: **Correct**
A multinational manufacturing company has multiple accounts in AWS to separate their various departments such as finance, human resources, engineering and many others. There is a requirement to ensure that certain access to services and actions are properly controlled to comply with the security policy of the company.

As the Solutions Architect, which is the most suitable way to set up the multi-account AWS environment of the company?

- **Set up a common IAM policy that can be applied across all AWS accounts.**
- **Connect all departments by setting up a cross-account access to each of the AWS accounts of the company. Create and attach IAM policies to your resources based on their respective departments to control access.**
- **Provide access to externally authenticated users via Identity Federation. Set up an IAM role to specify permissions for users from each department whose identity is federated from your organization or a third-party identity provider.**
- **Use AWS Organizations and Service Control Policies to control services on each account.**

  (Correct)

Explanation
Using AWS Organizations and Service Control Policies to control services on each account is the correct answer. Refer to the diagram below:



AWS Organizations offers policy-based management for multiple AWS accounts. With Organizations, you can create groups of accounts, automate account creation, apply and manage policies for those groups. Organizations enable you to centrally manage policies across multiple accounts without requiring custom scripts and manual processes. It allows you to create Service Control Policies (SCPs) that centrally control AWS service use across multiple AWS accounts.

The option that says: **Setting up a common IAM policy that can be applied across all AWS accounts** is incorrect because it is not possible to create a common IAM policy for multiple AWS accounts.

The option that says: **Connect all departments by setting up a cross-account access to each of the AWS accounts of the company. Create and attach IAM policies to your resources based on their respective departments to control access** is incorrect. Although you can set up cross-account access to each department, this entails a lot of configuration compared with using AWS Organizations and Service Control Policies (SCPs). Cross-account access would be a more suitable choice if you only have two accounts to manage, but not for multiple accounts.

The option that says: **Provide access to externally authenticated users via Identity Federation. Set up an IAM role to specify permissions for users from each department whose identity is federated from your organization or a third-party identity provider** is incorrect as this is focused on the Identity Federation authentication set up for your AWS accounts but not the IAM policy management for multiple AWS accounts. A

combination of AWS Organizations and Service Control Policies (SCPs) is a better choice compared to this option.

Reference:

https://aws.amazon.com/organizations/

Check out this AWS Organizations Cheat Sheet:

https://tutorialsdojo.com/aws-organizations/

Service Control Policies (SCP) vs. IAM Policies:

https://tutorialsdojo.com/service-control-policies-scp-vs-iam-policies/

Comparison of AWS Services Cheat Sheets:

https://tutorialsdojo.com/comparison-of-aws-services/

Question 3: **Incorrect**

A game company has a requirement of load balancing the incoming TCP traffic at the transport level (Layer 4) to their containerized gaming servers hosted in AWS Fargate. To maintain performance, it should handle millions of requests per second sent by gamers around the globe while maintaining ultra-low latencies.

Which of the following must be implemented in the current architecture to satisfy the new requirement?

- **Launch a new microservice in AWS Fargate that acts as a load balancer since using an ALB or NLB with Fargate is not possible.**
- **Create a new record in Amazon Route 53 with Weighted Routing policy to load balance the incoming traffic.**

(Incorrect)

- **Launch a new Network Load Balancer.**

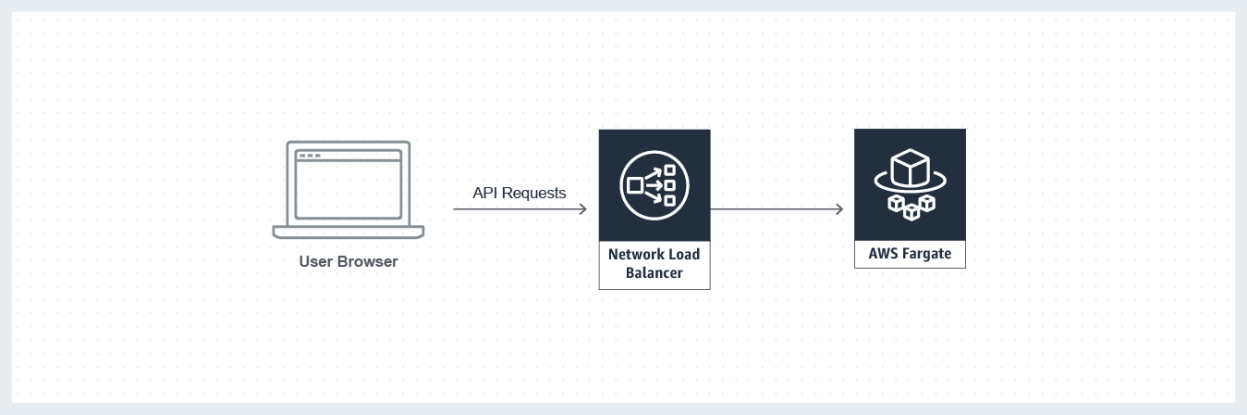
  (Correct)

- **Launch a new Application Load Balancer.**

Explanation

Elastic Load Balancing automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, IP addresses, and Lambda functions. It can handle the varying load of your application traffic in a single Availability Zone or across multiple Availability Zones. Elastic Load Balancing offers three types of load balancers that all feature the high availability, automatic scaling, and robust security necessary to make your applications fault-tolerant. They are: Application Load Balancer, Network Load Balancer, and Classic Load Balancer

Network Load Balancer is best suited for load balancing of TCP traffic where extreme performance is required. Operating at the connection level (Layer 4), Network Load Balancer routes traffic to targets within Amazon Virtual Private Cloud (Amazon VPC) and is capable of handling millions of requests per second while maintaining ultra-low latencies. Network Load Balancer is also optimized to handle sudden and volatile traffic patterns.



Hence, the correct answer is to **launch a new Network Load Balancer**.

The option that says: **Launch a new Application Load Balancer** is incorrect because it cannot handle TCP or Layer 4 connections, only Layer 7 (HTTP and HTTPS).

The option that says: **Create a new record in Amazon Route 53 with Weighted Routing policy to load balance the incoming traffic** is incorrect. Although Route 53 can act as a

load balancer by assigning each record a relative weight that corresponds to how much traffic you want to send to each resource, it is still not capable of handling millions of requests per second while maintaining ultra-low latencies. You have to use a Network Load Balancer instead.

The option that says: **Launch a new microservice in AWS Fargate that acts as a load balancer since using an ALB or NLB with Fargate is not possible** is incorrect because you can place an ALB and NLB in front of your AWS Fargate cluster.

References:

https://aws.amazon.com/elasticloadbalancing/features/#compare

https://docs.aws.amazon.com/AmazonECS/latest/developerguide/load-balancer-types.html

https://aws.amazon.com/getting-started/projects/build-modern-app-fargate-lambda-dynamodb-python/module-two/

Check out this AWS Elastic Load Balancing (ELB) Cheat Sheet:

https://tutorialsdojo.com/aws-elastic-load-balancing-elb/

Question 4: **Correct**
A company plans to use a cloud storage service to temporarily store its log files. The number of files to be stored is still unknown, but it only needs to be kept for 12 hours.

Which of the following is the most cost-effective storage class to use in this scenario?

- **Amazon S3 Glacier Deep Archive**
- **Amazon S3 Standard-IA**
- **Amazon S3 Standard**

(Correct)

-

Amazon S3 One Zone-IA

Explanation

Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance. Amazon S3 also offers a range of storage classes for the objects that you store. You choose a class depending on your use case scenario and performance access requirements. All of these storage classes offer high durability.

	Storage class	Designed for	Availability Zones	Min storage duration
◉	Standard	Frequently accessed data (more than once a month) with milliseconds access	≥ 3	-
○	Intelligent-Tiering	Data with changing or unknown access patterns	≥ 3	-
○	Standard-IA	Infrequently accessed data (once a month) with milliseconds access	≥ 3	30 days
○	One Zone-IA	Recreatable, infrequently accessed data (once a month) stored in a single Availability Zone with milliseconds access	1	30 days
○	Glacier Instant Retrieval	Long-lived archive data accessed once a quarter with instant retrieval in milliseconds	≥ 3	90 days
○	Glacier Flexible Retrieval (formerly Glacier)	Long-lived archive data accessed once a year with retrieval of minutes to hours	≥ 3	90 days
○	Glacier Deep Archive	Long-lived archive data accessed less than once a year with retrieval of hours	≥ 3	180 days
○	Reduced redundancy	Noncritical, frequently accessed data with milliseconds access (not recommended as S3 Standard is more cost effective)	≥ 3	-

The scenario requires you to select a cost-effective service that does not have a **minimum storage duration** since the data will only last for 12 hours. Among the options given, only Amazon S3 Standard has the feature of no minimum storage duration. It is also the most cost-effective storage service because you will only be

charged for the last 12 hours, unlike in other storage classes where you will still be charged based on its respective storage duration (e.g. 30 days, 90 days, 180 days). S3 Intelligent-Tiering also has no minimum storage duration and this is designed for data with changing or unknown access patters.

S3 Standard-IA is designed for long-lived but infrequently accessed data that is retained for months or years. Data that is deleted from S3 Standard-IA within 30 days will still be charged for a full 30 days.

S3 Glacier Deep Archive is designed for long-lived but rarely accessed data that is retained for 7-10 years or more. Objects that are archived to S3 Glacier Deep Archive have a minimum of 180 days of storage, and objects deleted before 180 days incur a pro-rated charge equal to the storage charge for the remaining days.

Hence, the correct answer is: **Amazon S3 Standard**.

Amazon S3 Standard-IA is incorrect because this storage class has a minimum storage duration of at least 30 days. Remember that the scenario requires the data to be kept for 12 hours only.

Amazon S3 One Zone-IA is incorrect. Just like S3 Standard-IA, this storage class has a minimum storage duration of at least 30 days.

Amazon S3 Glacier Deep Archive is incorrect. Although it is the most cost-effective storage class among all other options, it has a minimum storage duration of at least 180 days which is only suitable for backup and data archival. If you store your data in Glacier Deep Archive for only 12 hours, you will still be charged for the full 180 days.

References:

https://docs.aws.amazon.com/AmazonS3/latest/dev/storage-class-intro.html

https://aws.amazon.com/s3/storage-classes/

Check out this Amazon S3 Cheat Sheet:

https://tutorialsdojo.com/amazon-s3/

S3 Standard vs S3 Standard-IA vs S3 One Zone-IA Cheat Sheet:

https://tutorialsdojo.com/s3-standard-vs-s3-standard-ia-vs-s3-one-zone-ia/

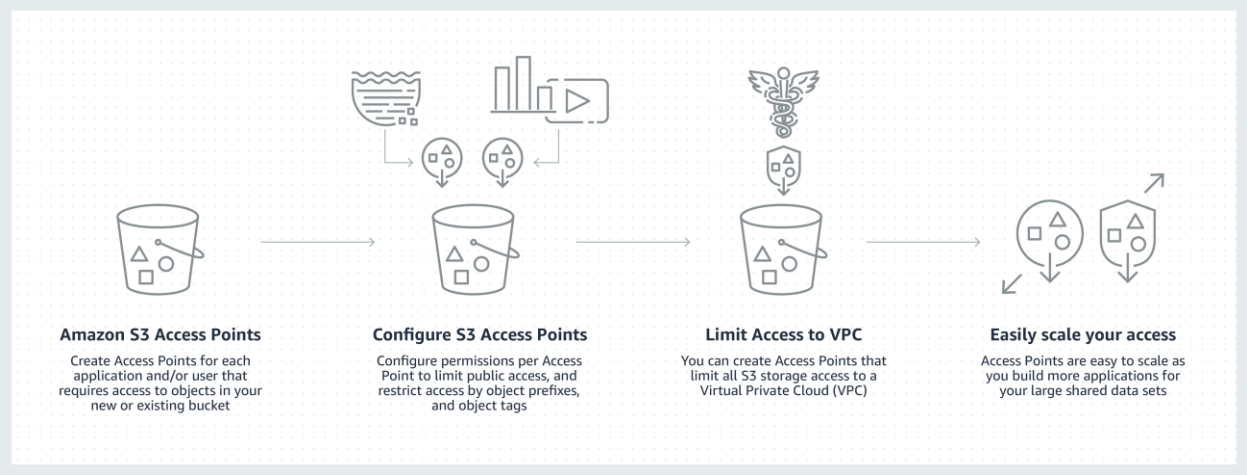
Question 5: **Correct**
A company is using Amazon S3 to store frequently accessed data. The S3 bucket is shared with external users that will upload files regularly. A Solutions Architect needs to implement a solution that will grant the bucket owner full access to all uploaded objects in the S3 bucket.

What action should be done to achieve this task?

- **Enable the Requester Pays feature in the Amazon S3 bucket.**
- **Enable server access logging and set up an IAM policy that will require the users to set the object's ACL to `bucket-owner-full-control`.**
- **Create a bucket policy that will require the users to set the object's ACL to `bucket-owner-full-control`.**

  (Correct)
- **Create a CORS configuration in the S3 bucket.**

Explanation
Amazon S3 stores data as objects within buckets. An object is a file and any optional metadata that describes the file. To store a file in Amazon S3, you upload it to a bucket. When you upload a file as an object, you can set permissions on the object and any metadata. Buckets are containers for objects. You can have one or more buckets. You can control access for each bucket, deciding who can create, delete, and list objects in it. You can also choose the geographical Region where Amazon S3 will store the bucket and its contents and view access logs for the bucket and its objects.



By default, an S3 object is owned by the AWS account that uploaded it even though the bucket is owned by another account. To get full access to the object, the object owner must explicitly grant the bucket owner access. You can create a bucket policy to require external users to grant `bucket-owner-full-control` when uploading objects so the bucket owner can have full access to the objects.

Hence, the correct answer is: **Create a bucket policy that will require the users to set the object's ACL to `bucket-owner-full-control`**.

The option that says: **Enable the Requester Pays feature in the Amazon S3 bucket** is incorrect because this won't grant the bucket owner full access to the uploaded objects in the S3 bucket. With Requester Pays buckets, the requester, instead of the bucket owner, pays the cost of the request and the data download from the bucket.

The option that says: **Create a CORS configuration in the S3 bucket** is incorrect because this only allows cross-origin access to your Amazon S3 resources. If you need to grant the bucket owner full control in the uploaded objects, you must create a bucket policy and require external users to grant `bucket-owner-full-control` when uploading objects.

The option that says: **Enable server access logging and set up an IAM policy that will require the users to set the bucket's ACL to `bucket-owner-full-control`** is incorrect because this only provides detailed records for the requests that are made to a bucket. In addition, the `bucket-owner-full-control` canned ACL must be associated with the bucket policy and not to an IAM policy. This will require the users to set the object's ACL (not the bucket's) to `bucket-owner-full-control`.

References:

https://aws.amazon.com/premiumsupport/knowledge-center/s3-bucket-owner-access/

https://aws.amazon.com//premiumsupport/knowledge-center/s3-require-object-ownership/

Check out this Amazon S3 Cheat Sheet:

https://tutorialsdojo.com/amazon-s3/

Question 6: **Incorrect**
An application is hosted in an Auto Scaling group of EC2 instances and a Microsoft SQL Server on Amazon RDS. There is a requirement that all in-flight data between your web servers and RDS should be secured.

Which of the following options is the MOST suitable solution that you should implement? (Select TWO.)

- ☑ **Enable the IAM DB authentication in RDS using the AWS Management Console.**

  (Incorrect)

- ☐ **Specify the TDE option in an RDS option group that is associated with that DB instance to enable transparent data encryption (TDE).**

- ☑ **Force all connections to your DB instance to use SSL by setting the `rds.force_ssl` parameter to true. Once done, reboot your DB instance.**

  (Correct)

- ☐ **Download the Amazon RDS Root CA certificate. Import the certificate to your servers and configure your application to use SSL to encrypt the connection to RDS.**

  (Correct)

- ☐

Configure the security groups of your EC2 instances and RDS to only allow traffic to and from port 443.

Explanation

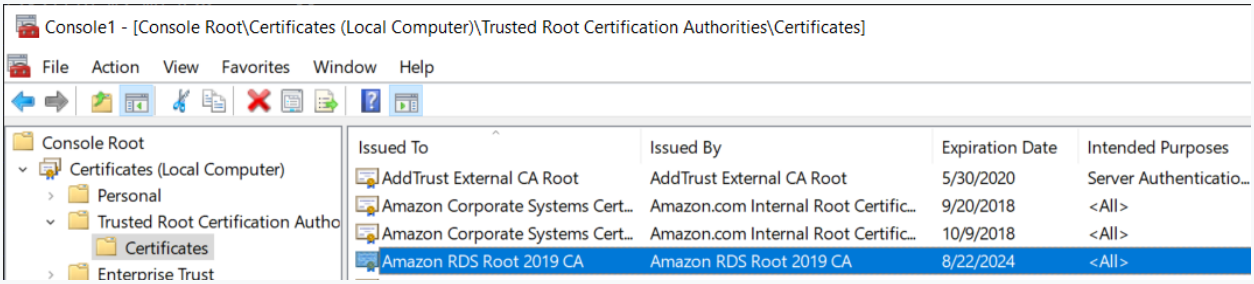
You can use Secure Sockets Layer (SSL) to encrypt connections between your client applications and your Amazon RDS DB instances running Microsoft SQL Server. SSL support is available in all AWS regions for all supported SQL Server editions.

When you create an SQL Server DB instance, Amazon RDS creates an SSL certificate for it. The SSL certificate includes the DB instance endpoint as the Common Name (CN) for the SSL certificate to guard against spoofing attacks.

There are 2 ways to use SSL to connect to your SQL Server DB instance:

- Force SSL for all connections — this happens transparently to the client, and the client doesn't have to do any work to use SSL.

- Encrypt specific connections — this sets up an SSL connection from a specific client computer, and you must do work on the client to encrypt connections.



You can force all connections to your DB instance to use SSL, or you can encrypt connections from specific client computers only. To use SSL from a specific client, you must obtain certificates for the client computer, import certificates on the client computer, and then encrypt the connections from the client computer.

If you want to force SSL, use the `rds.force_ssl` parameter. By default, the `rds.force_ssl` parameter is set to `false`. Set the `rds.force_ssl` parameter to `true` to force connections to use SSL. The `rds.force_ssl` parameter is static, so after you change the value, you must reboot your DB instance for the change to take effect.

Hence, the correct answers for this scenario are the options that say:

- Force all connections to your DB instance to use SSL by setting the `rds.force_ssl` parameter to true. Once done, reboot your DB instance.

- Download the Amazon RDS Root CA certificate. Import the certificate to your servers and configure your application to use SSL to encrypt the connection to RDS.

Specifying the TDE option in an RDS option group that is associated with that DB instance to enable transparent data encryption (TDE) is incorrect because transparent data encryption (TDE) is primarily used to encrypt stored data on your DB instances running Microsoft SQL Server and not the data that is in transit.

Enabling the IAM DB authentication in RDS using the AWS Management Console is incorrect because IAM database authentication is only supported in MySQL and PostgreSQL database engines. With IAM database authentication, you don't need to use a password when you connect to a DB instance but instead, you use an authentication token.

Configuring the security groups of your EC2 instances and RDS to only allow traffic to and from port 443 is incorrect because it is not enough to do this. You need to either force all connections to your DB instance to use SSL, or you can encrypt connections from specific client computers, just as mentioned above.

References:

https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/SQLServer.Concepts.General.SSL.Using.html

https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Appendix.SQLServer.Options.TDE.html

https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/UsingWithRDS.IAMDBAuth.html

Check out this Amazon RDS Cheat Sheet:

https://tutorialsdojo.com/amazon-relational-database-service-amazon-rds/

Question 7: **Correct**

A fast food company is using AWS to host their online ordering system which uses an Auto Scaling group of EC2 instances deployed across multiple Availability Zones with an Application Load Balancer in front. To better handle the incoming traffic from various digital devices, you are planning to implement a new routing system where requests which have a URL of <server>/api/android are forwarded to one specific target group named "Android-Target-Group". Conversely, requests which have a URL of

<server>/api/ios are forwarded to another separate target group named "iOS-Target-Group".

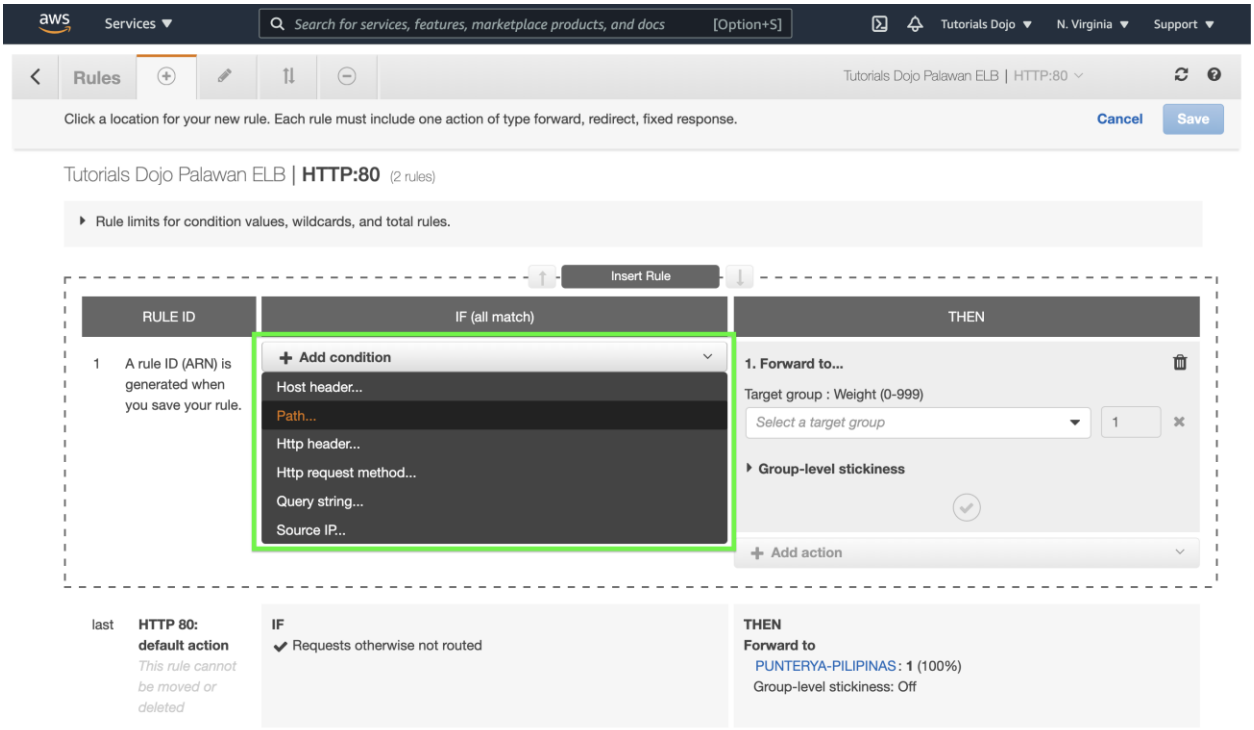
How can you implement this change in AWS?

- **Use host conditions to define rules that forward requests to different target groups based on the hostname in the host header. This enables you to support multiple domains using a single load balancer.**
- **Use path conditions to define rules that forward requests to different target groups based on the URL in the request.**

  (Correct)
- **Replace your ALB with a Gateway Load Balancer then use path conditions to define rules that forward requests to different target groups based on the URL in the request.**
- **Replace your ALB with a Network Load Balancer then use host conditions to define rules that forward requests to different target groups based on the URL in the request.**

Explanation

If your application is composed of several individual services, an Application Load Balancer can route a request to a service based on the content of the request such as Host field, Path URL, HTTP header, HTTP method, Query string, or Source IP address. Path-based routing allows you to route a client request based on the URL path of the HTTP header. Each path condition has one path pattern. If the URL in a request matches the path pattern in a listener rule exactly, the request is routed using that rule.



A path pattern is case-sensitive, can be up to 128 characters in length, and can contain any of the following characters. You can include up to three wildcard characters.

A–Z, a–z, 0–9

_ - . $ / ~ " ' @ : +

& (using &)

* (matches 0 or more characters)

? (matches exactly 1 character)

Example path patterns

`/img/*`

`/js/*`

You can use path conditions to define rules that forward requests to different target groups based on the URL in the request (also known as path-based routing). This type of routing is the most appropriate solution for this scenario hence, the correct answer is: **Use path conditions to define rules that forward requests to different target groups based on the URL in the request.**

The option that says: **Use host conditions to define rules that forward requests to different target groups based on the hostname in the host header. This enables you to support multiple domains using a single load balancer** is incorrect because host-based routing defines rules that forward requests to different target groups based on the hostname in the host header instead of the URL, which is what is needed in this scenario.

The option that says: **Replace your ALB with a Gateway Load Balancer then use path conditions to define rules that forward requests to different target groups based on the URL in the request** is incorrect because a Gateway Load Balancer does not support path-based routing. You must use an Application Load Balancer.

The option that says: **Replace your ALB with a Network Load Balancer then use host conditions to define rules that forward requests to different target groups based on the URL in the request** is incorrect because a Network Load Balancer is used for applications that need extreme network performance and static IP. It also does not support path-based routing which is what is needed in this scenario. Furthermore, the statement mentions host-based routing even though the scenario is about path-based routing.

References:

https://docs.aws.amazon.com/elasticloadbalancing/latest/application/introduction.html#application-load-balancer-benefits

https://docs.aws.amazon.com/elasticloadbalancing/latest/application/load-balancer-listeners.html#path-conditions

Check out this AWS Elastic Load Balancing (ELB) Cheat Sheet:

https://tutorialsdojo.com/aws-elastic-load-balancing-elb/

Application Load Balancer vs Network Load Balancer vs Classic Load Balancer:

https://tutorialsdojo.com/application-load-balancer-vs-network-load-balancer-vs-classic-load-balancer/

Question 8: **Incorrect**

A Solutions Architect is implementing a new High-Performance Computing (HPC) system in AWS that involves orchestrating several Amazon Elastic Container Service (Amazon ECS) tasks with an EC2 launch type that is part of an Amazon ECS cluster. The system will be frequently accessed by users around the globe and it is expected that there would be hundreds of ECS tasks running most of the time. The Architect must ensure that its storage system is optimized for high-frequency read and write operations. The output data of each ECS task is around 10 MB but the obsolete data will eventually be archived and deleted so the total storage size won't exceed 10 TB.

Which of the following is the MOST suitable solution that the Architect should recommend?

- **Launch an Amazon Elastic File System (Amazon EFS) with Provisioned Throughput mode and set the performance mode to `Max I/O`. Configure the EFS file system as the container mount point in the ECS task definition of the Amazon ECS cluster.**

  (Correct)

- **Launch an Amazon Elastic File System (Amazon EFS) file system with Bursting Throughput mode and set the performance mode to `General Purpose`. Configure the EFS file system as the container mount point in the ECS task definition of the Amazon ECS cluster.**

- **Set up an SMB file share by creating an Amazon FSx File Gateway in Storage Gateway. Set the file share as the container mount point in the ECS task definition of the Amazon ECS cluster.**

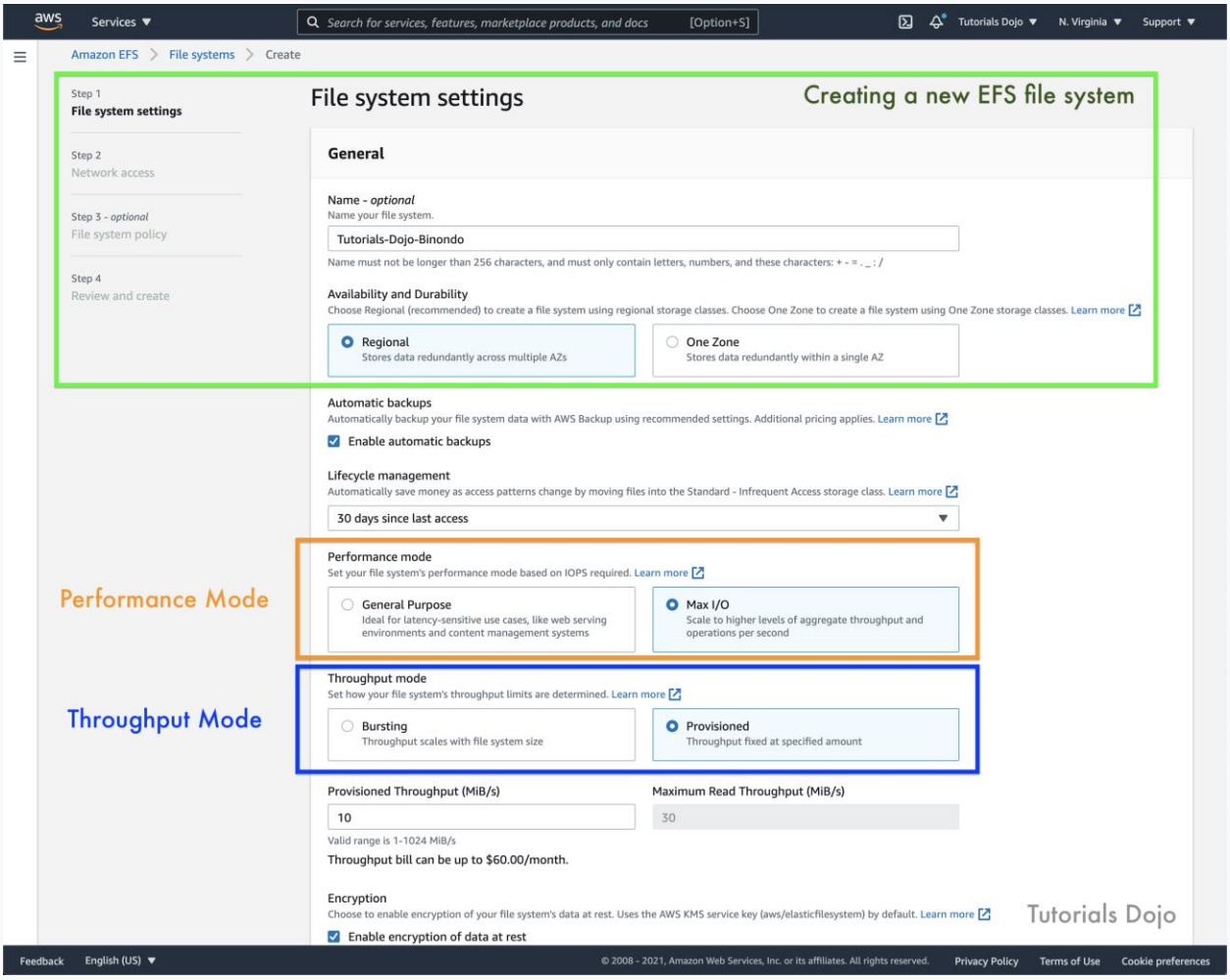
- **Launch an Amazon DynamoDB table with Amazon DynamoDB Accelerator (DAX) and DynamoDB Streams enabled. Configure the table to be accessible by all Amazon ECS cluster instances. Set the DynamoDB table as the container mount point in the ECS task definition of the Amazon ECS cluster.**

  (Incorrect)

Explanation

Amazon Elastic File System (Amazon EFS) provides simple, scalable file storage for use with your Amazon ECS tasks. With Amazon EFS, storage capacity is elastic, growing and shrinking automatically as you add and remove files. Your applications can have the storage they need when they need it.

You can use Amazon EFS file systems with Amazon ECS to access file system data across your fleet of Amazon ECS tasks. That way, your tasks have access to the same persistent storage, no matter the infrastructure or container instance on which they land. When you reference your Amazon EFS file system and container mount point in your Amazon ECS task definition, Amazon ECS takes care of mounting the file system in your container.



To support a wide variety of cloud storage workloads, Amazon EFS offers two performance modes:

- General Purpose mode

- Max I/O mode.

You choose a file system's performance mode when you create it, and it cannot be changed. The two performance modes have no additional costs, so your Amazon EFS file system is billed and metered the same, regardless of your performance mode.

There are two throughput modes to choose from for your file system:

- Bursting Throughput

- Provisioned Throughput

With Bursting Throughput mode, a file system's throughput scales as the amount of data stored in the EFS Standard or One Zone storage class grows. File-based workloads are typically spiky, driving high levels of throughput for short periods of time, and low levels of throughput the rest of the time. To accommodate this, Amazon EFS is designed to burst to high throughput levels for periods of time.

Provisioned Throughput mode is available for applications with high throughput to storage (MiB/s per TiB) ratios, or with requirements greater than those allowed by the Bursting Throughput mode. For example, say you're using Amazon EFS for development tools, web serving, or content management applications where the amount of data in your file system is low relative to throughput demands. Your file system can now get the high levels of throughput your applications require without having to pad your file system.

In the scenario, the file system will be frequently accessed by users around the globe so it is expected that there would be hundreds of ECS tasks running most of the time. The Architect must ensure that its storage system is optimized for high-frequency read and write operations.

Hence, the correct answer is: **Launch an Amazon Elastic File System (Amazon EFS) with Provisioned Throughput mode and set the performance mode to `Max I/O`. Configure the EFS file system as the container mount point in the ECS task definition of the Amazon ECS cluster.**

The option that says: **Set up an SMB file share by creating an Amazon FSx File Gateway in Storage Gateway. Set the file share as the container mount point in the ECS task definition of the Amazon ECS cluster** is incorrect. Although you can use an Amazon FSx for Windows File Server in this situation, it is not appropriate to use this since the application is not connected to an on-premises data center. Take note that the AWS Storage Gateway service is primarily used to integrate your existing on-premises storage to AWS.

The option that says: **Launch an Amazon Elastic File System (Amazon EFS) file system with Bursting Throughput mode and set the performance mode to `General Purpose`. Configure the EFS file system as the container mount point in the ECS task definition**

of the Amazon ECS cluster is incorrect because using Bursting Throughput mode won't be able to sustain the constant demand of the global application. Remember that the application will be frequently accessed by users around the world and there are hundreds of ECS tasks running most of the time.

The option that says: **Launch an Amazon DynamoDB table with Amazon DynamoDB Accelerator (DAX) and DynamoDB Streams enabled. Configure the table to be accessible by all Amazon ECS cluster instances. Set the DynamoDB table as the container mount point in the ECS task definition of the Amazon ECS cluster** is incorrect because you cannot directly set a DynamoDB table as a container mount point. In the first place, DynamoDB is a database and not a file system which means that it can't be "mounted" to a server.

References:

https://docs.aws.amazon.com/AmazonECS/latest/developerguide/tutorial-efs-volumes.html

https://docs.aws.amazon.com/efs/latest/ug/performance.html

https://docs.aws.amazon.com/AmazonECS/latest/developerguide/tutorial-wfsx-volumes.html

Check out this Amazon EFS Cheat Sheet:

https://tutorialsdojo.com/amazon-efs/

Question 9: **Correct**
A computer animation film studio has a web application running on an Amazon EC2 instance. It uploads 5 GB video objects to an Amazon S3 bucket. Video uploads are taking longer than expected, which impacts the performance of your application.

Which method will help improve the performance of the application?

- **Enable Enhanced Networking with the Elastic Network Adapter (ENA) on your EC2 Instances.**

Use Amazon S3 Multipart Upload API.

(Correct)

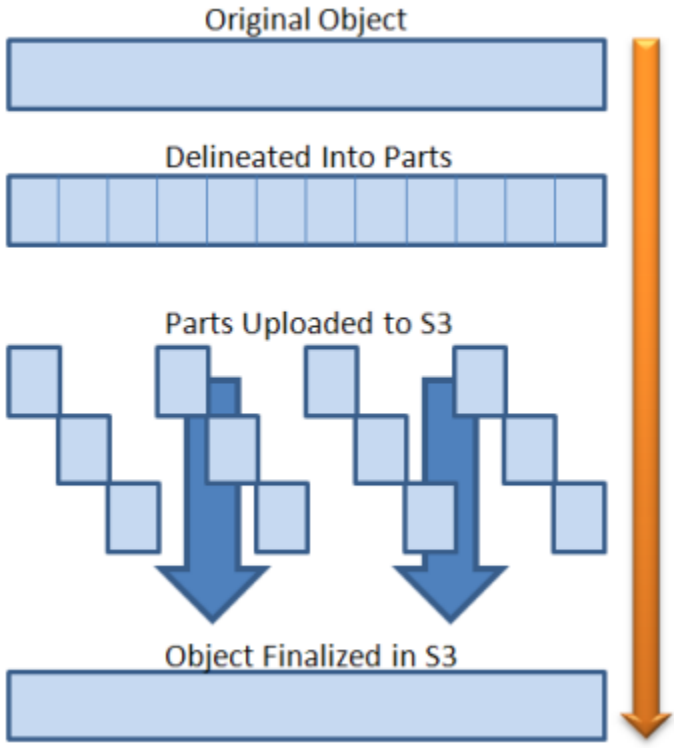
- **Leverage on Amazon CloudFront and use HTTP POST method to reduce latency.**

- **Use Amazon Elastic Block Store Provisioned IOPS and an Amazon EBS-optimized instance.**

Explanation

The main issue is the slow upload time of the video objects to Amazon S3. To address this issue, you can use Multipart upload in S3 to improve the throughput. It allows you to upload parts of your object in parallel thus, decreasing the time it takes to upload big objects. Each part is a contiguous portion of the object's data.

You can upload these object parts independently and in any order. If transmission of any part fails, you can retransmit that part without affecting other parts. After all parts of your object are uploaded, Amazon S3 assembles these parts and creates the object. In general, when your object size reaches 100 MB, you should consider using multipart uploads instead of uploading the object in a single operation.



Using multipart upload provides the following advantages:

Improved throughput - You can upload parts in parallel to improve throughput.

Quick recovery from any network issues - Smaller part size minimizes the impact of restarting a failed upload due to a network error.

Pause and resume object uploads - You can upload object parts over time. Once you initiate a multipart upload, there is no expiry; you must explicitly complete or abort the multipart upload.

Begin an upload before you know the final object size - You can upload an object as you are creating it.

Enabling Enhanced Networking with the Elastic Network Adapter (ENA) on your EC2 Instances is incorrect. Even though this will improve network performance, the issue will still persist since the problem lies in the upload time of the object to Amazon S3. You should use the Multipart upload feature instead.

Leveraging on Amazon CloudFront and using HTTP POST method to reduce latency is incorrect because CloudFront is a CDN service and is not used to expedite the upload process of objects to Amazon S3. Amazon CloudFront is a fast content delivery network (CDN) service that securely delivers data, videos, applications, and APIs to customers globally with low latency, high transfer speeds, all within a developer-friendly environment.

Using Amazon Elastic Block Store Provisioned IOPS and an Amazon EBS-optimized instance is incorrect. Although the use of Amazon Elastic Block Store Provisioned IOPS will speed up the I/O performance of the EC2 instance, the root cause is still not resolved since the primary problem here is the slow video upload to Amazon S3. There is no network contention in the EC2 instance.

