Question 1Skipped

The engineering team at a social media company has noticed that while some of the images stored in Amazon S3 are frequently accessed, others sit idle for a considerable span of time.

As a solutions architect, what is your recommendation to build the MOST cost-effective solution?

Store the images using the Amazon S3 Standard-IA storage class

Create a data monitoring application on an Amazon EC2 instance in the same region as the bucket storing the images. The application is triggered daily via Amazon CloudWatch and it changes the storage class of infrequently accessed objects to Amazon S3 Standard-IA and the frequently accessed objects are migrated to Amazon S3 Standard class

Correct answer

Store the images using the Amazon S3 Intelligent-Tiering storage class

Create a data monitoring application on an Amazon EC2 instance in the same region as the bucket storing the images. The application is triggered daily via Amazon CloudWatch and it changes the storage class of infrequently accessed objects to Amazon S3 One Zone-IA and the frequently accessed objects are migrated to Amazon S3 Standard class

Overall explanation

Correct option:

Store the images using the Amazon S3 Intelligent-Tiering storage class

The Amazon S3 Intelligent-Tiering storage class is designed to optimize costs by automatically moving data to the most cost-effective access tier, without performance impact or operational overhead. It works by storing objects in two access tiers: one tier that is optimized for frequent access and another lower-cost tier that is optimized for infrequent access.

For a small monthly monitoring and automation fee per object, Amazon S3 monitors access patterns of the objects in S3 Intelligent-Tiering and moves the ones that have not been accessed for 30 consecutive days to the infrequent access tier. If an object in the infrequent access tier is accessed, it is automatically moved back to the frequent access tier. Therefore using the Amazon S3 Intelligent-Tiering storage class is the correct solution for the given problem statement.

Amazon S3 Storage Classes Overview:

	S3 Standard	S3 Intelligent-Tiering*	S3 Standard-IA	S3 One Zone-IA†	S3 Glacier	S3 Glacier Deep Archive
Designed for durability	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)
Designed for availability	99.99%	99.9%	99.9%	99.5%	99.99%	99.99%
Availability SLA	99.9%	99%	99%	99%	99.9%	99.9%
Availability Zones	≥3	≥3	≥3	1	≥3	≥3
Minimum capacity charge per object	N/A	N/A	128KB	128KB	40KB	40KB
Minimum storage duration charge	N/A	30 days	30 days	30 days	90 days	180 days
Retrieval fee	N/A	N/A	per GB retrieved	per GB retrieved	per GB retrieved	per GB retrieved
First byte latency	milliseconds	milliseconds	milliseconds	milliseconds	select minutes or hours	select hours
Storage type	Object	Object	Object	Object	Object	Object
Lifecycle transitions	Yes	Yes	Yes	Yes	Yes	Yes

Incorrect options:

Store the images using the Amazon S3 Standard-IA storage class

Amazon S3 Standard-IA is for data that is accessed less frequently but requires rapid access when needed. Amazon S3 Standard-IA offers high durability, high throughput, and low latency of Amazon S3 Standard, with a low per GB storage price and per GB retrieval fee. This combination of low cost and high performance makes Amazon S3 Standard-IA ideal for long-term storage, backups, and as a data store for disaster recovery files. The minimum storage duration charge is 30 days. As some of the objects are frequently accessed, the per GB retrieval fee for Amazon S3 Standard-IA can cause the costs to shoot up, hence this option is incorrect.

Create a data monitoring application on an Amazon EC2 instance in the same region as the bucket storing the images. The application is triggered daily via Amazon CloudWatch and it changes the storage class of infrequently accessed objects to Amazon S3 One Zone-IA and the frequently accessed objects are migrated to Amazon S3 Standard class

Create a data monitoring application on an Amazon EC2 instance in the same region as the bucket storing the images. The application is triggered daily via Amazon CloudWatch and it changes the storage class of infrequently accessed objects to Amazon S3 Standard-IA and the frequently accessed objects are migrated to Amazon S3 Standard class

Creating a data monitoring application on an Amazon EC2 instance for managing the desired Amazon S3 storage class entails significant development cost as well as infrastructure maintenance effort. The Amazon S3 Intelligent-Tiering storage class does the job in a cost-effective way. Therefore both these options are incorrect.

Reference:

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

Domain

Design Cost-Optimized Architectures

Question 2Skipped

A social media application lets users upload photos and perform image editing operations. The application offers two classes of service: pro and lite. The product team wants the photos submitted by pro users to be processed before those submitted by lite users. Photos are uploaded to Amazon S3 and the job information is sent to Amazon SQS.

As a solutions architect, which of the following solutions would you recommend?

Create two Amazon SQS standard queues: one for pro and one for lite. Set the lite queue to use short polling and the pro queue to use long polling

Create two Amazon SQS FIFO queues: one for pro and one for lite. Set the lite queue to use short polling and the pro queue to use long polling

Create one Amazon SQS standard queue. Set the visibility timeout of the pro photos to zero. Set up Amazon EC2 instances to prioritize visibility settings so pro photos are processed first

Correct answer

Create two Amazon SQS standard queues: one for pro and one for lite. Set up Amazon EC2 instances to prioritize polling for the pro queue over the lite queue

Overall explanation

Correct option:

Create two Amazon SQS standard queues: one for pro and one for lite. Set up Amazon EC2 instances to prioritize polling for the pro queue over the lite queue

AWS recommends using separate queues to provide prioritization of work. Therefore, for the given use case, you need to create an Amazon SQS standard queue for processing pro users' photos and another Amazon SQS standard queue for processing lite users' photos. Then you can configure Amazon EC2 instances to prioritize polling for the pro queue over the lite queue.

Using Amazon SQS with other AWS infrastructure web services

Amazon SQS message queuing can be used with other AWS Services such as Redshift, DynamoDB, RDS, EC2, ECS, Lambda, and S3, to make distributed applications more scalable and reliable. Below are some common design patterns:

- **Work Queues**: Decouple components of a distributed application that may not all process the same amount of work simultaneously.
- **Buffer and Batch Operations**: Add scalability and reliability to your architecture, and smooth out temporary volume spikes without losing messages or increasing latency.
- **Request Offloading**: Move slow operations off of interactive request paths by enqueing the request.
- **Fanout**: Combine SQS with Simple Notification Service (SNS) to send identical copies of a message to multiple queues in parallel.
- **Priority**: Use separate queues to provide prioritization of work.
- **Scalability**: Because message queues decouple your processes, it's easy to scale up the send or receive rate of messages - simply add another process.
- **Resiliency**: When part of your system fails, it doesn't need to take the entire system down. Message queues decouple components of your system, so if a process that is reading messages from the queue fails, messages can still be added to the queue to be processed when the system recovers.

via - https://aws.amazon.com/sqs/features/

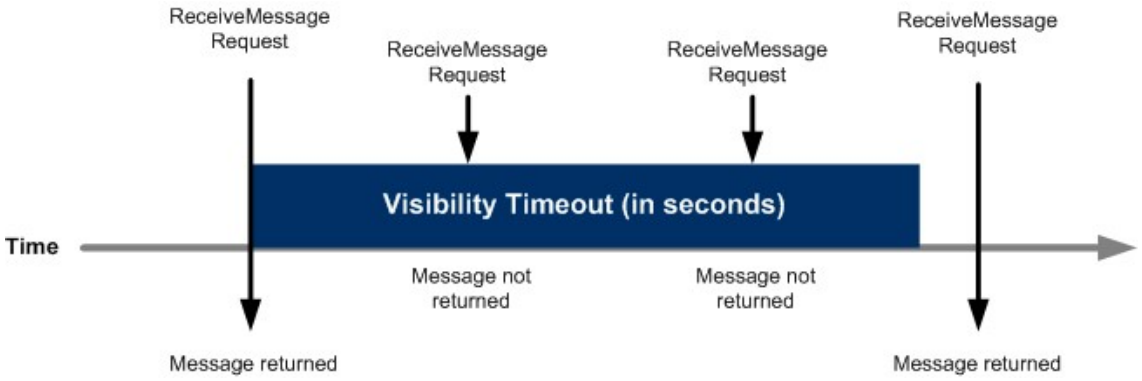
Incorrect options:

Create two Amazon SQS standard queues: one for pro and one for lite. Set the lite queue to use short polling and the pro queue to use long polling

Create two Amazon SQS FIFO queues: one for pro and one for lite. Set the lite queue to use short polling and the pro queue to use long polling

Amazon SQS long polling is a way to retrieve messages from your Amazon SQS queues. While the regular short polling returns immediately, even if the message queue being polled is empty, long-polling doesn't return a response until a message arrives in the message queue, or the long poll times out. Since long polling or short polling cannot impact the priority of processing for the two queues, so both these options are incorrect.

Create one Amazon SQS standard queue. Set the visibility timeout of the pro photos to zero. Set up Amazon EC2 instances to prioritize visibility settings so pro photos are processed first - To prevent other consumers from processing the message again, Amazon SQS sets a visibility timeout, a period of time during which Amazon SQS prevents other consumers from receiving and processing the message. The default visibility timeout for a message is 30 seconds. The minimum is 0 seconds. The maximum is 12 hours. Setting visibility timeout to zero can result in the same pro photo being processed by more than one consumer. This does not help in prioritizing the processing of pro photos over the lite photos.



via - https://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/sqs-visibility-timeout.html

References:

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

https://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/sqs-visibility-timeout.html

Domain

Design Resilient Architectures

Question 3Skipped

An application running on an Amazon EC2 instance needs to access a Amazon DynamoDB table in the same AWS account.

Which of the following solutions should a solutions architect configure for the necessary permissions?

Correct answer

Set up an IAM service role with the appropriate permissions to allow access to the Amazon DynamoDB table. Configure an instance profile to assign this IAM role to the Amazon EC2 instance

Set up an IAM user with the appropriate permissions to allow access to the Amazon DynamoDB table. Store the access credentials in the local storage and read them from within the application code directly

Set up an IAM user with the appropriate permissions to allow access to the Amazon DynamoDB table. Store the access credentials in an Amazon S3 bucket and read them from within the application code directly

Set up an IAM service role with the appropriate permissions to allow access to the Amazon DynamoDB table. Add the Amazon EC2 instance to the trust relationship policy document so that the instance can assume the role

Overall explanation

Correct option:

Set up an IAM service role with the appropriate permissions to allow access to the Amazon DynamoDB table. Configure an instance profile to assign this IAM role to the Amazon EC2 instance

A service role is an IAM role that a service assumes to perform actions on your behalf. Service roles provide access only within your account and cannot be used to grant access to services in other accounts. An IAM administrator can create, modify, and delete a service role from within IAM. When you create the service role, you define the trusted entity in the definition.

If you are going to use the role with Amazon EC2 or another AWS service that uses Amazon EC2, you must store the role in an instance profile. An instance profile is a container for a role that can be attached to an Amazon EC2 instance when launched. An instance profile can contain only one role, and that limit cannot be increased. If you create the role using the AWS Management Console, the instance profile is created for you with the same name as the role.

Incorrect options:

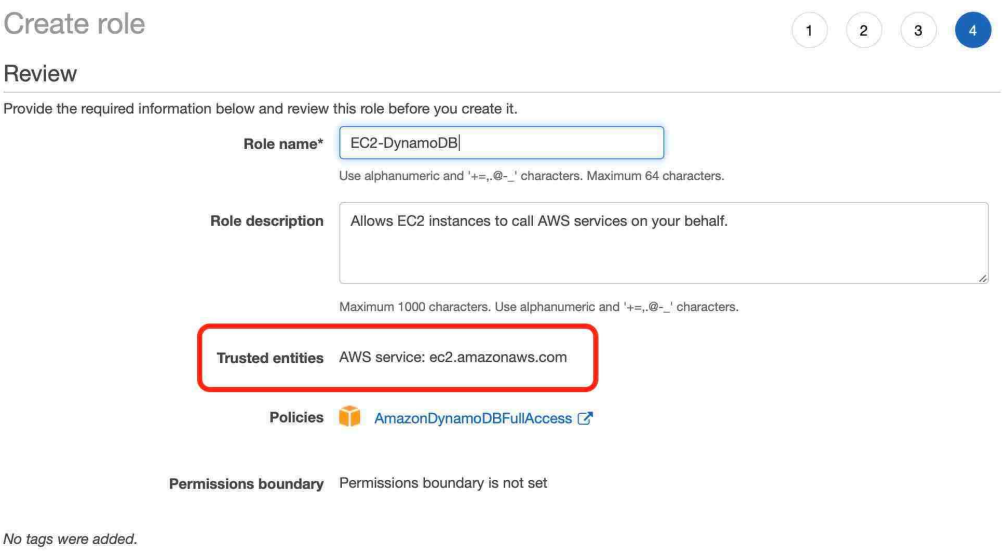
Set up an IAM user with the appropriate permissions to allow access to the Amazon DynamoDB table. Store the access credentials in an Amazon S3 bucket and read them from within the application code directly

Set up an IAM user with the appropriate permissions to allow access to the Amazon DynamoDB table. Store the access credentials in the local storage and read them from within the application code directly

You should never store the IAM access credentials for a user in Amazon S3 or local storage or a database. It's a security bad practice. It is always recommended to use IAM roles to configure access to other AWS resources from Amazon EC2 instances. Therefore both these options are incorrect.

Set up an IAM service role with the appropriate permissions to allow access to the Amazon DynamoDB table. Add the Amazon EC2 instance to the trust relationship policy document so that the instance can assume the role - There is no need for this option because when you create an IAM service role for Amazon EC2, the role automatically has Amazon EC2 identified as a trusted entity. Therefore this option is not correct.

Configuring a Service Role:



References:

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_create_for-service.html

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

Domain

Design Secure Architectures

Question 4Skipped

A development team wants to ensure that all objects uploaded to an Amazon S3 bucket are encrypted?

Which of the following options represents the correct solution?

Configure the bucket policy to deny if the PutObject does not have an aws:SecureTransport header set to true

Configure the bucket policy to deny if the PutObject does not have an s3:x-amz-acl header set

Configure the bucket policy to deny if the PutObject does not have an s3:x-amz-acl header set to private

Correct answer

Configure the bucket policy to deny if the PutObject does not have an x-amz-server-side-encryption header set

Overall explanation

Correct option:

Configure the bucket policy to deny if the PutObject does not have an x-amz-server-side-encryption header set

Amazon S3 encrypts your data at the object level as it writes to disks in AWS data centers, and decrypts it for you when you access it. You can encrypt objects by using client-side encryption or server-side encryption. Client-side encryption occurs when an object is encrypted before you upload it to Amazon S3, and the keys are not managed by AWS. With server-side encryption, Amazon manages the keys in one of three ways:

1. Server-side encryption with customer-provided encryption keys (SSE-C).
2. SSE-S3.
3. SSE-KMS.

Server-side encryption is about data encryption at rest—that is, Amazon S3 encrypts your data at the object level as it writes it to disks in its data centers and decrypts it for you when you access it. As long as you authenticate your request and you have access permissions, there is no difference in the way you access encrypted or unencrypted objects.

To encrypt an object at the time of upload, you need to add a header called x-amz-server-side-encryption to the request to tell S3 to encrypt the object using SSE-C, SSE-S3, or SSE-KMS.

In order to enforce object encryption, create an Amazon S3 bucket policy that denies any S3 Put request that does not include the x-amz-server-side-encryption header. There are two possible values for the x-amz-server-side-encryption header: AES256, which tells S3 to use S3-managed keys, and aws:kms, which tells Amazon S3 to use AWS KMS–managed keys.

Incorrect options:

Configure the bucket policy to deny if the PutObject does not have an s3:x-amz-acl header set to private - The x-amz-acl header is used to specify an ACL in the PutObject request. Access permissions are defined using this header.

Configure the bucket policy to deny if the PutObject does not have an aws:SecureTransport header set to true - By default, Amazon S3 allows both HTTP and HTTPS requests. aws:SecureTransport key is used to check if the request is sent through HTTP or HTTPS. When this key is true, it means that the request is sent through HTTPS.

Configure the bucket policy to deny if the PutObject does not have an s3:x-amz-acl header set - As discussed above, the s3:x-amz-acl header is used to set permissions on the specified S3 bucket and has nothing to do with encryption.

References:

https://aws.amazon.com/blogs/security/how-to-prevent-uploads-of-unencrypted-objects-to-amazon-s3/

https://docs.aws.amazon.com/AmazonS3/latest/userguide/amazon-s3-policy-keys.html

Domain

Design Secure Architectures

Question 5Skipped

A photo-sharing company is storing user profile pictures in an Amazon S3 bucket and an image analysis application is deployed on four Amazon EC2 instances. A solutions architect would like

to trigger an image analysis procedure only on one of the four Amazon EC2 instances for each photo uploaded.

What do you recommend?

Create an Amazon S3 Event Notification that sends a message to an Amazon SNS topic. Subscribe the Amazon EC2 instances to the Amazon SNS topic

Correct answer

Create an Amazon S3 Event Notification that sends a message to an Amazon SQS queue. Make the Amazon EC2 instances read from the Amazon SQS queue

Subscribe the Amazon EC2 instances to the Amazon S3 Inventory stream

Create an Amazon EventBridge event that reacts to objects uploads in Amazon S3 and invokes one of the Amazon EC2 instances

Overall explanation

Correct option:

Create an Amazon S3 Event Notification that sends a message to an Amazon SQS queue. Make the Amazon EC2 instances read from the Amazon SQS queue

The Amazon S3 event notification feature enables you to receive notifications when certain events happen in your bucket. To enable notifications, you must first add a notification configuration that identifies the events you want Amazon S3 to publish and the destinations where you want Amazon S3 to send the notifications.

Amazon S3 supports the following destinations where it can publish events:

Amazon Simple Notification Service (Amazon SNS) topic

Amazon Simple Queue Service (Amazon SQS) queue

AWS Lambda

Amazon Simple Queue Service (SQS) is a fully managed message queuing service that enables you to decouple and scale microservices, distributed systems, and serverless applications. SQS offers two types of message queues. Standard queues offer maximum throughput, best-effort ordering, and at-least-once delivery. SQS FIFO queues are designed to guarantee that messages are processed exactly once, in the exact order that they are sent.

Here we have to use Amazon S3 Event Notifications (which can send a message to either AWS Lambda, Amazon SNS, or Amazon SQS) to send a message to the Amazon SQS queue. By using Amazon SQS, we know only one Amazon EC2 instance among the four will pick up a message and process it.

Incorrect options:

Subscribe the Amazon EC2 instances to the Amazon S3 Inventory stream - Amazon S3 Inventory is a distractor. If you're curious - Amazon S3 inventory helps you manage your storage by creating lists of the objects in an Amazon S3 bucket on a defined schedule.

Create an Amazon EventBridge event that reacts to objects uploads in Amazon S3 and invokes one of the Amazon EC2 instances- Amazon EventBridge events cannot invoke applications on Amazon EC2 instances, so we have to rule out that answer.

Create an Amazon S3 Event Notification that sends a message to an Amazon SNS topic. Subscribe the Amazon EC2 instances to the Amazon SNS topic- Amazon Simple Notification Service (Amazon SNS) is a highly available, durable, secure, fully managed pub/sub messaging service that enables you to decouple microservices, distributed systems, and serverless applications.

Using Amazon SNS would send a message to each Amazon EC2 instance via the Amazon SNS topic, therefore making all of them work for each upload. This is not the intended behavior.

Reference:

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

https://aws.amazon.com/sqs/

Domain

Design Resilient Architectures

Question 6Skipped

The development team at a company manages a Python based nightly process with a runtime of 30 minutes. The process can withstand any interruptions in its execution and start over again. The process currently runs on the on-premises infrastructure and it needs to be migrated to AWS.

Which of the following options do you recommend as the MOST cost-effective solution?

Run on AWS Lambda

Run on Amazon EMR

Correct answer

Run on a Spot Instance with a persistent request type

Run on an Application Load Balancer

Overall explanation

Correct option:

Run on a Spot Instance with a persistent request type

A Spot Instance is an unused Amazon EC2 instance that is available for less than the On-Demand price. Because Spot Instances enable you to request unused Amazon EC2 instances at steep discounts, you can lower your Amazon EC2 costs significantly. The hourly price for a Spot Instance is called a Spot price. The request type (one-time or persistent) determines whether the request is opened again when Amazon EC2 interrupts a Spot Instance or if you stop a Spot Instance. If the request is persistent, the request is opened again after your Spot Instance is interrupted. If the request is persistent and you stop your Spot Instance, the request only opens after you start your Spot Instance.

Incorrect options:

Run on an Application Load Balancer - Application Load Balancer operates at the request level (layer 7), routing traffic to targets – Amazon EC2 instances, containers, IP addresses, and AWS Lambda functions based on the content of the request. Ideal for advanced load balancing of HTTP and HTTPS traffic, Application Load Balancer provides advanced request routing targeted at delivery of modern application architectures, including microservices and container-based applications.

Application Load Balancer helps distribute load for HTTP(S) requests. This option has been added as a distractor.

Run on Amazon EMR - Amazon EMR is the industry-leading cloud big data platform for processing vast amounts of data using open source tools such as Apache Spark, Apache Hive, Apache HBase, Apache Flink, Apache Hudi, and Presto. Amazon EMR uses Hadoop, an open-source framework, to distribute your data and processing across a resizable cluster of Amazon EC2 instances.

Amazon EMR is to run Big Data load that is meant to be run on Hadoop, this is also a distractor.

Run on AWS Lambda - AWS Lambda lets you run code without provisioning or managing servers. You pay only for the compute time you consume.

AWS Lambda would be the perfect fit if our script could run in less than 15 minutes, as this is the maximum timeout for AWS Lambda.

Reference:

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/spot-requests.html

Domain

Design Cost-Optimized Architectures

Question 7Skipped

A development team has noticed that one of the Amazon EC2 instances has been incorrectly configured with the 'DeleteOnTermination' attribute set to True for its root EBS volume.

As a Solution's Architect, can you suggest a way to disable this flag while the instance is still running?

The attribute cannot be updated when the instance is running. Stop the instance from Amazon EC2 console and then update the flag

Update the attribute using AWS management console. Select the Amazon EC2 instance and then uncheck the DeleteOnTermination check box for the root EBS volume

Correct answer

Set the DeleteOnTermination attribute to False using the command line

Set the DisableApiTermination attribute of the instance using the API

Overall explanation

Correct option:

When an instance terminates, the value of the DeleteOnTermination attribute for each attached EBS volume determines whether to preserve or delete the volume. By default, the DeleteOnTermination attribute is set to True for the root volume and is set to False for all other volume types.

Set the DeleteOnTermination attribute to False using the command line

If the instance is already running, you can set DeleteOnTermination to False using the command line.

Incorrect options:

Update the attribute using AWS management console. Select the Amazon EC2 instance and then uncheck the DeleteOnTermination check box for the root EBS volume - You can set the DeleteOnTermination attribute to False when you launch a new instance. It is not possible to update this attribute of a running instance from the AWS console.

Set the DisableApiTermination attribute of the instance using the API - By default, you can terminate your instance using the Amazon EC2 console, command-line interface, or API. To prevent your instance from being accidentally terminated using Amazon EC2, you can enable termination protection for the instance. The DisableApiTermination attribute controls whether the instance can be terminated using the console, CLI, or API. This option cannot be used to control the delete status for the EBS volume when the instance terminates.

The attribute cannot be updated when the instance is running. Stop the instance from Amazon EC2 console and then update the flag - This statement is wrong and given only as a distractor.

References:

https://aws.amazon.com/premiumsupport/knowledge-center/deleteontermination-ebs/

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/terminating-instances.html#delete-on-termination-running-instance

Domain

Design Secure Architectures

Question 8Skipped

The systems administrator at a company wants to set up a highly available architecture for a bastion host solution.

As a solutions architect, which of the following options would you recommend as the solution?

Create a public Application Load Balancer that links to Amazon EC2 instances that are bastion hosts managed by an Auto Scaling Group

Create an elastic IP address (EIP) and assign it to all Amazon EC2 instances that are bastion hosts managed by an Auto Scaling Group

Create a VPC Endpoint for a fleet of Amazon EC2 instances that are bastion hosts managed by an Auto Scaling Group

Correct answer

Create a public Network Load Balancer that links to Amazon EC2 instances that are bastion hosts managed by an Auto Scaling Group

Overall explanation

Correct option:

Create a public Network Load Balancer that links to Amazon EC2 instances that are bastion hosts managed by an Auto Scaling Group

Network Load Balancer is best suited for use-cases involving low latency and high throughput workloads that involve scaling to millions of requests per second. Network Load Balancer operates at the connection level (Layer 4), routing connections to targets - Amazon EC2 instances, microservices, and containers – within Amazon Virtual Private Cloud (Amazon VPC) based on IP protocol data.

Including bastion hosts in your VPC environment enables you to securely connect to your Linux instances without exposing your environment to the Internet. After you set up your bastion hosts, you can access the other instances in your VPC through Secure Shell (SSH) connections on Linux. Bastion hosts are also configured with security groups to provide fine-grained ingress control.

You need to remember that Bastion Hosts are using the SSH protocol, which is a TCP based protocol on port 22. They must be publicly accessible.

Here, the correct answer is to use a Network Load Balancer, which supports TCP traffic, and will automatically allow you to connect to the Amazon EC2 instance in the backend.

Incorrect options:

Create an elastic IP address (EIP) and assign it to all Amazon EC2 instances that are bastion hosts managed by an Auto Scaling Group - An elastic IP address (EIP) can only be attached to one Amazon EC2 instance at a time, so it won't provide you a highly available setup on its own. Note that if we had two Elastic IPs and two Bastion Hosts, this would work.

Create a VPC Endpoint for a fleet of Amazon EC2 instances that are bastion hosts managed by an Auto Scaling Group - A VPC endpoint enables you to privately connect your VPC to supported AWS services and VPC endpoint services powered by AWS PrivateLink without requiring an internet gateway, NAT device, VPN connection, or AWS Direct Connect connection. Instances in your VPC do not require public IP addresses to communicate with resources in the service. Traffic between your VPC and the other service does not leave the Amazon network.

VPC Endpoints are not used on top of Amazon EC2 instances. They're a way to access AWS services privately within your VPC (without using the public internet). This is a distractor.

Create a public Application Load Balancer that links to Amazon EC2 instances that are bastion hosts managed by an Auto Scaling Group - Application Load Balancer (ALB) operates at the request level (layer 7), routing traffic to targets – Amazon EC2 instances, containers, IP addresses and AWS Lambda functions based on the content of the request. Ideal for advanced load balancing of HTTP and HTTPS traffic, Application Load Balancer provides advanced request routing targeted at delivery of modern application architectures, including microservices and container-based applications.

An Application Load Balancer only supports HTTP traffic, which is layer 7, while the SSH protocol is based on TCP and is layer 4. So, the Application Load Balancer doesn't work.

References:

https://docs.aws.amazon.com/quickstart/latest/linux-bastion/architecture.html

https://docs.aws.amazon.com/elasticloadbalancing/latest/network/introduction.html

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

Domain

Design High-Performing Architectures

Question 9Skipped

The engineering team at an e-commerce company wants to set up a custom domain for internal usage such as internaldomainexample.com. The team wants to use the private hosted zones feature of Amazon Route 53 to accomplish this.

Which of the following settings of the VPC need to be enabled? (Select two)

enableDnsDomain

Correct selection

enableDnsSupport

enableVpcHostnames

enableVpcSupport

Correct selection

enableDnsHostnames

Overall explanation

Correct options:

enableDnsHostnames

enableDnsSupport

A private hosted zone is a container for records for a domain that you host in one or more Amazon virtual private clouds (VPCs). You create a hosted zone for a domain (such as example.com), and then you create records to tell Amazon Route 53 how you want traffic to be routed for that domain within and among your VPCs.

For each VPC that you want to associate with the Route 53 hosted zone, change the following VPC settings to true:

enableDnsHostnames

enableDnsSupport

Incorrect options:

enableVpcSupport

enableVpcHostnames

enableDnsDomain

The options enableVpcSupport, enableVpcHostnames and enableDnsDomain have been added as distractors.

Reference:

https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/hosted-zone-private-creating.html

Domain

Design Resilient Architectures

Question 10Skipped

As a Solutions Architect, you have set up a database on a single Amazon EC2 instance that has an Amazon EBS volume of type gp2. You currently have 300 gigabytes of space on the gp2 device. The Amazon EC2 instance is of type m5.large. The database performance has recently been poor and upon looking at Amazon CloudWatch, you realize the IOPS on the Amazon EBS volume is maxing out. The disk size of the database must not change because of a licensing issue.

How do you troubleshoot this issue?

Convert the Amazon EC2 instance to an i3.4xlarge

Increase the IOPS on the gp2 volume

Correct answer

Convert the gp2 volume to an io1

Stop the Amazon CloudWatch agent to improve performance

Overall explanation

Correct option:

Amazon EBS provides the following volume types, which differ in performance characteristics and price so that you can tailor your storage performance and cost to the needs of your applications. The volumes types fall into two categories:

SSD-backed volumes optimized for transactional workloads involving frequent read/write operations with small I/O size, where the dominant performance attribute is IOPS

HDD-backed volumes optimized for large streaming workloads where throughput (measured in MiB/s) is a better performance measure than IOPS

Convert the gp2 volume to an io1

Provisioned IOPS SSD (io1) volumes are designed to meet the needs of I/O-intensive workloads, particularly database workloads, that are sensitive to storage performance and consistency. Unlike gp2, which uses a bucket and credit model to calculate performance, an io1 volume

allows you to specify a consistent IOPS rate when you create the volume, and Amazon EBS delivers the provisioned performance 99.9 percent of the time.

