###########################

**S3 questions**

###########################

 **S3 bucket policy**

* An S3 bucket policy is a resource-based AWS Identity and Access Management (IAM) policy that you can attach to an S3 bucket to control access to the bucket and the objects within it. Policies can define permissions for specific actions, such as read or write access, and can specify conditions such as IP address or time of access.

 **Is there a way to upload a file that is greater than 100 megabytes in Amazon S3?**

* Yes, you can use the multipart upload feature to upload files larger than 100 MB. Multipart upload allows you to split the file into smaller parts, upload each part independently, and then combine them into a single object in S3.

 **What is the maximum number of S3 buckets you can create?**

* By default, you can create up to 100 S3 buckets per AWS account. This limit can be increased by submitting a request to AWS Support.

 **What is Amazon S3? Elaborate.**

* Amazon S3 (Simple Storage Service) is a scalable object storage service that allows you to store and retrieve any amount of data, at any time, from anywhere on the web. It is designed to provide high durability, availability, and scalability for a wide range of data types, including backups, archives, and big data analytics.

 **S3 encryption policy**

* S3 encryption policies define how data is encrypted when stored in S3. You can use server-side encryption (SSE) with S3 to encrypt your data at rest. AWS offers several options for SSE: SSE-S3 (using S3-managed keys), SSE-KMS (using AWS Key Management Service), and SSE-C (using customer-provided keys).

 **How to copy local file to S3 using CLI**

* To copy a local file to an S3 bucket using the AWS CLI, use the following command:

bash

Copy code

aws s3 cp localfile.txt s3://your-bucket-name/

 **AWS CLI command to copy a local file to an S3 bucket in AWS.**

* Same as the previous answer. The command is:

bash

Copy code

aws s3 cp localfile.txt s3://your-bucket-name/

 **What are S3 storage classes, and explain various types of S3 storage classes?**

* S3 offers several storage classes, each designed for different use cases:
  + **STANDARD**: High durability, availability, and performance for frequently accessed data.
  + **INTELLIGENT\_TIERING**: Optimizes costs by automatically moving data between frequent and infrequent access tiers.
  + **ONEZONE\_IA**: Lower-cost option for infrequent access data that doesn’t require multiple availability zone redundancy.
  + **GLACIER**: Low-cost storage for archival data with retrieval times ranging from minutes to hours.
  + **GLACIER DEEP ARCHIVE**: Lowest-cost storage for long-term data archiving with retrieval times in hours.
  + **REDUCED\_REDUNDANCY STORAGE (RRS)**: Lower durability compared to STANDARD, intended for non-critical data.

 **AWS S3 Multi-part upload.**

* Multipart upload allows you to upload large objects in parts, which can improve upload performance and reliability. You can upload parts in parallel, and if a part fails, you only need to retry that part, not the whole object.

 **Which one of the storage solutions offered by AWS would you use if you need extremely low pricing and data archiving?**

* For extremely low pricing and data archiving, you would use **S3 Glacier** or **S3 Glacier Deep Archive**. These classes are designed for long-term storage and offer significant cost savings compared to other storage classes.

 **In S3, the difference between public and private subnet.**

* This question pertains more to networking than S3. In AWS, a public subnet allows resources to directly access the internet, whereas a private subnet does not. S3 buckets can be configured to allow or deny public access regardless of the subnet configuration.

 **AWS CLI command to copy a local file to an S3 bucket in AWS.**

* Same as the previous answers. The command is:

bash

Copy code

aws s3 cp localfile.txt s3://your-bucket-name/

 **How will you configure an Amazon S3 bucket to serve static assets for your public web application?**

* To configure an S3 bucket to serve static assets, enable static website hosting in the S3 bucket properties. Set the index document (e.g., index.html) and optionally an error document. Make sure the bucket policy allows public read access to the files.

 **What do you know about the cross-region replication service offered by AWS?**

* Cross-Region Replication (CRR) automatically replicates objects from one S3 bucket to another in a different AWS region. It is useful for improving data durability, meeting regulatory requirements, and reducing latency for users in different geographical locations.

 **AWS S3 Multi-part upload.**

* Already covered in question 9.

 **Do you know what are the storage classes available for S3?**

* Already covered in question 8.

 **What is an S3 bucket? How many S3 buckets can we create in an AWS account?**

* An S3 bucket is a container for storing objects (files) in S3. By default, you can create up to 100 S3 buckets per AWS account.

 **You have to upload a file of around 120 megabytes in Amazon S3. How will you approach the uploading of this file?**

* For a 120 MB file, you should use multipart upload, which is suitable for files larger than 100 MB. This approach will allow you to upload the file in parts and combine them into a single object.

 **Can you explain versioning in an S3 bucket?**

* Versioning in S3 allows you to keep multiple versions of an object in a bucket. When versioning is enabled, each time you upload an object with the same key, a new version is created. This feature helps protect against accidental deletions or overwrites.

 **What is S3? Can we use CDN in front of S3?**

* S3 is an object storage service for storing and retrieving data. Yes, you can use Amazon CloudFront, a Content Delivery Network (CDN) service, in front of S3 to deliver content with low latency and high transfer speeds.

 **What is block storage and object storage? What do you mean by object storage?**

* Block storage is a type of storage where data is stored in fixed-sized blocks, which can be accessed independently. It is often used for databases and applications requiring low-latency access. Object storage, like S3, stores data as objects with a unique identifier. It is ideal for unstructured data such as files, images, and backups.

 **Which storage provides redundancy across the data center? S3**

* Amazon S3 provides redundancy across multiple data centers (availability zones) within a region, ensuring high durability and availability of stored data.

 **How can you monitor S3 cross-region replication to ensure consistency without checking the bucket?**

* You can use Amazon CloudWatch to monitor replication metrics and set up alarms for any issues. Additionally, S3 provides replication metrics and status reports that can be monitored via the S3 console or AWS SDKs.

 **I want to upload a 120MB file in S3. How do I do it? Does multipart upload utility allow more than 100MB?**

* For a 120 MB file, use the multipart upload feature. Multipart upload supports files larger than 100 MB and allows you to upload parts of the file in parallel.

 **Can you use an S3 bucket with an EC2 instance?**

* Yes, you can use an S3 bucket with an EC2 instance. You can access S3 from an EC2 instance using the AWS CLI, SDKs, or directly via HTTP requests. IAM roles attached to the EC2 instance can be used to manage permissions for accessing S3.

 **Can we host static and dynamic websites on S3? Can we host dynamic websites using S3? What is static hosting in S3?**

* You can host static websites on S3 by enabling static website hosting in the bucket properties. S3 is not suitable for hosting dynamic websites because it does not support server-side scripting. For dynamic content, you would need to use other AWS services like EC2 or AWS Lambda in conjunction with API Gateway.

 **What is replication in S3? S3 cross-region replication? What is the default condition before doing cross-region replication?**

* Replication in S3 involves copying objects from one bucket to another. Cross-Region Replication (CRR) copies objects between buckets in different AWS regions. By default, CRR requires versioning to be enabled on both source and destination buckets.

 **What is the default storage class for the S3 bucket?**

* The default storage class for newly uploaded objects is **STANDARD** unless specified otherwise.

 **What are the S3 data encryption types?**

* S3 supports several encryption options:
  + **Server-Side Encryption with S3 Managed Keys (SSE-S3)**
  + **Server-Side Encryption with AWS Key Management Service (SSE-KMS)**
  + **Server-Side Encryption with Customer-Provided Keys (SSE-C)**
  + **Client-Side Encryption**

 **Tell me the steps to do cross-region replication.**

* Steps to configure cross-region replication:
  1. Enable versioning on both source and destination buckets.
  2. Create an IAM role with permissions for S3 replication.
  3. Configure the source bucket's replication configuration to specify the destination bucket and the replication role.
  4. Optionally, configure replication rules to specify which objects to replicate.

 **What are the lifecycles in S3?**

* Lifecycle policies in S3 allow you to automate the transition of objects to different storage classes and the expiration of objects. You can set rules to move objects to cheaper storage classes like Glacier or delete them after a certain period.

 **If a bucket is public, then how do you make it private?**

* To make a public bucket private, you can:
  1. Update the bucket policy to deny public access.
  2. Remove any existing public access permissions from the bucket ACL.
  3. Configure S3 Block Public Access settings to restrict public access.

 **For website hosting, which service do you use?**

* For hosting static websites, you use Amazon S3. For dynamic websites or applications, you might use Amazon EC2, AWS Elastic Beanstalk, or AWS Lambda.

 **If I want to deploy a static website in AWS, what solution would you recommend as a Cloud Engineer?**

* For deploying a static website, Amazon S3 is a recommended solution. You can use Amazon CloudFront as a CDN for better performance and availability.

 **How to take backup of S3 using a script?**

* You can use the AWS CLI to script backups. For example, you can use the aws s3 sync command to synchronize data from an S3 bucket to another location:

bash

Copy code

aws s3 sync s3://source-bucket /local-backup-folder

 **S3 data store limit and time? How will you delete an S3 bucket?**

* S3 does not impose a limit on the amount of data you can store. To delete an S3 bucket, ensure the bucket is empty and then use the AWS Management Console or CLI command:

bash

Copy code

aws s3 rb s3://your-bucket-name --force

 **How does S3 work? S3 architecture.**

* Amazon S3 works by storing data in a distributed manner across multiple servers and data centers (availability zones) to ensure high durability and availability. Data is stored in objects within buckets. S3 uses a flat namespace for buckets but a hierarchical namespace for objects within buckets.

 **In which class do you store data? Is it possible to store data in Glacier?**

* You can store data in various S3 storage classes depending on your needs. For archival and low-cost storage, you can use **S3 Glacier** or **S3 Glacier Deep Archive**.

 **Object lock feature in S3?**

* S3 Object Lock enables you to store objects using a write-once, read-many (WORM) model. This feature can be used to prevent objects from being deleted or overwritten for a fixed retention period or indefinitely.

 **How many S3 buckets are there in your organization, and what do you store in them? Share the flow of how lifecycle management is handled.**

* This question would be specific to your organization. Generally, S3 buckets might be used for different purposes such as backups, logs, or application data. Lifecycle management involves creating policies to transition data to cheaper storage classes or delete it based on predefined rules.

 **How do S3 classes cost less?**

* S3 storage classes cost less by optimizing the storage and retrieval costs based on the access patterns. For example, infrequent access and archival classes like ONEZONE\_IA, GLACIER, and GLACIER DEEP ARCHIVE offer lower storage costs compared to STANDARD.

 **What storage classes are available in S3?**

* Already covered in question 8.

 **Accessing S3 bucket in another AWS account with cross-account access?**

* To access an S3 bucket from another AWS account, you need to set up a bucket policy or an IAM policy that grants the necessary permissions to the other account. Cross-account access can be controlled using bucket policies, IAM roles, or ACLs.

 **How do you use S3 in your project?**

* This would be specific to your project. Typically, S3 might be used for storing application data, backups, static assets, or logs.

 **How can you send a request to Amazon S3?**

* You can send requests to S3 using the AWS CLI, AWS SDKs for various programming languages, or by making direct HTTP requests to the S3 REST API.

 **Mention what the difference between Amazon S3 and EC2 is?**

* Amazon S3 is an object storage service for storing and retrieving data, while Amazon EC2 is a compute service that provides resizable compute capacity in the cloud for running applications and services.

 **How many buckets can you create in AWS by default?**

* By default, you can create up to 100 S3 buckets per AWS account.

 **What do you know about the S3 transfer acceleration service offered by Amazon?**

* S3 Transfer Acceleration speeds up the upload and download of files to and from S3 by using Amazon CloudFront’s globally distributed edge locations. It helps reduce latency and improve transfer speeds, especially for geographically distant clients.

 **What are the storage classes available in Amazon S3?**

* Already covered in question 8.

 **A company is going to launch one branch in the UK and need to continue with its existing main branch in the USA. The company has almost 15 GB of data which is stored in an S3 Bucket in the Ohio region and data is stored with the default storage class. The Company also wants to provide its updated & stored data in the London S3 bucket using one zone accessibility storage class to save storage costs. In addition, the company also wants that the data must be updated automatically in S3’s London bucket; if any data is modified or written in the S3 bucket in Ohio.**

* To achieve this, you should configure Cross-Region Replication (CRR) from the Ohio bucket to the London bucket. Set the London bucket to use the **ONEZONE\_IA** storage class to save on costs. Ensure versioning is enabled on both buckets and configure the replication rules to automatically replicate changes from the Ohio bucket to the London bucket.

###########################

**IAM**

###########################

 **S3 encryption policy**

* An S3 encryption policy defines how data is encrypted when stored in S3. You can use server-side encryption (SSE) to automatically encrypt objects at rest using S3-managed keys (SSE-S3), AWS Key Management Service keys (SSE-KMS), or customer-provided keys (SSE-C). The policy ensures that data is protected from unauthorized access.

 **IAM Roles: Why are they used? How to improve security?**

* IAM Roles are used to grant temporary access to AWS resources without the need to share long-term credentials. They are often used for applications or services to interact with AWS resources. To improve security, use roles with the principle of least privilege, enable MFA, and regularly review and rotate credentials.

 **What is the difference between IAM role and IAM user?**

* An IAM user is an entity that represents a person or application with permanent credentials to interact with AWS resources. An IAM role is a temporary security credential that can be assumed by IAM users, applications, or AWS services to access resources based on permissions attached to the role.

 **What is power user access in AWS?**

* Power user access in AWS typically refers to IAM users or roles that have broad access to AWS services and resources, excluding management of IAM users and groups. This level of access allows users to perform most actions within the AWS account, except for managing IAM roles and policies.

 **What are the different IAM categories you can control?**

* IAM allows you to control the following categories:
  + **Users**: Individual entities with permissions.
  + **Groups**: Collections of users that share permissions.
  + **Roles**: Temporary permissions assumed by users or services.
  + **Policies**: Documents that define permissions for users, groups, and roles.

 **What is resource-based policy? (IAM policy)**

* A resource-based policy is a policy attached directly to a resource, such as an S3 bucket or an SQS queue. These policies define who can access the resource and what actions they can perform. They are different from identity-based policies, which are attached to IAM users, groups, or roles.

 **What are the IAM identities?**

* IAM identities are entities that can be granted permissions. They include:
  + **IAM Users**: Individuals or applications with permanent credentials.
  + **IAM Roles**: Temporary security credentials for accessing AWS resources.
  + **IAM Groups**: Collections of IAM users.
  + **Federated Identities**: External identities authenticated via SAML or other federated methods.

 **What is a role in IAM? IAM Group?**

* An IAM Role is a set of permissions that can be assumed by users, AWS services, or other roles. Roles are used to delegate access with temporary credentials. An IAM Group is a collection of IAM users with the same permissions, which simplifies management by allowing you to assign policies to the group rather than individual users.

 **Your organization has around 50 IAM users. Now, it wants to introduce a new policy that will affect the access permissions of an IAM user. How can it implement this without having to apply the policy at the individual user level?**

* You can implement this by creating an IAM Group and attaching the new policy to the group. Add the relevant IAM users to the group, and the policy will apply to all users in that group, simplifying management.

 **What is a policy in IAM?**

* An IAM policy is a JSON document that defines permissions for accessing AWS resources. Policies specify what actions are allowed or denied, on which resources, and under what conditions. Policies can be attached to IAM users, groups, or roles.

 **What policy rules do you need to follow after creating scripts on Jenkins or Terraform and deploying them?**

* Follow these best practices:
  + Ensure policies follow the principle of least privilege.
  + Regularly review and update policies to reflect changes in requirements.
  + Use IAM roles for Jenkins or Terraform to limit permissions and avoid using long-term credentials.
  + Monitor and audit the use of permissions and access logs.

 **What is the difference between managed policy and inline policy?**

* **Managed Policy**: A standalone policy created and managed by AWS or by the user, which can be attached to multiple IAM users, groups, or roles. Managed policies can be AWS-managed or customer-managed.
* **Inline Policy**: A policy embedded directly within a single IAM user, group, or role. Inline policies are specific to that entity and are deleted when the entity is deleted.

 **Can you explain the access key in IAM user?**

* An access key consists of an Access Key ID and a Secret Access Key used for programmatic access to AWS services. It is used in combination with the AWS CLI, SDKs, or APIs to authenticate requests. Access keys should be handled securely and rotated periodically.

 **How to give access to an IAM user?**

* Access can be granted to an IAM user by attaching policies to the user directly or by adding the user to IAM groups with the desired policies. Policies define what actions the user can perform on which resources.

 **Have you created any user using IAM?**

* This would be specific to your experience. Generally, you create IAM users via the AWS Management Console, CLI, or API, and assign them necessary permissions.

 **Which policies did you attach to the user?**

* This would be specific to your setup. Typically, you attach policies based on the user’s role and required permissions, such as read/write access to certain AWS resources.

 **How many types of policies are there in IAM?**

* There are two main types of IAM policies:
  + **Managed Policies**: Policies that can be attached to multiple users, groups, or roles. These can be AWS-managed or customer-managed.
  + **Inline Policies**: Policies that are embedded directly into a single user, group, or role.

 **Have you created any customized policy?**

* This would be specific to your experience. Customized policies are created to meet specific requirements and are often used when predefined managed policies do not fit your needs.

 **How do you give access to any user or client to a particular object in your S3?**

* To give access to a specific object in S3, you can:
  + Update the S3 bucket policy or object ACL to grant permissions to the specific user or client.
  + Generate a pre-signed URL for temporary access to the object.

 **What is an IAM Role?**

* An IAM Role is a set of permissions that can be assumed by IAM users, AWS services, or other roles to perform actions on AWS resources. Roles are used to grant temporary access with specific permissions without sharing long-term credentials.

 **What is MFA in AWS?**

* Multi-Factor Authentication (MFA) is a security feature that requires users to provide two or more authentication factors to access AWS resources. This typically involves something the user knows (password) and something the user has (an MFA device or app).

 **What do you know about AWS IAM?**

* AWS IAM (Identity and Access Management) allows you to manage access to AWS resources securely. IAM lets you create and manage AWS users and groups, and use permissions to allow or deny access to AWS resources. It supports the use of roles for temporary access and integrates with MFA for enhanced security.

 **How do you give service-to-service access?**

* Service-to-service access is typically granted using IAM roles. You create a role with the necessary permissions and allow the service to assume that role. For example, you might allow an EC2 instance to assume a role to access S3 or DynamoDB.

