1. **Will you introduce yourself?**
   * My name is [Your Name], and I have been working in the IT industry for [X] years with a focus on cloud computing and DevOps. I have a strong background in [your background, e.g., software development, system administration, etc.], and I am passionate about automating processes, improving CI/CD pipelines, and ensuring smooth software delivery.
2. **What are the roles and responsibilities of a DevOps Engineer?**
   * A DevOps Engineer is responsible for automating and streamlining operations and processes, building and maintaining tools for deployment, monitoring, and operations. They work closely with software developers to ensure that development follows established processes and works as intended in the various environments.
3. **Why did you choose the Cloud/DevOps field?**
   * I chose the Cloud/DevOps field because of my interest in automation, cloud technologies, and continuous integration/continuous deployment (CI/CD). I enjoy solving complex problems and making systems more efficient and scalable.
4. **Can you explain what kind of architecture your company follows?**
   * My company follows a microservices architecture, where applications are broken down into smaller, independent services that communicate through APIs. We deploy these services using containerization technologies like Docker and orchestration tools like Kubernetes.
5. **What are your day-to-day activities in your company? OR What skill set do you use daily?**
   * My daily activities include managing CI/CD pipelines, monitoring system performance, troubleshooting issues, collaborating with development teams, and improving automation scripts. I use skills such as scripting (Bash, Python), containerization (Docker), orchestration (Kubernetes), and cloud services (AWS, Azure).
6. **Which technologies have you worked on?**
   * I have worked on technologies such as Docker, Kubernetes, Jenkins, Git, Ansible, Terraform, AWS, Azure, and various monitoring tools like Prometheus and Grafana.
7. **What are you doing in your project?**
   * In my current project, I am responsible for automating the deployment process, ensuring high availability of applications, monitoring system health, and maintaining infrastructure as code using Terraform.
8. **Why do you want to shift as a DevOps Engineer from cloud?**
   * I see DevOps as an extension of my cloud expertise, allowing me to further automate and optimize the deployment and operation of cloud resources, thus improving efficiency and reliability.
9. **What are different cloud service providers?**
   * The major cloud service providers are Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform (GCP), IBM Cloud, and Oracle Cloud.
10. **Why did you move to cloud from competitive exams?**
    * Moving to the cloud allowed me to apply my skills in a dynamic and growing field, offering more opportunities for innovation and impact compared to the structured and theoretical nature of competitive exams.
11. **How many team members do you have, and what are the configurations in your company?**
    * My team consists of [X] members, including developers, QA engineers, and operations staff. We follow an agile methodology with cross-functional teams responsible for different parts of the application lifecycle.
12. **How will you manage your client requirements?**
    * I manage client requirements by maintaining clear communication, setting realistic expectations, using project management tools to track progress, and ensuring that all changes go through a thorough testing and review process before deployment.
13. **Which troubleshoot did you do?**
    * I have troubleshooted various issues such as failed deployments, performance bottlenecks, system outages, and configuration errors. For example, I resolved a critical issue where an application was failing due to incorrect environment variable settings in the deployment script.
14. **Can you tell me any 5 Linux commands with their use case?**
    * ls: List directory contents.
    * grep: Search for patterns within files.
    * tail -f: Display the last part of a file and follow it for new content.
    * chmod: Change file permissions.
    * ps aux: Display information about running processes.
15. **Which languages are used in the backend and frontend for your project?**
    * The backend is developed using [e.g., Java, Python, Node.js], and the frontend uses [e.g., React, Angular, Vue.js].
16. **Explain your project architecture?**
    * The project uses a microservices architecture with services running in Docker containers, managed by Kubernetes. The backend services communicate through REST APIs, and the frontend is a single-page application (SPA). CI/CD pipelines are set up using Jenkins and deployed on AWS.
17. **Which DevOps tools have you worked with?**
    * I have worked with Jenkins, Git, Docker, Kubernetes, Ansible, Terraform, Prometheus, Grafana, and various cloud services like AWS and Azure.
18. **Role in your project?**
    * My role is to design and implement CI/CD pipelines, manage infrastructure as code, monitor system performance, and collaborate with development teams to ensure smooth deployments.
19. **Which webserver do you use?**
    * We use Nginx as our web server to handle HTTP requests and serve static content.
20. **In which language is coding done?**
    * Coding is done in languages such as Python, Java, JavaScript, and TypeScript.
21. **Do you set up infrastructure for any project?**
    * Yes, I set up infrastructure using tools like Terraform and Ansible to ensure consistency and scalability across environments.
22. **How many versions have you maintained in parallel?**
    * We typically maintain multiple versions in parallel, such as the current stable release, the development version, and hotfix branches.
23. **How do you deploy code?**
    * Code is deployed using CI/CD pipelines with Jenkins. The pipeline includes steps for building, testing, and deploying the application to staging and production environments.
24. **What documentation have you done in DevOps?**
    * I have documented CI/CD pipeline configurations, infrastructure as code scripts, deployment procedures, troubleshooting guides, and best practices for developers.
25. **What kind of backup are you taking? Is it automated or manual?**
    * We take automated backups using cloud-native solutions like AWS Backup and custom scripts to ensure data is regularly backed up and can be restored quickly in case of failure.
26. **Which bugs have you resolved in your 2-3 years of experience? How will you get notification of the bugs?**
    * I have resolved bugs related to deployment failures, performance issues, and security vulnerabilities. Notifications for bugs come through monitoring tools like Prometheus, alerting systems like PagerDuty, and issue tracking systems like Jira.
27. **What are the OS you launch?**
    * We primarily launch Linux-based operating systems such as Ubuntu and CentOS for our servers and containers.

-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

SDLC:

 **What is SDLC?**

* The Software Development Life Cycle (SDLC) is a process used by the software industry to design, develop, and test high-quality software. The SDLC aims to produce software that meets or exceeds customer expectations, reaches completion within times and cost estimates. The stages of SDLC typically include:
  1. **Planning:** Define project goals and plan the project schedule.
  2. **Requirement Analysis:** Gather and analyze business and user requirements.
  3. **Design:** Create the architecture and design the system components.
  4. **Implementation (or Coding):** Write and compile the code for the software.
  5. **Testing:** Test the software to find and fix bugs.
  6. **Deployment:** Deploy the software to a production environment.
  7. **Maintenance:** Provide ongoing support and updates after deployment.

 **Before DevOps and After DevOps?**

* **Before DevOps:**
  + Development and Operations teams worked in silos.
  + Longer development cycles and slower deployment times.
  + Manual testing and deployment processes prone to errors.
  + Delays in detecting and fixing bugs.
  + Lack of collaboration and communication between teams.
* **After DevOps:**
  + Development and Operations teams work together in a collaborative manner.
  + Continuous integration and continuous deployment (CI/CD) pipelines enable faster development cycles and more frequent releases.
  + Automated testing and deployment processes increase reliability and reduce errors.
  + Quick detection and resolution of bugs through continuous monitoring and feedback loops.
  + Improved collaboration, communication, and overall efficiency within teams.

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

* **JAR (Java ARchive) File:**
  + A JAR file is used to package Java class files and associated resources (text, images, etc.) into a single file for distribution.
  + It is typically used for libraries, executables, and components that can be used by Java applications.
  + JAR files can be executed directly if they include a manifest file specifying the main class.
* **WAR (Web Application ARchive) File:**
  + A WAR file is used to package a web application that can be deployed on a servlet/JSP container or web server.
  + It includes Java class files, JSP files, HTML files, JavaScript files, and other resources required for the web application.
  + The WAR file has a specific directory structure that includes directories like WEB-INF and META-INF.
* **EAR (Enterprise ARchive) File:**
  + An EAR file is used to package an enterprise application that can be deployed on an application server.
  + It can include JAR files, WAR files, and other resource files needed for the enterprise application.
  + EAR files are used for large-scale applications that might include multiple modules, such as web modules, EJB modules, and client modules.

CLOUD:

 **What is cloud computing?**

* Cloud computing is the delivery of computing services—including servers, storage, databases, networking, software, analytics, and intelligence—over the Internet (“the cloud”) to offer faster innovation, flexible resources, and economies of scale. Users can access these services on demand, typically paying only for what they use, which helps reduce operating costs, run infrastructure more efficiently, and scale as business needs change.

 **Types of clouds and their examples:**

* **Public Cloud:**
  + A public cloud is a type of cloud computing where services are delivered over the public internet and shared across multiple organizations. These clouds are owned and operated by third-party cloud service providers, which deliver their computing resources like servers and storage over the internet. Examples include:
    - **Amazon Web Services (AWS)**
    - **Microsoft Azure**
    - **Google Cloud Platform (GCP)**
* **Private Cloud:**
  + A private cloud is a cloud computing environment exclusively used by one organization. It can be physically located at the organization’s on-site data center or hosted by a third-party service provider. The services and infrastructure are maintained on a private network. Examples include:
    - **VMware Cloud**
    - **OpenStack**
    - **IBM Cloud Private**
* **Hybrid Cloud:**
  + A hybrid cloud is a computing environment that combines a public cloud and a private cloud by allowing data and applications to be shared between them. By allowing data and applications to move between private and public clouds, a hybrid cloud gives businesses greater flexibility and more deployment options. Examples include:
    - **Microsoft Azure (with Azure Stack)**
    - **Amazon Web Services (with VMware Cloud on AWS)**
    - **Google Cloud (with Anthos)**
* **Multi-Cloud:**
  + A multi-cloud approach involves using multiple cloud computing services from different providers. This strategy avoids reliance on a single cloud provider and can enhance redundancy and flexibility. Examples include using services from both:
    - **AWS and Azure**
    - **Azure and Google Cloud**
    - **AWS and IBM Cloud**

AWS:

 **Why did you choose AWS cloud?**

* I chose AWS cloud because it is one of the most comprehensive and widely adopted cloud platforms, offering a vast array of services and features. AWS provides robust infrastructure, scalability, reliability, and security, which are critical for modern applications. Additionally, AWS has a strong community and extensive documentation, making it easier to learn and implement solutions.