References:

https://docs.aws.amazon.com/AmazonS3/latest/dev/uploadobjusingmpu.html

http://docs.aws.amazon.com/AmazonS3/latest/dev/qfacts.html

Check out this Amazon S3 Cheat Sheet:

https://tutorialsdojo.com/amazon-s3/

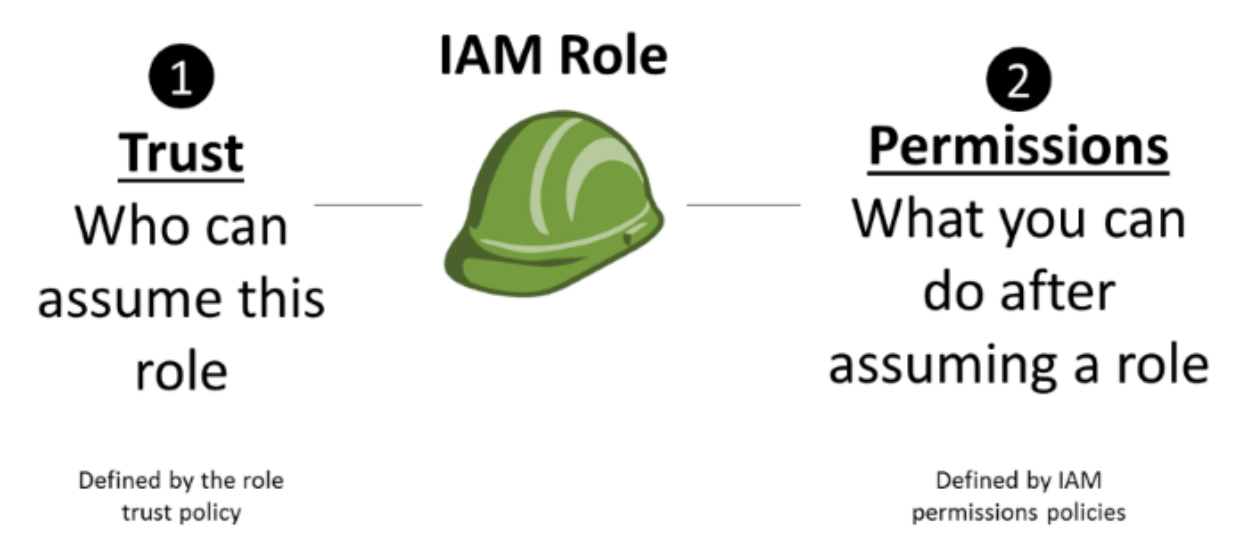
Question 10: **Correct**
A company deployed a web application to an EC2 instance that adds a variety of photo effects to a picture uploaded by the users. The application will put the generated photos to an S3 bucket by sending PUT requests to the S3 API.

What is the best option for this scenario considering that you need to have API credentials to be able to send a request to the S3 API?

- **Encrypt the API credentials and store in any directory of the EC2 instance.**
- **Store your API credentials in Amazon Glacier.**
- **Store the API credentials in the root web application directory of the EC2 instance.**
- **Create a role in IAM. Afterwards, assign this role to a new EC2 instance.**

  (Correct)

Explanation
The best option is to create a role in IAM. Afterward, assign this role to a new EC2 instance. Applications must sign their API requests with AWS credentials. Therefore, if you are an application developer, you need a strategy for managing credentials for your applications that run on EC2 instances.



You can securely distribute your AWS credentials to the instances, enabling the applications on those instances to use your credentials to sign requests while protecting your credentials from other users. However, it's challenging to securely distribute credentials to each instance, especially those that AWS creates on your behalf such as Spot Instances or instances in Auto Scaling groups. You must also be able to update the credentials on each instance when you rotate your AWS credentials.

In this scenario, you have to use IAM roles so that your applications can securely make API requests from your instances without requiring you to manage the security credentials that the applications use. Instead of creating and distributing your AWS credentials, you can delegate permission to make API requests using IAM roles.

Hence, the correct answer is: **Create a role in IAM. Afterwards, assign this role to a new EC2 instance.**

The option that says: **Encrypt the API credentials and storing in any directory of the EC2 instance** and **Store the API credentials in the root web application directory of the EC2 instance** are incorrect. Though you can store and use the API credentials in the EC2 instance, it will be difficult to manage just as mentioned above. You have to use IAM Roles.

The option that says: **Store your API credentials in Amazon S3 Glacier** is incorrect as Amazon S3 Glacier is used for data archives and not for managing API credentials.

Reference:

http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/iam-roles-for-amazon-ec2.html

Check out this Amazon EC2 Cheat Sheet:

https://tutorialsdojo.com/amazon-elastic-compute-cloud-amazon-ec2/

Question 11: **Correct**
A game development company operates several virtual reality (VR) and augmented reality (AR) games which use various RESTful web APIs hosted on their on-premises data center. Due to the unprecedented growth of their company, they decided to migrate their system to AWS Cloud to scale out their resources as well to minimize costs.

Which of the following should you recommend as the most cost-effective and scalable solution to meet the above requirement?

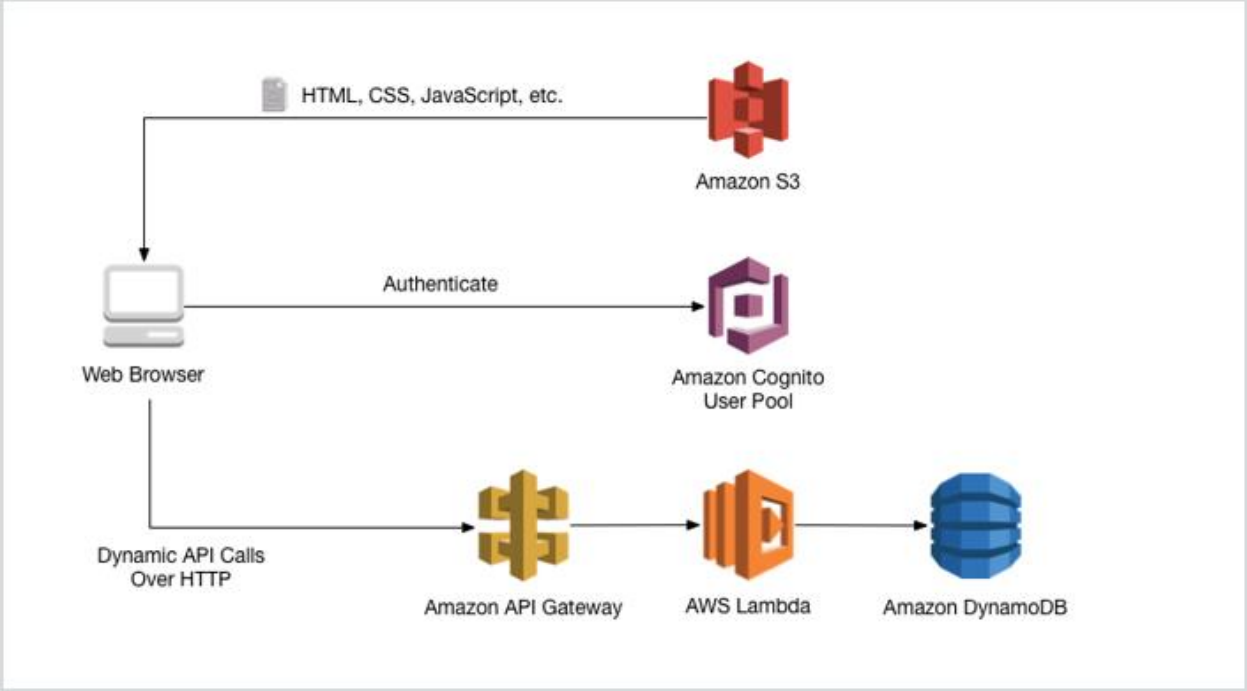
- **Host the APIs in a static S3 web hosting bucket behind a CloudFront web distribution.**
- **Use AWS Lambda and Amazon API Gateway.**

  (Correct)
- **Set up a micro-service architecture with ECS, ECR, and Fargate.**
- **Use a Spot Fleet of Amazon EC2 instances, each with an Elastic Fabric Adapter (EFA) for more consistent latency and higher network throughput. Set up an Application Load Balancer to distribute traffic to the instances.**

Explanation
With **AWS Lambda**, you pay only for what you use. You are charged based on the number of requests for your functions and the duration, the time it takes for your code to execute.

Lambda counts a request each time it starts executing in response to an event notification or invoke call, including test invokes from the console. You are charged for the total number of requests across all your functions.

Duration is calculated from the time your code begins executing until it returns or otherwise terminates, rounded up to the nearest 1ms. The price depends on the amount of memory you allocate to your function. The Lambda free tier includes 1M free requests per month and over 400,000 GB seconds of compute time per month.



The best possible answer here is to use a combination of AWS Lambda and Amazon API Gateway because this solution is both scalable and cost-effective. You will only be charged when you use your Lambda function, unlike having an EC2 instance that always runs even though you don't use it.

Hence, the correct answer is: **Use AWS Lambda and Amazon API Gateway.**

The option that says: **Setting up a micro-service architecture with ECS, ECR, and Fargate** is incorrect because ECS is mainly used to host Docker applications, and in addition, using ECS, ECR, and Fargate alone is not scalable and not recommended for this type of scenario.

The option that says: **Hosting the APIs in a static S3 web hosting bucket behind a CloudFront web distribution** is not a suitable option as there is no compute capability for S3 and you can only use it as a static website. Although this solution is scalable since uses CloudFront, the use of S3 to host the web APIs or the dynamic website is still incorrect.

The option that says: **Use a Spot Fleet of Amazon EC2 instances, each with an Elastic Fabric Adapter (EFA) for more consistent latency and higher network throughput. Set up an Application Load Balancer to distribute traffic to the instances** is incorrect. EC2

alone, without Auto Scaling, is not scalable. Even though you use Spot EC2 instance, it is still more expensive compared to Lambda because you will be charged only when your function is being used. An Elastic Fabric Adapter (EFA) is simply a network device that you can attach to your Amazon EC2 instance that enables you to achieve the application performance of an on-premises HPC cluster, with scalability, flexibility, and elasticity provided by the AWS Cloud. Although EFA is scalable, the Spot Fleet configuration of this option doesn't have Auto Scaling involved.

References:

https://docs.aws.amazon.com/apigateway/latest/developerguide/getting-started-with-lambda-integration.html

https://aws.amazon.com/lambda/pricing/

Check out this AWS Lambda Cheat Sheet:

https://tutorialsdojo.com/aws-lambda/

EC2 Container Service (ECS) vs Lambda:

https://tutorialsdojo.com/ec2-container-service-ecs-vs-lambda/

Question 12: **Incorrect**
A disaster recovery team is planning to back up on-premises records to a local file server share through SMB protocol. To meet the company's business continuity plan, the team must ensure that a copy of data from 48 hours ago is available for immediate access. Accessing older records with delay is tolerable.

Which should the DR team implement to meet the objective with the LEAST amount of configuration effort?

- **Use an AWS Storage File gateway with enough storage to keep data from the last 48 hours. Send the backups to an SMB share mounted as a local disk.**

  (Correct)

- **Create an AWS Backup plan to copy data backups to a local SMB share every 48 hours.**

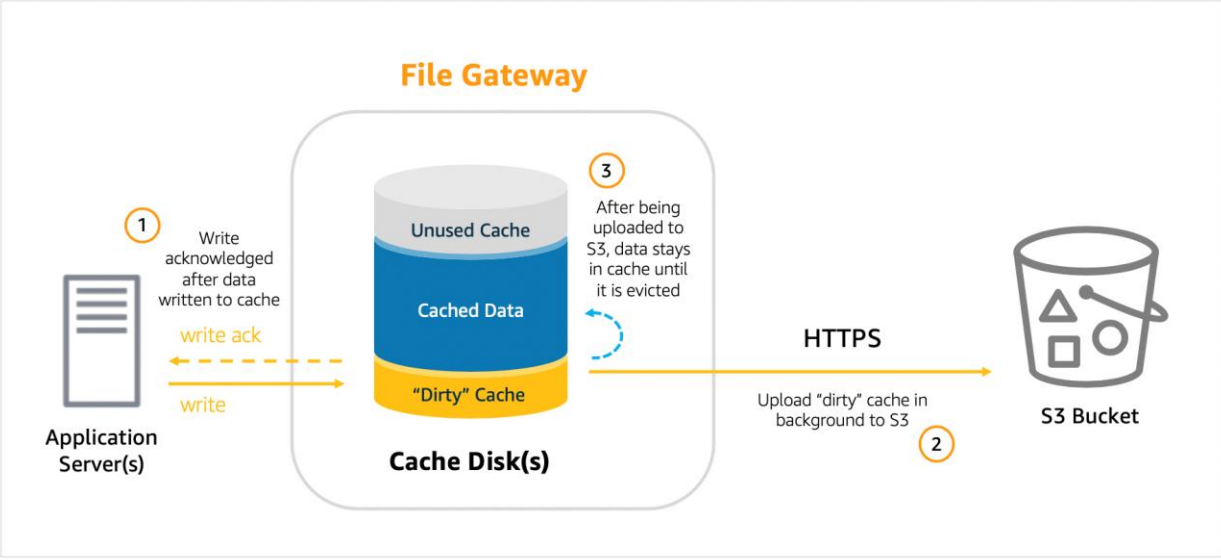
  (Incorrect)

- **Mount an Amazon EFS file system on the on-premises client and copy all backups to an NFS share.**

- **Create an SMB file share in Amazon FSx for Windows File Server that has enough storage to store all backups. Access the file share from on-premises.**

Explanation

Amazon S3 File Gateway presents a file interface that enables you to store files as objects in Amazon S3 using the industry-standard NFS and SMB file protocols, and access those files via NFS and SMB from your data center or Amazon EC2, or access those files as objects directly in Amazon S3.



When you deploy File Gateway, you specify how much disk space you want to allocate for local cache. This local cache acts as a buffer for writes and provides low latency access to data that was recently written to or read from Amazon S3. When a client writes data to a file via File Gateway, that data is first written to the local cache disk on the gateway itself. Once the data has been safely persisted to the local cache, only then does the File Gateway acknowledge the write back to the client. From there, File

Gateway transfers the data to the S3 bucket asynchronously in the background, optimizing data transfer using multipart parallel uploads and encrypting data in transit using HTTPS.

In this scenario, you can deploy an AWS Storage File Gateway to the on-premises client. After activating the File Gateway, create an SMB share and mount it as a local disk at the on-premises end. Copy the backups to the SMB share. You must ensure that you size the File Gateway's local cache appropriately to the backup data that needs immediate access. After the backup is done, you will be able to access the older data but with a delay. There will be a small delay since data (not in cache) needs to be retrieved from Amazon S3.

Hence, the correct answer is: **Use an AWS Storage File gateway with enough storage to keep data from the last 48 hours. Send the backups to an SMB share mounted as a local disk.**

The option that says: **Create an SMB file share in Amazon FSx for Windows File Server that has enough storage to store all backups. Access the file share from on-premises** is incorrect because this requires additional setup. You need to set up a Direct Connect or VPN connection from on-premises to AWS first in order for this to work.

The option that says: **Mount an Amazon EFS file system on the on-premises client and copy all backups to an NFS share** is incorrect because the file share required in the scenario needs to be using the SMB protocol.

The option that says: **Create an AWS Backup plan to copy data backups to a local SMB share every 48 hours** is incorrect. AWS Backup only works on AWS resources.

References:

https://aws.amazon.com/blogs/storage/easily-store-your-sql-server-backups-in-amazon-s3-using-file-gateway/

https://aws.amazon.com/storagegateway/faqs/

AWS Storage Gateway Overview:

https://youtu.be/pNb7xOBJjHE

Check out this AWS Storage Gateway Cheat Sheet:

https://tutorialsdojo.com/aws-storage-gateway/

Question 13: **Incorrect**
A company launched a cryptocurrency mining server on a Reserved EC2 instance in the us-east-1 region's private subnet that uses IPv6. Due to the financial data that the server contains, the system should be secured to prevent any unauthorized access and to meet regulatory compliance requirements.

In this scenario, which VPC feature will allow the EC2 instance to communicate to the Internet but prevents inbound IPv6 traffic?

- **Egress-only Internet gateway**

  (Correct)

- **NAT Gateway**

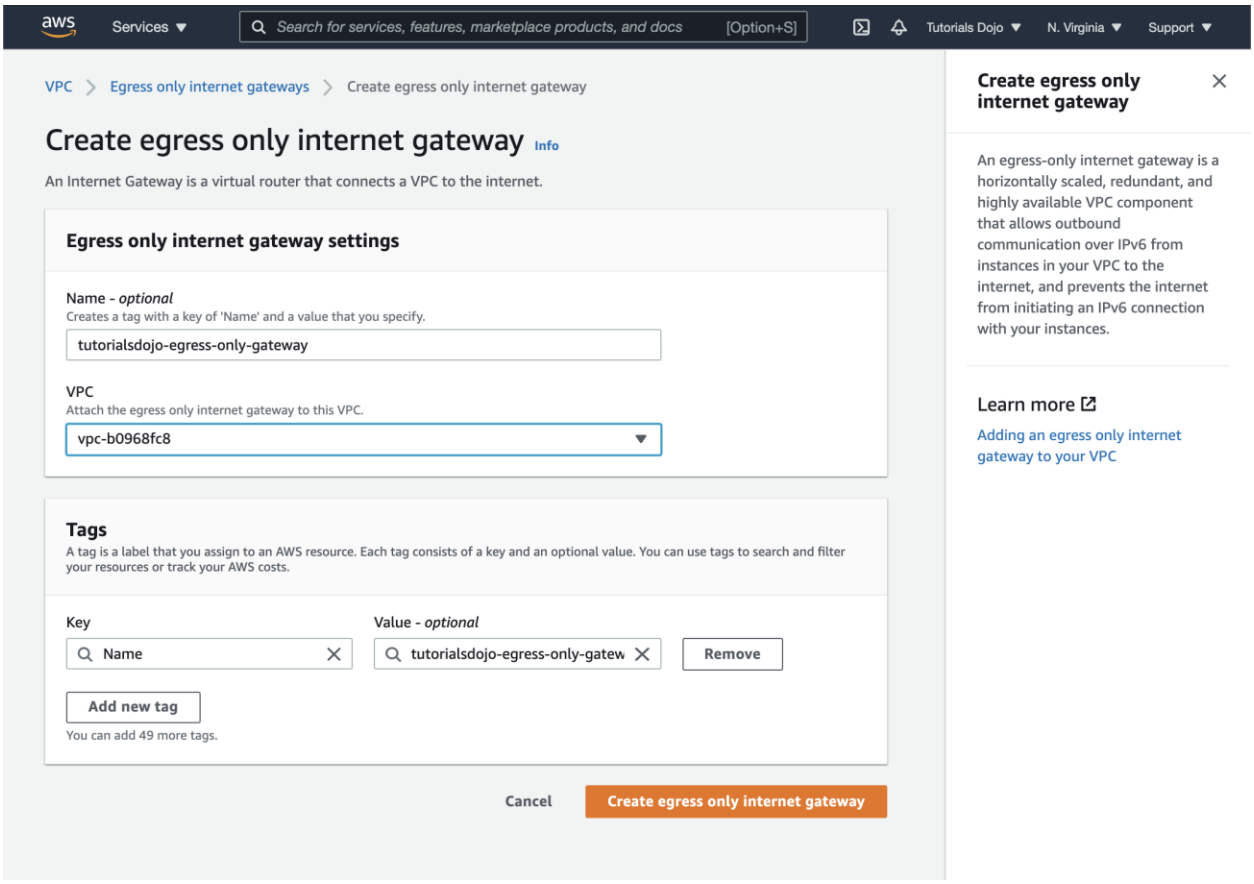
  (Incorrect)

- **NAT instances**

- **Internet Gateway**

Explanation
An **egress-only Internet gateway** is a horizontally scaled, redundant, and highly available VPC component that allows outbound communication over IPv6 from instances in your VPC to the Internet, and prevents the Internet from initiating an IPv6 connection with your instances.

Take note that an egress-only Internet gateway is for use with IPv6 traffic only. To enable outbound-only Internet communication over IPv4, use a NAT gateway instead.



Hence, the correct answer is: **Egress-only Internet gateway.**

NAT Gateway and NAT instances are incorrect. Although NAT64, a feature that enables the communication between IPv6 and IPv4 hosts, may be supported by these components, they are unable to block inbound IPv6 traffic. Both NAT 64 and egress-only gateways can be used to enable outbound Internet connectivity, for instances in a VPC, but they differ in their support for inbound traffic and their handling of IPv4 and IPv6 addresses. The most suitable VPC component to use is the scenario is the egress-only Internet gateway.

Internet Gateway is incorrect because this is primarily used to provide Internet access to your instances in the public subnet of your VPC and not for private subnets. However, with an Internet gateway, traffic originating from the public Internet will also be able to reach your instances. The scenario is asking you to prevent inbound access, so this is not the correct answer.

Reference:

https://docs.aws.amazon.com/vpc/latest/userguide/egress-only-internet-gateway.html

Amazon VPC Overview:

https://youtu.be/oIDHKeNxvQQ

Check out this Amazon VPC Cheat Sheet:

https://tutorialsdojo.com/amazon-vpc/

Question 14: **Correct**
A company is designing a customized text messaging service that targets its mobile app users. As part of its multi-engagement marketing campaign, a company needs to send a one-time confirmation message to all of its subscribers using Short Message Service (SMS). The solutions architect must design the system to allow a subscriber to reply to the SMS messages.

The customer responses must be kept for an entire year for analysis and targeted sale promotions. In addition, the SMS responses must also be collected, processed, and analyzed in near-real-time.

Which solution will meet these requirements with the LEAST operational overhead?

- **Set up an Amazon Connect contact flow to send the confirmation SMS messages to the mobile app users. Deploy an AWS Lambda function to process and analyze the responses. Store the data to Amazon S3 Glacier Flexible Retrieval**

- **Launch a new Amazon Simple Queue Service (Amazon SQS) queue to send out SMS messages. Use AWS Step Functions and AWS Lambda to collect, process, and analyze responses. Store the data to Amazon S3 Glacier Instant Retrieval.**

- **Create an Amazon Pinpoint journey for the multi-engagement SMS marketing campaign and an Amazon Kinesis Data Stream for analysis. Configure Amazon Pinpoint to send events to the Kinesis data stream for collection, processing, and analysis. Set the retention period of the Kinesis data stream to 365 days.**

  (Correct)

- 

Create a new topic in Amazon Simple Notification Service (Amazon SNS) and an Amazon Kinesis data stream configured with all its default settings. Send SMS messages using Amazon SNS. Integrate the Kinesis data stream to the SNS topic for data collection, archiving, and analysis.

Explanation

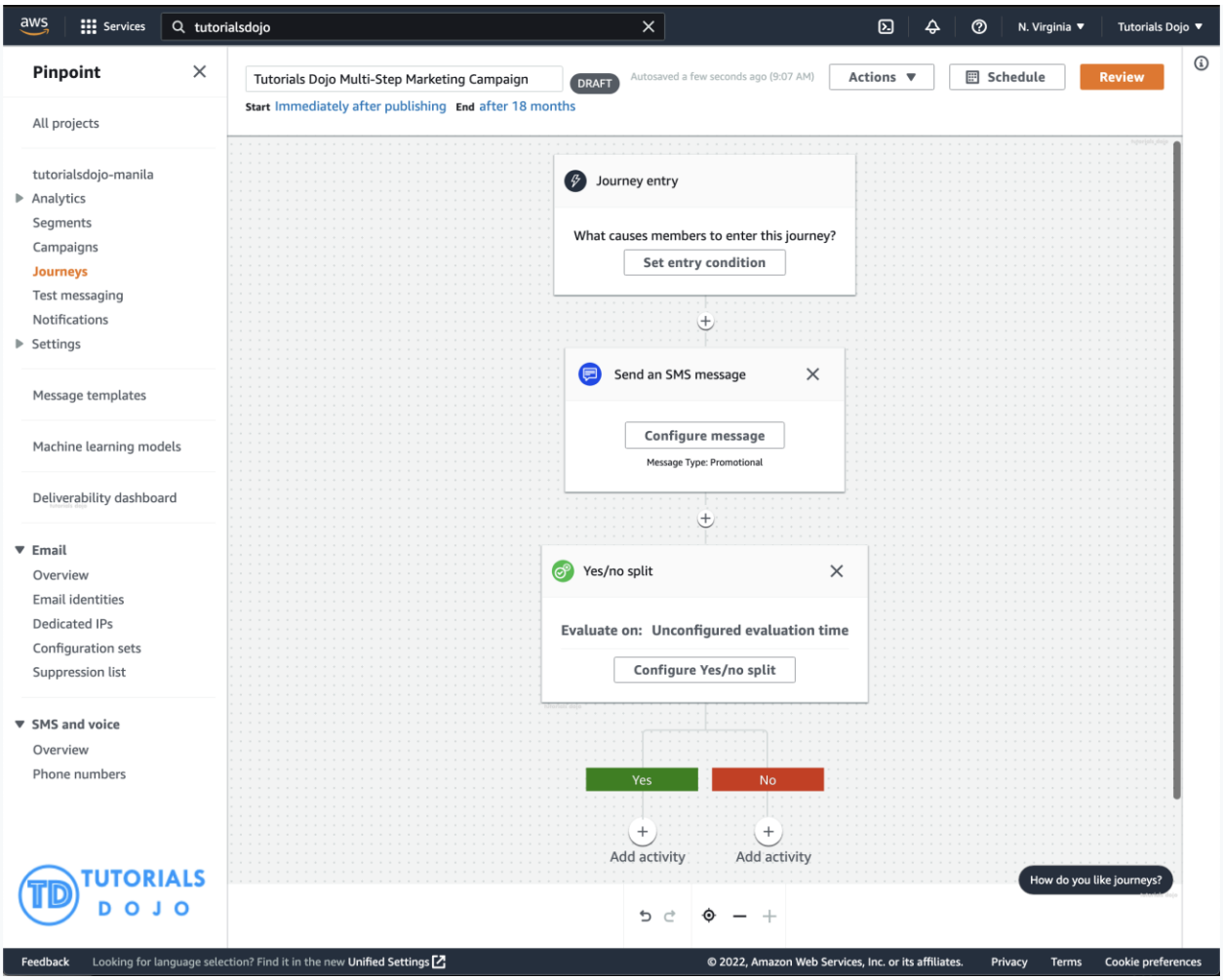
In Amazon Pinpoint, an *event* is an action that occurs when a user interacts with one of your applications, when you send a message from a campaign or journey, or when you send a transactional SMS or email message. For example, if you send an email message, several events occur:

- When you send the message, a *send* event occurs.

- When the message reaches the recipient's inbox, a *delivered* event occurs.

- When the recipient opens the message, an *open* event occurs.

You can configure Amazon Pinpoint to send information about events to Amazon Kinesis. The Kinesis platform offers services that you can use to collect, process, and analyze data from AWS services in real time.



Amazon Pinpoint can send event data to Kinesis Data Firehose, which streams this data to AWS data stores such as Amazon S3 or Amazon Redshift. Amazon Pinpoint can also stream data to Kinesis Data Streams, which ingests and stores multiple data streams for processing by analytics applications.

The Amazon Pinpoint event stream includes information about user interactions with applications (apps) that you connect to Amazon Pinpoint. It also includes information about all the messages that you send from campaigns, through any channel, and from journeys. This can also include any custom events that you've defined. Finally, it includes information about all the transactional email and SMS messages that you send.

Hence, the correct answer is: **Create an Amazon Pinpoint journey for the multi-engagement SMS marketing campaign and an Amazon Kinesis Data Stream for analysis. Configure Amazon Pinpoint to send events to the Kinesis data stream for collection, processing, and analysis. Set the retention period of the Kinesis data stream to 365 days.**

The option that says: **Create a new topic in Amazon Simple Notification Service (Amazon SNS) and an Amazon Kinesis data stream configured with all its default settings. Send SMS messages using Amazon SNS. Integrate the Kinesis data stream to the SNS topic for data collection, archiving, and analysis** is incorrect. Although Amazon SNS can send SMS to multiple users, it is not suitable for multi-engagement SMS marketing campaign. A better option is to use Amazon Pinpoint and create a custom journey for the multi-engagement campaign. Furthermore, the default retention period of Amazon Kinesis is only 24 hours and thus, this solution won't meet the requirements.

The option that says: **Launch a new Amazon Simple Queue Service (Amazon SQS) queue to send out SMS messages. Use AWS Step Functions and AWS Lambda to collect, process, and analyze responses. Store the data to Amazon S3 Glacier Instant Retrieval** is incorrect because using Amazon SQS alone is not capable of sending out SMS messages. In addition, the SMS marketing campaign is sent only once. It's better to process and store the responses in an Amazon Kinesis data stream and set its retention period to 1 year.

The option that says: **Set up an Amazon Connect contact flow to send the confirmation SMS messages to the mobile app users. Deploy an AWS Lambda function to process and analyze the responses. Store the data to Amazon S3 Glacier Flexible Retrieval** is incorrect. Using Amazon Connect to send a one-time SMS marketing campaign entails a lot of operational overhead to launch and maintain. The cost is significantly higher too. Take note that the scenario asks for a solution with the least amount of operational overhead.

References:

https://docs.aws.amazon.com/pinpoint/latest/developerguide/event-streams.html

https://docs.aws.amazon.com/pinpoint/latest/userguide/journeys-create.html

Question 15: **Correct**
A top university has recently launched its online learning portal where the students can take e-learning courses from the comforts of their homes. The portal is on a large On-Demand EC2 instance with a single Amazon Aurora database.

How can you improve the availability of your Aurora database to prevent any unnecessary downtime of the online portal?

- **Enable Hash Joins to improve the database query performance.**

- **Deploy Aurora to two Auto-Scaling groups of EC2 instances across two Availability Zones with an elastic load balancer which handles load balancing.**

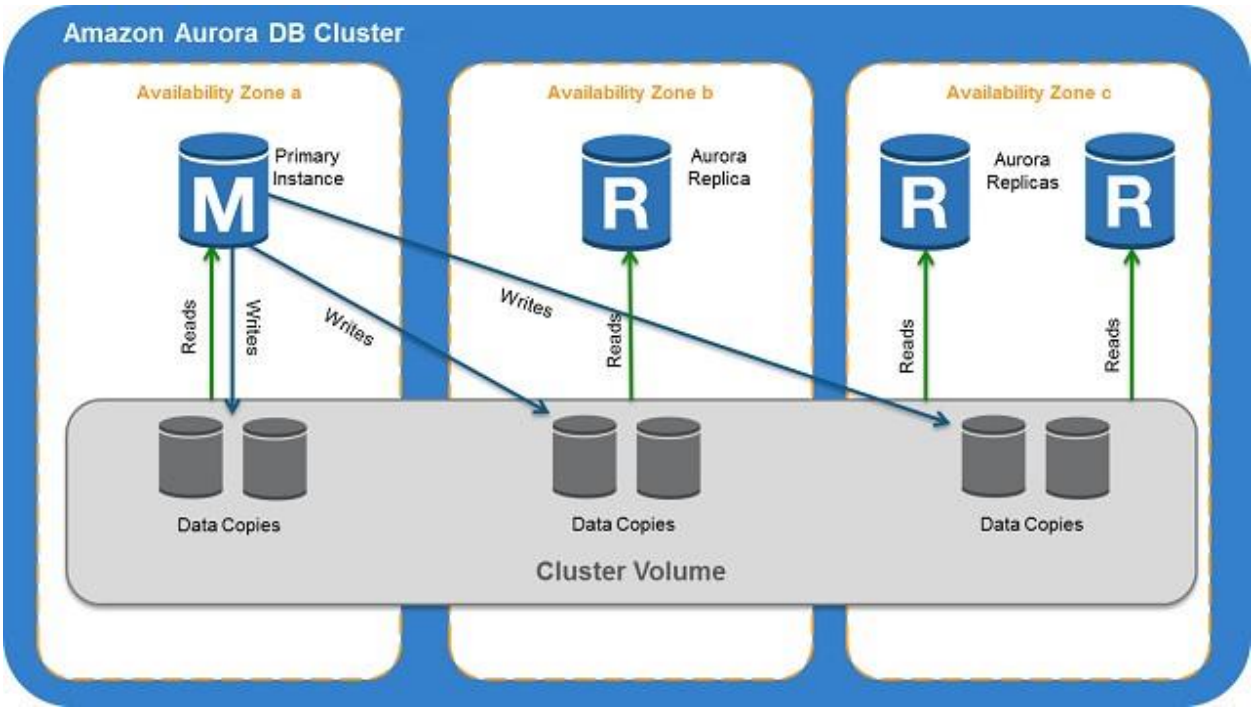
- **Create Amazon Aurora Replicas.**

  (Correct)

- **Use an Asynchronous Key Prefetch in Amazon Aurora to improve the performance of queries that join tables across indexes.**

Explanation

Amazon Aurora MySQL and **Amazon Aurora PostgreSQL** support Amazon Aurora Replicas, which share the same underlying volume as the primary instance. Updates made by the primary are visible to all Amazon Aurora Replicas. With Amazon Aurora MySQL, you can also create MySQL Read Replicas based on MySQL's binlog-based replication engine. In MySQL Read Replicas, data from your primary instance is replayed on your replica as transactions. For most use cases, including read scaling and high availability, it is recommended to use Amazon Aurora Replicas.



Read Replicas are primarily used for improving the read performance of the application. The most suitable solution in this scenario is to use Multi-AZ deployments instead, but since this option is not available, you can still set up Read Replicas which you can promote as your primary stand-alone DB cluster in the event of an outage.

Hence, the correct answer here is to **create Amazon Aurora Replicas**.

Deploying Aurora to two Auto-Scaling groups of EC2 instances across two Availability Zones with an elastic load balancer which handles load balancing is incorrect because Aurora is a managed database engine for RDS and not deployed on typical EC2 instances that you manually provision.

Enabling Hash Joins to improve the database query performance is incorrect because Hash Joins are mainly used if you need to join a large amount of data by using an equijoin and not for improving availability.

Using an Asynchronous Key Prefetch in Amazon Aurora to improve the performance of queries that join tables across indexes is incorrect because the Asynchronous Key Prefetch is mainly used to improve the performance of queries that join tables across indexes.

References:

https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/AuroraMySQL.BestPractices.html

https://aws.amazon.com/rds/aurora/faqs/

Amazon Aurora Overview:

https://youtu.be/iwS1h7rLNBQ

Check out this Amazon Aurora Cheat Sheet:

https://tutorialsdojo.com/amazon-aurora/

Question 16: **Correct**
A company deployed a fleet of Windows-based EC2 instances with IPv4 addresses launched in a private subnet. Several software installed in the EC2 instances are required to be updated via the Internet.

Which of the following services can provide the firm a highly available solution to safely allow the instances to fetch the software patches from the Internet but prevent outside network from initiating a connection?

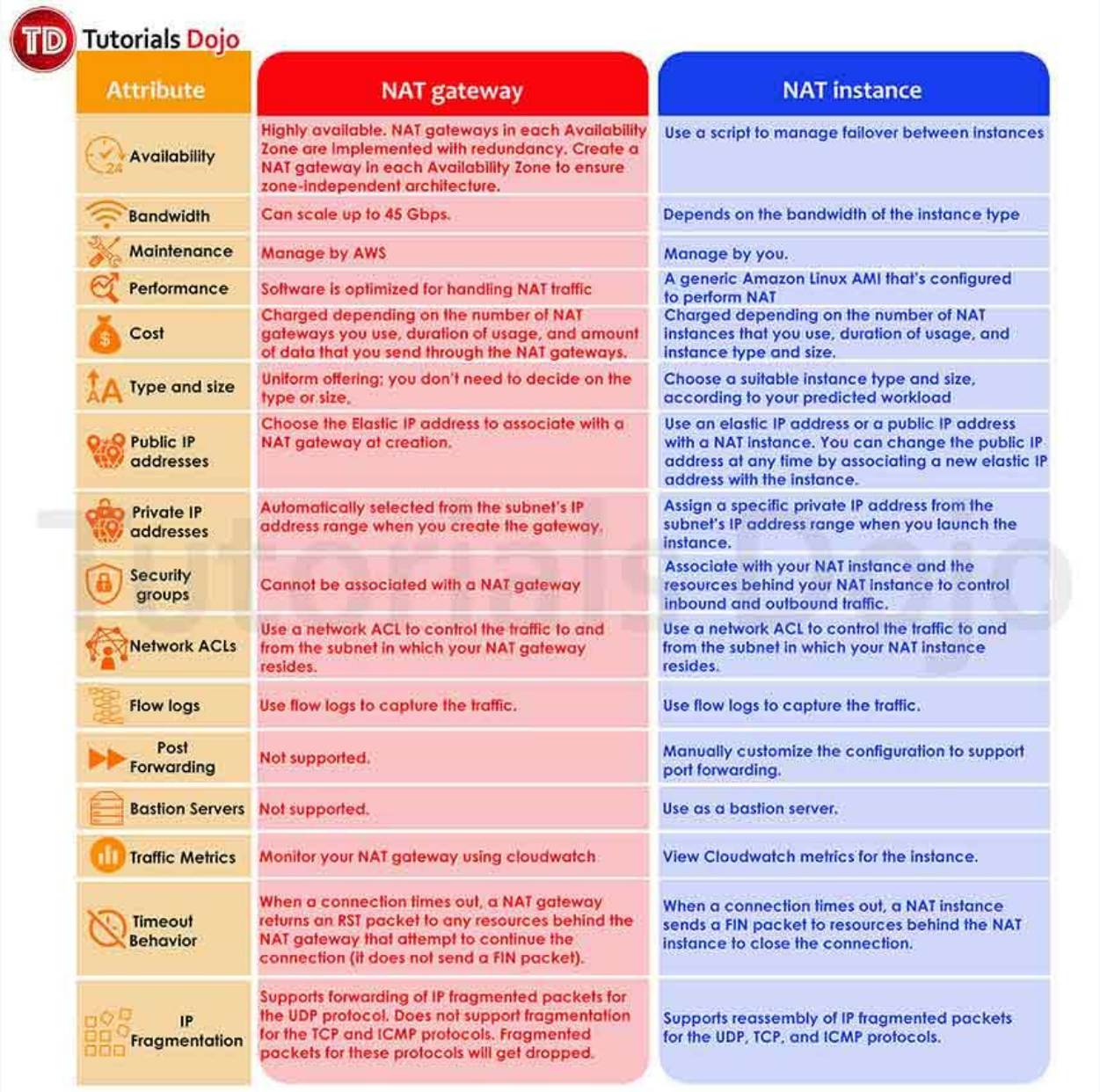
- **Egress-Only Internet Gateway**
- **VPC Endpoint**
- **NAT Instance**
- **NAT Gateway**

  (Correct)

Explanation
AWS offers two kinds of NAT devices — a NAT gateway or a NAT instance. It is recommended to use NAT gateways, as they provide better availability and bandwidth over NAT instances. The NAT Gateway service is also a managed service that does not require your administration efforts. A NAT instance is launched from a NAT AMI.

Just like a NAT instance, you can use a network address translation (NAT) gateway to enable instances in a private subnet to connect to the internet or other AWS services but prevent the internet from initiating a connection with those instances.

Here is a diagram showing the differences between NAT gateway and NAT instance:



Egress-Only Internet Gateway is incorrect because this is primarily used for VPCs that use IPv6 to enable instances in a private subnet to connect to the Internet or other AWS services but prevent the Internet from initiating a connection with those instances, just like what NAT Instance and NAT Gateway do. The scenario explicitly says that the EC2 instances are using IPv4 addresses which is why Egress-only Internet gateway is invalid, even though it can provide the required high availability.

VPC Endpoint is incorrect because this simply enables you to privately connect your VPC to supported AWS services and VPC endpoint services powered by PrivateLink without requiring an Internet gateway, NAT device, VPN connection, or AWS Direct Connect connection.