 **Your organization has four instances for production and another four for testing. You are asked to set up a group of IAM users that can only access the four production instances and not the other four testing instances. How will you achieve this?**

* Create IAM roles with specific permissions for accessing the production instances. Attach policies to these roles that restrict access to only the production resources. Then, assign these roles to the IAM users or groups who need access to the production instances.

 **What is IAM and what is the purpose of roles?**

* IAM (Identity and Access Management) is a service that allows you to manage access to AWS resources securely. IAM roles are used to grant permissions to AWS services or users to perform specific actions on AWS resources. Roles provide temporary security credentials and are a key component of AWS’s security model.

###########################################################################

**Cloudwatch, CloudFront, Cloudtrail, Cloudformation & monitoring**

###########################################################################

Here are the answers to your CloudTrail, CloudWatch, and CloudFront-related questions:

1. **What do you know about AWS CloudTrail?**
   * AWS CloudTrail is a service that enables governance, compliance, and operational and risk auditing of your AWS account. It logs API calls made in your AWS account, capturing details such as the request parameters, responses, and who made the call. These logs are stored in S3 and can be used for auditing and monitoring.
2. **How to set up Prometheus and Grafana?**
   * **Prometheus**:
     1. Install Prometheus on your server or container.
     2. Configure prometheus.yml to specify which endpoints to scrape for metrics.
     3. Start Prometheus with the configuration file.
   * **Grafana**:
     1. Install Grafana on your server or container.
     2. Add Prometheus as a data source in Grafana.
     3. Create dashboards and visualizations using the metrics collected by Prometheus.
3. **VPC Flow logs vs CloudWatch logs.**
   * **VPC Flow Logs**: Capture information about IP traffic going to and from network interfaces in your VPC. Useful for monitoring and troubleshooting network traffic.
   * **CloudWatch Logs**: Capture and monitor logs from AWS services, applications, and operating systems. Provides more detailed logging capabilities and allows for log analysis and metric creation.
4. **How to create a custom metric in CloudWatch logs?**
   * To create a custom metric:
     1. Use the CloudWatch Logs agent to publish logs to CloudWatch.
     2. In the CloudWatch console, go to Logs, select your log group, and create a metric filter.
     3. Define the filter pattern and specify the metric namespace and name.
5. **Explain Geo Restriction in CloudFront.**
   * Geo Restriction in CloudFront allows you to control who can access your content based on the geographical location of the request. You can either allow or block requests from specific countries or regions.
6. **Why CloudFront?**
   * Amazon CloudFront is a Content Delivery Network (CDN) that delivers content with low latency and high transfer speeds by caching content at edge locations closer to the user. It also provides security features like HTTPS, DDoS protection, and access controls.
7. **I have 50 servers; using which tool would you monitor and health-check the servers?**
   * For monitoring and health-checking servers, you might use tools like Amazon CloudWatch, Prometheus with Grafana, or third-party solutions such as Datadog, Nagios, or New Relic.
8. **Explain what you know about CloudFront CDN.**
   * CloudFront is a global CDN service that caches and delivers content from edge locations close to end users. It reduces latency, improves performance, and scales automatically to handle high traffic volumes. It supports various types of content, including static, dynamic, and streaming data.
9. **What is the difference between Amazon CloudWatch and CloudTrail? In what scenarios is CloudWatch used?**
   * **Amazon CloudWatch**: Monitors and manages operational metrics and logs from AWS resources and applications. Used for creating alarms, dashboards, and custom metrics.
   * **Amazon CloudTrail**: Provides a record of API calls made within your AWS account for auditing and compliance purposes. Used for tracking changes and access patterns.
10. **How can you create a dashboard using Amazon CloudWatch?**
    * To create a dashboard:
      1. Open the CloudWatch console and navigate to Dashboards.
      2. Click “Create Dashboard” and provide a name.
      3. Add widgets (e.g., graphs, numbers) by selecting metrics, logs, or custom data.
      4. Configure the widgets as needed and save the dashboard.
11. **How do you check the health of your application server? What are the parameters?**
    * Check server health by monitoring metrics such as CPU utilization, memory usage, disk I/O, network traffic, and application-specific metrics. Use tools like CloudWatch, Prometheus, or custom health checks to gather and analyze these parameters.
12. **How to recover an EC2 instance using CloudWatch?**
    * CloudWatch itself does not directly recover EC2 instances, but you can create alarms that trigger Auto Scaling actions or AWS Lambda functions to automate instance recovery or replacement based on specific metrics or events.
13. **What would I have to do if I want to access Amazon Simple Storage buckets and use the information for access audits?**
    * Use AWS CloudTrail to capture API activity related to S3 buckets. Enable CloudTrail logging to record all access requests and review the logs for audit purposes.
14. **What do you know about Amazon CloudWatch? Explain its benefits in brief.**
    * Amazon CloudWatch is a monitoring service that provides real-time insights into your AWS resources and applications. Benefits include:
      1. Collection and visualization of metrics and logs.
      2. Setting up alarms and automated actions based on thresholds.
      3. Centralized monitoring of AWS infrastructure and application performance.
15. **What is CloudWatch? What are different actions in an alarm?**
    * CloudWatch is a monitoring service for AWS resources and applications. Alarms in CloudWatch can perform actions such as:
      1. Sending notifications via SNS.
      2. Executing an Auto Scaling policy.
      3. Running an AWS Lambda function.
      4. EC2 actions like stopping, starting, or terminating instances.
16. **Will you monitor cross-region replication?**
    * Yes, you can monitor cross-region replication using CloudWatch metrics and alarms. Set up alarms to track replication status and ensure data consistency across regions.
17. **What different actions are present in an alarm?**
    * Actions in CloudWatch alarms include:
      1. Sending notifications to an SNS topic.
      2. Executing an Auto Scaling policy.
      3. Running a Lambda function.
      4. Performing EC2 instance actions (start, stop, terminate).
18. **If in CloudFront the content is not present at an edge location, what will happen when a request is made for that content?**
    * If the content is not present at an edge location, CloudFront will forward the request to the origin server (e.g., S3 bucket or custom server). Once the content is retrieved, it will be cached at the edge location for future requests.
19. **Can you tell me how an alarm is set in CloudWatch?**
    * To set an alarm in CloudWatch:
      1. Go to the CloudWatch console and navigate to Alarms.
      2. Click “Create Alarm” and select a metric or log.
      3. Define the conditions for the alarm (e.g., threshold, evaluation period).
      4. Configure actions to be taken when the alarm state is triggered.
      5. Name the alarm and create it.
20. **Your organization wants to monitor the read and write IOPS for its AWS MySQL RDS instance and then send real-time alerts to its internal operations team. Which service offered by Amazon can help your organization achieve this scenario?**
    * Amazon CloudWatch can monitor RDS instance metrics including read and write IOPS. You can set up CloudWatch Alarms to send real-time alerts to your operations team based on these metrics.
21. **Which of the following is a global Content Delivery Network service that securely delivers data to users with low latency and high transfer speed?**
    * **Amazon CloudFront** is the global CDN service provided by AWS that delivers content securely with low latency and high transfer speed.
22. **Tell me about CloudWatch metrics. Types of metrics.**
    * CloudWatch metrics are time-ordered data points that provide information about the performance of AWS resources and applications. Types of metrics include:
      1. **Standard Metrics**: Provided by AWS services (e.g., CPU utilization, disk I/O).
      2. **Custom Metrics**: Created by users to monitor application-specific data.
      3. **Log Metrics**: Derived from CloudWatch Logs using metric filters.
23. **What is the purpose of log groups in CloudWatch?**
    * Log groups in CloudWatch organize and manage log streams from various sources. They help aggregate and group logs for easier access, monitoring, and retention management.
24. **From CloudWatch, which services have you monitored?**
    * This would depend on your experience. Commonly monitored services include EC2, RDS, Lambda, S3, and ELB.
25. **What metrics have you created for CloudWatch?**
    * This would depend on your setup. Example metrics might include custom application performance metrics, error rates, or business KPIs.
26. **If any instance stops or crashes, how will you monitor it and manage it? How is SNS triggered for any such situation?**
    * Use CloudWatch alarms to monitor instance health and status. Configure alarms to trigger notifications to an SNS topic, which then sends alerts to your team. SNS can be set up to send notifications via email, SMS, or other endpoints.
27. **Which tool did you use for monitoring?**
    * This would be specific to your experience. Common tools include CloudWatch, Prometheus, Grafana, Datadog, Nagios, and others.
28. **How will you delete an alarm?**
    * To delete a CloudWatch alarm:
      1. Go to the CloudWatch console and navigate to Alarms.
      2. Select the alarm you want to delete.
      3. Click “Actions” and choose “Delete.”
      4. Confirm the deletion.
29. **State the differences between AWS CloudFormation and AWS Elastic Beanstalk.**
    * **AWS CloudFormation**: Provides infrastructure as code to define and deploy AWS resources in a structured way using templates. It offers full control over the AWS resources and their configurations.
    * **AWS Elastic Beanstalk**: A Platform as a Service (PaaS) that simplifies the deployment and management of applications by automatically handling the infrastructure, scaling, and monitoring. It abstracts the underlying infrastructure management.
30. **Explain the working of AWS Config with AWS CloudTrail.**
    * AWS Config monitors and records configurations of AWS resources, while CloudTrail logs API calls. Config uses CloudTrail logs to track changes to resource configurations and ensure compliance with defined rules.
31. **Suppose a request for any particular content is made in CloudFront, but the content is not present in the nearest edge location. What will happen in this scenario?**
    * CloudFront will forward the request to the origin server to retrieve the content. Once the content is retrieved, it will be cached at the edge location for subsequent requests.
32. **Does CloudWatch agent support IAM role?**
    * Yes, the CloudWatch agent can use IAM roles for permissions. When running on an EC2 instance, you can associate an IAM role with the instance, and the CloudWatch agent will use the role's permissions to send metrics and logs.
33. **What does CloudWatch log agent store?**
    * The CloudWatch Logs agent collects and sends log data from your servers to CloudWatch Logs. It stores logs such as application logs, system logs, and custom log data.
34. **How can I get my log data?**
    * Access log data via the CloudWatch console under Logs, or use the CloudWatch Logs API or CLI to retrieve and manage log data.
35. **What are events?**
    * Events in AWS refer to changes or actions that occur within your AWS environment, such as API calls, resource changes, or system state changes. AWS services like CloudWatch and CloudTrail use events for monitoring and triggering actions.
36. **What is AWS CloudTrail? What kind of logs are stored? What is CloudTrail?**
    * AWS CloudTrail records and logs API calls made within your AWS account. Logs include details such as the API request, response, request parameters, and the identity of the caller. CloudTrail helps in auditing and monitoring account activity.
37. **If someone accidentally deleted one server in a customer's environment, how would you know who did that?**
    * Use AWS CloudTrail to track the API calls related to the deletion. CloudTrail logs include the IAM user or role that performed the action, providing details on who deleted the server.
38. **If someone deleted a VPC, how would you check who deleted it?**
    * Similar to other actions, use CloudTrail to review the logs. CloudTrail will show the API calls related to the deletion of the VPC, including the identity of the user or role that initiated the deletion.
39. **Difference between CloudTrail and CloudWatch.**
    * **CloudTrail**: Focuses on logging API calls and changes to AWS resources for auditing and compliance.
    * **CloudWatch**: Monitors operational metrics and logs from AWS resources and applications, providing insights into performance and health.
40. **Four pillars of CloudTrail.**
    * The four pillars of AWS CloudTrail are:
      1. **Visibility**: Provides a record of all API calls and activities.
      2. **Security**: Helps in detecting unauthorized or suspicious activity.
      3. **Compliance**: Assists in meeting regulatory and compliance requirements.
      4. **Troubleshooting**: Facilitates the identification and resolution of operational issues.
41. **What are events?**
    * In AWS, events refer to significant actions or changes that occur in your environment. These can be system-generated notifications or API activity that trigger automated responses or logging.
42. **A client reports that they wanted to see an audit log of any changes made to AWS resources in their account. What can the client do to achieve this?**
    * The client can use AWS CloudTrail to view audit logs of changes made to AWS resources. CloudTrail logs API activity and provides detailed records of actions performed in the account.
43. **What do you mean by CloudFront, and what's its flow?**
    * CloudFront is a Content Delivery Network (CDN) that distributes content globally to improve performance and reduce latency. Its flow involves caching content at edge locations, which then serves requests from the nearest location to the user. If content is not cached, CloudFront retrieves it from the origin server.
44. **What is geotargeting in CloudFront?**
    * Geotargeting in CloudFront allows you to deliver different content to users based on their geographic location. You can configure CloudFront to serve specific content or apply different settings depending on the user's country or region.
45. **What are the different types of servers available in CloudFront?**
    * CloudFront does not have different types of servers but uses edge locations around the world to cache and deliver content. Each edge location acts as a server to serve cached content to users.
46. **What is the use of edge locations?**
    * Edge locations are data centers that CloudFront uses to cache and deliver content closer to end users, reducing latency and improving the speed of content delivery.
47. **What are the different distributions available in CloudFront?**
    * CloudFront distributions include:
      1. **Web Distributions**: For delivering static and dynamic content over HTTP/HTTPS.
      2. **RTMP Distributions**: For streaming media using Adobe Flash Media Server (note that RTMP is being phased out).
48. **Difference between CloudWatch and CloudFront.**
    * **CloudWatch**: A monitoring and management service for AWS resources and applications. Provides metrics, logs, and alarms.
    * **CloudFront**: A Content Delivery Network (CDN) service that caches and delivers content globally to improve performance and reduce latency.
49. **What is invalidation? How do you create an invalidation?**
    * Invalidation in CloudFront is the process of removing cached content from edge locations. To create an invalidation:
      1. Go to the CloudFront console.
      2. Select your distribution and choose “Invalidations.”
      3. Create a new invalidation request and specify the paths to be removed from the cache.
50. **How do you monitor your infrastructure?**
    * Monitor infrastructure using tools and services like AWS CloudWatch, AWS CloudTrail, Prometheus with Grafana, or third-party monitoring solutions. Collect and analyze metrics, logs, and set up alarms to ensure operational health and performance.

#############################

**AWS SNS:**

#############################

* What is a topic in SNS? What is the SNS service used for?
* Are SNS and SQS push-based or pull-based?
* Which protocols are supported by SNS?
* What is SNS? What is the practical use of SNS?
* What is the Simple Notification Service offered by AWS?
* What is Amazon SNS and when is it used? What are topics in SNS?
* Could you provide a practical example of how Amazon SNS is used

########################################

**Amazon RDS, Route 53 & DB**

########################################

Here are the answers to your questions about RDS, Route 53, and other related topics:

1. **What's RDS?**
   * Amazon RDS (Relational Database Service) is a managed database service that provides scalable and resizable relational databases in the cloud. It supports several database engines, including MySQL, PostgreSQL, MariaDB, Oracle, and SQL Server. RDS handles routine database tasks such as provisioning, patching, backup, recovery, and scaling, allowing users to focus on their applications.
2. **Explain some of the advantages of using AWS RDS.**
   * **Managed Service**: AWS handles database management tasks like backups, patching, and failover.
   * **Scalability**: Easy to scale up or down in terms of compute and storage.
   * **High Availability**: Multi-AZ deployments for automatic failover and backup.
   * **Performance**: Options for read replicas to improve read performance.
   * **Security**: Integration with AWS security services and encryption options.
3. **Which one of the following is a structured data store that supports indexing and data queries to both EC2 and S3? DynamoDB / MySQL / Aurora / SimpleDB**
   * **MySQL** and **Aurora** are structured data stores that support indexing and data queries. DynamoDB and SimpleDB are NoSQL databases, and while they provide indexing, they do not support the same level of structured querying as MySQL or Aurora.
4. **What type of instances did you use for RDS? Create table command.**
   * The type of instance depends on your workload requirements. For example, you might use db.m5.large for a balanced approach. A sample CREATE TABLE command for MySQL in RDS could be:

sql

Copy code

CREATE TABLE employees (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(100),

position VARCHAR(100),

hire\_date DATE

);