The only solution is to convert the volume into an io1 volume. This will allow us to keep the same disk size while independently increasing the IOPS for that volume.

Incorrect options:

Stop the Amazon CloudWatch agent to improve performance - The Amazon CloudWatch agent does not have any impact on the performance of the instance.

Increase the IOPS on the gp2 volume - General Purpose SSD (gp2) volumes offer cost-effective storage that is ideal for a broad range of workloads. These volumes deliver single-digit millisecond latencies and the ability to burst to 3,000 IOPS for extended periods. Between a minimum of 100 IOPS (at 33.33 GiB and below) and a maximum of 16,000 IOPS (at 5,334 GiB and above), baseline performance scales linearly at 3 IOPS per GiB of volume size. AWS designs gp2 volumes to deliver their provisioned performance 99% of the time. A gp2 volume can range in size from 1 GiB to 16 TiB.

IOPS cannot be directly increased on a gp2 volume without increasing its size, which is not possible due to the question's constraints.

Convert the Amazon EC2 instance to an i3.4xlarge - Converting the Amazon EC2 instance to i3.4xlarge won't improve the Amazon EBS drive's performance.

References:

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-volume-types.html#EBSVolumeTypes_gp2

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-volume-types.html#EBSVolumeTypes_piops

Domain

Design High-Performing Architectures

Question 11Skipped

A healthcare company runs a fleet of Amazon EC2 instances in two private subnets (named PR1 and PR2) across two Availability Zones (AZs) named A1 and A2. The Amazon EC2 instances need access to the internet for operating system patch management and third-party software maintenance. To facilitate this, the engineering team at the company wants to set up two Network Address Translation gateways (NAT gateways) in a highly available configuration.

Which of the following options would you suggest?

Set up a total of two NAT gateways. NAT gateway N1 should be set up in private subnet PR1 in Availability Zone A1. NAT gateway N2 should be set up in private subnet PR2 in Availability Zone A2

Set up a total of one NAT gateway. NAT gateway N1 should be set up in public subnet PU1 in any of the Availability Zones A1 or A2

Set up a total of two NAT gateways. Both NAT gateways N1 and N2 should be set up in a single public subnet PU1 in any of the Availability Zones A1 or A2

Correct answer

Set up a total of two NAT gateways. NAT gateway N1 should be set up in public subnet PU1 in Availability Zone A1. NAT gateway N2 should be set up in public subnet PU2 in Availability Zone A2

Overall explanation

Correct option:

Set up a total of two NAT gateways. NAT gateway N1 should be set up in public subnet PU1 in Availability Zone A1. NAT gateway N2 should be set up in public subnet PU2 in Availability Zone A2

A NAT gateway is a Network Address Translation (NAT) service. You can use a NAT gateway so that instances in a private subnet can connect to services outside your VPC but external services cannot initiate a connection with those instances.

For the given use case, the Amazon EC2 instances in the private subnets can connect to the internet through public NAT gateways in their respective Availability Zones (AZ). You should create public NAT gateway in the public subnet of each AZ and must associate an elastic IP address with the NAT gateway at creation. Then, you can route traffic from the NAT gateway to the internet gateway for the VPC.

If you have resources in multiple Availability Zones and they share one NAT gateway, and if the NAT gateway's Availability Zone is down, resources in the other Availability Zones lose internet access. To create a highly available or an Availability Zone independent architecture, create a NAT gateway in each Availability Zone and configure your routing to ensure that resources use the NAT gateway in the same Availability Zone.

NAT gateway basics

Each NAT gateway is created in a specific Availability Zone and implemented with redundancy in that zone. There is a quota on the number of NAT gateways that you can create in each Availability Zone. For more information, see Amazon VPC quotas.

If you have resources in multiple Availability Zones and they share one NAT gateway, and if the NAT gateway's Availability Zone is down, resources in the other Availability Zones lose internet access. To create an Availability Zone-independent architecture, create a NAT gateway in each Availability Zone and configure your routing to ensure that resources use the NAT gateway in the same Availability Zone.

The following characteristics and rules apply to NAT gateways:

- A NAT gateway supports the following protocols: TCP, UDP, and ICMP.
- NAT gateways are supported for IPv4 or IPv6 traffic. For IPv6 traffic, NAT gateway performs NAT64. By using this in conjunction with DNS64 (available on Route 53 resolver), your IPv6 workloads in a subnet in Amazon VPC can communicate with IPv4 resources. These IPv4 services may be present in the same VPC (in a separate subnet) or a different VPC, on your on-premises environment or on the internet.
- A NAT gateway supports 5 Gbps of bandwidth and automatically scales up to 100 Gbps. If you require more bandwidth, you can split your resources into multiple subnets and create a NAT gateway in each subnet.
- A NAT gateway can process one million packets per second and automatically scales up to ten million packets per second. Beyond this limit, a NAT gateway will drop packets. To prevent packet loss, split your resources into multiple subnets and create a separate NAT gateway for each subnet.
- A NAT gateway can support up to 55,000 simultaneous connections to each unique destination. This limit also applies if you create approximately 900 connections per second to a single destination (about 55,000 connections per minute). If the destination IP address, the destination port, or the protocol (TCP/UDP/ICMP) changes, you can create an additional 55,000 connections. For more than 55,000 connections, there is an increased chance of connection errors due to port allocation errors. These errors can be monitored by viewing the `ErrorPortAllocation` CloudWatch metric for your NAT gateway. For more information, see Monitor NAT gateways with Amazon CloudWatch.
- You can associate exactly one Elastic IP address with a public NAT gateway. You cannot disassociate an Elastic IP address from a NAT gateway after it's created. To use a different Elastic IP address for your NAT gateway, you must create a new NAT gateway with the required address, update your route tables, and then delete the existing NAT gateway if it's no longer required.
- A private NAT gateway receives an available private IP address from the subnet in which it is configured. The assigned private IP address persists until you delete the private NAT gateway. You cannot detach the private IP address and you cannot attach additional private IP addresses.
- You cannot associate a security group with a NAT gateway. You can associate security groups with your instances to control inbound and outbound traffic.
- You can use a network ACL to control the traffic to and from the subnet for your NAT gateway. NAT gateways use ports 1024–65535. For more information, see Control traffic to subnets using Network ACLs.
- A NAT gateway receives a network interface that's automatically assigned a private IP address from the IP address range of the subnet. You can view the network interface for the NAT gateway using the Amazon EC2 console. For more information, see Viewing details about a network interface. You cannot modify the attributes of this network interface.
- A NAT gateway cannot be accessed through a ClassicLink connection that is associated with your VPC.
- You cannot route traffic to a NAT gateway through a VPC peering connection, a Site-to-Site VPN connection, or AWS Direct Connect. A NAT gateway cannot be used by resources on the other side of these connections.

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

Incorrect options:

Set up a total of two NAT gateways. NAT gateway N1 should be set up in private subnet PR1 in Availability Zone A1. NAT gateway N2 should be set up in private subnet PR2 in Availability Zone A2 - For the Amazon EC2 instances in the private subnet, you can facilitate outbound internet connectivity in a highly available configuration by creating a public NAT gateway in the public subnet of each AZ. You cannot create NAT gateways in the private subnet for the given use case.

Set up a total of two NAT gateways. Both NAT gateways N1 and N2 should be set up in a single public subnet PU1 in any of the Availability Zones A1 or A2 - For the Amazon EC2 instances in the private subnet, you can facilitate outbound internet connectivity in a highly available configuration by creating a public NAT gateway in the public subnet of each AZ. You cannot create both NAT gateways in a single public subnet, as this configuration would not be highly available.

Set up a total of one NAT gateway. NAT gateway N1 should be set up in public subnet PU1 in any of the Availability Zones A1 or A2 - For the Amazon EC2 instances in the private subnet,

you can facilitate outbound internet connectivity in a highly available configuration by creating a public NAT gateway in the public subnet of each AZ. You cannot create a single NAT gateway, as this configuration would not be highly available.

Reference:

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

Domain

Design Resilient Architectures

Question 12Skipped

An application is hosted on multiple Amazon EC2 instances in the same Availability Zone (AZ). The engineering team wants to set up shared data access for these Amazon EC2 instances using Amazon EBS Multi-Attach volumes.

Which Amazon EBS volume type is the correct choice for these Amazon EC2 instances?

Throughput Optimized HDD Amazon EBS volumes

Correct answer

Provisioned IOPS SSD Amazon EBS volumes

Cold HDD Amazon EBS volumes

General-purpose SSD-based Amazon EBS volumes

Overall explanation

Correct option:

Provisioned IOPS SSD Amazon EBS volumes

Amazon EBS Multi-Attach enables you to attach a single Provisioned IOPS SSD (io1 or io2) volume to multiple instances that are in the same Availability Zone. You can attach multiple Multi-Attach enabled volumes to an instance or set of instances. Each instance to which the volume is attached has full read and write permission to the shared volume. Multi-Attach makes it easier for you to achieve higher application availability in clustered Linux applications that manage concurrent write operations.

Multi-Attach is supported exclusively on Provisioned IOPS SSD volumes.

Incorrect options:

General-purpose SSD-based Amazon EBS volumes - These SSD-backed Amazon EBS volumes provide a balance of price and performance. AWS recommends these volumes for most workloads. These volume types are not supported for Multi-Attach functionality.

Throughput Optimized HDD Amazon EBS volumes - These HDD-backed volumes provide a low-cost HDD designed for frequently accessed, throughput-intensive workloads. These volume types are not supported for Multi-Attach functionality.

Cold HDD Amazon EBS volumes - These HDD-backed volumes provide a lowest-cost HDD design for less frequently accessed workloads. These volume types are not supported for Multi-Attach functionality.

Reference:

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-volumes-multi.html

Domain

Design Resilient Architectures

Question 13Skipped

The CTO of an online home rental marketplace wants to re-engineer the caching layer of the current architecture for its relational database. The CTO wants the caching layer to have replication and archival support built into the architecture.

Which of the following AWS service offers the capabilities required for the re-engineering of the caching layer?

Correct answer

Amazon ElastiCache for Redis

Amazon ElastiCache for Memcached

Amazon DynamoDB Accelerator (DAX)

Amazon DocumentDB

Overall explanation

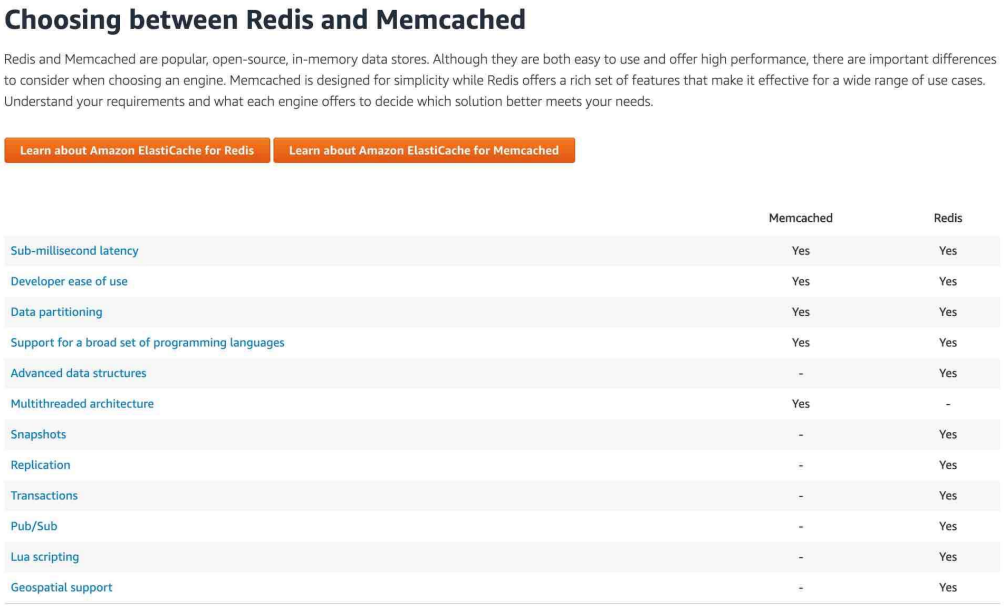
Correct option:

Amazon ElastiCache for Redis

Amazon ElastiCache for Redis is a blazing fast in-memory data store that provides sub-millisecond latency to power internet-scale real-time applications. Amazon ElastiCache for Redis is a great choice for real-time transactional and analytical processing use cases such as caching, chat/messaging, gaming leaderboards, geospatial, machine learning, media streaming, queues, real-time analytics, and session store. ElastiCache for Redis supports replication and archival snapshots right out of the box. Hence this is the correct option.

Exam Alert:

Please review this comparison sheet for Redis vs Memcached features:



Choosing between Redis and Memcached

Redis and Memcached are popular, open-source, in-memory data stores. Although they are both easy to use and offer high performance, there are important differences to consider when choosing an engine. Memcached is designed for simplicity while Redis offers a rich set of features that make it effective for a wide range of use cases. Understand your requirements and what each engine offers to decide which solution better meets your needs.

Learn about Amazon ElastiCache for Redis | Learn about Amazon ElastiCache for Memcached

	Memcached	Redis
Sub-millisecond latency	Yes	Yes
Developer ease of use	Yes	Yes
Data partitioning	Yes	Yes
Support for a broad set of programming languages	Yes	Yes
Advanced data structures	-	Yes
Multithreaded architecture	Yes	-
Snapshots	-	Yes
Replication	-	Yes
Transactions	-	Yes
Pub/Sub	-	Yes
Lua scripting	-	Yes
Geospatial support	-	Yes

via - https://aws.amazon.com/elasticache/redis-vs-memcached/

Incorrect options:

Amazon ElastiCache for Memcached - Amazon ElastiCache for Memcached is a Memcached-compatible in-memory key-value store service that can be used as a cache or a data store. Amazon ElastiCache for Memcached is a great choice for implementing an in-memory cache to decrease access latency, increase throughput, and ease the load off your relational or NoSQL database. Session stores are easy to create with Amazon ElastiCache for Memcached. ElastiCache for Memcached does not support replication and archival snapshots, so this option is ruled out.

Amazon DynamoDB Accelerator (DAX) - Amazon DynamoDB is a key-value and document database that delivers single-digit millisecond performance at any scale. It's a fully managed, multi-region, multi-master, durable database with built-in security, backup and restore, and in-memory caching for internet-scale applications. DAX is a DynamoDB-compatible caching service that enables you to benefit from fast in-memory performance for demanding applications. DAX cannot be used as a caching layer for a relational database.

Amazon DocumentDB - Amazon DocumentDB is a fast, scalable, highly available, and fully managed document database service that supports MongoDB workloads. As a document database, Amazon DocumentDB makes it easy to store, query, and index JSON data. DocumentDB cannot be used as a caching layer for a relational database.

References:

https://aws.amazon.com/elasticache/redis/

https://aws.amazon.com/elasticache/redis-vs-memcached/

Domain

Design High-Performing Architectures

Question 14Skipped

A retail company needs a secure connection between its on-premises data center and AWS Cloud. This connection does not need high bandwidth and will handle a small amount of traffic. The company wants a quick turnaround time to set up the connection.

What is the MOST cost-effective way to establish such a connection?

Set up an Internet Gateway between the on-premises data center and AWS cloud

Set up a bastion host on Amazon EC2

Correct answer

Set up an AWS Site-to-Site VPN connection

Set up AWS Direct Connect

Overall explanation

Correct option:

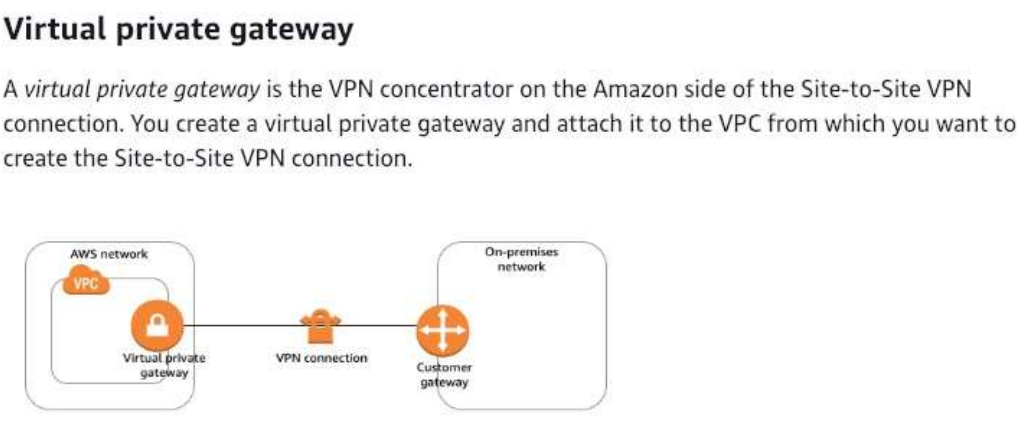
Set up an AWS Site-to-Site VPN connection

By default, instances that you launch into an Amazon VPC can't communicate with your own (remote) network. You can enable access to your remote network from your VPC by creating an AWS Site-to-Site VPN (Site-to-Site VPN) connection, and configuring routing to pass traffic through the connection. A VPN connection refers to the connection between your VPC and your own on-premises network.

An AWS Site-to-Site VPN connection offers two VPN tunnels between a virtual private gateway or a transit gateway on the AWS side, and a customer gateway (which represents a VPN device) on the remote (on-premises) side.

A virtual private gateway (VGW) is the VPN concentrator on the Amazon side of the AWS Site-to-Site VPN connection. You create a virtual private gateway and attach it to the VPC from which you want to create the AWS Site-to-Site VPN connection.

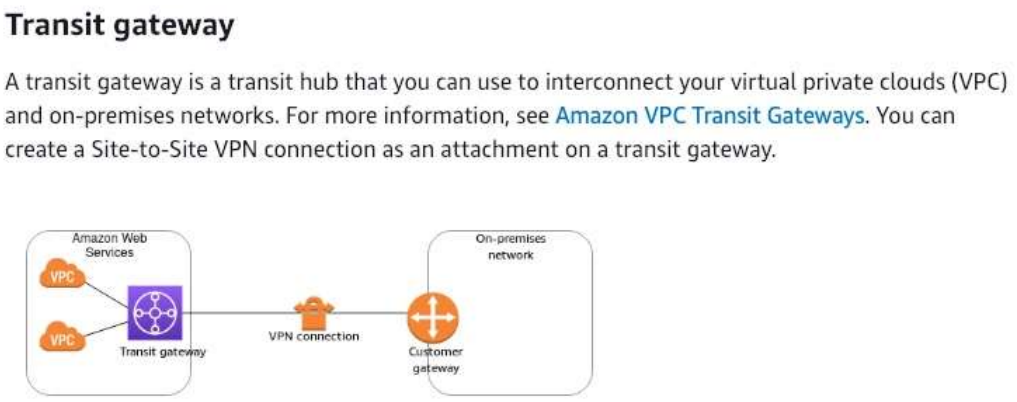
How virtual private gateway works:



via - https://docs.aws.amazon.com/vpn/latest/s2svpn/how_it_works.html

An AWS transit gateway is a transit hub that you can use to interconnect your virtual private clouds (VPC) and on-premises networks. For more information, see Amazon VPC Transit Gateways. You can create a Site-to-Site VPN connection as an attachment on a transit gateway.

How AWS transit gateway works:



via - https://docs.aws.amazon.com/vpn/latest/s2svpn/how_it_works.html

Incorrect options:

Set up a bastion host on Amazon EC2 - A bastion host is a server whose purpose is to provide access to a private network from an external network, such as the Internet. The bastion host runs on an Amazon EC2 instance that is typically in a public subnet of your Amazon VPC. Other Amazon EC2 instances can be in a subnet that is not publicly accessible, and they are set up with a security group that allows SSH access from the security group attached to the underlying Amazon EC2 instance running the bastion host. A bastion host cannot be used to set up a connection between its on-premises data center and AWS Cloud.

Set up AWS Direct Connect - AWS Direct Connect is a network service that provides an alternative to using the Internet to utilize AWS cloud services. AWS Direct Connect enables customers to have low latency, secure and private connections to AWS for workloads that require higher speed or lower latency than the internet. A Dedicated Connection is made through a 1 Gbps, 10 Gbps, or 100 Gbps Ethernet port dedicated to a single customer. AWS Direct Connect takes about a month to provision the connection, so this option is ruled out for the given use case.

Set up an Internet Gateway between the on-premises data center and AWS cloud - An Internet Gateway is a horizontally scaled, redundant, and highly available VPC component that allows communication between your VPC and the internet. An Internet Gateway cannot be used to set up a connection between its on-premises data center and AWS Cloud.

References:

https://docs.aws.amazon.com/vpn/latest/s2svpn/how_it_works.html

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

Domain

Design Secure Architectures

Question 15Skipped

The engineering team at a multi-national company uses AWS Firewall Manager to centrally configure and manage firewall rules across its accounts and applications using AWS Organizations.

Which of the following AWS resources can the AWS Firewall Manager configure rules on? (Select three)

Amazon Inspector

Correct selection

AWS Web Application Firewall (AWS WAF)

Correct selection

AWS Shield Advanced

Network access control list (network ACL)

Amazon GuardDuty

Correct selection

VPC Security Groups

Overall explanation

Correct options:

AWS Web Application Firewall (AWS WAF)

AWS Shield Advanced

VPC Security Groups

AWS Firewall Manager is a security management service which allows you to centrally configure and manage firewall rules across your accounts and applications in AWS Organizations. As new applications are created, Firewall Manager makes it easy to bring new applications and resources into compliance by enforcing a common set of security rules. Now you have a single service to build firewall rules, create security policies, and enforce them in a consistent, hierarchical manner across your entire infrastructure.

Using AWS Firewall Manager, you can centrally configure AWS WAF rules, AWS Shield Advanced protection, Amazon Virtual Private Cloud (VPC) security groups, AWS Network Firewalls, and Amazon Route 53 Resolver DNS Firewall rules across accounts and resources in your organization. It does not support Network ACLs as of today.

Q: What does AWS Firewall Manager configure?

Using AWS Firewall Manager, you can centrally configure AWS WAF rules, AWS Shield Advanced protections, Amazon Virtual Private Cloud (VPC) security groups, AWS Network Firewalls, and Amazon Route 53 Resolver DNS Firewall rules across accounts and resources in your organization.

Q: Does AWS Firewall Manager configure VPC security groups or Network ACLs?

Yes, AWS Firewall Manager does support configuration of VPC security groups. However, it does not support Network ACLs today.

Q: Which AWS resources can AWS Firewall Manager configure rules on?

Using AWS Firewall Manager, you can

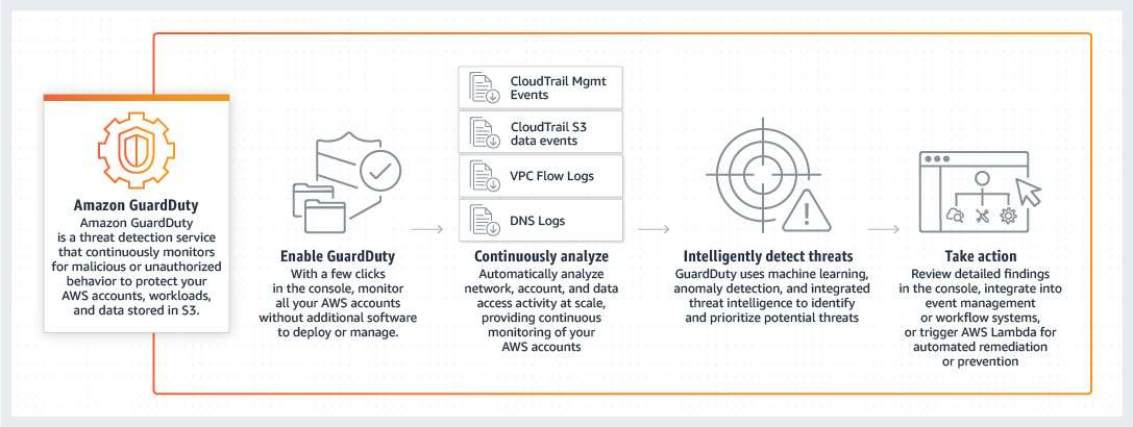
- Easily roll out AWS WAF rules across Application Load Balancer, API Gateways and Amazon CloudFront distributions.
- You can create AWS Shield Advanced protections for your Application Load Balancers, ELB Classic Load Balancers, Elastic IP Addresses and CloudFront distributions.
- You can configure new Amazon Virtual Private Cloud (VPC) security groups and audit any existing security groups for your Amazon EC2, Application Load Balancers (ALBs) and ENI resource types.
- You can also deploy AWS Network Firewalls across accounts and VPCs in your organization.
- Finally, with AWS Firewall Manager, you can also associate Amazon Route 53 Resolver DNS Firewall rules across VPCs in your organization.

via - https://aws.amazon.com/firewall-manager/faqs/

Incorrect options:

Amazon GuardDuty - Amazon GuardDuty offers threat detection that enables you to continuously monitor and protect your AWS accounts, workloads, and data stored in Amazon S3. Amazon GuardDuty analyzes continuous streams of meta-data generated from your account and network activity found in AWS CloudTrail Events, Amazon VPC Flow Logs, and DNS Logs.

How Amazon GuardDuty Works:



Amazon Inspector - Amazon Inspector is an automated security assessment service that helps you test the network accessibility of your Amazon EC2 instances and the security state of your applications running on the instances.

Network access control list (network ACL) - A network access control list (ACL) is an optional layer of security for your VPC that acts as a firewall for controlling traffic in and out of one or more subnets.

These three options are not in the list of AWS resources supported by AWS Firewall Manager, so these options are incorrect.

References:

https://aws.amazon.com/firewall-manager/faqs/

https://aws.amazon.com/guardduty/

https://aws.amazon.com/inspector/

Domain

Design Secure Architectures

Question 16Skipped

Your application is deployed on Amazon EC2 instances fronted by an Application Load Balancer. Recently, your infrastructure has come under attack. Attackers perform over 100 requests per second, while your normal users only make about 5 requests per second.

How can you efficiently prevent attackers from overwhelming your application?

Use AWS Shield Advanced and setup a rate-based rule

Define a network access control list (network ACL) on your Application Load Balancer

Configure Sticky Sessions on the Application Load Balancer

Correct answer

Use an AWS Web Application Firewall (AWS WAF) and setup a rate-based rule

Overall explanation

Correct option:

Use an AWS Web Application Firewall (AWS WAF) and setup a rate-based rule

AWS Web Application Firewall (AWS WAF) is a web application firewall that helps protect your web applications or APIs against common web exploits that may affect availability, compromise security, or consume excessive resources. AWS WAF gives you control over how traffic reaches your applications by enabling you to create security rules that block common attack patterns, such as SQL injection or cross-site scripting, and rules that filter out specific traffic patterns you define.

The correct answer is to use WAF (which has integration on top of your ALB) and define a rate-based rule.

Incorrect options:

Configure Sticky Sessions on the Application Load Balancer - Application Load Balancer (ALB) operates at the request level (layer 7), routing traffic to targets – Amazon EC2 instances, containers, IP addresses and Lambda functions based on the content of the request. Ideal for

advanced load balancing of HTTP and HTTPS traffic, Application Load Balancer provides advanced request routing targeted at delivery of modern application architectures, including microservices and container-based applications.

Sticky Sessions on your Application Load Balancer is a distractor here. Sticky sessions are a mechanism to route requests from the same client to the same target. Application Load Balancer supports sticky sessions using load balancer generated cookies. If you enable sticky sessions, the same target receives the request and can use the cookie to recover the session context.

Define a network access control list (network ACL) on your Application Load Balancer - A network access control list (network ACL) does not work, as this only helps to block specific IPs. On top of things, network access control list (network ACL) is defined at the subnet level, and not for an Application Load Balancer.

Use AWS Shield Advanced and setup a rate-based rule - AWS Shield is a managed Distributed Denial of Service (DDoS) protection service that safeguards applications running on AWS. AWS Shield provides always-on detection and automatic inline mitigations that minimize application downtime and latency, so there is no need to engage AWS Support to benefit from DDoS protection. There are two tiers of AWS Shield - Standard and Advanced.

AWS Shield Advanced provides enhanced resource-specific detection and employs advanced mitigation and routing techniques for sophisticated or larger attacks.

AWS Shield Advanced will give you DDoS protection overall, and you cannot set up rate-based rules in Shield.

References:

https://aws.amazon.com/waf/

https://docs.aws.amazon.com/waf/latest/developerguide/waf-rule-statement-type-rate-based.html

https://aws.amazon.com/shield/

https://docs.aws.amazon.com/elasticloadbalancing/latest/application/load-balancer-target-groups.html#sticky-sessions

Domain

Design Secure Architectures

Question 17Skipped

A startup wants to create a highly available architecture for its multi-tier application. Currently, the startup manages a single Amazon EC2 instance along with a single Amazon RDS MySQL DB instance. The startup has hired you as an AWS Certified Solutions Architect - Associate to build a solution that meets these requirements while minimizing the underlying infrastructure maintenance effort.

What will you recommend?