NAT Instance is incorrect. Although this can also enable instances in a private subnet to connect to the Internet or other AWS services and prevent the Internet from initiating a connection with those instances, it is not as highly available compared to a NAT Gateway.

References:

https://docs.aws.amazon.com/AmazonVPC/latest/UserGuide/vpc-nat-gateway.html

https://docs.aws.amazon.com/vpc/latest/userguide/vpc-nat-comparison.html

https://docs.aws.amazon.com/vpc/latest/userguide/egress-only-internet-gateway.html

Check out this Amazon VPC Cheat Sheet:

https://tutorialsdojo.com/amazon-vpc/

Question 17: **Incorrect**
A Solutions Architect is working for a weather station in Asia with a weather monitoring system that needs to be migrated to AWS. Since the monitoring system requires a low network latency and high network throughput, the Architect decided to launch the EC2 instances to a new cluster placement group. The system was working fine for a couple of weeks, however, when they try to add new instances to the placement group that already has running EC2 instances, they receive an 'insufficient capacity error'.

How will the Architect fix this issue?

- **Verify all running instances are of the same size and type and then try the launch again.**
- **Submit a capacity increase request to AWS as you are initially limited to only 12 instances per Placement Group.**
- **Create another Placement Group and launch the new instances in the new group.**

(Incorrect)

- **Stop and restart the instances in the Placement Group and then try the launch again.**

  (Correct)

Explanation
A cluster placement group is a logical grouping of instances within a single Availability Zone. A cluster placement group can span peered VPCs in the same Region. Instances in the same cluster placement group enjoy a higher per-flow throughput limit for TCP/IP traffic and are placed in the same high-bisection bandwidth segment of the network.

It is recommended that you launch the number of instances that you need in the placement group in a single launch request and that you use the same instance type for all instances in the placement group. If you try to add more instances to the placement group later, or if you try to launch more than one instance type in the placement group, you increase your chances of getting an insufficient capacity error. If you stop an

instance in a placement group and then start it again, it still runs in the placement group. However, the start fails if there isn't enough capacity for the instance.

If you receive a capacity error when launching an instance in a placement group that already has running instances, stop and start all of the instances in the placement group, and try the launch again. Restarting the instances may migrate them to hardware that has capacity for all the requested instances.

Stop and restart the instances in the Placement group and then try the launch again can resolve this issue. If the instances are stopped and restarted, AWS may move the instances to a hardware that has the capacity for all the requested instances.

Hence, the correct answer is: **Stop and restart the instances in the Placement Group and then try the launch again.**

The option that says: **Create another Placement Group and launch the new instances in the new group** is incorrect because to benefit from the enhanced networking, all the instances should be in the same Placement Group. Launching the new ones in a new Placement Group will not work in this case.

The option that says: **Verify all running instances are of the same size and type and then try the launch again** is incorrect because the capacity error is not related to the instance size.

The option that says: **Submit a capacity increase request to AWS as you are initially limited to only 12 instances per Placement Group** is incorrect because there is no such limit on the number of instances in a Placement Group.

References:

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/placement-groups.html#placement-groups-cluster

http://docs.amazonaws.cn/en_us/AWSEC2/latest/UserGuide/troubleshooting-launch.html#troubleshooting-launch-capacity

Check out this Amazon EC2 Cheat Sheet:

https://tutorialsdojo.com/amazon-elastic-compute-cloud-amazon-ec2/

Question 18: **Correct**

A company has a VPC for its Human Resource department and another VPC located in different AWS regions for its Finance department. The Solutions Architect must redesign the architecture to allow the finance department to access all resources that are in the human resource department, and vice versa. An Intrusion Prevention System (IPS) must also be integrated for active traffic flow inspection and to block any vulnerability exploits.

Which network architecture design in AWS should the Solutions Architect set up to satisfy the above requirement?

- **Create a Traffic Policy in Amazon Route 53 to connect the two VPCs. Configure the Route 53 Resolver DNS Firewall to do active traffic flow inspection and block any vulnerability exploits.**

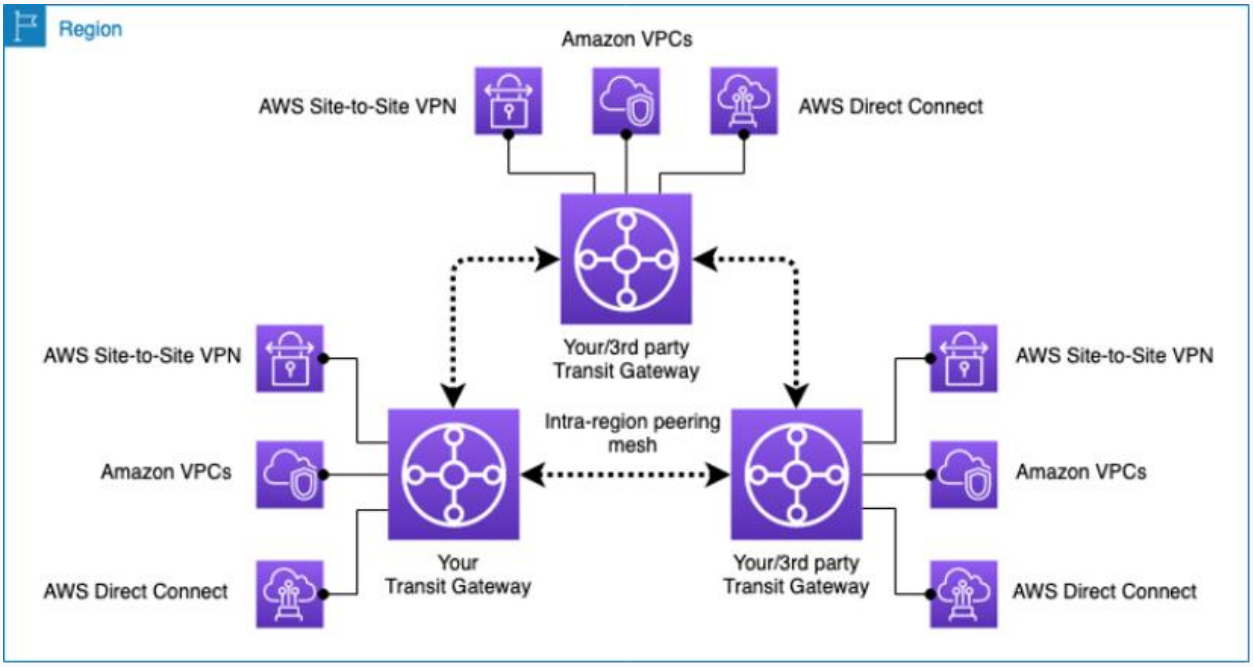
- **Establish a secure connection between the two VPCs using a NAT Gateway. Manage user sessions via the AWS Systems Manager Session Manager service.**

- **Create a Direct Connect Gateway and add VPC attachments to connect all departments. Configure AWS Security Hub to secure the application traffic travelling between the VPCs.**

- **Launch an AWS Transit Gateway and add VPC attachments to connect all departments. Set up AWS Network Firewall to secure the application traffic travelling between the VPCs.**

  (Correct)

Explanation
A *transit gateway* is a network transit hub that you can use to interconnect your virtual private clouds (VPCs) and on-premises networks. As your cloud infrastructure expands globally, inter-Region peering connects transit gateways together using the AWS Global Infrastructure. Your data is automatically encrypted and never travels over the public internet.



A transit gateway attachment is both a source and a destination of packets. You can attach the following resources to your transit gateway:

- One or more VPCs.

- One or more VPN connections

- One or more AWS Direct Connect gateways

- One or more Transit Gateway Connect attachments

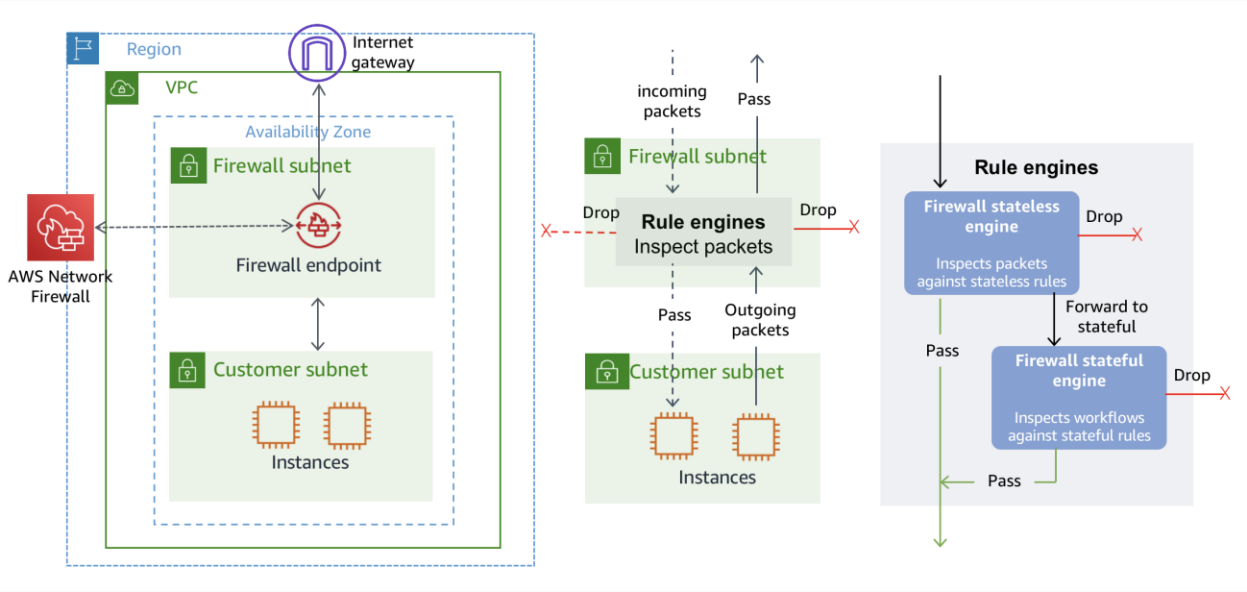
- One or more transit gateway peering connections

AWS Transit Gateway deploys an elastic network interface within VPC subnets, which is then used by the transit gateway to route traffic to and from the chosen subnets. You must have at least one subnet for each Availability Zone, which then enables traffic to reach resources in every subnet of that zone. During attachment creation, resources within a particular Availability Zone can reach a transit gateway only if a subnet is enabled within the same zone. If a subnet route table includes a route to the transit gateway, traffic is only forwarded to the transit gateway if the transit gateway has an attachment in the subnet of the same Availability Zone.

Intra-region peering connections are supported. You can have different transit gateways in different Regions.

AWS Network Firewall is a managed service that makes it easy to deploy essential network protections for all of your Amazon Virtual Private Clouds (VPCs). The service

can be setup with just a few clicks and scales automatically with your network traffic, so you don't have to worry about deploying and managing any infrastructure. AWS Network Firewall's flexible rules engine lets you define firewall rules that give you fine-grained control over network traffic, such as blocking outbound Server Message Block (SMB) requests to prevent the spread of malicious activity.



AWS Network Firewall includes features that provide protections from common network threats. AWS Network Firewall's stateful firewall can incorporate context from traffic flows, like tracking connections and protocol identification, to enforce policies such as preventing your VPCs from accessing domains using an unauthorized protocol. AWS Network Firewall's intrusion prevention system (IPS) provides active traffic flow inspection so you can identify and block vulnerability exploits using signature-based detection. AWS Network Firewall also offers web filtering that can stop traffic to known bad URLs and monitor fully qualified domain names.

Hence, the correct answer is: **Launch a Transit Gateway and add VPC attachments to connect all departments. Set up AWS Network Firewall to secure the application traffic travelling between the VPCs.**

The option that says: **Create a Traffic Policy in Amazon Route 53 to connect the two VPCs. Configure the Route 53 Resolver DNS Firewall to do active traffic flow inspection and block any vulnerability exploits** is incorrect because the Traffic Policy feature is commonly used in tandem with the geoproximity routing policy for creating and maintaining records in large and complex configurations. Moreover, the Route 53 Resolver DNS Firewall can only filter and regulate outbound DNS traffic for your virtual private cloud (VPC). It can neither do active traffic flow inspection nor block any vulnerability exploits.

The option that says: **Establish a secure connection between the two VPCs using a NAT Gateway. Manage user sessions via the AWS Systems Manager Session Manager service** is incorrect because a NAT Gateway is simply a Network Address Translation (NAT) service and can't be used to connect two VPCs in different AWS regions. This service allows your instances in a private subnet to connect to services outside your VPC but external services cannot initiate a connection with those instances. Furthermore, the AWS Systems Manager Session Manager service is meant for managing EC2 instances via remote SSH or PowerShell access. This is not used for managing user sessions.

The option that says: **Create a Direct Connect Gateway and add VPC attachments to connect all departments. Configure AWS Security Hub to secure the application traffic travelling between the VPCs** is incorrect. An AWS Direct Connect gateway is meant to be used in conjuction with an AWS Direct Connect connection to your on-premises network to connect with a Transit Gateway or a Virtual Private Gateway. You still need a Transit Gateway to connect the two VPCs that are in different AWS Regions. The AWS Security Hub is simply a cloud security posture management service that automates best practice checks, aggregates alerts, and supports automated remediation. It's important to note that it doesn't secure application traffic just by itself.

References:

https://docs.aws.amazon.com/AmazonVPC/latest/UserGuide/vpc-peering.html

https://aws.amazon.com/transit-gateway

https://aws.amazon.com/network-firewall

Check out these Amazon VPC and VPC Peering Cheat Sheets:

https://tutorialsdojo.com/amazon-vpc/

https://tutorialsdojo.com/vpc-peering/

Question 19: **Correct**

A company has multiple AWS sandbox accounts that are used by its development team. All developers must be given access to the contents of one of the main account's S3 buckets. For security purposes, any personally identifiable information (PII) or financial data uploaded in the bucket must be continuously monitored and removed.

How can this be done at the lowest possible cost and with the least amount of configuration effort?

- **Generate a pre-signed URL for the objects on the S3 bucket. Use the Amazon S3 Storage Lens to discover personally identifiable information (PII) or financial data.**

- **Configure cross-account replication on the S3 bucket. Integrate AWS Audit Manager with the S3 bucket to discover any personally identifiable information (PII) or financial data.**

- **Create an S3 bucket policy that grants access from the sandbox accounts. Use Amazon Macie to discover personally identifiable information (PII) or financial data.**

  (Correct)

- **Add S3 read permission to the IAM policy of each IAM user from the sandbox accounts. Use Amazon Detective to discover personally identifiable information (PII) or financial data.**

Explanation

In Amazon S3, you can grant users in another AWS account (Account B) granular cross-account access to objects owned by your account (Account A). Depending on the type of access that you want to provide, use one of the following solutions to grant cross-account access to objects:

- AWS Identity and Access Management (IAM) policies and resource-based bucket policies (for programmatic-only access to S3 bucket objects

- IAM policies and resource-based Access Control Lists (ACLs) for programmatic-only access to S3 bucket objects

- Cross-account IAM roles for programmatic and console access to S3 bucket objects.

Not all AWS services support resource-based policies. Therefore, you can use cross-account IAM roles to centralize permission management when providing cross-account access to multiple services. Using cross-account IAM roles simplifies provisioning