1. **You have deployed multiple EC2 instances across multiple availability zones to run your website. You have also deployed a Multi-AZ RDS MySQL Extra Large DB Instance. The site performs a high number of small read and write operations per second. After some time, you observed that there is read contention on RDS MySQL. What would be your approach to resolve the contention and optimize your website?**
   * **Read Replicas**: Create read replicas to offload read operations from the primary instance.
   * **Database Optimization**: Optimize queries and database schema to reduce contention.
   * **Indexing**: Ensure proper indexing to speed up query performance.
   * **Caching**: Implement caching strategies using tools like Amazon ElastiCache to reduce database load.
2. **Draw the structure of your web application. How do you deploy and build the project? How do you use middleware, RDS, and services?**
   * **Structure**: Typically includes a front-end (client-side), back-end (server-side), and database layer.
     + **Front-End**: HTML, CSS, JavaScript.
     + **Back-End**: Application server (e.g., EC2 instances) running application code.
     + **Database Layer**: AWS RDS for storing application data.
   * **Deployment and Build**: Use CI/CD pipelines with tools like Jenkins or AWS CodePipeline to automate deployment.
   * **Middleware**: Handles requests between the front-end and back-end, often including authentication and data processing.
   * **RDS**: Stores application data; accessed by the back-end for data operations.
3. **What is AWS Route 53, and why do we use it?**
   * AWS Route 53 is a scalable DNS web service designed to route end users to Internet applications by translating domain names into IP addresses. It offers high availability, reliability, and DNS management capabilities. It is used for domain registration, DNS routing, and health checking.
4. **What is the purpose of RDS in your project?**
   * RDS provides a managed relational database solution, ensuring ease of use, scalability, and availability for handling structured data. It supports high performance and security for your project's data needs.
5. **Could you explain the concepts of Amazon RDS, NoSQL, and MySQL?**
   * **Amazon RDS**: A managed service for relational databases.
   * **NoSQL**: A database design for non-relational, scalable, and distributed data stores, e.g., DynamoDB.
   * **MySQL**: A widely-used open-source relational database management system that RDS can manage.
6. **Is NoSQL a part of Amazon RDS?**
   * No, NoSQL databases are not part of Amazon RDS. Amazon RDS supports relational databases. For NoSQL databases, AWS offers services like DynamoDB and Amazon DocumentDB.
7. **What port is commonly used for MySQL?**
   * The default port for MySQL is **3306**.
8. **How do CloudTrail and Route 53 work together?**
   * CloudTrail logs API calls made to Route 53, providing visibility into DNS changes and management actions. This helps in auditing and tracking changes to DNS configurations.
9. **An application running on AWS uses an Amazon Aurora Multi-AZ deployment for its database. When evaluating performance metrics, a solutions architect discovered that the database reads are causing high I/O and adding latency to the write requests against the database. What should the solutions architect do to separate the read requests from the write requests?**
   * Implement **read replicas** for Amazon Aurora. This will separate read traffic from write traffic, reducing the I/O load on the primary instance and improving overall performance.
10. **Routing policy in Route 53.**
    * Route 53 supports several routing policies, including:
      + **Simple Routing**: Routes traffic to a single resource.
      + **Weighted Routing**: Distributes traffic based on weights assigned to resources.
      + **Latency Routing**: Routes traffic to the resource with the lowest latency.
      + **Failover Routing**: Routes traffic to a primary resource and a secondary resource in case of failure.
      + **Geolocation Routing**: Routes traffic based on the geographic location of the user.
11. **Which of the following Amazon Services would you choose if you want complex querying capabilities but not a whole data warehouse? RDS / Redshift / ElastiCache / DynamoDB**
    * **RDS** is the best choice for complex querying capabilities. Redshift is a data warehouse, ElastiCache is for caching, and DynamoDB is a NoSQL database.
12. **How do you save your data in a DB in a three-tier architecture?**
    * In a three-tier architecture, data is saved in the database layer (typically a relational database) via the application server. The data is stored in tables and managed by the database management system (DBMS) in the database tier.
13. **How do you ensure that the data stored in the DB is secure?**
    * **Encryption**: Use encryption at rest and in transit.
    * **Access Control**: Implement IAM policies and database access controls.
    * **Backups**: Regularly back up data and use automated backup features.
    * **Monitoring**: Enable monitoring and logging for suspicious activities.
    * **Patching**: Apply security patches and updates promptly.
14. **An organization has multiple facilities in various continents such as North America, Europe, and the Asia Pacific. The organization is designing a new distributed application to manage and optimize its global supply chain and its manufacturing process. It needs to design the process in such a way that the booked order in one continent should be able to support data failover with a short Recovery Time Objective (RTO). The uptime of the application should not impact manufacturing, what kind of solution would you recommend as a solution architect?**
    * Implement **cross-region replication** and **multi-region deployments**. Use services like Amazon RDS with Multi-AZ and cross-region replication or Amazon Aurora Global Databases to ensure data availability and failover with minimal RTO.
15. **Which data are you going to save in the database? Explain with a real-time example.**
    * Data such as user information, transaction records, and application configurations. For example, in an e-commerce application, you would save customer details, order history, and product information in the database.
16. **If you are running your DB instance as Multi-AZ deployment, can you use standby DB instances along with your primary DB instance?**
    * Yes, Multi-AZ deployments provide a standby DB instance that automatically takes over in case of a failure of the primary DB instance, but the standby is not used for read operations or queries.
17. **If you launched a standby RDS, will it be launched in the same availability zone as your primary?**
    * No, in a Multi-AZ deployment, the standby RDS instance is launched in a different availability zone from the primary to ensure high availability and fault tolerance.
18. **Name some of the DB engines which can be used in AWS RDS.**
    * **MySQL**, **PostgreSQL**, **MariaDB**, **Oracle**, **SQL Server**, and **Amazon Aurora**.
19. **Your organization is developing a new multi-tier web application in AWS. Being a fairly new and small organization, there’s limited staff. But, the organization requires high availability. This new application comprises complex queries and table joins. Which Amazon service will be the best solution for your organization’s requirements?**
    * **Amazon RDS** with a relational database engine like MySQL or PostgreSQL would be suitable for handling complex queries and table joins, while providing high availability through Multi-AZ deployments.
20. **Your firm has its offices in various parts of the world and is involved in multi-regional deployment on AWS. For data persistence, your firm uses MYSQL 5.6. Your firm has recently announced that it needs to regularly collect batch process data from each region and generate regional reports. The reports will then be forwarded to various branch offices. What course of action will you suggest to perform this task in the shortest possible time?**
    * Use **Amazon RDS with read replicas** for each region to collect and process data. For batch processing and report generation, use **AWS Data Pipeline** or **AWS Glue** to automate data collection and report generation.
21. **Your firm’s application is responsible for retrieving data from your subscriber’s/user’s mobile devices every 10 minutes. The retrieved data is stored in DynamoDB. The information is extracted into S3 for each user. Once the data is extracted, the application helps in data visualization on the user end. As a senior architect in your firm, you are asked to optimize the backend architecture so that the firm can slash the costs. What are your recommendations?**
    * **Optimize DynamoDB usage** by reviewing and adjusting read and write capacity modes or using on-demand capacity if traffic is variable.
    * **Use S3 storage classes** to manage data cost-effectively.
    * **Implement caching** for frequently accessed data using services like Amazon ElastiCache.
    * **Optimize data extraction** and visualization processes to minimize the number of data transfers and compute resources required.

######################################################

**AWS Load balancers, EIP & AutoScaling**

######################################################

 **What are the different types of Load Balancers in AWS services?**

* **Application Load Balancer (ALB)**: Operates at the application layer (Layer 7) and routes HTTP/HTTPS traffic based on content.
* **Network Load Balancer (NLB)**: Operates at the transport layer (Layer 4) and routes TCP/UDP traffic, handling millions of requests per second with ultra-low latency.
* **Classic Load Balancer (CLB)**: Operates at both Layer 4 and Layer 7, and is the oldest type, now mostly replaced by ALB and NLB for new deployments.

 **Explain the auto-scaling feature of EC2 along with its benefits.**

* **Auto-Scaling** automatically adjusts the number of EC2 instances in your application based on demand.
* **Benefits**:
  + **Cost Efficiency**: Reduces costs by scaling down during low demand.
  + **High Availability**: Ensures application availability by adding instances during high demand.
  + **Fault Tolerance**: Automatically replaces unhealthy instances.

 **What are Load Balancers?**

* Load balancers distribute incoming network or application traffic across multiple servers to ensure no single server becomes overwhelmed. This improves the reliability and performance of applications.

 **Why do we need autoscaling?**

* **Autoscaling** ensures that your application has the right amount of resources at any given time. It helps in:
  + Handling varying traffic loads.
  + Reducing costs by scaling down resources when not needed.
  + Maintaining high availability and performance.

 **Why do we need Autoscaling Groups?**

* Autoscaling Groups manage the scaling of EC2 instances based on policies, health checks, and schedules. They ensure:
  + Consistent scaling across multiple instances.
  + Automatic replacement of failed instances.
  + Control over the minimum and maximum number of instances.

 **Which of the following services can be used if you want to capture client connection information from your load balancer at a particular time interval?**

* **AWS CloudWatch Logs** can capture and store logs from the load balancer, including client connection information.

 **Out of the following options, where does the user specify the maximum number of instances with the autoscaling commands?**

* **Autoscaling Group** is where you specify the maximum number of instances.

 **I created a web application with autoscaling. I observed that the traffic on my application is the highest on Wednesdays and Fridays between 9 AM and 7 PM. What would be the best solution for me to handle the scaling?**

* Configure **scheduled scaling policies** to increase the number of instances during peak times on Wednesdays and Fridays and reduce them during off-peak hours.

 **Load balancer types? Differences? Where to use?**

* **Application Load Balancer (ALB)**: Best for HTTP/HTTPS traffic, content-based routing, and microservices architectures.
* **Network Load Balancer (NLB)**: Best for TCP/UDP traffic, high performance, and handling millions of requests per second.
* **Classic Load Balancer (CLB)**: Suitable for simple Layer 4 and Layer 7 balancing but is being phased out in favor of ALB and NLB.

 **What is an elastic load balancer?**

* **Elastic Load Balancer (ELB)** is a managed service that automatically distributes incoming traffic across multiple targets, such as EC2 instances, in one or more Availability Zones.

 **Why do you use Elastic load balancer? Which load balancer have you used in your project?**

* **Elastic Load Balancer** is used to ensure high availability, improve fault tolerance, and distribute traffic efficiently. In projects, I have used **Application Load Balancer** for HTTP/HTTPS traffic and **Network Load Balancer** for TCP/UDP traffic.

 **A company's web application is using multiple Linux Amazon EC2 instances and storing data on Amazon EBS volumes. The company is looking for a solution to increase the resiliency of the application in case of a failure and to provide storage that complies with atomicity, consistency, isolation, and durability (ACID). What should a solutions architect do to meet these requirements?**

* Use **Amazon RDS** for database services to meet ACID requirements. For increased resiliency, ensure that EC2 instances are behind an **Elastic Load Balancer** and use **Amazon EBS** with replication.

 **You have an application running on an EC2 instance. You need to reduce the load on your instance as soon as the CPU utilization reaches 80 percent. How will you accomplish the job?**

* Create an **Auto Scaling Policy** that triggers when CPU utilization exceeds 80 percent to add more instances to handle the load.

 **A start-up company has a web application based in the us-east-1 Region with multiple Amazon EC2 instances running behind an Application Load Balancer across multiple Availability Zones. As the company's user base grows in the us-west-1 region, the company needs a solution with low latency and improved high availability. What should a solutions architect do to achieve it?**

* Implement **Amazon Route 53** for geo-routing to direct users in the us-west-1 region to a load balancer in that region. Consider setting up an **Application Load Balancer** in us-west-1 and configure **cross-region replication** if needed.

 **Can you change the instance type of the instances that are running in your application tier and are also using autoscaling? If yes, then how?**

* Yes, update the **Launch Configuration** or **Launch Template** used by the Auto Scaling Group with the new instance type and then apply the changes. Instances will be replaced with new ones using the updated configuration.

 **Explain how you can vertically scale an Amazon instance?**

* To vertically scale an EC2 instance, stop the instance, change the instance type to a more powerful type, and then start the instance again. This increases the instance's CPU, memory, and storage capabilities.

 **Suppose, you hosted an application on AWS that lets the users render images and do some general computing. Which of the below listed services can you use to route the incoming user traffic?**

* **Amazon Route 53** can be used to route incoming user traffic to your application.

 **You have an application running on your Amazon EC2 instance. You want to reduce the load on your instance as soon as the CPU utilization reaches 100 percent. How will you do that?**

* Set up an **Auto Scaling Policy** to add more instances when CPU utilization reaches 100 percent, thereby reducing the load on the current instance.

 **You have connected four instances to ELB. To automatically terminate your unhealthy instances and replace them with new ones, which functionality would you use?**

* Use **Auto Scaling Groups** with health checks to automatically terminate and replace unhealthy instances.

 **Can you vertically scale an Amazon Instance?**

* Yes, vertically scale by stopping the instance, changing its type, and restarting it to increase its resources.

 **When should you use the classic load balancer and the application load balancer?**

* **Classic Load Balancer (CLB)**: Suitable for basic load balancing of HTTP/HTTPS and TCP traffic but is mostly phased out.
* **Application Load Balancer (ALB)**: Best for HTTP/HTTPS traffic with advanced features like content-based routing and host-based routing.

 **What is the use of lifecycle hooks in Autoscaling?**

* Lifecycle hooks allow you to perform custom actions when instances are launching or terminating, such as running configuration scripts or performing pre-termination cleanup.

 **How can we use Network Load Balancer and Application Load Balancer together in the same project?**

* Use **Network Load Balancer (NLB)** for TCP/UDP traffic that requires high performance and **Application Load Balancer (ALB)** for HTTP/HTTPS traffic with content-based routing. NLB can forward traffic to ALB if needed.

 **What is an elastic IP?**

* An **Elastic IP (EIP)** is a static, public IP address designed for dynamic cloud computing. It allows you to easily reassign an IP address to a different EC2 instance.

 **EIP default launch limit.**

* The default limit for Elastic IPs per AWS account is typically 5. This limit can be increased by requesting a limit increase through the AWS Support Center.

 **You accidentally stopped an EC2 instance in a VPC with an associated Elastic IP. If you start the instance again, what will be the result?**

* The Elastic IP remains associated with the instance if you start it again, provided the instance is in the same VPC.

 **How many elastic IPs can we create in one VPC?**

* The limit for Elastic IPs in a VPC is typically 5 per region per AWS account. This limit can be increased by making a request to AWS.

 **Can we use multiple load balancers on a single EC2 machine?**

* Yes, multiple load balancers can be configured to route traffic to a single EC2 instance, each handling different types of traffic or using different protocols.

 **A customer wants to stream 50 servers. What type of load balancer is used?**

* **Network Load Balancer (NLB)** is suitable for handling high-throughput and low-latency traffic, making it ideal for streaming scenarios.

 **What is connection draining?**

* **Connection draining** (or deregistration delay) allows existing connections to complete before the load balancer stops routing traffic to an instance that is being deregistered or terminated.

 **What are the different types of load balancers?**

* **Application Load Balancer (ALB)**
* **Network Load Balancer (NLB)**
* **Classic Load Balancer (CLB)**

 **Explain horizontal and vertical scaling in AWS?**

* **Horizontal Scaling**: Adding more instances to handle increased load (scaling out/in).
* **Vertical Scaling**: Increasing the resources (CPU, memory) of a single instance (scaling up/down).

 **You have 50 database servers. Which load balancer would you use?**

* **Network Load Balancer (NLB)** is suitable for distributing database traffic, especially if it involves high throughput and low latency requirements.

 **What do you know about load balancer? Types of load balancer? How do you create them?**

* Load balancers distribute traffic to multiple targets to ensure high availability and reliability. Types include **Application Load Balancer**, **Network Load Balancer**, and **Classic Load Balancer**. You create them via the AWS Management Console, AWS CLI, or AWS SDKs by specifying target groups, listeners, and routing rules.

 **Can you explain the working of the network load balancer and its practical use?**

* **Network Load Balancer (NLB)** operates at Layer 4 and handles TCP and UDP traffic. It is designed for high performance, handling millions of requests per second with low latency. It is used for scenarios requiring high throughput and reliability, such as gaming or real-time data processing.

 **Can you assign a static IP to a load balancer?**

* Yes, **Network Load Balancer (NLB)** supports static IP addresses.

 **Difference between ALB and NLB.**

* **ALB (Application Load Balancer)**: Operates at Layer 7 (Application Layer), supports HTTP/HTTPS, provides advanced routing capabilities.
* **NLB (Network Load Balancer)**: Operates at Layer 4 (Transport Layer), supports TCP/UDP, handles high-throughput, low-latency traffic.

 **Have you worked on provisioning on AWS like DB Load balancer and servers?**

* Yes, I have experience with provisioning load balancers (ALB, NLB) and servers (EC2 instances) to manage traffic and scale applications effectively.

 **Why does ELB hold many IP addresses?**

* **Elastic Load Balancer (ELB)** may use multiple IP addresses to distribute traffic efficiently across multiple targets and to provide high availability and fault tolerance.

 **Why is the classic load balancer not used many times?**

* **Classic Load Balancer (CLB)** is often replaced by ALB and NLB due to their advanced features like content-based routing, improved performance, and support for modern protocols.

 **What is a listener? Listener component in LB.**

* A **listener** is a process that checks for connection requests. It is configured with a protocol and port (e.g., HTTP on port 80) and forwards requests to the target groups based on the rules defined.

 **HTTP vs HTTPS.**

* **HTTP**: Hypertext Transfer Protocol, unencrypted data transfer.
* **HTTPS**: HTTP Secure, encrypted data transfer using TLS/SSL.

 **In LB, is it possible to change the default port number? What is the condition?**

* Yes, you can change the default port number when configuring a listener for the load balancer. Ensure that your target group and backend instances are set up to handle the new port.

 **404 error?**

* A **404 error** indicates that the requested resource could not be found on the server. It is typically a client-side error indicating that the URL is incorrect or the resource does not exist.

 **In LB, which one is faster and why?**

* **Network Load Balancer (NLB)** is generally faster than ALB because it operates at Layer 4 and handles TCP/UDP traffic with lower latency, while ALB performs content-based routing at Layer 7, which adds processing overhead.

 **How will load decide where to go?**

* The load balancer uses defined rules, algorithms (such as round-robin, least connections), and routing decisions based on the type of load balancer (ALB, NLB) to distribute traffic to the targets.

 **What is the difference between load balancer and auto scaling?**

* **Load Balancer** distributes incoming traffic across multiple instances to ensure high availability and reliability.
* **Auto Scaling** adjusts the number of instances in response to changes in traffic or load to maintain performance and cost efficiency.

 **How to add an existing instance in an autoscaling group?**

* You cannot directly add an existing instance to an Auto Scaling Group. Instead, you can:
  + Create a new launch configuration or template with the desired instance type.
  + Update the Auto Scaling Group to use this configuration/template.
  + Terminate existing instances, and the Auto Scaling Group will launch new instances using the updated configuration.

 **What is a launch group in autoscaling?**

* **Launch Configuration** or **Launch Template** defines the settings for new instances launched by the Auto Scaling Group, including instance type, AMI, and other parameters.

 **Can you explain what is auto scaling?**

* **Auto Scaling** is a service that automatically adjusts the number of EC2 instances in your application based on demand, ensuring optimal performance and cost efficiency.

 **What are the situations to move the traffic or how many ways to distribute the traffic from one EC2 to another, what weight you provide?**

* Traffic can be distributed using:
  + **Load Balancers** (e.g., ALB, NLB) with different routing algorithms.
  + **DNS-based routing** with Route 53.
  + **Application-level adjustments** (e.g., using a reverse proxy).
  + Weights can be assigned in routing rules or configurations to control traffic distribution.

 **If a thousand requests come to your web page and the load increases, what do you do and how can you decrease the load?**

* **Scale Out**: Add more instances using Auto Scaling.
* **Optimize**: Implement caching, use a Content Delivery Network (CDN) like CloudFront, or optimize the application’s performance.
* **Load Balancing**: Distribute traffic across multiple instances with a load balancer.

 **Benefits of auto scaling?**

* **Cost Savings**: Only pay for the resources you need.
* **Improved Availability**: Automatically replaces failed instances and scales during high demand.
* **Performance**: Maintains application performance by adjusting resources as needed.

 **Policies in auto scaling.**

* **Scaling Policies**: Define how and when to add or remove instances based on metrics (e.g., CPU utilization).
* **Scheduled Policies**: Scale resources based on a schedule (e.g., increase capacity at specific times).
* **Dynamic Scaling Policies**: Adjust capacity based on demand in real time.