 **Difference between zone and regions:**

* **Regions:**
  + A region is a geographical area that contains multiple, isolated locations known as Availability Zones. Each region is completely independent and isolated from other regions to provide fault tolerance and stability.
* **Availability Zones (AZs):**
  + An Availability Zone is a data center within a region. Each region typically has multiple AZs to ensure high availability and fault tolerance. AZs within a region are connected through low-latency links, allowing you to replicate data and distribute applications across multiple AZs for redundancy.

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

* Some AWS services are region-specific, meaning they are available only in certain regions. Examples include:
  + **Amazon EC2 (Elastic Compute Cloud)**
  + **Amazon RDS (Relational Database Service)**
  + **Amazon VPC (Virtual Private Cloud)**
  + **AWS Lambda**

 **Non-region specific services:**

* These services are globally available and not tied to a specific region. Examples include:
  + **Amazon Route 53 (DNS and Domain Name Registration)**
  + **AWS IAM (Identity and Access Management)**
  + **AWS CloudFront (Content Delivery Network)**
  + **Amazon S3 (Simple Storage Service)** (though S3 data is stored in regions, the service itself is globally accessible)

 **What do you actually work on in AWS, and which services do you use?**

* I work on various AWS services depending on the project requirements. Some of the services I frequently use include:
  + **Amazon EC2:** For computing resources.
  + **Amazon S3:** For object storage.
  + **Amazon RDS:** For managed relational databases.
  + **AWS Lambda:** For serverless computing.
  + **Amazon VPC:** For network isolation.
  + **Amazon CloudWatch:** For monitoring and logging.
  + **AWS IAM:** For access control and security.
  + **AWS CloudFormation/Terraform:** For infrastructure as code.

 **What is AWS?**

* AWS (Amazon Web Services) is a subsidiary of Amazon providing on-demand cloud computing platforms and APIs to individuals, companies, and governments on a metered pay-as-you-go basis. It offers a wide range of services including computing power, storage, databases, machine learning, and more.

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

* AWS can be used across various platforms and environments, including:
  + **Web applications:** Hosting websites and web applications.
  + **Mobile applications:** Backend services for mobile apps.
  + **Enterprise applications:** Running business-critical applications.
  + **Big data analytics:** Processing and analyzing large datasets.
  + **Machine learning:** Training and deploying machine learning models.
  + **IoT:** Connecting and managing IoT devices.
  + **Game development:** Backend services for online games.

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

* I manage infrastructure using tools like:
  + **AWS CloudFormation:** For defining and provisioning infrastructure as code.
  + **Terraform:** An open-source infrastructure as code tool that allows you to build, change, and version infrastructure safely and efficiently.
  + **AWS Management Console:** For manual management and configuration.
  + **AWS CLI and SDKs:** For automating tasks and integrating AWS services into applications.

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

* **Region:**
  + A region is a geographically isolated area that consists of multiple Availability Zones. Each region operates independently to provide fault tolerance and stability.
* **Availability Zone (AZ):**
  + An Availability Zone is a discrete data center within a region. AZs are designed to be isolated from failures in other AZs, with low-latency connectivity between them. This allows you to architect applications to be highly available and fault-tolerant by distributing resources across multiple AZs within a region.

 **What is elastic computing?**

* Elastic computing refers to the ability to dynamically scale compute resources up or down based on demand. AWS Elastic Compute Cloud (EC2) provides this flexibility by allowing users to provision and terminate instances as needed, ensuring that resources match the current workload.

 **How many instances can I run?**

* By default, you can run up to 20 On-Demand instances per region. However, this limit can be increased by submitting a request to AWS support. The limit for other instance types like Spot and Reserved instances may vary.

 **What are the different EC2 instances based on their cost?**

* **On-Demand Instances:** Pay for compute capacity by the hour or second with no long-term commitments.
* **Reserved Instances:** Purchase instances at a significant discount for a one-year or three-year term.
* **Spot Instances:** Bid for unused EC2 capacity at potentially lower costs.
* **Savings Plans:** Flexible pricing model offering significant savings on EC2 usage, regardless of instance family, size, OS, or region.
* **Dedicated Hosts:** Physical servers dedicated for your use.

 **What type of instances you worked on? Tell me its configuration:**

* **T2.medium:** 2 vCPUs, 4 GiB RAM
* **M5.large:** 2 vCPUs, 8 GiB RAM
* **C5.xlarge:** 4 vCPUs, 8 GiB RAM

 **Types of instances? Different types of EC2 instances you have worked on:**

* **General Purpose:** T2, T3, M5
* **Compute Optimized:** C5, C5n
* **Memory Optimized:** R5
* **Storage Optimized:** I3
* **GPU Instances:** P3, G4

 **What is the difference between stopping and termination of instances?**

* **Stopping:** The instance shuts down and can be restarted later. The EBS volume remains attached, and you are not charged for instance usage but for EBS storage.
* **Termination:** The instance is shut down and cannot be restarted. The associated EBS volumes are deleted unless the "Delete on Termination" flag is disabled.

 **What is EC2? What does the Elastic word indicate in EC2?**

* EC2 (Elastic Compute Cloud) is a web service that provides resizable compute capacity in the cloud. The term "Elastic" indicates the ability to scale compute resources up or down as needed.

 **Can you change types of instance?**

* Yes, you can change the instance type by stopping the instance, modifying the instance type, and then starting the instance again.

 **Can you tell the frequency of launching EC2 in your project? Difference between t2.micro and t3.large? What is T4?**

* **Frequency:** Depends on the project needs, but typically instances are launched multiple times during development and testing phases.
* **t2.micro:** 1 vCPU, 1 GiB RAM
* **t3.large:** 2 vCPUs, 8 GiB RAM
* **T4:** As of my knowledge cutoff in 2021, AWS has not released a T4 instance family.

 **What will happen when you delete an EC2 instance with an attached volume and EIP?**

* The instance is terminated, the attached volume (if configured to delete on termination) is deleted, and the Elastic IP (EIP) is disassociated but not deleted; it remains in your account.

 **Which key is used to access the instance through PuTTY?**

* The private key file (with .pem extension) is used, which needs to be converted to .ppk format using PuTTYgen.

 **Suppose I do SSH enable and in the outbound SSH is not enabled. How do I access the EC2 instance?**

* Ensure that outbound rules in the security group allow SSH traffic (port 22). Also, ensure the local firewall or network ACLs are not blocking the SSH traffic.

 **Suppose we launch an EC2 instance due to some reason my server is shutdown and my IP change, so how can you overcome it?**

* Use an Elastic IP (EIP) which remains static even if you stop and start the instance.

 **Types of EC2 instances? Which are the cheapest?**

* **Types:** General Purpose, Compute Optimized, Memory Optimized, Storage Optimized, GPU Instances.
* **Cheapest:** T3 and T4g instances are typically the cheapest for general-purpose use.

 **Can I upgrade my EC2 instance after launching?**

* Yes, you can change the instance type by stopping the instance, modifying the instance type, and starting the instance again.

 **What are On-Demand and Spot instances?**

* **On-Demand Instances:** Pay for compute capacity by the hour/second with no long-term commitments.
* **Spot Instances:** Bid for unused EC2 capacity at potentially lower costs.

 **What is instance store volume exactly?**

* Instance store volumes are temporary block-level storage for your instance. Data on instance store volumes persists only during the lifetime of the instance.

 **How can we safeguard our EC2 instance?**

* Use security groups, network ACLs, IAM roles, encrypted EBS volumes, regular updates, monitoring, and logging.

 **What is the meaning of General Purpose instance?**

* General Purpose instances provide a balance of compute, memory, and networking resources and can be used for a variety of diverse workloads.

 **Different ways to launch EC2 instance or ways to launch instances (explain in brief):**

* **AWS Management Console:** A web-based interface.
* **AWS CLI:** Command-line interface to script actions.
* **AWS SDKs:** Integrate with applications using programming languages.
* **AWS CloudFormation:** Use templates to manage infrastructure as code.

 **On-Demand vs Spot instance? When to use Spot instances?**

* **On-Demand:** Pay for compute capacity by the hour/second with no long-term commitments.
* **Spot:** Bid for unused EC2 capacity at potentially lower costs.
* **Use Spot instances:** For fault-tolerant, flexible applications, like big data analysis, containerized workloads, CI/CD, web servers, etc.

 **Why do we use On-Demand instances?**

* On-Demand instances are used for their flexibility and lack of long-term commitment, making them suitable for unpredictable workloads.

 **EC2 is region-specific or zone-specific?**

* EC2 is region-specific. Instances are launched in specific regions, which can have multiple availability zones.

 **Without disk is it possible to launch EC2 instance?**

* No, an EC2 instance requires at least one root volume to launch.

 **How many EC2 instances can we launch, i.e., default numbers?**

* By default, you can run up to 20 On-Demand instances per region. This limit can be increased by submitting a request to AWS support.

 **T2.large and its specification:**

* **T2.large:** 2 vCPUs, 8 GiB RAM

 **Can you explain the step-by-step process to launch an EC2 instance?**

1. Open the **AWS Management Console**.
2. Navigate to **EC2 Dashboard**.
3. Click on **Launch Instance**.
4. Choose an **AMI (Amazon Machine Image)**.
5. Select an **instance type** (e.g., T2.micro).
6. Configure **instance details** (number of instances, network settings).
7. Add **storage** (EBS volumes).
8. Add **tags** (optional).
9. Configure **security group** (allow inbound/outbound traffic).
10. Review and **launch**.
11. Select or create a **key pair**.
12. Launch the instance.