Create an Auto-Scaling group with a desired capacity of a total of two Amazon EC2 instances in a single Availability Zone. Configure an Application Load Balancer having a

target group of these Amazon EC2 instances. Set up Amazon RDS MySQL DB in a multi-AZ configuration

Create an Auto-Scaling group with a desired capacity of a total of two Amazon EC2 instances across two Availability Zones. Configure an Application Load Balancer having a target group of these Amazon EC2 instances. Set up a read replica of the Amazon RDS MySQL DB in another Availability Zone

Provision a second Amazon EC2 instance in another Availability Zone. Provision a second Amazon RDS MySQL DB in another Availabililty Zone. Leverage Amazon Route 53 for equal distribution of incoming traffic to the Amazon EC2 instances. Use a custom script to sync data across the two MySQL DBs

Correct answer

Create an Auto-Scaling group with a desired capacity of a total of two Amazon EC2 instances across two Availability Zones. Configure an Application Load Balancer having a target group of these Amazon EC2 instances. Set up Amazon RDS MySQL DB in a multi-AZ configuration

Overall explanation

Correct option:

Create an Auto-Scaling group with a desired capacity of a total of two Amazon EC2 instances across two Availability Zones. Configure an Application Load Balancer having a target group of these Amazon EC2 instances. Set up Amazon RDS MySQL DB in a multi-AZ configuration

Amazon EC2 Auto Scaling is a fully managed service designed to launch or terminate Amazon EC2 instances automatically to help ensure you have the correct number of Amazon EC2 instances available to handle the load for your application.

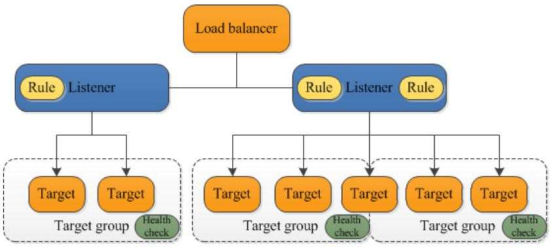
Application Load Balancer components

A *load balancer* serves as the single point of contact for clients. The load balancer distributes incoming application traffic across multiple targets, such as EC2 instances, in multiple Availability Zones. This increases the availability of your application. You add one or more listeners to your load balancer.

A *listener* checks for connection requests from clients, using the protocol and port that you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets. Each rule consists of a priority, one or more actions, and one or more conditions. When the conditions for a rule are met, then its actions are performed. You must define a default rule for each listener, and you can optionally define additional rules.

Each *target group* routes requests to one or more registered targets, such as EC2 instances, using the protocol and port number that you specify. You can register a target with multiple target groups. You can configure health checks on a per target group basis. Health checks are performed on all targets registered to a target group that is specified in a listener rule for your load balancer.

The following diagram illustrates the basic components. Notice that each listener contains a default rule, and one listener contains another rule that routes requests to a different target group. One target is registered with two target groups.



via - https://docs.aws.amazon.com/autoscaling/ec2/userguide/what-is-amazon-ec2-auto-scaling.html

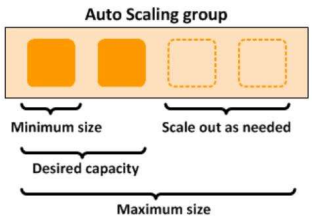
Application Load Balancer automatically distributes your incoming traffic across multiple targets, such as Amazon EC2 instances, containers, and IP addresses, in one or more Availability Zones. It monitors the health of its registered targets, and routes traffic only to the healthy targets.

What is Amazon EC2 Auto Scaling?

PDF | RSS

Amazon EC2 Auto Scaling helps you ensure that you have the correct number of Amazon EC2 instances available to handle the load for your application. You create collections of EC2 instances, called *Auto Scaling groups*. You can specify the minimum number of instances in each Auto Scaling group, and Amazon EC2 Auto Scaling ensures that your group never goes below this size. You can specify the maximum number of instances in each Auto Scaling group, and Amazon EC2 Auto Scaling ensures that your group never goes above this size. If you specify the desired capacity, either when you create the group or at any time thereafter, Amazon EC2 Auto Scaling ensures that your group has this many instances. If you specify scaling policies, then Amazon EC2 Auto Scaling can launch or terminate instances as demand on your application increases or decreases.

For example, the following Auto Scaling group has a minimum size of one instance, a desired capacity of two instances, and a maximum size of four instances. The scaling policies that you define adjust the number of instances, within your minimum and maximum number of instances, based on the criteria that you specify.



via - https://docs.aws.amazon.com/elasticloadbalancing/latest/application/introduction.html

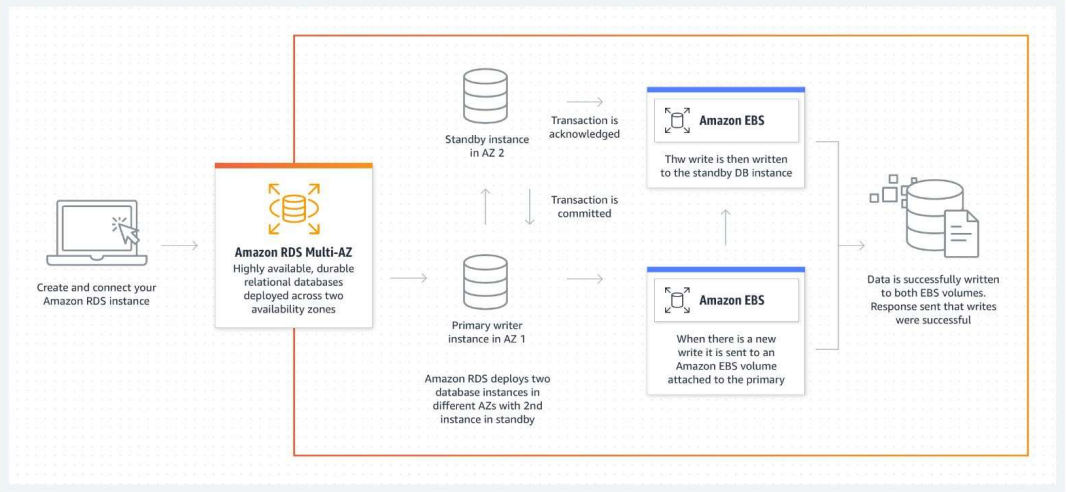
In a multi-AZ deployment, Amazon RDS automatically provisions and maintains a synchronous "standby" replica in a different Availability Zone. Updates to your DB Instance are synchronously replicated across Availability Zones to the standby to keep both in sync and protect your latest database updates against DB instance failure.

Amazon RDS Multi-AZ with one standby

Automatic fail over	Protect database performance	Enhance durability	Increase availability
Support high availability for your application with automatic database failover that completes as quickly as 60 seconds with zero data loss and no manual intervention.	Avoid suspending I/O activity on your primary during backup by backing up from your standby instance.	Use Amazon RDS Multi-AZ synchronous replication technologies to keep data on your standby database instance up to date with the primary.	Enhance availability by deploying a standby instance in a second AZ, and achieve fault tolerance in the event of an AZ or database instance failure.

How it works

In an Amazon RDS Multi-AZ deployment, Amazon RDS automatically creates a primary database (DB) instance and synchronously replicates the data to an instance in a different AZ. When it detects a failure, Amazon RDS automatically fails over to a standby instance without manual intervention.



via - https://aws.amazon.com/rds/features/multi-az/

To create a highly available architecture for the given use case, you need to set up an Auto-Scaling group with a desired capacity of a total of two Amazon EC2 instances across two Availability Zones and then point the Application Load Balancer to the target group having the Amazon EC2 instances.

Incorrect options:

Create an Auto-Scaling group with a desired capacity of a total of two Amazon EC2 instances across two Availability Zones. Configure an Application Load Balancer having a target group of these Amazon EC2 instances. Set up a read replica of the Amazon RDS MySQL DB in another Availability Zone - A read replica cannot be used to enhance the

availability of an Amazon RDS MySQL DB. You must use the multi-AZ configuration of Amazon RDS MySQL for this use case.

Create an Auto-Scaling group with a desired capacity of a total of two Amazon EC2 instances in a single Availability Zone. Configure an Application Load Balancer having a target group of these Amazon EC2 instances. Set up Amazon RDS MySQL DB in a multi-AZ configuration - Having the Amazon EC2 instances in a single Availability Zone will not create a highly available solution. In the case of an outage for the entire Availability Zone, the Amazon EC2 instances would be unreachable. Hence this option is incorrect.

Provision a second Amazon EC2 instance in another Availability Zone. Provision a second Amazon RDS MySQL DB in another Availabililty Zone. Leverage Amazon Route 53 for equal distribution of incoming traffic to the Amazon EC2 instances. Use a custom script to sync data across the two MySQL DBs - This option has been added as a distractor. It requires significant monitoring and development effort to keep the Amazon EC2 instances highly available as well as keep the MySQL DBs in sync.

References:

https://docs.aws.amazon.com/autoscaling/ec2/userguide/what-is-amazon-ec2-auto-scaling.html

https://docs.aws.amazon.com/elasticloadbalancing/latest/application/introduction.html

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

Domain

Design Resilient Architectures

Question 18Skipped

A company is transferring a significant volume of data from on-site storage to AWS, where it will be accessed by Windows, Mac, and Linux-based Amazon EC2 instances within the same AWS region using both SMB and NFS protocols. Part of this data will be accessed regularly, while the rest will be accessed less frequently. The company requires a hosting solution for this data that minimizes operational overhead.

What solution would best meet these requirements?

Set up an Amazon FSx for OpenZFS instance. Configure an FSx for OpenZFS file ystem on the root volume and migrate the data to the FSx for OpenZFS volume

Set up an Amazon Elastic File System (Amazon EFS) volume that uses EFS Intelligent-Tiering. Use AWS DataSync to migrate the data to the EFS volume

Correct answer

Set up an Amazon FSx for ONTAP instance. Configure an FSx for ONTAP file system on the root volume and migrate the data to the FSx for ONTAP volume

Set up an Amazon Elastic File System (Amazon EFS) volume that uses EFS Infrequent Access. Use AWS DataSync to migrate the data to the EFS volume

Overall explanation

Correct option:

Set up an Amazon FSx for ONTAP instance. Configure an FSx for ONTAP file system on the root volume and migrate the data to the FSx for ONTAP volume

Amazon FSx for NetApp ONTAP is a storage service that allows customers to launch and run fully managed ONTAP file systems in the cloud. ONTAP is NetApp's file system technology that provides a widely adopted set of data access and data management capabilities.

Amazon FSx for NetApp ONTAP Overview via - https://aws.amazon.com/fsx/netapp-ontap/

The given use case mandates that the storage on AWS will be accessed by Windows, Mac, and Linux-based Amazon EC2 instances within the same AWS region using both SMB and NFS protocols. Amongst the Amazon FSx family, FSx for ONTAP is the only file system that supports this key requirement.

Selecting a file system based on workload requirements

Amazon FSx file systems offer feature sets, performance profiles, and data management capabilities that support a wide variety of use cases. You can choose a file system that enables you to cost-effectively power your workload with the necessary reliability, functionality, performance, and security.

	FSx for NetApp ONTAP	FSx for OpenZFS	FSx for Windows File Server	FSx for Lustre
Performance and Scale				
Latency	<1 ms	<0.5 ms	<1 ms	<1 ms
Max. throughput per file system	72-80 GB/s*	10-21 GB/s*	12-20 GB/s*	1000 GB/s
Max. throughput available to a single client accessing a file system	18 GB/s	10 GB/s	20 GB/s	21 GB/s
Max. IOPS per file system	Millions	1-2 million	Hundreds of thousands	Millions
Maximum file system size	Virtually unlimited (10s of PBs)	512 TiB	64 TiB	Multiple PBs
** The lower number in the range refers to baseline throughput. The upper number in the range refers to higher levels of throughput enabled by automatic caching of frequently accessed data, network and disk performance bursting, and efficiencies from data compression.*				
Accessibility and Integrations				
Client compatibility	Windows, Linux macOS	Windows, Linux, macOS	Windows, Linux, macOS	Linux
Protocol support	SMB 2.0, 2.1, 3.0, 3.1.1 NFS 3, 4.0, 4.1, 4.2 iSCSI (shared block storage)	NFS 3, 4.0, 4.1, 4.2	SMB 2.0, 2.1, 3.0, 3.1.1	Custom (POSIX-compliant) protocol optimized for performance

via - https://aws.amazon.com/fsx/when-to-choose-fsx/

Incorrect options:

Set up an Amazon Elastic File System (Amazon EFS) volume that uses EFS Intelligent-Tiering. Use AWS DataSync to migrate the data to the EFS volume

Set up an Amazon Elastic File System (Amazon EFS) volume that uses EFS Infrequent Access. Use AWS DataSync to migrate the data to the EFS volume

Amazon EFS is not supported on Windows instances. So, both these options are incorrect.

Set up an Amazon FSx for OpenZFS instance. Configure an FSx for OpenZFS file ystem on the root volume and migrate the data to the FSx for OpenZFS volume - Amazon FSx for OpenZFS is a fully managed file storage service that lets you launch, run, and scale fully managed file systems built on the open-source OpenZFS file system. FSx for OpenZFS makes it easy to migrate your on-premises file servers without changing your applications or how you manage data, and to build new high-performance, data-intensive applications on the cloud. FSx for OpenZFS is compatible with Windows, Linux, macOS clients. It supports NFS 3, 4.0, 4.1, 4.2 protocols, however, it does NOT support the SMB protocol.

References:

https://aws.amazon.com/fsx/netapp-ontap/

https://aws.amazon.com/fsx/when-to-choose-fsx/

https://aws.amazon.com/fsx/openzfs/faqs/

https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/AmazonEFS.html

Domain

Design Secure Architectures

Question 19Skipped

A company needs an Active Directory service to run directory-aware workloads in the AWS Cloud and it should also support configuring a trust relationship with any existing on-premises Microsoft Active Directory.

Which AWS Directory Service is the best fit for this requirement?

Correct answer

AWS Directory Service for Microsoft Active Directory (AWS Managed Microsoft AD)

Active Directory Connector

Simple Active Directory (Simple AD)

AWS Transit Gateway

Overall explanation

Correct option:

AWS Directory Service for Microsoft Active Directory (AWS Managed Microsoft AD)

AWS Directory Service lets you run Microsoft Active Directory (AD) as a managed service. AWS Directory Service for Microsoft Active Directory, also referred to as AWS Managed Microsoft AD, is powered by Windows Server 2012 R2. When you select and launch this directory type, it is created as a highly available pair of domain controllers connected to your virtual private cloud (VPC).

With AWS Managed Microsoft AD, you can run directory-aware workloads in the AWS Cloud, including Microsoft SharePoint and custom .NET and SQL Server-based applications. You can

also configure a trust relationship between AWS Managed Microsoft AD in the AWS Cloud and your existing on-premises Microsoft Active Directory, providing users and groups with access to resources in either domain, using single sign-on (SSO).

AWS Managed Microsoft AD is your best choice if you need actual Active Directory features to support AWS applications or Windows workloads, including Amazon Relational Database Service for Microsoft SQL Server. It's also best if you want a standalone AD in the AWS Cloud that supports Office 365 or you need an LDAP directory to support your Linux applications.

Incorrect options:

Active Directory Connector - AD Connector is a directory gateway with which you can redirect directory requests to your on-premises Microsoft Active Directory without caching any information in the cloud. AD Connector is your best choice when you want to use your existing on-premises directory with compatible AWS services.

Simple Active Directory (Simple AD) - Simple AD is a standalone directory in the cloud, where you create and manage user identities and manage access to applications. Simple AD provides a subset of the features offered by AWS Managed Microsoft AD. However, note that Simple AD does not support features such as multi-factor authentication (MFA), trust relationships with other domains, Active Directory Administrative Center, PowerShell support, Active Directory recycle bin, group managed service accounts, and schema extensions for POSIX and Microsoft applications.

AWS Transit Gateway - AWS Transit Gateway connects VPCs and on-premises networks through a central hub. Transit Gateway is not an Active Directory service.

References:

https://docs.aws.amazon.com/directoryservice/latest/admin-guide/what_is.html

https://docs.aws.amazon.com/directoryservice/latest/admin-guide/directory_simple_ad.html

https://docs.aws.amazon.com/directoryservice/latest/admin-guide/directory_ad_connector.html

Domain

Design Secure Architectures

Question 20Skipped

A healthcare company wants to run its applications on single-tenant hardware to meet compliance guidelines.

Which of the following is the MOST cost-effective way of isolating the Amazon EC2 instances to a single tenant?

Correct answer

Dedicated Instances

On-Demand Instances

Spot Instances

Dedicated Hosts

Overall explanation

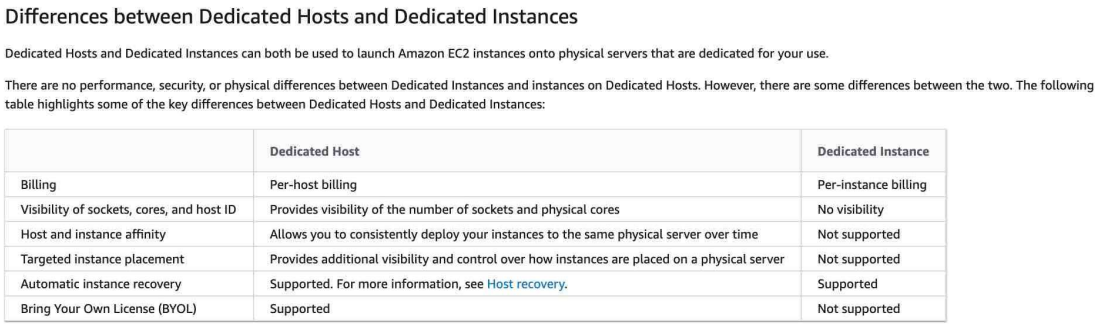
Correct option:

Dedicated Instances

Dedicated Instances are Amazon EC2 instances that run in a virtual private cloud (VPC) on hardware that's dedicated to a single customer. Dedicated Instances that belong to different AWS accounts are physically isolated at a hardware level, even if those accounts are linked to a single-payer account. However, Dedicated Instances may share hardware with other instances from the same AWS account that are not Dedicated Instances.

A Dedicated Host is also a physical server that's dedicated for your use. With a Dedicated Host, you have visibility and control over how instances are placed on the server.

Differences between Dedicated Hosts and Dedicated Instances:



Differences between Dedicated Hosts and Dedicated Instances

Dedicated Hosts and Dedicated Instances can both be used to launch Amazon EC2 instances onto physical servers that are dedicated for your use.

There are no performance, security, or physical differences between Dedicated Instances and instances on Dedicated Hosts. However, there are some differences between the two. The following table highlights some of the key differences between Dedicated Hosts and Dedicated Instances:

	Dedicated Host	Dedicated Instance
Billing	Per-host billing	Per-instance billing
Visibility of sockets, cores, and host ID	Provides visibility of the number of sockets and physical cores	No visibility
Host and instance affinity	Allows you to consistently deploy your instances to the same physical server over time	Not supported
Targeted instance placement	Provides additional visibility and control over how instances are placed on a physical server	Not supported
Automatic instance recovery	Supported. For more information, see Host recovery.	Supported
Bring Your Own License (BYOL)	Supported	Not supported

via - https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/dedicated-hosts-overview.html#dedicated-hosts-dedicated-instances

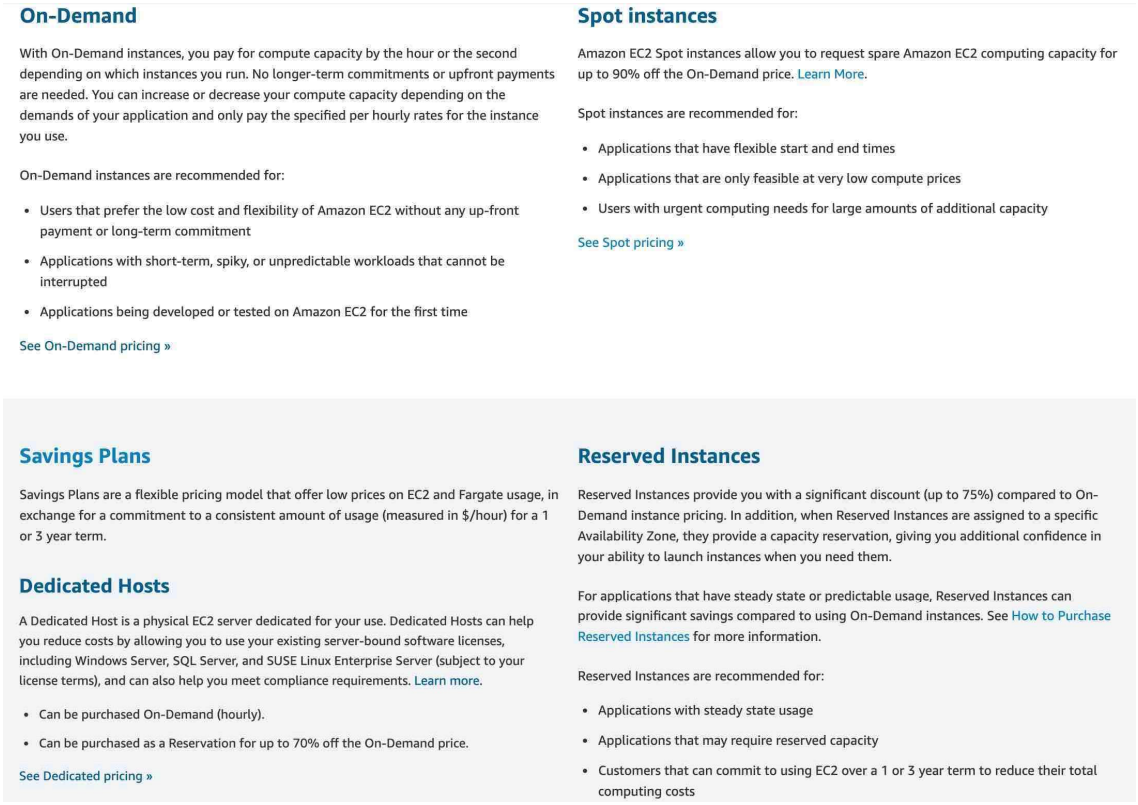
Incorrect options:

Spot Instances - A Spot Instance is an unused Amazon EC2 instance that is available for less than the On-Demand price. Your Spot Instance runs whenever capacity is available and the maximum price per hour for your request exceeds the Spot price. Any instance present with unused capacity will be allocated. Even though this is cost-effective, it does not fulfill the single-tenant hardware requirement of the client and hence is not the correct option.

Dedicated Hosts - An Amazon EC2 Dedicated Host is a physical server with Amazon EC2 instance capacity fully dedicated to your use. Dedicated Hosts allow you to use your existing software licenses on Amazon EC2 instances. With a Dedicated Host, you have visibility and control over how instances are placed on the server. This option is costlier than the Dedicated Instance and hence is not the right choice for the current requirement.

On-Demand Instances - With On-Demand Instances, you pay for the compute capacity by the second with no long-term commitments. You have full control over its lifecycle—you decide when to launch, stop, hibernate, start, reboot, or terminate it. Hardware isolation is not possible and on-demand has one of the costliest instance charges and hence is not the correct answer for current requirements.

High Level Overview of Amazon EC2 Instance Purchase Options:



via - https://aws.amazon.com/ec2/pricing/

References:

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

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

Domain

Design Secure Architectures

Question 21Skipped

A company uses a legacy on-premises reporting application that operates on gigabytes of .json files and represents years of data. The legacy application cannot handle the growing size of .json files. New .json files are added daily from various data sources to a central on-premises storage location. The company wants to continue to support the legacy application. The company has hired you as a solutions architect to build a solution that can manage ongoing data updates from your on-premises application to Amazon S3.

Which of the following solutions would you suggest to address the given requirement?

Set up AWS DataSync on-premises. Configure AWS DataSync to continuously replicate the .json files between the company's on-premises storage and the company's Amazon S3 bucket

Set up AWS DataSync on-premises. Configure AWS DataSync to continuously replicate the .json files between on-premises and Amazon Elastic File System (Amazon EFS). Enable replication from Amazon EFS to the company's Amazon S3 bucket

Set up an on-premises volume gateway. Configure data sources to write the .json files to the volume gateway. Point the legacy analytics application to the volume gateway. The volume gateway should replicate data to Amazon S3

Correct answer

Set up an on-premises file gateway. Configure data sources to write the .json files to the file gateway. Point the legacy analytics application to the file gateway. The file gateway should replicate the .json files to Amazon S3

Overall explanation

Correct option:

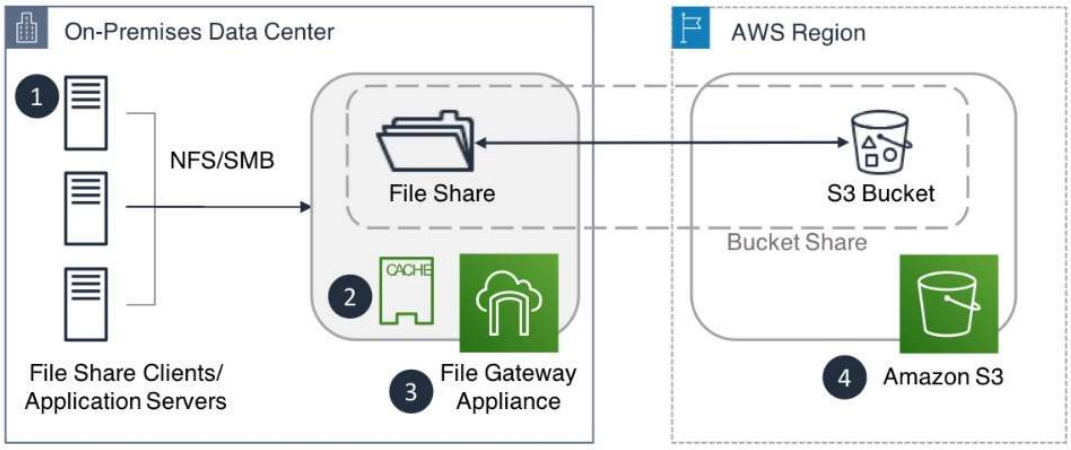
Set up an on-premises file gateway. Configure data sources to write the .json files to the file gateway. Point the legacy analytics application to the file gateway. The file gateway should replicate the .json files to Amazon S3

A file gateway provides a simple solution for presenting one or more Amazon S3 buckets and their objects as a mountable NFS or SMB file share to one or more clients on-premises.

The file gateway is deployed as a virtual machine in VMware ESXi or Microsoft Hyper-V environments on-premises, or in an Amazon Elastic Compute Cloud (Amazon EC2) instance in AWS. File gateway can also be deployed in data center and remote office locations on a Storage Gateway hardware appliance. When deployed, file gateway provides a seamless connection between on-premises NFS (v3.0 or v4.1) or SMB (v1 or v2) clients—typically applications—and Amazon S3 buckets hosted in a given AWS Region. The file gateway employs a local read/write cache to provide low-latency access to data for file share clients in the same local area network (LAN) as the file gateway.

A bucket share consists of a file share hosted from a file gateway across a single Amazon S3 bucket. The file gateway virtual machine appliance currently supports up to 10 bucket shares.

File Gateway Architecture:



via - https://docs.aws.amazon.com/whitepapers/latest/file-gateway-hybrid-cloud-storage-architectures/file-gateway-architecture.html

Incorrect options:

Set up an on-premises volume gateway. Configure data sources to write the .json files to the volume gateway. Point the legacy analytics application to the volume gateway. The volume gateway should replicate data to Amazon S3 - The Volume Gateway provides block storage to your on-premises applications using iSCSI connectivity. Data on the volumes is stored in Amazon S3 and you can take point in time copies of volumes that are stored in AWS as Amazon EBS snapshots. Volume Gateway is for block storage and not for file storage, so it is not the right option.

Set up AWS DataSync on-premises. Configure AWS DataSync to continuously replicate the .json files between the company's on-premises storage and the company's Amazon S3 bucket

Set up AWS DataSync on-premises. Configure AWS DataSync to continuously replicate the .json files between on-premises and Amazon Elastic File System (Amazon EFS). Enable replication from Amazon EFS to the company's Amazon S3 bucket

AWS recommends that you should use AWS DataSync to migrate existing data to Amazon S3, and subsequently use the File Gateway configuration of AWS Storage Gateway to retain access to the migrated data and for ongoing updates from your on-premises file-based applications. Therefore, both these options are incorrect, as they use DataSync for ongoing replication.

Reference:

https://docs.aws.amazon.com/whitepapers/latest/file-gateway-hybrid-cloud-storage-architectures/file-gateway-architecture.html

Domain

Design High-Performing Architectures

Question 22Skipped

You are deploying a critical monolith application that must be deployed on a single web server, as it hasn't been created to work in distributed mode. Still, you want to make sure your setup can automatically recover from the failure of an Availability Zone (AZ).

Which of the following options should be combined to form the MOST cost-efficient solution? (Select three)

Correct selection

Create an elastic IP address (EIP) and use the Amazon EC2 user-data script to attach it

Correct selection

Assign an Amazon EC2 Instance Role to perform the necessary API calls

Create an Application Load Balancer and a target group with the instance(s) of the Auto Scaling Group

Correct selection

Create an auto-scaling group that spans across 2 Availability Zones, which min=1, max=1, desired=1

Create a Spot Fleet request

Create an auto-scaling group that spans across 2 Availability Zones, which min=1, max=2, desired=2

Overall explanation

Correct options:

Create an auto-scaling group that spans across 2 Availability Zones, which min=1, max=1, desired=1

Amazon EC2 Auto Scaling helps you ensure that you have the correct number of Amazon EC2 instances available to handle the load for your application. You create collections of EC2 instances, called Auto Scaling groups. You can specify the minimum number of instances in each Auto Scaling group, and Amazon EC2 Auto Scaling ensures that your group never goes below this size.