 **Scale up and scale down?**

* **Scale Up**: Increase the instance size (vertical scaling) or number of instances (horizontal scaling).
* **Scale Down**: Decrease the instance size or number of instances.

 **How much time is required to pass a health check?**

* Health check timing is configurable. Typically, a health check might take 30 seconds to 1 minute, depending on the health check interval and timeout settings.

 **Min and max. desired value? Explain.**

* **Min**: Minimum number of instances in the Auto Scaling Group.
* **Max**: Maximum number of instances in the Auto Scaling Group.
* **Desired**: The target number of instances that the Auto Scaling Group should maintain based on the scaling policies.

 **What is horizontal and vertical scaling?**

* **Horizontal Scaling**: Adding more instances to handle increased load (scaling out/in).
* **Vertical Scaling**: Increasing the capacity of existing instances (scaling up/down).

 **Does rebalancing work in auto scaling?**

* Yes, **rebalancing** works in Auto Scaling by redistributing instances across Availability Zones to maintain a balanced distribution based on the current configuration.

 **Warm-up time?**

* **Warm-up Time** is the period during which newly launched instances are considered “warm” and should not be terminated or replaced. It allows instances to fully initialize and handle traffic before being evaluated by health checks.

 **What information do you collect from your client regarding auto scaling?**

* **Application Requirements**: Traffic patterns, performance metrics.
* **Scaling Policies**: Desired thresholds for scaling up or down.
* **Capacity Needs**: Minimum and maximum instance counts.
* **Health Check Configuration**: Criteria for instance health.

 **In a target group, you find that an instance is unhealthy. How do you troubleshoot that?**

* **Check Health Checks**: Verify health check configurations and thresholds.
* **Examine Logs**: Look at application and instance logs for errors.
* **Review Instance Metrics**: Check CPU, memory, and network usage.
* **Verify Network Configuration**: Ensure that security groups and network settings are correct.

 **How would you set up a load balancer for an application?**

* **Create a Load Balancer**: Choose the type (ALB, NLB, CLB).
* **Configure Listeners**: Define protocols and ports.
* **Set Up Target Groups**: Specify the instances or IPs to receive traffic.
* **Define Routing Rules**: Determine how traffic is distributed to target groups.
* **Configure Health Checks**: Set up criteria for instance health.

 **What is an Elastic IP?**

* An **Elastic IP (EIP)** is a static, public IP address designed for dynamic cloud computing that can be associated with or reassigned to EC2 instances as needed.

 **Explain various types of elastic load balancers in AWS.**

* **Application Load Balancer (ALB)**: Best for HTTP/HTTPS traffic with advanced routing.
* **Network Load Balancer (NLB)**: Best for TCP/UDP traffic with high performance.
* **Classic Load Balancer (CLB)**: Older type, suitable for basic load balancing tasks.

############################################

**VPC, Security Group & Networking:**

############################################

 **Why do we need to create a VPC?**

* A Virtual Private Cloud (VPC) allows you to define a virtualized network environment in AWS. It provides control over your networking environment, including IP address ranges, subnets, route tables, and network gateways. It helps to isolate and secure resources, manage network traffic, and control connectivity.

 **Why IGW?**

* An Internet Gateway (IGW) allows communication between instances in your VPC and the internet. It provides a target in your VPC route tables for internet-routable traffic and performs network address translation (NAT) for instances with public IP addresses.

 **Mention the differences between security groups and a network access control list.**

* **Security Groups:**
  + Stateful: Return traffic is automatically allowed if the initial request is allowed.
  + Operate at the instance level.
  + Can only allow inbound traffic; outbound traffic is allowed by default unless explicitly restricted.
* **Network Access Control Lists (NACLs):**
  + Stateless: Return traffic must be explicitly allowed.
  + Operate at the subnet level.
  + Can both allow and deny inbound and outbound traffic based on rules.

 **Why do we make subnets?**

* Subnets divide a VPC’s IP address range into smaller segments. This enables better organization, isolation, and control of resources. Subnets can be designated as public (accessible from the internet) or private (not directly accessible from the internet).

 **Can you name the additional network interface that can be created and attached to your Amazon EC2 instance launched in your VPC?**

* Elastic Network Interface (ENI) is the additional network interface that can be created and attached to an EC2 instance.

 **NGW in AWS.**

* A NAT Gateway (NGW) enables instances in a private subnet to initiate outbound traffic to the internet or other AWS services but prevents unsolicited inbound traffic from reaching those instances.

 **You want to modify the security group rules while it is being used by multiple EC2 instances. Will you be able to do that? If yes, will the new rules be implemented on all previously running EC2 instances that were using that security group?**

* Yes, you can modify security group rules at any time. The new rules are automatically applied to all instances associated with that security group.

 **Difference between IGW and NGW.**

* **IGW (Internet Gateway):**
  + Provides internet access to instances with public IP addresses.
  + Used for inbound and outbound internet traffic.
* **NGW (NAT Gateway):**
  + Allows instances in private subnets to access the internet for outbound traffic while blocking unsolicited inbound traffic.
  + Used primarily for instances without public IP addresses.

 **What is VPC?**

* A Virtual Private Cloud (VPC) is a virtual network dedicated to your AWS account. It provides a logically isolated section of the AWS cloud where you can launch AWS resources in a virtual network that you define.

 **How to create VPC?**

* You can create a VPC using the AWS Management Console, AWS CLI, or an AWS SDK. In the console, navigate to the VPC dashboard, click "Create VPC," and follow the prompts to configure CIDR blocks, subnets, and other settings.

 **How many VPCs do you have in your project?**

* This would depend on your specific AWS environment. There’s a limit of 5 VPCs per region by default, which can be increased by requesting a limit raise from AWS.

 **What is VPC peering?**

* VPC Peering allows you to connect two VPCs to enable routing of traffic between them using private IP addresses. This connection can be within the same region or across different regions.

 **How to connect virtual machines between private subnets?**

* To connect virtual machines between private subnets, ensure they are in the same VPC or use VPC Peering if they are in different VPCs. Ensure that route tables and security group rules allow traffic between the subnets.

 **What kind of subnets do you have in the company?**

* This depends on the specific design of your company’s network. Generally, you have public subnets (for internet-facing resources) and private subnets (for resources that do not need direct internet access).

 **Explain Project Infrastructure.**

* This would involve describing the specific setup of your project’s cloud infrastructure, including VPC configuration, subnet arrangements, security groups, NAT gateways, and other networking and compute resources.

 **How many servers do you have in various environment subnets?**

* This is specific to your project and would require reviewing your infrastructure setup.

 **Explain different methods to access the server in your private subnet.**

* Access can be achieved via:
  + Bastion hosts (jump boxes) in a public subnet.
  + VPN connections.
  + AWS Systems Manager Session Manager.
  + Direct Connect (if configured).

 **VPC - CIDR block to allocate 250 IP addresses to 250 EC2 instances.**

* To allocate 250 IP addresses, a CIDR block of /23 (which provides 510 IP addresses) would be appropriate. A /24 block only provides 256 IP addresses, including reserved addresses.

 **What are the security best practices for Amazon EC2?**

* Use security groups and NACLs for traffic control.
* Regularly update and patch instances.
* Use IAM roles with least privilege.
* Enable logging and monitoring.
* Encrypt sensitive data.
* Backup important data.

 **Difference between NACL and Security Group.**

* Already addressed in question 3.

 **Which of the following is not an option in security groups? List of users / Posts / IP addresses / List of protocols**

* List of users is not an option in security groups.

 **How many total VPCs per account/region and subnets per VPC can you have?**

* By default, you can have up to 5 VPCs per region per account, but this limit can be increased. Each VPC can have up to 200 subnets by default.

 **Why do we need NACL?**

* Network Access Control Lists (NACLs) provide an additional layer of security by controlling inbound and outbound traffic at the subnet level. They can be used to apply more granular network access control compared to security groups.

 **As you have worked in the banking domain, where security is a primary concern, what security measures and options have you followed?**

* Implemented stringent security policies, used encryption for data in transit and at rest, applied least privilege access controls, regularly monitored and audited access logs, and ensured compliance with regulatory standards.

 **A company is using a VPC peering connection option to connect its multiple VPCs in a single region to allow for cross-VPC communication. A recent increase in account creations and VPCs has made it difficult to maintain the VPC peering strategy, and the company expects to grow to hundreds of VPCs. There are also new requests to create site-to-site VPNs with some of the VPCs. A solutions architect has been tasked with creating a centrally networking setup for multiple accounts and VPNs. Which networking solution would you recommend to resolve it?**

* AWS Transit Gateway is recommended. It provides a scalable and central hub for connecting multiple VPCs and on-premises networks, simplifying management and routing.

 **Which protocol layer does NACL reside on?**

* NACLs operate at the network layer (Layer 3) of the OSI model.

 **You are an AWS Architect in your company, and you are asked to create a new VPC in the N.Virginia Region with two Public and two Private subnets using the following CIDR blocks: VPC CIDR = 10.10.10.0/24, Public Subnet - Subnet01 : 10.10.10.0/26 Subnet02 : 10.10.10.64/26 Private Subnet Subnet03: 10.10.10.128/26 , Subnet04: 10.10.10.192/26. Using the above CIDRs you created a new VPC, and you launched EC2 instances in all subnets as per the need. Now, you are facing an issue in private instances that you are unable to update operating systems from the internet. So, what architectural changes and configurations will you suggest to resolve the issue?**

* Create a NAT Gateway in one of the public subnets and update the route tables for the private subnets to route outbound traffic through the NAT Gateway. This allows private instances to access the internet for updates while keeping them protected from inbound traffic.

 **You have created a VPC with private and public subnets. In what kind of subnet would you launch the database servers?**

* Database servers should be launched in private subnets to prevent direct access from the internet, enhancing security.

 **What do you know about NAT gateways in AWS?**

* A NAT Gateway enables instances in a private subnet to access the internet for outbound traffic (e.g., for updates) while preventing unsolicited inbound traffic from reaching those instances. It is managed by AWS and scales automatically.

 **In a VPC with private and public subnets, where should database servers ideally be launched?**

* Database servers should ideally be launched in private subnets to ensure they are not directly accessible from the internet.

 **What do you do if any suspicious thing happens in your VPC?**

* Investigate the incident using AWS CloudTrail logs, VPC Flow Logs, and other monitoring tools. Review security groups and NACL configurations, and take appropriate remediation actions based on the findings. Implement additional security measures as needed.

 **Is the property of broadcast or multicast supported by Amazon VPC?**

* Amazon VPC does not support broadcast traffic. Multicast is also not natively supported, but AWS provides options like Transit Gateway for some similar functionalities.

 **VPC private linked points.**

* VPC PrivateLink allows private connectivity to services across VPCs within the AWS network, bypassing the need for public IPs. This improves security and reduces the risk of exposure to the public internet.

 **What is the significance of VPC Peering? Advantages? Why do we need it?**

* VPC Peering allows you to route traffic between VPCs using private IP addresses. It is useful for connecting resources in different VPCs securely, without using the public internet. Advantages include simplified network architecture and improved security.

 **Diff between Network layer and transport layer.**

* **Network Layer (Layer 3):** Handles routing of packets between devices across different networks. Examples include IP addresses and routing protocols.
* **Transport Layer (Layer 4):** Manages end-to-end communication and ensures data is transferred reliably between devices. Examples include TCP and UDP.

 **Diff between Apache and Nginx.**

* **Apache:**
  + Process-based architecture.
  + Suitable for dynamic content.
  + Highly configurable with a large number of modules.
* **Nginx:**
  + Event-driven, asynchronous architecture.
  + High performance for static content and as a reverse proxy.
  + Efficient resource usage and scalability.

 **Why do we need to create a VPC?**

* Already addressed in question 1.

 **Why IGW? NGW in AWS? Difference between IGW and NGW.**

* Already addressed in questions 2, 6, and 8.

 **VPC - CIDR block to allocate 250 IP addresses to 250 EC2 instances.**

* Already addressed in question 18.

 **What you understand by a security group? Explain types? Its permissions?**

* A Security Group is a virtual firewall for EC2 instances that controls inbound and outbound traffic. Permissions are defined by rules that specify allowed IP addresses and protocols. Security Groups are stateful.

 **Why is a security group required?**

* Security Groups are required to control access to EC2 instances, providing a means to define which traffic is allowed to and from your instances.

 **What are the security best practices for Amazon EC2?**

* Already addressed in question 19.

 **How will you rescue something in the cloud?**

* Implement disaster recovery strategies, maintain regular backups, use automated recovery tools, and have a well-defined incident response plan.

 **What is the default outgoing traffic to the security group?**

* By default, all outbound traffic is allowed from a security group, unless restricted by outbound rules.

 **What's the difference between NACL and security group?**

* Already addressed in questions 3 and 20.

 **What are the key differences between public & private subnets?**

* **Public Subnets:**
  + Directly accessible from the internet via an IGW.
  + Typically used for resources that need internet access.
* **Private Subnets:**
  + Not directly accessible from the internet.
  + Used for internal resources that do not need direct internet access.

 **What are the key differences between IGW & NAT?**

* Already addressed in question 8.

 **What are the levels of security in AWS?**

* **Physical Security:** Data centers and hardware protection.
* **Network Security:** VPC, security groups, NACLs, and DDoS protection.
* **Application Security:** IAM roles, encryption, and security best practices.
* **Data Security:** Encryption at rest and in transit, backup, and disaster recovery.

 **What is a VPC endpoint?**

* A VPC endpoint allows private connections between your VPC and supported AWS services without requiring an internet gateway, NAT device, VPN connection, or AWS Direct Connect connection.

 **What do you do if any suspicious thing happens in your VPC?**

* Already addressed in question 31.

 **Is it necessary to use NACL in a private subnet?**

* Not strictly necessary; private subnets can rely on security groups. NACLs provide an additional layer of security and control at the subnet level.

 **Do you need an internet gateway to use peering connections?**

* No, an internet gateway is not required for VPC Peering connections. Peering connections facilitate private communication between VPCs.

 **Explain in detail about AWS VPC.**

* Already addressed in question 9.

 **Should one launch EC2 instances in a VPC? If yes, why?**

* Yes, EC2 instances should be launched in a VPC to leverage network isolation, security, and control. A VPC provides a secure and isolated network environment for your instances.

 **Can you launch Amazon Elastic Compute Cloud (EC2) instances with predetermined private IP addresses? If yes, then with which Amazon service it is possible?**

* Yes, you can launch EC2 instances with predetermined private IP addresses using the AWS Management Console, CLI, or API by specifying the IP address when launching the instance.

 **Suppose, I created a subnet and launched an EC2 instance in the subnet with default settings. Which of the following options will be ready to use on the EC2 instance as soon as it is launched?**

* By default, the EC2 instance will have a private IP address and access to other resources in the VPC based on the subnet’s route table and security group settings. Public IP assignment requires additional configurations.