cross-account access to S3 objects that are stored in multiple S3 buckets. As a result, you don't need to manage multiple policies for S3 buckets. This method allows cross-account access to objects owned or uploaded by another AWS account or AWS services. If you don't use cross-account IAM roles, then the object ACL must be modified.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": ["arn:aws:iam::111111111111:user/dev1",
                        "arn:aws:iam::222222222222:user/dev2",
                        "arn:aws:iam::333333333333:user/dev3"]
            },
            "Action": "s3:GetObject",
            "Resource": [
                "arn:aws:s3:::MainBucket/*"
            ]
        }
    ]
}
```

In the scenario, the best approach to granting the developers access to the main account's S3 bucket is by configuring the bucket policy to allow IAM users from different accounts to call the GetObject method. This is a neater and simpler solution than the rest because you control access from a single location without any additional costs.

Hence, the correct answer is: **Create an S3 bucket policy that grants access from the sandbox accounts. Use Amazon Macie to discover personally identifiable information (PII) or financial data.**

The option that says: **Configure cross-account replication on the S3 bucket. Integrate AWS Audit Manager with the S3 bucket to discover any personally identifiable information (PII) or financial data** is incorrect. This can work, but it is an inefficient way of solving the problem. The developers only need to access the S3 objects in another account; they do not need to own a copy of them. On top of that, replication incurs additional costs. In addition, the AWS Audit Manager simply helps you continuously audit your AWS usage to simplify how you assess risk and compliance with regulations and industry standards. AWS Audit Manager is not capable of discovering personally identifiable information (PII) or financial data in your S3 bucket.

The option that says: **Generate a pre-signed URL for the objects on the S3 bucket. Use the Amazon S3 Storage Lens to discover personally identifiable information (PII) or financial data** is incorrect. Since objects shared using presigned URLs are time-limited, you'd have to regenerate the URL for each object every time it expires and resend the new link to the developers. This approach does not scale well and is not a good use for the S3 presigned URL. Moreover, the Amazon S3 Storage Lens feature just provides a single view of object storage usage and activity across your entire Amazon S3 storage.

The option that says: **Add S3 read permission to the IAM policy of each IAM user from the sandbox accounts. Use Amazon Detective to discover personally identifiable information (PII) or financial data** is incorrect. You would have to jump from one account to another to set this up. It works, but depending on the number of accounts and IAM users, it will entail a lot of configuration overhead. Although Amazon Detective is a security service, it does not have any capability to discover any PII or financial data in your S3 bucket. Its primary purpose is to analyze and visualize security data to rapidly get to the root cause of potential security issues.

References:

https://aws.amazon.com/premiumsupport/knowledge-center/cross-account-access-s3/

https://docs.aws.amazon.com/AmazonS3/latest/userguide/example-walkthroughs-managing-access-example2.html

https://aws.amazon.com/macie/

Check out this Amazon S3 Cheat Sheet:

https://tutorialsdojo.com/amazon-s3/

Question 20: **Correct**
A startup is building a microservices architecture in which the software is composed of small independent services that communicate over well-defined APIs. In building large-scale systems, fine-grained decoupling of microservices is a recommended practice to implement. The decoupled services should scale horizontally from each other to improve scalability.

What is the difference between Horizontal scaling and Vertical scaling?

- 

Horizontal scaling means running the same software on smaller containers such as Docker and Kubernetes using ECS or EKS. Vertical scaling is adding more servers to the existing pool and doesn't run into limitations of individual servers.

- **Vertical scaling means running the same software on bigger machines which is limited by the capacity of the individual server. Horizontal scaling is adding more servers to the existing pool and doesn't run into limitations of individual servers.**

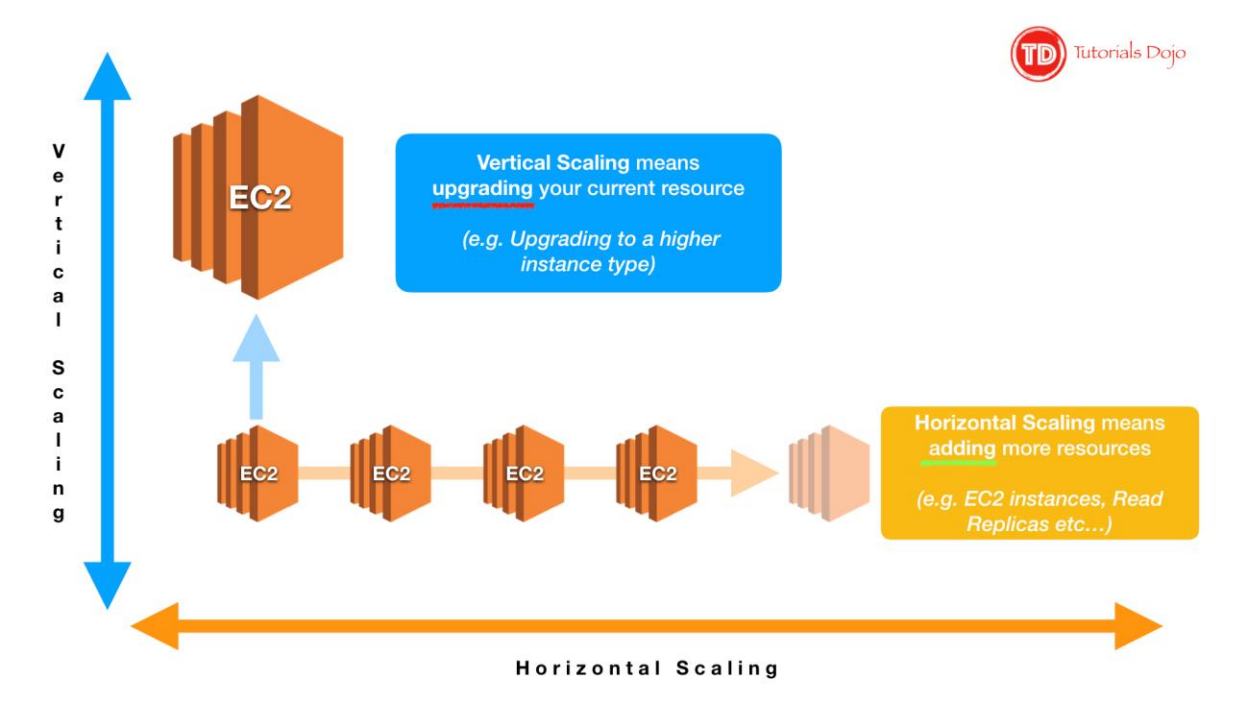
  (Correct)

- **Vertical scaling means running the same software on a fully serverless architecture using Lambda. Horizontal scaling means adding more servers to the existing pool and it doesn't run into limitations of individual servers.**

- **Horizontal scaling means running the same software on bigger machines which is limited by the capacity of individual servers. Vertical scaling is adding more servers to the existing pool and doesn't run into limitations of individual servers.**

Explanation

Vertical scaling means running the same software on bigger machines which is limited by the capacity of the individual server. Horizontal scaling is adding more servers to the existing pool and doesn't run into limitations of individual servers.



Fine-grained decoupling of microservices is a best practice for building large-scale systems. It's a prerequisite for performance optimization since it allows choosing the appropriate and optimal technologies for a specific service. Each service can be implemented with the appropriate programming languages and frameworks, leverage the optimal data persistence solution, and be fine-tuned with the best-performing service configurations.

Properly decoupled services can be scaled horizontally and independently from each other. Vertical scaling, which is running the same software on bigger machines, is limited by the capacity of individual servers and can incur downtime during the scaling process. Horizontal scaling, which is adding more servers to the existing pool, is highly dynamic and doesn't run into limitations of individual servers. The scaling process can be completely automated.

Furthermore, the resiliency of the application can be improved because failing components can be easily and automatically replaced. Hence, the correct answer is the option that says: **Vertical scaling means running the same software on bigger machines which is limited by the capacity of the individual server. Horizontal scaling is adding more servers to the existing pool and doesn't run into limitations of individual servers.**

The option that says: **Vertical scaling means running the same software on a fully serverless architecture using Lambda. Horizontal scaling means adding more servers to the existing pool and it doesn't run into limitations of individual servers** is incorrect

because Vertical scaling is not about running the same software on a fully serverless architecture. AWS Lambda is not required for scaling.

The option that says: **Horizontal scaling means running the same software on bigger machines which is limited by the capacity of individual servers. Vertical scaling is adding more servers to the existing pool and doesn't run into limitations of individual servers** is incorrect because the definitions for the two concepts were switched. Vertical scaling means running the same software on bigger machines which is limited by the capacity of the individual server. Horizontal scaling is adding more servers to the existing pool and doesn't run into limitations of individual servers.

The option that says: **Horizontal scaling means running the same software on smaller containers such as Docker and Kubernetes using ECS or EKS. Vertical scaling is adding more servers to the existing pool and doesn't run into limitations of individual servers** is incorrect because Horizontal scaling is not related to using ECS or EKS containers on a smaller instance.

Reference:

https://docs.aws.amazon.com/aws-technical-content/latest/microservices-on-aws/microservices-on-aws.pdf#page=8

Tutorials Dojo's AWS Certified Solutions Architect Associate Exam Study Guide:

https://tutorialsdojo.com/aws-certified-solutions-architect-associate/

Question 21: **Incorrect**
A manufacturing company launched a new type of IoT sensor. The sensor will be used to collect large streams of data records. You need to create a solution that can ingest and analyze the data in real-time with millisecond response times.

Which of the following is the best option that you should implement in this scenario?

- **Ingest the data using Amazon Kinesis Data Streams and create an AWS Lambda function to store the data in Amazon DynamoDB.**

  (Correct)

-

Ingest the data using Amazon Simple Queue Service and create an AWS Lambda function to store the data in Amazon Redshift.

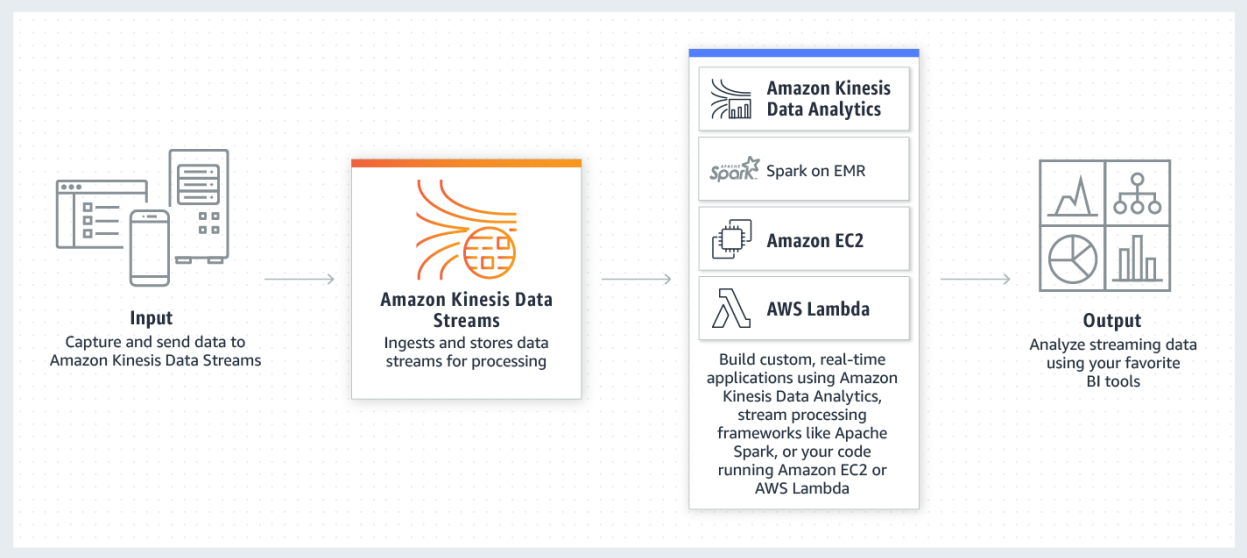
- **Ingest the data using Amazon Kinesis Data Streams and create an AWS Lambda function to store the data in Amazon Redshift.**

  (Incorrect)

- **Ingest the data using Amazon Kinesis Data Firehose and create an AWS Lambda function to store the data in Amazon DynamoDB.**

Explanation

Amazon Kinesis Data Streams enables you to build custom applications that process or analyze streaming data for specialized needs. You can continuously add various types of data such as clickstreams, application logs, and social media to an Amazon Kinesis data stream from hundreds of thousands of sources. Within seconds, the data will be available for your Amazon Kinesis Applications to read and process from the stream.



Based on the given scenario, the key points are "ingest and analyze the data in real-time" and "millisecond response times". For the first key point and based on the given options, you can use Amazon Kinesis Data Streams because it can collect and process large streams of data records in real time. For the second key point, you should use Amazon DynamoDB since it supports single-digit millisecond response times at any scale.

Hence, the correct answer is: **Ingest the data using Amazon Kinesis Data Streams and create an AWS Lambda function to store the data in Amazon DynamoDB.**

The option that says: **Ingest the data using Amazon Kinesis Data Streams and create an AWS Lambda function to store the data in Amazon Redshift** is incorrect because Amazon Redshift only delivers sub-second response times. Take note that as per the scenario, the solution must have millisecond response times to meet the requirements. Amazon DynamoDB Accelerator (DAX), which is a fully managed, highly available, in-memory cache for Amazon DynamoDB, can deliver microsecond response times.

The option that says: **Ingest the data using Amazon Kinesis Data Firehose and create an AWS Lambda function to store the data in Amazon DynamoDB** is incorrect. Amazon Kinesis Data Firehose only supports Amazon S3, Amazon Redshift, Amazon Elasticsearch, and an HTTP endpoint as the destination.

The option that says: **Ingest the data using Amazon Simple Queue Service and create an AWS Lambda function to store the data in Amazon Redshift** is incorrect because Amazon SQS can't analyze data in real time. You have to use an Amazon Kinesis Data Stream to process the data in near-real-time.

References:

https://aws.amazon.com/kinesis/data-streams/faqs/

https://aws.amazon.com/dynamodb/

Check out this Amazon Kinesis Cheat Sheet:

https://tutorialsdojo.com/amazon-kinesis/

Question 22: **Incorrect**
A Solutions Architect created a brand new IAM User with a default setting using AWS CLI. This is intended to be used to send API requests to Amazon S3, DynamoDB, Lambda, and other AWS resources of the company's cloud infrastructure.

Which of the following must be done to allow the user to make API calls to the AWS resources?

- **Do nothing as the IAM User is already capable of sending API calls to your AWS resources.**

Enable Multi-Factor Authentication for the user.

- 

Assign an IAM Policy to the user to allow it to send API calls.

(Incorrect)

- 

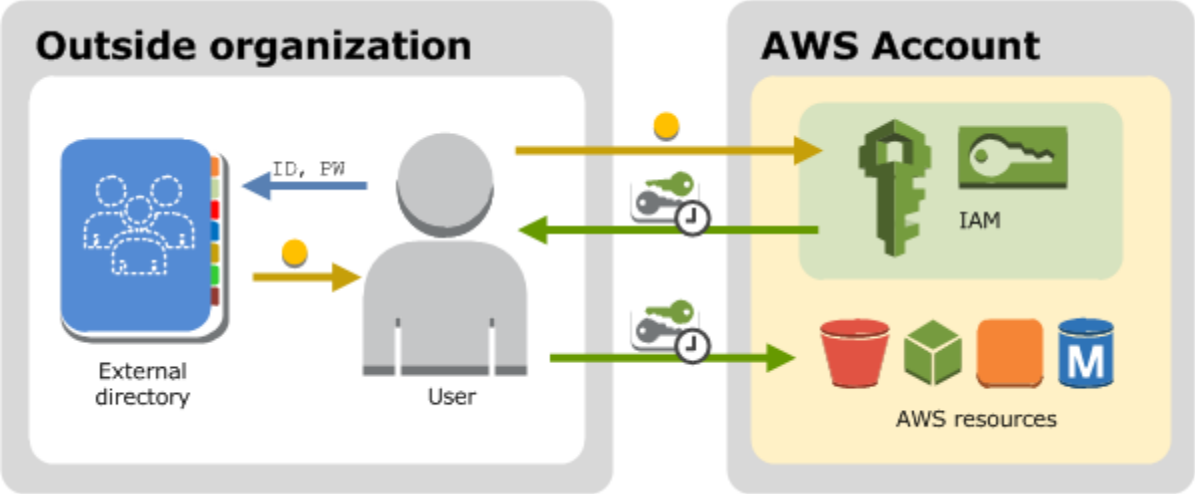
Create a set of Access Keys for the user and attach the necessary permissions.

(Correct)

Explanation
You can choose the credentials that are right for your IAM user. When you use the AWS Management Console to create a user, you must choose to include at least a console password or access keys. By default, a brand new IAM user created using the AWS CLI or AWS API has no credentials of any kind. You must create the type of credentials for an IAM user based on the needs of your user.

Access keys are long-term credentials for an IAM user or the AWS account root user. You can use access keys to sign programmatic requests to the AWS CLI or AWS API (directly or using the AWS SDK). Users need their own access keys to make programmatic calls to AWS from the AWS Command Line Interface (AWS CLI), Tools for Windows PowerShell, the AWS SDKs, or direct HTTP calls using the APIs for individual AWS services.

To fill this need, you can create, modify, view, or rotate access keys (access key IDs and secret access keys) for IAM users. When you create an access key, IAM returns the access key ID and secret access key. You should save these in a secure location and give them to the user.



The option that says: **Do nothing as the IAM User is already capable of sending API calls to your AWS resources** is incorrect because by default, a brand new IAM user created using the AWS CLI or AWS API has no credentials of any kind. Take note that in the scenario, you created the new IAM user using the AWS CLI and not via the AWS Management Console, where you must choose to at least include a console password or access keys when creating a new IAM user.

Enabling Multi-Factor Authentication for the user is incorrect because this will still not provide the required Access Keys needed to send API calls to your AWS resources. You have to grant the IAM user with Access Keys to meet the requirement.

Assigning an IAM Policy to the user to allow it to send API calls is incorrect because adding a new IAM policy to the new user will not grant the needed Access Keys needed to make API calls to the AWS resources.

References:

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_access-keys.html

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_users.html#id_users_creds

Check out this AWS IAM Cheat Sheet:

https://tutorialsdojo.com/aws-identity-and-access-management-iam/

Question 23: **Incorrect**

A cryptocurrency company wants to go global with its international money transfer app. Your project is to make sure that the database of the app is highly available in multiple regions.

What are the benefits of adding Multi-AZ deployments in Amazon RDS? (Select TWO.)

- ☑

  Provides enhanced database durability in the event of a DB instance component failure or an Availability Zone outage.

  (Correct)

- ☐

  Significantly increases the database performance.

- ☐

  Increased database availability in the case of system upgrades like OS patching or DB Instance scaling.

  (Correct)

- ☑

  Creates a primary DB Instance and synchronously replicates the data to a standby instance in a different Availability Zone (AZ) in a different region.

  (Incorrect)

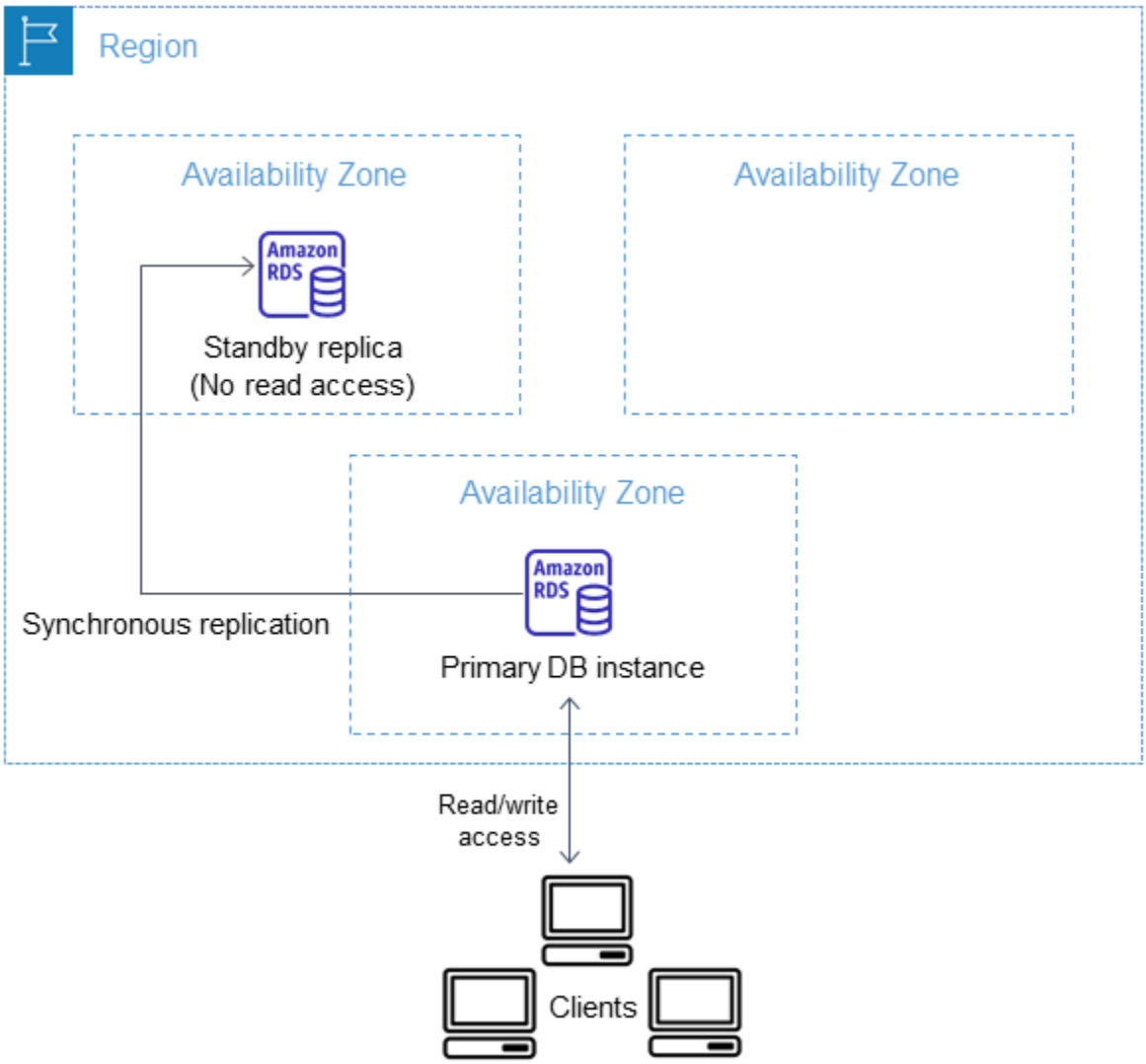
- ☐

  Provides SQL optimization.

Explanation
Amazon RDS Multi-AZ deployments provide enhanced availability and durability for Database (DB) Instances, making them a natural fit for production database workloads. When you provision a Multi-AZ DB Instance, Amazon RDS automatically creates a primary DB Instance and synchronously replicates the data to a standby instance in a different Availability Zone (AZ). Each AZ runs on its own physically distinct, independent infrastructure and is engineered to be highly reliable.

In case of an infrastructure failure, Amazon RDS performs an automatic failover to the standby (or to a read replica in the case of Amazon Aurora), so that you can resume

database operations as soon as the failover is complete. Since the endpoint for your DB Instance remains the same after a failover, your application can resume database operation without the need for manual administrative intervention.



The chief benefits of running your DB instance as a Multi-AZ deployment are enhanced database durability and availability. The increased availability and fault tolerance offered by Multi-AZ deployments make them a natural fit for production environments.

Hence, the correct answers are the following options:

- Increased database availability in the case of system upgrades like OS patching or DB Instance scaling.

- Provides enhanced database durability in the event of a DB instance component failure or an Availability Zone outage.

The option that says: **Creates a primary DB Instance and synchronously replicates the data to a standby instance in a different Availability Zone (AZ) in a different region** is incorrect. RDS synchronously replicates the data to a standby instance in a different Availability Zone (AZ) that is in the same region and not in a different one.

The options that say: **Significantly increases the database performance and Provides SQL optimization** are incorrect as it does not affect the performance nor provide SQL optimization.

References:

https://aws.amazon.com/rds/details/multi-az/

https://aws.amazon.com/rds/faqs/

Check out this Amazon RDS Cheat Sheet:

https://tutorialsdojo.com/amazon-relational-database-service-amazon-rds/

Question 24: **Correct**
A company is using an Amazon RDS for MySQL 5.6 with Multi-AZ deployment enabled and several web servers across two AWS Regions. The database is currently experiencing highly dynamic reads due to the growth of the company's website. The Solutions Architect tried to test the read performance from the secondary AWS Region and noticed a notable slowdown on the SQL queries.

Which of the following options would provide a read replication latency of less than 1 second?

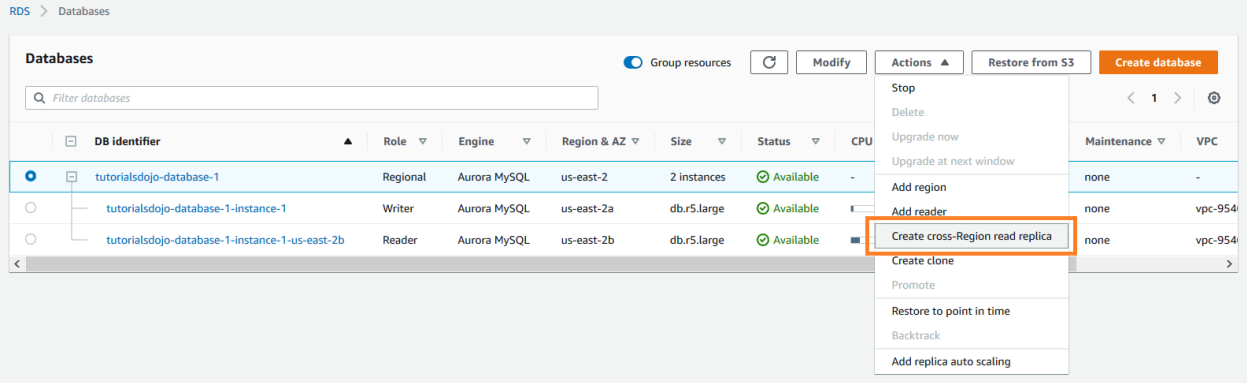
- **Upgrade the MySQL database engine.**
- **Use Amazon ElastiCache to improve database performance.**
- **Migrate the existing database to Amazon Aurora and create a cross-region read replica.**

(Correct)

- **Create an Amazon RDS for MySQL read replica in the secondary AWS Region.**

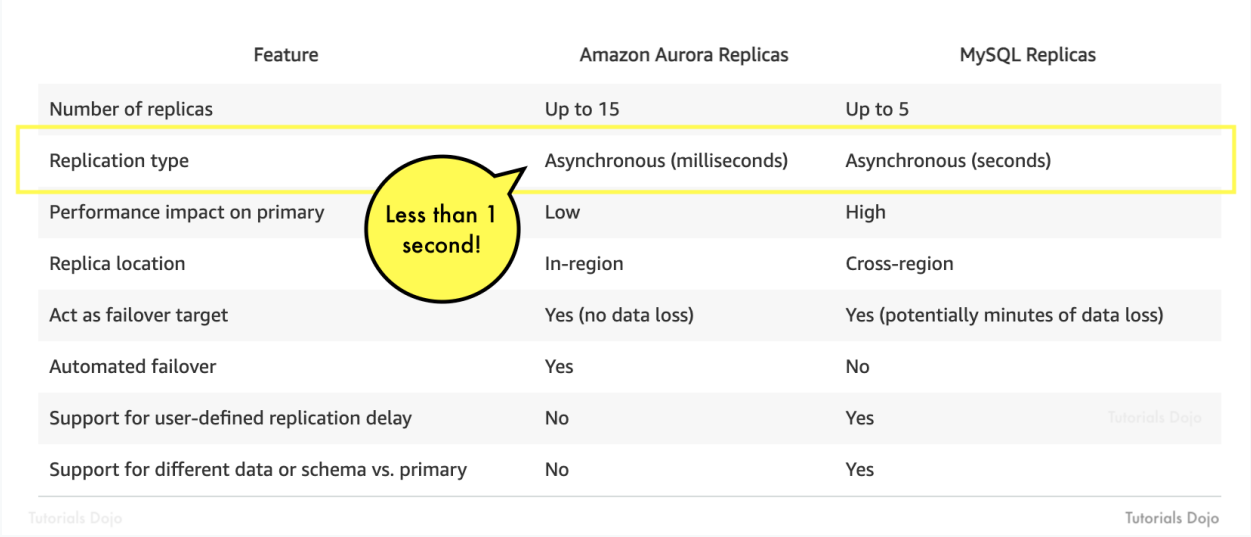
Explanation

Amazon Aurora is a MySQL and PostgreSQL-compatible relational database built for the cloud, that combines the performance and availability of traditional enterprise databases with the simplicity and cost-effectiveness of open source databases. Amazon Aurora is up to five times faster than standard MySQL databases and three times faster than standard PostgreSQL databases.



It provides the security, availability, and reliability of commercial databases at 1/10th the cost. Amazon Aurora is fully managed by Amazon RDS, which automates time-consuming administration tasks like hardware provisioning, database setup, patching, and backups.

Based on the given scenario, there is a significant slowdown after testing the read performance from the secondary AWS Region. Since the existing setup is an Amazon RDS for MySQL, you should migrate the database to Amazon Aurora and create a cross-region read replica.



The read replication latency of less than 1 second is only possible if you would use Amazon Aurora replicas. Aurora replicas are independent endpoints in an Aurora DB cluster, best used for scaling read operations and increasing availability. You can create up to 15 replicas within an AWS Region.

Hence, the correct answer is: **Migrate the existing database to Amazon Aurora and create a cross-region read replica.**

The option that says: **Upgrade the MySQL database engine** is incorrect because upgrading the database engine wouldn't improve the read replication latency to milliseconds. To achieve the read replication latency of less than 1-second requirement, you need to use Amazon Aurora replicas.

The option that says: **Use Amazon ElastiCache to improve database performance** is incorrect. Amazon ElastiCache won't be able to improve the database performance because it is experiencing highly dynamic reads. This option would be helpful if the database frequently receives the same queries.

The option that says: **Create an Amazon RDS for MySQL read replica in the secondary AWS Region** is incorrect because MySQL replicas won't provide you a read replication latency of less than 1 second. RDS Read Replicas can only provide asynchronous replication in seconds and not in milliseconds. You have to use Amazon Aurora replicas in this scenario.

References:

https://aws.amazon.com/rds/aurora/faqs/

https://docs.aws.amazon.com/AmazonRDS/latest/AuroraUserGuide/AuroraMySQL.Replication.CrossRegion.html

Amazon Aurora Overview:

https://youtu.be/iwS1h7rLNBQ

Check out this Amazon Aurora Cheat Sheet:

https://tutorialsdojo.com/amazon-aurora/

Question 25: **Incorrect**
An application is using a Lambda function to process complex financial data that run for 15 minutes on average. Most invocations were successfully processed. However, you noticed that there are a few terminated invocations throughout the day, which caused data discrepancy in the application.

Which of the following is the most likely cause of this issue?

- **The Lambda function contains a recursive code and has been running for over 15 minutes.**
- **The failed Lambda Invocations contain a `ServiceException` error which means that the AWS Lambda service encountered an internal error.**

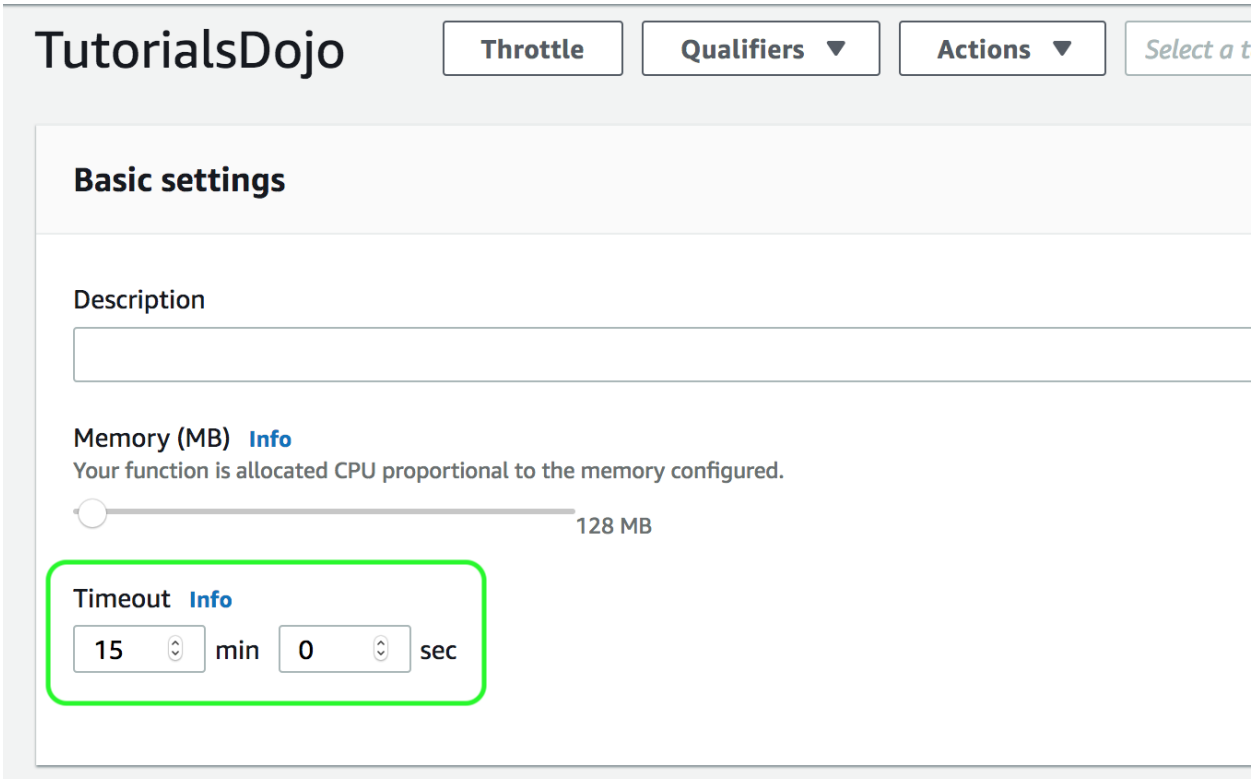
  (Incorrect)
- **The failed Lambda functions have been running for over 15 minutes and reached the maximum execution time.**

  (Correct)
- **The concurrent execution limit has been reached.**

Explanation
A **Lambda function** consists of code and any associated dependencies. In addition, a Lambda function also has configuration information associated with it. Initially, you specify the configuration information when you create a Lambda function. Lambda provides an API for you to update some of the configuration data.

You pay for the AWS resources that are used to run your Lambda function. To prevent your Lambda function from running indefinitely, you specify a **timeout**. When the specified timeout is reached, AWS Lambda terminates execution of your Lambda function. It is recommended that you set this value based on your expected execution time. The default timeout is 3 seconds, and the maximum execution duration per request in AWS Lambda is 900 seconds, which is equivalent to 15 minutes.

Hence, the correct answer is the option that says: **The failed Lambda functions have been running for over 15 minutes and reached the maximum execution time**.



Take note that you can invoke a Lambda function synchronously either by calling the `Invoke` operation or by using an AWS SDK in your preferred runtime. If you anticipate a long-running Lambda function, your client may time out before function execution completes. To avoid this, update the client timeout or your SDK configuration.

The option that says: **The concurrent execution limit has been reached** is incorrect because, by default, the AWS Lambda limits the total concurrent executions across all

functions within a given region to 1000. By setting a concurrency limit on a function, Lambda guarantees that allocation will be applied specifically to that function, regardless of the amount of traffic processing the remaining functions. If that limit is exceeded, the function will be throttled but not terminated, which is in contrast with what is happening in the scenario.

The option that says: **The Lambda function contains a recursive code and has been running for over 15 minutes** is incorrect because having a recursive code in your Lambda function does not directly result to an abrupt termination of the function execution. This is a scenario wherein the function automatically calls itself until some arbitrary criteria is met. This could lead to an unintended volume of function invocations and escalated costs, but not an abrupt termination because Lambda will throttle all invocations to the function.

The option that says: **The failed Lambda Invocations contain a `ServiceException` error which means that the AWS Lambda service encountered an internal error** is incorrect. Although this is a valid root cause, it is unlikely to have several **ServiceException** errors throughout the day unless there is an outage or disruption in AWS. Since the scenario says that the Lambda function runs for about 10 to 15 minutes, the maximum execution duration is the most likely cause of the issue and not the AWS Lambda service encountered an internal error.

References:

https://docs.aws.amazon.com/lambda/latest/dg/limits.html

https://docs.aws.amazon.com/lambda/latest/dg/resource-model.html

AWS Lambda Overview - Serverless Computing in AWS:

https://youtu.be/bPVX1zHwAnY

Check out this AWS Lambda Cheat Sheet:

https://tutorialsdojo.com/aws-lambda/

Question 26: **Correct**
A Solutions Architect is trying to enable Cross-Region Replication to an S3 bucket but this option is disabled. Which of the following options is a valid reason for this?

- 

This is a premium feature which is only for AWS Enterprise accounts.

- **The Cross-Region Replication feature is only available for Amazon S3 - Infrequent Access.**

- **The Cross-Region Replication feature is only available for Amazon S3 - One Zone-IA**

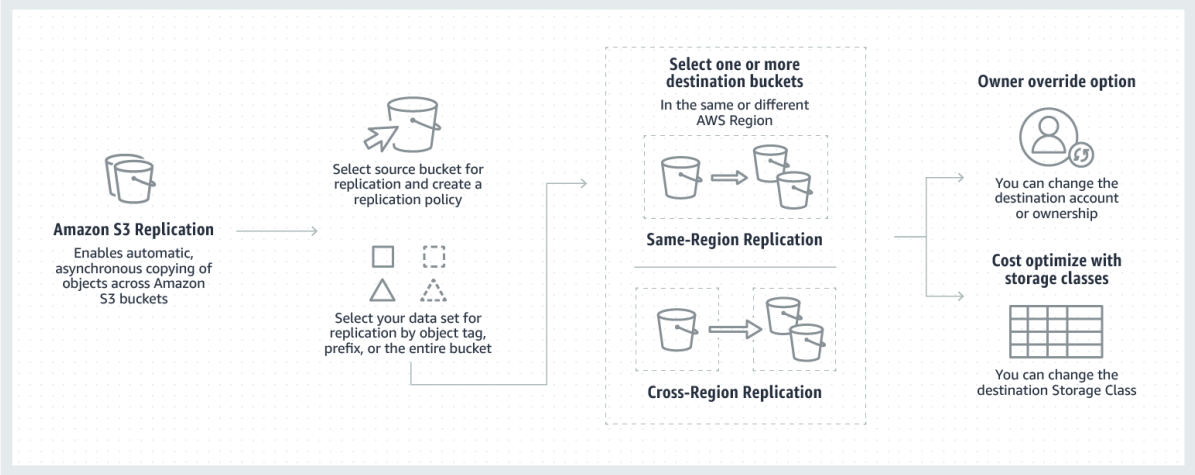
- **In order to use the Cross-Region Replication feature in S3, you need to first enable versioning on the bucket.**

  (Correct)

Explanation

To enable the cross-region replication feature in S3, the following items should be met:

1. The source and destination buckets must have versioning enabled.
2. The source and destination buckets must be in different AWS Regions.
3. Amazon S3 must have permission to replicate objects from that source bucket to the destination bucket on your behalf.



The options that say: **The Cross-Region Replication feature is only available for Amazon S3 - One Zone-IA and The Cross-Region Replication feature is only available for Amazon S3 - Infrequent Access** are incorrect as this feature is available to all types of S3 classes.

The option that says: **This is a premium feature which is only for AWS Enterprise accounts** is incorrect as this CRR feature is available to all Support Plans.

References:

https://docs.aws.amazon.com/AmazonS3/latest/dev/crr.html

https://aws.amazon.com/blogs/aws/new-cross-region-replication-for-amazon-s3/

Check out this Amazon S3 Cheat Sheet:

https://tutorialsdojo.com/amazon-s3/

Question 27: **Correct**
A company plans to host a movie streaming app in AWS. The chief information officer (CIO) wants to ensure that the application is highly available and scalable. The application is deployed to an Auto Scaling group of EC2 instances on multiple AZs. A load balancer must be configured to distribute incoming requests evenly to all EC2 instances across multiple Availability Zones.

Which of the following features should the Solutions Architect use to satisfy these criteria?

- **Amazon VPC IP Address Manager (IPAM)**
- **Cross-zone load balancing**

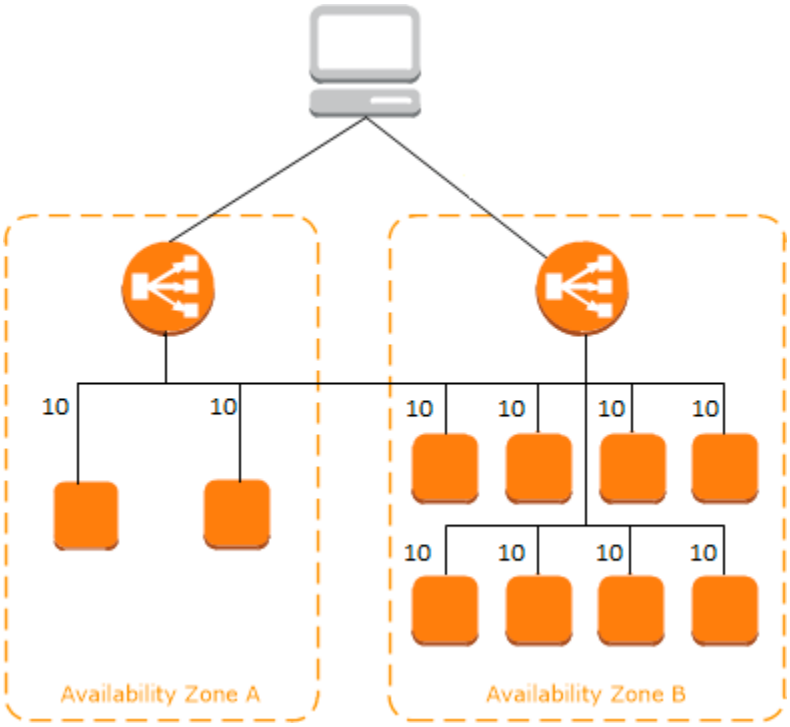
  (Correct)
- **AWS Direct Connect SiteLink**
- **Path-based Routing**

Explanation

The nodes for your load balancer distribute requests from clients to registered targets. When cross-zone load balancing is enabled, each load balancer node distributes traffic across the registered targets in all enabled Availability Zones. When cross-zone load balancing is disabled, each load balancer node distributes traffic only across the registered targets in its Availability Zone.

The following diagrams demonstrate the effect of cross-zone load balancing. There are two enabled Availability Zones, with two targets in Availability Zone A and eight targets in Availability Zone B. Clients send requests, and Amazon Route 53 responds to each request with the IP address of one of the load balancer nodes. This distributes traffic such that each load balancer node receives 50% of the traffic from the clients. Each load balancer node distributes its share of the traffic across the registered targets in its scope.

If cross-zone load balancing is enabled, each of the 10 targets receives 10% of the traffic. This is because each load balancer node can route 50% of the client traffic to all 10 targets.

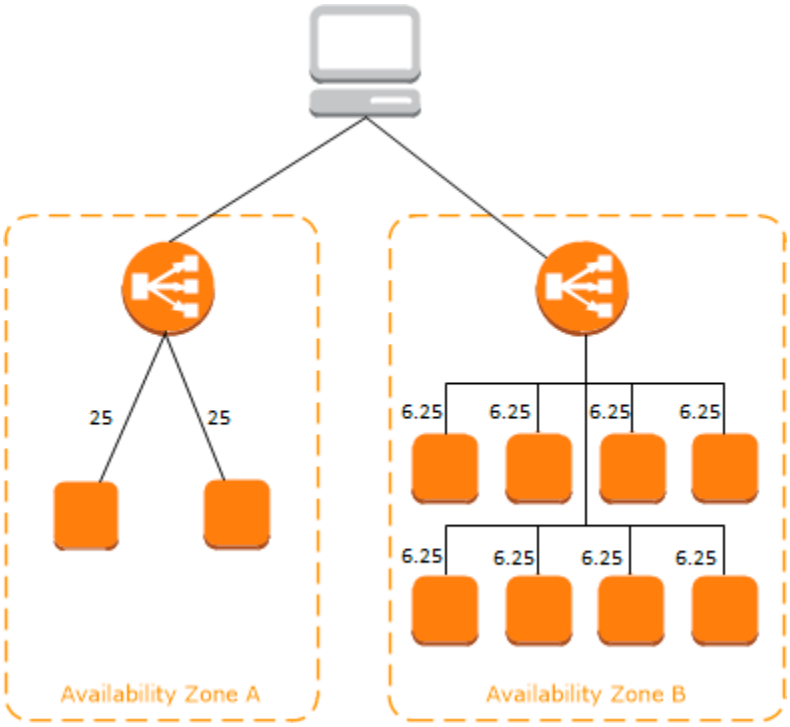


If cross-zone load balancing is disabled:

Each of the two targets in Availability Zone A receives 25% of the traffic.

Each of the eight targets in Availability Zone B receives 6.25% of the traffic.

This is because each load balancer node can route 50% of the client traffic only to targets in its Availability Zone.



With Application Load Balancers, cross-zone load balancing is always enabled.

With Network Load Balancers and Gateway Load Balancers, cross-zone load balancing is disabled by default. After you create the load balancer, you can enable or disable cross-zone load balancing at any time.

When you create a Classic Load Balancer, the default for cross-zone load balancing depends on how you create the load balancer. With the API or CLI, cross-zone load balancing is disabled by default. With the AWS Management Console, the option to enable cross-zone load balancing is selected by default. After you create a Classic Load Balancer, you can enable or disable cross-zone load balancing at any time

Hence, the right answer is to enable **cross-zone load balancing.**

Amazon VPC IP Address Manager (IPAM) is incorrect because this is merely a feature in Amazon VPC that provides network administrators with an automated IP management workflow. It does not enable your load balancers to distribute incoming requests evenly to all EC2 instances across multiple Availability Zones.

Path-based Routing is incorrect because this feature is based on the paths that are in the URL of the request. It automatically routes traffic to a particular target group based

on the request URL. This feature will not set each of the load balancer nodes to distribute traffic across the registered targets in all enabled Availability Zones.

AWS Direct Connect SiteLink is incorrect because this is a feature of AWS Direct Connect connection and not of Amazon Elastic Load Balancing. The AWS Direct Connect SiteLink feature simply lets you create connections between your on-premises networks through the AWS global network backbone.

References:

https://docs.aws.amazon.com/elasticloadbalancing/latest/userguide/how-elastic-load-balancing-works.html

https://aws.amazon.com/elasticloadbalancing/features

https://aws.amazon.com/blogs/aws/network-address-management-and-auditing-at-scale-with-amazon-vpc-ip-address-manager/

AWS Elastic Load Balancing Overview:

https://youtu.be/UBl5dw59DO8

Check out this AWS Elastic Load Balancing (ELB) Cheat Sheet:

https://tutorialsdojo.com/aws-elastic-load-balancing-elb/

Question 28: **Correct**
A tech company is having an issue whenever they try to connect to the newly created EC2 instance using a Remote Desktop connection from a computer. Upon checking, the Solutions Architect has verified that the instance has a public IP and the Internet gateway and route tables are in place.

What else should he do to resolve this issue?

- **Adjust the security group to allow inbound traffic on port 3389 from the company's IP address.**

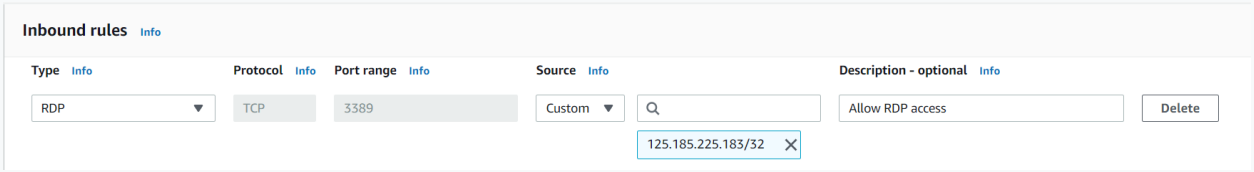
(Correct)

- **Adjust the security group to allow inbound traffic on port 22 from the company's IP address.**

- **You should create a new instance since there might be some issue with the instance**

- **You should restart the EC2 instance since there might be some issue with the instance**

Explanation
Since you are using a Remote Desktop connection to access your EC2 instance, you have to ensure that the Remote Desktop Protocol is allowed in the security group. By default, the server listens on TCP port 3389 and UDP port 3389.



Hence, the correct answer is: **Adjust the security group to allow inbound traffic on port 3389 from the company's IP address.**

The option that says: **Adjust the security group to allow inbound traffic on port 22 from the company's IP address** is incorrect as port 22 is used for SSH connections and not for RDP.

The options that say: **You should restart the EC2 instance since there might be some issue with the instance** and **You should create a new instance since there might be some issue with the instance** are incorrect as the EC2 instance is newly created and hence, unlikely to cause the issue. You have to check the security group first if it allows the Remote Desktop Protocol (3389) before investigating if there is indeed an issue on the specific instance.

Reference:

https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/troubleshooting-windows-instances.html#rdp-issues

https://docs.aws.amazon.com/vpc/latest/userguide/VPC_SecurityGroups.html

Check out this Amazon EC2 Cheat Sheet:

https://tutorialsdojo.com/amazon-elastic-compute-cloud-amazon-ec2/

Question 29: **Incorrect**
A website hosted on Amazon ECS container instances loads slowly during peak traffic, affecting its availability. Currently, the container instances are run behind an Application Load Balancer, and CloudWatch alarms are configured to send notifications to the operations team if there is a problem in availability so they can scale out if needed. A solutions architect needs to create an automatic scaling solution when such problems occur.

Which solution could satisfy the requirement? (Select TWO.)

- ☑ **Create an AWS Auto Scaling policy that scales out the ECS cluster when the service's CPU utilization is too high.**

  (Correct)

- ☐ **Create an AWS Auto Scaling policy that scales out the ECS service when the service's memory utilization is too high.**

  (Correct)

- ☐ **Create an AWS Auto Scaling policy that scales out the ECS cluster when the ALB target group's CPU utilization is too high.**

- ☐ **Create an AWS Auto Scaling policy that scales out an ECS service when the ALB endpoint becomes unreachable.**

- ☑

  Create an AWS Auto Scaling policy that scales out the ECS service when the ALB hits a high CPU utilization.

  (Incorrect)

Explanation
AWS Auto Scaling monitors your applications and automatically adjusts capacity to maintain steady, predictable performance at the lowest possible cost. Using AWS Auto Scaling, it's easy to set up application scaling for multiple resources across multiple services in minutes. The service provides a simple, powerful user interface that lets you build scaling plans for resources, including Amazon EC2 instances and Spot Fleets, Amazon ECS tasks, Amazon DynamoDB tables and indexes, and Amazon Aurora Replicas.

In this scenario, you can set up a scaling policy that triggers a scale-out activity to an ECS service or ECS container instance based on the metric that you prefer.

The following metrics are available for instances:

-CPU Utilization

-Disk Reads

-Disk Read Operations

-Disk Writes

-Disk Write Operations

-Network In

-Network Out

-Status Check Failed (Any)

-Status Check Failed (Instance)

-Status Check Failed (System)

The following metrics are available for ECS Service:

-ECSServiceAverageCPUUtilization—Average CPU utilization of the service.

-ECSServiceAverageMemoryUtilization—Average memory utilization of the service.

-ALBRequestCountPerTarget—Number of requests completed per target in an Application Load Balancer target group.

Hence, the correct answers are:

- Create an AWS Auto scaling policy that scales out the ECS service when the service's memory utilization is too high.

- Create an AWS Auto scaling policy that scales out the ECS cluster when the service's CPU utilization is too high.

The option that says: **Create an AWS Auto scaling policy that scales out an ECS service when the ALB endpoint becomes unreachable** is incorrect. This would be a different problem that needs to be addressed differently if this is the case. An unreachable ALB endpoint could mean other things like a misconfigured security group or network access control lists.

The option that says: **Create an AWS Auto scaling policy that scales out the ECS service when the ALB hits a high CPU utilization** is incorrect. You cannot track nor view the CPU utilization of an ALB.

The option that says: **Create an AWS Auto scaling policy that scales out the ECS cluster when the ALB target group's CPU utilization is too high** is incorrect. AWS Auto Scaling does not support this metric for ALB.

References:

https://docs.aws.amazon.com/AmazonECS/latest/developerguide/service-configure-auto-scaling.html

https://docs.aws.amazon.com/autoscaling/ec2/userguide/as-instance-monitoring.html

Check out this AWS Auto Scaling Cheat Sheet:

https://tutorialsdojo.com/aws-auto-scaling/

Question 30: **Incorrect**
A company is running an on-premises application backed by a 1TB MySQL 8.0 database. A couple of times each month, the production data is fully copied to a staging

database at the request of the analytics team. The team can't work on the staging database until the copy is finished, which takes hours.

Throughout this period, the application experiences intermittent downtimes as well. To expedite the process for the analytics team, a solutions architect must redesign the application's architecture in AWS. The application must also be highly resilient to disruptions.

Which combination of actions best satisfies the given set of requirements while being the most cost-effective? (Select TWO)

- [x] **Use an Amazon Aurora database with Multi-AZ Replicas.**

  (Correct)

- [ ] **Clone the production database in the staging environment using Aurora cloning.**

  (Correct)

- [x] **Replicate the production database to a staging database using the `mysqldump` client utility**

  (Incorrect)

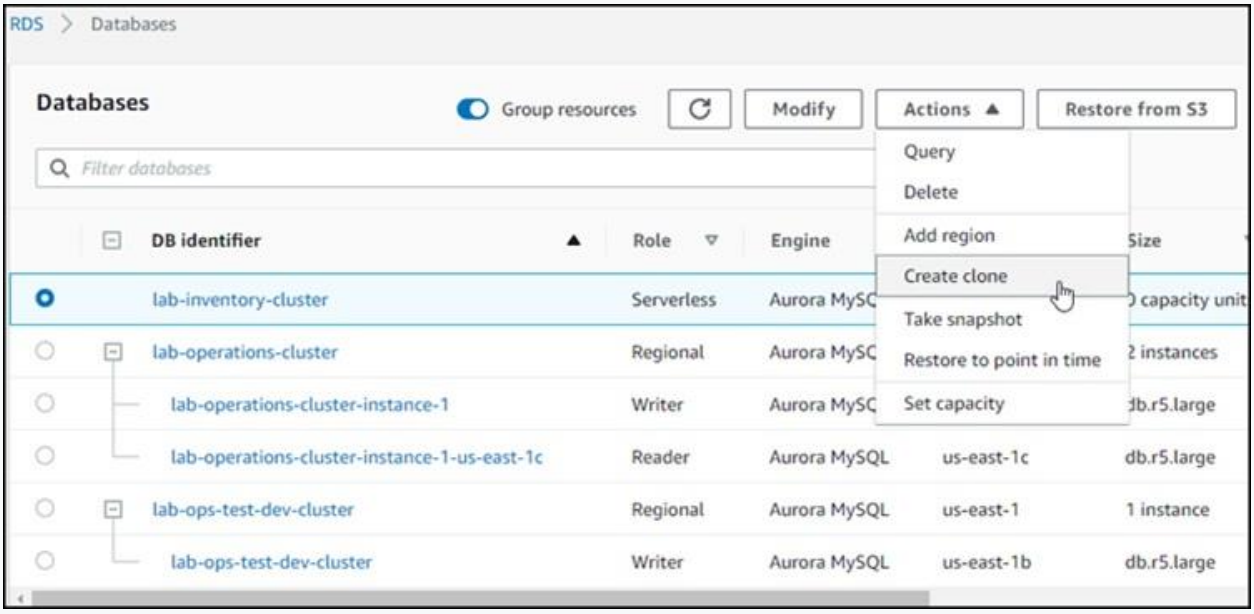
- [ ] **Take a manual snapshot and restore it to a database in the staging environment.**

- [ ] **Use an Amazon RDS database in a Multi-AZ Deployments configuration**

Explanation

The resiliency of an application pertains to its ability to recover from infrastructure or service disruptions. Both Amazon Aurora and Amazon RDS can give you a highly resilient infrastructure by deploying replicas in multiple availability zones. Both database services can perform an automatic failover to a standby instance in the event of failure. However, only Amazon Aurora has the ability to replicate a database in a fast

and efficient manner without impacting performance, thanks to its underlying storage system. By using Aurora cloning, you can create a new cluster that uses the same Aurora cluster volume and has the same data as the original. The process is designed to be fast and cost-effective. The new cluster with its associated data volume is called a clone.



Creating a clone is faster and more space-efficient than physically copying the data using other techniques, such as restoring from a snapshot like you would in Amazon RDS or using the native `mysqldump` utility.

Therefore, the correct answers are:

- Use an Amazon Aurora database with Multi-AZ Replicas.

- Clone the production database in the staging environment using Aurora cloning.

The option that says: **Use an Amazon RDS database in a Multi-AZ Deployments configuration** is incorrect. While Amazon RDS is a valid option, Amazon Aurora best fits what's described in the scenario due to its cloning feature.

The option that says: **Take a manual snapshot and restore it to a database in the staging environment** is incorrect. Using Aurora Cloning is a faster and more cost-effective method than restoring from a snapshot.

The option that says: **Replicate the production database to a staging database using the `mysqldump` client utility** is incorrect as this is a slow process and will impact the performance of the source database.

References:

https://docs.aws.amazon.com/AmazonRDS/latest/AuroraUserGuide/Aurora.Managing.Clone.html

https://aws.amazon.com/blogs/aws/amazon-aurora-fast-database-cloning/