So we have an Auto Scaling Group with desired=1, across two AZ, so that if an instance goes down, it is automatically recreated in another AZ. So this option is correct.

Create an elastic IP address (EIP) and use the Amazon EC2 user-data script to attach it

Application Load Balancer (ALB) operates at the request level (layer 7), routing traffic to targets – Amazon EC2 instances, containers, IP addresses, and Lambda functions based on the content of the request. Ideal for advanced load balancing of HTTP and HTTPS traffic, Application Load Balancer provides advanced request routing targeted at delivery of modern application architectures, including microservices and container-based applications.

An Elastic IP address is a static IPv4 address designed for dynamic cloud computing. An Elastic IP address is associated with your AWS account. With an Elastic IP address, you can mask the failure of an instance or software by rapidly remapping the address to another instance in your account.

Now, between the ALB and the Elastic IP. If we use an ALB, things will still work, but we will have to pay for the provisioned ALB which sends traffic to only one Amazon EC2 instance. Instead, to minimize costs, we must use an Elastic IP.

Assign an Amazon EC2 Instance Role to perform the necessary API calls

For that Elastic IP to be attached to our Amazon EC2 instance, we must use an EC2 user data script, and our Amazon EC2 instance must have the correct IAM permissions to perform the API call, so we need an Amazon EC2 instance role.

Incorrect options:

Create a Spot Fleet request - A Spot Instance is an unused Amazon EC2 instance that is available for less than the On-Demand price. Because Spot Instances enable you to request unused EC2 instances at steep discounts, you can lower your Amazon EC2 costs significantly. The hourly price for a Spot Instance is called a Spot price.

The Spot Fleet selects the Spot Instance pools that meet your needs and launches Spot Instances to meet the target capacity for the fleet. By default, Spot Fleets are set to maintain

target capacity by launching replacement instances after Spot Instances in the fleet are terminated.

Spot Fleets requests would not fit our purpose as we are looking at a critical application. Spot instances can be terminated. So this option is incorrect.

Create an auto-scaling group that spans across 2 Availability Zones, which min=1, max=2, desired=2 - An Auto Scaling Group with desired=2 would create two instances, and this won't work for us as our monolith application is not made to work with two instances as per the given use-case.

Create an Application Load Balancer and a target group with the instance(s) of the Auto Scaling Group - If we use an Application Load Balancer (ALB), things will still work, but we will have to pay for the provisioned ALB which sends traffic to only one Amazon EC2 instance. So this option is not correct.

References:

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-spot-instances.html

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/elastic-ip-addresses-eip.html

Domain

Design Resilient Architectures

Question 23Skipped

You are looking to build an index of your files in Amazon S3, using Amazon RDS PostgreSQL. To build this index, it is necessary to read the first 250 bytes of each object in Amazon S3, which contains some metadata about the content of the file itself. There are over 100,000 files in your S3 bucket, amounting to 50 terabytes of data.

How can you build this index efficiently?

Correct answer

Create an application that will traverse the S3 bucket, issue a Byte Range Fetch for the first 250 bytes, and store that information in Amazon RDS

Create an application that will traverse the Amazon S3 bucket, then use S3 Select Byte Range Fetch parameter to get the first 250 bytes, and store that information in Amazon RDS

Create an application that will traverse the Amazon S3 bucket, read all the files one by one, extract the first 250 bytes, and store that information in Amazon RDS

Use the Amazon RDS Import feature to load the data from Amazon S3 to PostgreSQL, and run a SQL query to build the index

Overall explanation

Correct option:

Create an application that will traverse the S3 bucket, issue a Byte Range Fetch for the first 250 bytes, and store that information in Amazon RDS

Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance.

Using the Range HTTP header in a GET Object request, you can fetch a byte-range from an object, transferring only the specified portion. You can use concurrent connections to Amazon S3 to fetch different byte ranges from within the same object. This helps you achieve higher aggregate throughput versus a single whole-object request. Fetching smaller ranges of a large object also allows your application to improve retry times when requests are interrupted.

A byte-range request is a perfect way to get the beginning of a file and ensuring we remain efficient during our scan of our Amazon S3 bucket. So this is the correct option.

Incorrect options:

Use the Amazon RDS Import feature to load the data from Amazon S3 to PostgreSQL, and run a SQL query to build the index - You cannot import data from Amazon S3 into Amazon RDS, so this option is incorrect.

Create an application that will traverse the Amazon S3 bucket, read all the files one by one, extract the first 250 bytes, and store that information in Amazon RDS - If you build an application that loads all the files from Amazon S3, that would work, but you would read 50TB of data and that may be very expensive and slow. So this option is incorrect.

Create an application that will traverse the Amazon S3 bucket, then use S3 Select Byte Range Fetch parameter to get the first 250 bytes, and store that information in Amazon RDS - Amazon S3 Select is a new Amazon S3 capability designed to pull out only the data you need from an object, which can dramatically improve the performance and reduce the cost of applications that need to access data in Amazon S3. You cannot use Byte Range Fetch parameter with S3 Select to traverse the Amazon S3 bucket and get the first bytes of a file. So this option is incorrect.

Exam Alert:

Please note that with Amazon S3 Select, you can scan a subset of an object by specifying a range of bytes to query using the ScanRange parameter. This capability lets you parallelize scanning the whole object by splitting the work into separate Amazon S3 Select requests for a series of non-overlapping scan ranges. Use the Amazon S3 Select ScanRange parameter and Start at (Byte) and End at (Byte).

Requests using scan ranges

With Amazon S3 Select, you can scan a subset of an object by specifying a range of bytes to query. This capability lets you parallelize scanning the whole object by splitting the work into separate Amazon S3 Select requests for a series of non-overlapping scan ranges. Scan ranges don't need to be aligned with record boundaries. An Amazon S3 Select scan range request runs across the byte range that you specify. A record that starts within the scan range specified but extends beyond the scan range will be processed by the query. For example; the following shows an Amazon S3 object containing a series of records in a line-delimited CSV format:

```
A,B
C,D
D,E
E,F
G,H
I,J
```

Use the Amazon S3 Select ScanRange parameter and *Start* at (Byte) 1 and *End* at (Byte) 4. So the scan range would start at "," and scan till the end of record starting at **"C"** and return the result **C, D** because that is the end of the record.

Amazon S3 Select scan range requests support Parquet, CSV (without quoted delimiters), and JSON objects (in LINES mode only). CSV and JSON objects must be uncompressed. For line-based CSV and JSON objects, when a scan range is specified as part of the Amazon S3 Select request, all records that start within the scan range are processed. For Parquet objects, all of the row groups that start within the scan range requested are processed.

Amazon S3 Select scan range requests are available to use on the Amazon S3 CLI, API and SDK. You can use the ScanRange parameter in the Amazon S3 Select request for this feature. For more information, see the Amazon S3 SELECT Object Content in the *Amazon Simple Storage Service API Reference*.

via - https://docs.aws.amazon.com/AmazonS3/latest/dev/selecting-content-from-objects.html

Reference:

https://docs.aws.amazon.com/AmazonS3/latest/dev/optimizing-performance-guidelines.html#optimizing-performance-guidelines-get-range

Domain

Design High-Performing Architectures

Question 24Skipped

The engineering team at an IT company is deploying an Online Transactional Processing (OLTP) application that needs to support relational queries. The application will have unpredictable spikes of usage that the team does not know in advance.

Which database would you recommend using?

Amazon DynamoDB with Provisioned Capacity and Auto Scaling

Amazon DynamoDB with On-Demand Capacity

Amazon ElastiCache

Correct answer

Amazon Aurora Serverless

Overall explanation

Correct option:

Amazon Aurora Serverless

Amazon Aurora Serverless is an on-demand, auto-scaling configuration for Amazon Aurora (MySQL-compatible and PostgreSQL-compatible editions), where the database will automatically start up, shut down, and scale capacity up or down based on your application's needs. It enables you to run your database in the cloud without managing any database instances. It's a simple, cost-effective option for infrequent, intermittent, or unpredictable workloads. The database design for an OLTP application fits the relational model, therefore you can infer an OLTP system as a Relational Database.

Amazon Aurora Serverless is the perfect way to create a database that can scale down to 0 servers, and scale up to many servers, as an OLTP database. So this is the correct option.

Incorrect options:

Amazon DynamoDB with Provisioned Capacity and Auto Scaling

Amazon DynamoDB with On-Demand Capacity

Amazon DynamoDB is a key-value and document database that delivers single-digit millisecond performance at any scale. It's a fully managed, multi-region, multi-master, durable database with built-in security, backup and restore, and in-memory caching for internet-scale applications.

Amazon DynamoDB is a NoSQL database and doesn't do relational queries, therefore it's a choice we have to eliminate, even though the two modes proposed here help us cope with an unpredictable amount of usage. So both these options are incorrect.

Amazon ElastiCache - Amazon ElastiCache allows you to seamlessly set up, run, and scale popular open-Source compatible in-memory data stores in the cloud. Build data-intensive apps or boost the performance of your existing databases by retrieving data from high throughput and low latency in-memory data stores. Amazon ElastiCache is a popular choice for real-time use cases like Caching, Session Stores, Gaming, Geospatial Services, Real-Time Analytics, and Queuing. Amazon Elasticache is used as a caching layer in front of relational databases. Amazon ElastiCache is a NoSQL database and doesn't facilitate relational queries, so this option is ruled out.

References:

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

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

Domain

Design Resilient Architectures

Question 25Skipped

A development team is looking for a solution that saves development time and deployment costs for an application that uses a high-throughput request-response message pattern.

Which of the following Amazon SQS queue types is the best fit to meet this requirement?

Amazon Simple Queue Service (Amazon SQS) FIFO queues

Correct answer

Amazon Simple Queue Service (Amazon SQS) temporary queues

Amazon Simple Queue Service (Amazon SQS) dead-letter queues

Amazon Simple Queue Service (Amazon SQS) delay queues

Overall explanation

Correct option:

Amazon Simple Queue Service (Amazon SQS) temporary queues

Temporary queues help you save development time and deployment costs when using common message patterns such as request-response. You can use the Temporary Queue Client to create high-throughput, cost-effective, application-managed temporary queues.

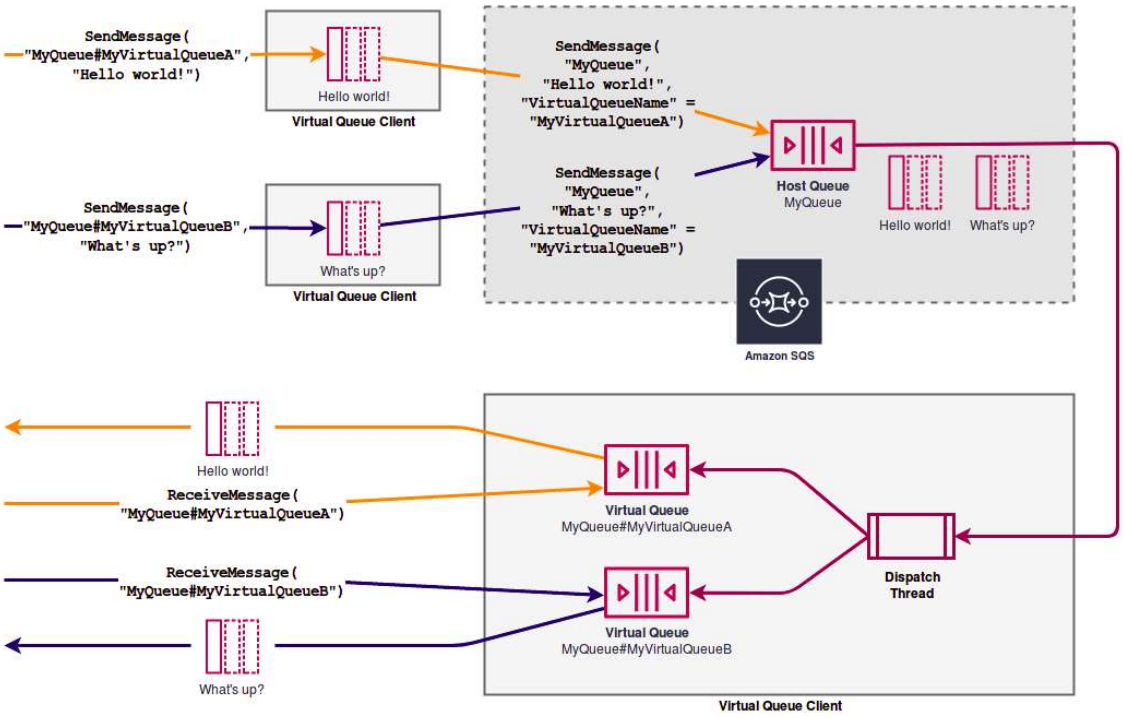
The client maps multiple temporary queues—application-managed queues created on demand for a particular process—onto a single Amazon SQS queue automatically. This allows your application to make fewer API calls and have a higher throughput when the traffic to each temporary queue is low. When a temporary queue is no longer in use, the client cleans up the temporary queue automatically, even if some processes that use the client aren't shut down cleanly.

The following are the benefits of temporary queues:

1. They serve as lightweight communication channels for specific threads or processes.
2. They can be created and deleted without incurring additional costs.
3. They are API-compatible with static (normal) Amazon SQS queues. This means that existing code that sends and receives messages can send messages to and receive messages from virtual queues.

To better support short-lived, lightweight messaging destinations, AWS recommends Amazon SQS Temporary Queue Client. This client makes it easy to create and delete many temporary messaging destinations without inflating your AWS bill. The key concept behind the client is the virtual queue. Virtual queues let you multiplex many low-traffic queues onto a single Amazon SQS queue. Creating a virtual queue only instantiates a local buffer to hold messages for consumers as they arrive; there is no API call to SQS and no costs associated with creating a virtual queue.

End-to-end process for sending messages through virtual queues:



via - https://aws.amazon.com/blogs/compute/simple-two-way-messaging-using-the-amazon-sqs-temporary-queue-client/

Incorrect options:

Amazon Simple Queue Service (Amazon SQS) dead-letter queues - Amazon SQS supports dead-letter queues, which other queues (source queues) can target for messages that can't be processed (consumed) successfully. Dead-letter queues are useful for debugging your application or messaging system because they let you isolate problematic messages to determine why their processing doesn't succeed. Amazon SQS does not create the dead-letter queue automatically. You must first create the queue before using it as a dead-letter queue.

Amazon Simple Queue Service (Amazon SQS) FIFO queues - Amazon SQS FIFO (First-In-First-Out) queues are designed to enhance messaging between applications when the order of operations and events is critical, or where duplicates can't be tolerated. FIFO queues also provide exactly-once processing but have a limited number of transactions per second (TPS).

Amazon Simple Queue Service (Amazon SQS) delay queues - Delay queues let you postpone the delivery of new messages to a queue for a number of seconds, for example, when your consumer application needs additional time to process messages. If you create a delay queue, any messages that you send to the queue remain invisible to consumers for the duration of the delay period. The default (minimum) delay for a queue is 0 seconds. The maximum is 15 minutes.

References:

https://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/sqs-temporary-queues.html

https://aws.amazon.com/blogs/compute/simple-two-way-messaging-using-the-amazon-sqs-temporary-queue-client/

Domain

Design High-Performing Architectures

Question 26Skipped

A company has noticed several provisioned throughput exceptions on its Amazon DynamoDB database due to major spikes in the writes to the database. The development team wants to decouple the application layer from the database layer and dedicate a worker process to writing the data to Amazon DynamoDB.

Which middleware do you recommend on using that can scale infinitely and meet these requirements in the most cost effective way?

Amazon DynamoDB DAX

Amazon Kinesis Data Streams

Correct answer

Amazon Simple Queue Service (Amazon SQS)

Amazon Simple Notification Service (Amazon SNS)

Overall explanation

Correct option:

Amazon Simple Queue Service (Amazon SQS)

Amazon Simple Queue Service (SQS) is a fully managed message queuing service that enables you to decouple and scale microservices, distributed systems, and serverless applications. Amazon SQS offers two types of message queues. Standard queues offer maximum throughput, best-effort ordering, and at-least-once delivery. Amazon SQS FIFO queues are designed to guarantee that messages are processed exactly once, in the exact order that they are sent.

Using Amazon SQS as a middleware will help us sustain the write throughput during write peaks and therefore this option is the best fit for the given use-case.

Incorrect options:

Amazon DynamoDB DAX - Amazon DynamoDB Accelerator (DAX) is a fully managed, highly available, in-memory cache for DynamoDB that delivers up to a 10x performance improvement – from milliseconds to microseconds – even at millions of requests per second. DAX does all the heavy lifting required to add in-memory acceleration to your DynamoDB tables, without requiring developers to manage cache invalidation, data population, or cluster management.

DAX is used for caching reads, not to help with writes. So this option is ruled out.

Amazon Kinesis Data Streams - Amazon Kinesis Data Streams (KDS) is a massively scalable and durable real-time data streaming service. KDS can continuously capture gigabytes of data per second from hundreds of thousands of sources such as website clickstreams, database event streams, financial transactions, social media feeds, IT logs, and location-tracking events. The throughput of an Amazon Kinesis data stream is designed to scale without limits via increasing the number of shards within a data stream. Kinesis is used to process consistent real-time data and does not scale as cost effectively as SQS to handle spikes in traffic.

Amazon Simple Notification Service (Amazon SNS) - Amazon Simple Notification Service (Amazon SNS) is a highly available, durable, secure, fully managed pub/sub messaging service that enables you to decouple microservices, distributed systems, and serverless applications. Amazon SNS won't keep our data if it cannot be delivered, so this option is incorrect.

References:

https://aws.amazon.com/sqs/

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

Domain

Design High-Performing Architectures

Question 27Skipped

The data engineering team at a company wants to analyze Amazon S3 storage access patterns to decide when to transition the right data to the right storage class.

Which of the following represents a correct option regarding the capabilities of Amazon S3 Analytics storage class analysis?

Storage class analysis only provides recommendations for Standard to Standard One-Zone IA classes

Storage class analysis only provides recommendations for Standard to Glacier Flexible Retrieval classes

Correct answer

Storage class analysis only provides recommendations for Standard to Standard IA classes

Storage class analysis only provides recommendations for Standard to Glacier Deep Archive classes

Overall explanation

Correct option:

Storage class analysis only provides recommendations for Standard to Standard IA classes

By using Amazon S3 analytics Storage Class Analysis you can analyze storage access patterns to help you decide when to transition the right data to the right storage class. This new Amazon S3 analytics feature observes data access patterns to help you determine when to transition less frequently accessed STANDARD storage to the STANDARD_IA (IA, for infrequent access) storage class.

Storage class analysis only provides recommendations for Standard to Standard IA classes.

After storage class analysis observes the infrequent access patterns of a filtered set of data over a period of time, you can use the analysis results to help you improve your lifecycle configurations. You can configure storage class analysis to analyze all the objects in a bucket. Or, you can configure filters to group objects together for analysis by common prefix (that is, objects that have names that begin with a common string), by object tags, or by both prefix and tags.

Incorrect options:

Storage class analysis only provides recommendations for Standard to Standard One-Zone IA classes

Storage class analysis only provides recommendations for Standard to Glacier Deep Archive classes

Storage class analysis only provides recommendations for Standard to Glacier Flexible Retrieval classes

These three options contradict the explanation provided above, so these options are incorrect.

References:

https://docs.aws.amazon.com/AmazonS3/latest/userguide/Welcome.html

https://docs.aws.amazon.com/AmazonS3/latest/userguide/analytics-storage-class.html

Domain

Design Cost-Optimized Architectures

Question 28Skipped

A systems administration team has a requirement to run certain custom scripts only once during the launch of the Amazon Elastic Compute Cloud (Amazon EC2) instances that host their application.

Which of the following represents the best way of configuring a solution for this requirement with minimal effort?

Run the custom scripts as instance metadata scripts on the Amazon EC2 instances

Correct answer

Run the custom scripts as user data scripts on the Amazon EC2 instances

Update Amazon EC2 instance configuration to ensure that the custom scripts, added as user data scripts, are run only during the boot process

Use AWS CLI to run the user data scripts only once while launching the instance

Overall explanation

Correct option:

Run the custom scripts as user data scripts on the Amazon EC2 instances

When you launch an instance in Amazon EC2, you have the option of passing user data to the instance that can be used to perform common automated configuration tasks and even run scripts after the instance starts. You can pass two types of user data to Amazon EC2: shell scripts and cloud-init directives.

By default, user data scripts and cloud-init directives run only during the boot cycle when you first launch an instance. Hence, no extra configuration is needed, apart from including the custom scripts in user data scripts.

Incorrect options:

Update Amazon EC2 instance configuration to ensure that the custom scripts, added as user data scripts, are run only during the boot process - You can update your configuration to ensure that your user data scripts and cloud-init directives run every time you restart your instance. By default, the scripts are run, only once during the boot process while first launching the instance.

Run the custom scripts as instance metadata scripts on the Amazon EC2 instances- Instance metadata is data about your instance that you can use to configure or manage the running instance. Metadata cannot be used to run custom scripts.

Use AWS CLI to run the user data scripts only once while launching the instance - This statement is incorrect and used only as a distractor.

References:

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/user-data.html

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-instance-metadata.html

Domain

Design Resilient Architectures

Question 29Skipped

A financial services company stores confidential data on an Amazon Simple Storage Service (S3) bucket. The compliance guidelines require that files be stored with server-side encryption. The encryption used must be Advanced Encryption Standard (AES-256) and the company does not want to manage the encryption keys.

Which of the following options represents the most cost-optimal solution for the given use case?

Server-side encryption with AWS KMS keys (SSE-KMS)

Server-side encryption with customer-provided keys (SSE-C)

Correct answer

Server-side encryption with Amazon S3 managed keys (SSE-S3)

Client Side Encryption

Overall explanation

Correct option:

Server-side encryption with Amazon S3 managed keys (SSE-S3)

Using Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3), each object is encrypted with a unique key employing strong multi-factor encryption. As an additional safeguard, it encrypts the key itself with a master key that it regularly rotates. Amazon S3 server-side encryption uses one of the strongest block ciphers available, 256-bit Advanced Encryption Standard (AES-256), to encrypt your data. There are no additional fees for using server-side encryption with Amazon S3-managed keys (SSE-S3).

Incorrect options:

Server-side encryption with customer-provided keys (SSE-C) - You manage the encryption keys and Amazon S3 manages the encryption as it writes to disks and decryption when you access your objects.

Client Side Encryption - You can encrypt data client-side and upload the encrypted data to Amazon S3. In this case, you manage the encryption process, the encryption keys, and related tools.

Server-side encryption with AWS KMS keys (SSE-KMS) - Similar to SSE-S3 and also provides you with an audit trail of when your key was used and by whom. Additionally, you have the option to create and manage encryption keys yourself. Although SSE-KMS provides an option where AWS manages the encryption key on your behalf, however, this entails a usage fee for the KMS key. So this option is not the best fit for the given use case.

Reference:

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

Domain

Design Secure Architectures

Question 30Skipped

A team has around 200 users, each of these having an IAM user account in AWS. Currently, they all have read access to an Amazon S3 bucket. The team wants 50 among them to have write and read access to the buckets.

How can you provide these users access in the least possible time, with minimal changes?

Create a policy and assign it manually to the 50 users

Correct answer

Create a group, attach the policy to the group and place the users in the group

Create an AWS Multi-Factor Authentication (AWS MFA) user with read / write access and link 50 IAM with AWS MFA

Update the Amazon S3 bucket policy

Overall explanation

Correct option:

Create a group, attach the policy to the group and place the users in the group

An IAM group is a collection of IAM users. You can use groups to specify permissions for a collection of users, which can make those permissions easier to manage for those users. For example, you could have a group called Admins and give that group the types of permissions that administrators typically need. Any user in that group automatically has the permissions that are assigned to the group. If a new user joins your organization and should have administrator privileges, you can assign the appropriate permissions by adding the user to that group.

Here creating a group, assigning users to that group and attaching policies to that group is the best way.

Incorrect options:

Update the Amazon S3 bucket policy - Updating the Amazon S3 bucket policy could work but would not scale, as the size of the S3 bucket policy is limited (Bucket policies are limited to 20 KB in size).

Create a policy and assign it manually to the 50 users - An IAM user is an entity that you create in AWS. The IAM user represents the person or service who uses the IAM user to interact with AWS. Primary use for IAM users is to give people the ability to sign in to the AWS Management Console for interactive tasks and to make programmatic requests to AWS services using the API or CLI. A user in AWS consists of a name, a password to sign in to the AWS Management Console, and up to two access keys that can be used with the API or CLI.

A policy is an object in AWS that, when associated with an identity or resource, defines their 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.

Identity-based policies – Attach managed and inline policies to IAM identities (users, groups to which users belong, or roles). Identity-based policies grant permissions to an identity.

Resource-based policies – Attach inline policies to resources. The most common examples of resource-based policies are Amazon S3 bucket policies and IAM role trust policies. Resource-based policies grant permissions to the principal that is specified in the policy. Principals can be in the same account as the resource or in other accounts.

Creating a policy and assigning it manually to users would work but would be hard to scale and manage.

Create an AWS Multi-Factor Authentication (AWS MFA) user with read / write access and link 50 IAM with AWS MFA - AWS MFA adds extra security because it requires users to provide unique authentication from an AWS supported MFA mechanism in addition to their regular sign-in credentials when they access AWS websites or services. AWS MFA cannot help in terms of granting read/write access to only 50 of the IAM users.

References:

https://docs.aws.amazon.com/IAM/latest/UserGuide/id.html

https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies.html

Domain

Design Secure Architectures

Question 31Skipped

The DevOps team at a major financial services company uses Multi-Availability Zone (Multi-AZ) deployment for its MySQL Amazon RDS database in order to automate its database replication and augment data durability. The DevOps team has scheduled a maintenance window for a database engine level upgrade for the coming weekend.

Which of the following is the correct outcome during the maintenance window?

Any database engine level upgrade for an Amazon RDS database instance with Multi-AZ deployment triggers the standby database instance to be upgraded which is then followed by the upgrade of the primary database instance. This does not cause any downtime for the duration of the upgrade

Correct answer

Any database engine level upgrade for an Amazon RDS database instance with Multi-AZ deployment triggers both the primary and standby database instances to be upgraded at the same time. This causes downtime until the upgrade is complete

Any database engine level upgrade for an Amazon RDS database instance with Multi-AZ deployment triggers the primary database instance to be upgraded which is then followed by the upgrade of the standby database instance. This does not cause any downtime for the duration of the upgrade

Any database engine level upgrade for an Amazon RDS database instance with Multi-AZ deployment triggers both the primary and standby database instances to be upgraded at the same time. However, this does not cause any downtime until the upgrade is complete

Overall explanation

Correct option:

Any database engine level upgrade for an Amazon RDS database instance with Multi-AZ deployment triggers both the primary and standby database instances to be upgraded at the same time. This causes downtime until the upgrade is complete

Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while automating time-consuming administration tasks such as hardware provisioning, database setup, patching, and backups.

Upgrades to the database engine level require downtime. Even if your Amazon RDS DB instance uses a Multi-AZ deployment, both the primary and standby DB instances are upgraded at the same time. This causes downtime until the upgrade is complete, and the duration of the downtime varies based on the size of your database instance.

Amazon RDS DB Engine Maintenance:

How do I minimize downtime during required Amazon RDS maintenance?

Last updated: 2020-03-13

I received a maintenance notification that says one of my Amazon Relational Database Service (Amazon RDS) DB instances requires maintenance. What are some strategies that I can use to minimize downtime?

Resolution

Occasionally, AWS performs maintenance to the hardware, operating system (OS), or database engine version for a DB instance or cluster. For more information, see Maintaining a DB Instance and Upgrading a DB Instance Engine Version.

For information about pending maintenance events for your Amazon RDS DB instances, check the **Events** pane of the Amazon RDS console. Then, check for engine-specific maintenance events. You can run describe-pending-maintenance-actions using the AWS Command Line Interface (AWS CLI) or the Amazon RDS API for DescribeDBInstances. You can also check Amazon RDS Recommendations for **Pending maintenance available**.

Hardware maintenance

Before maintenance is scheduled, you receive an email notification about scheduled hardware maintenance windows that includes the time of the maintenance and the Availability Zones that are affected. During hardware maintenance, Single-AZ deployments are unavailable for a few minutes. Multi-AZ deployments are unavailable for the time it takes the instance to failover (usually about 60 seconds) if the Availability Zone is affected by the maintenance. If only the secondary Availability Zone is affected, then there is no failover or downtime.

OS maintenance

After OS maintenance is scheduled for the next maintenance window, maintenance can be postponed by adjusting your preferred maintenance window. Maintenance can also be deferred by choosing **Defer Upgrade** from the **Actions** dropdown menu. To minimize downtime, modify the Amazon RDS DB instance to a Multi-AZ deployment. For Multi-AZ deployments, OS maintenance is applied to the secondary instance first, then the instance fails over, and then the primary instance is updated. The downtime is during failover. For more information, see Maintenance for Multi-AZ Deployments.