#############################

**Kubernetes (k8s) & ECS**

#############################

1. **What are the different types of Load Balancers in AWS services?**
   * **Application Load Balancer (ALB)**: Operates at the application layer (Layer 7) and routes HTTP/HTTPS traffic based on content.
   * **Network Load Balancer (NLB)**: Operates at the transport layer (Layer 4) and routes TCP/UDP traffic, handling millions of requests per second with ultra-low latency.
   * **Classic Load Balancer (CLB)**: Operates at both Layer 4 and Layer 7, and is the oldest type, now mostly replaced by ALB and NLB for new deployments.
2. **Explain the auto-scaling feature of EC2 along with its benefits.**
   * **Auto-Scaling** automatically adjusts the number of EC2 instances in your application based on demand.
   * **Benefits**:
     + **Cost Efficiency**: Reduces costs by scaling down during low demand.
     + **High Availability**: Ensures application availability by adding instances during high demand.
     + **Fault Tolerance**: Automatically replaces unhealthy instances.
3. **What are Load Balancers?**
   * Load balancers distribute incoming network or application traffic across multiple servers to ensure no single server becomes overwhelmed. This improves the reliability and performance of applications.
4. **Why do we need autoscaling?**
   * **Autoscaling** ensures that your application has the right amount of resources at any given time. It helps in:
     + Handling varying traffic loads.
     + Reducing costs by scaling down resources when not needed.
     + Maintaining high availability and performance.
5. **Why do we need Autoscaling Groups?**
   * Autoscaling Groups manage the scaling of EC2 instances based on policies, health checks, and schedules. They ensure:
     + Consistent scaling across multiple instances.
     + Automatic replacement of failed instances.
     + Control over the minimum and maximum number of instances.
6. **Which of the following services can be used if you want to capture client connection information from your load balancer at a particular time interval?**
   * **AWS CloudWatch Logs** can capture and store logs from the load balancer, including client connection information.
7. **Out of the following options, where does the user specify the maximum number of instances with the autoscaling commands?**
   * **Autoscaling Group** is where you specify the maximum number of instances.
8. **I created a web application with autoscaling. I observed that the traffic on my application is the highest on Wednesdays and Fridays between 9 AM and 7 PM. What would be the best solution for me to handle the scaling?**
   * Configure **scheduled scaling policies** to increase the number of instances during peak times on Wednesdays and Fridays and reduce them during off-peak hours.
9. **Load balancer types? Differences? Where to use?**
   * **Application Load Balancer (ALB)**: Best for HTTP/HTTPS traffic, content-based routing, and microservices architectures.
   * **Network Load Balancer (NLB)**: Best for TCP/UDP traffic, high performance, and handling millions of requests per second.
   * **Classic Load Balancer (CLB)**: Suitable for simple Layer 4 and Layer 7 balancing but is being phased out in favor of ALB and NLB.
10. **What is an elastic load balancer?**
    * **Elastic Load Balancer (ELB)** is a managed service that automatically distributes incoming traffic across multiple targets, such as EC2 instances, in one or more Availability Zones.
11. **Why do you use Elastic load balancer? Which load balancer have you used in your project?**
    * **Elastic Load Balancer** is used to ensure high availability, improve fault tolerance, and distribute traffic efficiently. In projects, I have used **Application Load Balancer** for HTTP/HTTPS traffic and **Network Load Balancer** for TCP/UDP traffic.
12. **A company's web application is using multiple Linux Amazon EC2 instances and storing data on Amazon EBS volumes. The company is looking for a solution to increase the resiliency of the application in case of a failure and to provide storage that complies with atomicity, consistency, isolation, and durability (ACID). What should a solutions architect do to meet these requirements?**
    * Use **Amazon RDS** for database services to meet ACID requirements. For increased resiliency, ensure that EC2 instances are behind an **Elastic Load Balancer** and use **Amazon EBS** with replication.
13. **You have an application running on an EC2 instance. You need to reduce the load on your instance as soon as the CPU utilization reaches 80 percent. How will you accomplish the job?**
    * Create an **Auto Scaling Policy** that triggers when CPU utilization exceeds 80 percent to add more instances to handle the load.
14. **A start-up company has a web application based in the us-east-1 Region with multiple Amazon EC2 instances running behind an Application Load Balancer across multiple Availability Zones. As the company's user base grows in the us-west-1 region, the company needs a solution with low latency and improved high availability. What should a solutions architect do to achieve it?**
    * Implement **Amazon Route 53** for geo-routing to direct users in the us-west-1 region to a load balancer in that region. Consider setting up an **Application Load Balancer** in us-west-1 and configure **cross-region replication** if needed.
15. **Can you change the instance type of the instances that are running in your application tier and are also using autoscaling? If yes, then how?**
    * Yes, update the **Launch Configuration** or **Launch Template** used by the Auto Scaling Group with the new instance type and then apply the changes. Instances will be replaced with new ones using the updated configuration.
16. **Explain how you can vertically scale an Amazon instance?**
    * To vertically scale an EC2 instance, stop the instance, change the instance type to a more powerful type, and then start the instance again. This increases the instance's CPU, memory, and storage capabilities.
17. **Suppose, you hosted an application on AWS that lets the users render images and do some general computing. Which of the below listed services can you use to route the incoming user traffic?**
    * **Amazon Route 53** can be used to route incoming user traffic to your application.
18. **You have an application running on your Amazon EC2 instance. You want to reduce the load on your instance as soon as the CPU utilization reaches 100 percent. How will you do that?**
    * Set up an **Auto Scaling Policy** to add more instances when CPU utilization reaches 100 percent, thereby reducing the load on the current instance.
19. **You have connected four instances to ELB. To automatically terminate your unhealthy instances and replace them with new ones, which functionality would you use?**
    * Use **Auto Scaling Groups** with health checks to automatically terminate and replace unhealthy instances.
20. **Can you vertically scale an Amazon Instance?**
    * Yes, vertically scale by stopping the instance, changing its type, and restarting it to increase its resources.
21. **When should you use the classic load balancer and the application load balancer?**
    * **Classic Load Balancer (CLB)**: Suitable for basic load balancing of HTTP/HTTPS and TCP traffic but is mostly phased out.
    * **Application Load Balancer (ALB)**: Best for HTTP/HTTPS traffic with advanced features like content-based routing and host-based routing.
22. **What is the use of lifecycle hooks in Autoscaling?**
    * Lifecycle hooks allow you to perform custom actions when instances are launching or terminating, such as running configuration scripts or performing pre-termination cleanup.
23. **How can we use Network Load Balancer and Application Load Balancer together in the same project?**
    * Use **Network Load Balancer (NLB)** for TCP/UDP traffic that requires high performance and **Application Load Balancer (ALB)** for HTTP/HTTPS traffic with content-based routing. NLB can forward traffic to ALB if needed.
24. **What is an elastic IP?**
    * An **Elastic IP (EIP)** is a static, public IP address designed for dynamic cloud computing. It allows you to easily reassign an IP address to a different EC2 instance.
25. **EIP default launch limit.**
    * The default limit for Elastic IPs per AWS account is typically 5. This limit can be increased by requesting a limit increase through the AWS Support Center.
26. **You accidentally stopped an EC2 instance in a VPC with an associated Elastic IP. If you start the instance again, what will be the result?**
    * The Elastic IP remains associated with the instance if you start it again, provided the instance is in the same VPC.
27. **How many elastic IPs can we create in one VPC?**
    * The limit for Elastic IPs in a VPC is typically 5 per region per AWS account. This limit can be increased by making a request to AWS.
28. **Can we use multiple load balancers on a single EC2 machine?**
    * Yes, multiple load balancers can be configured to route traffic to a single EC2 instance, each handling different types of traffic or using different protocols.
29. **A customer wants to stream 50 servers. What type of load balancer is used?**
    * **Network Load Balancer (NLB)** is suitable for handling high-throughput and low-latency traffic, making it ideal for streaming scenarios.
30. **What is connection draining?**
    * **Connection draining** (or deregistration delay) allows existing connections to complete before the load balancer stops routing traffic to an instance that is being deregistered or terminated.
31. **What are the different types of load balancers?**
    * **Application Load Balancer (ALB)**
    * **Network Load Balancer (NLB)**
    * **Classic Load Balancer (CLB)**
32. **Explain horizontal and vertical scaling in AWS?**
    * **Horizontal Scaling**: Adding more instances to handle increased load (scaling out/in).
    * **Vertical Scaling**: Increasing the resources (CPU, memory) of a single instance (scaling up/down).
33. **You have 50 database servers. Which load balancer would you use?**
    * **Network Load Balancer (NLB)** is suitable for distributing database traffic, especially if it involves high throughput and low latency requirements.
34. **What do you know about load balancer? Types of load balancer? How do you create them?**
    * Load balancers distribute traffic to multiple targets to ensure high availability and reliability. Types include **Application Load Balancer**, **Network Load Balancer**, and **Classic Load Balancer**. You create them via the AWS Management Console, AWS CLI, or AWS SDKs by specifying target groups, listeners, and routing rules.
35. **Can you explain the working of the network load balancer and its practical use?**
    * **Network Load Balancer (NLB)** operates at Layer 4 and handles TCP and UDP traffic. It is designed for high performance, handling millions of requests per second with low latency. It is used for scenarios requiring high throughput and reliability, such as gaming or real-time data processing.
36. **Can you assign a static IP to a load balancer?**
    * Yes, **Network Load Balancer (NLB)** supports static IP addresses.
37. **Difference between ALB and NLB.**
    * **ALB (Application Load Balancer)**: Operates at Layer 7 (Application Layer), supports HTTP/HTTPS, provides advanced routing capabilities.
    * **NLB (Network Load Balancer)**: Operates at Layer 4 (Transport Layer), supports TCP/UDP, handles high-throughput, low-latency traffic.
38. **Have you worked on provisioning on AWS like DB Load balancer and servers?**
    * Yes, I have experience with provisioning load balancers (ALB, NLB) and servers (EC2 instances) to manage traffic and scale applications effectively.
39. **Why does ELB hold many IP addresses?**
    * **Elastic Load Balancer (ELB)** may use multiple IP addresses to distribute traffic efficiently across multiple targets and to provide high availability and fault tolerance.
40. **Why is the classic load balancer not used many times?**
    * **Classic Load Balancer (CLB)** is often replaced by ALB and NLB due to their advanced features like content-based routing, improved performance, and support for modern protocols.
41. **What is a listener? Listener component in LB.**
    * A **listener** is a process that checks for connection requests. It is configured with a protocol and port (e.g., HTTP on port 80) and forwards requests to the target groups based on the rules defined.
42. **HTTP vs HTTPS.**
    * **HTTP**: Hypertext Transfer Protocol, unencrypted data transfer.
    * **HTTPS**: HTTP Secure, encrypted data transfer using TLS/SSL.
43. **In LB, is it possible to change the default port number? What is the condition?**
    * Yes, you can change the default port number when configuring a listener for the load balancer. Ensure that your target group and backend instances are set up to handle the new port.
44. **404 error?**
    * A **404 error** indicates that the requested resource could not be found on the server. It is typically a client-side error indicating that the URL is incorrect or the resource does not exist.
45. **In LB, which one is faster and why?**
    * **Network Load Balancer (NLB)** is generally faster than ALB because it operates at Layer 4 and handles TCP/UDP traffic with lower latency, while ALB performs content-based routing at Layer 7, which adds processing overhead.
46. **How will load decide where to go?**
    * The load balancer uses defined rules, algorithms (such as round-robin, least connections), and routing decisions based on the type of load balancer (ALB, NLB) to distribute traffic to the targets.
47. **What is the difference between load balancer and auto scaling?**
    * **Load Balancer** distributes incoming traffic across multiple instances to ensure high availability and reliability.
    * **Auto Scaling** adjusts the number of instances in response to changes in traffic or load to maintain performance and cost efficiency.
48. **How to add an existing instance in an autoscaling group?**
    * You cannot directly add an existing instance to an Auto Scaling Group. Instead, you can:
      + Create a new launch configuration or template with the desired instance type.
      + Update the Auto Scaling Group to use this configuration/template.
      + Terminate existing instances, and the Auto Scaling Group will launch new instances using the updated configuration.
49. **What is a launch group in autoscaling?**
    * **Launch Configuration** or **Launch Template** defines the settings for new instances launched by the Auto Scaling Group, including instance type, AMI, and other parameters.
50. **Can you explain what is auto scaling?**
    * **Auto Scaling** is a service that automatically adjusts the number of EC2 instances in your application based on demand, ensuring optimal performance and cost efficiency.
51. **What are the situations to move the traffic or how many ways to distribute the traffic from one EC2 to another, what weight you provide?**
    * Traffic can be distributed using:
      + **Load Balancers** (e.g., ALB, NLB) with different routing algorithms.
      + **DNS-based routing** with Route 53.
      + **Application-level adjustments** (e.g., using a reverse proxy).
      + Weights can be assigned in routing rules or configurations to control traffic distribution.
52. **If a thousand requests come to your web page and the load increases, what do you do and how can you decrease the load?**
    * **Scale Out**: Add more instances using Auto Scaling.
    * **Optimize**: Implement caching, use a Content Delivery Network (CDN) like CloudFront, or optimize the application’s performance.
    * **Load Balancing**: Distribute traffic across multiple instances with a load balancer.
53. **Benefits of auto scaling?**
    * **Cost Savings**: Only pay for the resources you need.
    * **Improved Availability**: Automatically replaces failed instances and scales during high demand.
    * **Performance**: Maintains application performance by adjusting resources as needed.
54. **Policies in auto scaling.**
    * **Scaling Policies**: Define how and when to add or remove instances based on metrics (e.g., CPU utilization).
    * **Scheduled Policies**: Scale resources based on a schedule (e.g., increase capacity at specific times).
    * **Dynamic Scaling Policies**: Adjust capacity based on demand in real time.
55. **Scale up and scale down?**
    * **Scale Up**: Increase the instance size (vertical scaling) or number of instances (horizontal scaling).
    * **Scale Down**: Decrease the instance size or number of instances.
56. **How much time is required to pass a health check?**
    * Health check timing is configurable. Typically, a health check might take 30 seconds to 1 minute, depending on the health check interval and timeout settings.
57. **Min and max. desired value? Explain.**
    * **Min**: Minimum number of instances in the Auto Scaling Group.
    * **Max**: Maximum number of instances in the Auto Scaling Group.
    * **Desired**: The target number of instances that the Auto Scaling Group should maintain based on the scaling policies.
58. **What is horizontal and vertical scaling?**
    * **Horizontal Scaling**: Adding more instances to handle increased load (scaling out/in).
    * **Vertical Scaling**: Increasing the capacity of existing instances (scaling up/down).
59. **Does rebalancing work in auto scaling?**
    * Yes, **rebalancing** works in Auto Scaling by redistributing instances across Availability Zones to maintain a balanced distribution based on the current configuration.
60. **Warm-up time?**
    * **Warm-up Time** is the period during which newly launched instances are considered “warm” and should not be terminated or replaced. It allows instances to fully initialize and handle traffic before being evaluated by health checks.
61. **What information do you collect from your client regarding auto scaling?**
    * **Application Requirements**: Traffic patterns, performance metrics.
    * **Scaling Policies**: Desired thresholds for scaling up or down.
    * **Capacity Needs**: Minimum and maximum instance counts.
    * **Health Check Configuration**: Criteria for instance health.
62. **In a target group, you find that an instance is unhealthy. How do you troubleshoot that?**
    * **Check Health Checks**: Verify health check configurations and thresholds.
    * **Examine Logs**: Look at application and instance logs for errors.
    * **Review Instance Metrics**: Check CPU, memory, and network usage.
    * **Verify Network Configuration**: Ensure that security groups and network settings are correct.
63. **How would you set up a load balancer for an application?**
    * **Create a Load Balancer**: Choose the type (ALB, NLB, CLB).
    * **Configure Listeners**: Define protocols and ports.
    * **Set Up Target Groups**: Specify the instances or IPs to receive traffic.
    * **Define Routing Rules**: Determine how traffic is distributed to target groups.
    * **Configure Health Checks**: Set up criteria for instance health.
64. **What is an Elastic IP?**
    * An **Elastic IP (EIP)** is a static, public IP address designed for dynamic cloud computing that can be associated with or reassigned to EC2 instances as needed.
65. **Explain various types of elastic load balancers in AWS.**
    * **Application Load Balancer (ALB)**: Best for HTTP/HTTPS traffic with advanced routing.
    * **Network Load Balancer (NLB)**: Best for TCP/UDP traffic with high performance.
    * **Classic Load Balancer (CLB)**: Older type, suitable for basic load balancing tasks.

4o mini

• EKS, ECS questions • Services and nodes in K8s • Replication controller and replica sets in K8s • Why secrets in Kubernetes? • Automatic code changes in Kubernetes from Git, how to configure it? • Stateful and stateless application differences? • What state does the Security Group have? • Terraform dynamic block. • Services in K8s. • How to deploy a stateful resource in Kubernetes? • What is Ingress in K8s? • Some scenario-based questions in K8s. • What is a container orchestration engine? • Can you explain the differences between Amazon ECS and Amazon EKS? give anser below questions

ChatGPT

**EKS, ECS Questions**

1. **What is Amazon EKS?**
   * **Amazon Elastic Kubernetes Service (EKS)** is a managed Kubernetes service that simplifies running Kubernetes on AWS. It handles the control plane management, including scaling, patching, and securing Kubernetes clusters, allowing you to focus on deploying and managing your applications.
2. **What is Amazon ECS?**
   * **Amazon Elastic Container Service (ECS)** is a fully managed container orchestration service that supports Docker containers and allows you to run and scale containerized applications on AWS. It integrates with other AWS services and provides options for scheduling and managing containers.

####################################

**Linux, commands & Scripting**

####################################

1. **List some AWS CLI commands**
   * aws s3 ls – List S3 buckets or objects.
   * aws ec2 describe-instances – Describe EC2 instances.
   * aws iam create-user – Create an IAM user.
   * aws lambda invoke – Invoke a Lambda function.
   * aws cloudformation deploy – Deploy a CloudFormation stack.
2. **How can you create a zero-byte file using which command?**
   * You can use the touch command: touch filename.
3. **How to check if a particular port is open or not?**
   * Use netstat -an | grep port\_number or ss -an | grep port\_number.
4. **How to kill services in Linux? (Command to use)**
   * Use kill followed by the process ID: kill PID. For a more forceful termination, use kill -9 PID.
5. **How to check the services running in Linux? (Command to use)**
   * Use systemctl list-units --type=service or service --status-all.
6. **Using which command will you list out all the empty files in Linux?**
   * Use find /path/to/directory -type f -empty.
7. **Define local variable.**
   * A local variable is a variable that is only accessible within the scope of a function or a block in a script or program.
8. **Define global variable.**
   * A global variable is a variable that is accessible throughout the entire program or script, across all functions and blocks.
9. **Difference between local and global variables.**
   * **Local Variables:** Accessible only within a specific function or block. They are created when the function is called and destroyed when the function exits.
   * **Global Variables:** Accessible throughout the entire program or script. They are created when the program starts and exist until the program terminates.
10. **Conditional in shell script.**
    * In shell scripts, conditionals are used to make decisions. They are typically written using if, then, elif, and else statements. For example:

bash

Copy code

if [ condition ]; then

# commands

elif [ another\_condition ]; then

# commands

else

# commands

fi

1. **What commands can you use to check ports and running processes in Linux?**
   * For checking ports: netstat -tuln or ss -tuln.
   * For checking running processes: ps aux or top.
2. **What information is stored in fstab?**
   * The fstab file contains information about disk partitions and their mount points. It includes device files, mount points, file system types, mount options, and dump and pass values.
3. **Write a shell script for comparing two numbers.**

bash

Copy code

#!/bin/bash

echo "Enter first number: "

read num1

echo "Enter second number: "

read num2

if [ $num1 -gt $num2 ]; then

echo "$num1 is greater than $num2"

elif [ $num1 -lt $num2 ]; then

echo "$num1 is less than $num2"

else

echo "$num1 is equal to $num2"

fi

1. **In Linux, what diff command have you used?**
   * The diff command compares two files line by line. Example: diff file1 file2.
2. **How will you check connectivity between servers?**
   * Use ping to check basic connectivity: ping server\_ip.
   * Use telnet or nc to check specific ports: telnet server\_ip port or nc -zv server\_ip port.
3. **How familiar are you with AWS CLI?**
   * I am familiar with AWS CLI and can use it for managing various AWS services, including EC2, S3, IAM, and CloudFormation.
4. **How will you check patterns from multiple files?**
   * Use grep with the -r (recursive) option to search for patterns in multiple files: grep -r "pattern" /path/to/directory.
5. **Can you provide step-by-step AWS CLI commands for syncing data from an S3 bucket to a local machine?**

bash

Copy code

aws s3 sync s3://bucket-name /local/path

1. **If you want to delete some files older than 30 days, how will you do that in Linux?**

bash

Copy code

find /path/to/directory -type f -mtime +30 -exec rm {} \;

1. **Command to find empty files in a given directory?**

bash

Copy code

find /path/to/directory -type f -empty

1. **Commands you will use for configuring SSH connectivity between 2 machines and what files will be present in the .ssh folder?**
   * **Configure SSH:** Use ssh-keygen to generate a key pair, ssh-copy-id user@remote\_host to copy the public key to the remote machine.
   * **.ssh Folder Files:** id\_rsa (private key), id\_rsa.pub (public key), known\_hosts (hosts), config (SSH configuration).
2. **How to schedule a shell script in Unix machines?**
   * Use cron to schedule tasks. Edit the crontab file with crontab -e and add a line like:

bash

Copy code

\* \* \* \* \* /path/to/script.sh

1. **Command to get load average?**
   * Use uptime or top. Example with uptime:

bash

Copy code

uptime

1. **Need to identify IP addresses in the log file and count the IP addresses in the log file?**

bash

Copy code

awk '{print $1}' /path/to/logfile | sort | uniq -c

1. **What is the command for checking the running process? How to get the PID of the process?**
   * Use ps aux to check running processes. The PID is in the second column. Alternatively, use pgrep process\_name to get the PID directly.
2. **Command to get whether a certain port is listening or not?**
   * Use netstat -an | grep port\_number or ss -tuln | grep port\_number.
3. **How to identify the number of params that have been sent to the shell script?**
   * Use $# in the shell script to get the number of positional parameters.

bash

Copy code

echo "Number of parameters: $#"

1. **Command to delete an empty line in a file?**
   * Use sed to delete empty lines:

bash

Copy code

sed -i '/^$/d' /path/to/file

1. **What is exit status?**
   * The exit status is a code returned by a command or script to indicate success or failure. A status of 0 usually means success, while any non-zero value indicates an error.
2. **Given a machine, how will you identify which machine it is?**
   * Use hostname to get the machine name and uname -a for detailed system information.
3. **Any functional shell script you have created?**
   * Example: A script to back up files.

bash

Copy code

#!/bin/bash

tar -czvf /backup/backup-$(date +%F).tar.gz /path/to/files

1. **What is the weight terminology used in shell scripting?**
   * **Weight terminology** usually refers to how the system prioritizes or assigns importance to certain tasks or processes, though it's not a standard term in shell scripting.
2. **Types of control instructions available in shell scripting?**
   * **Conditional Statements:** if, elif, else, case.
   * **Loops:** for, while, until.
   * **Control Flow:** break, continue, return.
3. **Alternative for if-else statement in bash?**
   * Use case statements:

bash

Copy code

case $variable in

pattern1)

# commands ;;

pattern2)

# commands ;;

\*)

# default commands ;;

esac

1. **How will you take MYSQL backup with the help of a shell script?**

bash

Copy code

#!/bin/bash

TIMESTAMP=$(date +"%F")

BACKUP\_DIR="/backup/$TIMESTAMP"

mkdir -p "$BACKUP\_DIR"

mysqldump -u username -p'password' database\_name > "$BACKUP\_DIR/backup.sql"

Feel free to adjust these answers based on your needs or context!