Check out this Amazon Aurora Cheat Sheet:

https://tutorialsdojo.com/amazon-aurora/

Question 31: **Incorrect**
A company has a fleet of running Spot EC2 instances behind an Application Load Balancer. The incoming traffic comes from various users across multiple AWS regions and you would like to have the user's session shared among the fleet of instances. You are required to set up a distributed session management layer that will provide a scalable and shared data storage for the user sessions.

Which of the following would be the best choice to meet the requirement while still providing sub-millisecond latency for the users?

- **ElastiCache in-memory caching**

  (Correct)

- **ELB sticky sessions**

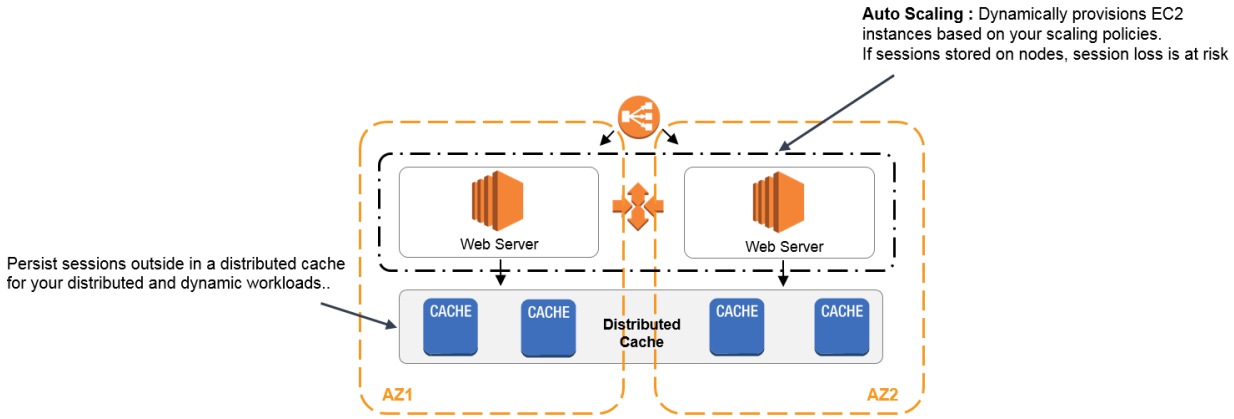
  (Incorrect)

- **Multi-AZ RDS**

- **Multi-master DynamoDB**

Explanation

For sub-millisecond latency caching, **ElastiCache** is the best choice. In order to address scalability and to provide a shared data storage for sessions that can be accessed from any individual web server, you can abstract the HTTP sessions from the web servers themselves. A common solution for this is to leverage an In-Memory Key/Value store such as Redis and Memcached.



ELB sticky sessions is incorrect because the scenario does not require you to route a user to the particular web server that is managing that individual user's session. Since the session state is shared among the instances, the use of the ELB sticky sessions feature is not recommended in this scenario.

Multi-master DynamoDB and Multi-AZ RDS are incorrect. Although you can use DynamoDB and RDS for storing session state, these two are not the best choices in terms of cost-effectiveness and performance when compared to ElastiCache. There is a significant difference in terms of latency if you used DynamoDB and RDS when you store the session data.

References:

https://aws.amazon.com/caching/session-management/

https://d0.awsstatic.com/whitepapers/performance-at-scale-with-amazon-elasticache.pdf

Check out this Amazon Elasticache Cheat Sheet:

https://tutorialsdojo.com/amazon-elasticache/

Redis (cluster mode enabled vs. disabled) vs. Memcached:

https://tutorialsdojo.com/redis-cluster-mode-enabled-vs-disabled-vs-memcached/

Question 32: **Incorrect**
A company has several websites and hosts its infrastructure on the AWS Cloud. The mission-critical web applications are hosted on fleets of Amazon EC2 instances behind Application Load Balancers. The company uses AWS Certificate Manager (ACM) provided certificate on the ALBs to enable HTTPS access on its websites. The security team wants to get notified 30 days before the expiration of the SSL certificates.

Which of the following can the Solutions Architect implement to meet this request? (Select TWO.)

- ☐ **Modify all certificates to use the AWS Certificate Manager Private Certificate Authority. Create an Amazon EventBridge (Amazon CloudWatch Events) rule that will check for ACM events that shows certificates expiring within 30 days. Set the target to invoke an AWS Lambda function to send a message to an Amazon SNS topic.**

- ☑ **Create an Amazon EventBridge (Amazon CloudWatch Events) rule that will check AWS Health or ACM expiration events related to ACM certificates. Send an alert notification to an Amazon Simple Notification Service (Amazon SNS) topic when a certificate is going to expire in 30 days.**

  (Correct)

- ☐ **Utilize AWS Trusted Advisor to check for the ACM certificates that will expire in 30 days. Using this metric, create an Amazon CloudWatch alarm that will send an alert to an AWS Systems Manager OpsItem.**

- ☑ **Use AWS Config to manually create a rule that checks for certificate expiry on ACM. Create an Amazon EventBridge (Amazon CloudWatch Events) rule to**

send an alert to an Amazon Simple Notification Service (Amazon SNS) topic when AWS Config flags a resource.

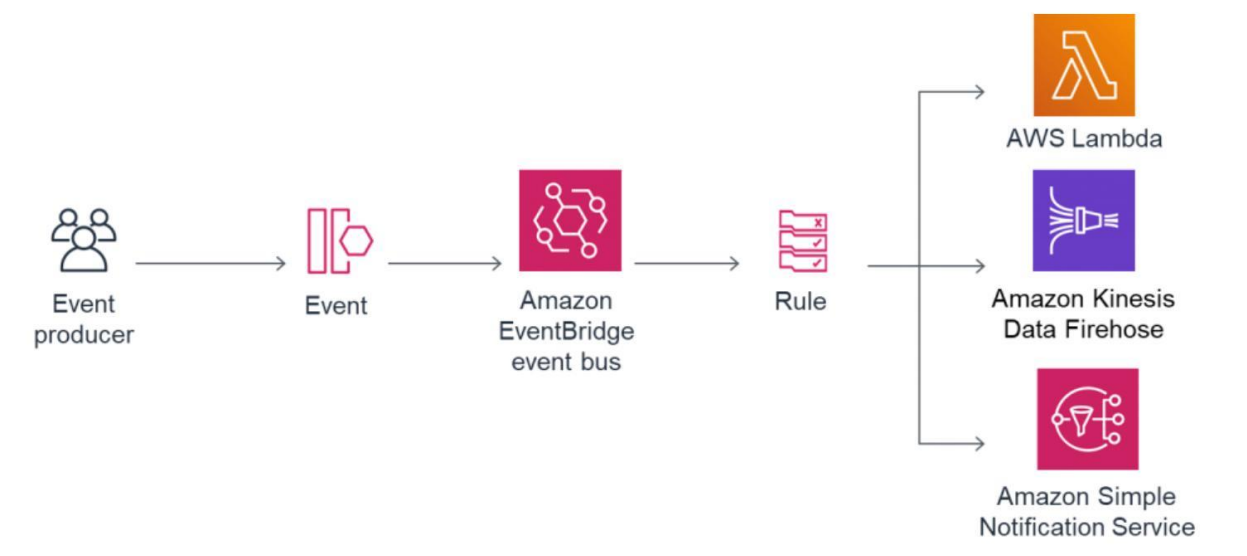
(Incorrect)

- ☐

Create an Amazon EventBridge (Amazon CloudWatch Events) rule and schedule it to run every day to identify the expiring ACM certificates. Configure to rule to check the `DaysToExpiry` metric of all ACM certificates in Amazon CloudWatch. Send an alert notification to an Amazon Simple Notification Service (Amazon SNS) topic when a certificate is going to expire in 30 days.

(Correct)

Explanation
You can use **Amazon CloudWatch Events** to automate your AWS services and respond automatically to system events such as application availability issues or resource changes. CloudWatch Events are turned into actions using **Amazon EventBridge**.



AWS Health events are generated for ACM certificates that are eligible for renewal. Health events are generated in two scenarios:

-On successful renewal of a public or private certificate.

-When a customer must take action for a renewal to occur. This may mean clicking a link in an email message (for email-validated certificates), or resolving an error. One of

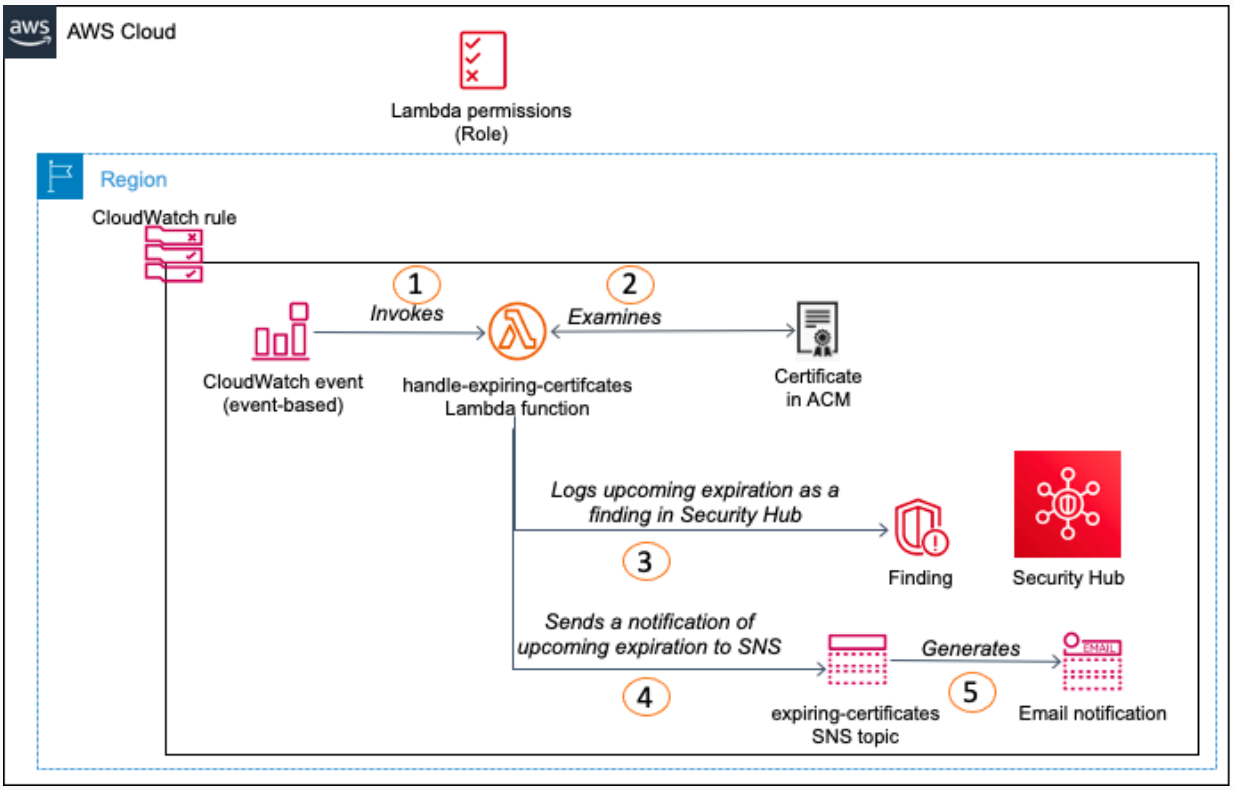
the following event codes is included with each event. The codes are exposed as variables that you can use for filtering.

-AWS_ACM_RENEWAL_STATE_CHANGE (the certificate has been renewed, has expired, or is due to expire)

-CAA_CHECK_FAILURE (CAA check failed)

-AWS_ACM_RENEWAL_FAILURE (for certificates signed by a private CA)

ACM sends daily expiration events for all active certificates (public, private and imported) starting 45 days prior to expiration. You can use expiration events to set up automation to reimport certificates into ACM. You can create CloudWatch rules based on these events and use the CloudWatch console to configure actions that take place when the events are detected.



For this scenario, you can have two possible options to implement. The first option uses an AWS ACM built-in Certificate Expiration event, which is raised through Amazon EventBridge, to invoke a Lambda function. In this option, the function is configured to publish the result as a finding in Security Hub, and also as an SNS topic used for email subscriptions. As a result, an administrator can be notified of a specific expiring certificate, or an IT service management (ITSM) system can automatically open a case or incident through email or SNS.

The second option uses the recently launched **DaysToExpiry** metric to schedule a batch search of expiring certificates and to log all the findings. The metric also provides a single SNS notification for all expiring certificates.

Therefore, the correct answers are:

- Create an Amazon EventBridge (Amazon CloudWatch Events) rule that will check AWS Health or ACM expiration events related to ACM certificates. Send an alert notification to an Amazon Simple Notification Service (Amazon SNS) topic when a certificate is going to expire in 30 days

- Create an Amazon EventBridge (Amazon CloudWatch Events) rule and schedule it to run every day to identify the expiring ACM certificates. Configure to rule to check the `DaysToExpiry` metric of all ACM certificates in Amazon CloudWatch. Send an alert notification to an Amazon Simple Notification Service (Amazon SNS) topic when a certificate is going to expire in 30 days.

The option that says: **Use AWS Config to manually create a rule that checks for certificate expiry on ACM. Create an Amazon EventBridge (Amazon CloudWatch Events) rule to send an alert to an Amazon Simple Notification Service (Amazon SNS) topic when AWS Config flags a resource** is incorrect. AWS Certificate Manager automatically generates AWS Health events. Manually creating a custom AWS Config rule to check for SSL expiry is unnecessary. In addition, AWS Config already provides a built-in acm-certificate-expiration-check managed rule that you can use.

The option that says: **Modify all certificates to use the AWS Certificate Manager Private Certificate Authority. Create an Amazon EventBridge (Amazon CloudWatch Events) rule that will check for ACM events that shows certificates expiring within 30 days. Set the target to invoke an AWS Lambda function to send a message to an Amazon SNS topic** is incorrect. There's absolutely no need to modify all the existing SSL certificates to use the AWS Certificate Manager Private Certificate Authority. This feature simply provides you a highly-available private CA service without the upfront investment and ongoing maintenance costs of operating your own private CA.

The option that says: **Utilize AWS Trusted Advisor to check for the ACM certificates that will expire in 30 days. Using this metric, create an Amazon CloudWatch alarm that will send an alert to an AWS Systems Manager OpsItem** is incorrect. A Systems Manager OpsItem is simply a feature of the AWS Systems Manager OpsCenter service. This is just is an operational work item that provides data to the OpsItem widgets of Amazon Systems Manager Explorer. In addition, the AWS Trusted Advisor checks do not include expiring SSL certificates in AWS Certificate Manager

References:

https://docs.aws.amazon.com/acm/latest/userguide/supported-events.html

https://docs.aws.amazon.com/acm/latest/userguide/event-sns-response.html

https://aws.amazon.com/blogs/security/how-to-monitor-expirations-of-imported-certificates-in-aws-certificate-manager-acm/

Check out these Amazon CloudWatch and AWS Certificate Manager Cheat Sheets:

https://tutorialsdojo.com/amazon-cloudwatch/

https://tutorialsdojo.com/aws-certificate-manager/

Question 33: **Correct**

A multinational corporate and investment bank is regularly processing steady workloads of accruals, loan interests, and other critical financial calculations every night from 10 PM to 3 AM on their on-premises data center for their corporate clients. Once the process is done, the results are then uploaded to the Oracle General Ledger which means that the processing should not be delayed or interrupted. The CTO has decided to move its IT infrastructure to AWS to save costs. The company needs to reserve compute capacity in a specific Availability Zone to properly run their workloads.

As the Senior Solutions Architect, how can you implement a cost-effective architecture in AWS for their financial system?

- **Use Dedicated Hosts which provide a physical host that is fully dedicated to running your instances, and bring your existing per-socket, per-core, or per-VM software licenses to reduce costs.**

- **Use Regional Reserved Instances to reserve capacity on a specific Availability Zone and lower down the operating cost through its billing discounts.**

- **Use On-Demand Capacity Reservations, which provide compute capacity that is always available on the specified recurring schedule.**

  (Correct)

- 

Use On-Demand EC2 instances which allows you to pay for the instances that you launch and use by the second. Reserve compute capacity in a specific Availability Zone to avoid any interruption.

Explanation

On-Demand Capacity Reservations enable you to reserve compute capacity for your Amazon EC2 instances in a specific Availability Zone for any duration. This gives you the ability to create and manage Capacity Reservations independently from the billing discounts offered by Savings Plans or Regional Reserved Instances.

By creating Capacity Reservations, you ensure that you always have access to EC2 capacity when you need it, for as long as you need it. You can create Capacity Reservations at any time, without entering into a one-year or three-year term commitment, and the capacity is available immediately. Billing starts as soon as the capacity is provisioned and the Capacity Reservation enters the active state. When you no longer need it, cancel the Capacity Reservation to stop incurring charges.

	Capacity Reservations	Zonal Reserved Instances	Regional Reserved Instances	Savings Plans
Term	No commitment required. Can be created and canceled as needed.	Requires a fixed one-year or three-year commitment		
Capacity benefit	Capacity reserved in a specific Availability Zone.		No capacity reserved.	
Billing discount	No billing discount. †	Provides a billing discount.		
Instance Limits	Your On-Demand Instance limits per Region apply.	Default is 20 per Availability Zone. You can request a limit increase.	Default is 20 per Region. You can request a limit increase.	No limit.

When you create a Capacity Reservation, you specify:

- The Availability Zone in which to reserve the capacity

- The number of instances for which to reserve capacity

- The instance attributes, including the instance type, tenancy, and platform/OS

Capacity Reservations can only be used by instances that match their attributes. By default, they are automatically used by running instances that match the attributes. If you don't have any running instances that match the attributes of the Capacity Reservation, it remains unused until you launch an instance with matching attributes.

In addition, you can use Savings Plans and Regional Reserved Instances with your Capacity Reservations to benefit from billing discounts. AWS automatically applies your discount when the attributes of a Capacity Reservation match the attributes of a Savings Plan or Regional Reserved Instance.

Hence, the correct answer is to **use On-Demand Capacity Reservations, which provide compute capacity that is always available on the specified recurring schedule.**

Using On-Demand EC2 instances which allows you to pay for the instances that you launch and use by the second. Reserve compute capacity in a specific Availability Zone to avoid any interruption is incorrect because although an On-Demand instance is stable and suitable for processing critical data, it costs more than any other option. Moreover, the critical financial calculations are only done every night from 10 PM to 3 AM only and not 24/7. This means that your compute capacity will not be utilized for a total of 19 hours every single day. On-Demand instances cannot reserve compute capacity at all. So this option is incorrect.

Using Regional Reserved Instances to reserve capacity on a specific Availability Zone and lower down the operating cost through its billing discounts is incorrect because this feature is available in Zonal Reserved Instances only and not on Regional Reserved Instances.

Using Dedicated Hosts which provide a physical host that is fully dedicated to running your instances, and bringing your existing per-socket, per-core, or per-VM software licenses to reduce costs is incorrect because the use of a fully dedicated physical host is not warranted in this scenario. Moreover, this will be underutilized since you only run the process for 5 hours (from 10 PM to 3 AM only), wasting 19 hours of compute capacity every single day.

References:

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-capacity-reservations.html

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/instance-purchasing-options.html

Check out this Amazon EC2 Cheat Sheet:

https://tutorialsdojo.com/amazon-elastic-compute-cloud-amazon-ec2/

Question 34: **Correct**
A multinational company has been building its new data analytics platform with high-performance computing workloads (HPC) which requires a scalable, POSIX-compliant storage service. The data need to be stored redundantly across multiple AZs and allows

concurrent connections from thousands of EC2 instances hosted on multiple Availability Zones.

Which of the following AWS storage service is the most suitable one to use in this scenario?

- **Amazon S3**
- **Amazon EBS Volumes**
- **Amazon Elastic File System**

  (Correct)
- **Amazon ElastiCache**

Explanation
In this question, you should take note of this phrase, "allows concurrent connections from multiple EC2 instances". There are various AWS storage options that you can choose but whenever these criteria show up, always consider using EFS instead of using EBS Volumes which is mainly used as a "block" storage and can only have one connection to one EC2 instance at a time.

Amazon EFS is a fully-managed service that makes it easy to set up and scale file storage in the Amazon Cloud. With a few clicks in the AWS Management Console, you can create file systems that are accessible to Amazon EC2 instances via a file system interface (using standard operating system file I/O APIs) and supports full file system access semantics (such as strong consistency and file locking).

Amazon EFS file systems can automatically scale from gigabytes to petabytes of data without needing to provision storage. Tens, hundreds, or even thousands of Amazon EC2 instances can access an Amazon EFS file system at the same time, and Amazon EFS provides consistent performance to each Amazon EC2 instance. Amazon EFS is designed to be highly durable and highly available.

References:

https://docs.aws.amazon.com/efs/latest/ug/performance.html

https://aws.amazon.com/efs/faq/

Check out this Amazon EFS Cheat Sheet:

https://tutorialsdojo.com/amazon-efs/

Here's a short video tutorial on Amazon EFS:

https://youtu.be/AvgAozsfCrY

Question 35: **Correct**
A company has several unencrypted EBS snapshots in their VPC. The Solutions Architect must ensure that all of the new EBS volumes restored from the unencrypted snapshots are automatically encrypted.

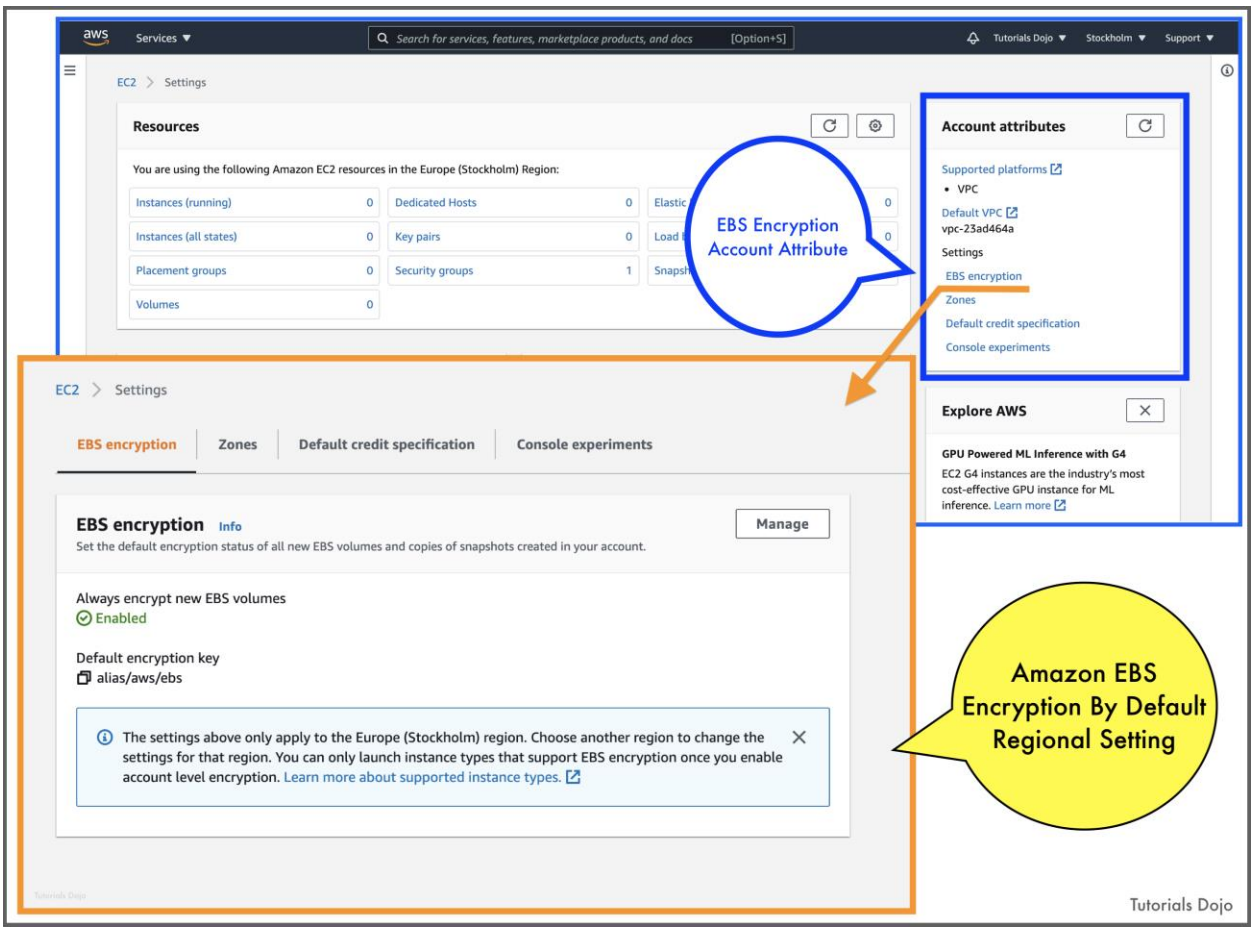
What should be done to accomplish this requirement?

- **Launch new EBS volumes and specify the symmetric customer master key (CMK) for encryption.**
- **Launch new EBS volumes and encrypt them using an asymmetric customer master key (CMK).**
- **Enable the EBS Encryption By Default feature for specific EBS volumes.**
- **Enable the EBS Encryption By Default feature for the AWS Region.**

(Correct)

Explanation

You can configure your AWS account to enforce the encryption of the new EBS volumes and snapshot copies that you create. For example, Amazon EBS encrypts the EBS volumes created when you launch an instance and the snapshots that you copy from an unencrypted snapshot.



Encryption by default has no effect on existing EBS volumes or snapshots. The following are important considerations in EBS encryption:

- Encryption by default is a Region-specific setting. If you enable it for a Region, you cannot disable it for individual volumes or snapshots in that Region.

- When you enable encryption by default, you can launch an instance only if the instance type supports EBS encryption.

- Amazon EBS does not support asymmetric CMKs.

When migrating servers using AWS Server Migration Service (SMS), do not turn on encryption by default. If encryption by default is already on and you are experiencing delta replication failures, turn off encryption by default. Instead, enable AMI encryption when you create the replication job.

You cannot change the CMK that is associated with an existing snapshot or encrypted volume. However, you can associate a different CMK during a snapshot copy operation so that the resulting copied snapshot is encrypted by the new CMK.

Although there is no direct way to encrypt an existing unencrypted volume or snapshot, you can encrypt them by creating either a volume or a snapshot. If you enabled encryption by default, Amazon EBS encrypts the resulting new volume or snapshot using your default key for EBS encryption. Even if you have not enabled encryption by default, you can enable encryption when you create an individual volume or snapshot. Whether you enable encryption by default or in individual creation operations, you can override the default key for EBS encryption and use symmetric customer-managed CMK.

Hence, the correct answer is: **Enable the EBS Encryption By Default feature for the AWS Region.**

The option that says: **Launch new EBS volumes and encrypt them using an asymmetric customer master key (CMK)** is incorrect because Amazon EBS does not support asymmetric CMKs. To encrypt an EBS snapshot, you need to use symmetric CMK.

The option that says: **Launch new EBS volumes and specify the symmetric customer master key (CMK) for encryption** is incorrect. Although this solution will enable data encryption, this process is manual and can potentially cause some unencrypted EBS volumes to be launched. A better solution is to enable the EBS Encryption By Default feature. It is stated in the scenario that all of the new EBS volumes restored from the unencrypted snapshots must be automatically encrypted.

The option that says: **Enable the EBS Encryption By Default feature for specific EBS volumes** is incorrect because the Encryption By Default feature is a Region-specific setting and thus, you can't enable it to selected EBS volumes only.

References:

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EBSEncryption.html#encryption-by-default

https://docs.aws.amazon.com/kms/latest/developerguide/services-ebs.html

Check out this Amazon EBS Cheat Sheet:

https://tutorialsdojo.com/amazon-ebs/

Comparison of Amazon S3 vs Amazon EBS vs Amazon EFS:

https://tutorialsdojo.com/amazon-s3-vs-ebs-vs-efs/

Question 36: **Incorrect**
A software development company has hundreds of Amazon EC2 instances with multiple Application Load Balancers (ALBs) across multiple AWS Regions. The public applications hosted in their EC2 instances are accessed on their on-premises network. The company needs to reduce the number of IP addresses that it needs to regularly whitelist on the corporate firewall device.

Which of the following approach can be used to fulfill this requirement?

- **Launch a Network Load Balancer with an associated Elastic IP address. Set the ALBs in multiple Regions as targets.**

  (Incorrect)

- **Use AWS Global Accelerator and create multiple endpoints for all the available AWS Regions. Associate all the private IP addresses of the EC2 instances to the corresponding endpoints.**

- **Use AWS Global Accelerator and create an endpoint group for each AWS Region. Associate the Application Load Balancer from each region to the corresponding endpoint group.**

  (Correct)

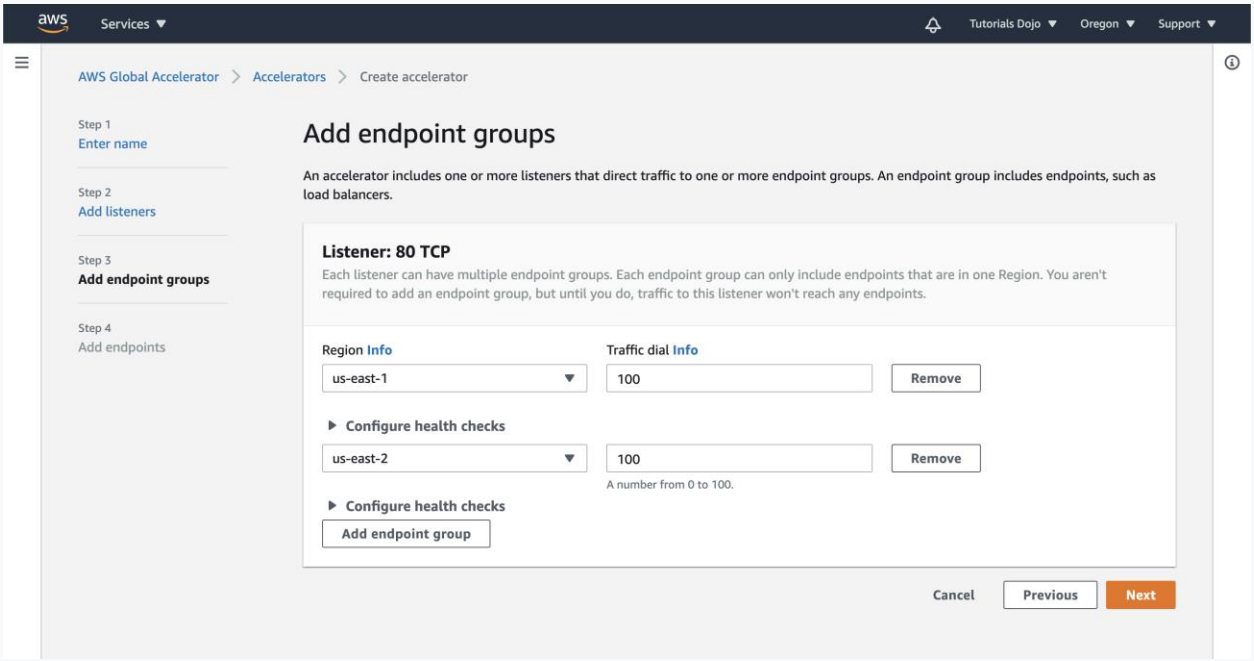
-

Create a new Lambda function that tracks the changes in the IP addresses of all ALBs across multiple AWS Regions. Schedule the function to run and update the corporate firewall every hour using Amazon CloudWatch Events.

Explanation

AWS Global Accelerator is a service that improves the availability and performance of your applications with local or global users. It provides static IP addresses that act as a fixed entry point to your application endpoints in a single or multiple AWS Regions, such as your Application Load Balancers, Network Load Balancers, or Amazon EC2 instances.

When the application usage grows, the number of IP addresses and endpoints that you need to manage also increase. AWS Global Accelerator allows you to scale your network up or down. AWS Global Accelerator lets you associate regional resources, such as load balancers and EC2 instances, to two static IP addresses. You only whitelist these addresses once in your client applications, firewalls, and DNS records.



With AWS Global Accelerator, you can add or remove endpoints in the AWS Regions, run blue/green deployment, and A/B test without needing to update the IP addresses in your client applications. This is particularly useful for IoT, retail, media, automotive, and healthcare use cases in which client applications cannot be updated frequently.

If you have multiple resources in multiple regions, you can use AWS Global Accelerator to reduce the number of IP addresses. By creating an endpoint group, you can add all of your EC2 instances from a single region in that group. You can add additional endpoint groups for instances in other regions. After it, you can then associate the appropriate ALB endpoints to each of your endpoint groups. The created accelerator would have two static IP addresses that you can use to create a security rule in your firewall device.

Instead of regularly adding the Amazon EC2 IP addresses in your firewall, you can use the static IP addresses of AWS Global Accelerator to automate the process and eliminate this repetitive task.

Hence, the correct answer is: **Use AWS Global Accelerator and create an endpoint group for each AWS Region. Associate the Application Load Balancer from each region to the corresponding endpoint group.**

The option that says: **Use AWS Global Accelerator and create multiple endpoints for all the available AWS Regions. Associate all the private IP addresses of the EC2 instances to the corresponding endpoints** is incorrect. It is better to create one endpoint group instead of multiple endpoints. Moreover, you have to associate the ALBs in AWS Global Accelerator and not the underlying EC2 instances.

The option that says: **Create a new Lambda function that tracks the changes in the IP addresses of all ALBs across multiple AWS Regions. Schedule the function to run and update the corporate firewall every hour using Amazon CloudWatch Events** is incorrect because this approach entails a lot of administrative overhead and takes a significant amount of time to implement. Using a custom Lambda function is actually not necessary since you can simply use AWS Global Accelerator to achieve this requirement.

The option that says: **Launch a Network Load Balancer with an associated Elastic IP address. Set the ALBs in multiple Regions as targets** is incorrect. Although you can allocate an Elastic IP address to your ELB, it is not suitable to route traffic to your ALBs across multiple Regions. You have to use AWS Global Accelerator instead.

References:

https://docs.aws.amazon.com/global-accelerator/latest/dg/about-endpoint-groups.html

https://aws.amazon.com/global-accelerator/faqs/

https://docs.aws.amazon.com/global-accelerator/latest/dg/introduction-how-it-works.html

Check out this AWS Global Accelerator Cheat Sheet:

https://tutorialsdojo.com/aws-global-accelerator/

Question 37: **Incorrect**
A Solutions Architect needs to set up the required compute resources for the application which have workloads that require high, sequential read and write access to very large data sets on local storage.

Which of the following instance type is the most suitable one to use in this scenario?

- **General Purpose Instances**
- **Memory Optimized Instances**

  (Incorrect)
- **Storage Optimized Instances**

  (Correct)
- **Compute Optimized Instances**

Explanation
Storage optimized instances are designed for workloads that require high, sequential read and write access to very large data sets on local storage. They are optimized to deliver tens of thousands of low-latency, random I/O operations per second (IOPS) to applications.

C: Compute Optimized Instances	**Cost-effective high performance at a low price per compute ratio**
D: Storage Optimized Instances	High disk throughput
G: Accelerated Computing Instances	Graphics-intensive GPU instances
H: Storage Optimized Instances	HDD-based local storage for high disk throughput
I: Storage Optimized Instances	High storage instances, low latency, high random I/O performance, high sequential read throughput, and high IOPS
M: General Purpose Instances	Fixed performance
P: Accelerated Computing Instances	General purpose GPU instances
F: Accelerated Computing Instances	Reconfigurable FPGA instances
R: Memory Optimized Instances	Memory-intensive applications
T: General Purpose Instances	Burstable performance instances
X: Memory Optimized Instances	Large-scale, enterprise-class, in-memory applications, and high-performance databases

Hence, the correct answer is: **Storage Optimized Instances.**

Memory Optimized Instances is incorrect because these are designed to deliver fast performance for workloads that process large data sets in memory, which is quite different from handling high read and write capacity on local storage.

Compute Optimized Instances is incorrect because these are ideal for compute-bound applications that benefit from high-performance processors, such as batch processing workloads and media transcoding.

General Purpose Instances is incorrect because these are the most basic type of instances. They provide a balance of compute, memory, and networking resources, and can be used for a variety of workloads. Since you are requiring higher read and write capacity, storage optimized instances should be selected instead.

Reference:

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/storage-optimized-instances.html

Amazon EC2 Overview:

https://www.youtube.com/watch?v=7VsGIHT_jQE

Check out this Amazon EC2 Cheat Sheet:

https://tutorialsdojo.com/amazon-elastic-compute-cloud-amazon-ec2/

Question 38: **Correct**
A company has an application that uses multiple EC2 instances located in various AWS regions such as US East (Ohio), US West (N. California), and EU (Ireland). The manager instructed the Solutions Architect to set up a latency-based routing to route incoming traffic for www.tutorialsdojo.com to all the EC2 instances across all AWS regions.

Which of the following options can satisfy the given requirement?

- **Use a Network Load Balancer to distribute the load to the multiple EC2 instances across all AWS Regions.**
- **Use Route 53 to distribute the load to the multiple EC2 instances across all AWS Regions.**

  (Correct)
- **Use an Application Load Balancer to distribute the load to the multiple EC2 instances across all AWS Regions.**

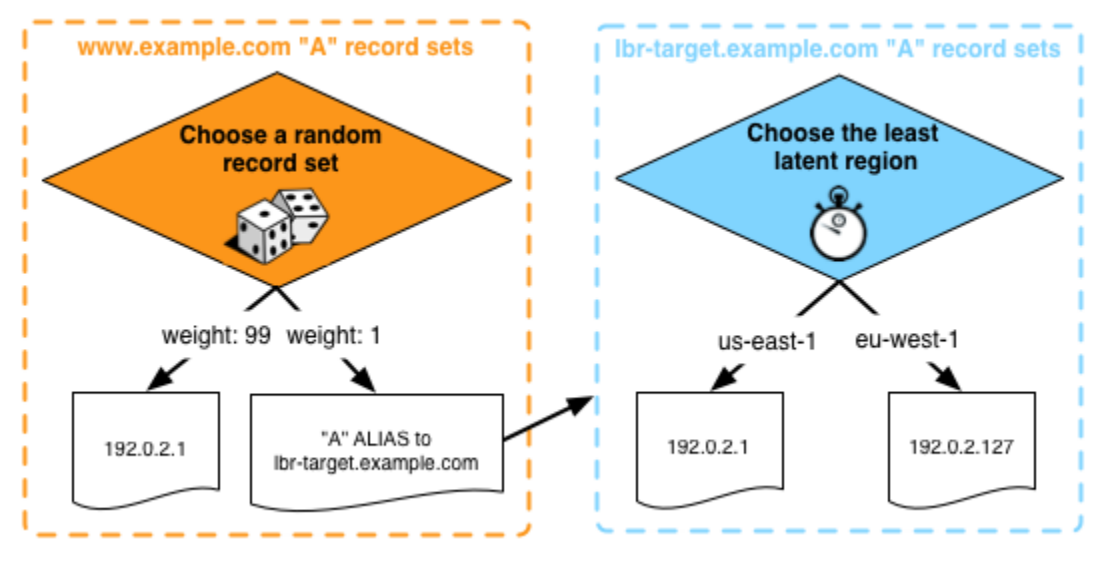
Use AWS DataSync to distribute the load to the multiple EC2 instances across all AWS Regions.

Explanation

If your application is hosted in multiple AWS Regions, you can improve performance for your users by serving their requests from the AWS Region that provides the lowest latency.

You can create latency records for your resources in multiple AWS Regions by using latency-based routing. In the event that Route 53 receives a DNS query for your domain or subdomain such as tutorialsdojo.com or portal.tutorialsdojo.com, it determines which AWS Regions you've created latency records for, determines which region gives the user the lowest latency, and then selects a latency record for that region. Route 53 responds with the value from the selected record which can be the IP address for a web server or the CNAME of your elastic load balancer.

Hence, **using Route 53 to distribute the load to the multiple EC2 instances across all AWS Regions** is the correct answer.



Using a Network Load Balancer to distribute the load to the multiple EC2 instances across all AWS Regions and **using an Application Load Balancer to distribute the load to the multiple EC2 instances across all AWS Regions** are both incorrect because load balancers distribute traffic only within their respective regions and not to other AWS regions by default. Although Network Load Balancers support connections from clients to IP-based targets in peered VPCs across different AWS Regions, the scenario didn't mention that the VPCs have peered with each other. It is best to use Route 53 instead to balance the incoming load to two or more AWS regions more effectively.

Using AWS DataSync to distribute the load to the multiple EC2 instances across all AWS Regions is incorrect because the AWS DataSync service simply provides a fast

way to move large amounts of data online between on-premises storage and Amazon S3 or Amazon Elastic File System (Amazon EFS).

References:

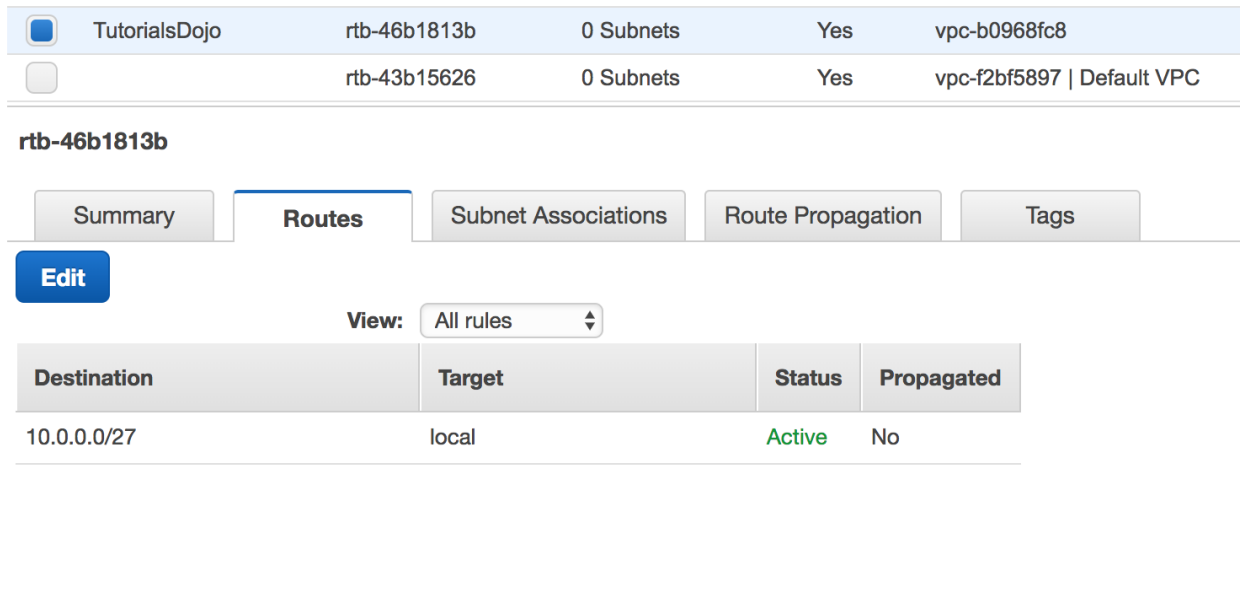
https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/routing-policy.html#routing-policy-latency

https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/TutorialAddingLBRRegion.html

Check out this Amazon Route 53 Cheat Sheet:

https://tutorialsdojo.com/amazon-route-53/

Question 39: **Correct**
A company created a VPC with a single subnet then launched an On-Demand EC2 instance in that subnet. You have attached an Internet gateway (IGW) to the VPC and verified that the EC2 instance has a public IP. The main route table of the VPC is as shown below:



However, the instance still cannot be reached from the Internet when you tried to connect to it from your computer. Which of the following should be made to the route table to fix this issue?

-

Add the following entry to the route table: 10.0.0.0/27 -> Your Internet Gateway

- **Add this new entry to the route table: 0.0.0.0/0 -> Your Internet Gateway**

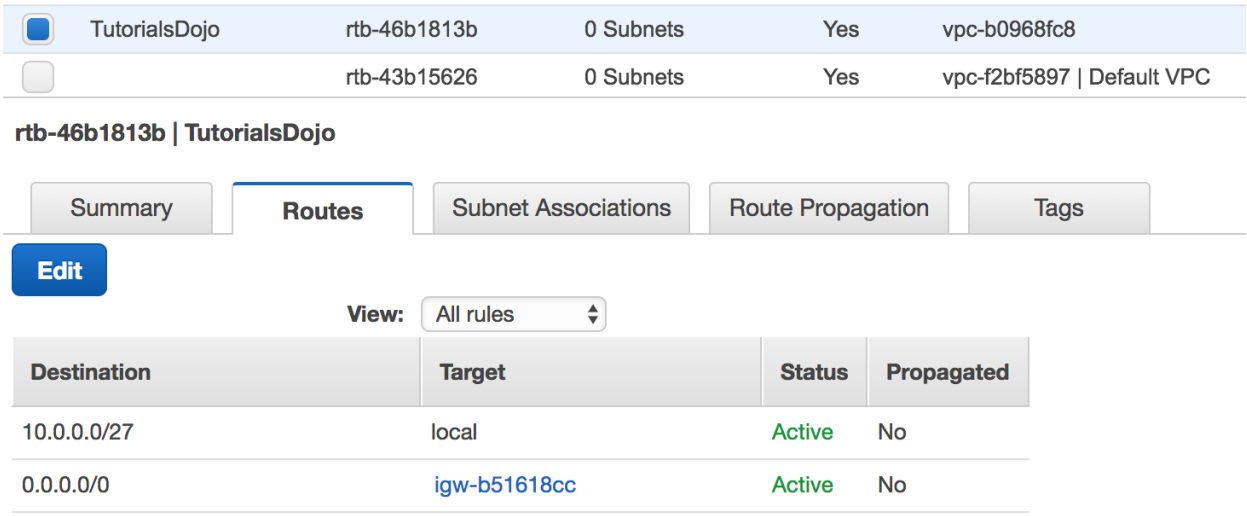
  (Correct)

- **Modify the above route table: 10.0.0.0/27 -> Your Internet Gateway**

- **Add this new entry to the route table: 0.0.0.0/27 -> Your Internet Gateway**

Explanation
Apparently, the route table does not have an entry for the Internet Gateway. This is why you cannot connect to the EC2 instance. To fix this, you have to add a route with a destination of `0.0.0.0/0` for IPv4 traffic or `::/0` for IPv6 traffic, and then a target of the Internet gateway ID (`igw-xxxxxxxx`).

This should be the correct route table configuration after adding the new entry.



Reference:

http://docs.aws.amazon.com/AmazonVPC/latest/UserGuide/VPC_Route_Tables.html

Check out this Amazon VPC Cheat Sheet:

https://tutorialsdojo.com/amazon-vpc/

Question 40: **Correct**
A company has an application architecture that stores both the access key ID and the secret access key in a plain text file on a custom Amazon Machine Image (AMI). The EC2 instances, which are created by using this AMI, are using the stored access keys to connect to a DynamoDB table.

What should the Solutions Architect do to make the current architecture more secure?

- **Put the access keys in an Amazon S3 bucket instead.**
- **Put the access keys in Amazon Glacier instead.**
- **Do nothing. The architecture is already secure because the access keys are already in the Amazon Machine Image.**
- **Remove the stored access keys in the AMI. Create a new IAM role with permissions to access the DynamoDB table and assign it to the EC2 instances.**

  (Correct)

Explanation
You should use an IAM role to manage *temporary* credentials for applications that run on an EC2 instance. When you use an IAM role, you don't have to distribute long-term credentials (such as a user name and password or access keys) to an EC2 instance.

Instead, the role supplies temporary permissions that applications can use when they make calls to other AWS resources. When you launch an EC2 instance, you specify an IAM role to associate with the instance. Applications that run on the instance can then use the role-supplied temporary credentials to sign API requests.

Hence, the best option here is to remove the stored access keys first in the AMI. Then, create a new IAM role with permissions to access the DynamoDB table and assign it to the EC2 instances.

Putting the access keys in Amazon Glacier or **in an Amazon S3 bucket** are incorrect because S3 and Glacier are mainly used as a storage option. It is better to use an IAM role instead of storing access keys in these storage services.

The option that says: **Do nothing. The architecture is already secure because the access keys are already in the Amazon Machine Image** is incorrect because you can make the architecture more secure by using IAM.

Reference:

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_use_switch-role-ec2.html

Check out this AWS Identity & Access Management (IAM) Cheat Sheet:

https://tutorialsdojo.com/aws-identity-and-access-management-iam/

Question 41: **Correct**
A company is looking for a way to analyze the calls between customers and service agents. Each conversation is transcribed, JSON-formatted, and saved to an Amazon S3 bucket. The company's solutions architect is tasked to design a solution for extracting and visualizing sentiments from the transcribed files.

Which solution meets the requirements while minimizing the amount of operational overhead?

- **Create an Amazon Comprehend analysis job. Index the sentiment along with the transcript to an Amazon OpenSearch cluster. Visualize the results using Amazon Managed Grafana.**

- **Train a custom Natural Language Processing (NLP) model using Amazon SageMaker. Index the sentiment along with the transcript to an Amazon OpenSearch cluster. Visualize the results using the OpenSearch Dashboard.**

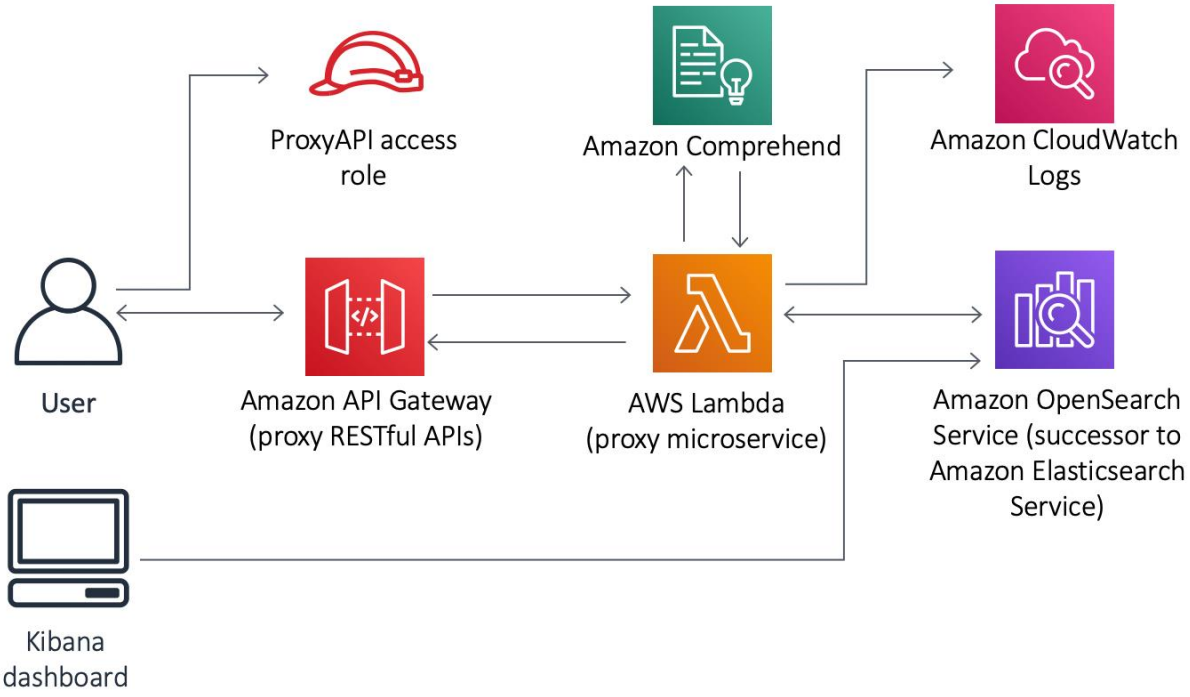
- **Analyze the JSON files with Amazon Textract. Index the sentiment along with the transcript to an Amazon OpenSearch cluster. Visualize the results using Amazon Managed Grafana.**

- **Create an Amazon Comprehend analysis job. Index the sentiment along with the transcript to an Amazon OpenSearch cluster. Visualize the results using the OpenSearch Dashboard.**

  (Correct)

Explanation

Amazon Comprehend uses machine learning to help you uncover the insights and relationships in your unstructured data. The service identifies the language of the text; extracts key phrases, places, people, brands, or events; understands how positive or negative the text is; analyzes text using tokenization and parts of speech, and automatically organizes a collection of text files by topic. You can also use AutoML capabilities in Amazon Comprehend to build a custom set of entities or text classification models that are tailored uniquely to your organization's needs.



In this scenario, you can build the application with the help of Amazon Comprehend. You could expose the application through a RESTful endpoint, have it invoke a Lambda function that will call Amazon Comprehend for sentiment analysis, and index data into an Amazon OpenSearch cluster.

Hence, the correct answer is: **Create an Amazon Comprehend analysis job. Index the sentiment along with the transcript to an Amazon OpenSearch cluster. Visualize the results using the OpenSearch Dashboard.**

The option that says: **Analyze the JSON files with Amazon Textract. Index the sentiment along with the transcript to an Amazon OpenSearch cluster. Visualize the results using Amazon Managed Grafana** is incorrect. Amazon Textract is just an AI service used to extract text data from scanned documents in PNG, JPEG, TIFF, PDF formats and is not capable of running sentiment analysis. Furthermore, Grafana is more suited for the visualization of time-series data such as system metrics (CPU load, disk storage, memory utilization, temperature, etc). While there are hacks you can use to visualize non-time series like the one in the scenario, they come with additional overhead on your part. The built-in OpenSearch dashboard is enough to do the job.

The option that says: **Create an Amazon Comprehend analysis job. Index the sentiment along with the transcript to an Amazon OpenSearch cluster. Visualize the results using Amazon Managed Grafana** is incorrect. The Amazon OpenSearch dashboard is a more suitable service to use than Grafana since the sentiment data is already processed by an Amazon OpenSearch cluster. This solution needs a separate AWS data source configuration in the Grafana workspace console to integrate and read the sentiment data, which entails an additional operational overhead.

The option that says: **Train a custom Natural Language Processing (NLP) model using Amazon SageMaker. Index the sentiment along with the transcript to an Amazon OpenSearch cluster. Visualize the results using the Amazon QuickSight Dashboard** is incorrect. Although this may be a viable option, training your own ML model rather than using the readily available Amazon Comprehend service requires more time and effort. The same is true in using the Amazon QuickSight dashboard instead of the OpenSearch Dashboard. It takes a lot of steps to properly integrate Amazon QuickSight and an OpenSearch cluster which might cause delays in the project implementation.

References:

https://aws.amazon.com/solutions/implementations/text-analysis-with-amazon-opensearch-service-and-amazon-comprehend/

https://docs.aws.amazon.com/opensearch-service/latest/developerguide/walkthrough.html#walkthrough-analysis

Check out this Amazon Comprehend Cheat Sheet:

https://tutorialsdojo.com/amazon-comprehend/

Question 42: **Correct**
A Python application running on VMware Cloud on AWS must connect to an Amazon DynamoDB table called `tutorialsdojo`. Considering the principle of least privilege access, a solutions architect must form an IAM policy that allows the application to read, write, update and delete items from the `tutorialsdojo` table only.

Which IAM policy would satisfy the requirements?