DB engine maintenance

Upgrades to the database engine level require downtime. Even if your RDS DB instance uses a Multi-AZ deployment, both the primary and standby DB instances are upgraded at the same time. This causes downtime until the upgrade is complete, and the duration of the downtime varies based on the size of your DB instance. For more information, see the section for your DB engine in Upgrading a DB Instance Engine Version.

via - https://aws.amazon.com/premiumsupport/knowledge-center/rds-required-maintenance/

Incorrect options:

Any database engine level upgrade for an Amazon RDS database instance with Multi-AZ deployment triggers both the primary and standby database instances to be upgraded at the same time. However, this does not cause any downtime until the upgrade is complete - For Amazon RDS database engine level upgrade, primary and standby database instances are upgraded at the same time and it causes downtime until the upgrade is complete, hence this option is incorrect.

Any database engine level upgrade for an Amazon RDS database instance with Multi-AZ deployment triggers the standby database instance to be upgraded which is then followed by the upgrade of the primary database instance. This does not cause any downtime for the duration of the upgrade - For Amazon RDS database engine level upgrade, primary and standby database instances are upgraded at the same time and it causes downtime until the upgrade is complete, hence this option is incorrect.

Any database engine level upgrade for an Amazon RDS database instance with Multi-AZ deployment triggers the primary database instance to be upgraded which is then followed by the upgrade of the standby database instance. This does not cause any downtime for the duration of the upgrade - For Amazon RDS database engine level upgrade, primary and standby database instances are upgraded at the same time and it causes downtime until the upgrade is complete, hence this option is incorrect.

Reference:

https://aws.amazon.com/premiumsupport/knowledge-center/rds-required-maintenance/

Domain

Design Resilient Architectures

Question 32Skipped

A startup uses a fleet of Amazon EC2 servers to manage its CRM application. These Amazon EC2 servers are behind Elastic Load Balancing (ELB). Which of the following configurations are NOT allowed for Elastic Load Balancing?

Use the Elastic Load Balancing to distribute traffic for four Amazon EC2 instances. All the four instances are deployed in Availability Zone A of us-east-1 region

Use the Elastic Load Balancing to distribute traffic for four Amazon EC2 instances. All the four instances are deployed in Availability Zone B of us-west-1 region

Use the Elastic Load Balancing to distribute traffic for four Amazon EC2 instances. All the four instances are deployed across two Availability Zones of us-east-1 region

Correct answer

Use the Elastic Load Balancing to distribute traffic for four Amazon EC2 instances. Two of these instances are deployed in Availability Zone A of us-east-1 region and the other two instances are deployed in Availability Zone B of us-west-1 region

Overall explanation

Correct option:

Use the Elastic Load Balancing to distribute traffic for four Amazon EC2 instances. Two of these instances are deployed in Availability Zone A of us-east-1 region and the other two instances are deployed in Availability Zone B of us-west-1 region

Elastic Load Balancer automatically distributes incoming traffic across multiple targets – Amazon EC2 instances, containers, IP addresses, and Lambda functions – in multiple Availability Zones and ensures only healthy targets receive traffic. ELB cannot distribute incoming traffic for targets deployed in different regions. This configuration is NOT allowed for the Elastic Load Balancer and therefore this is the correct option.

Incorrect options:

Use the Elastic Load Balancing to distribute traffic for four Amazon EC2 instances. All the four instances are deployed across two Availability Zones of us-east-1 region

Use the Elastic Load Balancing to distribute traffic for four Amazon EC2 instances. All the four instances are deployed in Availability Zone A of us-east-1 region

Use the Elastic Load Balancing to distribute traffic for four Amazon EC2 instances. All the four instances are deployed in Availability Zone B of us-west-1 region

These three options are valid configurations for the Elastic Load Balancing to distribute traffic (either within an Availability Zone or between two Availability Zones).

Reference:

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

Domain

Design Resilient Architectures

Question 33Skipped

A company has media files that need to be shared internally. Users are first authenticated using Active Directory and then they access files on a Microsoft Windows platform. The engineering manager wants to keep the same user permissions but wants the company to migrate the storage layer to AWS Cloud as the company is reaching its storage capacity limit on the on-premises infrastructure.

What should a solutions architect recommend to meet this requirement?

Provision Amazon EC2 with Windows OS, attach multiple Amazon EBS volumes, and move all media files

Correct answer

Set up Amazon FSx for Windows File Server and move all the media files

Set up Amazon EFS and move all media files

Create a corporate Amazon S3 bucket and move all media files

Overall explanation

Correct option:

Set up Amazon FSx for Windows File Server and move all the media files

Amazon FSx for Windows File Server provides fully managed, highly reliable, and scalable file storage that is accessible over the industry-standard Server Message Block (SMB) protocol. It is built on Windows Server, delivering a wide range of administrative features such as user quotas, end-user file restore, and Microsoft Active Directory (AD) integration. To support a wide spectrum of workloads, Amazon FSx provides high levels of throughput and IOPS and consistent sub-millisecond latencies.

Amazon FSx file storage is accessible from Windows, Linux, and macOS compute instances and devices running on AWS or on-premises. Thousands of compute instances and devices can access a file system concurrently. Amazon FSx for Windows File Server supports Microsoft Active Directory (AD) integration so the same user permissions and access credentials can be used to access the files on FSx Windows File Server.

Incorrect options:

Create a corporate Amazon S3 bucket and move all media files - Amazon S3 is object-based storage and it does not support file storage. Hence S3 is not the correct option.

Set up Amazon EFS and move all media files - Amazon EFS provides scalable file storage for use with Amazon EC2. You can use an EFS file system as a common data source for workloads and applications running on multiple instances. EFS is not compatible with the Windows platform, so this option is ruled out.

Provision Amazon EC2 with Windows OS, attach multiple Amazon EBS volumes, and move all media files - Multi-attach Amazon EBS volumes are supported only for Nitro EC2 instances which are Linux-based. So this option is ruled out.

Reference:

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

Domain

Design Secure Architectures

Question 34Skipped

A big data analytics company is looking to archive the on-premises data into a POSIX compliant file storage system on AWS Cloud. The archived data would be accessed for just about a week in a year.

As a solutions architect, which of the following AWS services would you recommend as the MOST cost-optimal solution?

Amazon S3 Standard-IA

Amazon EFS Standard

Amazon S3 Standard

Correct answer

Amazon EFS Infrequent Access

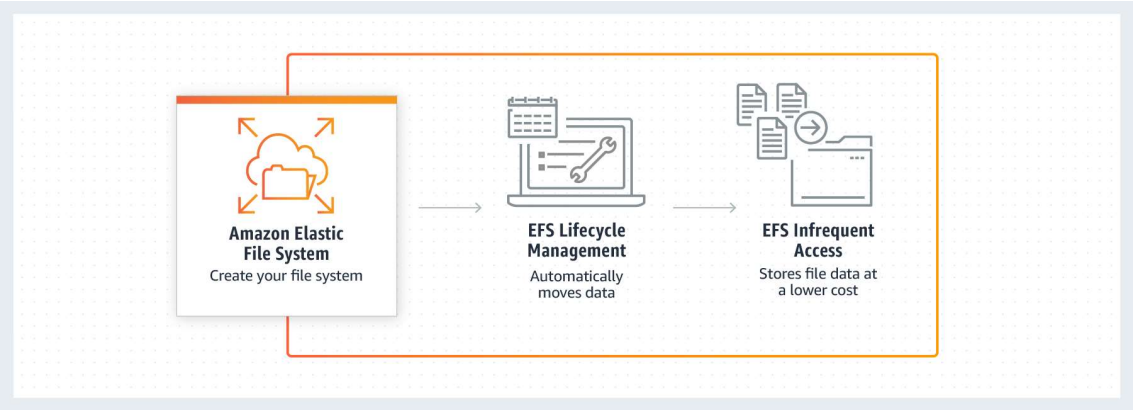
Overall explanation

Correct option:

Amazon EFS Infrequent Access

Amazon Elastic File System (Amazon EFS) provides a simple, scalable, fully managed, elastic, NFS file system for use with AWS Cloud services and on-premises resources. Amazon EFS Infrequent Access (EFS IA) is a storage class that provides price/performance that is cost-optimized for files not accessed every day, with storage prices up to 92% lower compared to Amazon EFS Standard. The EFS IA storage class costs only $0.025/GB-month. To get started with EFS IA, simply enable EFS Lifecycle Management for your file system by selecting a lifecycle policy that matches your needs.

How Amazon EFS Infrequent Access Works:



via - https://aws.amazon.com/efs/features/infrequent-access/

Incorrect options:

Amazon EFS Standard - Amazon EFS Infrequent Access is more cost-effective than EFS Standard for the given use-case, therefore this option is incorrect.

Amazon S3 Standard

Amazon S3 Standard-IA

Both these options are object-based storage, whereas the given use-case requires a POSIX compliant file storage solution. Hence these two options are incorrect.

Reference:

https://aws.amazon.com/efs/features/infrequent-access/

Domain

Design Cost-Optimized Architectures

Question 35Skipped

Your company has created a data warehouse using Amazon Redshift that is used to analyze data from Amazon S3. From the usage pattern, you have detected that after 30 days, the data is rarely queried in Amazon Redshift and it's not "hot data" anymore. You would like to preserve the SQL querying capability on your data and get the queries started immediately. Also, you want to adopt a pricing model that allows you to save the maximum amount of cost on Amazon Redshift.

What do you recommend? (Select two)

Correct selection

Analyze the cold data with Amazon Athena

Create a smaller Amazon Redshift Cluster with the cold data

Move the data to Amazon S3 Glacier Deep Archive after 30 days

Correct selection

Move the data to Amazon S3 Standard IA after 30 days

Migrate the Amazon Redshift underlying storage to Amazon S3 IA

Overall explanation

Correct options:

Move the data to Amazon S3 Standard IA after 30 days

Amazon S3 Standard-IA is for data that is accessed less frequently but requires rapid access when needed. Amazon S3 Standard-IA offers high durability, high throughput, and low latency of S3 Standard, with a low per GB storage price and per GB retrieval fee. This combination of low cost and high performance makes S3 Standard-IA ideal for long-term storage, backups, and as a data store for disaster recovery files. The minimum storage duration charge is 30 days.

Analyze the cold data with Amazon Athena

Amazon Athena is an interactive query service that makes it easy to analyze data directly in Amazon S3 using standard SQL. Athena is serverless, so there is no infrastructure to set up or manage, and customers pay only for the queries they run. You can use Amazon Athena to process logs, perform ad-hoc analysis, and run interactive queries.

Moving the data to Amazon S3 glacier will prevent us from being able to query it. Therefore, we should migrate the data to Amazon S3 Standard IA and use Amazon Athena to analyze the cold data.

Incorrect options:

Migrate the Amazon Redshift underlying storage to Amazon S3 IA - Amazon Redshift is a fully-managed petabyte-scale cloud-based data warehouse product designed for large scale data set storage and analysis. An Amazon Redshift data warehouse is a collection of computing resources called nodes, which are organized into a group called a cluster. Each cluster runs an Amazon Redshift engine and contains one or more databases. An Amazon Redshift cluster consists of nodes. Each cluster has a leader node and one or more compute nodes. The leader node receives queries from client applications, parses the queries, and develops query execution plans. The leader node then coordinates the parallel execution of these plans with the compute nodes and aggregates the intermediate results from these nodes. It then finally returns the results to the client applications.

Redshift's internal storage does not have "tiers" of storage classes like Amazon S3, so this option is also ruled out.

Create a smaller Amazon Redshift Cluster with the cold data - Creating a smaller cluster with the cold data would not decrease the storage cost of Amazon Redshift, which will only increase with time as we keep on creating data. Therefore this option is ruled out.

Move the data to Amazon S3 Glacier Deep Archive after 30 days - Amazon S3 Glacier Deep Archive (GDA) is a secure, durable, and extremely low-cost Amazon S3 cloud storage class for data archiving and long-term backup. It is designed to deliver 99.999999999% durability, and provide comprehensive security and compliance capabilities that can help meet even the most stringent regulatory requirements. GDA has a first-byte latency of several hours, so this option is incorrect.

References:

https://aws.amazon.com/athena/

https://docs.aws.amazon.com/redshift/latest/mgmt/working-with-clusters.html

Domain

Design Cost-Optimized Architectures

Question 36Skipped

A company maintains its business-critical customer data on an on-premises system in an encrypted format. Over the years, the company has transitioned from using a single encryption key to multiple encryption keys by dividing the data into logical chunks. With the decision to move all the data to an Amazon S3 bucket, the company is now looking for a technique to encrypt each file with a different encryption key to provide maximum security to the migrated on-premises data.

How will you implement this requirement without adding the overhead of splitting the data into logical groups?

Use Multi-Region keys for client-side encryption in the AWS S3 Encryption Client to generate unique keys for each file of data

Configure a single Amazon S3 bucket to hold all data. Use server-side encryption with AWS KMS (SSE-KMS) and use encryption context to generate a different key for each file/object that you store in the S3 bucket

Correct answer

Configure a single Amazon S3 bucket to hold all data. Use server-side encryption with Amazon S3 managed keys (SSE-S3) to encrypt the data

Store the logically divided data into different Amazon S3 buckets. Use server-side encryption with Amazon S3 managed keys (SSE-S3) to encrypt the data

Overall explanation

Correct option:

Configure a single Amazon S3 bucket to hold all data. Use server-side encryption with Amazon S3 managed keys (SSE-S3) to encrypt the data

Server-side encryption is the encryption of data at its destination by the application or service that receives it. Amazon S3 encrypts your data at the object level as it writes it to disks in its data centers and decrypts it for you when you access it. When you use server-side encryption with Amazon S3 managed keys (SSE-S3), each object is encrypted with a unique key. As an additional safeguard, it encrypts the key itself with a root key that it regularly rotates.

Note: Amazon S3 now applies server-side encryption with Amazon S3 managed keys (SSE-S3) as the base level of encryption for every bucket in Amazon S3. Starting January 5, 2023, all new object uploads to Amazon S3 will be automatically encrypted at no additional cost and with no impact on performance.

Incorrect options:

Store the logically divided data into different Amazon S3 buckets. Use server-side encryption with Amazon S3 managed keys (SSE-S3) to encrypt the data - Server-side encryption with Amazon S3 managed keys (SSE-S3) is the easiest way to implement the given requirement, as there is no additional overhead of splitting data. Multiple S3 buckets are redundant for this requirement.

Use Multi-Region keys for client-side encryption in the AWS S3 Encryption Client to generate unique keys for each file of data - Server-side encryption is the encryption of data at its destination by the application or service that receives it. The requirement is about server-side encryption and not about client-side encryption, hence this choice is incorrect.

Configure a single Amazon S3 bucket to hold all data. Use server-side encryption with AWS KMS (SSE-KMS) and use encryption context to generate a different key for each file/object that you store in the S3 bucket - An encryption context is a set of key-value pairs that contain additional contextual information about the data. When an encryption context is specified for an encryption operation, Amazon S3 must specify the same encryption context for the decryption operation. The encryption context offers another level of security for the encryption key. However, it is not useful for generating unique keys.

References:

https://docs.aws.amazon.com/AmazonS3/latest/userguide/serv-side-encryption.html

https://docs.aws.amazon.com/AmazonS3/latest/userguide/UsingKMSEncryption.html

Domain

Design Secure Architectures

Question 37Skipped

An e-commerce website is migrating towards a microservices-based approach for their website and plans to expose their website from the same load balancer, linked to different target groups with different URLs: checkout.mycorp.com, www.mycorp.com, mycorp.com/products, and mycorp.com/orders. The website would like to use Amazon ECS on the backend to manage these microservices and possibly host the same container of the application multiple times on the same Amazon EC2 instance.

Which feature can help you achieve this with minimal effort?

Network Load Balancer + dynamic port mapping

Classic Load Balancer + dynamic port mapping

Correct answer

Application Load Balancer + dynamic port mapping

Application Load Balancer + Reverse Proxy running as a Docker daemon on each Amazon ECS host

Overall explanation

Correct option:

Application Load Balancer + dynamic port mapping

Application Load Balancer can automatically distribute incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, IP addresses, and AWS Lambda functions. It can handle the varying load of your application traffic in a single Availability Zone or across multiple Availability Zones (AZs).

Dynamic port mapping with an Application Load Balancer makes it easier to run multiple tasks on the same Amazon ECS service on an Amazon ECS cluster.

Incorrect option:

Application Load Balancer + Reverse Proxy running as a Docker daemon on each Amazon ECS host - Dynamic Port Mapping is available for the Application Load Balancer. A reverse proxy solution would work but would be too much work to manage. Here the Application Load Balancer has a feature that provides a direct dynamic port mapping feature and integration with the Amazon ECS service so we will leverage that.

Classic Load Balancer + dynamic port mapping - Classic Load Balancer provides basic load balancing across multiple Amazon EC2 instances and operates at both the request level and connection level. Classic Load Balancer is intended for applications that were built within the Amazon EC2-Classic network.

With the Classic Load Balancer, you must statically map port numbers on a container instance. The Classic Load Balancer does not allow you to run multiple copies of a task on the same instance because of the ports conflict. An Application Load Balancer uses dynamic port mapping so that you can run multiple tasks from a single service on the same container instance.

Network Load Balancer + dynamic port mapping - Network Load Balancer is best suited for use-cases involving low latency and high throughput workloads that involve scaling to millions of requests per second. Network Load Balancer operates at the connection level (Layer 4), routing connections to targets - Amazon EC2 instances, microservices, and containers – within Amazon Virtual Private Cloud (Amazon VPC) based on IP protocol data.

References:

https://aws.amazon.com/premiumsupport/knowledge-center/dynamic-port-mapping-ecs/

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

Domain

Design Resilient Architectures

Question 38Skipped

A software engineering intern at a company is documenting the features offered by Amazon EC2 Spot instances and Spot fleets.

Can you help the intern by selecting the correct options that identify the key characteristics of these two types of Spot entities? (Select two)

Correct selection

A Spot fleet can consist of a set of Spot Instances and optionally On-Demand Instances that are launched to meet your target capacity

Spot fleets are spare EC2 capacity that can save you up 90% off of On-Demand prices. Spot fleets are usually interrupted by Amazon EC2 for capacity requirements with a 2-minute notification

Spot fleets allow you to request Amazon EC2 Spot instances for 1 to 6 hours at a time to avoid being interrupted

Correct selection

Spot instances are spare Amazon EC2 capacity that can save you up 90% off of On-Demand prices. Spot instances can be interrupted by Amazon EC2 for capacity requirements with a 2-minute notification

A Spot fleet can only consist of a set of Spot Instances that are launched to meet your target capacity

Overall explanation

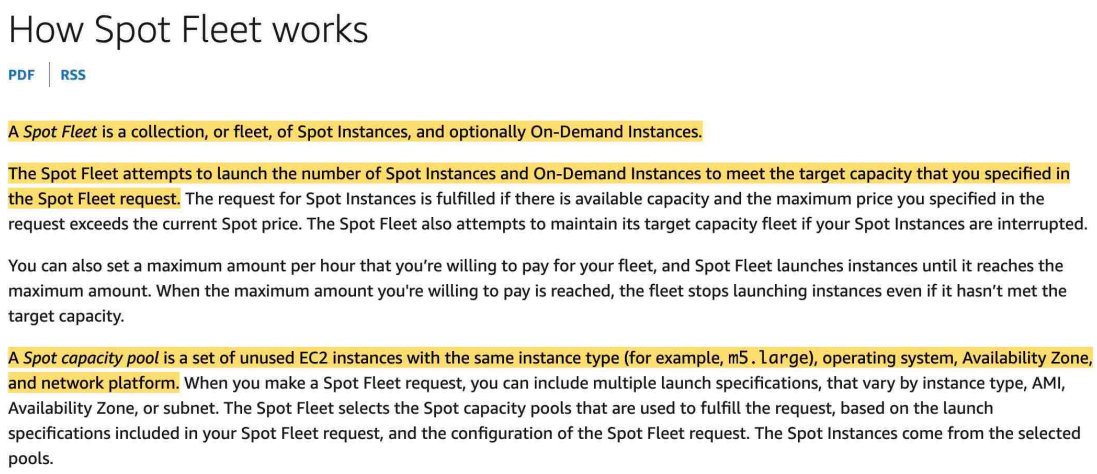
Correct options:

Spot instances are spare Amazon EC2 capacity that can save you up 90% off of On-Demand prices. Spot instances can be interrupted by Amazon EC2 for capacity requirements with a 2-minute notification

Spot instances are spare Amazon EC2 capacity that can save you up 90% off of On-Demand prices that Amazon Web Services can interrupt with a 2-minute notification. Because Spot Instances enable you to request unused EC2 instances at steep discounts, you can lower your Amazon EC2 costs significantly. Spot Instances are a cost-effective choice if you can be flexible about when your applications run and if your applications can be interrupted.

A Spot fleet can consist of a set of Spot Instances and optionally On-Demand Instances that are launched to meet your target capacity

A Spot fleet is a collection, or fleet, of Spot Instances, and optionally On-Demand Instances. The Spot fleet attempts to launch the number of Spot Instances and On-Demand Instances to meet the target capacity that you specified in the Spot fleet request. A Spot fleet allows you to automatically request and manage multiple Spot instances that provide the lowest price per unit of capacity for your cluster or application, like a batch processing job, a Hadoop workflow, or an HPC grid computing job.



How Spot Fleet works

PDF | RSS

A *Spot Fleet* is a collection, or fleet, of Spot Instances, and optionally On-Demand Instances.

The Spot Fleet attempts to launch the number of Spot Instances and On-Demand Instances to meet the target capacity that you specified in the Spot Fleet request. The request for Spot Instances is fulfilled if there is available capacity and the maximum price you specified in the request exceeds the current Spot price. The Spot Fleet also attempts to maintain its target capacity fleet if your Spot Instances are interrupted.

You can also set a maximum amount per hour that you're willing to pay for your fleet, and Spot Fleet launches instances until it reaches the maximum amount. When the maximum amount you're willing to pay is reached, the fleet stops launching instances even if it hasn't met the target capacity.

A *Spot capacity pool* is a set of unused EC2 instances with the same instance type (for example, `m5.large`), operating system, Availability Zone, and network platform. When you make a Spot Fleet request, you can include multiple launch specifications, that vary by instance type, AMI, Availability Zone, or subnet. The Spot Fleet selects the Spot capacity pools that are used to fulfill the request, based on the launch specifications included in your Spot Fleet request, and the configuration of the Spot Fleet request. The Spot Instances come from the selected pools.

via - https://docs.amazonaws.cn/en_us/AWSEC2/latest/UserGuide/how-spot-fleet-works.html

Incorrect options:

A Spot fleet can only consist of a set of Spot Instances that are launched to meet your target capacity

Spot fleets are spare EC2 capacity that can save you up 90% off of On-Demand prices. Spot fleets are usually interrupted by Amazon EC2 for capacity requirements with a 2-minute notification

These two options contradict the explanation provided above, so these options are incorrect.

Spot fleets allow you to request Amazon EC2 Spot instances for 1 to 6 hours at a time to avoid being interrupted - You could use Spot blocks (now deprecated) to request Amazon EC2 Spot instances for 1 to 6 hours to avoid being interrupted. So, Spot fleets cannot be used for this purpose.

References:

https://www.amazonaws.cn/en/ec2/spot-instances/faqs/

https://docs.amazonaws.cn/en_us/AWSEC2/latest/UserGuide/how-spot-fleet-works.html

Domain

Design Cost-Optimized Architectures

Question 39Skipped

A company has multiple Amazon EC2 instances operating in a private subnet which is part of a custom VPC. These instances are running an image processing application that needs to access images stored on Amazon S3. Once each image is processed, the status of the corresponding record needs to be marked as completed in a Amazon DynamoDB table.

How would you go about providing private access to these AWS resources which are not part of this custom VPC?

Create a gateway endpoint for Amazon DynamoDB and add it as a target in the route table of the custom VPC. Create an Origin Access Identity for Amazon S3 and then connect to the S3 service using the private IP address

Correct answer

Create a separate gateway endpoint for Amazon S3 and Amazon DynamoDB each. Add two new target entries for these two gateway endpoints in the route table of the custom VPC

Create a separate interface endpoint for Amazon S3 and Amazon DynamoDB each. Then connect to these services by adding these as targets in the route table of the custom VPC

Create a gateway endpoint for Amazon S3 and add it as a target in the route table of the custom VPC. Create an interface endpoint for Amazon DynamoDB and then add it as a target in the route table of the custom VPC

Overall explanation

Correct option:

Create a separate gateway endpoint for Amazon S3 and Amazon DynamoDB each. Add two new target entries for these two gateway endpoints in the route table of the custom VPC

Endpoints are virtual devices. They are horizontally scaled, redundant, and highly available VPC components. They allow communication between instances in your VPC and services without imposing availability risks or bandwidth constraints on your network traffic.

A VPC endpoint enables you to privately connect your VPC to supported AWS services and VPC endpoint services powered by AWS PrivateLink without requiring an internet gateway, NAT device, VPN connection, or AWS Direct Connect connection. Instances in your VPC do not require public IP addresses to communicate with resources in the service. Traffic between your VPC and the other service does not leave the Amazon network.

There are two types of VPC endpoints: interface endpoints and gateway endpoints. An interface endpoint is an elastic network interface with a private IP address from the IP address range of your subnet that serves as an entry point for traffic destined to a supported service.

A gateway endpoint is a gateway that you specify as a target for a route in your route table for traffic destined to a supported AWS service. The following AWS services are supported:

Amazon S3

Amazon DynamoDB

Incorrect options:

Create a gateway endpoint for Amazon S3 and add it as a target in the route table of the custom VPC. Create an interface endpoint for Amazon DynamoDB and then add it as a target in the route table of the custom VPC

Create a separate interface endpoint for Amazon S3 and Amazon DynamoDB each. Then connect to these services by adding these as targets in the route table of the custom VPC

Amazon DynamoDB supports AWS PrivateLink. With AWS PrivateLink, you can simplify private network connectivity between virtual private clouds (VPCs), DynamoDB, and your on-premises

data centers using interface VPC endpoints and private IP addresses. So, Amazon DynamoDB supports both interface endpoints as well as gateway endpoints. However, to use the interface endpoints, you need to connect to the given services using the private IP address, instead of creating an entry as a target in the route table of the custom VPC. Therefore, both these options are incorrect.

Create a gateway endpoint for Amazon DynamoDB and add it as a target in the route table of the custom VPC. Create an Origin Access Identity for Amazon S3 and then connect to the S3 service using the private IP address - Origin Access Identity (OAI) is used within the context of Amazon CloudFront. To restrict access to content that you serve from Amazon S3 buckets, you can create a special Amazon CloudFront user called an origin access identity (OAI) and associate it with your distribution. You cannot use OAI to facilitate access to Amazon S3 from a VPC.

References:

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

https://aws.amazon.com/about-aws/whats-new/2024/03/amazon-dynamodb-aws-privatelink/

Domain

Design Secure Architectures

Question 40Skipped

A company is developing a document management application on AWS. The application runs on Amazon EC2 instances in multiple Availability Zones (AZs). The company requires the document store to be highly available and the documents need to be returned immediately when requested. The engineering team has configured the application to use Amazon Elastic Block Store (Amazon EBS) to store the documents but the team is willing to consider other options to meet the availability requirement.

As a solutions architect, which of the following will you recommend?

Correct answer

Set up Amazon EBS as the Amazon EC2 instance root volume and then configure the application to use Amazon S3 as the document store

Create snapshots for the Amazon EBS volumes regularly and then build new volumes using those snapshots in additional Availability Zones

Set up Amazon EBS as the Amazon EC2 instance root volume and then configure the application to use Amazon S3 Glacier as the document store

Provision at least three Provisioned IOPS Amazon Instance Store volumes for the Amazon EC2 instances and then mount these volumes to multiple Amazon EC2 instances

Overall explanation

Correct option:

Set up Amazon EBS as the Amazon EC2 instance root volume and then configure the application to use Amazon S3 as the document store

Instances that use Amazon EBS for the root device automatically have an Amazon EBS volume attached. When you launch an Amazon EBS-backed instance, AWS creates an Amazon EBS volume for each Amazon EBS snapshot referenced by the AMI you use. An Amazon EBS-backed instance can be stopped and later restarted without affecting data stored in the attached volumes.

Amazon S3 provides access to reliable, fast, and inexpensive data storage infrastructure. It is designed to make web-scale computing easier by enabling you to store and retrieve any amount of data, at any time, from within Amazon EC2 or anywhere on the web. S3 is highly available and can be configured to work as a document store for the given use case.

Incorrect options:

Set up Amazon EBS as the Amazon EC2 instance root volume and then configure the application to use Amazon S3 Glacier as the document store - As the documents need to be returned immediately when requested, Amazon S3 Glacier is not the right fit, since there is a lag of several minutes/hours when you want to read data from Glacier.

Create snapshots for the Amazon EBS volumes regularly and then build new volumes using those snapshots in additional Availability Zones - You can back up the data on your Amazon EBS volumes to Amazon S3 by taking point-in-time snapshots. Snapshots are incremental backups, which means that only the blocks on the device that have changed after your most recent snapshot are saved. Hence, using Amazon EBS volumes as a primary storage solution is ineffective, and creating recurring snapshots is a management nightmare for the current use case.