###############################################################

**AWS, Scenarios & Cloud computing extra questions:**

###############################################################

 **What are the edge locations in AWS?**

* **Edge locations** are data centers used by Amazon CloudFront to cache content close to end-users, reducing latency and improving access speed for content delivery.

 **What is Snowball in AWS?**

* **AWS Snowball** is a physical device used to transfer large amounts of data into and out of AWS securely. It helps in cases where internet transfer would be too slow or impractical.

 **What is Redshift in AWS?**

* **Amazon Redshift** is a fully managed data warehouse service that allows you to run complex queries and perform analytics on large volumes of data quickly and cost-effectively.

 **What is the boot time taken for an instance stored backed AMI?**

* **Instance store-backed AMIs** typically have faster boot times compared to EBS-backed AMIs because the instance store volumes are physically attached to the host machine.

 **Which of the following options will you use if you have to move data over long distances using the Internet, from instances that are spread across countries to your Amazon S3 bucket?**

* **AWS Direct Connect** or **AWS DataSync** are suitable options for transferring data over long distances. **DataSync** automates data transfer, while **Direct Connect** provides a dedicated network connection.

 **I am unable to list on port; how to check and how to configure any port in Linux?**

* **Check port status** using netstat -tuln or ss -tuln. **Configure a port** by editing firewall rules (e.g., iptables or firewalld) and ensuring the application is listening on the desired port.

 **Unable to ping Instance We launched a Windows 2019 IIS server in the Ohio region and deployed a dynamic website in this server, in addition, the webserver also connected with a backend MS-SQL server to store and access data related to the application. Our users were able to access the website over the Internet. The next day our client informed us that they were able to access the website, but weren’t able to ping the server from the Internet. To ensure ICMP rule in Security Group, we checked, and the Security Group had allowed rule from 0.0.0.0/0. Would you try to help troubleshoot the issue?**

* **ICMP ping requests might be blocked** by the Network ACLs or the instance’s operating system firewall (Windows Firewall). Check both the ACL rules and the Windows Firewall settings to ensure ICMP traffic is allowed.

 **Usually, you have noticed that one EBS volume can be connected with one EC2 instance, our company wants to run a business-critical application on multiple instances in a single region and need to store all instances' output in a single storage within the VPC. Instead of using EFS, our company is recommending the use of multi-attach volume with instances. As an architect, you need to suggest what instance type and EBS volumes they should use.**

* **For multi-attach EBS volumes**, use **Provisioned IOPS SSD (io2)** volumes for high performance and reliability. Ensure instances are compatible with multi-attach (e.g., certain Nitro-based instance types).

 **What is SimpleDB?**

* **Amazon SimpleDB** is a non-relational database service that provides a simple interface for storing and querying structured data. It is designed for applications requiring flexible, schema-less storage.

 **What is Amazon SQS and what is its purpose?**

* **Amazon SQS (Simple Queue Service)** is a fully managed message queuing service that enables you to decouple and scale microservices, distributed systems, and serverless applications by sending, storing, and receiving messages.

 **What kind of IP address can you use for your customer gateway (CGW) address?**

* **The customer gateway (CGW) address** must be a static IP address assigned to your on-premises hardware or virtual appliance. This IP address is used for establishing a VPN connection with AWS.

 **Which service offered by Amazon will you choose if you want to collect and process e-commerce data for near real-time analysis?**

* **Amazon Kinesis** is suitable for collecting, processing, and analyzing streaming data in near real-time. You might also use **AWS Lambda** or **Amazon Redshift** for processing and analysis.

 **Which service provided by AWS can you use to transfer objects from your data center when you are using Amazon CloudFront?**

* **AWS DataSync** can be used to transfer data from your on-premises data center to Amazon S3, which can then be served via Amazon CloudFront.

 **What are the important features of Amazon CloudSearch?**

* **Amazon CloudSearch** offers features such as:
  + **Full-text search** with support for text processing.
  + **Faceted search** for filtering results based on metadata.
  + **Autocomplete** and **suggestions**.
  + **Multi-language support**.
  + **Scaling** to handle large volumes of search traffic.

 **I created a key in North Virginia region to encrypt my data in Oregon region. I also added three users to the key and an external AWS account. Then, to encrypt an object in S3, when I tried to use the same key, it was not listed. Where did I go wrong?**

* **KMS keys are region-specific.** You need to create the KMS key in the same region as your S3 bucket (Oregon) to encrypt objects stored in S3.

 **Which Amazon solution will you use if you want to accelerate moving petabytes of data in and out of AWS, using storage devices that are designed to be secure for data transfer?**

* **AWS Snowball** is designed for securely transferring large amounts of data into and out of AWS.

 **What do you mean by AWS?**

* **AWS (Amazon Web Services)** is a comprehensive and widely adopted cloud platform that offers a variety of services, including computing power, storage, and databases, among others, delivered via the internet.

 **I want to start and stop my EC2 instance at 9 am and 7 pm; how do I achieve this in AWS?**

* **Use AWS Lambda and CloudWatch Events** to create scheduled tasks that start and stop your EC2 instance at specific times. Define CloudWatch rules to trigger the Lambda function at 9 am and 7 pm.

 **I want to host a static website; which service should I use in AWS, and why?**

* **Amazon S3** is ideal for hosting static websites because it offers scalable storage, easy setup, and low cost. You can configure an S3 bucket for static website hosting and serve content directly from it.

 **I host a website on a Tomcat server with a public IP, but I am unable to access it. How do I troubleshoot on the OS level, instance level, and VPC level?**

* **OS Level**: Check if Tomcat is running and listening on the correct port.
* **Instance Level**: Verify that the security group allows inbound traffic on the required port.
* **VPC Level**: Ensure that Network ACLs and route tables allow the necessary traffic.

 **What is SELinux?**

* **SELinux (Security-Enhanced Linux)** is a security module in Linux that provides an additional layer of security by enforcing access controls and policies for processes and files.

 **A Company has a running Web Application Server in the N. Virginia region and the server has a large size EBS volume of approximately 500 GB. To see the demand of business, the company needs to migrate the server from the current region to another AWS account’s Mumbai location. What is the best way to migrate the server from the current location to the Mumbai region? And what information does the AWS administrator require about AWS A/C?**

* **The best way** is to create an EBS snapshot, copy the snapshot to the Mumbai region, and then create a new volume from the snapshot in the Mumbai region. For migrating between AWS accounts, ensure that the snapshot is shared with the target account.

 **Which Amazon solution will you use if you want to accelerate moving petabytes of data in and out of AWS, using storage devices that are designed to be secure for data transfer?**

* **AWS Snowball** or **AWS Snowmobile** (for even larger data transfers) would be suitable for securely transferring large volumes of data.

 **Your organization wants to send and receive compliance emails to its clients using its own email address and domain. What service would you suggest for achieving the same in an easy and cost-effective way?**

* **Amazon SES (Simple Email Service)** is a cost-effective solution for sending and receiving emails using your own domain.

 **Your company wants you to propose a solution so that the company’s data center can be connected to Amazon cloud network. What would be your proposal?**

* **AWS Direct Connect** would be suitable for establishing a dedicated network connection from your data center to AWS, providing a more reliable and faster connection than internet-based connections.

 **What is AWS Lambda?**

* **AWS Lambda** is a serverless computing service that allows you to run code in response to events without provisioning or managing servers. You pay only for the compute time you consume.

 **What are the types of AMI provided by AWS?**

* **Types of AMIs (Amazon Machine Images)**:
  + **AWS Marketplace AMIs**: Pre-configured by third parties.
  + **Custom AMIs**: Created by users with specific configurations.
  + **AWS-provided AMIs**: Standard images provided by AWS.

 **What is Amazon ElastiCache?**

* **Amazon ElastiCache** is a fully managed caching service that supports Redis and Memcached, designed to improve application performance by caching frequently accessed data.

 **What do you know about Amazon WorkSpaces?**

* **Amazon WorkSpaces** is a managed, secure Desktop-as-a-Service (DaaS) solution that allows you to provision and manage virtual desktops in the cloud.

 **Your organization is using DynamoDB for its application. This application collects data from its users every 10 minutes and stores it in DynamoDB. Then every day, after a particular time interval, the data (respective to each user) is extracted from DynamoDB and sent to S3. Then, the application visualizes this data to the users. You are asked to propose a solution to help optimize the backend of the application for latency at a lower cost. What would you recommend?**

* **Use DynamoDB Streams** to capture changes and process them with AWS Lambda to send data to S3. This approach can help with real-time data processing and reduce costs by leveraging serverless architecture.

 **What is AWS EMR?**

* **AWS EMR (Elastic MapReduce)** is a cloud service that simplifies running big data frameworks like Apache Hadoop, Spark, and HBase to process and analyze large datasets.

 **What are the different types of cloud services?**

* **Infrastructure as a Service (IaaS)**: Provides virtualized computing resources over the internet (e.g., EC2, S3).
* **Platform as a Service (PaaS)**: Provides a platform to develop, run, and manage applications (e.g., AWS Elastic Beanstalk).
* **Software as a Service (SaaS)**: Delivers software applications over the internet (e.g., AWS WorkMail).

 **What is Amazon EMR?**

* (Duplicate question, already answered above.)

 **What are the types of AMI provided by AWS?**

* (Duplicate question, already answered above.)

 **List different types of cloud services.**

* **IaaS**: Compute, storage, networking.
* **PaaS**: App hosting, databases.
* **SaaS**: Email, CRM applications.

 **What are the various layers of Cloud Architecture explained in AWS training?**

* **AWS Cloud Architecture Layers**:
  + **Infrastructure Layer**: Compute, storage, networking.
  + **Platform Layer**: Databases, messaging services.
  + **Application Layer**: Application hosting, developer tools.
  + **Management and Governance Layer**: Monitoring, security, cost management.

 **What are the main differences between AWS and OpenStack?**

* **AWS**: A fully managed cloud service provider with extensive services and global infrastructure.
* **OpenStack**: An open-source cloud platform that allows you to build and manage your own private or public cloud infrastructure.

 **What is the service that exists to redundantly cache data and images in AWS?**

* **Amazon CloudFront** is a content delivery network (CDN) that caches content such as data and images at edge locations to reduce latency and improve performance.

 **How do you solve a bug in the production environment if you do not have access to CloudWatch logs?**

* **Troubleshoot by** checking application logs stored on the instance, verifying configurations, and using diagnostic tools like netstat or curl to test connectivity and performance.

 **Explain with a diagram how the request flows from the customer to the application and database and back to the customer in your infrastructure.**

* **Diagram Explanation**:
  1. **Customer**: Sends a request via the internet.
  2. **CloudFront**: Routes the request to the nearest edge location.
  3. **Load Balancer**: Distributes traffic to multiple EC2 instances.
  4. **EC2 Instance**: Processes the request and may query a database.
  5. **Database (RDS/DynamoDB)**: Retrieves or stores data.
  6. **Response**: The application sends data back through the load balancer and CloudFront to the customer.

 **AWS Lambda overview and how it works.**

* **AWS Lambda** runs code in response to events without managing servers. You define functions that are triggered by events such as S3 uploads, DynamoDB updates, or HTTP requests.

 **Shell or any other scripting language to invoke Lambda function once an EC2 instance is stopped.**

* **Use AWS CLI or Boto3 (Python)** to invoke a Lambda function. For example, with AWS CLI: aws lambda invoke --function-name YourFunctionName outputfile.txt.

 **How do you face downtime? How do you troubleshoot in AWS?**

* **Facing Downtime**: Identify the issue by checking CloudWatch metrics, instance logs, and service status.
* **Troubleshooting**: Review logs, use AWS health dashboards, and test components of your architecture to isolate and resolve the problem.

 **AWS Lambda overview and how it works.**

* (Duplicate question, already answered above.)

 **Shell or any other scripting language to invoke Lambda function once an EC2 instance is stopped.**

* (Duplicate question, already answered above.)

 **AWS Lifecycle Manager, AWS CodeCommit, AWS CodePipeline, AWS CloudFormation.**

* **AWS Lifecycle Manager**: Automates the creation, retention, and deletion of EBS snapshots.
* **AWS CodeCommit**: A source control service for managing Git repositories.
* **AWS CodePipeline**: A continuous integration and delivery service for automating build and deployment pipelines.
* **AWS CloudFormation**: Provides infrastructure as code to create and manage AWS resources using templates.

 **What is cloud computing?**

* **Cloud computing** is the delivery of computing services (servers, storage, databases, networking, software) over the internet ("the cloud") to offer flexible resources, faster innovation, and economies of scale.

 **Types of clouds and its examples?**

* **Public Cloud**: AWS, Azure, Google Cloud.
* **Private Cloud**: VMware vSphere, OpenStack.
* **Hybrid Cloud**: A combination of public and private clouds, e.g., AWS Outposts.

 **What do you use for automatic backup of instances and volumes in your project?**

* **Use AWS Backup** for automatic backups of EC2 instances, EBS volumes, and other AWS resources. You can also use **Amazon Data Lifecycle Manager** for EBS snapshots.

 **Why you choose AWS cloud?**

* **AWS offers** a broad range of services, high scalability, reliable performance, global reach, and a pay-as-you-go pricing model, making it a flexible and cost-effective solution for various business needs.

 **Difference between zone and regions.**

* **Region**: A geographic area that contains multiple isolated locations (data centers).
* **Availability Zone**: An isolated location within a region with its own power, networking, and connectivity.

 **Region-specific services in AWS.**

* **Region-specific services** include those that are only available in certain AWS regions, such as AWS Outposts or certain compliance-related services.

 **Non-region specific services.**

* **Non-region specific services** include AWS IAM, AWS CloudTrail, and AWS Config, which operate globally across regions.

 **What do you actually work in AWS, which services you used?**

* This would be specific to your role and project requirements. Typically, users work with services like EC2, S3, RDS, Lambda, CloudFormation, and others depending on their use case.

 **What is AWS?**

* (Duplicate question, already answered above.)

 **What are platforms where we can use AWS?**

* **AWS platforms** include a wide range of services for computing, storage, databases, machine learning, analytics, and more, accessible through web, mobile, and desktop applications.

 **How do you manage your infrastructure?**

* **Manage infrastructure** using AWS services such as **AWS CloudFormation** for infrastructure as code, **AWS Systems Manager** for configuration management, and **AWS OpsWorks** for managing stacks and applications.

 **Can you tell me the difference between availability zone and the region?**

* (Duplicate question, already answered above.)

 **How do you take automatic rollbacks of your application? What if it fails in production?**

* **Use AWS CodeDeploy** to automate application deployments with rollback capabilities. For failures, you can set up alarms and notifications to address issues quickly.

 **Your organization has decided to have all their workload on the public cloud. But, due to certain security concerns, your organization decides to distribute some of the workload on private servers. You are asked to suggest a cloud architecture for your organization. What will be your suggestion?**

* **Consider a hybrid cloud architecture** with AWS for public cloud workloads and private servers for sensitive data. Use AWS Direct Connect or VPN for secure connectivity between the public cloud and private servers.

 **How do you manage the cost pricing of each environment, i.e., DEV, QA, UAT?**

* **Use AWS Cost Allocation Tags**, **AWS Budgets**, and **Cost Explorer** to track and manage costs by environment. Set up separate accounts or use resource tags to distinguish between different environments.

 **Approach for migrating all servers to AWS and managing cost-cutting?**

* **Plan the migration** by using AWS Migration Hub, **optimize costs** by selecting appropriate instance types, using Reserved Instances or Savings Plans, and taking advantage of AWS Cost Management tools.

 **Your organization has decided to transfer its business processes to the public cloud. However, they want some of their information/data to be accessed only by the management team. The rest of the resources will be shared among the employees of the firm. You have to suggest a suitable cloud architecture for your firm along with the reason for the choice.**

* **Implement a multi-tier architecture** with IAM roles and policies to restrict access. Use AWS Organizations to manage accounts and control access, and encrypt sensitive data with AWS KMS.

 **Suppose your firm is hosting an application on AWS that helps users render images and perform general computation tasks. Your firm’s management team has suggested using an application load balancer for routing the incoming traffic on the hosted application. Explain how an application load balancer is a good choice for routing the incoming traffic?**

* **An Application Load Balancer (ALB)** provides intelligent routing based on content and supports HTTP/HTTPS traffic. It can handle varying loads, distribute traffic across multiple targets, and provide features like SSL termination and path-based routing.