 **CPU and RAM configuration of T2.large:**

* **T2.large:** 2 vCPUs, 8 GiB RAM

 **How many categories available in EC2 instance?**

* Categories include General Purpose, Compute Optimized, Memory Optimized, Storage Optimized, and GPU Instances.

 **How will you calculate the cost of EC2 instance and other services?**

* Use the **AWS Pricing Calculator** to estimate costs based on instance type, usage hours, storage, and additional services.

 **How many EC2 instances can we create in a VPC?**

* There is no specific limit for EC2 instances in a VPC; the limit is based on the region-specific instance limits.

 **How to connect different machines if one is a created instance and one is a Linux machine?**

* Use **SSH** to connect from the Linux machine to the EC2 instance using the private key associated with the instance.

 **Who will take the decision on which configuration is needed?**

* The decision is typically made by the **DevOps team**, **system architects**, or **project managers** based on the project requirements.

 **EC2 instance is region-specific or non-region specific?**

* EC2 instances are **region-specific**.

 **Which EC2 instance you launch? Why? Tell me the configuration:**

* I usually launch **T2.medium** for development and testing purposes because it offers a balance of cost and performance.

 **I am not able to connect to an EC2 instance. What are the issues you troubleshoot?**

* Check **security group** rules for inbound/outbound traffic.
* Verify **network ACLs**.
* Ensure the **instance is running**.
* Verify **private key** usage.
* Check **VPC and subnet** configurations.
* Ensure **SSH service** is running on the instance.

 **What are the packages you install in your VM which are updated?**

* Update and install packages like **Apache**, **Nginx**, **Java**, **Python**, **Node.js**, and **security patches** as needed.

 **I need two JAVA versions which run parallel, so how can we do this on Windows?**

* Install multiple versions of Java and use a **version manager** like **SDKMAN** or manually set the **JAVA\_HOME** and update the **PATH** variable accordingly.

 **How will you take data backup and which data will you take backup in your organization?**

* Use **AWS Backup**, **EBS snapshots**, and **S3** for data backup. Backup critical data like **databases**, **application data**, **config files**, and **logs**.

 **Difference between spot and reserved instances:**

* **Spot Instances:** Bid for unused EC2 capacity at lower costs, suitable for fault-tolerant workloads.
* **Reserved Instances:** Purchase at a discount for a one-year or three-year term, suitable for predictable workloads.

 **What are the types of root devices?**

* **EBS-backed:** The root device is an EBS volume.
* **Instance store-backed:** The root device is an instance store volume.

 **Before creating an EC2 instance, I need to run a script. So, where can I run my script?**

* Use **EC2 user data** to run initialization scripts when launching the instance.

 **What is Tenancy in EC2?**

* Tenancy defines the type of hardware dedicated to your instances. Options include **Shared**, **Dedicated Instances**, and **Dedicated Hosts**.

 **What is T2? What is its meaning? And its CPU and RAM configuration?**

* **T2** instances are burstable performance instances. They offer a baseline level of CPU performance with the ability to burst above the baseline.
* **T2.micro:** 1 vCPU, 1 GiB RAM
* **T2.medium:** 2 vCPUs, 4 GiB RAM
* **T2.large:** 2 vCPUs, 8 GiB RAM

**Security Group**

|  |
| --- |
| what you understand by security group? explain types ?its permissions? |
| Why security group is required? |
| How will you rescue something in cloud? |
| What is the default outgoing traffic to the security group? |

**Security Group**: A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. In AWS, security groups are used to control the traffic to and from EC2 instances.

**Types**:

1. **Inbound Rules**: Define the incoming traffic to your instances.
2. **Outbound Rules**: Define the outgoing traffic from your instances.

**Permissions**:

* Security groups can allow or deny traffic based on IP protocols, port numbers, and source/destination IP addresses.

**Why Security Group is Required**:

* **Control Traffic**: They help control the flow of traffic to and from instances.
* **Security**: They provide a layer of security by allowing only the necessary traffic to your instances.
* **Management**: They help manage network traffic easily with defined rules.

**Default Outgoing Traffic**

By default, a security group allows all outbound traffic from an instance. This means that any instance associated with a security group can send traffic out unless specifically restricted.

**Rescuing Instances in the Cloud**

To rescue an instance in the cloud:

1. **Stop the Instance**.
2. **Detach the Root Volume**.
3. **Attach the Root Volume to Another Instance**.
4. **Fix the Issue (e.g., remove a misconfiguration or malware)**.
5. **Reattach the Volume to the Original Instance**.
6. **Start the Instance**.

-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Key Pair**

|  |
| --- |
| My keypair is lost so it is possible to launch that instance? |
| Can you explain what is the keypair? |
| What is use of keypairs |
| How to connect different machine if 1 is created instance & 1 is linux machine |

**Key Pair**: A key pair consists of a public key and a private key. The public key is stored by AWS, and the private key is kept by the user.

**Use of Key Pairs**:

* **SSH Access**: They are used to securely connect to your instances via SSH.
* **Authentication**: They serve as a means of authentication without using passwords.

**Lost Key Pair**:

* If you lose your key pair, you cannot access the instance. You would need to create a new instance or use other methods such as creating a new key pair and attaching the volume to a new instance.

**Connecting Different Machines**

To connect a created instance and a Linux machine:

1. **Using SSH**: Use the SSH command with the private key associated with the instance.
2. **Example**:

bash

Copy code

ssh -i /path/to/private-key.pem ec2-user@instance-ip

-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Amazon Machine Image (AMI)**

|  |
| --- |
| what is AMI ? how to create it ? |
| what is diffrence between market palce and Cummunity AMI? |
| How can i check how many AMI i shared ? |
| What does AMI include |
| Can I launch multiple instance using Ami? |
| Can we share AMI to cross region? |
| How to get VM backup |

**1 AMI**: An Amazon Machine Image (AMI) provides the information required to launch an instance. It includes:

* **Template**: A template for the root volume of the instance.
* **Launch Permissions**: Controls which AWS accounts can use the AMI to launch instances.
* **Block Device Mapping**: Specifies the volumes to attach to the instance when it is launched.

**Creating an AMI**:

1. **From an Instance**: Create an AMI from an existing EC2 instance using the AWS Management Console or CLI.
2. **From a Snapshot**: Create an AMI from EBS snapshots.

**2 Marketplace vs. Community AMI**

* **Marketplace AMI**: Pre-configured by vendors and available for purchase. Often include licensed software.
* **Community AMI**: Created by AWS users and shared with the community. Usually free to use but come with no guarantees or support.

**3 Checking Shared AMIs**

To check how many AMIs you have shared:

1. **AWS Console**: Go to the AMIs section in the EC2 dashboard and filter by "Owned by me" and "Public".

**4 Launching Multiple Instances Using an AMI**

Yes, you can launch multiple instances from a single AMI. When you launch an instance, you select an AMI to use, and the same AMI can be used to launch multiple instances.

**5 Sharing AMIs Across Regions**

No, AMIs are region-specific by default. However, you can copy an AMI from one region to another using the AWS Management Console or CLI.

**6 VM Backup**

To back up a VM:

1. **Create a Snapshot**: Take a snapshot of the volumes.
2. **Create an AMI**: Create an AMI from the instance or snapshots.
3. **Use Backup Solutions**: Use AWS Backup or third-party backup solutions for more advanced backup management

EBS:

|  |  |
| --- | --- |
| 1 | What is EBS?Types of EBS? |
| 2 | Can you delete our own snapshot ? |
| 3 | Use of snapshot |
| 4 | What are the different types of EBS are available? |
| 5 | Types of volumes in your organization? |
| 6 | How will you encrypt the EBS volume and EC2 |
| 7 | How can we attach snapshot in diffrent region? |
| 8 | i want miragte volume to another account is it possible? |
| 9 | Is it possible to decrease size of volume? |
| 10 | while creating volume encryption is possible?share encryption name? |
| 11 | Is volume attach or not how will you check? |
| 12 | Which protocol you used for EBS? |
| 13 | What do you mean by incremental snapshot |
| 14 | what is SNS?what is practical use of SNS? |
| 15 | how to deleted ebs snapshots? |
| 16 | can we auto delete snapshots? |
| 17 | Explain me some Ebs Volume types & which one is fast |
| 18 | Storing size of EBS?its pricing structure? |
| 19 | Linux command for see attach EBS volume? |
| 20 | EBS and instance store volume difference |
| 21 | Can EBS support file system interface? |
| 22 | EBS is faster than S3?which is cheap EBS or S3 |
| 23 | which storage you preffered for data store? why? |

 **What is EBS? Types of EBS?**

* **EBS (Elastic Block Store)** is a scalable block storage service provided by AWS. It provides persistent storage that can be attached to EC2 instances.
* **Types of EBS volumes**:
  + **General Purpose SSD (gp3 and gp2)**: Balanced performance for a broad range of applications.
  + **Provisioned IOPS SSD (io2 and io1)**: High performance for demanding applications.
  + **Throughput Optimized HDD (st1)**: Low cost for frequently accessed, throughput-intensive workloads.
  + **Cold HDD (sc1)**: Lowest cost for infrequently accessed data.

 **Can you delete your own snapshot?**

* Yes, you can delete your own snapshots using the AWS Management Console, AWS CLI, or AWS SDKs.