Provision at least three Provisioned IOPS Amazon Instance Store volumes for the Amazon EC2 instances and then mount these volumes to multiple Amazon EC2 instances - You cannot mount Instance Store volumes to multiple Amazon EC2 instances. An instance store provides temporary block-level storage for your instance. This storage is located on disks that are physically attached to the host computer.

References:

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

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

Domain

Design Resilient Architectures

Question 41Skipped

An e-commerce application uses a relational database that runs several queries that perform joins on multiple tables. The development team has found that these queries are slow and expensive, therefore these are a good candidate for caching. The application needs to use a caching service that supports multi-threading.

As a solutions architect, which of the following services would you recommend for the given use case?

AWS Global Accelerator

Correct answer

Amazon ElastiCache for Memcached

Amazon ElastiCache for Redis

Amazon DynamoDB Accelerator (DAX)

Overall explanation

Correct option:

Amazon ElastiCache for Memcached

Amazon ElastiCache is a web service that makes it easy to deploy, operate, and scale an in-memory data store and cache in the cloud. The service improves the performance of web applications by allowing you to retrieve information from fast, managed, in-memory data stores, instead of relying entirely on slower disk-based databases.

Memcached is an open-source, distributed, in-memory key-value store that can retrieve data in milliseconds. Caching site information with Memcached can help you improve the performance and scalability of your site while controlling cost.

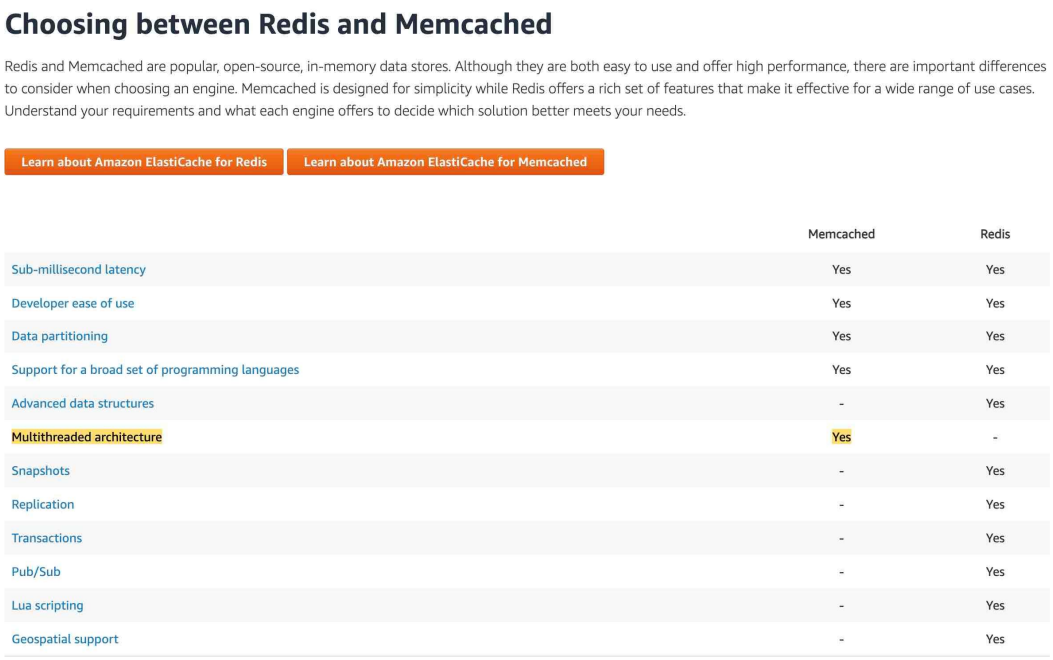
Choose Memcached if the following apply to you:

You need the simplest model possible.

You need to run large nodes with multiple cores or threads (support for multi-threading).

You need the ability to scale out and in, adding and removing nodes as demand on your system increases and decreases.

You need to cache objects.



Choosing between Redis and Memcached

Redis and Memcached are popular, open-source, in-memory data stores. Although they are both easy to use and offer high performance, there are important differences to consider when choosing an engine. Memcached is designed for simplicity while Redis offers a rich set of features that make it effective for a wide range of use cases. Understand your requirements and what each engine offers to decide which solution better meets your needs.

Learn about Amazon ElastiCache for Redis | Learn about Amazon ElastiCache for Memcached

	Memcached	Redis
Sub-millisecond latency	Yes	Yes
Developer ease of use	Yes	Yes
Data partitioning	Yes	Yes
Support for a broad set of programming languages	Yes	Yes
Advanced data structures	-	Yes
Multithreaded architecture	Yes	-
Snapshots	-	Yes
Replication	-	Yes
Transactions	-	Yes
Pub/Sub	-	Yes
Lua scripting	-	Yes
Geospatial support	-	Yes

via - https://aws.amazon.com/elasticache/redis-vs-memcached/

Incorrect options:

Amazon ElastiCache for Redis - Redis, which stands for Remote Dictionary Server, is a fast, open-source, in-memory key-value data store for use as a database, cache, message broker, and queue. Redis now delivers sub-millisecond response times enabling millions of requests per second for real-time applications in Gaming, Ad-Tech, Financial Services, Healthcare, and IoT. Redis is a popular choice for caching, session management, gaming, leaderboards, real-time analytics, geospatial, ride-hailing, chat/messaging, media streaming, and pub/sub apps.

Redis does not support multi-threading, so this option is not the right fit for the given use case.

Amazon DynamoDB Accelerator (DAX) - Amazon DynamoDB Accelerator (DAX) is a fully managed, highly available, in-memory cache for Amazon DynamoDB. DAX does not support relational databases.

AWS Global Accelerator - AWS Global Accelerator is a networking service that helps you improve the availability and performance of the applications that you offer to your global users. This option has been added as a distractor, it has nothing to do with database caching.

References:

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

https://docs.aws.amazon.com/AmazonElastiCache/latest/mem-ug/elasticache-use-cases.html

https://aws.amazon.com/elasticache/redis-vs-memcached/

Domain

Design Resilient Architectures

Question 42Skipped

A retail company's procurement application becomes slow when traffic spikes. The application has a three-tier architecture (web, application and database tier) that uses synchronous transactions. The engineering team at the company has identified certain bottlenecks in the application tier but it does not want to change the underlying application architecture.

As a solutions architect, which of the following solutions would you suggest to meet the required application response times while accounting for any traffic spikes?

Correct answer

Leverage horizontal scaling for the web and application tiers by using Auto Scaling groups and Application Load Balancer

Leverage vertical scaling for the application instance by provisioning a larger Amazon EC2 instance size

Leverage Amazon SQS with asynchronous AWS Lambda calls to decouple the application and data tiers

Leverage horizontal scaling for the application's persistence layer by adding Oracle RAC on AWS

Overall explanation

Correct option:

Leverage horizontal scaling for the web and application tiers by using Auto Scaling groups and Application Load Balancer

A horizontally scalable system is one that can increase capacity by adding more computers to the system. This is in contrast to a vertically scalable system, which is constrained to running its processes on only one computer; in such systems, the only way to increase performance is to add more resources into one computer in the form of faster (or more) CPUs, memory or storage.

Horizontally scalable systems are oftentimes able to outperform vertically scalable systems by enabling parallel execution of workloads and distributing those across many different computers.

Elastic Load Balancing is used to automatically distribute your incoming application traffic across all the Amazon EC2 instances that you are running. You can use Elastic Load Balancing to manage incoming requests by optimally routing traffic so that no one instance is overwhelmed.

To use Elastic Load Balancing with your Auto Scaling group, you attach the load balancer to your Auto Scaling group to register the group with the load balancer. Your load balancer acts as a single point of contact for all incoming web traffic to your Auto Scaling group.

When you use Elastic Load Balancing with your Auto Scaling group, it's not necessary to register individual Amazon EC2 instances with the load balancer. Instances that are launched by your Auto Scaling group are automatically registered with the load balancer. Likewise, instances that are terminated by your Auto Scaling group are automatically deregistered from the load balancer.

This option will require fewer design changes, it's mostly configuration changes and the ability for the web/application tier to be able to communicate across instances. Hence, this is the right solution for the current use case.

Incorrect options:

Leverage Amazon SQS with asynchronous AWS Lambda calls to decouple the application and data tiers - This is incorrect as it uses asynchronous AWS Lambda calls and the application uses synchronous transactions. The question says there should be no change in the application architecture.

Leverage horizontal scaling for the application's persistence layer by adding Oracle RAC on AWS - The issue is not with the persistence layer at all. This option has only been used as a distractor.

You can deploy scalable Oracle Real Application Clusters (RAC) on Amazon EC2 using Amazon Machine Images (AMI) on AWS Marketplace. Oracle RAC is a shared-everything database cluster technology from Oracle that allows a single database (a set of data files) to be concurrently accessed and served by one or many database server instances.

Leverage vertical scaling for the application instance by provisioning a larger Amazon EC2 instance size - Vertical scaling is just a band-aid solution and will not work long term.

References:

https://docs.aws.amazon.com/autoscaling/ec2/userguide/autoscaling-load-balancer.html

https://aws.amazon.com/blogs/compute/operating-lambda-understanding-event-driven-architecture-part-1/

Domain

Design Resilient Architectures

Question 43Skipped

A company manages a High Performance Computing (HPC) application that needs to be deployed on Amazon EC2 instances. The application requires high levels of inter-node communications and high network traffic between the instances.

As a solutions architect, which of the following options would you recommend to the engineering team at the company? (Select two)

Deploy Amazon EC2 instances behind a Network Load Balancer

Deploy Amazon EC2 instances in a spread placement group

Deploy Amazon EC2 instances in a partition placement group

Correct selection

Deploy Amazon EC2 instances in a cluster placement group

Correct selection

Deploy Amazon EC2 instances with Elastic Fabric Adapter (EFA)

Overall explanation

Correct options:

Deploy Amazon EC2 instances with Elastic Fabric Adapter (EFA)

Elastic Fabric Adapter (EFA) is a network interface for Amazon EC2 instances that enables customers to run applications requiring high levels of inter-node communications at scale on AWS. Its custom-built operating system (OS) bypass hardware interface enhances the performance of inter-instance communications, which is critical to scaling these applications. Therefore this option is correct.

Deploy Amazon EC2 instances in a cluster placement group

Cluster placement groups pack instances close together inside an Availability Zone. They are recommended when the majority of the network traffic is between the instances in the group. These are also recommended for applications that benefit from low network latency, high network throughput, or both. Therefore this option is one of the correct answers.

Incorrect options:

Deploy Amazon EC2 instances in a spread placement group - A spread placement group is a group of instances that are each placed on distinct racks, with each rack having its own network and power source. The instances are placed across distinct underlying hardware to reduce

correlated failures. You can have a maximum of seven running instances per Availability Zone per group. Since the spread placement group can span across multiple Availability Zones in the same Region, it cannot support high levels of inter-node communications and high network traffic. So this option is incorrect.

Deploy Amazon EC2 instances in a partition placement group - A partition placement group spreads your instances across logical partitions such that groups of instances in one partition do not share the underlying hardware with groups of instances in different partitions. This strategy is typically used by large distributed and replicated workloads, such as Hadoop, Cassandra, and Kafka. A partition placement group can have a maximum of seven partitions per Availability Zone. Since the partition placement group can have partitions in multiple Availability Zones in the same Region, it cannot support high levels of inter-node communications and high network traffic. So this option is incorrect.

Deploy Amazon EC2 instances behind a Network Load Balancer - A load balancer serves as the single point of contact for clients. The load balancer distributes incoming traffic across multiple targets, such as Amazon EC2 instances. A Network Load Balancer functions at the fourth layer of the Open Systems Interconnection (OSI) model. Network Load Balancer cannot facilitate high network traffic between instances. Network Load Balancer cannot support high levels of inter-node communication between EC2 instances. This option just serves as a distractor.

References:

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

https://aws.amazon.com/hpc/efa/

https://docs.aws.amazon.com/elasticloadbalancing/latest/network/introduction.html

Domain

Design High-Performing Architectures

Question 44Skipped

A Hollywood production studio is looking at transferring their existing digital media assets of around 20 petabytes to AWS Cloud in the shortest possible timeframe.

Which of the following is an optimal solution for this requirement, given that the studio's data centers are located at a remote location?

AWS Storage Gateway

AWS Snowball

Correct answer

AWS Snowmobile

AWS Direct Connect

Overall explanation

Correct option:

AWS Snowmobile

AWS Snowmobile is an Exabyte-scale data transfer service used to move extremely large amounts of data to AWS. You can transfer up to 100PB per Snowmobile, a 45-foot long ruggedized shipping container, pulled by a semi-trailer truck. Snowmobile makes it easy to move massive volumes of data to the cloud, including video libraries, image repositories, or even a complete data center migration. Transferring data with Snowmobile is more secure, fast, and cost-effective. AWS recommends using Snowmobile to migrate large datasets of 10PB or more in a single location. For datasets less than 10PB or distributed in multiple locations, you should use Snowball.

Incorrect options:

AWS Snowball - The AWS Snowball service uses physical storage devices to transfer large amounts of data between Amazon Simple Storage Service (Amazon S3) and client's onsite data storage location at faster-than-internet speeds. Snowball provides powerful interfaces that you can use to create jobs, track data, and track the status of your jobs through to completion. AWS recommends snowball only if you want to transfer greater than 10 TB of data between your on-premises data centers and Amazon S3.

AWS Storage Gateway - AWS Storage Gateway is a hybrid cloud storage service that gives you on-premises access to virtually unlimited cloud storage. Used for key hybrid storage solutions that include moving tape backups to the cloud, reducing on-premises storage with cloud-backed file shares, providing low latency access to data in AWS for on-premises applications, as well as various migration, archiving, processing, and disaster recovery use cases. This is not an optimal solution since the studio's data centers are in remote locations where internet speed may not optimal, thereby increasing both cost and time for migrating 20TB of data.

AWS Direct Connect - AWS Direct Connect is a network service that provides an alternative to using the Internet to connect a customer’s on-premises sites to AWS. Data is transmitted through a private network connection between AWS and a customer’s datacenter or corporate network. Direct Connect connection takes significant cost as well as time to provision. This is not the correct solution since the studio wants the data transfer to be done in the shortest possible time.

Reference:

https://aws.amazon.com/snowmobile/

Domain

Design Cost-Optimized Architectures

Question 45Skipped

The engineering team at a company wants to create a daily big data analysis job leveraging Spark for analyzing online/offline sales and customer loyalty data to create customized reports on a client-by-client basis. The big data analysis job needs to read the data from Amazon S3 and output it back to Amazon S3.

Which technology do you recommend to run the Big Data analysis job? (Select two)

AWS Batch

Amazon Athena

Correct selection

Amazon EMR

Amazon Redshift

Correct selection

AWS Glue

Overall explanation

Correct options:

Amazon EMR

Amazon EMR is the industry-leading cloud big data platform for processing vast amounts of data using open source tools such as Apache Spark, Apache Hive, Apache HBase, Apache Flink, Apache Hudi, and Presto. With EMR you can run Petabyte-scale analysis at less than half of the cost of traditional on-premises solutions and over 3x faster than standard Apache Spark. EMR is used for launching Hadoop / Spark clusters. For short-running jobs, you can spin up and spin down clusters and pay per second for the instances used. For long-running workloads, you can create highly available clusters that automatically scale to meet demand. Amazon EMR uses Hadoop, an open-source framework, to distribute your data and processing across a resizable cluster of Amazon EC2 instances.

AWS Glue

AWS Glue is a fully managed extract, transform, and load (ETL) service that makes it easy for customers to prepare and load their data for analytics. AWS Glue job is meant to be used for batch ETL data processing. AWS Glue ETL jobs can use Amazon S3, data stores in a VPC, or on-premises JDBC data stores as a source. AWS Glue jobs extract data, transform it, and load the resulting data back to S3, data stores in a VPC, or on-premises JDBC data stores as a target.

Incorrect options:

Amazon Redshift - Amazon Redshift is a fully-managed petabyte-scale cloud-based data warehouse product designed for large scale data set storage and analysis. An Amazon Redshift data warehouse is a collection of computing resources called nodes, which are organized into a group called a cluster. Each cluster runs an Amazon Redshift engine and contains one or more databases. An Amazon Redshift cluster consists of nodes. Each cluster has a leader node and one or more compute nodes. The leader node receives queries from client applications, parses the queries, and develops query execution plans. The leader node then coordinates the parallel execution of these plans with the compute nodes and aggregates the intermediate results from these nodes. It then finally returns the results to the client applications.

Amazon Athena - Amazon Athena is an interactive query service that makes it easy to analyze data directly in Amazon S3 using standard SQL. Athena is serverless, so there is no infrastructure to set up or manage, and customers pay only for the queries they run. You can use Athena to process logs, perform ad-hoc analysis, and run interactive queries.

AWS Batch - AWS Batch can be used to plan, schedule, and execute your batch computing workloads on Amazon EC2 Instances. AWS Batch dynamically provisions the optimal quantity and type of compute resources (e.g., CPU or memory optimized compute resources) based on the volume and specific resource requirements of the batch jobs submitted.

References:

https://aws.amazon.com/emr/

https://aws.amazon.com/blogs/big-data/how-to-access-and-analyze-on-premises-data-stores-using-aws-glue/

Domain

Design High-Performing Architectures

Question 46Skipped

A Big Data consulting company runs large distributed and replicated workloads on the on-premises data center. The company now wants to move these workloads to Amazon EC2 instances by using the placement groups feature and it wants to minimize correlated hardware failures.

Which of the following represents the correct placement group configuration for the given requirement?

Multi-AZ placement groups

Cluster placement groups

Spread placement groups

Correct answer

Partition placement groups

Overall explanation

Correct option:

Partition placement groups

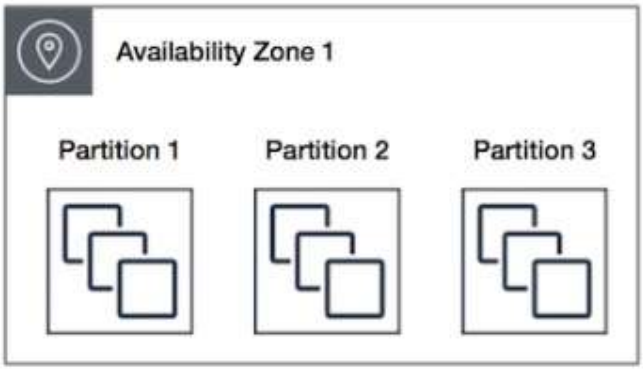
Partition placement groups help reduce the likelihood of correlated hardware failures for your application. When using partition placement groups, Amazon EC2 divides each group into logical segments called partitions. Amazon EC2 ensures that each partition within a placement group has its own set of racks. Each rack has its own network and power source. No two partitions within a placement group share the same racks, allowing you to isolate the impact of a hardware failure within your application.

The following image is a simple visual representation of a partition placement group in a single Availability Zone. It shows instances that are placed into a partition placement group with three partitions—Partition 1, Partition 2, and Partition 3. Each partition comprises multiple instances. The instances in a partition do not share racks with the instances in the other partitions, allowing you to contain the impact of a single hardware failure to only the associated partition.

Partition placement groups can be used to deploy large distributed and replicated workloads, such as HDFS, HBase, and Cassandra, across distinct racks. When you launch instances into a partition placement group, Amazon EC2 tries to distribute the instances evenly across the number of partitions that you specify. You can also launch instances into a specific partition to have more control over where the instances are placed.

A partition placement group can have partitions in multiple Availability Zones in the same Region. A partition placement group can have a maximum of seven partitions per Availability Zone. The number of instances that can be launched into a partition placement group is limited only by the limits of your account.

Partition placement groups:



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

Incorrect options:

Cluster placement groups - 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. Cluster placement groups are recommended for applications that benefit from low network latency, high network throughput, or both. They are also recommended when the majority of the network traffic is between the instances in the group. As the instances are packed close together inside an Availability Zone, this option is not correct for the given use case.

Cluster placement groups:

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

Spread placement groups - A spread placement group is a group of instances that are each placed on distinct racks, with each rack having its own network and power source. Spread placement groups are recommended for applications that have a small number of critical instances that should be kept separate from each other. Launching instances in a spread placement group reduces the risk of simultaneous failures that might occur when instances share the same racks. Spread placement groups provide access to distinct racks, and are therefore suitable for mixing instance types or launching instances over time. As the use-case talks about running large distributed and replicated workloads, so it needs more instances, therefore this option is not the right fit for the given use-case.

A spread placement group can span multiple Availability Zones in the same Region. You can have a maximum of seven running instances per Availability Zone per group.

The following image shows seven instances in a single Availability Zone that are placed into a spread placement group. The seven instances are placed on seven different racks.

Spread placement groups:

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

Multi-AZ placement groups - This is a made-up option, given as a distractor. You should note that the Partition and Spread placement groups can span across multiple Availability Zones in the same Region.

Reference:

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

Domain

Design High-Performing Architectures

Question 47Skipped

A Big Data company wants to optimize its daily Extract-Transform-Load (ETL) process that migrates and transforms data from its Amazon S3 based data lake to an Amazon Redshift cluster. The team wants to manage this daily job in a serverless environment.

Which AWS service is the best fit to manage this process without the need to configure or manage the underlying compute resources?

Amazon EMR

Correct answer

AWS Glue

AWS Data Pipeline

AWS Database Migration Service (DMS)

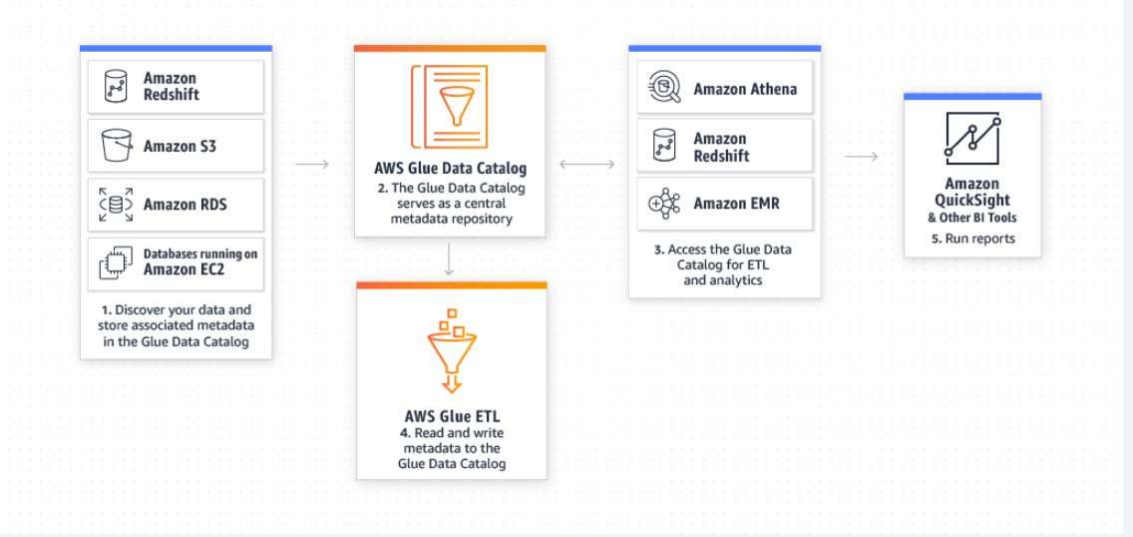
Overall explanation

Correct option:

AWS Glue

AWS Glue provides a managed ETL service that runs on a serverless Apache Spark environment. This allows you to focus on your ETL job and not worry about configuring and managing the underlying compute resources. AWS Glue takes a data-first approach and allows you to focus on the data properties and data manipulation to transform the data to a form where you can derive business insights. It provides an integrated data catalog that makes metadata available for ETL as well as querying via Amazon Athena and Amazon Redshift Spectrum.

Create a unified catalog to find data across multiple data stores using AWS Glue:



via - https://aws.amazon.com/glue/

AWS Glue automates much of the effort required for data integration. AWS Glue crawls your data sources, identifies data formats, and suggests schemas to store your data. It automatically generates the code to run your data transformations and loading processes. You can use AWS Glue to easily run and manage thousands of ETL jobs or to combine and replicate data across multiple data stores using SQL.

AWS Glue runs in a serverless environment. There is no infrastructure to manage, and AWS Glue provisions, configures, and scales the resources required to run your data integration jobs. You pay only for the resources your jobs use while running.

AWS Glue is the right fit since the company is looking at a managed ETL service without having the overhead of configuring, maintaining, or managing any servers.

via - https://docs.aws.amazon.com/prescriptive-guidance/latest/patterns/load-data-from-amazon-s3-to-amazon-redshift-using-aws-glue.html

Incorrect options:

AWS Data Pipeline - AWS Data Pipeline provides a managed orchestration service that gives you greater flexibility in terms of the execution environment, access and control over the compute resources that run your code, as well as the code itself that does data processing. AWS Data Pipeline launches compute resources in your account allowing you direct access to the Amazon EC2 instances or Amazon EMR clusters. As this option provides access to the underlying EC2 instances so it's not a serverless solution. Therefore this option is incorrect for the given use case.

Amazon EMR - EMR is a web service to easily and cost-effectively process vast amounts of data. EMR utilizes a hosted Hadoop framework running on the web-scale infrastructure of Amazon Elastic Compute Cloud (Amazon EC2) and Amazon Simple Storage Service (Amazon S3). As this option provides access to the underlying Amazon EC2 instances so it's not a serverless solution. Therefore this option is incorrect for the given use case.

AWS Database Migration Service (DMS) - AWS Database Migration Service (DMS) helps you migrate databases to AWS easily and securely. For use cases that require a database migration from on-premises to AWS or database replication between on-premises sources and sources on AWS, AWS recommends you use AWS DMS. Once your data is in AWS, you can use AWS Glue to move, combine, replicate, and transform data from your data source into another database or data warehouse, such as Amazon Redshift. As the use-case talks about data migration and transformation between AWS services, so AWS Glue is a better fit than DMS.

References:

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

https://docs.aws.amazon.com/prescriptive-guidance/latest/patterns/load-data-from-amazon-s3-to-amazon-redshift-using-aws-glue.html

Domain

Design High-Performing Architectures

Question 48Skipped

As a Solutions Architect, you would like to completely secure the communications between your Amazon CloudFront distribution and your Amazon S3 bucket which contains the static files for your website. Users should only be able to access the Amazon S3 bucket through Amazon CloudFront and not directly.

What do you recommend?

Create a bucket policy to only authorize the IAM role attached to the Amazon CloudFront distribution

Make the Amazon S3 bucket public

Update the Amazon S3 bucket security groups to only allow traffic from the Amazon CloudFront security group

Correct answer

Create an origin access identity (OAI) and update the Amazon S3 Bucket Policy

Overall explanation

Correct option:

Create an origin access identity (OAI) and update the Amazon S3 Bucket Policy

To restrict access to content that you serve from Amazon S3 buckets, you need to follow the following steps:

1. Create a special Amazon CloudFront user called an origin access identity (OAI) and associate it with your distribution.
2. Configure your Amazon S3 bucket permissions so that Amazon CloudFront can use the OAI to access the files in your bucket and serve them to your users. Make sure that users can't use a direct URL to the Amazon S3 bucket to access a file there.

After you take these steps, users can only access your files through Amazon CloudFront, not directly from the Amazon S3 bucket.

In general, if you're using an Amazon S3 bucket as the origin for a Amazon CloudFront distribution, you can either allow everyone to have access to the files there, or you can restrict access. If you restrict access by using, for example, Amazon CloudFront signed URLs or signed cookies, you also won't want people to be able to view files by simply using the direct Amazon S3 URL for the file. Instead, you want them to only access the files by using the Amazon CloudFront URL, so your content remains protected.

Incorrect options:

Update the Amazon S3 bucket security groups to only allow traffic from the Amazon CloudFront security group - Amazon S3 buckets don't have security groups, hence this is an incorrect option.

Make the Amazon S3 bucket public - If the Amazon S3 bucket is made public, it can be accessed by anyone directly. This is not the requirement.

Create a bucket policy to only authorize the IAM role attached to the Amazon CloudFront distribution - You cannot attach IAM roles to the Amazon CloudFront distribution. Here you need to use an OAI.

Reference:

https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/private-content-restricting-access-to-s3.html

Domain

Design Secure Architectures

Question 49Skipped

A company is deploying a publicly accessible web application. To accomplish this, the engineering team has designed the VPC with a public subnet and a private subnet. The

application will be hosted on several Amazon EC2 instances in an Auto Scaling group. The team also wants Transport Layer Security (TLS) termination to be offloaded from the Amazon EC2 instances.

Which solution should a solutions architect implement to address these requirements in the most secure manner?

Set up a Network Load Balancer in the private subnet. Create an Auto Scaling group in the public subnet and associate it with the Network Load Balancer

Correct answer

Set up a Network Load Balancer in the public subnet. Create an Auto Scaling group in the private subnet and associate it with the Network Load Balancer

Set up a Network Load Balancer in the public subnet. Create an Auto Scaling group in the public subnet and associate it with the Network Load Balancer

Set up a Network Load Balancer in the private subnet. Create an Auto Scaling group in the private subnet and associate it with the Network Load Balancer

Overall explanation

Correct option:

Set up a Network Load Balancer in the public subnet. Create an Auto Scaling group in the private subnet and associate it with the Network Load Balancer

A load balancer serves as the single point of contact for clients. The load balancer distributes incoming traffic across multiple targets, such as Amazon EC2 instances. This increases the availability of your application. You add one or more listeners to your load balancer.