- 

```
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Action": [
"dynamodb:Put*",
"dynamodb:Delete*",
"dynamodb:Get*",
"dynamodb:Update*"
],
"Resource": "arn:aws:dynamodb:us-east-
2:123456789012:table/tutorialsdojo"
}
]
}
```

- 

```
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Action": [
"dynamodb:PutItem",
"dynamodb:DeleteItem",
"dynamodb:GetItem",
"dynamodb:UpdateItem"
],
```

```
"Resource": "arn:aws:dynamodb:us-east-2:123456789012:table/*"
}
]
}
```

-

```
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Action": "*",
"Resource": "arn:aws:dynamodb:us-east-
2:123456789012:table/tutorialsdojo"
}
]
}
```

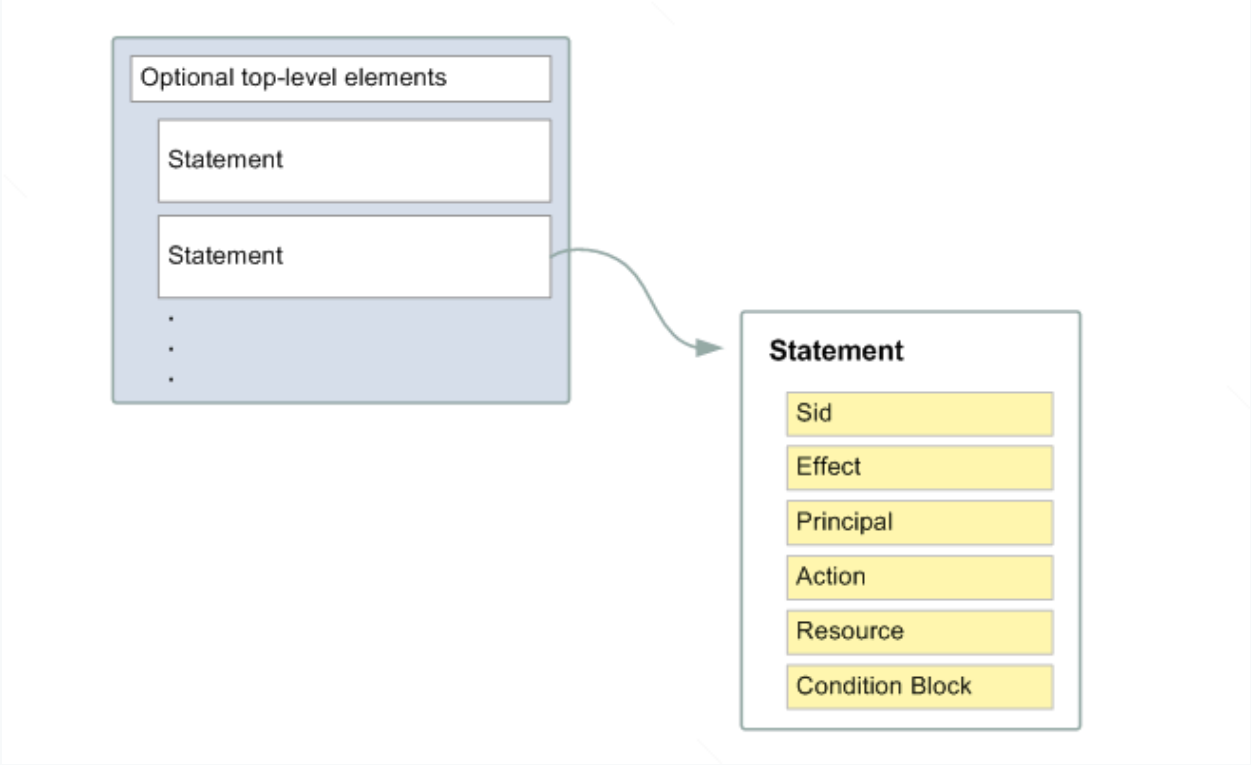
-

```
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Action": [
"dynamodb:PutItem",
"dynamodb:DeleteItem",
"dynamodb:GetItem",
"dynamodb:UpdateItem"
],
"Resource": "arn:aws:dynamodb:us-east-
2:123456789012:table/tutorialsdojo"
}
]
}
```

(Correct)

Explanation

You manage access in AWS by creating policies and attaching them to IAM identities (users, groups of users, or roles) or AWS resources. A policy is an object in AWS that, when associated with an identity or resource, defines its permissions. AWS evaluates these policies when an IAM principal (user or role) makes a request. Permissions in the policies determine whether the request is allowed or denied.



There are various elements that make up an IAM policy statement, the following are required when creating an IAM policy for IAM Identities:

- **Effect**: The effect can be **Allow** or **Deny**. By default, IAM users don't have permission to use resources and API actions, so all requests are denied. An explicit allow overrides the default. An explicit deny overrides any allows.

- **Action**: The action is the specific API action for which you are granting or denying permission.

- **Resource**: The resource that's affected by the action. You specify a resource using an Amazon Resource Name (ARN).

IAM supports the “*” wildcard in both the resources and actions attributes, making it easier to automatically select multiple matching items. This grants broad permissions, often for many permissions or resources.

Since the scenario is asking for granular control (read, write, delete) over a particular DynamoDB table, all options containing wildcards must be eliminated.

Hence, the correct answer is:

```
1. {
2.   "Version": "2012-10-17",
```

```
3.    "Statement": [
4.      {
5.      "Effect": "Allow",
6.      "Action": [
7.      "dynamodb:PutItem",
8.      "dynamodb:DeleteItem",
9.      "dynamodb:GetItem",
10.     "dynamodb:UpdateItem"
11.     ],
12.     "Resource": "arn:aws:dynamodb:us-east-2:123456789012:table/tutorialsdojo"
13.     }
14.   ]
15.   }
16. }
```

The following policy is incorrect because it does not adhere to the principle of least privilege access due to the use of wildcards. Remember that the scenario mentioned that the user should only have the permission to read, write, update and delete an item. Therefore, you have to explicitly mention the specific APIs to allow these operations – GetItem, PutItem, UpdateItem and DeleteItem respectively. If you use the wildcard for Update*, then it will also include other actions that the user is not authorized to invoke such as UpdateContinuousBackups, UpdateContributorInsights, UpdateGlobalTable, UpdateGlobalTableSettings et cetera:

```
1. {
2.     "Version": "2012-10-17",
3.     "Statement": [{
4.         "Effect": "Allow",
5.         "Action": [
6.             "dynamodb:Put*",
7.             "dynamodb:Delete*",
8.             "dynamodb:Get*",
9.             "dynamodb:Update*"
10.        ],
11.        "Resource": "arn:aws:dynamodb:us-east-
   2:123456789012:table/tutorialsdojo"
12.    }]
13. }
14.
```

The policy below is quite insecure since it is used a DynamoDB wildcard in the "Resource field" of the IAM Policy. The scenario clearly says that you have to allow access to certain operations for the tutorialsdojo table only, and not for all DynamoDB tables in your AWS account. Thus, this policy is incorrect:

```
1. {
2.   "Version": "2012-10-17",
3.   "Statement": [
4.     {
5.     "Effect": "Allow",
```

```
    "Action": [
    "dynamodb:PutItem",
    "dynamodb:DeleteItem",
    "dynamodb:GetItem",
    "dynamodb:UpdateItem"
    ],
    "Resource": "arn:aws:dynamodb:us-east-2:123456789012:table/*"
    }
  ]
  }
}
```

Lastly, the IAM Policy depicted below is not right since you are technically allowing all API operations for the tutorialsdojo DynamoDB table:

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": "*",
        "Resource": "arn:aws:dynamodb:us-east-
   2:123456789012:table/tutorialsdojo"
    }]
}
```

References:

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/iam-policy-structure.html

https://docs.aws.amazon.com/lambda/latest/operatorguide/wildcard-permissions-iam.html

Check out this AWS IAM Cheat Sheet:

https://tutorialsdojo.com/aws-identity-and-access-management-iam/

Question 43: **Incorrect**
A global news network created a CloudFront distribution for their web application. However, you noticed that the application's origin server is being hit for each request instead of the AWS Edge locations, which serve the cached objects. The issue occurs even for the commonly requested objects.

What could be a possible cause of this issue?

- **There are two primary origins configured in your Amazon CloudFront Origin Group.**

  (Incorrect)

- **The Cache-Control max-age directive is set to zero.**

  (Correct)

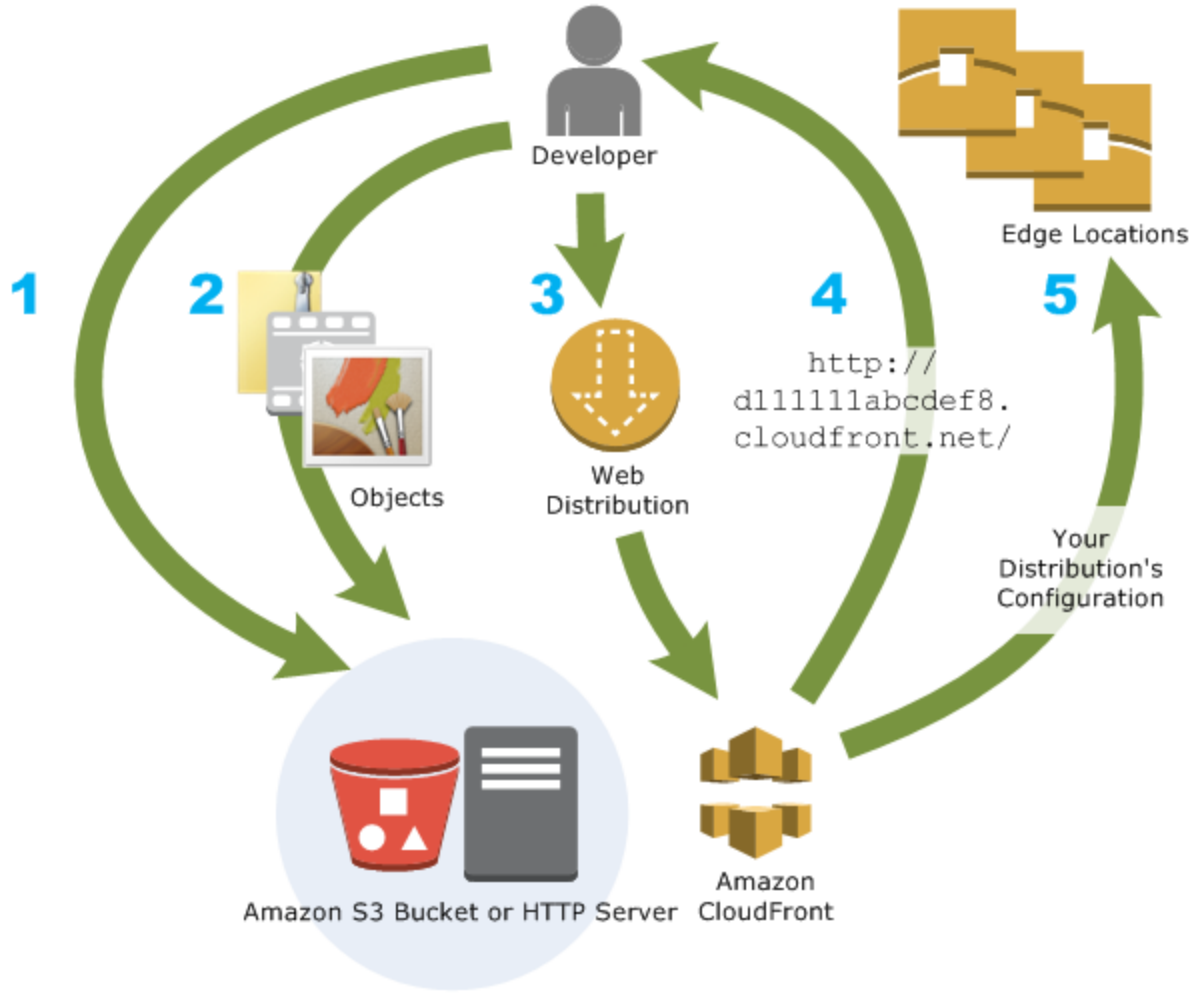
- **An object is only cached by Cloudfront once a successful request has been made hence, the objects were not requested before, which is why the request is still directed to the origin server.**

- **The file sizes of the cached objects are too large for CloudFront to handle.**

Explanation

You can control how long your objects stay in a CloudFront cache before CloudFront forwards another request to your origin. Reducing the duration allows you to serve dynamic content. Increasing the duration means your users get better performance because your objects are more likely to be served directly from the edge cache. A longer duration also reduces the load on your origin.

Typically, CloudFront serves an object from an edge location until the cache duration that you specified passes — that is, until the object expires. After it expires, the next time the edge location gets a user request for the object, CloudFront forwards the request to the origin server to verify that the cache contains the latest version of the object.



The `Cache-Control` and `Expires` headers control how long objects stay in the cache. The `Cache-Control max-age` directive lets you specify how long (in seconds) you want an object to remain in the cache before CloudFront gets the object again from the origin server. The minimum expiration time CloudFront supports is 0 seconds for web distributions and 3600 seconds for RTMP distributions.

In this scenario, the main culprit is that the Cache-Control max-age directive is set to a low value, which is why the request is always directed to your origin server.

Hence, the correct answer is: **The Cache-Control max-age directive is set to zero.**

The option that says: **An object is only cached by CloudFront once a successful request has been made hence, the objects were not requested before, which is why the request is still directed to the origin server** is incorrect because the issue also occurs even for the commonly requested objects. This means that these objects were successfully requested before but due to a zero Cache-Control max-age directive value, it causes this issue in CloudFront.

The option that says: **The file sizes of the cached objects are too large for CloudFront to handle** is incorrect because this is not related to the issue in caching.

The option that says: **There are two primary origins configured in your Amazon CloudFront Origin Group** is incorrect because you cannot set two origins in CloudFront in the first place. An origin group includes two origins which are the primary origin and the second origin that will be used for the actual failover. It also includes the failover criteria that you need to specify. In this scenario, the issue is more on the cache hit ratio and not on origin failovers.

Reference:

http://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/Expiration.html

Check out this Amazon CloudFront Cheat Sheet:

https://tutorialsdojo.com/amazon-cloudfront/

Question 44: **Correct**
A web application, which is hosted in your on-premises data center and uses a MySQL database, must be migrated to AWS Cloud. You need to ensure that the network traffic to and from your RDS database instance is encrypted using SSL. For improved security, you have to use the profile credentials specific to your EC2 instance to access your database, instead of a password.

Which of the following should you do to meet the above requirement?

- **Launch a new RDS database instance with the Backtrack feature enabled.**
- **Set up an RDS database and enable the IAM DB Authentication.**

  (Correct)

Launch the mysql client using the `--ssl-ca` parameter when connecting to the database.

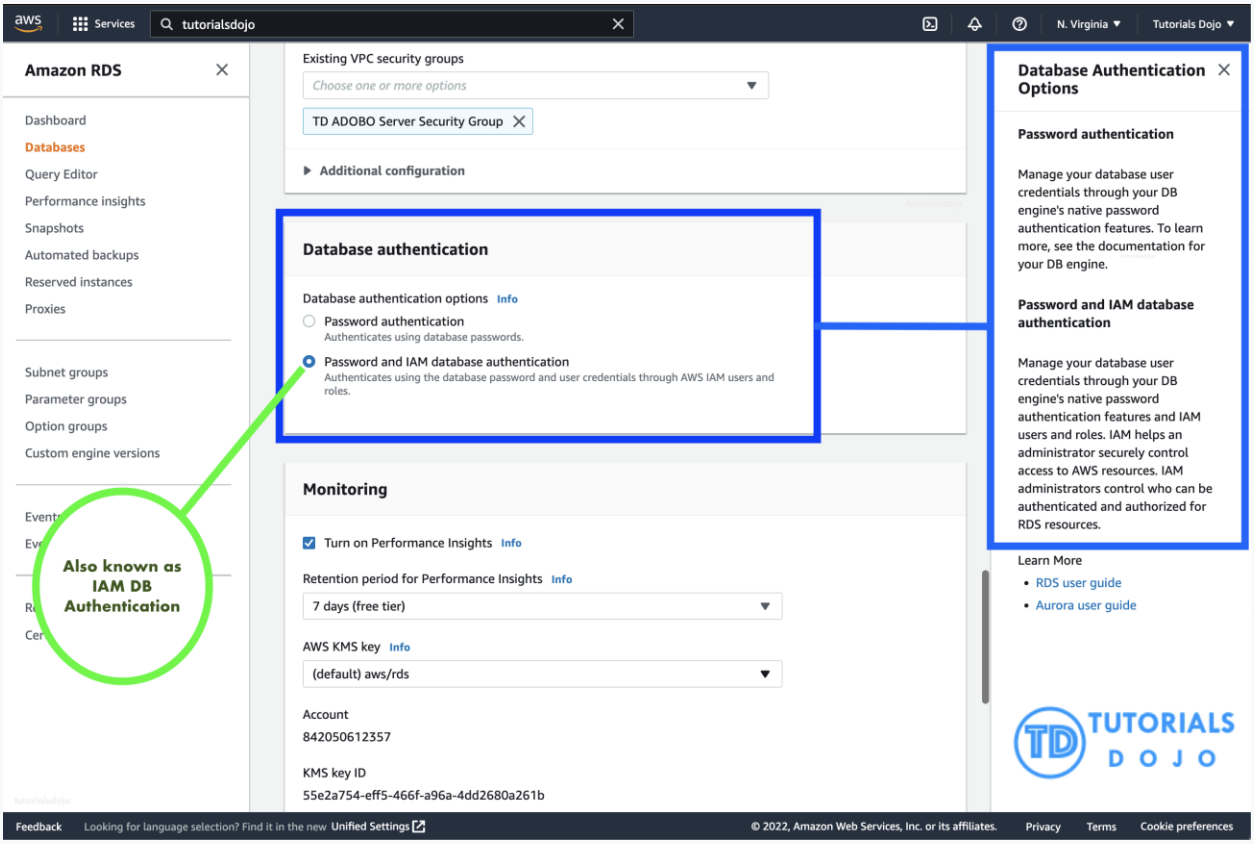
- 

Configure your RDS database to enable encryption.

Explanation

You can authenticate to your DB instance using AWS Identity and Access Management (IAM) database authentication. IAM database authentication works with MySQL and PostgreSQL. With this authentication method, you don't need to use a password when you connect to a DB instance. Instead, you use an authentication token.

An *authentication token* is a unique string of characters that Amazon RDS generates on request. Authentication tokens are generated using AWS Signature Version 4. Each token has a lifetime of 15 minutes. You don't need to store user credentials in the database, because authentication is managed externally using IAM. You can also still use standard database authentication.



IAM database authentication provides the following benefits:

- Network traffic to and from the database is encrypted using Secure Sockets Layer (SSL).

- You can use IAM to centrally manage access to your database resources, instead of managing access individually on each DB instance.

- For applications running on Amazon EC2, you can use profile credentials specific to your EC2 instance to access your database instead of a password, for greater security

Hence, **setting up an RDS database and enable the IAM DB Authentication** is the correct answer since IAM DB Authentication allows the use of the profile credentials specific to your EC2 instance to access your RDS database instead of a password.

The option that says: **Launch a new RDS database instance with the Backtrack feature enabled** is incorrect because the Backtrack feature simply "rewinds" the DB cluster to the time you specify. Backtracking is not a replacement for backing up your DB cluster so that you can restore it to a point in time. However, you can easily undo mistakes using the backtrack feature if you mistakenly perform a destructive action, such as a `DELETE` without a `WHERE` clause.

The option that says: **Configure your RDS database to enable encryption** is incorrect because this encryption feature in RDS is mainly for securing your Amazon RDS DB instances and snapshots at rest. The data that is encrypted at rest includes the underlying storage for DB instances, its automated backups, Read Replicas, and snapshots.

The option that says: **Launch the mysql client using the `--ssl-ca` parameter when connecting to the database** is incorrect because even though using the `--ssl-ca` parameter can provide SSL connection to your database, you still need to use IAM database connection to use the profile credentials specific to your EC2 instance to access your database instead of a password.

References:

https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/UsingWithRDS.IAMDBAuth.html

https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/UsingWithRDS.IAMDBAuth.Connecting.html

Check out this Amazon RDS cheat sheet:

https://tutorialsdojo.com/amazon-relational-database-service-amazon-rds/

Question 45: **Incorrect**
A company requires corporate IT governance and cost oversight of all of its AWS resources across its divisions around the world. Their corporate divisions want to maintain administrative control of the discrete AWS resources they consume and ensure that those resources are separate from other divisions.

Which of the following options will support the autonomy of each corporate division while enabling the corporate IT to maintain governance and cost oversight? (Select TWO.)

- ☑ **Use AWS Consolidated Billing by creating AWS Organizations to link the divisions' accounts to a parent corporate account.**

  (Correct)

- ☐ **Enable IAM cross-account access for all corporate IT administrators in each child account.**

  (Correct)

- ☐ **Use AWS Trusted Advisor and AWS Resource Groups Tag Editor**

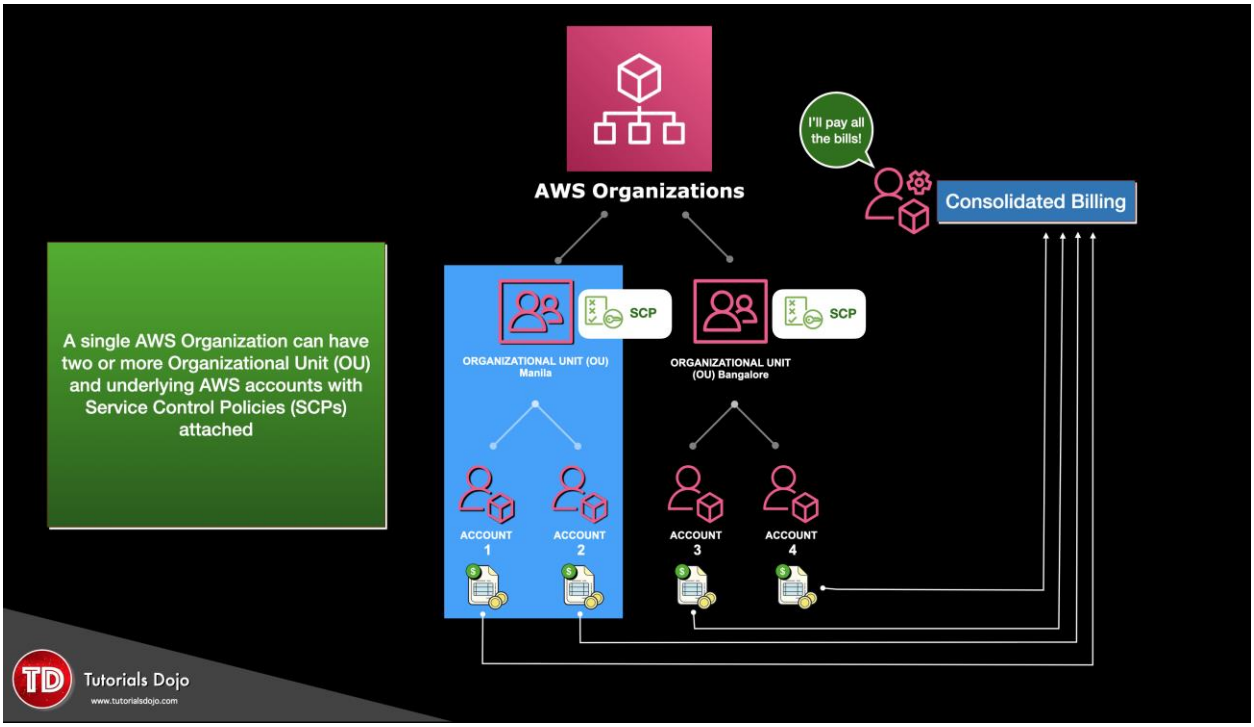
- ☑ **Create separate VPCs for each division within the corporate IT AWS account. Launch an AWS Transit Gateway with equal-cost multipath routing (ECMP) and VPN tunnels for intra-VPC communication.**

  (Incorrect)

- ☐ **Create separate Availability Zones for each division within the corporate IT AWS account. Improve communication between the two AZs using the AWS Global Accelerator.**

Explanation
You can use an IAM role to delegate access to resources that are in different AWS accounts that you own. You share resources in one account with users in a different account. By setting up cross-account access in this way, you don't need to create

individual IAM users in each account. In addition, users don't have to sign out of one account and sign into another in order to access resources that are in different AWS accounts.



You can use the consolidated billing feature in AWS Organizations to consolidate payment for multiple AWS accounts or multiple AISPL accounts. With consolidated billing, you can see a combined view of AWS charges incurred by all of your accounts. You can also get a cost report for each member account that is associated with your master account. Consolidated billing is offered at no additional charge. AWS and AISPL accounts can't be consolidated together.

The combined use of IAM and Consolidated Billing will support the autonomy of each corporate division while enabling corporate IT to maintain governance and cost oversight. Hence, the correct choices are:

- Enable IAM cross-account access for all corporate IT administrators in each child account

- Use AWS Consolidated Billing by creating AWS Organizations to link the divisions' accounts to a parent corporate account

Using AWS Trusted Advisor and AWS Resource Groups Tag Editor is incorrect. Trusted Advisor is an online tool that provides you real-time guidance to help you provision your resources following AWS best practices. It only provides you alerts on areas where you do not adhere to best practices and tells you how to improve them. It does not assist in

maintaining governance over your AWS accounts. Additionally, the AWS Resource Groups Tag Editor simply allows you to add, edit, and delete tags to multiple AWS resources at once for easier identification and monitoring.

Creating separate VPCs for each division within the corporate IT AWS account. Launch an AWS Transit Gateway with equal-cost multipath routing (ECMP) and VPN tunnels for intra-VPC communication is incorrect because creating separate VPCs would not separate the divisions from each other since they will still be operating under the same account and therefore contribute to the same billing each month. AWS Transit Gateway connects VPCs and on-premises networks through a central hub and acts as a cloud router where each new connection is only made once. For this particular scenario, it is suitable to use AWS Organizations instead of setting up an AWS Transit Gateway since the objective is for maintaining administrative control of the AWS resources and not for network connectivity.

Creating separate Availability Zones for each division within the corporate IT AWS account. Improve communication between the two AZs using the AWS Global Accelerator is incorrect because you do not need to create Availability Zones. They are already provided for you by AWS right from the start, and not all services support multiple AZ deployments. In addition, having separate Availability Zones in your VPC does not meet the requirement of supporting the autonomy of each corporate division. The AWS Global Accelerator is a service that uses the AWS global network to optimize the network path from your users to your applications and not between your Availability Zones.

References:

http://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/consolidated-billing.html

https://docs.aws.amazon.com/IAM/latest/UserGuide/tutorial_cross-account-with-roles.html

Check out this AWS Billing and Cost Management Cheat Sheet:

https://tutorialsdojo.com/aws-billing-and-cost-management/

Question 46: **Incorrect**
A company plans to launch an application that tracks the GPS coordinates of delivery trucks in the country. The coordinates are transmitted from each delivery truck every five seconds. You need to design an architecture that will enable real-time processing of

these coordinates from multiple consumers. The aggregated data will be analyzed in a separate reporting application.

Which AWS service should you use for this scenario?

- **Amazon Kinesis**

  (Correct)

- **Amazon Simple Queue Service**

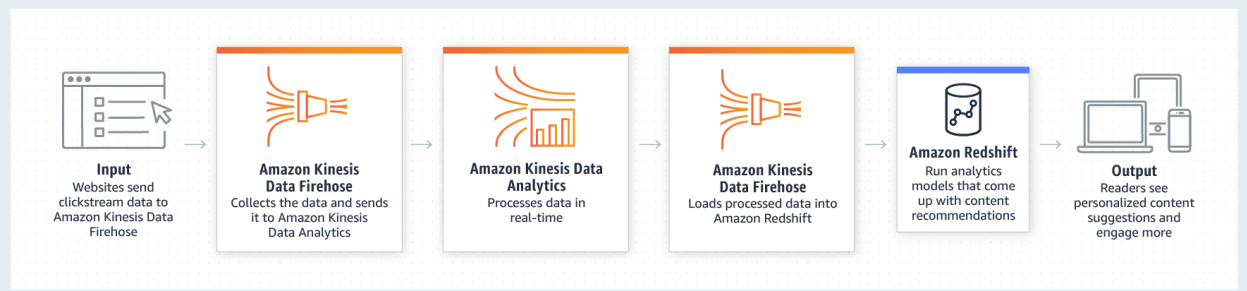
- **Amazon AppStream**

- **AWS Data Pipeline**

  (Incorrect)

Explanation

Amazon Kinesis makes it easy to collect, process, and analyze real-time, streaming data so you can get timely insights and react quickly to new information. It offers key capabilities to cost-effectively process streaming data at any scale, along with the flexibility to choose the tools that best suit the requirements of your application.



With Amazon Kinesis, you can ingest real-time data such as video, audio, application logs, website clickstreams, and IoT telemetry data for machine learning, analytics, and other applications. Amazon Kinesis enables you to process and analyze data as it arrives and responds instantly instead of having to wait until all your data are collected before the processing can begin.