 **Use of snapshot**

* Snapshots are used to back up the state of an EBS volume at a specific point in time. They can be used for creating new volumes, disaster recovery, and migration of data.

 **What are the different types of EBS available?**

* As mentioned above, the primary types are:
  + General Purpose SSD (gp3 and gp2)
  + Provisioned IOPS SSD (io2 and io1)
  + Throughput Optimized HDD (st1)
  + Cold HDD (sc1)

 **Types of volumes in your organization?**

* This would depend on the specific setup of your AWS environment, but commonly used volumes include general-purpose SSDs (gp3), provisioned IOPS SSDs (io2), and throughput-optimized HDDs (st1).

 **How will you encrypt the EBS volume and EC2?**

* **EBS Encryption**: You can enable encryption when creating a new EBS volume or use AWS Key Management Service (KMS) to encrypt existing volumes.
* **EC2 Encryption**: Data in transit can be encrypted using protocols such as HTTPS. For data at rest, you can use encrypted EBS volumes.

 **How can we attach a snapshot in a different region?**

* To use a snapshot in a different region, you need to copy the snapshot to the target region using the AWS Management Console, AWS CLI, or AWS SDKs. Once copied, you can create a new EBS volume from the snapshot in the target region.

 **I want to migrate a volume to another account. Is it possible?**

* Yes, you can share EBS snapshots with another AWS account and then use that snapshot to create a new volume in the target account.

 **Is it possible to decrease the size of a volume?**

* No, AWS does not support decreasing the size of an EBS volume directly. However, you can create a new, smaller volume, copy the data, and then switch volumes.

 **While creating a volume, is encryption possible? Share encryption name?**

* Yes, you can enable encryption when creating a volume. You use AWS Key Management Service (KMS) for encryption, and you can specify the KMS key ID or use the default AWS key.

 **Is the volume attached or not? How will you check?**

* You can check if an EBS volume is attached to an EC2 instance using the AWS Management Console, AWS CLI (describe-volumes), or AWS SDKs.

 **Which protocol do you use for EBS?**

* EBS volumes use the iSCSI protocol over a network interface.

 **What do you mean by incremental snapshot?**

* Incremental snapshots only save the changes made since the last snapshot, reducing storage costs and backup time.

 **What is SNS? What is the practical use of SNS?**

* **SNS (Simple Notification Service)** is a fully managed messaging service that allows you to send notifications or messages to a large number of subscribers or endpoints. It’s often used for sending alerts, notifications, and coordinating messages between distributed systems.

 **How to delete EBS snapshots?**

* You can delete EBS snapshots via the AWS Management Console, AWS CLI (delete-snapshot), or AWS SDKs.

 **Can we auto-delete snapshots?**

* Yes, you can use AWS Data Lifecycle Manager (DLM) to create policies that automatically delete snapshots based on criteria you define.

 **Explain some EBS Volume types & which one is fast**

* **General Purpose SSD (gp3)**: Balanced performance.
* **Provisioned IOPS SSD (io2)**: High performance for IOPS-intensive applications. It is the fastest option for performance-critical applications.

 **Storing size of EBS? Its pricing structure?**

* EBS volumes range from 1 GiB to 64 TiB. Pricing depends on the volume type and size, with additional costs for IOPS and provisioned throughput.

 **Linux command to see attached EBS volume?**

* You can use lsblk or df -h to list attached volumes and their mount points.

 **EBS and instance store volume difference?**

* **EBS**: Persistent storage that persists beyond instance termination.
* **Instance Store**: Temporary storage that is lost when the instance is stopped or terminated.

 **Can EBS support file system interface?**

* EBS volumes support file systems (e.g., ext4, NTFS) once they are mounted to an EC2 instance.

 **Is EBS faster than S3? Which is cheaper: EBS or S3?**

* **EBS** is generally faster for block-level storage and is used for applications requiring high performance. **S3** is object storage and is typically used for large-scale data storage. S3 is usually cheaper for large-scale data storage compared to EBS.

 **Which storage do you prefer for data storage and why?**

* This depends on the use case. For high-performance applications requiring low latency, **EBS** is preferred. For scalable, cost-effective data storage, **S3** is often chosen.

**Elastic IPs (EIPs)**

1. **What is Elastic IP?**
   * **Elastic IP (EIP)** is a static, public IPv4 address designed for dynamic cloud computing. It allows you to associate a fixed IP address with your EC2 instance, which remains constant even if you stop and start the instance.
2. **EIP default launch limit**
   * By default, you can have up to 5 Elastic IP addresses per AWS account per region. This limit can be increased by submitting a request to AWS Support.
3. **How many Elastic IPs can we create in one VPC?**
   * Elastic IPs are associated with your AWS account rather than specific VPCs. However, they are used within the context of a VPC. The number of Elastic IPs you can allocate in a VPC is subject to the same limit as for the entire account (default is 5 per region).

----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Load Balancer**

|  |  |
| --- | --- |
| 1 | Can we use multiple load balancer in single EC2 machine? |
| 2 | customer want to stream 50 server what type of load balancer is used? |  |
| 3 | what is connection draining? | NLB (https://aws.amazon.com/blogs/aws/new-udp-load-balancing-for-network-load-balancer/) |
| 4 | What are the different types of load balancer? | sunil shetty of AWS |
| 5 | you have 50 database server which load balancer you used? |  |
| 6 | What you know about load balancer?types of load balancer? how you create? |  |
| 7 | Can you explain the working of network load balancer?and its pratical use |  |
| 8 | Can you assign static IP to load balancer? |  |
| 9 | Diff bet ALB and NLB |  |
| 10 | Have you worked on provisioning on AWS like DB Load balancer and servers | <https://blog.cloudcraft.co/alb-vs-nlb-which-aws-load-balancer-fits-your-needs/> |
| 11 | Why does ELB hold many IP addresses? |  |
| 12 | Why classic load balancer is not used many time? |  |
| 13 | what is listener?Listner component in LB | <https://cloudacademy.com/blog/application-load-balancer-vs-classic-load-balancer/> |
| 14 | HTTP vs HTTPS | **A listener is a process that checks for connection requests, using the protocol and port that you configure** |
| 15 | In LB it is possible to change default port number? what is the condition? |  |
| 16 | 404 error? | <https://docs.aws.amazon.com/elasticbeanstalk/latest/dg/environments-cfg-alb.html> |
| 17 | In LB which one is faster and why ? |  |
| 18 | How will load decide where to go ? | <https://aws.amazon.com/elasticloadbalancing/features/> |
|  |  |  |

1. **Can we use multiple load balancers on a single EC2 machine?**
   * Yes, you can use multiple load balancers to route traffic to a single EC2 instance. Each load balancer can distribute traffic to the instance based on different rules or protocols.
2. **Customer wants to stream to 50 servers. What type of load balancer is used?**
   * For streaming data to multiple servers, a **Network Load Balancer (NLB)** is often used due to its high performance and ability to handle large volumes of TCP traffic.
3. **What is connection draining?**
   * **Connection draining** allows existing connections to complete before the load balancer stops routing traffic to an instance that is being deregistered or terminated. This ensures a smooth transition and prevents dropped connections.
4. **What are the different types of load balancers?**
   * **Application Load Balancer (ALB)**: Operates at the application layer (Layer 7), supports HTTP and HTTPS, and is ideal for advanced routing.
   * **Network Load Balancer (NLB)**: Operates at the network layer (Layer 4), supports TCP and UDP, and is designed for high performance and low latency.
   * **Classic Load Balancer (CLB)**: Operates at both the network and application layers but is older and less feature-rich compared to ALB and NLB.
5. **You have 50 database servers; which load balancer would you use?**
   * For database servers, **Network Load Balancer (NLB)** is typically used due to its ability to handle high-throughput, low-latency traffic and support for TCP connections.
6. **What do you know about load balancers? Types of load balancers? How do you create them?**
   * Load balancers distribute incoming traffic across multiple targets (EC2 instances, containers, etc.) to ensure high availability and fault tolerance.
   * Types include ALB, NLB, and CLB.
   * To create a load balancer, use the AWS Management Console, AWS CLI (create-load-balancer), or AWS SDKs. You need to configure listeners, target groups, and health checks.
7. **Can you explain the working of Network Load Balancer (NLB) and its practical use?**
   * **NLB** operates at Layer 4 (transport layer) and routes TCP/UDP traffic to targets based on IP address and port. It is designed for high performance, handling millions of requests per second while maintaining ultra-low latency. It is ideal for applications requiring high throughput and low latency.
8. **Can you assign a static IP to a load balancer?**
   * **NLB** supports static IP addresses and allows you to associate Elastic IPs with the load balancer. **ALB** does not support static IPs directly, but you can use Route 53 with a custom domain for a static endpoint.
9. **Difference between ALB and NLB?**
   * **ALB**: Works at Layer 7, supports advanced routing based on HTTP headers, paths, and hostnames. Ideal for web applications and microservices.
   * **NLB**: Works at Layer 4, handles TCP and UDP traffic, provides high performance and low latency. Suitable for applications requiring high throughput.
10. **Have you worked on provisioning on AWS like DB, Load Balancer, and servers?**
    * Yes, provisioning involves setting up and configuring AWS resources such as databases, load balancers, and EC2 instances to meet application requirements.
11. **Why does ELB hold many IP addresses?**
    * **Elastic Load Balancer (ELB)**, especially NLB, can hold many IP addresses to handle large volumes of traffic and distribute it efficiently across multiple instances.
12. **Why is the Classic Load Balancer not used many times?**
    * **Classic Load Balancer (CLB)** is less feature-rich compared to ALB and NLB. It lacks advanced features such as content-based routing and is less suitable for modern, complex applications.
13. **What is a listener? Listener component in LB?**
    * A **listener** is a process that checks for connection requests using a specific protocol and port. It routes traffic to the appropriate target group based on the rules you configure. For example, an HTTP listener on port 80 or an HTTPS listener on port 443.
14. **HTTP vs HTTPS**
    * **HTTP (Hypertext Transfer Protocol)**: Transfers data in plaintext, not encrypted.
    * **HTTPS (HTTP Secure)**: Uses SSL/TLS to encrypt data during transfer, providing secure communication.
15. **In LB, is it possible to change the default port number? What is the condition?**
    * Yes, you can configure listeners on non-default ports. For instance, you can set an ALB or NLB to listen on any port as needed, not just the default ports (80 for HTTP, 443 for HTTPS).
16. **404 error?**
    * A **404 error** indicates that the requested resource or page was not found on the server. It often occurs if the URL is incorrect or the resource has been moved or deleted.
17. **In LB, which one is faster and why?**
    * **Network Load Balancer (NLB)** is typically faster than ALB for TCP traffic because it operates at Layer 4 and can handle millions of requests per second with low latency.
18. **How will the load decide where to go?**
    * The load balancer uses routing rules, algorithms (like round-robin, least connections), and health checks to decide how to distribute traffic to different instances or targets.