 **You have recently assigned various EC2 instances for your business website across different availability zones. Since your website performs a large number of read/write operations per minute, you have also used a Multi-AZ RDS DB instance (extra-large). It was going smoothly as per your plans until you discovered read contention on RDS MySQL. How are you going to solve this issue for enhancing the performance of your website?**

* **Optimize read performance** by using Amazon RDS Read Replicas to offload read operations from the primary instance. You can also consider database indexing, query optimization, and increasing the instance size if needed.

 **Your firm wants to connect the data center of its organization to the Amazon cloud environment for faster accessibility and performance. What course of action will you suggest for the stated scenario?**

* **Suggest AWS Direct Connect** for a dedicated and high-performance connection between your data center and AWS, reducing latency and increasing bandwidth.

 **Explain various types of cloud service models in brief.**

* **Infrastructure as a Service (IaaS)**: Provides virtualized computing resources (e.g., AWS EC2).
* **Platform as a Service (PaaS)**: Offers a platform to build, deploy, and manage applications (e.g., AWS Elastic Beanstalk).
* **Software as a Service (SaaS)**: Delivers software applications over the internet (e.g., AWS WorkMail).

##################################

**Elastic Computing (EC2):**

##################################

Here are the answers to your EC2-related questions:

1. **What is elastic computing?**
   * Elastic computing refers to the ability to automatically scale computing resources up or down based on demand. It ensures that the right amount of resources is allocated to handle workloads efficiently, optimizing cost and performance.
2. **How many instances can I run?**
   * The number of EC2 instances you can run is limited by your account's service quotas. By default, AWS allows a certain number of instances per region, but you can request an increase if needed.
3. **Explain what T2 instances are?**
   * T2 instances are a type of burstable performance instance in AWS. They provide a baseline level of CPU performance with the ability to burst to higher levels of CPU performance when needed. T2 instances use credits to manage burst performance.
4. **Can you change the private IP address of an EC2 instance while it is in running or in a stopped state?**
   * You cannot change the private IP address of an EC2 instance while it is running. However, you can change it when the instance is stopped by modifying the network interface associated with the instance.
5. **What does an AMI include?**
   * An AMI (Amazon Machine Image) includes a template for your instance that contains an operating system, application server, applications, and related configurations. It provides the necessary information to launch a fully configured instance.
6. **What are the different EC2 instances based on their cost? Or can you explain EC2 pricing model?**
   * EC2 instances are priced based on instance type, usage, and pricing model (On-Demand, Reserved, Spot). On-Demand instances are billed per hour or second, Reserved instances offer discounted rates for a commitment of 1 or 3 years, and Spot instances are available at a lower cost but can be terminated by AWS based on capacity needs.
7. **What type of instances have you worked on? Tell me their configuration.**
   * I have worked with various instance types, including T2.micro (1 vCPU, 1 GB RAM) for small-scale applications and M5.large (2 vCPUs, 8 GB RAM) for more demanding applications. Configuration varies based on application requirements and workload.
8. **What is Amazon Elastic Compute Cloud (EC2), and also explain its features?**
   * Amazon EC2 is a web service that provides resizable compute capacity in the cloud. Features include flexible instance types, auto-scaling, various storage options, support for different operating systems, and integration with other AWS services.
9. **Discuss the pricing models for the Amazon EC2 instance.**
   * **On-Demand Instances:** Pay for compute capacity by the hour or second with no long-term commitments.
   * **Reserved Instances:** Reserve instances for a one or three-year term and receive a significant discount compared to On-Demand pricing.
   * **Spot Instances:** Purchase unused capacity at reduced rates, with the risk of instances being terminated when AWS needs the capacity back.
10. **Types of instances? Different types of EC2 instances you have worked on.**
    * EC2 instances are categorized by use case, such as General Purpose (e.g., T2, M5), Compute Optimized (e.g., C5), Memory Optimized (e.g., R5), Storage Optimized (e.g., I3), and GPU Instances (e.g., P3). I have worked on T2.micro, M5.large, and C5.xlarge instances.
11. **What is the difference between stopping and termination of instances?**
    * **Stopping:** The instance is shut down but can be restarted later. The EBS volume is retained, and you are not billed for instance usage, only for storage.
    * **Termination:** The instance is permanently deleted, and any attached EBS volumes (unless marked as "delete on termination") are also deleted. Termination is irreversible.
12. **What is EC2? What does the elastic word indicate in EC2?**
    * EC2 (Elastic Compute Cloud) is a scalable compute service that provides virtual servers in the cloud. The term "elastic" indicates the ability to automatically scale computing resources up or down based on demand.
13. **What are the essential modules used in EC2?**
    * Essential modules include EC2 Instances, EBS (Elastic Block Store) for storage, Elastic Load Balancing (ELB), Auto Scaling, Security Groups, and Network Interfaces.
14. **Types of EC2 instances (General purpose, CPU optimized, etc).**
    * **General Purpose:** T2, M5
    * **Compute Optimized:** C5
    * **Memory Optimized:** R5, X1
    * **Storage Optimized:** I3, D2
    * **GPU Instances:** P3, G4
15. **Can you change types of instances?**
    * Yes, you can change the instance type of a running EC2 instance by stopping the instance, modifying its instance type, and then restarting it. The instance type change requires a reboot.
16. **Can you tell the frequency of launching EC2 in your project? Difference between t2.micro and t3.large? What is T4?**
    * The frequency of launching EC2 instances depends on the project’s needs. **T2.micro** has 1 vCPU and 1 GB of RAM, suitable for low-traffic applications. **T3.large** has 2 vCPUs and 8 GB of RAM, providing more compute power and memory. **T4g** instances are a newer generation offering better performance and cost efficiency with ARM-based processors.
17. **What will happen when you delete an EC2 instance which has attached volumes and EIP?**
    * When you terminate an EC2 instance, any attached EBS volumes that are marked for deletion will be deleted. Elastic IPs (EIPs) are disassociated but not deleted, and you may incur charges if you do not release them.
18. **Which key is used to access an instance through Putty?**
    * You use a private key file (usually a .pem file) to access an EC2 instance through PuTTY. The .pem file must be converted to a .ppk file using PuTTYgen.
19. **Suppose I enable SSH and in the outbound SSH is not enabled, how do I access the EC2 instance?**
    * If outbound SSH is not enabled, you can use a VPN or a bastion host to access the instance. Alternatively, ensure that security group rules allow inbound SSH traffic.
20. **Suppose we launch an EC2 instance, due to some reason my server is shutdown, and my IP changes, how can you overcome it?**
    * To handle IP changes, you can use an Elastic IP, which provides a static IP address that can be reassigned to different instances as needed.
21. **Types of EC2 instances? Which are the cheapest?**
    * Types include General Purpose, Compute Optimized, Memory Optimized, Storage Optimized, and GPU Instances. **T2.micro** and **T3.micro** are among the cheapest instances, suitable for low-demand applications.
22. **Can I upgrade my EC2 instance after launching?**
    * Yes, you can upgrade an EC2 instance by stopping the instance, changing its instance type, and then starting it again.
23. **What are on-demand and spot instances?**
    * **On-Demand Instances:** Pay for compute capacity by the hour or second with no long-term commitments.
    * **Spot Instances:** Purchase unused capacity at reduced rates, with the risk of termination if AWS needs the capacity back.
24. **What is instance store volume exactly?**
    * Instance store volumes are temporary storage that is physically attached to the host machine. They are ideal for temporary data but are lost when the instance is stopped or terminated.
25. **How can we safeguard our EC2 instance?**
    * Safeguard EC2 instances by configuring security groups and network ACLs, using IAM roles for access control, enabling encryption for data in transit and at rest, and implementing regular backups.
26. **What is the meaning of General purpose instance?**
    * General purpose instances are designed to offer a balanced mix of compute, memory, and network resources. They are suitable for a wide range of applications and workloads.
27. **Different ways to launch EC2 instance or ways to launch instances (explain in brief).**
    * **AWS Management Console:** Graphical interface for launching instances.
    * **AWS CLI:** Command-line tool for scripting and automation.
    * **AWS SDKs:** Programmatic access via APIs.
    * **CloudFormation:** Infrastructure as code for automated deployments.
28. **On-Demand vs Spot instance? When do spot instances use?**
    * **On-Demand Instances:** Used when you need predictable costs and continuous availability.
    * **Spot Instances:** Used for flexible, cost-effective computing, such as batch processing or fault-tolerant applications.
29. **Why do we use On-Demand instances?**
    * On-Demand instances are used for applications with unpredictable workloads, where the flexibility to scale up or down without long-term commitments is required.
30. **EC2 is region-specific or zone-specific?**
    * EC2 instances are region-specific but can be launched in any availability zone within the region. Each region is isolated from others to ensure high availability.
31. **Without disk, is it possible to launch an EC2 instance?**
    * No, an EC2 instance requires at least one disk for the operating system and data. You can use an EBS volume or instance store for this purpose.
32. **How many EC2 instances can we launch, i.e. default numbers?**
    * The default limit varies by region and account. Typically, you start with a limit of 20 instances per region, but this can be increased by requesting a quota adjustment from AWS.
33. **How many categories are available in EC2 instances?**
    * EC2 instances are categorized into several types, including General Purpose, Compute Optimized, Memory Optimized, Storage Optimized, and GPU Instances.
34. **How will you calculate the cost of an EC2 instance and other services?**
    * The cost can be calculated based on instance type, pricing model (On-Demand, Reserved, Spot), and additional resources like EBS volumes and data transfer. AWS Pricing Calculator can help estimate costs.
35. **How many EC2 instances can we create in a VPC?**
    * There is no specific limit on the number of EC2 instances you can create in a VPC; however, you are subject to account limits and VPC resource constraints.
36. **How to connect different machines if one is a created instance and one is a Linux machine?**
    * You can connect instances using SSH if they are on the same network or VPC. Ensure security group rules and network ACLs allow inbound and outbound traffic between the instances.
37. **Who will take the decision about which configuration is needed?**
    * The decision is typically made by the project or system architect, in collaboration with developers and operations teams, based on application requirements and performance needs.
38. **Is an EC2 instance region-specific or non-region specific?**
    * EC2 instances are region-specific, meaning they are launched within a specific AWS region, but they can be accessed from anywhere, provided network and security configurations allow it.
39. **Which EC2 instance did you launch? Why? Tell me the configuration.**
    * I launched an **M5.large** instance (2 vCPUs, 8 GB RAM) for its balanced compute, memory, and network resources, suitable for medium-sized applications requiring moderate performance.
40. **I am not able to connect to an EC2 instance. What are the issues you troubleshoot?**
    * Troubleshoot by checking security group rules, network ACLs, instance status, SSH key pair configuration, and network connectivity. Verify that the instance is running and the correct port (e.g., 22 for SSH) is open.
41. **What are the packages you install in your VM that are updated?**
    * Packages such as system updates, security patches, and application dependencies are regularly updated. This includes OS-level updates and any specific software used in the application stack.
42. **I need 2 JAVA versions which run parallel, so how can we do this on Windows?**
    * On Windows, you can install multiple versions of Java and manage them using environment variables. Configure the JAVA\_HOME variable for each application or use tools like SDKMAN! for managing multiple Java versions.
43. **How will you take data backup and which data will you take backup of in your organization?**
    * Data backup involves creating snapshots of EBS volumes, using AWS Backup for automated backups, and storing backups in S3. Critical data such as databases, application data, and configuration files are typically backed up.
44. **Difference between spot and reserve instance.**
    * **Spot Instances:** Purchased at a lower price but can be terminated if AWS needs the capacity back.
    * **Reserved Instances:** Purchased with a long-term commitment (1 or 3 years) at a discounted rate compared to On-Demand pricing, with guaranteed availability.
45. **What are the types of root devices?**
    * **EBS Root Device:** Uses an EBS volume for the root disk, providing persistence and flexibility.
    * **Instance Store Root Device:** Uses instance store volumes for the root disk, which are temporary and data is lost when the instance is stopped or terminated.
46. **Before creating an EC2 instance, I need to run a script. So where can I run my script?**
    * You can use **User Data** scripts to automate the initialization of your instance. These scripts run when the instance is first launched.
47. **What is Tenancy in EC2?**
    * Tenancy refers to the type of hardware your instance runs on. Options include **Shared Tenancy** (default, where instances share hardware) and **Dedicated Tenancy** (where instances run on dedicated hardware).
48. **What is T2 or what does it mean? What is its CPU and RAM configuration?**
    * T2 is a burstable performance instance type. **T2.micro** has 1 vCPU and 1 GB RAM, providing a baseline level of CPU performance with the ability to burst when needed.
49. **My keypair is lost, so is it possible to launch that instance?**
    * If you lose the key pair, you cannot directly access the instance. You can create a new key pair, stop the instance, detach the root volume, attach it to another instance, and modify the authorized\_keys file to regain access.
50. **Can you explain what is the keypair?**
    * A key pair consists of a public key and a private key. The public key is placed on the instance, and the private key is used to securely connect to the instance via SSH.
51. **What is the use of keypairs?**
    * Key pairs are used for secure access to EC2 instances. The private key is used to authenticate and establish a secure SSH connection to the instance.
52. **How to connect different machines if one is a created instance and one is a Linux machine?**
    * Use SSH to connect between instances. Ensure both instances are in the same network or VPC, and configure security groups and network ACLs to allow communication.
53. **What is AMI? How to create it?**
    * An AMI (Amazon Machine Image) is a pre-configured template for launching EC2 instances. You can create an AMI by selecting an existing instance, configuring it as needed, and then creating an image from the instance.
54. **What is the difference between marketplace and Community AMI?**
    * **Marketplace AMIs:** Provided by third-party vendors and may come with licensing fees.
    * **Community AMIs:** Free to use and provided by the AWS community or AWS itself.
55. **How can I check how many AMIs I have shared?**
    * You can check shared AMIs through the EC2 Management Console under the "AMIs" section, or use the AWS CLI command describe-images to list AMIs and their permissions.
56. **What does AMI include?**
    * An AMI includes an operating system, application server, applications, and related configurations necessary to launch a fully operational instance.
57. **Can I launch multiple instances using an AMI?**
    * Yes, you can launch multiple instances using the same AMI. This allows you to deploy multiple instances with identical configurations quickly.
58. **Can we share an AMI to a cross-region?**
    * Yes, you can share an AMI across regions by copying it to the desired region and then sharing it from there.
59. **How to get VM backup?**
    * VM backups can be created using snapshots of EBS volumes, AWS Backup, or third-party backup solutions. Ensure backups are regularly scheduled and stored in a secure location.
60. **If you have an EC2 instance and the instance key is lost, how do you log in to that instance without changing the key?**
    * If the key pair is lost, you need to create a new key pair, stop the instance, detach the root volume, attach it to another instance, and then modify the authorized\_keys file to regain access.

#################################

**EBS, EFS & Other storages**

#################################

Here are the answers to your questions:

1. **How can you connect an EBS volume to multiple instances?**
   * **EBS volumes cannot be attached to multiple instances simultaneously.** However, you can attach an EBS volume to one instance, and then detach it and attach it to another instance. For shared storage across multiple instances, use Amazon EFS or Amazon FSx for Windows File Server.
2. **What is EBS? Types of EBS?**
   * **Amazon EBS (Elastic Block Store)** is a scalable, high-performance block storage service designed for use with Amazon EC2. It provides persistent storage that can be used as the root device or as additional storage for your instances.
   * **Types of EBS Volumes:**
     + **General Purpose SSD (gp3, gp2)**: Balanced price and performance for most workloads.
     + **Provisioned IOPS SSD (io2, io1)**: High performance for I/O-intensive applications.
     + **Throughput Optimized HDD (st1)**: Low-cost HDD volume designed for frequently accessed, throughput-intensive workloads.
     + **Cold HDD (sc1)**: Lowest-cost HDD volume designed for infrequently accessed data.
3. **Can you delete your own snapshot?**
   * **Yes, you can delete your own EBS snapshots.** This can be done through the AWS Management Console, AWS CLI, or SDKs.
4. **Which of the following services is a data storage system that also has REST API interface and uses secure HMAC-SHA1 authentication keys?**
   * **Amazon S3** is a data storage service that provides a REST API interface and uses HMAC-SHA1 for authentication.
5. **How do you reserve volume for a user?**
   * **In AWS, you don’t “reserve” volumes for specific users directly.** Instead, you can create and manage EBS volumes and attach them to EC2 instances as needed. For reserved instances, you reserve compute capacity, not storage.
6. **Use of snapshot.**
   * **Snapshots** are used for backup, disaster recovery, and data migration. They provide point-in-time backups of EBS volumes and are incremental, which means only changes since the last snapshot are stored.
7. **How did you resolve the issue if the disk space is full on the server?**
   * **To resolve disk space issues:**
     + **Clean up unnecessary files and logs.**
     + **Resize the EBS volume (if using EBS).**
     + **Add additional storage.**
     + **Move data to S3 or another storage solution.**
     + **Implement data lifecycle policies or automated cleanup scripts.**
8. **How would you handle a situation where the relational database engine crashes often whenever the traffic to your RDS instances increases, given that the replica of RDS instance is not promoted as the master instance?**
   * **Consider the following actions:**
     + **Scale the instance size** to handle increased traffic.
     + **Enable read replicas** to offload read traffic.
     + **Optimize database queries** and indexes.
     + **Implement automated backups** and **monitor performance** using CloudWatch.
9. **What do you understand by volume & snapshot in AWS?**
   * **Volume**: A block storage device you attach to an EC2 instance, such as an EBS volume.
   * **Snapshot**: A point-in-time backup of an EBS volume. Snapshots are incremental and can be used to create new volumes or restore data.
10. **The data on the root volumes of store-backed and EBS-backed instances get deleted by default when they are terminated. If you want to prevent that from happening, which instance would you use?**
    * **To prevent data deletion, use an EBS-backed instance** and configure the root volume's delete-on-termination attribute to false.
11. **Is it possible to switch from an Instance-backed root volume to an EBS-backed root volume at any time?**
    * **Yes, you can switch from an instance-store-backed root volume to an EBS-backed root volume** by creating an AMI from the instance, then launching a new instance from that AMI with EBS as the root volume.
12. **What are the different types of EBS available?**
    * **Types of EBS volumes include:**
      + **General Purpose SSD (gp3, gp2)**
      + **Provisioned IOPS SSD (io2, io1)**
      + **Throughput Optimized HDD (st1)**
      + **Cold HDD (sc1)**
13. **In which situation would you select provisioned IOPS over Standard RDS storage?**
    * **Select Provisioned IOPS (io1/io2) when** you need high performance and consistent IOPS for I/O-intensive workloads, such as high-performance databases.
14. **Types of volumes in your organization?**
    * **Typical volume types might include:**
      + **General Purpose SSD (gp3, gp2)**
      + **Provisioned IOPS SSD (io2, io1)**
      + **Throughput Optimized HDD (st1)**
      + **Cold HDD (sc1)**
15. **How will you encrypt the EBS volume and EC2?**
    * **To encrypt an EBS volume**, use the AWS Management Console, CLI, or SDKs to enable encryption. For EC2 instances, ensure that the EBS volumes are encrypted at launch. Use the AWS Key Management Service (KMS) to manage encryption keys.
16. **In AWS, three different storage services are available, such as EFS, S3, and EBS. When should I use Amazon EFS vs. Amazon S3 vs. Amazon Elastic Block Store (EBS)?**
    * **Amazon EFS**: For shared file storage that can be accessed by multiple instances simultaneously.
    * **Amazon S3**: For scalable object storage with easy web access, ideal for backups, data archiving, and static content.
    * **Amazon EBS**: For block storage that attaches directly to EC2 instances, suitable for applications requiring high performance or low-latency access.
17. **How can we attach a snapshot in a different region?**
    * **Copy the snapshot** to the target region using the AWS Management Console, CLI, or SDKs. Once copied, you can create a volume from the snapshot in the new region.
18. **I want to migrate a volume to another account. Is it possible?**
    * **Yes, you can migrate an EBS volume to another account** by creating a snapshot of the volume, sharing the snapshot with the target account, and then creating a volume from the snapshot in the target account.
19. **The data on the root volumes of store-backed and EBS-backed instances get deleted by default when they are terminated. If you want to prevent that from happening, which instance would you use? And ensure if the EC2 instance is restarted, the data or configuration in the EC2 instance should not be lost.**
    * **Use an EBS-backed instance** and ensure the root volume’s delete-on-termination attribute is set to false. For persistence across reboots, use additional EBS volumes for important data.
20. **Is it possible to decrease the size of a volume?**
    * **No, you cannot directly decrease the size of an EBS volume.** To reduce the size, you would need to create a new, smaller volume and migrate the data.
21. **How can you save the data on root volume on an EBS-backed machine?**
    * **To save data on an EBS-backed instance,** ensure regular snapshots are taken. You can also move important data to separate EBS volumes or use automated backups.
22. **While creating a volume, is encryption possible? Share encryption name.**
    * **Yes, encryption is possible while creating an EBS volume.** Use AWS Key Management Service (KMS) for encryption, specifying an encryption key from KMS.
23. **Is the volume attached or not? How will you check?**
    * **To check if a volume is attached,** use the AWS Management Console, AWS CLI (describe-volumes), or SDKs to list volumes and their attachment status.
24. **Which protocol did you use for EBS?**
    * **Amazon EBS uses the iSCSI protocol** for communication between the volume and EC2 instances.
25. **What do you mean by incremental snapshot?**
    * **An incremental snapshot** captures only the changes made since the last snapshot, reducing storage costs and time needed for backup operations.
26. **How to delete EBS snapshots?**
    * **To delete EBS snapshots,** use the AWS Management Console, AWS CLI (delete-snapshot), or SDKs to remove the snapshot.
27. **Can we auto-delete snapshots?**
    * **Auto-deletion of snapshots is not directly supported.** However, you can use AWS Backup or implement lifecycle policies to automate snapshot management.
28. **A company currently operates a web application backed by an Amazon RDS MySQL database. It has automated backups that are run daily and are not encrypted. A security audit requires future backups to be encrypted and unencrypted backups to be destroyed. The company will make at least one encrypted backup before destroying the old backups. What should be done to enable encryption for future backups?**
    * **Enable encryption for automated backups** by modifying the RDS instance to use an encrypted storage option. After ensuring that encrypted backups are running, delete the unencrypted backups.
29. **Explain some EBS Volume types and which one is fast?**
    * **EBS Volume Types:**
      + **General Purpose SSD (gp3, gp2)**: Good balance of price and performance.
      + **Provisioned IOPS SSD (io2, io1)**: High performance for I/O-intensive workloads; generally the fastest option.
      + **Throughput Optimized HDD (st1)**: Cost-effective for throughput-intensive workloads.
      + **Cold HDD (sc1)**: Lowest-cost option for infrequent access.
    * **Fastest Volume Type**: **Provisioned IOPS SSD (io2)** is the fastest.
30. **Storing the size of EBS? Its pricing structure?**
    * **EBS volume pricing** is based on the volume type, size, and IOPS (for provisioned IOPS). You are billed for the provisioned size of the volume and the amount of IOPS for provisioned IOPS volumes. Data transfer and snapshot costs are additional.
31. **Linux command to see attached EBS volume?**
    * **Use the command** lsblk to list block devices and check attached EBS volumes.
32. **EBS and instance store volume difference.**
    * **EBS (Elastic Block Store)**: Persistent storage that survives instance termination and can be detached and reattached.
    * **Instance Store**: Temporary storage that is physically attached to the host machine and data is lost when the instance is stopped or terminated.
33. **Can EBS support a file system interface?**
    * **EBS itself does not provide a file system interface**, but you can format and mount an EBS volume to use as a file system on an EC2 instance.
34. **Is EBS faster than S3? Which is cheaper, EBS or S3?**
    * **EBS** is faster for block storage and low-latency access compared to S3, which is object storage designed for durability and scalability. **EBS** is generally more expensive than S3.
35. **Which storage do you prefer for data storage? Why?**
    * **The choice depends on the use case:**
      + **EBS** for block storage with low-latency access.
      + **S3** for scalable object storage and long-term data archiving.
      + **EFS** for shared file storage across multiple instances.
36. **What is the difference between EBS and EFS?**
    * **EBS (Elastic Block Store)**: Block storage used as an individual volume attached to EC2 instances, offering low-latency and high performance.
    * **EFS (Elastic File System)**: File storage that can be accessed concurrently by multiple EC2 instances, suitable for shared storage needs.
37. **What is the difference between EFS and S3?**
    * **EFS (Elastic File System)**: Provides a file system interface and is ideal for shared access by multiple instances.
    * **S3 (Simple Storage Service)**: Object storage with a RESTful API, suitable for scalable, long-term storage and data backups.
38. **How is a 5GB file stored in S3 and EFS? What are the underlying protocols?**
    * **In S3**, the 5GB file is stored as an object with metadata and is accessed via HTTP/HTTPS using RESTful API.
    * **In EFS**, the file is stored in a file system and accessed over NFS (Network File System) protocol.
39. **Cost-wise, compare which storage is cheapest and which one is the costliest? EFS is costly.**
    * **Cheapest**: **Amazon S3** for object storage.
    * **Costliest**: **Amazon EFS** for file storage due to its shared file system capabilities.
40. **How do you take a backup of EFS?**
    * **Back up EFS** using AWS Backup service, or manually copy data to another EFS file system or to S3.
41. **Difference between EBS vs EFS vs S3.**
    * **EBS**: Block storage for single instance use, with low-latency performance.
    * **EFS**: Shared file storage accessible by multiple instances, using NFS.
    * **S3**: Object storage for scalable, long-term storage with RESTful access.
42. **Which volume type is not bootable?**
    * **Cold HDD (sc1)** is typically used for data that is infrequently accessed and is not suitable for boot volumes.
43. **Is EFS similar to Windows FSX?**
    * **Amazon FSx for Windows File Server** is similar to EFS in that it provides file storage, but FSx is specifically designed for Windows-based workloads and uses the SMB protocol, while EFS uses NFS.
44. **What is the positive use of EFS?**
    * **EFS** is ideal for applications that require a shared file system across multiple EC2 instances, such as content management systems, home directories, and application data.
45. **If your storage is full, how will you increase it?**
    * **For EBS volumes,** increase size by modifying the volume in the AWS Management Console or CLI and then resizing the file system. For S3 or EFS, you can add more storage without additional configuration.

######################################################

**Devops, Project & Introductory questions:**

######################################################

 **Challenging task you've faced.**

* One challenging task I faced was migrating a large-scale application from on-premises infrastructure to the cloud. The project involved redesigning the architecture to ensure scalability and high availability, managing data migration, and minimizing downtime. I tackled this challenge by creating a detailed migration plan, conducting thorough testing, and coordinating with multiple teams to address potential issues proactively.

 **What is Agile?**

* Agile is a project management and software development methodology that emphasizes iterative development, collaboration, and flexibility. It focuses on delivering small, incremental improvements to the product, allowing teams to respond quickly to changes in requirements and feedback.

 **What does your application consist of in your project?**

* My application typically consists of a front-end interface, a back-end API, and a database. It may also include various microservices, integrations with third-party APIs, and a deployment pipeline for continuous integration and continuous deployment (CI/CD).

 **What is a declarative programming language?**

* A declarative programming language focuses on describing what the program should accomplish rather than how it should achieve it. In declarative programming, you specify the desired outcomes, and the underlying system handles the implementation details. Examples include SQL for database queries and Terraform for infrastructure as code.

 **Tell us about a past project and the tools you worked with.**

* In a recent project, I worked on implementing a CI/CD pipeline for a microservices-based application. We used tools like Jenkins for automation, Docker for containerization, Kubernetes for orchestration, and AWS for cloud infrastructure. The project involved configuring pipelines for automated testing, building, and deployment.

 **What are your daily tasks, and what are you mainly assigned with?**

* My daily tasks include monitoring system performance, troubleshooting issues, managing deployments, and maintaining infrastructure. I am primarily assigned to ensure that our applications run smoothly, handle incidents, and work on optimizing our CI/CD pipelines.

 **What security measures did you take while developing the infrastructure?**

* Security measures include implementing firewalls and access controls, using encryption for data at rest and in transit, conducting regular vulnerability scans, and following best practices for identity and access management (IAM). Additionally, I ensure that all software and dependencies are up-to-date with security patches.

 **Tell me about your infrastructure.**

* Our infrastructure is built on a cloud platform with a mix of virtual machines and containerized services. We use Kubernetes for orchestration, AWS for cloud services, and Terraform for infrastructure management. Our setup includes load balancers, auto-scaling groups, and monitoring tools to ensure high availability and performance.

 **How are you optimizing the cost of the project?**

* Cost optimization strategies include using reserved instances for predictable workloads, implementing auto-scaling to handle variable traffic, optimizing resource allocation, and regularly reviewing and adjusting our cloud services usage. We also use cost monitoring tools to track and analyze spending.

 **Will you introduce yourself?**

* I’m [Your Name], a DevOps Engineer with experience in managing and automating infrastructure, deploying applications, and optimizing performance. I have a background in cloud technologies, CI/CD pipelines, and monitoring systems. I enjoy solving complex problems and improving system efficiency.

 **What are the roles and responsibilities of a DevOps Engineer?**

* A DevOps Engineer is responsible for managing the development and operations lifecycle, including automating deployment processes, monitoring system performance, ensuring high availability, managing infrastructure, and collaborating with development and operations teams to streamline workflows.

 **Why did you choose the Cloud/DevOps field?**

* I chose the Cloud/DevOps field because it combines my interests in technology and problem-solving with the opportunity to work on cutting-edge tools and practices. The dynamic nature of the field and its impact on improving software delivery and operational efficiency are what drive my passion for it.

 **How do you give code to the testing team for testing or for dev QA and production teams?**

* I use a CI/CD pipeline to automate the process of code delivery. Code is pushed to a version control system, which triggers automated build and test processes. Once the code passes all tests, it is reviewed and merged into the main branch, from where it is deployed to staging and production environments based on the deployment strategy.

 **Why do you want to shift as a DevOps engineer from cloud?**

* Transitioning to a DevOps role from a cloud-focused position allows me to leverage my cloud expertise while expanding my skill set to include more aspects of the development and operations lifecycle. It offers an opportunity to work on end-to-end solutions and drive improvements across the entire software delivery process.

 **Role in your project?**

* My role in the project involves managing the CI/CD pipeline, automating infrastructure provisioning, monitoring system performance, and addressing any issues that arise. I also collaborate with development and operations teams to ensure smooth deployments and optimal system performance.

 **Can you explain what kind of architecture in your company you follow?**

* Our company follows a microservices architecture, where applications are broken down into smaller, independent services that communicate over APIs. This architecture allows for greater scalability, flexibility, and ease of maintenance. We use Kubernetes for container orchestration and deploy services on cloud infrastructure.

 **What are day-to-day activities in your company? OR What skill set do you use daily?**

* Day-to-day activities include monitoring system health, managing deployments, automating repetitive tasks, and collaborating with teams to resolve issues. Skills used daily include scripting, working with CI/CD tools, containerization, cloud services management, and troubleshooting.

 **Which technologies have you worked on?**

* I have worked with a variety of technologies, including Docker for containerization, Kubernetes for orchestration, AWS for cloud services, Terraform for infrastructure as code, Jenkins for CI/CD, and various monitoring and logging tools such as Prometheus and ELK Stack.

 **What are you doing in your project?**

* In my current project, I am responsible for maintaining and optimizing our CI/CD pipeline, managing cloud infrastructure, and ensuring that applications are deployed efficiently and securely. I also work on automating processes and troubleshooting issues as they arise.

 **How many team members do you have, and what are the configurations in your company?**

* My team consists of [Number] members, including developers, QA engineers, and operations staff. Our configurations include a mix of cloud-based and on-premises infrastructure, with a focus on containerized applications and microservices.

 **How will you manage your client requirement?**

* I manage client requirements by gathering detailed specifications, setting clear expectations, and maintaining open communication throughout the project. I prioritize requirements based on impact and feasibility, and regularly update clients on progress and any potential issues.

 **Which troubleshoot have you done?**

* I have troubleshooted various issues, including network connectivity problems, performance bottlenecks, deployment failures, and application errors. My approach involves identifying the root cause, analyzing logs, and implementing fixes or workarounds as needed.

 **Can you tell me any 5 Linux commands with their use case?**

* ls - Lists directory contents.
* grep - Searches for specific patterns in files.
* ps - Displays information about active processes.
* chmod - Changes file permissions.
* df - Shows disk space usage.

 **Which web server do you use?**

* I use various web servers depending on the project requirements, including Apache HTTP Server and Nginx. Each has its strengths, with Nginx often used for its high performance and ability to handle a large number of concurrent connections.

 **In which language is coding done?**

* Coding in my projects is done in several languages, including Python for scripting, JavaScript for front-end development, and Go or Java for back-end services. The choice of language depends on the specific requirements of the application and team preferences.

 **Do you set up infra for any project?**

* Yes, I set up infrastructure for various projects using tools like Terraform and AWS CloudFormation. This involves configuring resources such as virtual machines, databases, and networking components to meet the project’s needs.

 **How many versions have you maintained parallelly?**

* I have maintained multiple versions of applications, typically including development, staging, and production environments. This allows for testing new features and bug fixes before deploying them to production.

 **How do you deploy code?**

* Code is deployed using a CI/CD pipeline, which automates the build, test, and deployment processes. This involves pushing code to a version control system, running automated tests, and deploying the code to various environments based on the pipeline configuration.

 **What documentation have you done in DevOps?**

* I have created documentation for infrastructure setup, CI/CD pipelines, deployment processes, troubleshooting guides, and user manuals. This documentation helps ensure consistency and provides a reference for current and future team members.

 **What kind of backup are you taking? Is it automated or manual?**

* We use automated backup solutions for critical data, including regular snapshots and backups of databases and configuration files. Automated backups are scheduled to run at specified intervals to ensure data is consistently protected.

 **Which bugs have you resolved in your 2-3 years of experience? How do you get notification of the bugs?**

* I have resolved bugs related to performance issues, deployment errors, and security vulnerabilities. I receive notifications of bugs through monitoring tools, issue tracking systems, and alerts configured for specific error conditions.

 **What are the OS you launch?**

* I work with various operating systems, including Linux distributions (e.g., Ubuntu, CentOS) and Windows Server. The choice of OS depends on the requirements of the application and the infrastructure setup.

 **Explain your project architecture.**

* My project architecture involves a microservices approach, with each service handling a specific function and communicating over APIs. We use containerization with Docker, orchestration with Kubernetes, and cloud services for hosting and scaling.

 **Which DevOps tools have you worked with?**

* I have worked with a range of DevOps tools, including Jenkins for CI/CD, Docker for containerization, Kubernetes for orchestration, Terraform for infrastructure management, and monitoring tools like Prometheus and Grafana.

 **What is SDLC?**

* SDLC (Software Development Life Cycle) is a process that outlines the stages of software development from initial planning through to deployment and maintenance. It includes phases such as requirements gathering, design, development, testing, deployment, and maintenance.

 **Before DevOps and After DevOps?**

* Before DevOps, development and operations teams often worked in silos, leading to slower delivery and more issues in production. After adopting DevOps practices, the focus shifted to collaboration, automation, and continuous improvement, resulting in faster delivery, improved quality, and more efficient operations.

 **What is the difference between WAR, EAR, JAR files?**

* WAR (Web Application Archive) files are used for packaging web applications, EAR (Enterprise Archive) files package Java EE applications that include web, EJB, and other components, and JAR (Java Archive) files are used for packaging Java class files and associated metadata into a single file.

 **What are your daily responsibilities?**

* My daily responsibilities include managing deployments, monitoring system performance, automating tasks, troubleshooting issues, and collaborating with teams to ensure smooth operation and continuous improvement of our systems.

 **What is the most challenging job you have faced, and how did you accomplish it?**

* The most challenging job was migrating a legacy application to a microservices architecture. I accomplished it by carefully planning the migration, breaking down the application into manageable services, testing each service individually, and ensuring minimal disruption to users.

 **How would you handle a situation when you have no one to help with a task?**

* I would approach the task methodically by breaking it down into smaller steps, leveraging available resources such as documentation and online forums, and applying my problem-solving skills. If necessary, I would escalate the issue to a higher authority for guidance.

 **Have you worked on documentation for user guides and troubleshooting procedures?**

* Yes, I have worked on documentation for user guides, which help users understand how to interact with applications, and troubleshooting procedures, which provide guidance on resolving common issues and errors.

 **How many members do you have in your team?**

* My team consists of [Number] members, including various roles such as developers, QA engineers, and operations staff, all working together to ensure the successful delivery and maintenance of our projects.

 **Explain the architecture you are using for your project in detail.**

* Our project architecture follows a microservices model, with services deployed in containers and managed by Kubernetes. We use a combination of cloud-based resources for scalability and high availability, and our architecture includes load balancers, service discovery, and automated scaling based on demand.