Reference:

https://aws.amazon.com/kinesis/

Check out this Amazon Kinesis Cheat Sheet:

https://tutorialsdojo.com/amazon-kinesis/

Question 47: **Correct**
A company has several EC2 Reserved Instances in their account that need to be decommissioned and shut down since they are no longer used by the development team. However, the data is still required by the audit team for compliance purposes.

Which of the following steps can be taken in this scenario? (Select TWO.)

- ☑ **Take snapshots of the EBS volumes and terminate the EC2 instances.**

  (Correct)

- ☐ **Convert the EC2 instances to Spot instances with a persistent Spot request type.**

- ☐ **Stop all the running EC2 instances.**

- ☐ **Convert the EC2 instance to On-Demand instances**

- ☑ **You can opt to sell these EC2 instances on the AWS Reserved Instance Marketplace**

  (Correct)

Explanation

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers. Amazon EC2's simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon's proven computing environment.

The first requirement as per the scenario is to decommission and shut down several EC2 Reserved Instances. However, it is also mentioned that the audit team still requires the data for compliance purposes. To fulfill the given requirements, you can first create a snapshot of the instance to save its data and then sell the instance to the Reserved Instance Marketplace.

The Reserved Instance Marketplace is a platform that supports the sale of third-party and AWS customers' unused Standard Reserved Instances, which vary in terms of length and pricing options. For example, you may want to sell Reserved Instances after moving instances to a new AWS region, changing to a new instance type, ending projects before the term expiration, when your business needs change, or if you have unneeded capacity.

Hence, the correct answers are:

- You can opt to sell these EC2 instances on the AWS Reserved Instance Marketplace.

- Take snapshots of the EBS volumes and terminate the EC2 instances.

The option that says: **Convert the EC2 instance to On-Demand instances** is incorrect because it's stated in the scenario that the development team no longer needs several EC2 Reserved Instances. By converting it to On-Demand instances, the company will still have instances running in their infrastructure and this will result in additional costs.

The option that says: **Convert the EC2 instances to Spot instances with a persistent Spot request type** is incorrect because the requirement in the scenario is to terminate or shut down several EC2 Reserved Instances. Converting the existing instances to Spot instances will not satisfy the given requirement.

The option that says: **Stop all the running EC2 instances** is incorrect because doing so will still incur storage cost. Take note that the requirement in the scenario is to decommission and shut down several EC2 Reserved Instances. Therefore, this approach won't fulfill the given requirement.

References:

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ri-market-general.html

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-creating-snapshot.html

Check out this Amazon EC2 Cheat Sheet:

https://tutorialsdojo.com/amazon-elastic-compute-cloud-amazon-ec2/

AWS Container Services Overview:

https://www.youtube.com/watch?v=5QBgDX7O7pw

Question 48: **Correct**
A Solutions Architect is developing a three-tier cryptocurrency web application for a FinTech startup. The Architect has been instructed to restrict access to the database tier to only accept traffic from the application-tier and deny traffic from other sources. The application-tier is composed of application servers hosted in an Auto Scaling group of EC2 instances.

Which of the following options is the MOST suitable solution to implement in this scenario?

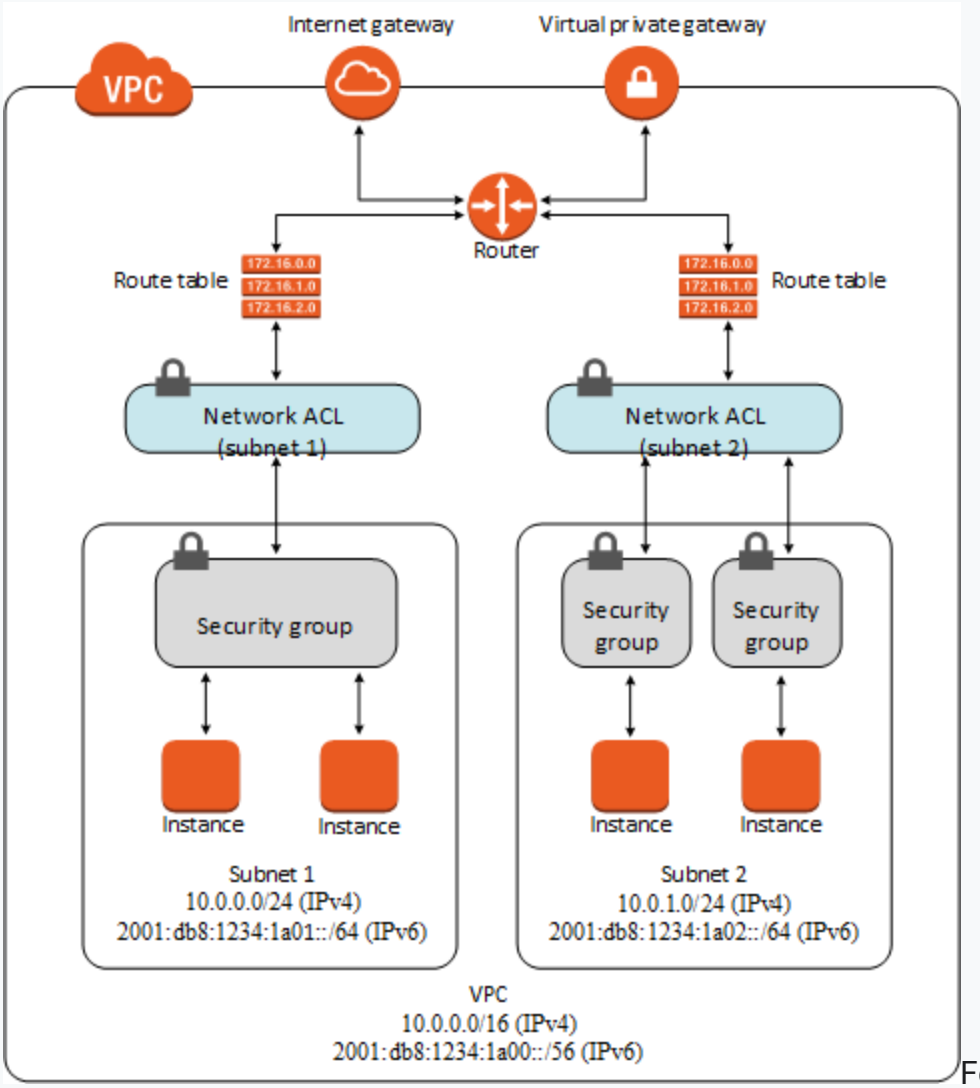
- **Set up the Network ACL of the database subnet to deny all inbound non-database traffic from the subnet of the application-tier.**
- **Set up the security group of the database tier to allow database traffic from a specified list of application server IP addresses.**
- **Set up the Network ACL of the database subnet to allow inbound database traffic from the subnet of the application-tier.**
-

Set up the security group of the database tier to allow database traffic from the security group of the application servers.

(Correct)

Explanation

A **security group** acts as a virtual firewall for your instance to control inbound and outbound traffic. When you launch an instance in a VPC, you can assign up to five security groups to the instance. Security groups act at the instance level, not the subnet level. Therefore, each instance in a subnet in your VPC could be assigned to a different set of security groups. If you don't specify a particular group at launch time, the instance is automatically assigned to the default security group for the VPC.



For each security group, you add *rules* that control the inbound traffic to instances and a separate set of rules that control the outbound traffic. This section describes the basic things you need to know about security groups for your VPC and their rules.

You can add or remove rules for a security group which is also referred to as *authorizing* or *revoking* inbound or outbound access. A rule applies either to inbound traffic (ingress) or outbound traffic (egress). You can grant access to a specific CIDR range or to another security group in your VPC or in a peer VPC (which requires a VPC peering connection).

In the scenario, the servers of the application-tier are in an Auto Scaling group which means that the number of EC2 instances could grow or shrink over time. An Auto Scaling group could also cover one or more Availability Zones (AZ) which have their own subnets. Hence, the most suitable solution would be to **set up the security group of the database tier to allow database traffic from the security group of the application servers** since you can utilize the security group of the application-tier Auto Scaling group as the source for the security group rule in your database tier.

Setting up the security group of the database tier to allow database traffic from a specified list of application server IP addresses is incorrect because the list of application server IP addresses will change over time since an Auto Scaling group can add or remove EC2 instances based on the configured scaling policy. This will create inconsistencies in your application because the newly launched instances, which are not included in the initial list of IP addresses, will not be able to access the database.

Setting up the Network ACL of the database subnet to deny all inbound non-database traffic from the subnet of the application-tier is incorrect because doing this could affect the other EC2 instances of other applications, which are also hosted in the same subnet of the application-tier. For example, a large subnet with a CIDR block of /16 could be shared by several applications. Denying all inbound non-database traffic from the entire subnet will impact other applications which use this subnet.

Setting up the Network ACL of the database subnet to allow inbound database traffic from the subnet of the application-tier is incorrect. Although this solution can work, the subnet of the application-tier could be shared by another tier or another set of EC2 instances other than the application-tier. This means that you would inadvertently be granting database access to unauthorized servers hosted in the same subnet other than the application-tier.

References:

https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Security.html#VPC_Security_Comparison

http://docs.aws.amazon.com/AmazonVPC/latest/UserGuide/VPC_SecurityGroups.html

Check out this Amazon VPC Cheat Sheet:

https://tutorialsdojo.com/amazon-vpc/

Question 49: **Incorrect**
A company is planning to deploy a High Performance Computing (HPC) cluster in its VPC that requires a scalable, high-performance file system. The storage service must be optimized for efficient workload processing, and the data must be accessible via a fast and scalable file system interface. It should also work natively with Amazon S3 that enables you to easily process your S3 data with a high-performance POSIX interface.

Which of the following is the MOST suitable service that you should use for this scenario?

- **Amazon Elastic Block Storage (EBS)**
- **Amazon Elastic File System (EFS)**

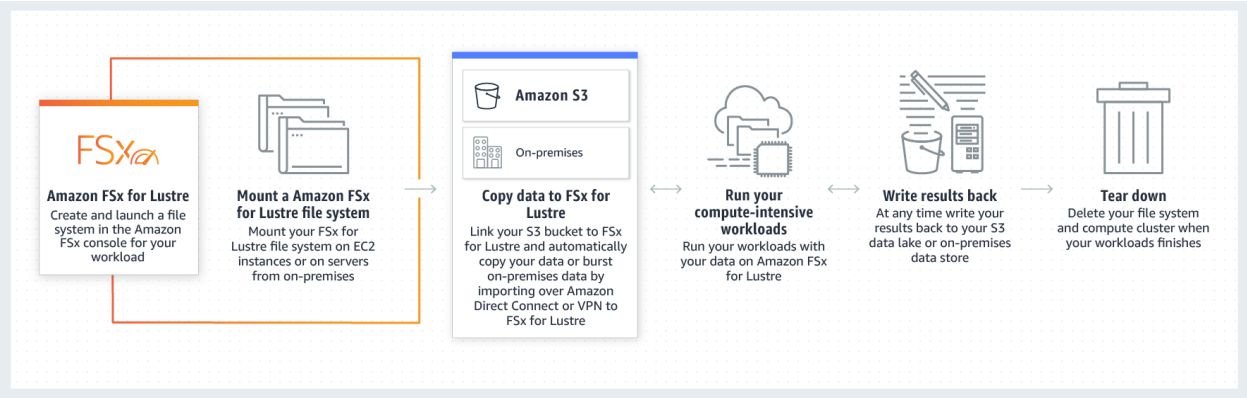
  (Incorrect)
- **Amazon FSx for Lustre**

  (Correct)
- **Amazon FSx for Windows File Server**

Explanation
Amazon FSx for Lustre provides a high-performance file system optimized for fast processing of workloads such as machine learning, high performance computing (HPC), video processing, financial modeling, and electronic design automation (EDA). These workloads commonly require data to be presented via a fast and scalable file system interface and typically have data sets stored on long-term data stores like Amazon S3.

Operating high-performance file systems typically requires specialized expertise and administrative overhead, requiring you to provision storage servers and tune complex performance parameters. With Amazon FSx, you can launch and run a file system that provides sub-millisecond access to your data and allows you to read and write data at speeds of up to hundreds of gigabytes per second of throughput and millions of IOPS.

Amazon FSx for Lustre works natively with Amazon S3, making it easy for you to process cloud data sets with high-performance file systems. When linked to an S3 bucket, an FSx for Lustre file system transparently presents S3 objects as files and allows you to write results back to S3. You can also use FSx for Lustre as a standalone high-performance file system to burst your workloads from on-premises to the cloud. By copying on-premises data to an FSx for Lustre file system, you can make that data available for fast processing by compute instances running on AWS. With Amazon FSx, you pay for only the resources you use. There are no minimum commitments, upfront hardware or software costs, or additional fees.



For Windows-based applications, Amazon FSx provides fully managed Windows file servers with features and performance optimized for "lift-and-shift" business-critical application workloads including home directories (user shares), media workflows, and ERP applications. It is accessible from Windows and Linux instances via the SMB protocol. If you have Linux-based applications, Amazon EFS is a cloud-native fully managed file system that provides simple, scalable, elastic file storage accessible from Linux instances via the NFS protocol.

For compute-intensive and fast processing workloads, like high-performance computing (HPC), machine learning, EDA, and media processing, Amazon FSx for Lustre, provides a file system that's optimized for performance, with input and output stored on Amazon S3.

Hence, the correct answer is: **Amazon FSx for Lustre.**

Amazon Elastic File System (EFS) is incorrect. Although the EFS service can be used for HPC applications, it doesn't natively work with Amazon S3. It doesn't have the capability to easily process your S3 data with a high-performance POSIX interface, unlike Amazon FSx for Lustre.

Amazon FSx for Windows File Server is incorrect. Although this service is a type of Amazon FSx, it does not work natively with Amazon S3. This service is a fully managed

native Microsoft Windows file system that is primarily used for your Windows-based applications that require shared file storage to AWS.

Amazon Elastic Block Storage (EBS) is incorrect because this service is not a scalable, high-performance file system.

References:

https://aws.amazon.com/fsx/lustre/

https://aws.amazon.com/getting-started/use-cases/hpc/3/

Check out this Amazon FSx Cheat Sheet:

https://tutorialsdojo.com/amazon-fsx/

Question 50: **Correct**
A global medical research company has a molecular imaging system that provides each client with frequently updated images of what is happening inside the human body at the molecular and cellular levels. The system is hosted in AWS and the images are hosted in an S3 bucket behind a CloudFront web distribution. When a fresh batch of images is uploaded to S3, it is required to keep the previous ones in order to prevent them from being overwritten.

Which of the following is the most suitable solution to solve this issue?

- **Invalidate the files in your CloudFront web distribution**
- **Add Cache-Control no-cache, no-store, or private directives in the S3 bucket**
- **Add a separate cache behavior path for the content and configure a custom object caching with a Minimum TTL of 0**
-

Use versioned objects

(Correct)

Explanation
To control the versions of files that are served from your distribution, you can either invalidate files or give them versioned file names. If you want to update your files frequently, AWS recommends that you primarily use file versioning for the following reasons:

- Versioning enables you to control which file a request returns even when the user has a version cached either locally or behind a corporate caching proxy. If you invalidate the file, the user might continue to see the old version until it expires from those caches.

- CloudFront access logs include the names of your files, so versioning makes it easier to analyze the results of file changes.

- Versioning provides a way to serve different versions of files to different users.

- Versioning simplifies rolling forward and back between file revisions.

- Versioning is less expensive. You still have to pay for CloudFront to transfer new versions of your files to edge locations, but you don't have to pay for invalidating files.

Invalidating the files in your CloudFront web distribution is incorrect because even though using invalidation will solve this issue, this solution is more expensive as compared to **using versioned objects**.

Adding a separate cache behavior path for the content and configuring a custom object caching with a Minimum TTL of 0 is incorrect because this alone is not enough to solve the problem. A cache behavior is primarily used to configure a variety of CloudFront functionality for a given URL path pattern for files on your website. Although this solution may work, it is still better to use versioned objects where you can control which image will be returned by the system even when the user has another version cached either locally or behind a corporate caching proxy.

Adding Cache-Control no-cache, no-store, or private directives in the S3 bucket is incorrect because although it is right to configure your origin to add the ***Cache-Control*** or ***Expires*** header field, you should do this to your objects and not on the entire S3 bucket.

References:

https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/UpdatingExistingObjects.html

https://aws.amazon.com/premiumsupport/knowledge-center/prevent-cloudfront-from-caching-files/

https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/Invalidation.html#PayingForInvalidation

Check out this Amazon CloudFront Cheat Sheet:

https://tutorialsdojo.com/amazon-cloudfront/

Question 51: **Correct**

A new DevOps engineer has created a CloudFormation template for a web application and she raised a <code>pull request</code> in GIT for you to check and review. After checking the template, you immediately told her that the template will not work. Which of the following is the reason why this CloudFormation template will fail to deploy the stack?

```
{
 "AWSTemplateFormatVersion":"2010-09-09",
 "Parameters":{
   "VPCId":{
     "Type":"String",
     "Description":"manila"
 },
 "SubnetId":{
   "Type":"String",
   "Description":"subnet-b46032ec"
 }
},
"Outputs":{
 "InstanceId":{
  "Value":{
   "Ref":"manilaInstance"
  },
  "Description":"Instance Id"
 }
}
```

- 

  The `Conditions` section is missing.

- 

The `Resources` section is missing.

(Correct)

- **An invalid section named `Parameters` is present. This will cause the CloudFormation stack to fail.**

- **The value of the `AWSTemplateFormatVersion` is incorrect. It should be 2017-06-06.**

Explanation

In **CloudFormation**, a template is a JSON or a YAML-formatted text file that describes your AWS infrastructure. Templates include several major sections. The Resources section is the only required section. Some sections in a template can be in any order. However, as you build your template, it might be helpful to use the logical ordering of the following list, as values in one section might refer to values from a previous section. Take note that all of the sections here are optional, except for Resources, which is the only one required.

- Format Version

- Description

- Metadata

- Parameters

- Mappings

- Conditions

- Transform

- Resources (required)

- Outputs

Reference:

http://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/template-anatomy.html

Check out this AWS CloudFormation Cheat Sheet:

https://tutorialsdojo.com/aws-cloudformation/

AWS CloudFormation - Templates, Stacks, Change Sets:

https://www.youtube.com/watch?v=9Xpuprxg7aY

Question 52: **Incorrect**
An application is loading hundreds of JSON documents into an Amazon S3 bucket every hour which is registered in AWS Lake Formation as a data catalog. The Data Analytics team uses Amazon Athena to run analyses on these data, but due to the volume, most queries take a long time to complete.

What change should be made to improve the query performance while ensuring data security?

- **Compress the data into GZIP format before storing it in the S3 bucket. Apply an IAM policy with `aws:SourceArn` and `aws:SourceAccount` global condition context keys in Lake Formation that prevents cross-service confused deputy problems and other security issues.**

- **Transform the JSON data into Apache Parquet format. Ensure that the user has an `lakeformation:GetDataAccess` IAM permission for underlying data access control.**

  (Correct)

- **Convert the JSON documents into CSV format. Provide fine-grained named resource access control to specific databases or tables in AWS Lake Formation.**

- 

Apply minification on the data and implement the Lake Formation tag-based access control (LF-TBAC) authorization strategy to ensure security.

(Incorrect)

Explanation
Amazon Athena supports a wide variety of data formats like CSV, TSV, JSON, or Textfiles and also supports open-source columnar formats such as Apache ORC and Apache Parquet. Athena also supports compressed data in Snappy, Zlib, LZO, and GZIP formats. By compressing, partitioning, and using columnar formats you can improve performance and reduce your costs.

Parquet and ORC file formats both support predicate pushdown (also called predicate filtering). Parquet and ORC both have blocks of data that represent column values. Each block holds statistics for the block, such as max/min values. When a query is being executed, these statistics determine whether the block should be read or skipped.

Athena charges you by the amount of data scanned per query. You can save on costs and get better performance if you partition the data, compress data, or convert it to columnar formats such as Apache Parquet.

Dataset	Size on Amazon S3	Query Run time	Data Scanned	Cost
Data stored as CSV files	1 TB	236 seconds	1.15 TB	$5.75
Data stored in Apache Parquet format*	**130 GB**	**6.78 seconds**	**2.51 GB**	**$0.01**
Savings / Speedup	**87% less with Parquet**	**34x faster**	**99% less data scanned**	**99.7% savings**

Apache Parquet is an open-source columnar storage format that is 2x faster to unload and takes up 6x less storage in Amazon S3 as compared to other text formats. One can `COPY` Apache Parquet and Apache ORC file formats from Amazon S3 to your Amazon Redshift cluster. Using AWS Glue, one can configure and run a job to transform CSV data to Parquet. Parquet is a columnar format that is well suited for AWS analytics services like Amazon Athena and Amazon Redshift Spectrum.

When an integrated AWS service requests access to data in an Amazon S3 location that is access-controlled by AWS Lake Formation, Lake Formation supplies temporary credentials to access the data. To enable Lake Formation to control access to underlying data at an Amazon S3 location, you register that location with Lake Formation.

To enable Lake Formation principals to read and write underlying data with access controlled by Lake Formation permissions:

- The Amazon S3 locations that contain the data must be registered with Lake Formation.

- Principals who create Data Catalog tables that point to underlying data locations must have data location permissions.

- Principals who read and write underlying data must have Lake Formation data access permissions on the Data Catalog tables that point to the underlying data locations.

- Principals who read and write underlying data must have the `lakeformation:GetDataAccess` IAM permission.

Thus, the correct answer is: **Transform the JSON data into Apache Parquet format. Ensure that the user has an `lakeformation:GetDataAccess` IAM permission for underlying data access control.**

The option that says: **Convert the JSON documents into CSV format. Provide fine-grained named resource access control to specific databases or tables in AWS Lake Formation** is incorrect because Athena queries performed against row-based formats like CSV are slower than columnar file formats like Apache Parquet.

The option that says:**Apply minification on the data and implement the Lake Formation tag-based access control (LF-TBAC) authorization strategy using IAM Tags to ensure security** is incorrect. Although minifying the JSON file might reduce its overall file size, there won't be a significant difference in terms of querying performance. LF-TBAC is a type of an attribute-based access control (ABAC) that defines permissions based on certain attributes, such as tags in AWS. LF-TBAC uses LF-Tags to grant Lake Formation permissions and not regular IAM Tags.

The option that says: **Compress the data into GZIP format before storing in the S3 bucket. Apply an IAM policy with `aws:SourceArn` and `aws:SourceAccount` global condition context keys in Lake Formation that prevents cross-service confused deputy problems and other security issues.** is incorrect. Compressing the files prior to storing them in Amazon S3 will only save storage costs. As for query performance, it won't have much improvement. In addition, using an IAM Policy to prevent cross-service confused deputy issues is not warranted in this scenario. Having an `lakeformation:GetDataAccess` IAM permission for underlying data access control should suffice.

References:

https://aws.amazon.com/blogs/big-data/top-10-performance-tuning-tips-for-amazon-athena/

https://docs.aws.amazon.com/lake-formation/latest/dg/access-control-underlying-data.html

https://docs.aws.amazon.com/lake-formation/latest/dg/TBAC-overview.html

Check out this Amazon Athena Cheat Sheet:

https://tutorialsdojo.com/amazon-athena/

Question 53: **Incorrect**
A company developed a financial analytics web application hosted in a Docker container using MEAN (MongoDB, Express.js, AngularJS, and Node.js) stack. You want to easily port that web application to AWS Cloud which can automatically handle all the tasks such as balancing load, auto-scaling, monitoring, and placing your containers across your cluster.

Which of the following services can be used to fulfill this requirement?

- **Amazon Elastic Container Service (Amazon ECS)**

  (Incorrect)

- **AWS CloudFormation**

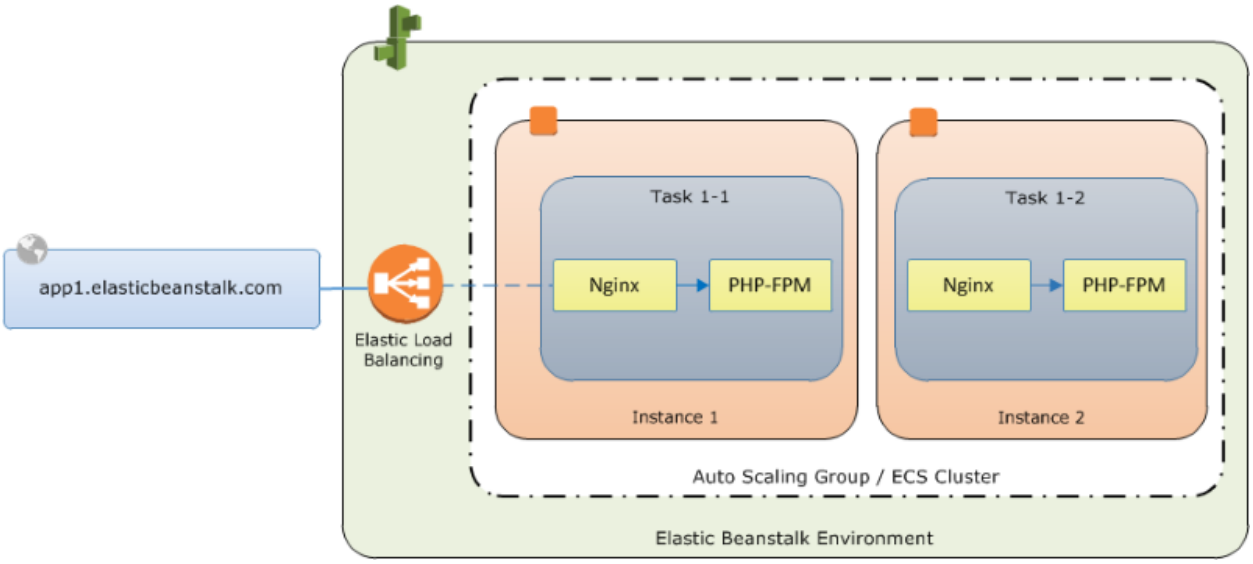
- **AWS Compute Optimizer**

- **AWS Elastic Beanstalk**

  (Correct)

Explanation
AWS Elastic Beanstalk supports the deployment of web applications from Docker containers. With Docker containers, you can define your own runtime environment. You can choose your own platform, programming language, and any application dependencies (such as package managers or tools), that aren't supported by other

platforms. Docker containers are self-contained and include all the configuration information and software your web application requires to run.



By using Docker with Elastic Beanstalk, you have an infrastructure that automatically handles the details of capacity provisioning, load balancing, scaling, and application health monitoring. You can manage your web application in an environment that supports the range of services that are integrated with Elastic Beanstalk, including but not limited to VPC, RDS, and IAM.

Hence, the correct answer is: **AWS Elastic Beanstalk.**

Amazon Elastic Container Service (Amazon ECS) is incorrect. Although it also provides Service Auto Scaling, Service Load Balancing, and Monitoring with CloudWatch, these features are not ***automatically*** enabled by default unlike with Elastic Beanstalk. Take note that the scenario requires a service that will ***automatically*** *handle all the tasks such as balancing load, auto-scaling, monitoring, and placing your containers across your cluster.* You will have to manually configure these things if you wish to use ECS. With Elastic Beanstalk, you can manage your web application in an environment that supports the range of services easier.

AWS CloudFormation is incorrect. While you can deploy the infrastructure for your application thru CloudFormation templates, you will be the one responsible for connecting the AWS resources needed to build your application environment. With ElasticBeanstalk, all you have to do is upload your code; ElasticBeanstalk will automatically set up the environment for your application.

AWS Compute Optimizer is incorrect. Compute Optimizer simply analyzes your workload and recommends the optimal AWS resources needed to improve performance and reduce costs.

Reference:

https://docs.aws.amazon.com/elasticbeanstalk/latest/dg/create_deploy_docker.html

Check out this AWS Elastic Beanstalk Cheat Sheet:

https://tutorialsdojo.com/aws-elastic-beanstalk/

AWS Elastic Beanstalk Overview:

https://youtu.be/rx7e7Fej1Oo

Elastic Beanstalk vs CloudFormation vs OpsWorks vs CodeDeploy:

https://tutorialsdojo.com/elastic-beanstalk-vs-cloudformation-vs-opsworks-vs-codedeploy/

Comparison of AWS Services Cheat Sheets:

https://tutorialsdojo.com/comparison-of-aws-services/

Question 54: **Correct**
A tech company is running two production web servers hosted on Reserved EC2 instances with EBS-backed root volumes. These instances have a consistent CPU load of 90%. Traffic is being distributed to these instances by an Elastic Load Balancer. In addition, they also have Multi-AZ RDS MySQL databases for their production, test, and development environments.

What recommendation would you make to reduce cost in this AWS environment without affecting availability and performance of mission-critical systems? Choose the best answer.

- **Consider using On-demand instances instead of Reserved EC2 instances**

- **Consider removing the Elastic Load Balancer**
- **Consider using Spot instances instead of reserved EC2 instances**
- **Consider not using a Multi-AZ RDS deployment for the development and test database**

  (Correct)

Explanation

One thing that you should notice here is that the company is using Multi-AZ databases in all of their environments, including their development and test environment. This is costly and unnecessary as these two environments are not critical. It is better to use Multi-AZ for production environments to reduce costs, which is why the option that says: **Consider not using a Multi-AZ RDS deployment for the development and test database** is the correct answer.

The option that says: **Consider using On-demand instances instead of Reserved EC2 instances** is incorrect because selecting Reserved instances is cheaper than On-demand instances for long term usage due to the discounts offered when purchasing reserved instances.

The option that says: **Consider using Spot instances instead of reserved EC2 instances** is incorrect because the web servers are running in a production environment. Never use Spot instances for production level web servers unless you are sure that they are not that critical in your system. This is because your spot instances can be terminated once the maximum price goes over the maximum amount that you specified.

The option that says: **Consider removing the Elastic Load Balancer** is incorrect because the Elastic Load Balancer is crucial in maintaining the elasticity and reliability of your system.

References:

https://aws.amazon.com/rds/details/multi-az/

https://aws.amazon.com/pricing/cost-optimization/

Amazon RDS Overview:

https://www.youtube.com/watch?v=aZmpLl8K1UU

Check out this Amazon RDS Cheat Sheet:

https://tutorialsdojo.com/amazon-relational-database-service-amazon-rds/

Question 55: **Correct**
An online stock trading system is hosted in AWS and uses an Auto Scaling group of EC2 instances, an RDS database, and an Amazon ElastiCache for Redis. You need to improve the data security of your in-memory data store by requiring the user to enter a password before they are granted permission to execute Redis commands.

Which of the following should you do to meet the above requirement?

- **Create a new Redis replication group and set the `AtRestEncryptionEnabled` parameter to `true`.**
- **Do nothing. This feature is already enabled by default.**
- **None of the above.**
- **Authenticate the users using Redis AUTH by creating a new Redis Cluster with both the `--transit-encryption-enabled` and `--auth-token` parameters enabled.**

  (Correct)
-

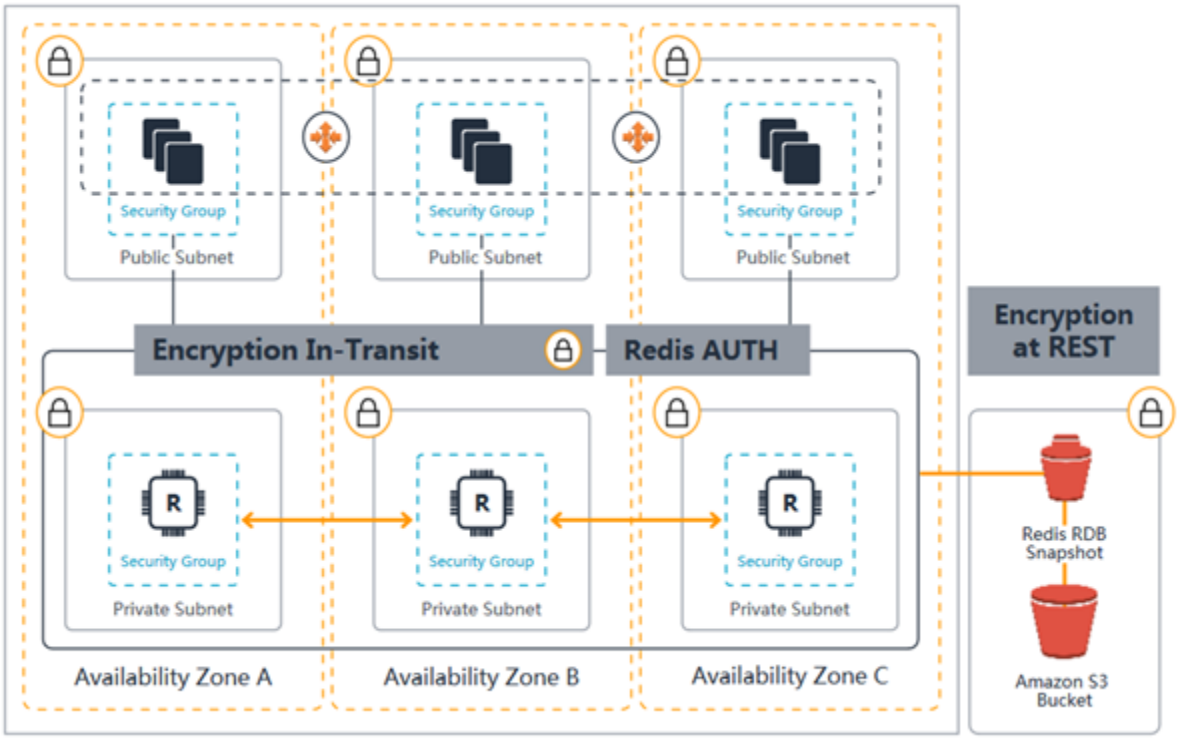
Enable the in-transit encryption for Redis replication groups.

Explanation

Using Redis `AUTH` command can improve data security by requiring the user to enter a password before they are granted permission to execute Redis commands on a password-protected Redis server.

Hence, the correct answer is to **authenticate the users using Redis AUTH by creating a new Redis Cluster with both the `--transit-encryption-enabled` and `--auth-token` parameters enabled**.

To require that users enter a password on a password-protected Redis server, include the parameter `--auth-token` with the correct password when you create your replication group or cluster and on all subsequent commands to the replication group or cluster.



The option that says: **Enabling the in-transit encryption for Redis replication groups** is incorrect. Although in-transit encryption is part of the solution, it is missing the most important thing which is the Redis AUTH option.

The option that says: **Creating a new Redis replication group and setting the `AtRestEncryptionEnabled` parameter to `true`** is incorrect because the Redis At-Rest Encryption feature only secures the data inside the in-memory data store. You have to use Redis AUTH option instead.

The option that says: **Do nothing. This feature is already enabled by default** is incorrect because the Redis AUTH option is disabled by default.

References:

https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/auth.html

https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/encryption.html

Check out this Amazon Elasticache cheat sheet:

https://tutorialsdojo.com/amazon-elasticache/

Redis Append-Only Files vs. Redis Replication:

https://tutorialsdojo.com/redis-append-only-files-vs-redis-replication/

Comparison of AWS Services Cheat Sheets:

https://tutorialsdojo.com/comparison-of-aws-services/

Question 56: **Correct**
A company plans to design an application that can handle batch processing of large amounts of financial data. The Solutions Architect is tasked to create two Amazon S3 buckets to store the input and output data. The application will transfer the data between multiple EC2 instances over the network to complete the data processing.

Which of the following options would reduce the data transfer costs?

- **Deploy the Amazon EC2 instances in the same Availability Zone.**

  (Correct)

-

Deploy the Amazon EC2 instances in the same AWS Region.

-

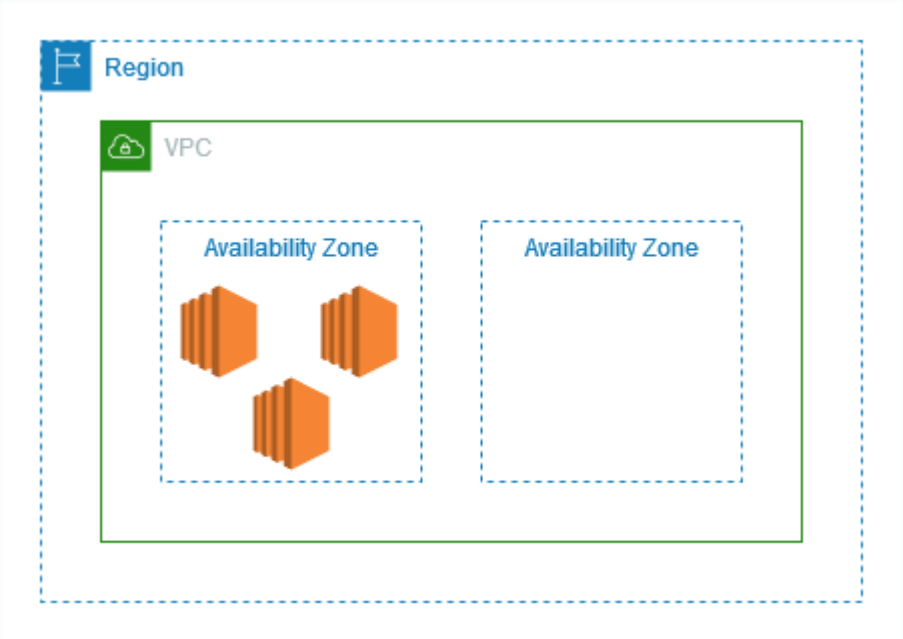
Deploy the Amazon EC2 instances in private subnets in different Availability Zones.

-

Deploy the Amazon EC2 instances behind an Application Load Balancer.

Explanation

Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the Amazon Web Services (AWS) Cloud. Using Amazon EC2 eliminates your need to invest in hardware up front, so you can develop and deploy applications faster. You can use Amazon EC2 to launch as many or as few virtual servers as you need, configure security and networking, and manage storage. Amazon EC2 enables you to scale up or down to handle changes in requirements or spikes in popularity, reducing your need to forecast traffic.



In this scenario, you should deploy all the EC2 instances in the same Availability Zone. If you recall, data transferred between Amazon EC2, Amazon RDS, Amazon Redshift, Amazon ElastiCache instances, and Elastic Network Interfaces in the same Availability Zone is free. Instead of using the public network to transfer the data, you can use the private network to reduce the overall data transfer costs.

Hence, the correct answer is: **Deploy the Amazon EC2 instances in the same Availability Zone.**

The option that says: **Deploy the Amazon EC2 instances in the same AWS Region** is incorrect because even if the instances are deployed in the same Region, they could still be charged with inter-Availability Zone data transfers if the instances are distributed across different availability zones. You must deploy the instances in the same Availability Zone to avoid the data transfer costs.

The option that says: **Deploy the Amazon EC2 instances behind an Application Load Balancer** is incorrect because this approach won't reduce the overall data transfer costs. An Application Load Balancer is primarily used to distribute the incoming traffic to underlying EC2 instances.

The option that says: **Deploy the Amazon EC2 instances in private subnets in different Availability Zones** is incorrect. Although the data transfer between instances in private subnets is free, there will be an issue with retrieving the data in Amazon S3. Remember that you won't be able to connect to your Amazon S3 bucket if you are using a private subnet unless you have a VPC Endpoint.