vpc

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | What is the connectivity options for VPC? | VPC |  | <https://www.tutorialspoint.com/amazon_web_services/amazon_web_services_virtual_private_cloud.htm> |
| 2 | Can we use existing AMI in new vpc? |  |  |
| 3 | what is difference between NAT gatway and NAT instance? |  | <https://docs.aws.amazon.com/vpc/latest/userguide/vpc-nat-comparison.html> |
| 4 | What is subnet? |  |  |
| 5 | How many vpc can I create? |  |  |
| 6 | how many subnet have in one VPC? |  |  |
| 7 | Do you know the difference between security group and NACL? |  |  |
| 8 | Explain NACL |  |  |
| 9 | how to monitor AWS VPC and employees not able access server in another VPC how to troubleshoot? |  |  |
| 10 | where we use peering connection ? or VPC peering for what purpose? in VPC peering I am not able communicate?how I can troubleshoot? |  |  |
| 11 | 172.16.1.1 it is public or private? |  |  |
| 12 | you have 50 database server where would you place it in VPC? |  |  |
| 13 | Is it possible to establish peering connection to vpc in different regions? |  |  |
| 14 | How can a EC2 instance in private vpc established a connection to the internet? |  |  |
| 15 | Can you explain the difference between public and private subnet? |  |  |
| 16 | What is CIDR? |  |  |
| 17 | Tell me any five networking commands with their use case? |  | <https://mindmajix.com/linux-networking-commands-best-examples> |
| 18 | NAT instance is a part of which service? VPC |  |  |
| 19 | What is route table? |  |  |
| 20 | Can you tell me the difference between state full and stateless firewall? |  |  |
| 21 | which cofiguration have you done in your private VPC ? |  |  |
| 22 | diff between NAT and NACL |  |  |
| 23 | how to access private vpc from public ? |  |  |
| 24 | Components of vpc |  |  |
| 25 | How to change CIDR range if vpc is created |  |  |
| 26 | VPC Endpoints |  |  |
| 27 | VPC peering connection manual or Teraform |  |  |
| 28 | Site to side VPN |  |  |
| 29 | NAT gateway where we create and what will be in the route table |  |  |
| 30 | How many IGW connect to VPC |  |  |
| 31 | Diff between private and public subnet |  |  |
| 32 | Where do we keep the NAT gateway |  |  |
| 33 | What is diff between USER and ROLE |  |  |
| 34 | what is positive use of private ip address? |  |  |
| 35 | what is VPC and IGW ? |  |  |
| 36 | Can we change EC2 instance private ip address when it is ruuning state or in stopped state...Reason? |  |  |
| 37 | How do you secure EC2 inside VPC |  |  |
| 38 | Sercurity in VPC? |  |  |
| 39 | property of broadcasting and multicasting |  | <https://www.geeksforgeeks.org/difference-between-broadcast-and-multicast/> |
| 40 | What is routing table and its purpose? |  |  |
| 41 | can you launch EC2 instance with predefine private IP address? |  |  |
| 42 | What is IP address? |  |  |
| 43 | Suppose i have 2 CIDR block so is it possible to connect using same IP |  |  |
| 44 | Different layers of OSI model |  |  |
| 45 | what is rule number? |  |  |
| 46 | what do you mean by transitive peering? |  |  |
| 47 | If i give you IP and it is not connecting then what troubleshoot you will do? |  |  |
|  |  |  |  |  |

1. **Connectivity options for VPC:**
   * Internet Gateway (IGW)
   * Virtual Private Gateway (VGW) for VPN
   * VPC Peering
   * AWS Direct Connect
   * VPC Endpoints (Interface and Gateway)
2. **Using existing AMI in a new VPC:**
   * Yes, you can use an existing AMI in a new VPC. The AMI is independent of the VPC.
3. **Difference between NAT Gateway and NAT Instance:**
   * NAT Gateway: Managed service, highly available, scalable, and easier to use.
   * NAT Instance: Self-managed, requires scaling and high availability configuration, more control over traffic.
4. **What is a subnet?**
   * A subnet is a range of IP addresses in your VPC. It allows you to segment your VPC into smaller, manageable sections.
5. **How many VPCs can you create?**
   * You can create up to 5 VPCs per region by default. This limit can be increased by requesting a limit increase.
6. **How many subnets in one VPC?**
   * There is no fixed limit on the number of subnets you can have in a VPC, but there is a limit on the number of IP addresses.
7. **Difference between Security Group and NACL:**
   * Security Group: Stateful, operates at the instance level, and allows or denies traffic based on rules.
   * Network ACL (NACL): Stateless, operates at the subnet level, and allows or denies traffic based on rules.
8. **Explain NACL:**
   * Network ACL (NACL) is a firewall for controlling traffic entering or leaving a subnet. It can have rules for both inbound and outbound traffic.
9. **Monitoring AWS VPC and troubleshooting access issues:**
   * Use CloudWatch Logs and VPC Flow Logs for monitoring. Check route tables, security groups, and NACLs for troubleshooting.
10. **Purpose of VPC Peering:**
    * VPC Peering allows you to connect two VPCs to route traffic between them. For troubleshooting, check route tables, security groups, and ensure that peering connections are properly configured.
11. **172.16.1.1 - Public or private?**
    * It is a private IP address, as it falls within the private IP address range (172.16.0.0 - 172.31.255.255).
12. **Placing 50 database servers in VPC:**
    * Typically placed in private subnets for security and to prevent direct internet access.
13. **Peering connection across regions:**
    * Yes, it is possible with inter-region VPC peering.
14. **EC2 instance in a private VPC accessing the internet:**
    * Use a NAT Gateway or NAT Instance in a public subnet.
15. **Difference between public and private subnet:**
    * Public Subnet: Has a route to an IGW, instances can communicate with the internet.
    * Private Subnet: No direct route to the internet, instances typically use a NAT Gateway for external communication.
16. **What is CIDR?**
    * Classless Inter-Domain Routing, a method for allocating IP addresses and routing IP packets.
17. **Five networking commands:**
    * ping - Test connectivity to another host.
    * traceroute - Track the path packets take to a destination.
    * netstat - Display network connections and statistics.
    * nslookup - Query DNS to obtain domain name or IP address mapping.
    * ifconfig / ip - Display or configure network interfaces.
18. **NAT Instance service:**
    * NAT Instance is a part of the VPC service.
19. **What is a route table?**
    * A route table contains rules (routes) that determine where network traffic from your subnet or gateway is directed.
20. **Difference between stateful and stateless firewall:**
    * Stateful: Tracks active connections and maintains connection state.
    * Stateless: Treats each packet independently, without tracking connections.
21. **Private VPC configuration:**
    * Example configurations might include subnets, route tables, security groups, and NACLs.
22. **Difference between NAT and NACL:**
    * NAT (Network Address Translation) allows instances in a private subnet to access the internet.
    * NACL is used to control traffic to and from subnets.
23. **Accessing private VPC from public:**
    * Use a VPN connection, VPC Peering, or Transit Gateway.
24. **Components of VPC:**
    * Subnets, Route Tables, Internet Gateways, NAT Gateways, VPC Peering, Security Groups, NACLs.
25. **Changing CIDR range of a VPC:**
    * You cannot change the CIDR block of an existing VPC. You must create a new VPC with the desired CIDR block.
26. **VPC Endpoints:**
    * VPC Endpoints allow private connections to AWS services without going through the internet. Types include Interface and Gateway endpoints.
27. **VPC Peering connection manual or Terraform:**
    * Both methods are valid. Terraform can automate and manage peering connections as part of infrastructure as code.
28. **Site-to-Site VPN:**
    * Connects your on-premises network to your VPC securely over the internet using IPsec.
29. **Creating NAT Gateway and route table:**
    * Create a NAT Gateway in a public subnet and update the route table of private subnets to direct internet-bound traffic to the NAT Gateway.
30. **IGWs per VPC:**
    * You can have only one Internet Gateway attached to a VPC.
31. **Difference between private and public subnet:**
    * Public Subnet: Routes traffic to and from the internet.
    * Private Subnet: No direct internet access, uses NAT for internet-bound traffic.
32. **Where to keep NAT Gateway:**
    * NAT Gateway should be placed in a public subnet.
33. **Difference between USER and ROLE:**
    * User: An individual identity with specific permissions.
    * Role: An AWS identity with specific permissions that can be assumed by users or services.
34. **Positive use of private IP address:**
    * Private IP addresses are used for internal communication within a network and are not routable over the internet.
35. **What is VPC and IGW?**
    * VPC: Virtual Private Cloud, a virtual network within AWS.
    * IGW: Internet Gateway, a gateway that allows communication between instances in a VPC and the internet.
36. **Changing EC2 instance private IP address:**
    * You can change the private IP address of an EC2 instance when it is stopped. The address can be reassigned or modified.
37. **Securing EC2 inside VPC:**
    * Use Security Groups and NACLs, apply encryption, and follow best practices for instance security.
38. **Security in VPC:**
    * Involves configuring Security Groups, NACLs, encryption, IAM roles, and monitoring.
39. **Broadcasting and multicasting:**
    * Broadcasting: Sending data to all devices on a network segment.
    * Multicasting: Sending data to a specific group of devices.
40. **Routing Table:**
    * Directs network traffic based on destination IP addresses and routing rules.
41. **Launching EC2 with predefined private IP:**
    * Yes, you can specify a private IP address when launching an EC2 instance in a VPC.
42. **What is IP address?**
    * An IP address is a unique identifier assigned to each device on a network for communication.
43. **Connecting with 2 CIDR blocks:**
    * Each CIDR block should have unique IP ranges. Overlapping IP ranges are not allowed.
44. **Different layers of OSI model:**
    * Physical, Data Link, Network, Transport, Session, Presentation, Application.
45. **Rule number:**
    * In Security Groups and NACLs, rule numbers are used to prioritize rules and define their order.
46. **Transitive Peering:**
    * Transitive Peering allows indirect communication between VPCs via a third VPC. However, AWS does not support transitive peering directly.
47. **Troubleshooting connectivity:**
    * Check IP configuration, security groups, NACLs, route tables, and ensure the network path is correct.