With a Network Load Balancer, you can offload the decryption/encryption of Transport Layer Security (TLS) traffic from your application servers to the Network Load Balancer, which helps you optimize the performance of your backend application servers while keeping your workloads secure. Additionally, Network Load Balancers preserve the source IP of the clients to the back-end applications, while terminating Transport Layer Security (TLS) on the load balancer.

An Auto Scaling group contains a collection of Amazon EC2 instances that are treated as a logical grouping for the purposes of automatic scaling and management. An Auto Scaling group also enables you to use Amazon EC2 Auto Scaling features such as health check replacements and scaling policies. Both maintaining the number of instances in an Auto Scaling group and automatic scaling are the core functionality of the Amazon EC2 Auto Scaling service.

The NLB has to be accessible over the internet and hence has to be in a public subnet and will act as a single point-of-contact for all incoming traffic. NLB will forward the incoming traffic to the Amazon EC2 instances managed by the ASG in the private subnet.

Exam Alert:

You should note that the Application Load Balancer also supports Transport Layer Security (TLS) offloading. The Classic Load Balancer supports SSL offloading.

Incorrect options:

Set up a Network Load Balancer in the public subnet. Create an Auto Scaling group in the public subnet and associate it with the Network Load Balancer - The Auto Scaling group with its target EC2 instances should be in the private subnet to avoid access to EC2 instances over the public internet. Having EC2 instances in the public subnet would weaken the security posture of the application. Hence, this option is incorrect.

Set up a Network Load Balancer in the private subnet. Create an Auto Scaling group in the public subnet and associate it with the Network Load Balancer

Set up a Network Load Balancer in the private subnet. Create an Auto Scaling group in the private subnet and associate it with the Network Load Balancer

NLB should be in the public subnet as it represents the internet-facing component of the web tier. Therefore, both these options are incorrect.

Reference:

https://aws.amazon.com/blogs/aws/new-tls-termination-for-network-load-balancers/

Domain

Design Secure Architectures

Question 50Skipped

Your e-commerce application is using an Amazon RDS PostgreSQL database and an analytics workload also runs on the same database. When the analytics workload is run, your e-commerce application slows down which further affects your sales.

Which of the following is the MOST cost-optimal solution to fix this issue?

Correct answer

Create a Read Replica in the same Region as the Master database and point the analytics workload there

Create a Read Replica in another Region as the Master database and point the analytics workload there

Migrate the analytics application to AWS Lambda

Enable Multi-AZ for the Amazon RDS database and run the analytics workload on the standby database

Overall explanation

Correct option:

Create a Read Replica in the same Region as the Master database and point the analytics workload there

Amazon RDS Read Replicas provide enhanced performance and durability for RDS database (DB) instances. They make it easy to elastically scale out beyond the capacity constraints of a single DB instance for read-heavy database workloads. For the MySQL, MariaDB, PostgreSQL, Oracle, and SQL Server database engines, Amazon RDS creates a second DB instance using a snapshot of the source DB instance. It then uses the engines' native asynchronous replication

to update the read replica whenever there is a change to the source database instance. Read replicas can be within an Availability Zone, Cross-AZ, or Cross-Region.

Creating a Read Replica is the answer. As we want to minimize the costs, we need to launch the Read Replica in the same Region as you are not charged for the data transfer incurred in replicating data between your source database instance and read replica within the same AWS Region.

Exam Alert:

Please review this comparison vis-a-vis Multi-AZ vs Read Replica for Amazon RDS:

Multi-AZ deployments, multi-region deployments, and read replicas

Amazon RDS Multi-AZ deployments complement multi-region deployments and read replicas. While all three features increase availability and durability by maintaining additional copies of your data, there are differences between them:

Multi-AZ deployments	Multi-Region deployments	Read replicas
Main purpose is high availability	Main purpose is disaster recovery and local performance	Main purpose is scalability
Non-Aurora: synchronous replication; Aurora: asynchronous replication	Asynchronous replication	Asynchronous replication
Non-Aurora: only the primary instance is active; Aurora: all instances are active	All regions are accessible and can be used for reads	All read replicas are accessible and can be used for readscaling
Non-Aurora: automated backups are taken from standby; Aurora: automated backups are taken from shared storage layer	Automated backups can be taken in each region	No backups configured by default
Always span at least two Availability Zones within a single region	Each region can have a Multi-AZ deployment	Can be within an Availability Zone, Cross-AZ, or Cross-Region
Non-Aurora: database engine version upgrades happen on primary; Aurora: all instances are updated together	Non-Aurora: database engine version upgrade is independent in each region; Aurora: all instances are updated together	Non-Aurora: database engine version upgrade is independent from source instance; Aurora: all instances are updated together
Automatic failover to standby (non-Aurora) or read replica (Aurora) when a problem is detected	Aurora allows promotion of a secondary region to be the master	Can be manually promoted to a standalone database instance (non-Aurora) or to be the primary instance (Aurora)

via - https://aws.amazon.com/rds/features/multi-az/

Incorrect options:

Enable Multi-AZ for the Amazon RDS database and run the analytics workload on the standby database - Amazon RDS Multi-AZ deployments provide enhanced availability and durability for RDS 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). Multi-AZ spans at least two Availability Zones within a single region.

Enabling Multi-AZ helps make our database highly-available, but the standby database is not accessible and cannot be used for reads or write. It's just a database that will become primary when the other database encounters a failure. So this option is not correct.

Migrate the analytics application to AWS Lambda- AWS Lambda lets you run code without provisioning or managing servers. You pay only for the compute time you consume.

Running the application on AWS Lambda will not help, as it will still run against the main database and slow down our e-commerce application.

Create a Read Replica in another Region as the Master database and point the analytics workload there - This is incorrect because we have to pay for inter-Region data replication charges for the Read Replica, whereas the replication of data within a single Region is free.

References:

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

https://aws.amazon.com/rds/features/read-replicas/

Domain

Design Cost-Optimized Architectures

Question 51Skipped

During a review, a security team has flagged concerns over an Amazon EC2 instance querying IP addresses used for cryptocurrency mining. The Amazon EC2 instance does not host any authorized application related to cryptocurrency mining.

Which AWS service can be used to protect the Amazon EC2 instances from such unauthorized behavior in the future?

AWS Web Application Firewall (AWS WAF)

AWS Firewall Manager

AWS Shield Advanced

Correct answer

Amazon GuardDuty

Overall explanation

Correct option:

Amazon GuardDuty

Amazon GuardDuty continuously monitors for malicious or unauthorized behavior to help protect your AWS resources, including your AWS accounts and access keys. Amazon GuardDuty identifies any unusual or unauthorized activity, like cryptocurrency mining or infrastructure deployments in a region that has never been used. Powered by threat intelligence and machine learning, GuardDuty is continuously evolving to help you protect your AWS environment.

The cryptocurrency finding expands the service's ability to detect Amazon EC2 instances querying IP addresses associated with the cryptocurrency-related activity. The finding type is: CryptoCurrency:EC2/BitcoinTool.B, CryptoCurrency:EC2/BitcoinTool.B!DNS.

This finding informs you that the listed Amazon EC2 instance in your AWS environment is querying a domain name that is associated with Bitcoin or other cryptocurrency-related activity. Bitcoin is a worldwide cryptocurrency and digital payment system that can be exchanged for

other currencies, products, and services. Bitcoin is a reward for bitcoin mining and is highly sought after by threat actors.

If you use the Amazon EC2 instance to mine or manage cryptocurrency, or this instance is otherwise involved in blockchain activity, this finding could represent expected activity for your environment. If this is the case in your AWS environment, AWS recommends that you set up a suppression rule for this finding.

Incorrect options:

AWS Web Application Firewall (AWS WAF) - AWS WAF is a web application firewall that helps protect your web applications or APIs against common web exploits and bots that may affect availability, compromise security, or consume excessive resources. AWS WAF gives you control over how traffic reaches your applications by enabling you to create security rules that control bot traffic and block common attack patterns, such as SQL injection or cross-site scripting.

AWS Shield Advanced - For higher levels of protection against attacks targeting your applications running on Amazon Elastic Compute Cloud (EC2), Elastic Load Balancing (ELB), Amazon CloudFront, AWS Global Accelerator, and Amazon Route 53 resources, you can subscribe to AWS Shield Advanced. In addition to the network and transport layer protections that come with Standard, AWS Shield Advanced provides additional detection and mitigation against large and sophisticated DDoS attacks, near real-time visibility into attacks, and integration with AWS WAF, a web application firewall. AWS Shield Advanced also gives you 24x7 access to the AWS DDoS Response Team (DRT) and protection against DDoS-related spikes in your Amazon Elastic Compute Cloud (EC2), Elastic Load Balancing (ELB), Amazon CloudFront, AWS Global Accelerator, and Amazon Route 53 charges.

AWS Firewall Manager - AWS Firewall Manager is a security management service that allows you to centrally configure and manage firewall rules across your accounts and applications in AWS Organizations. As new applications are created, Firewall Manager makes it easy to bring new applications and resources into compliance by enforcing a common set of security rules. Now you have a single service to build firewall rules, create security policies, and enforce them in a consistent, hierarchical manner across your entire infrastructure, from a central administrator account.

None of these three services can detect unauthorized cryptocurrency mining activity on EC2 instances, so these options are incorrect.

Reference:

https://docs.aws.amazon.com/guardduty/latest/ug/guardduty_finding-types-ec2.html#cryptocurrency-ec2-bitcointoolbdns

Domain

Design Secure Architectures

Question 52Skipped

A security consultant is designing a solution for a company that wants to provide developers with individual AWS accounts through AWS Organizations, while also maintaining standard security controls. Since the individual developers will have AWS account root user-level access

to their own accounts, the consultant wants to ensure that the mandatory AWS CloudTrail configuration that is applied to new developer accounts is not modified.

Which of the following actions meets the given requirements?

Configure a new trail in AWS CloudTrail from within the developer accounts with the organization trails option enabled

Set up a service-linked role for AWS CloudTrail with a policy condition that allows changes only from an Amazon Resource Name (ARN) in the master account

Set up an IAM policy that prohibits changes to AWS CloudTrail and attach it to the root user

Correct answer

Set up a service control policy (SCP) that prohibits changes to AWS CloudTrail, and attach it to the developer accounts

Overall explanation

Correct option:

Set up a service control policy (SCP) that prohibits changes to AWS CloudTrail, and attach it to the developer accounts

Service control policy (SCP) is a type of organization policy that you can use to manage permissions in your organization. SCPs offer central control over the maximum available permissions for all accounts in your organization. SCPs help you to ensure your accounts stay within your organization's access control guidelines.

An SCP restricts permissions for IAM users and roles in member accounts, including the member account's root user. Any account has only those permissions permitted by every parent above it. If a permission is blocked at any level above the account, either implicitly (by not being included in an Allow policy statement) or explicitly (by being included in a Deny policy statement), a user or role in the affected account can't use that permission, even if the account administrator attaches the AdministratorAccess IAM policy with */ permissions to the user.

SCPs don't affect users or roles in the management account. They affect only the member accounts in your organization.

Incorrect options:

Configure a new trail in AWS CloudTrail from within the developer accounts with the organization trails option enabled - Configuring each developer account individually is not a viable solution to start with. In addition, any configuration changes can be undone by the user once they are logged into their individual accounts as root users.

Set up an IAM policy that prohibits changes to AWS CloudTrail and attach it to the root user - The root user can modify this IAM policy itself, so this option is not correct.

Set up a service-linked role for AWS CloudTrail with a policy condition that allows changes only from an Amazon Resource Name (ARN) in the master account - A service-linked role is a unique type of IAM role that is linked directly to an AWS service. Service-linked roles are predefined by the service and include all the permissions that the service requires to call other

AWS services on your behalf. The linked service also defines how you create, modify, and delete a service-linked role.

The linked service defines the permissions of its service-linked roles, and unless defined otherwise, only that service can assume the roles. The defined permissions include the trust policy and the permissions policy, and that permissions policy cannot be attached to any other entity such as the ARN in the master account.

Reference:

https://docs.aws.amazon.com/organizations/latest/userguide/orgs_manage_policies_scps.html

Domain

Design Secure Architectures

Question 53Skipped

An engineering team wants to orchestrate multiple Amazon ECS task types running on Amazon EC2 instances that are part of the Amazon ECS cluster. The output and state data for all tasks need to be stored. The amount of data output by each task is approximately 20 megabytes and there could be hundreds of tasks running at a time. As old outputs are archived, the storage size is not expected to exceed 1 terabyte.

As a solutions architect, which of the following would you recommend as an optimized solution for high-frequency reading and writing?

Use Amazon DynamoDB table that is accessible by all ECS cluster instances

Use Amazon EFS with Bursting Throughput mode

Correct answer

Use Amazon EFS with Provisioned Throughput mode

Use an Amazon EBS volume mounted to the Amazon ECS cluster instances

Overall explanation

Correct option:

Amazon EFS file systems are distributed across an unconstrained number of storage servers. This distributed data storage design enables file systems to grow elastically to petabyte scale. It also enables massively parallel access from compute instances, including Amazon EC2, Amazon ECS, and AWS Lambda, to your data.

Use Amazon EFS with Provisioned Throughput mode

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.

If your file system is in the Provisioned Throughput mode, you can increase the Provisioned Throughput of your file system as often as you want. You can decrease your file system throughput in Provisioned Throughput mode as long as it's been more than 24 hours since the last decrease. Additionally, you can change between Provisioned Throughput mode and the default Bursting Throughput mode as long as it's been more than 24 hours since the last throughput mode change.

Provisioned throughput

With Provisioned throughput, you specify a level of throughput that the file system can drive independent of the file system's size or burst credit balance. Use Provisioned throughput if you know your workload's performance requirements, or if your application drives throughput at 5% or more of the average-to-peak ratio.

For file systems using Provisioned throughput, you are charged for the amount of throughput enabled for the file system. The throughput amount billed in a month is based on the throughput provisioned in excess of your file system's included baseline throughput from Standard storage, up to the prevailing Bursting baseline throughput limits in the AWS Region.

If the file system's baseline throughput exceeds the Provisioned throughput amount, then it automatically uses the Bursting throughput allowed for the file system (up to the prevailing \Bursting baseline throughput limits in that AWS Region).

For information about per-RegionProvisioned throughput limits, see Amazon EFS quotas that you can increase.

Bursting throughput

Bursting throughput is recommended for workloads that require throughput that scales with the amount of storage in your file system. With Bursting throughput, the base throughput is proportionate to the file system's size in the Standard storage class, at a rate of 50 KiBps per each GiB of storage. Burst credits accrue when the file system consumes below its base throughput rate, and are deducted when throughput exceeds the base rate.

When burst credits are available, a file system can drive throughput up to 100 MiBps per TiB of storage, up to the AWS Region limit, with a minimum of 100 MiBps. If no burst credits are available, a file system can drive up to 50 MiBps per TiB of storage, with a minimum of 1 MiBps.

For information about per-Region Bursting throughput, see General resource quotas that cannot be changed.

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

Incorrect options:

Use Amazon EFS with Bursting Throughput mode - With Bursting Throughput mode, a file system's throughput scales as the amount of data stored in the standard 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. By default, AWS recommends that you run your application in the Bursting Throughput mode. But, if you're planning to migrate large amounts of data into your file system, consider switching to Provisioned Throughput mode.

The use-case mentions that the solution should be optimized for high-frequency reading and writing even when the old outputs are archived, therefore Provisioned Throughput mode is a better fit as it guarantees high levels of throughput your applications require without having to pad your file system.

Use an Amazon EBS volume mounted to the Amazon ECS cluster instances - Amazon EFS has a higher throughput than Amazon EBS. In addition, Amazon EBS can be attached to multiple Amazon EC2 instances when the underlying EBS type is io1/io2 and the instance is of Nitro type. The use-case does not provide any such details, so this option is ruled out.

Use Amazon DynamoDB table that is accessible by all ECS cluster instances - Amazon DynamoDB is not a fit for this scenario as each task output is 20 MB but the storage limit for each item in a Amazon DynamoDB table is 400 KB. You could write custom code to split the task

output data into multiple items but it is not an optimal solution compared to using Amazon EFS in Provisioned Throughput mode.

References:

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

https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Limits.html#limits-items

Domain

Design High-Performing Architectures

Question 54Skipped

A company's real-time streaming application is running on AWS. As the data is ingested, a job runs on the data and takes 30 minutes to complete. The workload frequently experiences high latency due to large amounts of incoming data. A solutions architect needs to design a scalable and serverless solution to enhance performance.

Which combination of steps should the solutions architect take? (Select two)

Provision Amazon EC2 instances in an Auto Scaling group to process the data

Correct selection

Set up Amazon Kinesis Data Streams to ingest the data

Set up AWS Lambda with AWS Step Functions to process the data

Set up AWS Database Migration Service (AWS DMS) to ingest the data

Correct selection

Set up AWS Fargate with Amazon ECS to process the data

Overall explanation

Correct options:

Set up Amazon Kinesis Data Streams to ingest the data

Set up AWS Fargate with Amazon ECS to process the data

Amazon Kinesis Data Streams (KDS) is a massively scalable and durable real-time data streaming service. KDS can continuously capture gigabytes of data per second from hundreds of thousands of sources such as website clickstreams, database event streams, financial transactions, social media feeds, IT logs, and location-tracking events. The data collected is available in milliseconds to enable real-time analytics use cases such as real-time dashboards, real-time anomaly detection, dynamic pricing, and more.

AWS Fargate is a serverless compute engine for containers that works with both Amazon Elastic Container Service (ECS) and Amazon Elastic Kubernetes Service (EKS). Fargate makes it easy for you to focus on building your applications. Fargate removes the need to provision and manage servers, lets you specify and pay for resources per application, and improves security through application isolation by design.

For the given use case, we can use Kinesis Data Streams as the ingestion layer and the containerized ECS application on AWS Fargate as the processing layer. Both these components are serverless and can scale to offer the desired performance.

Incorrect options:

Set up AWS Database Migration Service (AWS DMS) to ingest the data - AWS Database Migration Service helps you migrate databases to AWS quickly and securely. DMS cannot be used for real-time data ingestion. Hence, this option is incorrect.

Set up AWS Lambda with AWS Step Functions to process the data - The maximum timeout value for any AWS Lambda function is 15 minutes. When the specified timeout is reached, AWS Lambda terminates the execution of your Lambda function. Since the use case talks about a job that runs for 30 minutes, Lambda is not an option here.

Provision Amazon EC2 instances in an Auto Scaling group to process the data - The given requirement is for a serverless solution to process the data. Hence, provisioning an Amazon EC2 instance is clearly not the right solution.

Reference:

https://aws.amazon.com/blogs/big-data/building-a-scalable-streaming-data-processor-with-amazon-kinesis-data-streams-on-aws-fargate/

Domain

Design High-Performing Architectures

Question 55Skipped

A digital media streaming company wants to use Amazon CloudFront to distribute its content only to its service subscribers. As a solutions architect, which of the following solutions would you suggest to deliver restricted content to the bona fide end users? (Select two)

Require HTTPS for communication between Amazon CloudFront and your custom origin

Require HTTPS for communication between Amazon CloudFront and your S3 origin

Correct selection

Use Amazon CloudFront signed URLs

Forward HTTPS requests to the origin server by using the ECDSA or RSA ciphers

Correct selection

Use Amazon CloudFront signed cookies

Overall explanation

Correct options:

Use Amazon CloudFront signed URLs

Many companies that distribute content over the internet want to restrict access to documents, business data, media streams, or content that is intended for selected users, for example, users who have paid a fee.

To securely serve this private content by using Amazon CloudFront, you can do the following:

Require that your users access your private content by using special Amazon CloudFront signed URLs or signed cookies.

A signed URL includes additional information, for example, expiration date and time, that gives you more control over access to your content. So this is a correct option.

Use Amazon CloudFront signed cookies

Amazon CloudFront signed cookies allow you to control who can access your content when you don't want to change your current URLs or when you want to provide access to multiple restricted files, for example, all of the files in the subscribers' area of a website. So this is also a correct option.

Incorrect options:

Require HTTPS for communication between Amazon CloudFront and your custom origin

Require HTTPS for communication between Amazon CloudFront and your S3 origin

Requiring HTTPS for communication between Amazon CloudFront and your custom origin (or S3 origin) only enables secure access to the underlying content. You cannot use HTTPS to restrict access to your private content. So both these options are incorrect.

Forward HTTPS requests to the origin server by using the ECDSA or RSA ciphers - This option is just added as a distractor. You cannot use HTTPS to restrict access to your private content.

Reference:

https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/private-content-signed-urls.html

https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/private-content-signed-cookies.html

Domain

Design Resilient Architectures

Question 56Skipped

An e-commerce company uses Amazon RDS MySQL DB to store the data. The analytics department at the company runs its reports on the same database. The engineering team has noticed sluggish performance on the database when the analytics reporting process is in progress.

As an AWS Certified Solutions Architect - Associate, which of the following would you suggest as the MOST cost-optimal solution to improve the performance?

Create a read-replica with half compute capacity and half storage capacity as the primary. Point the reporting queries to run against the read replica

Correct answer

Create a read-replica with the same compute capacity and the same storage capacity as the primary. Point the reporting queries to run against the read replica

Create a standby instance in a multi-AZ configuration with half compute capacity and half storage capacity as the primary. Point the reporting queries to run against the standby instance

Create a standby instance in a multi-AZ configuration with the same compute capacity and the same storage capacity as the primary. Point the reporting queries to run against the standby instance

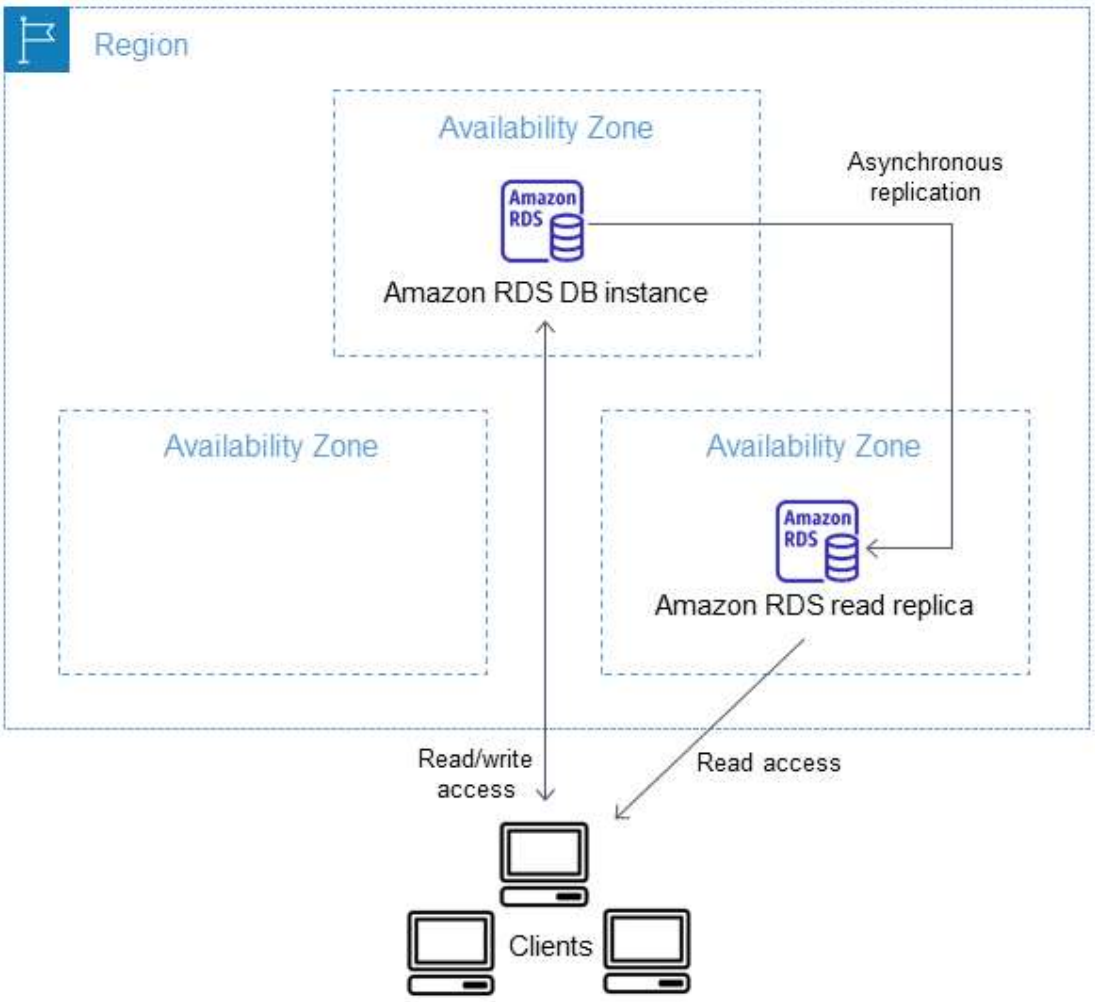
Overall explanation

Correct option:

Create a read-replica with the same compute capacity and the same storage capacity as the primary. Point the reporting queries to run against the read replica

Amazon RDS uses the MariaDB, Microsoft SQL Server, MySQL, Oracle, and PostgreSQL DB engines' built-in replication functionality to create a special type of database instance called a read replica from a source database instance. The source database instance becomes the primary database instance. Updates made to the primary database instance are asynchronously copied to the read replica. You can reduce the load on your primary DB instance by routing read queries from your applications to the read replica.

Amazon RDS Read Replicas:



via - https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER_ReadRepl.html

Overview of Amazon RDS read replicas

Deploying one or more read replicas for a given source DB instance might make sense in a variety of scenarios, including the following:

- Scaling beyond the compute or I/O capacity of a single DB instance for read-heavy database workloads. You can direct this excess read traffic to one or more read replicas.
- Serving read traffic while the source DB instance is unavailable. In some cases, your source DB instance might not be able to take I/O requests, for example due to I/O suspension for backups or scheduled maintenance. In these cases, you can direct read traffic to your read replicas. For this use case, keep in mind that the data on the read replica might be "stale" because the source DB instance is unavailable.
- Business reporting or data warehousing scenarios where you might want business reporting queries to run against a read replica, rather than your production DB instance.
- Implementing disaster recovery. You can promote a read replica to a standalone instance as a disaster recovery solution if the primary DB instance fails.

By default, a read replica is created with the same storage type as the source DB instance. However, you can create a read replica that has a different storage type from the source DB instance based on the options listed in the following table.

via - https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER_ReadRepl.html

You can use read replicas to improve the performance of your Amazon RDS MySQL DB by handling business reporting or data warehousing scenarios where you might want business

reporting queries to run against your read replica, rather than your production database instance.

You can create up to five read replicas from one DB instance. For replication to operate effectively, each read replica should have the same amount of compute and storage resources as the source database instance. If you scale the source database instance, also scale the read replicas.

Configuring read replicas with MySQL

Before a MySQL DB instance can serve as a replication source, make sure to enable automatic backups on the source DB instance. To do this, set the backup retention period to a value other than 0. This requirement also applies to a read replica that is the source DB instance for another read replica. Automatic backups are supported for read replicas running any version of MySQL. You can configure replication based on binary log coordinates for a MySQL DB instance.

On RDS for MySQL version 5.7.23 and higher MySQL 5.7 versions and RDS for MySQL 8.0.26 and higher 8.0 versions, you can configure replication using global transaction identifiers (GTIDs). For more information, see Using GTID-based replication for Amazon RDS for MySQL.

You can create up to five read replicas from one DB instance. For replication to operate effectively, each read replica should have the same amount of compute and storage resources as the source DB instance. If you scale the source DB instance, also scale the read replicas.

If a read replica is running any version of MySQL, you can specify it as the source DB instance for another read replica. For example, you can create ReadReplica1 from MyDBInstance, and then create ReadReplica2 from ReadReplica1. Updates made to MyDBInstance are replicated to ReadReplica1 and then replicated from ReadReplica1 to ReadReplica2. You can't have more than four instances involved in a replication chain. For example, you can create ReadReplica1 from MySourceDBInstance, and then create ReadReplica2 from ReadReplica1, and then create ReadReplica3 from ReadReplica2, but you can't create a ReadReplica4 from ReadReplica3.

If you promote a MySQL read replica that is in turn replicating to other read replicas, those read replicas remain active. Consider an example where MyDBInstance1 replicates to MyDBInstance2, and MyDBInstance2 replicates to MyDBInstance3. If you promote MyDBInstance2, replication from MyDBInstance1 to MyDBInstance2 no longer occurs, but MyDBInstance2 still replicates to MyDBInstance3.

To enable automatic backups on a read replica for RDS for MySQL, first create the read replica. Then modify the read replica to enable automatic backups.

You can run multiple read replica create or delete actions at the same time that reference the same source DB instance. To do this, stay within the limit of five read replicas for each source instance.

A read replica of a MySQL DB instance can't use a lower DB engine version than its source DB instance.

via
- https://docs.amazonaws.cn/en_us/AmazonRDS/latest/UserGuide/USER_MySQL.Replication.ReadReplicas.html

Incorrect options:

Create a read-replica with half compute capacity and half storage capacity as the primary. Point the reporting queries to run against the read replica - As mentioned in the explanation

above, you should create a read-replica with the same compute capacity and the same storage capacity as the primary.