References:

https://aws.amazon.com/ec2/pricing/on-demand/

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

https://aws.amazon.com/blogs/mt/using-aws-cost-explorer-to-analyze-data-transfer-costs/

Amazon EC2 Overview:

https://www.youtube.com/watch?v=7VsGIHT_jQE

Check out this Amazon EC2 Cheat Sheet:

https://tutorialsdojo.com/amazon-elastic-compute-cloud-amazon-ec2/

Question 57: **Incorrect**
A Solutions Architect needs to launch a web application that will be served globally using Amazon CloudFront. The application is hosted in an Amazon EC2 instance which will be configured as the origin server to process and serve dynamic content to its customers.

Which of the following options provides high availability for the application?

- **Use Amazon S3 to serve the dynamic content of your web application and configure the S3 bucket to be part of an origin group.**

- **Launch an Auto Scaling group of EC2 instances and configure it to be part of an origin group.**

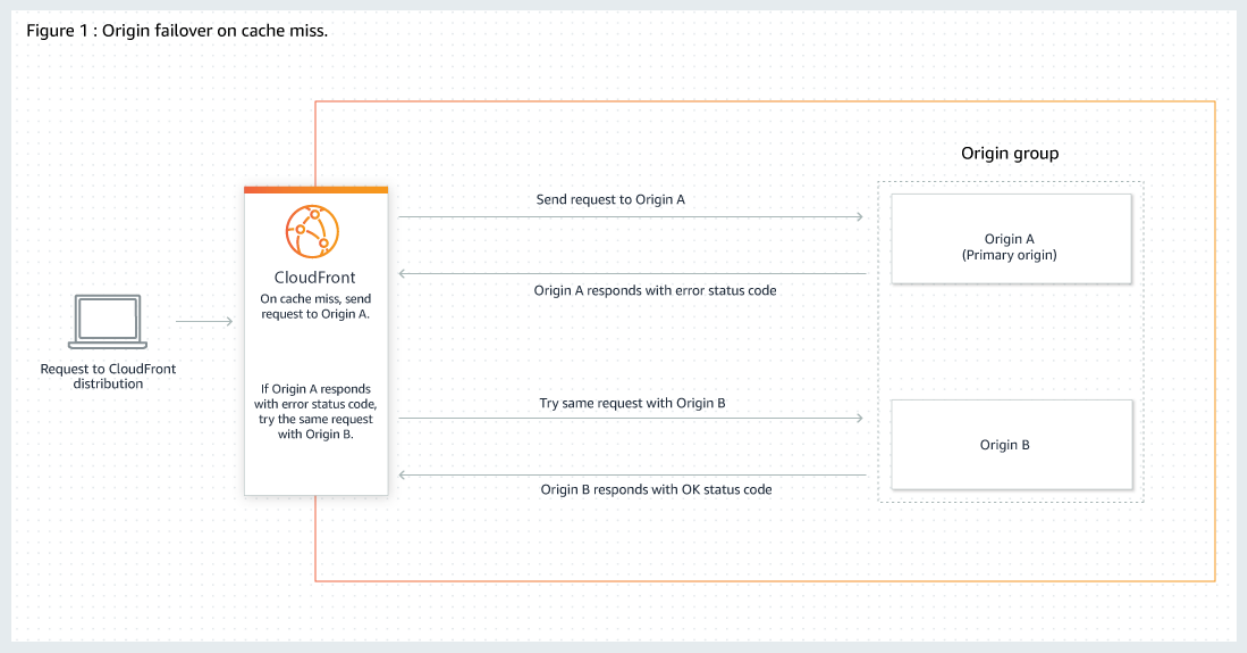
- **Provision two EC2 instances deployed in different Availability Zones and configure them to be part of an origin group.**

  (Correct)

- **Use Lambda@Edge to improve the performance of your web application and ensure high availability. Set the Lambda@Edge functions to be part of an origin group.**

  (Incorrect)

Explanation
An origin is a location where content is stored and from which CloudFront gets content to serve to viewers. **Amazon CloudFront** is a service that speeds up the distribution of your static and dynamic web content, such as .html, .css, .js, and image files, to your users. CloudFront delivers your content through a worldwide network of data centers called edge locations. When a user requests content that you're serving with CloudFront, the user is routed to the edge location that provides the lowest latency (time delay) so that content is delivered with the best possible performance.



You can also set up CloudFront with origin failover for scenarios that require high availability. An origin group may contain two origins: a primary and a secondary. If the primary origin is unavailable or returns specific HTTP response status codes that indicate a failure, CloudFront automatically switches to the secondary origin. To set up origin failover, you must have a distribution with at least two origins.

The scenario uses an EC2 instance as an origin. Take note that we can also use an EC2 instance or a custom origin in configuring CloudFront. To achieve high availability in an EC2 instance, we need to deploy the instances in two or more Availability Zones. You also need to configure the instances to be part of the origin group to ensure that the application is highly available.

Hence, the correct answer is: **Provision two EC2 instances deployed in different Availability Zones and configure them to be part of an origin group.**

The option that says: **Use Amazon S3 to serve the dynamic content of your web application and configure the S3 bucket to be part of an origin group** is incorrect because Amazon S3 can only serve static content. If you need to host dynamic content, you have to use an Amazon EC2 instance instead.

The option that says: **Launch an Auto Scaling group of EC2 instances and configure it to be part of an origin group** is incorrect because you must have at least two origins to set up an origin failover in CloudFront. In addition, you can't directly use a single Auto Scaling group as an origin.

The option that says: **Use Lambda@Edge to improve the performance of your web application and ensure high availability. Set the Lambda@Edge functions to be part of**

an origin group is incorrect because Lambda@Edge is primarily used for serverless edge computing. You can't set Lambda@Edge functions as part of your origin group in CloudFront.

References:

https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/high_availability_origin_failover.html

https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/Introduction.html

https://aws.amazon.com/cloudfront/faqs/

Check out this Amazon CloudFront Cheat Sheet:

https://tutorialsdojo.com/amazon-cloudfront/

Question 58: **Correct**
A Solutions Architect is managing a three-tier web application that processes credit card payments and online transactions. Static web pages are used on the front-end tier while the application tier contains a single Amazon EC2 instance that handles long-running processes. The data is stored in a MySQL database. The Solutions Architect is instructed to decouple the tiers to create a highly available application.

Which of the following options can satisfy the given requirement?

- **Move all the static assets, web pages, and the backend application to a larger instance. Use Auto Scaling in Amazon EC2 instance. Migrate the database to Amazon Aurora.**

- **Move all the static assets and web pages to Amazon S3. Re-host the application to Amazon Elastic Container Service (Amazon ECS) containers and enable Service Auto Scaling. Migrate the database to Amazon RDS with Multi-AZ deployments configuration.**

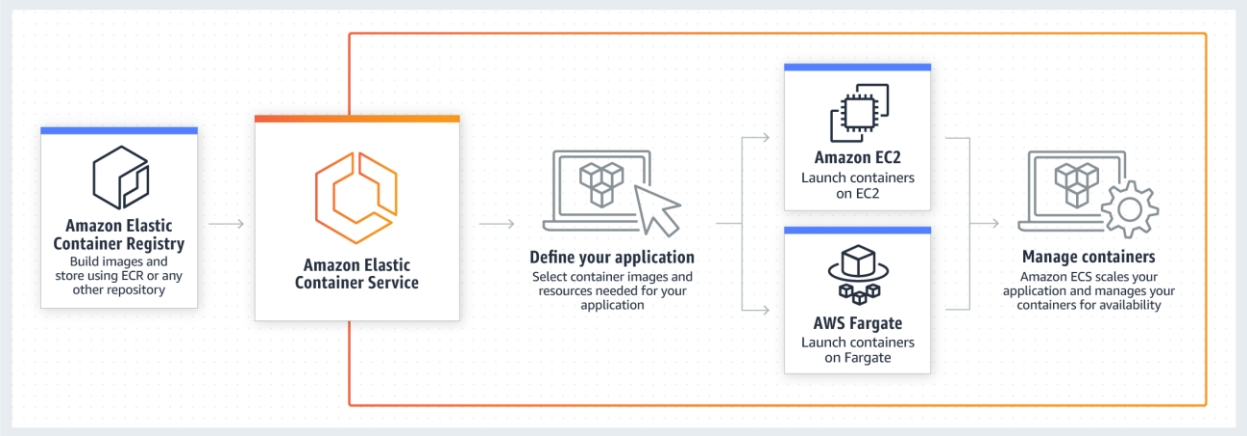
  (Correct)

- **Move all the static assets and web pages to Amazon CloudFront. Use Auto Scaling in Amazon EC2 instance. Migrate the database to Amazon RDS with Multi-AZ deployments configuration.**

- **Move all the static assets to Amazon S3. Set concurrency limit in AWS Lambda to move the application to a serverless architecture. Migrate the database to Amazon DynamoDB.**

Explanation

Amazon Elastic Container Service (ECS) is a highly scalable, high performance container management service that supports Docker containers and allows you to easily run applications on a managed cluster of Amazon EC2 instances. Amazon ECS makes it easy to use containers as a building block for your applications by eliminating the need for you to install, operate, and scale your own cluster management infrastructure. Amazon ECS lets you schedule long-running applications, services, and batch processes using Docker containers. Amazon ECS maintains application availability and allows you to scale your containers up or down to meet your application's capacity requirements.



The requirement in the scenario is to decouple the services to achieve a highly available architecture. To accomplish this requirement, you must move the existing setup to each AWS services. For static assets, you should use Amazon S3. You can use Amazon ECS for your web application and then migrate the database to Amazon RDS with Multi-AZ deployment. Decoupling your app with application integration services allows them to remain interoperable, but if one service has a failure or spike in workload, it won't affect the rest of them.

Hence, the correct answer is: **Move all the static assets and web pages to Amazon S3. Re-host the application to Amazon Elastic Container Service (Amazon ECS) containers**

and enable Service Auto Scaling. Migrate the database to Amazon RDS with Multi-AZ deployments configuration.

The option that says: **Move all the static assets to Amazon S3. Set concurrency limit in AWS Lambda to move the application to a serverless architecture. Migrate the database to Amazon DynamoDB** is incorrect because Lambda functions can't process long-running processes. Take note that a Lambda function has a maximum processing time of 15 minutes.

The option that says: **Move all the static assets, web pages, and the backend application to a larger instance. Use Auto Scaling in Amazon EC2 instance. Migrate the database to Amazon Aurora** is incorrect because static assets are more suitable and cost-effective to be stored in S3 instead of storing them in an EC2 instance.

The option that says: **Move all the static assets and web pages to Amazon CloudFront. Use Auto Scaling in Amazon EC2 instance. Migrate the database to Amazon RDS with Multi-AZ deployments configuration** is incorrect because you can't store data in Amazon CloudFront. Technically, you only store cache data in CloudFront, but you can't host applications or web pages using this service. You have to use Amazon S3 to host the static web pages and use CloudFront as the CDN.

References:

https://docs.aws.amazon.com/AmazonECS/latest/developerguide/service-auto-scaling.html

https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Concepts.MultiAZ.html

Check out this Amazon ECS Cheat Sheet:

https://tutorialsdojo.com/amazon-elastic-container-service-amazon-ecs/

Question 59: **Correct**
A commercial bank has designed its next-generation online banking platform to use a distributed system architecture. As their Software Architect, you have to ensure that their architecture is highly scalable, yet still cost-effective. Which of the following will provide the most suitable solution for this scenario?

- 

Launch an Auto-Scaling group of EC2 instances to host your application services and an SQS queue. Include an Auto Scaling trigger to watch the SQS queue size which will either scale in or scale out the number of EC2 instances based on the queue.

(Correct)

- **Launch multiple On-Demand EC2 instances to host your application services and an SQS queue which will act as a highly-scalable buffer that stores messages as they travel between distributed applications.**

- **Launch multiple EC2 instances behind an Application Load Balancer to host your application services and SNS which will act as a highly-scalable buffer that stores messages as they travel between distributed applications.**

- **Launch multiple EC2 instances behind an Application Load Balancer to host your application services, and SWF which will act as a highly-scalable buffer that stores messages as they travel between distributed applications.**

Explanation

There are three main parts in a distributed messaging system: the components of your distributed system which can be hosted on EC2 instance; your queue (distributed on Amazon SQS servers); and the messages in the queue.

To improve the scalability of your distributed system, you can add Auto Scaling group to your EC2 instances.

References:

https://docs.aws.amazon.com/autoscaling/ec2/userguide/as-using-sqs-queue.html

https://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/sqs-basic-architecture.html

Check out this AWS Auto Scaling Cheat Sheet:

https://tutorialsdojo.com/aws-auto-scaling/

Question 60: **Correct**
A Solutions Architect designed a real-time data analytics system based on Kinesis Data Stream and Lambda. A week after the system has been deployed, the users noticed that it performed slowly as the data rate increases. The Architect identified that the performance of the Kinesis Data Streams is causing this problem.

Which of the following should the Architect do to improve performance?

- **Improve the performance of the stream by decreasing the number of its shards using the `MergeShard` command.**

- **Replace the data stream with Amazon Kinesis Data Firehose instead.**

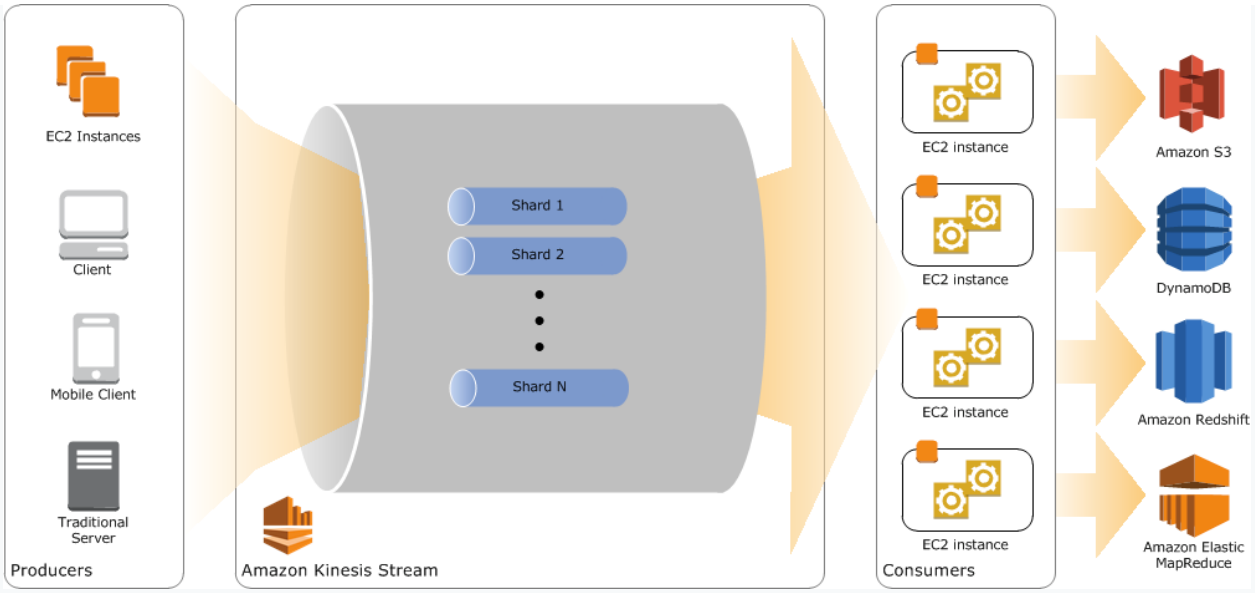
- **Increase the number of shards of the Kinesis stream by using the `UpdateShardCount` command.**

  (Correct)

- **Implement Step Scaling to the Kinesis Data Stream.**

Explanation
Amazon Kinesis Data Streams supports *resharding*, which lets you adjust the number of shards in your stream to adapt to changes in the rate of data flow through the stream. Resharding is considered an advanced operation.

There are two types of resharding operations: shard split and shard merge. In a shard split, you divide a single shard into two shards. In a shard merge, you combine two shards into a single shard. Resharding is always *pairwise* in the sense that you cannot split into more than two shards in a single operation, and you cannot merge more than two shards in a single operation. The shard or pair of shards that the resharding operation acts on are referred to as *parent* shards. The shard or pair of shards that result from the resharding operation are referred to as *child* shards.



Splitting increases the number of shards in your stream and therefore increases the data capacity of the stream. Because you are charged on a per-shard basis, splitting increases the cost of your stream. Similarly, merging reduces the number of shards in your stream and therefore decreases the data capacity—and cost—of the stream.

If your data rate increases, you can also increase the number of shards allocated to your stream to maintain the application performance. You can reshard your stream using the **UpdateShardCount** API. The throughput of an Amazon Kinesis data stream is designed to scale without limits via increasing the number of shards within a data stream. Hence, the correct answer is to **increase the number of shards of the Kinesis stream by using the** `UpdateShardCount` **command**.

Replacing the data stream with Amazon Kinesis Data Firehose instead is incorrect because the throughput of Kinesis Firehose is not exceptionally higher than Kinesis Data Streams. In fact, the throughput of an Amazon Kinesis data stream is designed to scale **without** limits via increasing the number of shards within a data stream.

Improving the performance of the stream by decreasing the number of its shards using the `MergeShard` **command** is incorrect because merging the shards will effectively decrease the performance of the stream rather than improve it.

Implementing Step Scaling to the Kinesis Data Stream is incorrect because there is no Step Scaling feature for Kinesis Data Streams. This is only applicable for EC2.

References:

https://aws.amazon.com/blogs/big-data/scale-your-amazon-kinesis-stream-capacity-with-updateshardcount/

https://aws.amazon.com/kinesis/data-streams/faqs/

https://docs.aws.amazon.com/streams/latest/dev/kinesis-using-sdk-java-resharding.html

Check out this Amazon Kinesis Cheat Sheet:

https://tutorialsdojo.com/amazon-kinesis/

Question 61: **Correct**
A company plans to implement a network monitoring system in AWS. The Solutions Architect launched an EC2 instance to host the monitoring system and used CloudWatch to monitor, store, and access the log files of the instance.

Which of the following provides an automated way to send log data to CloudWatch Logs from the Amazon EC2 instance?

- **CloudWatch Logs agent**

  (Correct)

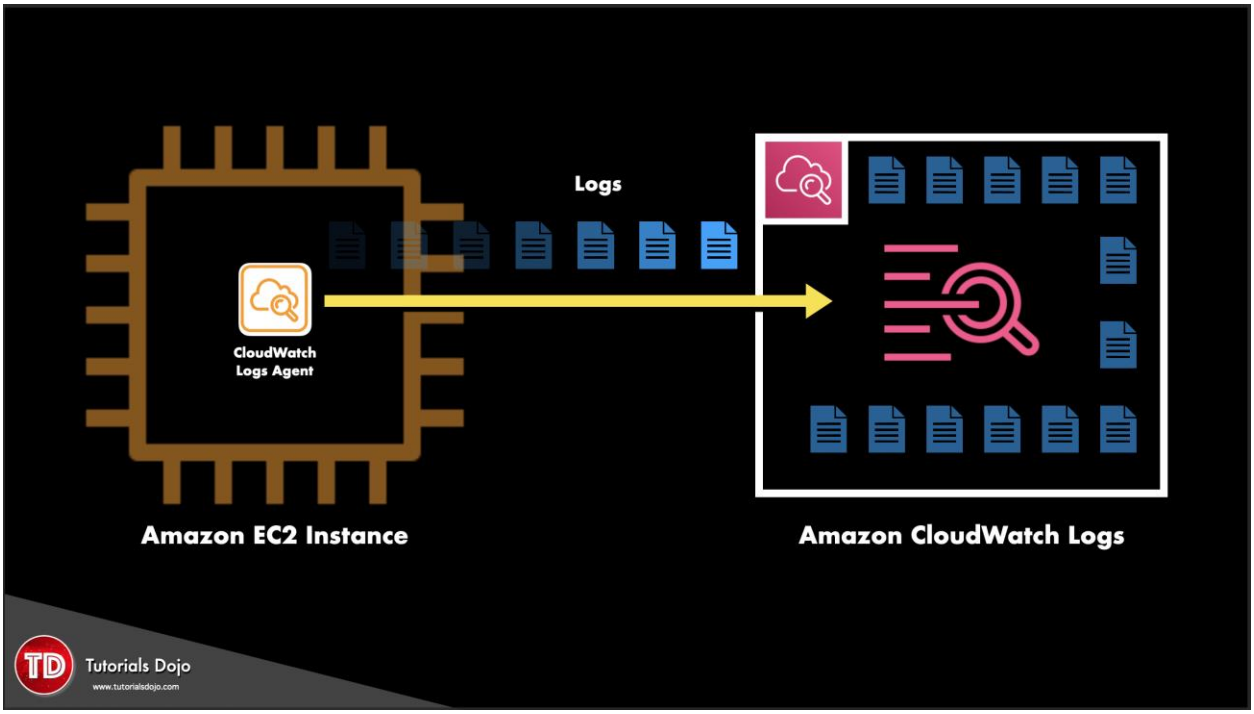
- **CloudTrail with log file validation**

- **CloudTrail Processing Library**

- **AWS Transfer for SFTP**

Explanation
CloudWatch Logs enables you to centralize the logs from all of your systems, applications, and AWS services that you use, in a single, highly scalable service. You can then easily view them, search them for specific error codes or patterns, filter them based on specific fields, or archive them securely for future analysis. CloudWatch Logs enables you to see all of your logs, regardless of their source, as a single and consistent

flow of events ordered by time, and you can query them and sort them based on other dimensions, group them by specific fields, create custom computations with a powerful query language, and visualize log data in dashboards.



The CloudWatch Logs agent is comprised of the following components:

- A plug-in to the AWS CLI that pushes log data to CloudWatch Logs.

- A script (daemon) that initiates the process to push data to CloudWatch Logs.

- A cron job that ensures that the daemon is always running.

CloudWatch Logs agent provides an automated way to send log data to CloudWatch Logs from Amazon EC2 instances hence, **CloudWatch Logs agent** is the correct answer.

CloudTrail with log file validation is incorrect as this is mainly used for tracking the API calls of your AWS resources and not for sending EC2 logs to CloudWatch.

AWS Transfer for SFTP is incorrect as this is only a fully managed SFTP service for Amazon S3 used for tracking the traffic coming into the VPC and not for EC2 instance monitoring. This service enables you to easily move your file transfer workloads that use the Secure Shell File Transfer Protocol (SFTP) to AWS without needing to modify your applications or manage any SFTP servers. This can't be used to send log data from your EC2 instance to Amazon CloudWatch.

CloudTrail Processing Library is incorrect because this is just a Java library that provides an easy way to process AWS CloudTrail logs. It cannot send your log data to CloudWatch Logs.

References:

https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/WhatIsCloudWatchLogs.html

https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/AgentReference.html

Check out this Amazon CloudWatch Cheat Sheet:

https://tutorialsdojo.com/amazon-cloudwatch/

Question 62: **Correct**
An airline company receives a lot of requests to book flights, update booking details, and flight check-ins. Since these requests flood the customer support teams, the management wants to build a self-service solution that can handle these requests without a human agent. This solution should be text-based wherein users can type their concerns in a chat box and an AI will analyze their intention, provide answers, or fulfill pre-defined actions automatically.

Which of the following options is the recommended solution for the above requirements?

- **Deploy a conversational chatbot using Amazon Rekognition. Define conversation flow for specific user intentions. Create AWS Lambda functions that can be invoked depending on user intentions.**
- **Work with an AWS Managed Service Provider (MSP) to deploy a conversational chatbot using Amazon Polly for natural-language processing (NLU). Integrate AWS Lambda functions as code hooks to perform actions based on user requests.**

Deploy a conversational chatbot using Amazon Lex. Define conversation flow for specific user intentions. Integrate AWS Lambda functions as code hooks to perform actions based on user requests.

(Correct)

-

Create a conversational chatbot using Amazon Comprehend for natural-language processing (NLU). Depending on the user's intent, invoke AWS Lambda functions that can perform the needed actions.

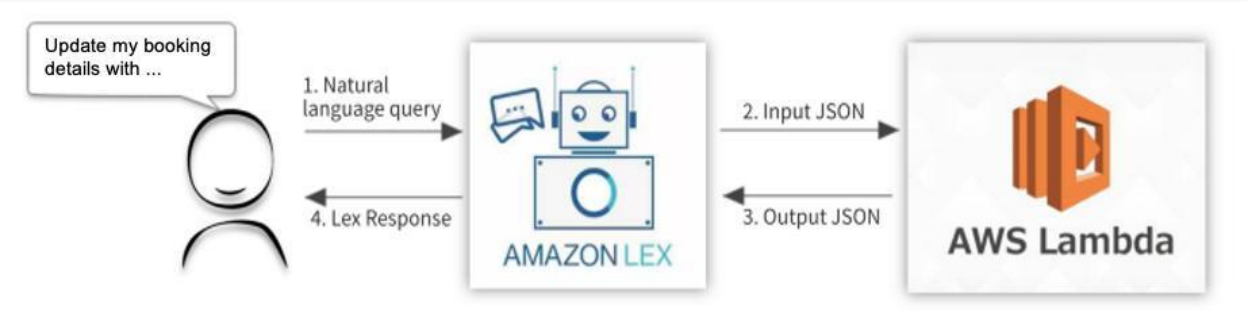
Explanation

Amazon Lex enables you to build applications using a speech or text interface powered by the same technology that powers Amazon Alexa. Amazon Lex provides the deep functionality and flexibility of natural language understanding (NLU) and automatic speech recognition (ASR), so you can build highly engaging user experiences with lifelike conversational interactions and create new categories of products.

Amazon Lex enables any developer to build conversational chatbots quickly. With Amazon Lex, no deep learning expertise is necessary—to create a bot, you just specify the basic conversation flow in the Amazon Lex console. The console provides a graphical user interface that you use to build a production-ready bot for your application.

After you create a bot, you deploy it on one of the supported platforms or integrate it into your own application. When a user interacts with the bot, the client application sends requests to the bot using the Amazon Lex runtime API. For example, when a user says "I want to order pizza," your client sends this input to Amazon Lex using one of the runtime API operations. Users can provide input as speech or text.

You can also create **Lambda functions** and use them in an intent. Use these Lambda function code hooks to perform runtime activities such as initialization, validation of user input, and intent fulfillment.



Therefore, the correct answer is: **Deploy a conversational chatbot using Amazon Lex. Define conversation flow for specific user intentions. Integrate AWS Lambda functions**

as code hooks to perform actions based on user requests. With Amazon Lex, you can build conversational chatbots quickly. You can configure intents and actions depending on text inputted by users.

The option that says: **Create a conversational chatbot using Amazon Comprehend for natural-language processing (NLU). Depending on the user's intent, invoke AWS Lambda functions that can perform the needed actions** is incorrect. Amazon Comprehend is a natural language processing (NLP) service that uses machine learning to find insights and relationships in texts. It is not used to build chatbot applications.

The option that says: **Work with an AWS Managed Service Provider (MSP) to deploy a conversational chatbot using Amazon Polly for natural-language processing (NLU). Integrate AWS Lambda functions as code hooks to perform actions based on user requests** is incorrect. Amazon Polly converts text into lifelike speech. It is designed for text-to-speech applications, not for conversational chatbots. In addition, the AWS Managed Service Provider (MSP) program simply validates AWS Partners with a proven track record and experience, providing end-to-end AWS solutions to customers at any stage of the cloud journey. You don't need an MSP to deploy an Amazon Polly chatbot.

The option that says: **Deploy a conversational chatbot using Amazon Rekognition. Define conversation flow for specific user intentions. Create AWS Lambda functions that can be invoked depending on user intentions** is incorrect. Amazon Rekognition is used to identify objects, people, text, scenes, and activities in images and videos; not to build chatbot applications.

References:

https://docs.aws.amazon.com/lex/latest/dg/programming-model.html#prog-model-lambda

https://docs.aws.amazon.com/lex/latest/dg/howitworks-manage-prompts.html

https://docs.aws.amazon.com/lex/latest/dg/what-is.html

Check out these Amazon Lex and AWS Lambda Cheat Sheets:

https://tutorialsdojo.com/amazon-lex/

https://tutorialsdojo.com/aws-lambda/

Question 63: **Correct**

An online shopping platform has been deployed to AWS using Elastic Beanstalk. They simply uploaded their Node.js application, and Elastic Beanstalk automatically handles the details of capacity provisioning, load balancing, scaling, and application health monitoring. Since the entire deployment process is automated, the DevOps team is not sure where to get the application log files of their shopping platform.

In Elastic Beanstalk, where does it store the application files and server log files?

- **Application files are stored in S3. The server log files can only be stored in the attached EBS volumes of the EC2 instances, which were launched by AWS Elastic Beanstalk.**

- **Application files are stored in S3. The server log files can also optionally be stored in S3 or in CloudWatch Logs.**

  (Correct)

- **Application files are stored in S3. The server log files can be optionally stored in CloudTrail or in CloudWatch Logs.**

- **Application files are stored in S3. The server log files can be stored directly in Glacier or in CloudWatch Logs.**

Explanation

AWS Elastic Beanstalk stores your application files and optionally, server log files in Amazon S3. If you are using the AWS Management Console, the AWS Toolkit for Visual Studio, or AWS Toolkit for Eclipse, an Amazon S3 bucket will be created in your account and the files you upload will be automatically copied from your local client to Amazon S3. Optionally, you may configure Elastic Beanstalk to copy your server log files every hour to Amazon S3. You do this by editing the environment configuration settings.

Thus, the correct answer is the option that says: **Application files are stored in S3. The server log files can also optionally be stored in S3 or in CloudWatch Logs.**

With **CloudWatch Logs**, you can monitor and archive your Elastic Beanstalk application, system, and custom log files from Amazon EC2 instances of your environments. You can also configure alarms that make it easier for you to react to specific log stream events that your metric filters extract. The CloudWatch Logs agent installed on each

Amazon EC2 instance in your environment publishes metric data points to the CloudWatch service for each log group you configure. Each log group applies its own filter patterns to determine what log stream events to send to CloudWatch as data points. Log streams that belong to the same log group share the same retention, monitoring, and access control settings. You can configure Elastic Beanstalk to automatically stream logs to the CloudWatch service.

The option that says: **Application files are stored in S3. The server log files can only be stored in the attached EBS volumes of the EC2 instances, which were launched by AWS Elastic Beanstalk** is incorrect because the server log files can also be stored in either S3 or CloudWatch Logs, and not only on the EBS volumes of the EC2 instances which are launched by AWS Elastic Beanstalk.

The option that says: **Application files are stored in S3. The server log files can be stored directly in Glacier or in CloudWatch Logs** is incorrect because the server log files can optionally be stored in either S3 or CloudWatch Logs, but not directly to Glacier. You can create a lifecycle policy to the S3 bucket to store the server logs and archive it in Glacier, but there is no direct way of storing the server logs to Glacier using Elastic Beanstalk unless you do it programmatically.

The option that says: **Application files are stored in S3. The server log files can be optionally stored in CloudTrail or in CloudWatch Logs** is incorrect because the server log files can optionally be stored in either S3 or CloudWatch Logs, but not directly to CloudTrail as this service is primarily used for auditing API calls.

Reference:

https://aws.amazon.com/elasticbeanstalk/faqs/

AWS Elastic Beanstalk Overview:

https://www.youtube.com/watch?v=rx7e7Fej1Oo

Check out this AWS Elastic Beanstalk Cheat Sheet:

https://tutorialsdojo.com/aws-elastic-beanstalk/

Question 64: **Incorrect**

There is a technical requirement by a financial firm that does online credit card processing to have a secure application environment on AWS. They are trying to decide on whether to use KMS or CloudHSM.

Which of the following statements is right when it comes to CloudHSM and KMS?

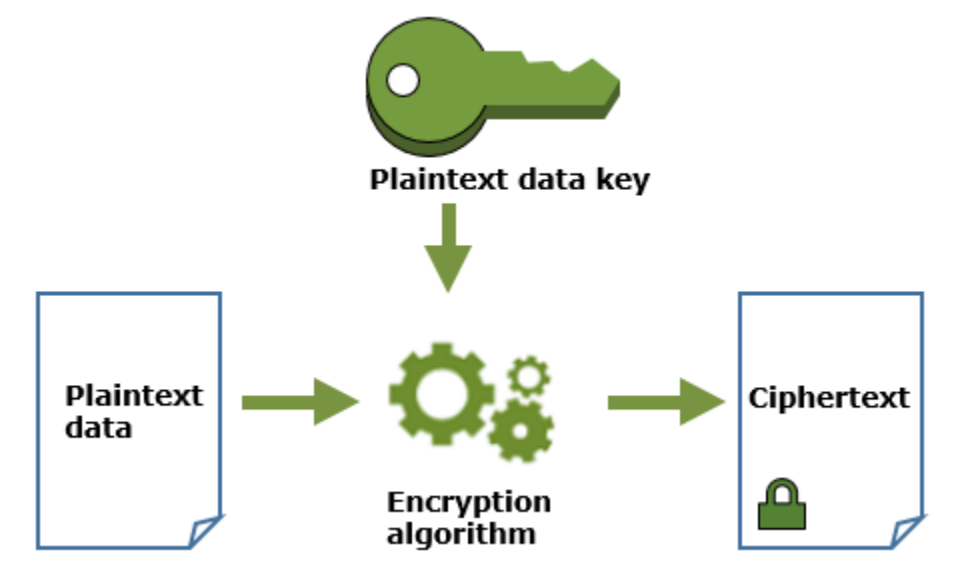
- **No major difference. They both do the same thing.**
- **If you want a managed service for creating and controlling your encryption keys but don't want or need to operate your own HSM, consider using AWS CloudHSM.**

  (Incorrect)
- **You should consider using AWS CloudHSM over AWS KMS if you require your keys stored in dedicated, third-party validated hardware security modules under your exclusive control.**

  (Correct)
- **AWS CloudHSM should always be used for any payment transactions.**

Explanation

AWS Key Management Service (AWS KMS) is a managed service that makes it easy for you to create and control the encryption keys used to encrypt your data. The master keys that you create in AWS KMS are protected by FIPS 140-2 validated cryptographic modules. AWS KMS is integrated with most other AWS services that encrypt your data with encryption keys that you manage. AWS KMS is also integrated with AWS CloudTrail to provide encryption key usage logs to help meet your auditing, regulatory and compliance needs.



By using AWS KMS, you gain more control over access to data you encrypt. You can use the key management and cryptographic features directly in your applications or through AWS services that are integrated with AWS KMS. Whether you are writing applications for AWS or using AWS services, AWS KMS enables you to maintain control over who can use your customer master keys and gain access to your encrypted data. AWS KMS is integrated with AWS CloudTrail, a service that delivers log files to an Amazon S3 bucket that you designate. By using CloudTrail, you can monitor and investigate how and when your master keys have been used and by whom.

If you want a managed service for creating and controlling your encryption keys, but you don't want or need to operate your own HSM, consider using AWS Key Management Service.

Hence, the correct answer is: **You should consider using AWS CloudHSM over AWS KMS if you require your keys stored in dedicated, third-party validated hardware security modules under your exclusive control.**

The option that says: **No major difference. They both do the same thing** is incorrect because KMS and CloudHSM are two different services. If you want a managed service for creating and controlling your encryption keys without operating your own HSM, you have to consider using AWS Key Management Service.

The option that says: **If you want a managed service for creating and controlling your encryption keys, but you don't want or need to operate your own HSM, consider using AWS CloudHSM** is incorrect because you have to consider using AWS KMS if you want a managed service for creating and controlling your encryption keys, without operating your own HSM.

The option that says: **AWS CloudHSM should always be used for any payment transactions** is incorrect because this is not always the case. AWS CloudHSM is a cloud-based hardware security module (HSM) that enables you to easily generate and use your own encryption keys on the AWS Cloud.

References:

https://docs.aws.amazon.com/kms/latest/developerguide/overview.html

https://docs.aws.amazon.com/kms/latest/developerguide/concepts.html#data-keys

https://docs.aws.amazon.com/cloudhsm/latest/userguide/introduction.html

Check out this AWS Key Management Service (KMS) Cheat Sheet:

https://tutorialsdojo.com/aws-key-management-service-aws-kms/

Question 65: **Incorrect**
An intelligence agency is currently hosting a learning and training portal in AWS. Your manager instructed you to launch a large EC2 instance with an attached EBS Volume and enable Enhanced Networking. What are the valid case scenarios in using Enhanced Networking? (Select TWO.)

- ☑ **When you need a higher packet per second (PPS) performance**

  (Correct)

- ☐ **When you need high latency networking**

- ☐ **When you need a low packet-per-second performance**

- ☐ **When you need a consistently lower inter-instance latencies**

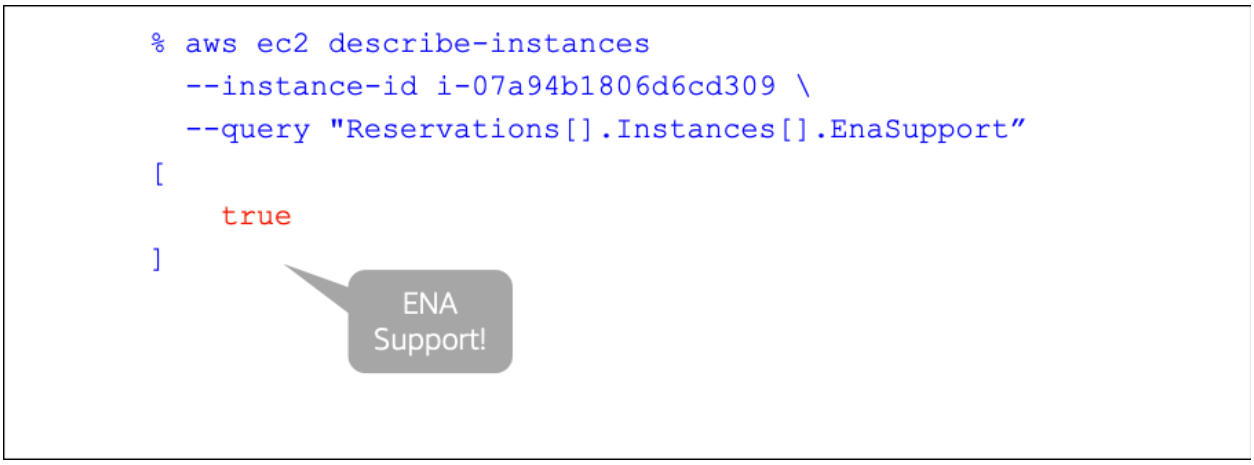
(Correct)

- ☑

  When you need a dedicated connection to your on-premises data center

  (Incorrect)

Explanation
Enhanced networking uses single root I/O virtualization (SR-IOV) to provide high-performance networking capabilities on supported instance types. SR-IOV is a method of device virtualization that provides higher I/O performance and lower CPU utilization when compared to traditional virtualized network interfaces. Enhanced networking provides higher bandwidth, higher packet per second (PPS) performance, and consistently lower inter-instance latencies. There is no additional charge for using enhanced networking.



The option that says: **When you need a low packet-per-second performance** is incorrect because you want to increase packet-per-second performance and not lower it when you enable enhanced networking.

The option that says: **When you need high latency networking** is incorrect because higher latencies mean a slower network, which is the opposite of what you want to happen when you enable enhanced networking.

The option that says: **When you need a dedicated connection to your on-premises data center** is incorrect because enabling enhanced networking does not provide a dedicated connection to your on-premises data center. Use AWS Direct Connect or enable VPN tunneling instead for this purpose.

Reference:

http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/enhanced-networking.html

Check out this Amazon EC2 Cheat Sheet:

https://tutorialsdojo.com/amazon-elastic-compute-cloud-amazon-ec2/