Autoscaling:

|  |  |
| --- | --- |
| 1 | What is the difference between load balancer and auto scaling |
| 2 | how to add existing instance in autoscalling group? |
| 3 | What is launch group in autoscaling? |
| 4 | Can you explain what is autoscaling? |
| 5 | What are the situation to move the traffic or how many ways to distribute the traffic from one ec2 to another, what weight you provide |
| 6 | if supposed thousand of request comes on your web page and lods was increases what you do and how can you decrease load. |
| 7 | Benefit of autoscaling? |
| 8 | Policies in autoscalling |
| 9 | Scale up and scale down ? |
| 10 | how much time require to pass health check |
| 11 | Min and max. desired value ? explain |
| 12 | what is horizonatal and vertical scaling? |
| 13 | is rebalancing work in Auto scaling |
| 14 | warm up time ? |
| 15 | what information you collect from your client regarding to autoscaling? |
| 16 | use of health check? |
| 17 | which instance you create for autoscalling |

 **Difference between Load Balancer and Auto Scaling:**

* **Load Balancer**: Distributes incoming network traffic across multiple instances to ensure no single instance is overwhelmed, improving availability and fault tolerance.
* **Auto Scaling**: Automatically adjusts the number of instances in a group based on traffic or load. It scales out (adds instances) or scales in (removes instances) to handle changes in demand.

 **Adding an Existing Instance to an Auto Scaling Group:**

* You typically can’t directly add an existing instance to an Auto Scaling group. Instead, you would need to configure a launch configuration or template and allow Auto Scaling to launch new instances based on that configuration.

 **Launch Configuration/Template in Auto Scaling:**

* **Launch Configuration**: Defines the instance type, AMI, and other settings used to launch new instances in an Auto Scaling group.
* **Launch Template**: A newer and more flexible option that allows for versioning and additional parameters.

 **Explanation of Auto Scaling:**

* Auto Scaling automatically adjusts the number of running instances in your application’s infrastructure to match demand. It helps ensure that you have the right number of instances available at any time, which improves performance and reduces costs.

 **Ways to Distribute Traffic from One EC2 to Another:**

* **Load Balancer**: Distributes traffic across multiple instances based on predefined algorithms (e.g., round-robin, least connections).
* **Weighted Routing**: If using a load balancer or Route 53, you can assign weights to different instances or endpoints to control traffic distribution.

 **Handling Increased Load:**

* **Auto Scaling**: Automatically adds more instances to handle the increased load.
* **Load Balancing**: Ensures the traffic is evenly distributed across available instances.
* **Caching**: Use caching mechanisms to reduce the load on your servers.
* **Optimize Application**: Optimize the application for better performance and efficiency.

 **Benefits of Auto Scaling:**

* **Cost Efficiency**: Reduces costs by automatically decreasing the number of instances when demand is low.
* **Improved Availability**: Ensures that the application can handle traffic spikes without downtime.
* **Automatic Scaling**: Adjusts resources based on traffic and load without manual intervention.

 **Policies in Auto Scaling:**

* **Scaling Policies**: Define how and when to scale in or out based on metrics like CPU utilization or network traffic.
* **Health Check Policies**: Determine when to replace unhealthy instances.
* **Scheduled Scaling**: Scales based on a predefined schedule.

 **Scale Up and Scale Down:**

* **Scale Up**: Increase the size or capacity of existing instances (vertical scaling).
* **Scale Down**: Decrease the size or capacity of existing instances.
* **Scale Out**: Add more instances (horizontal scaling).
* **Scale In**: Remove instances.

 **Time Required to Pass Health Check:**

* This depends on the configuration of your health check settings. Commonly, it involves a series of health check attempts and time intervals before an instance is deemed healthy or unhealthy.

 **Min, Max, and Desired Values:**

* **Min**: The minimum number of instances that should be running in the Auto Scaling group.
* **Max**: The maximum number of instances that can be running.
* **Desired**: The number of instances that you want to maintain based on current load and policies.

 **Horizontal and Vertical Scaling:**

* **Horizontal Scaling**: Adding more instances to handle increased load (scale out) or removing instances (scale in).
* **Vertical Scaling**: Increasing or decreasing the size or power of existing instances (e.g., upgrading instance types).

 **Rebalancing in Auto Scaling:**

* **Rebalancing**: Ensures that instances are distributed evenly across different Availability Zones. This process helps maintain high availability and fault tolerance.

 **Warm-Up Time:**

* The period after launching a new instance during which it is not yet considered fully available. This time allows the instance to fully initialize and become ready to handle traffic.

 **Information to Collect from Clients Regarding Auto Scaling:**

* **Traffic Patterns**: Understanding peak usage times and traffic patterns.
* **Application Requirements**: Specific needs for scaling, such as CPU, memory, and storage requirements.
* **Budget Constraints**: Limits on cost for scaling operations.

 **Use of Health Check:**

* Health checks determine if an instance is functioning properly. Unhealthy instances are replaced to maintain the performance and reliability of the application.

 **Instances for Auto Scaling:**

* Typically, you would use instances that are defined in your launch configuration or template. These should be well-suited for your application’s requirements and expected traffic.

S3

 **S3 Storage Classes:**

* Standard
* Intelligent-Tiering
* One Zone-IA (Infrequent Access)
* Glacier
* Glacier Deep Archive
* Reduced Redundancy Storage (deprecated but still available)

 **S3 Bucket:**

* An S3 bucket is a container for storing objects in Amazon S3.
* By default, you can create up to 100 S3 buckets per AWS account. This limit can be increased by contacting AWS support.

 **Versioning in S3:**

* Versioning allows you to keep multiple versions of an object in a bucket. This helps in recovering objects from accidental deletions or overwrites.

 **What is S3 and CDN:**

* S3 (Simple Storage Service) is an object storage service that offers high durability and availability.
* Yes, you can use Amazon CloudFront (CDN) in front of S3 to distribute content globally with low latency.

 **Block Storage vs. Object Storage:**

* Block Storage: Stores data in blocks and is typically used with databases and applications. Example: Amazon EBS.
* Object Storage: Stores data as objects, each with metadata and a unique identifier. Example: Amazon S3.

 **Redundancy:**

* Amazon S3 provides redundancy across multiple data centers within a region. Data is replicated automatically to ensure high durability.

 **Monitoring S3 Cross-Region Replication:**

* Use Amazon CloudWatch metrics and S3 replication metrics to monitor the status and performance of cross-region replication.

 **Uploading a 120MB File to S3:**

* Use multipart upload for files larger than 100MB. AWS SDKs and the AWS CLI support multipart uploads to efficiently upload large files.

 **Using S3 with EC2:**

* Yes, EC2 instances can interact with S3 buckets. You can use the AWS SDK or AWS CLI on EC2 to upload, download, and manage objects in S3.

 **Static vs. Dynamic Website Hosting on S3:**

* You can host static websites on S3. S3 does not support dynamic website hosting; you would need to use services like AWS Lambda or AWS Elastic Beanstalk for dynamic content.

 **Replication in S3:**

* S3 replication copies objects from one bucket to another, either within the same region or across different regions.
* **Cross-Region Replication (CRR)**: Replicates objects to a different AWS region. The default condition is that replication must be enabled in both the source and destination buckets, and the destination bucket must be in a different region.

 **Default Storage Class:**

* The default storage class for new objects in S3 is Standard.

 **S3 Data Encryption Types:**

* Server-Side Encryption (SSE): SSE-S3, SSE-KMS, SSE-C.
* Client-Side Encryption: Managed by the client before uploading to S3.

 **Steps for Cross-Region Replication:**

* Enable versioning on both source and destination buckets.
* Set up an IAM role with necessary permissions for replication.
* Configure the replication rules in the source bucket’s properties.

 **S3 Lifecycle Policies:**

* Define actions to transition objects to different storage classes or delete them after a certain period.