Create a standby instance in a multi-AZ configuration with the same compute capacity and the same storage capacity as the primary. Point the reporting queries to run against the standby instance

Create a standby instance in a multi-AZ configuration with half compute capacity and half storage capacity as the primary. Point the reporting queries to run against the standby instance

Multi-AZ deployments are not a read scaling solution, so you cannot use a standby to serve read traffic. The standby is there just for failover. Hence both these options are incorrect.

References:

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

https://docs.amazonaws.cn/en_us/AmazonRDS/latest/UserGuide/USER_MySQL.Replication.ReadReplicas.html

Domain

Design Resilient Architectures

Question 57Skipped

The engineering team at a startup is evaluating the most optimal block storage volume type for the Amazon EC2 instances hosting its flagship application. The storage volume should support very low latency but it does not need to persist the data when the instance terminates. As a solutions architect, you have proposed using Instance Store volumes to meet these requirements.

Which of the following would you identify as the key characteristics of the Instance Store volumes? (Select two)

Instance store is reset when you stop or terminate an instance. Instance store data is preserved during hibernation

Correct selection

If you create an Amazon Machine Image (AMI) from an instance, the data on its instance store volumes isn't preserved

Correct selection

You can't detach an instance store volume from one instance and attach it to a different instance

You can specify instance store volumes for an instance when you launch or restart it

An instance store is a network storage type

Overall explanation

Correct options:

You can't detach an instance store volume from one instance and attach it to a different instance

You can specify instance store volumes for an instance only when you launch it. You can't detach an instance store volume from one instance and attach it to a different instance. The data in an instance store persists only during the lifetime of its associated instance. If an instance reboots (intentionally or unintentionally), data in the instance store persists.

If you create an Amazon Machine Image (AMI) from an instance, the data on its instance store volumes isn't preserved

If you create an AMI from an instance, the data on its instance store volumes isn't preserved and isn't present on the instance store volumes of the instances that you launch from the AMI.

Incorrect options:

Instance store is reset when you stop or terminate an instance. Instance store data is preserved during hibernation - When you stop, hibernate, or terminate an instance, every block of storage in the instance store is reset. Therefore, this option is incorrect.

You can specify instance store volumes for an instance when you launch or restart it - You can specify instance store volumes for an instance only when you launch it.

An instance store is a network storage type - An instance store provides temporary block-level storage for your instance. This storage is located on disks that are physically attached to the host computer.

Reference:

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

Domain

Design High-Performing Architectures

Question 58Skipped

A company helps its customers legally sign highly confidential contracts. To meet the strong industry requirements, the company must ensure that the signed contracts are encrypted using the company's proprietary algorithm. The company is now migrating to AWS Cloud using Amazon Simple Storage Service (Amazon S3) and would like you, the solution architect, to advise them on the encryption scheme to adopt.

What do you recommend?

Server-side encryption with customer-provided keys (SSE-C)

Server-side encryption with Amazon S3 managed keys (SSE-S3)

Correct answer

Client Side Encryption

Server-side encryption with AWS KMS keys (SSE-KMS)

Overall explanation

Correct option:

Client Side Encryption

Client-side encryption is the act of encrypting your data locally to help ensure its security in transit and at rest. To encrypt your objects before you send them to Amazon S3, use the Amazon S3 Encryption Client. When your objects are encrypted in this manner, your objects aren't exposed to any third party, including AWS. Amazon S3 receives your objects already encrypted; Amazon S3 does not play a role in encrypting or decrypting your objects. You can use both the Amazon S3 Encryption Client and server-side encryption to encrypt your data. When you send encrypted objects to Amazon S3, Amazon S3 doesn't recognize the objects as being encrypted, it only detects typical objects.

Incorrect options:

Server-side encryption with AWS KMS keys (SSE-KMS) - AWS Key Management Service (AWS KMS) is a service that combines secure, highly available hardware and software to provide a key management system scaled for the cloud. When you use server-side encryption with AWS KMS (SSE-KMS), you can specify a customer-managed CMK that you have already created. SSE-KMS provides you with an audit trail that shows when your CMK was used and by whom.

Server-side encryption with Amazon S3 managed keys (SSE-S3) - When you use Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3), each object is encrypted with a unique key.

Server-side encryption with customer-provided keys (SSE-C) - With Server-Side Encryption with Customer-Provided Keys (SSE-C), you manage the encryption keys and Amazon S3 manages the encryption, as it writes to disks, and decryption when you access your objects.

References:

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

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

https://docs.aws.amazon.com/AmazonS3/latest/dev/serv-side-encryption.html

Domain

Design Secure Architectures

Question 59Skipped

A company has moved its business critical data to Amazon Elastic File System (Amazon EFS) which will be accessed by multiple Amazon EC2 instances.

As an AWS Certified Solutions Architect - Associate, which of the following would you recommend to exercise access control such that only the permitted Amazon EC2 instances can read from the Amazon EFS file system? (Select two)

Set up the IAM policy root credentials to control and configure the clients accessing the Amazon EFS file system

Correct selection

Use VPC security groups to control the network traffic to and from your file system

Use Amazon GuardDuty to curb unwanted access to Amazon EFS file system

Correct selection

Use an IAM policy to control access for clients who can mount your file system with the required permissions

Use network access control list (network ACL) to control the network traffic to and from your Amazon EC2 instance

Overall explanation

Correct options:

Use VPC security groups to control the network traffic to and from your file system

Use an IAM policy to control access for clients who can mount your file system with the required permissions

You control which Amazon EC2 instances can access your Amazon EFS file system by using VPC security group rules and AWS Identity and Access Management (IAM) policies. Use VPC security groups to control the network traffic to and from your file system. Attach an IAM policy to your file system to control which clients can mount your file system and with what permissions, and you may use Amazon EFS Access Points to manage application access. Control access to files and directories with POSIX-compliant user and group-level permissions.

Files and directories in an Amazon EFS file system support standard Unix-style read, write, and execute permissions based on the user ID and group IDs. When an NFS client mounts an Amazon EFS file system without using an access point, the user ID and group ID provided by the client is trusted. You can also use Amazon EFS access points to override user ID and group IDs used by the NFS client. When users attempt to access files and directories, Amazon EFS checks their user IDs and group IDs to verify that each user has permission to access the objects.

Incorrect options:

Use network access control list (network ACL) to control the network traffic to and from your Amazon EC2 instance - Network ACLs operate at the subnet level and not at the instance level.

Set up the IAM policy root credentials to control and configure the clients accessing the Amazon EFS file system - There is no such thing as an IAM policy root credentials and this statement has been added as a distractor.

Use Amazon GuardDuty to curb unwanted access to Amazon EFS file system - Amazon GuardDuty is a threat detection service that continuously monitors for malicious activity and unauthorized behavior to protect your AWS accounts, workloads, and data stored in Amazon S3. It cannot be used for access control to the Amazon EFS file system.

References:

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

https://docs.aws.amazon.com/efs/latest/ug/accessing-fs-nfs-permissions.html

https://docs.aws.amazon.com/efs/latest/ug/iam-access-control-nfs-efs.html

Domain

Design Secure Architectures

Question 60Skipped

You have deployed a database technology that has a synchronous replication mode to survive disasters in data centers. The database is therefore deployed on two Amazon EC2 instances in two Availability Zones (AZs). The database must be publicly available so you have deployed the Amazon EC2 instances in public subnets. The replication protocol currently uses the Amazon EC2 public IP addresses.

What can you do to decrease the replication cost?

Use an Elastic Fabric Adapter (EFA)

Assign elastic IP address (EIP) to the Amazon EC2 instances and use them for the replication

Correct answer

Use the Amazon EC2 instances private IP for the replication

Create a Private Link between the two Amazon EC2 instances

Overall explanation

Correct option:

Use the Amazon EC2 instances private IP for the replication

The source of the cost is that traffic between two EC2 instances is going over the public internet, thus incurring high costs. Here, the correct answer is to use Private IP, so that the network remains private, for a minimal cost.

Incorrect options:

Assign elastic IP address (EIP) to the Amazon EC2 instances and use them for the replication - Using Elastic IPs will not solve the problem as the traffic will still be going over the public internet.

Create a Private Link between the two Amazon EC2 instances - AWS PrivateLink simplifies the security of data shared with cloud-based applications by eliminating the exposure of data to the public Internet. AWS PrivateLink provides private connectivity between VPCs, AWS services, and on-premises applications, securely on the Amazon network.

Private Link is a distractor in this question. Private Link is leveraged to create a private connection between an application that is fronted by an NLB in an account, and an Elastic Network Interface (ENI) in another account, without the need of VPC peering and allowing the connections between the two to remain within the AWS network.

Use an Elastic Fabric Adapter (EFA) - The Elastic Fabric Adapter (EFA) is a network interface for Amazon EC2 instances that enables customers to run HPC applications requiring high levels of

inter-instance communications, like computational fluid dynamics, weather modeling, and reservoir simulation, at scale on AWS. This option is not relevant to the given use-case.

References:

https://aws.amazon.com/privatelink/

https://aws.amazon.com/hpc/efa/

Domain

Design Cost-Optimized Architectures

Question 61Skipped

You are using AWS Lambda to implement a batch job for a big data analytics workflow. Based on historical trends, a similar job runs for 30 minutes on average. The AWS Lambda function pulls data from Amazon S3, processes it, and then writes the results back to Amazon S3. When you deployed your AWS Lambda function, you noticed an issue where the AWS Lambda function abruptly failed after 15 minutes of execution.

As a solutions architect, which of the following would you identify as the root cause of the issue?

The AWS Lambda function chosen runtime is wrong

The AWS Lambda function is running out of memory

The AWS Lambda function is missing IAM permissions

Correct answer

The AWS Lambda function is timing out

Overall explanation

Correct option:

AWS Lambda lets you run code without provisioning or managing servers. You pay only for the compute time you consume.

With AWS Lambda, you can run code for virtually any type of application or backend service - all with zero administration. Just upload your code and Lambda takes care of everything required to run and scale your code with high availability. You can set up your code to automatically trigger from other AWS services or call it directly from any web or mobile app. AWS Lambda functions can be configured to run up to 15 minutes per execution. You can set the timeout to any value between 1 second and 15 minutes.

The AWS Lambda function is timing out

AWS Lambda functions time out after 15 minutes, and are not usually meant for long-running jobs.

Incorrect options:

The AWS Lambda function is running out of memory - Memory errors will not result in the abrupt termination of the function with no error message.

The AWS Lambda function chosen runtime is wrong - AWS Lambda function execution will fail if there is an issue with runtime. So, this is not the issue in the current case.

The AWS Lambda function is missing IAM permissions - Without enough permissions, AWS Lambda would not have been able to start its execution at all. So, permissions are not an issue here.

Reference:

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

Domain

Design High-Performing Architectures

Question 62Skipped

The engineering team at a retail company is planning to migrate to AWS Cloud from the on-premises data center. The team is evaluating Amazon Relational Database Service (Amazon RDS) as the database tier for its flagship application. The team has hired you as an AWS Certified Solutions Architect Associate to advise on Amazon RDS Multi-AZ capabilities.

Which of the following would you identify as correct for Amazon RDS Multi-AZ? (Select two)

For automated backups, I/O activity is suspended on your primary database since backups are not taken from standby database

Correct selection

Amazon RDS applies operating system updates by performing maintenance on the standby, then promoting the standby to primary and finally performing maintenance on the old primary, which becomes the new standby

Correct selection

Amazon RDS automatically initiates a failover to the standby, in case primary database fails for any reason

To enhance read scalability, a Multi-AZ standby instance can be used to serve read requests

Updates to your database Instance are asynchronously replicated across the Availability Zone to the standby in order to keep both in sync

Overall explanation

Correct options:

Amazon RDS applies operating system updates by performing maintenance on the standby, then promoting the standby to primary and finally performing maintenance on the old primary, which becomes the new standby

Running a DB instance as a Multi-AZ deployment can further reduce the impact of a maintenance event because Amazon RDS applies operating system updates by following these steps:

Perform maintenance on the standby.

Promote the standby to primary.

Perform maintenance on the old primary, which becomes the new standby.

When you modify the database engine for your DB instance in a Multi-AZ deployment, then Amazon RDS upgrades both the primary and secondary DB instances at the same time. In this case, the database engine for the entire Multi-AZ deployment is shut down during the upgrade.

Amazon RDS automatically initiates a failover to the standby, in case primary database fails for any reason

You also benefit from enhanced database availability when running your DB instance as a Multi-AZ deployment. If an Availability Zone failure or DB instance failure occurs, your availability impact is limited to the time automatic failover takes to complete.

Another implied benefit of running your DB instance as a Multi-AZ deployment is that DB instance failover is automatic and requires no administration. In an Amazon RDS context, this means you are not required to monitor DB instance events and initiate manual DB instance recovery in the event of an Availability Zone failure or DB instance failure.

Incorrect options:

For automated backups, I/O activity is suspended on your primary database since backups are not taken from standby database - The availability benefits of Multi-AZ also extend to planned maintenance. For example, with automated backups, I/O activity is no longer suspended on your primary during your preferred backup window, since backups are taken from the standby.

To enhance read scalability, a Multi-AZ standby instance can be used to serve read requests - A Multi-AZ standby cannot serve read requests. Multi-AZ deployments are designed to provide enhanced database availability and durability, rather than read scaling benefits. As such, the feature uses synchronous replication between primary and standby. AWS implementation makes sure the primary and the standby are constantly in sync, but precludes using the standby for read or write operations.

Updates to your database Instance are asynchronously replicated across the Availability Zone to the standby in order to keep both in sync - When you create your DB instance to run as a Multi-AZ deployment, Amazon RDS automatically provisions and maintains a synchronous "standby" replica in a different Availability Zone. Updates to your DB Instance are synchronously replicated across the Availability Zone to the standby in order to keep both in sync and protect your latest database updates against DB instance failure.

Reference:

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

Domain

Design Resilient Architectures

Question 63Skipped

A company is experiencing stability issues with their cluster of self-managed RabbitMQ message brokers and the company now wants to explore an alternate solution on AWS.

As a solutions architect, which of the following AWS services would you recommend that can provide support for quick and easy migration from RabbitMQ?

Amazon SQS FIFO (First-In-First-Out)

Amazon Simple Queue Service (Amazon SQS) Standard

Correct answer

Amazon MQ

Amazon Simple Notification Service (Amazon SNS)

Overall explanation

Correct option:

Amazon MQ

Amazon MQ is a managed message broker service for Apache ActiveMQ that makes it easy to set up and operate message brokers in the cloud. Message brokers allow different software systems–often using different programming languages, and on different platforms–to communicate and exchange information. If an organization is using messaging with existing applications and wants to move the messaging service to the cloud quickly and easily, AWS recommends Amazon MQ for such a use case. So this is the correct option.

Incorrect options:

Amazon Simple Notification Service (Amazon SNS) - Amazon Simple Notification Service (Amazon SNS) is a highly available, durable, secure, fully managed pub/sub messaging service that enables you to decouple microservices, distributed systems, and serverless applications. Amazon SNS provides topics for high-throughput, push-based, many-to-many messaging. SNS does not provide support for migration from RabbitMQ as its a fully managed pub/sub messaging service. Hence this option is incorrect.

Amazon Simple Queue Service (Amazon SQS) Standard - Amazon SQS Standard offers a reliable, highly scalable hosted queue for storing messages as they travel between computers. Amazon SQS lets you easily move data between distributed application components and helps you build applications in which messages are processed independently (with message-level ack/fail semantics), such as automated workflows. SQS Standard does not provide support for migration from RabbitMQ. Hence this option is incorrect.

Amazon SQS FIFO (First-In-First-Out) - Amazon SQS FIFO (First-In-First-Out) has all the capabilities of the standard queue. They are used when the order of operations and events is critical, or where duplicates can't be tolerated. SQS FIFO does not provide support for migration from RabbitMQ. Hence this option is incorrect.

Reference:

https://aws.amazon.com/amazon-mq/

https://aws.amazon.com/blogs/compute/migrating-from-rabbitmq-to-amazon-mq/

Domain

Design Resilient Architectures

Question 64Skipped

To support critical production workloads that require maximum resiliency, a company wants to configure network connections between its Amazon VPC and the on-premises infrastructure. The company needs AWS Direct Connect connections with speeds greater than 1 Gbps.

As a solutions architect, which of the following will you suggest as the best architecture for this requirement?

Correct answer

Opt for two separate AWS Direct Connect connections terminating on separate devices in more than one Direct Connect location

Use AWS Managed VPN as a backup for AWS Direct Connect connections to ensure maximum resiliency

Opt for one AWS Direct Connect connection at each of the multiple Direct Connect locations

Opt for at least two AWS Direct Connect connections terminating on different devices at a single Direct Connect location

Overall explanation

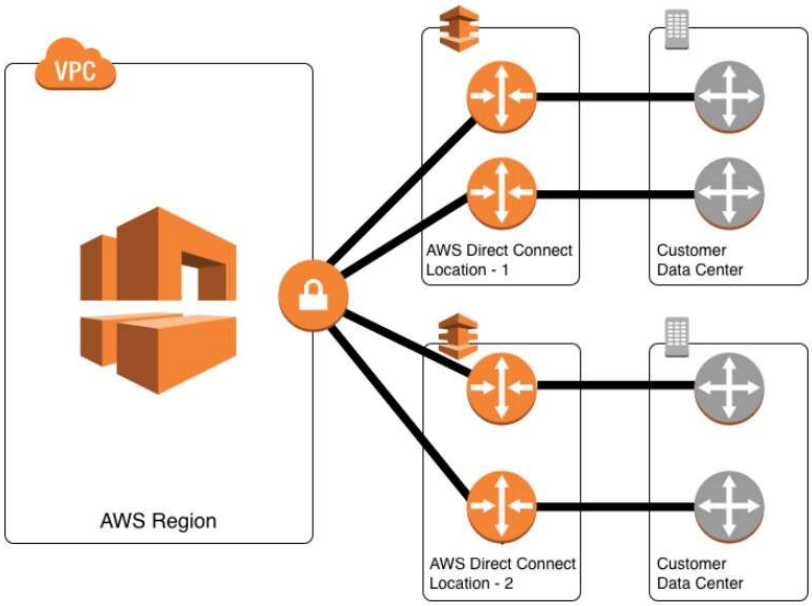
Correct option:

Opt for two separate AWS Direct Connect connections terminating on separate devices in more than one Direct Connect location

Maximum resilience is achieved by separate connections terminating on separate devices in more than one location. This configuration offers customers maximum resilience to failure. As shown in the figure above, such a topology provides resilience to device failure, connectivity failure, and complete location failure. You can use Direct Connect Gateway to access any AWS Region (except AWS Regions in China) from any AWS Direct Connect locations.

Maximum Resiliency for Critical Workloads:

Maximum Resiliency for Critical Workloads



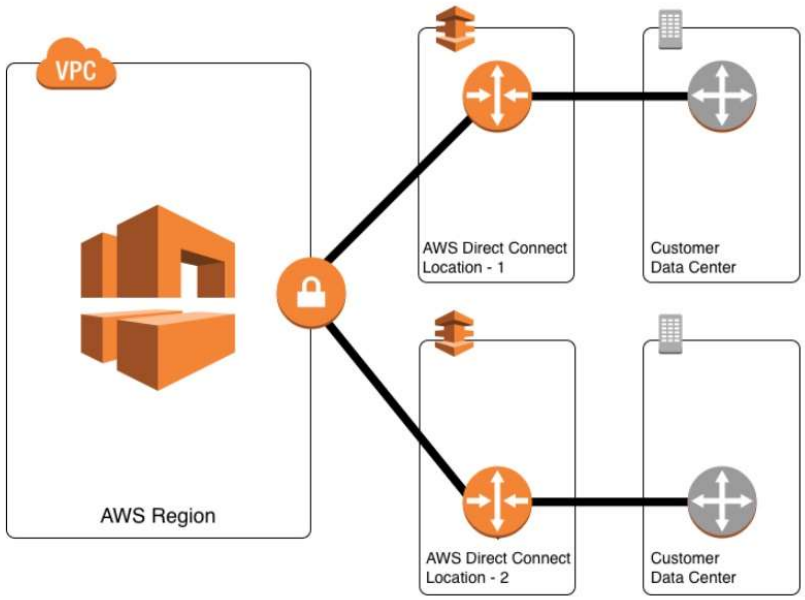
via - https://aws.amazon.com/directconnect/resiliency-recommendation/

Incorrect options:

Opt for one AWS Direct Connect connection at each of the multiple Direct Connect locations - For critical production workloads that require high resiliency, it is recommended to have one connection at multiple locations. As shown in the figure below, such a topology ensures resilience to connectivity failure due to a fiber cut or a device failure as well as a complete location failure. You can use Direct Connect Gateway to access any AWS Region (except AWS Regions in China) from any AWS Direct Connect location.

High Resiliency for Critical Workloads:

High Resiliency for Critical Workloads

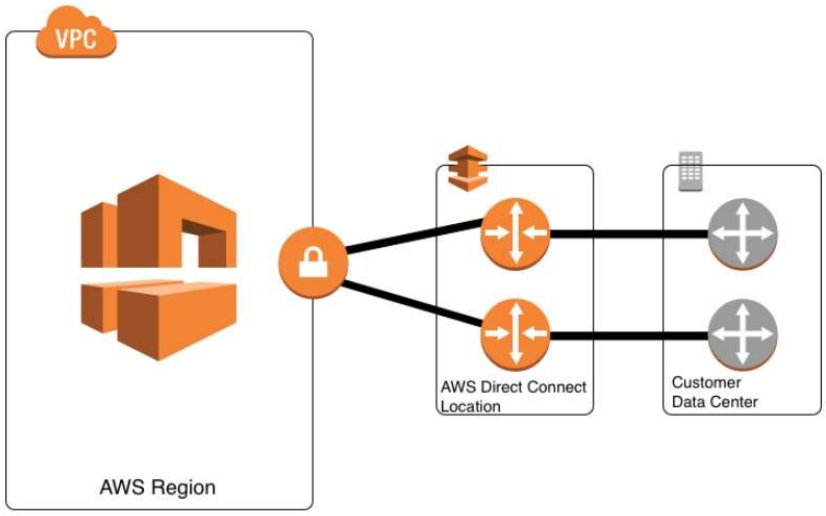


via - https://aws.amazon.com/directconnect/resiliency-recommendation/

Opt for at least two AWS Direct Connect connections terminating on different devices at a single Direct Connect location - For non-critical production workloads and development workloads that do not require high resiliency, it is recommended to have at least two connections terminating on different devices at a single location. As shown in the figure above, such a topology helps in the case of the device failure at a location but does not help in the event of a total location failure.

Non Critical Production Workloads or Development Workloads:

Non Critical Production Workloads or Development Workloads



via - https://aws.amazon.com/directconnect/resiliency-recommendation/

Use AWS Managed VPN as a backup for AWS Direct Connect connections to ensure maximum resiliency - It is important to understand that AWS Managed VPN supports up to 1.25 Gbps throughput per VPN tunnel and does not support Equal Cost Multi-Path (ECMP) for egress data path in the case of multiple AWS Managed VPN tunnels terminating on the same VGW. Thus, AWS does not recommend customers use AWS Managed VPN as a backup for AWS Direct Connect connections with speeds greater than 1 Gbps.

Reference:

https://aws.amazon.com/directconnect/resiliency-recommendation/

Domain

Design Resilient Architectures

Question 65Skipped

The DevOps team at an e-commerce company has deployed a fleet of Amazon EC2 instances under an Auto Scaling group (ASG). The instances under the ASG span two Availability Zones (AZ) within the us-east-1 region. All the incoming requests are handled by an Application Load Balancer (ALB) that routes the requests to the Amazon EC2 instances under the Auto Scaling Group. As part of a test run, two instances (instance 1 and 2, belonging to AZ A) were manually terminated by the DevOps team causing the Availability Zones (AZ) to have unbalanced resources. Later that day, another instance (belonging to AZ B) was detected as unhealthy by the Application Load Balancer's health check.

Can you identify the correct outcomes for these events? (Select two)

Correct selection

As the resources are unbalanced in the Availability Zones, Amazon EC2 Auto Scaling will compensate by rebalancing the Availability Zones. When rebalancing, Amazon EC2 Auto Scaling launches new instances before terminating the old ones, so that rebalancing does not compromise the performance or availability of your application

Amazon EC2 Auto Scaling creates a new scaling activity to terminate the unhealthy instance and launch the new instance simultaneously

Amazon EC2 Auto Scaling creates a new scaling activity for launching a new instance to replace the unhealthy instance. Later, Amazon EC2 Auto Scaling creates a new scaling activity for terminating the unhealthy instance and then terminates it

Correct selection

Amazon EC2 Auto Scaling creates a new scaling activity for terminating the unhealthy instance and then terminates it. Later, another scaling activity launches a new instance to replace the terminated instance

As the resources are unbalanced in the Availability Zones, Amazon EC2 Auto Scaling will compensate by rebalancing the Availability Zones. When rebalancing, Amazon EC2 Auto Scaling terminates old instances before launching new instances, so that rebalancing does not cause extra instances to be launched

Overall explanation

Correct options:

As the resources are unbalanced in the Availability Zones, Amazon EC2 Auto Scaling will compensate by rebalancing the Availability Zones. When rebalancing, Amazon EC2 Auto Scaling launches new instances before terminating the old ones, so that rebalancing does not compromise the performance or availability of your application

Amazon EC2 Auto Scaling helps you ensure that you have the correct number of Amazon EC2 instances available to handle the load for your application. You create collections of EC2 instances, called Auto Scaling groups. You can specify the minimum number of instances in each Auto Scaling group, and Amazon EC2 Auto Scaling ensures that your group never goes below this size. Actions such as changing the Availability Zones (AZ) for your group or explicitly terminating or detaching instances can lead to the Auto Scaling group becoming unbalanced between Availability Zones. Amazon EC2 Auto Scaling compensates by rebalancing the Availability Zones.

When rebalancing, Amazon EC2 Auto Scaling launches new instances before terminating the old ones, so that rebalancing does not compromise the performance or availability of your application. Therefore, this option is correct.

Availability Zone Rebalancing Overview:

Rebalancing Activities

After certain actions occur, your Auto Scaling group can become unbalanced between Availability Zones. Amazon EC2 Auto Scaling compensates by rebalancing the Availability Zones. The following actions can lead to rebalancing activity:

- You change the Availability Zones for your group.
- You explicitly terminate or detach instances and the group becomes unbalanced.
- An Availability Zone that previously had insufficient capacity recovers and has additional capacity available.
- An Availability Zone that previously had a Spot price above your maximum price now has a Spot price below your maximum price.

When rebalancing, Amazon EC2 Auto Scaling launches new instances before terminating the old ones, so that rebalancing does not compromise the performance or availability of your application.

Because Amazon EC2 Auto Scaling attempts to launch new instances before terminating the old ones, being at or near the specified maximum capacity could impede or completely halt rebalancing activities. To avoid this problem, the system can temporarily exceed the specified maximum capacity of a group by a 10 percent margin (or by a 1-instance margin, whichever is greater) during a rebalancing activity. The margin is extended only if the group is at or near maximum capacity and needs rebalancing, either because of user-requested rezoning or to compensate for zone availability issues. The extension lasts only as long as needed to rebalance the group typically a few minutes.

via - https://docs.aws.amazon.com/autoscaling/ec2/userguide/auto-scaling-benefits.html

Amazon EC2 Auto Scaling creates a new scaling activity for terminating the unhealthy instance and then terminates it. Later, another scaling activity launches a new instance to replace the terminated instance

However, the scaling activity of Auto Scaling works in a different sequence compared to the rebalancing activity. Auto Scaling creates a new scaling activity for terminating the unhealthy instance and then terminates it. Later, another scaling activity launches a new instance to replace the terminated instance.

Incorrect options:

Amazon EC2 Auto Scaling creates a new scaling activity for launching a new instance to replace the unhealthy instance. Later, Amazon EC2 Auto Scaling creates a new scaling activity for terminating the unhealthy instance and then terminates it - This option contradicts the correct sequence of events outlined earlier for scaling activity created by Amazon EC2 Auto Scaling. Actually, Auto Scaling first terminates the unhealthy instance and then launches a new instance. Hence this is incorrect.

As the resources are unbalanced in the Availability Zones, Amazon EC2 Auto Scaling will compensate by rebalancing the Availability Zones. When rebalancing, Amazon EC2 Auto Scaling terminates old instances before launching new instances, so that rebalancing does not cause extra instances to be launched - This option contradicts the correct sequence of events outlined earlier for rebalancing activity. When rebalancing, Amazon EC2 Auto Scaling launches new instances before terminating the old ones. Hence this is incorrect.

Amazon EC2 Auto Scaling creates a new scaling activity to terminate the unhealthy instance and launch the new instance simultaneously - This is a made-up option as both the terminate and launch activities can't happen simultaneously. This option has been added as a distractor.

References:

https://docs.aws.amazon.com/autoscaling/ec2/userguide/what-is-amazon-ec2-auto-scaling.html

https://docs.aws.amazon.com/autoscaling/ec2/userguide/auto-scaling-benefits.html

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

https://docs.aws.amazon.com/autoscaling/ec2/userguide/healthcheck.html

Domain

Design High-Performing Architectures