 **Making a Public Bucket Private:**

* Update the bucket policy and access control lists (ACLs) to restrict public access.

 **Website Hosting Service:**

* For static website hosting, use Amazon S3. For dynamic websites, consider using AWS Elastic Beanstalk, AWS Lambda, or Amazon EC2.

 **Deploying a Static Website:**

* Recommend using Amazon S3 for storage and Amazon CloudFront for CDN. Consider Route 53 for DNS management.

 **Backing Up S3 with Script:**

* Use AWS CLI commands like aws s3 sync to create backups of S3 buckets. You can automate this with a script.

 **S3 Data Store Limit and Deleting Buckets:**

* There is no limit on the amount of data you can store in S3.
* To delete an S3 bucket, use the AWS Management Console, AWS CLI (aws s3 rb s3://bucket-name), or SDK.

 **How S3 Works and Architecture:**

* S3 stores data as objects in buckets. It is designed for high durability and availability, and its architecture includes multiple redundant data centers within a region.

 **Storage Classes and Glacier:**

* You can store data in various S3 storage classes, including Glacier for long-term archival storage.

 **Object Lock Feature in S3:**

* Object Lock enables you to enforce retention policies and prevent objects from being deleted or overwritten for a specified retention period.

 **S3 Buckets in Organization and Lifecycle Management:**

* The number of buckets and their contents vary by organization. Life cycle policies manage data transitions and deletions.

 **Cost of S3 Storage Classes:**

* Lower-cost storage classes like Glacier and Intelligent-Tiering are designed to be cost-effective for data that is infrequently accessed or archived.

EFS

|  |  |
| --- | --- |
| 1 | What is the difference between ebs and efs? |
| 2 | what is difference between EFS and S3? |
| 3 | how 5gb file stored in S3 and EFS? And what are the underlying protocols? |
| 4 | cost wise compair which of the storage is chipest and which one costly?---efs is costly |
| 5 | how taking backup of EFS |
| 6 | diff between EBS vs EFS vs S3 |
| 7 | Which volume type is not Bootable? |
| 8 | EFS similar to windows FSX |
| 9 | what is positive use of EFS ? |

**AWS Storage Options**

1. **Difference Between EBS and EFS:**
   * **EBS (Elastic Block Store)**: Block storage service used with EC2 instances. It's suitable for applications requiring a file system with low latency, such as databases.
   * **EFS (Elastic File System)**: File storage service that can be mounted across multiple EC2 instances. It's ideal for applications needing a shared file system.
2. **Difference Between EFS and S3:**
   * **EFS**: File storage service with file system semantics, suitable for shared access among multiple EC2 instances.
   * **S3 (Simple Storage Service)**: Object storage service for storing and retrieving any amount of data, suitable for web-scale applications, backups, and archives.
3. **Storing a 5GB File:**
   * **EFS**: Stores files as part of a file system and uses NFS (Network File System) protocol.
   * **S3**: Stores files as objects within buckets and uses HTTP/HTTPS protocol for access.
4. **Cost Comparison:**
   * **S3**: Generally cheaper for large-scale object storage with its pay-as-you-go model.
   * **EFS**: More expensive due to its managed file system capabilities and the ability to support multiple instances.
5. **Taking Backup of EFS:**
   * Use AWS Backup or create EFS-to-EFS backups through manual or automated scripts.
6. **Difference Between EBS, EFS, and S3:**
   * **EBS**: Block storage, single-instance attachment, low-latency, often used for databases.
   * **EFS**: File storage, multiple-instance attachment, scalable file system, used for shared access.
   * **S3**: Object storage, scalable, web-accessible, used for storing large volumes of unstructured data.
7. **Non-Bootable Volume Type:**
   * **EFS**: Not bootable; it's used for file storage rather than boot volumes.
8. **EFS vs. FSx:**
   * **EFS**: POSIX-compliant file system, designed for Linux instances.
   * **FSx**: Windows File Server, provides a managed Windows file system compatible with Windows environments.
9. **Positive Uses of EFS:**
   * Provides scalable and shared file storage for multiple EC2 instances.
   * Supports applications that require a common file system accessible by multiple instances.

**IAM (Identity and Access Management)**

|  |  |
| --- | --- |
| 1 | What is the difference between IAM role and IAM user |
| 2 | what is power user access in AWS? |
| 3 | what are the different IAM catagory you can control? |
| 4 | what is resorce base policy --- IAM policy |
| 5 | What are the IAM identities? |
| 6 | What is the role in IAM? IAM Group ? |
| 7 | What is policy in IAM? |
| 8 | What is the difference between manage policy and inline policy? |
| 9 | Can you explain the access key in IAM user? |
| 10 | how to give access to IAM user ? |
| 11 | what is IAM Role? |
| 12 | what is MFA in aws? |

1. **IAM Role vs. IAM User:**
   * **IAM User**: Represents a person or service needing AWS access. Has long-term credentials.
   * **IAM Role**: Represents a set of permissions that can be assumed by trusted entities (e.g., users, services).
2. **Power User Access:**
   * An IAM user with administrative privileges but not the ability to manage IAM users and groups.
3. **IAM Categories to Control:**
   * Users, groups, roles, permissions, policies, and MFA.
4. **Resource-Based Policies:**
   * Policies attached directly to AWS resources (e.g., S3 bucket policies).
5. **IAM Identities:**
   * Users, groups, roles, and service-linked roles.
6. **IAM Role vs. IAM Group:**
   * **IAM Role**: A set of permissions that can be assumed by entities.
   * **IAM Group**: A collection of IAM users with common permissions.
7. **IAM Policy:**
   * A document that defines permissions for actions on AWS resources.
8. **Managed Policy vs. Inline Policy:**
   * **Managed Policy**: Reusable policy that can be attached to multiple users, groups, or roles.
   * **Inline Policy**: Policy embedded directly within a user, group, or role.
9. **IAM User Access Key:**
   * Provides programmatic access to AWS services, used with SDKs and CLI.
10. **Giving Access to IAM User:**
    * Attach policies to the IAM user or their group to grant permissions.
11. **IAM Role:**
    * A set of permissions that can be assumed by trusted entities.
12. **MFA (Multi-Factor Authentication):**
    * An additional layer of security requiring a second form of verification besides the password.

**Monitoring and Logging**

|  |  |
| --- | --- |
| 1 | i have 50 server using which tool moniter and helth check the server?? |
| 2 | how to recover E2 instance using cloud watch?? |
| 3 | what is cloud watch?what is different actions in alarm? |
| 4 | will you monitor cross region replication? |
| 5 | what are different actions present in alarm?? |
| 6 | Can you tell me how alarm is set in cloudwatch? |
| 7 | Tell me about cloudwatch metrics?type of matrics |
| 8 | What is the purpose of log groups in cloudwatch? |
| 9 | from cloudwatch which services have you monitored ? |
| 10 | What metrics you have created for cloud watch |
| 11 | If any instance is stop or crashed how will you monitor it and manage it How the SNS is triggered for any this situation |
| 12 | which tool you used for monintoring |
| 13 | how will you delete alram |
| 14 | does cloudwatch agent support IAM role? |
| 15 | what is cloudwatch log agent stores ? |
| 16 | how can i get my log data? |
| 17 | what is events ? |

1. **Monitoring Tools for Servers:**
   * **Amazon CloudWatch**, **Nagios**, **Prometheus**, **Datadog**, **Zabbix**.
2. **Recovering EC2 Instance Using CloudWatch:**
   * Use CloudWatch alarms to trigger automated recovery actions.
3. **CloudWatch Overview:**
   * A monitoring service providing data and actionable insights from AWS cloud resources. Actions in alarms can include sending notifications, auto-scaling, and executing Lambda functions.
4. **Cross-Region Replication Monitoring:**
   * Yes, CloudWatch can monitor cross-region replication using custom metrics or logs.
5. **Different Actions in CloudWatch Alarms:**
   * Send notifications, auto-scaling, execute Lambda functions, stop/start/reboot instances.
6. **Setting Alarms in CloudWatch:**
   * Create an alarm based on metrics, set thresholds, and specify actions to take when the threshold is breached.
7. **CloudWatch Metrics:**
   * Metrics track performance and operational data, e.g., CPU utilization, disk I/O, network traffic.
8. **Purpose of Log Groups:**
   * Organize and manage logs from various sources, applying retention policies and access control.
9. **Services Monitored via CloudWatch:**
   * EC2, RDS, S3, Lambda, DynamoDB, and more.
10. **Metrics Created in CloudWatch:**
    * Custom metrics for application-specific data.
11. **Instance Stop or Crash Monitoring:**
    * Use CloudWatch alarms to detect instance state changes and SNS to send notifications or trigger recovery actions.
12. **Tool Used for Monitoring:**
    * CloudWatch, along with other third-party tools.
13. **Deleting Alarms:**
    * Use the CloudWatch console or API to delete alarms.
14. **CloudWatch Agent and IAM Role:**
    * Yes, CloudWatch Agent supports IAM roles for permissions.
15. **CloudWatch Log Agent Storage:**
    * Logs are stored in CloudWatch Logs service.
16. **Getting Log Data:**
    * Access through the CloudWatch console, CLI, or SDKs.
17. **Events:**
    * Refers to the occurrence of a particular condition or change in AWS resources, often used to trigger responses.

**SNS (Simple Notification Service)**

|  |  |
| --- | --- |
| 1 | what is topic in SNS? SNS service use ? |
| 2 | SNS and SQS are push baesd or pull based ? |
| 3 | which protocols are SNS supported ? |

1. **SNS Topic:**
   * A communication channel used to send messages to multiple subscribers.
2. **SNS and SQS Push/Pull:**
   * **SNS**: Push-based.
   * **SQS**: Pull-based.
3. **SNS Supported Protocols:**
   * HTTP/HTTPS, Email/Email-JSON, Short Message Service (SMS), Amazon SQS, Lambda, and more.

**CloudWatch Logs vs. VPC Flow Logs**

|  |  |
| --- | --- |
| 1 | what is difference between cloud watch log and VPC flow log? |
| 2 | How can you monitor a traffic in your VPC? |
| 3 | where vpc logs are stored ? |

1. **Difference Between CloudWatch Logs and VPC Flow Logs:**
   * **CloudWatch Logs**: Logs from various AWS services and applications.
   * **VPC Flow Logs**: Capture information about IP traffic going to and from network interfaces in a VPC.
2. **Monitoring Traffic in VPC:**
   * Use VPC Flow Logs to capture and analyze network traffic.
3. **Where VPC Logs are Stored:**
   * Stored in CloudWatch Logs or S3, depending on configuration.

**CloudTrail and Route 53**

|  |  |
| --- | --- |
| 1 | what is AWS trail ? What kind of logs stored? What is cloud trail? |
| 2 | someone accidentally deleted one server in customer environment how you know who did that? |
| 3 | some one deleted VPC how will you check who deleted VPC? |
| 4 | Difference between colud trail and cloudwatch |
| 5 | four pillars of cloud trails |
| 6 | what is events? |

1. **AWS CloudTrail:**
   * Logs API calls made in AWS, providing audit trails and security monitoring.
2. **Accidentally Deleted Server:**
   * Use CloudTrail logs to identify who made the API call that deleted the server.
3. **Deleted VPC:**
   * CloudTrail logs can be used to trace the deletion of a VPC and identify the responsible entity.
4. **Difference Between CloudTrail and CloudWatch:**
   * **CloudTrail**: Focuses on API call logs and user activity.
   * **CloudWatch**: Focuses on operational metrics and logs from resources.
5. **Four Pillars of CloudTrail:**
   * Governance, Compliance, and Risk Management; Security Analysis; Troubleshooting and Resolution; Resource Change Tracking.
6. **Events:**
   * Changes or occurrences in AWS resources that can trigger responses or be recorded.

**CloudFront**

|  |  |
| --- | --- |
| 1 | What do you mean by cloudFront?and it's flow?? |
| 2 | what is geo targeting in cloud front? |
| 3 | What are the different type of servers available in cloud front? |
| 4 | What is the use of edge locations? |
| 5 | What are the different distributions available in the cloud front? |
| 6 | diff between cloudwatch and cloudfront |
| 7 | what is invalidation?how to create invalidation ? |

1. **CloudFront Overview:**
   * A content delivery network (CDN) that caches and delivers content from edge locations to improve performance and reduce latency.
2. **Geo-Targeting:**
   * CloudFront can deliver content based on the geographic location of users.
3. **Types of Servers in CloudFront:**
   * Edge locations and origin servers.
4. **Use of Edge Locations:**
   * Cache content close to end-users to reduce latency and improve performance.
5. **Different Distributions in CloudFront:**
   * Web distributions and RTMP distributions.
6. **CloudWatch vs. CloudFront:**
   * **CloudWatch**: Monitoring and logging.
   * **CloudFront**: Content delivery and caching.
7. **Invalidation:**
   * Process of removing cached content from CloudFront edge locations. Create invalidations via the CloudFront console or API.

**RDS (Relational Database Service)**

|  |  |
| --- | --- |
| 1 | Whats is RDS? |
| 2 | What type instances you used for RDS Create table command |
| 3 | Draw structure of your web applications how to deploy and build project how you use middleware RDS and service. |

1. **What is RDS:**
   * Managed relational database service supporting multiple database engines like MySQL, PostgreSQL, SQL Server, Oracle, and MariaDB.
2. **RDS Instance Types for Create Table Command:**
   * Choose instance types based on workload requirements; e.g., db.t3.micro, db.m5.large.
3. **Web Application Deployment and RDS Usage:**
   * Design involves deploying web applications, using RDS for database management, and integrating middleware as needed

**Route 53**

|  |  |
| --- | --- |
| 1 | what is AWS route 53?why we use? |
| 2 | how cloud trail and route 53 work together? |
| 3 | Routing policy in route 53 |

1. **AWS Route 53:**
   * Managed DNS service that provides domain registration, DNS routing, and health checking.
2. **CloudTrail and Route 53 Interaction:**
   * CloudTrail logs API calls made to Route 53 for auditing and monitoring changes.
3. **Routing Policy in Route 53:**
   * Policies include Simple, Weighted, Latency, Failover, and Geolocation.

==============================================================================================================================================================================================

6) What does an AMI include?

An AMI includes the following things

* A template for the root volume for the instance
* Launch permissions decide which AWS accounts can avail the AMI to launch instances
* A block device mapping that determines the volumes to attach to the instance when it is launched

7) How can you send a request to Amazon S3?

Amazon S3 is a REST service, and you can send a request by using the REST API or the AWS SDK wrapper libraries that wrap the underlying Amazon S3 REST API.

8) Mention what the difference between Amazon S3 and EC2 is?

The difference between EC2 and Amazon S3 is that

EC2 S3

It is a cloud web service used for hosting your application

It is a data storage system where any amount of data can be stored

It is like a huge computer machine which can run either Linux or Windows and can handle applications like PHP, Python, Apache, or any databases

It has a REST interface and uses secure HMAC-SHA1 authentication keys

9) How many buckets can you create in AWS by default?

By default, you can create up to 100 buckets in each of your AWS accounts.

10) Explain can you vertically scale an Amazon instance? How?

Yes, you can vertically scale on the Amazon instance. For that

* Spin up a new larger instance than the one you are currently running
* Pause that instance and detach the root webs volume from the server and discard
* Then stop your live instance and detach its root volume
* Note the unique device ID and attach that root volume to your new server
* And start it again

11) Explain what T2 instances is?

T2 instances are designed to provide moderate baseline performance and the capability to burst to higher performance as required by the workload.

12) In VPC with private and public subnets, database servers should ideally be launched into which subnet?

With private and public subnets in VPC, database servers should ideally launch into private subnets.

13) Mention what the security best practices for Amazon EC2 are?

For secure Amazon EC2 best practices, follow the following steps

* Use AWS identity and access management to control access to your AWS resources
* Restrict access by allowing only trusted hosts or networks to access ports on your instance
* Review the rules in your security groups regularly
* Only open up permissions that you require
* Disable password-based login, for example, launched from your AMI

14) Explain how the buffer is used in Amazon web services?

The buffer is used to make the system more robust to manage traffic or load by synchronizing different components. Usually, components receive and process the requests in an unbalanced way. With the help of a buffer, the components will be balanced and will work at the same speed to provide faster services.

15) While connecting to your instance what are the possible connection issues one might face?

The possible connection errors one might encounter while connecting instances are

Connection timed out

User key not recognized by the server

Host key not found, permission denied

An unprotected private key file

Server refused our key or No supported authentication method available

Error using MindTerm on Safari Browser

Error using Mac OS X RDP Client

16) What are key-pairs in AWS?

Key-pairs are secure login information for your virtual machines. To connect to the instances, you can use key-pairs which contain a public-key and private-key.

17) What are the different types of instances?

Following are the types of instances:

1. General purpose
2. Computer Optimized
3. Memory Optimized
4. Storage Optimized
5. Accelerated Computing

18) Is the property of broadcast or multicast supported by Amazon VPC?

No, currently Amazon VPI does not provide support for broadcast or multicast.

19) How many Elastic IPs are allowed to be created by AWS?

5 VPC Elastic IP addresses are allowed for each AWS account.

20) Explain default storage class in S3

The default storage class is a Standard frequently accessed.

21) What are the Roles?

Roles are used to provide permissions to entities which you can trust within your AWS account. Roles are very similar to users. However, with roles, you do not require to create any username and password to work with the resources.

22) What are the edge locations?

Edge location is the area where the contents will be cached. So, when a user is trying to access any content, the content will automatically be searched in the edge location.

23) What is VPC?

aws-logoVPC stands for Virtual Private Cloud. It allows you to customize your networking configuration. It is a network which is logically isolated from another network in the cloud. It allows you to have your IP address range, internet gateways, subnet, and security groups.

24) Explain snowball

Snowball is a data transport option. It used source appliances to a large amount of data into and out of AWS. With the help of snowball, you can transfer a massive amount of data from one place to another. It helps you to reduce networking costs.

25) What is a redshift?

Redshift is a big data warehouse product. It is a fast and powerful, fully managed data warehouse service in the cloud.

26) What are the advantages of auto-scaling?

Following are the advantages of autoscaling

1. Offers fault tolerance
2. Better availability
3. Better cost management

27) What is meant by subnet?

A large section of IP Addresses divided into chunks is known as subnets.

28) Can you establish a Peering connection to a VPC in a different region?

Yes, we can establish a peering connection to a VPC in a different region. It is called inter-region VPC peering connection.

29) What is SQS?

Simple Queue Service is also known as SQS. It is distributed queuing service which acts as a mediator for two controllers.

30) How many subnets can you have per VPC?

You can have 200 subnets per VPC.

31) DNS and Load Balancer service comes under which type of cloud service?

DNS and Load Balancer and DNS services come under IAAS-storage cloud service.

32) What is the role of AWS CloudTrail?

CloudTrail is a specially designed tool for logging and tracking API calls. It helps to audit all S3 bucket accesses.

33) When was EC2 officially launched?

EC2 officially launched in the year 2006.

34) What is SimpleDB?

SimpleDB is a data repository of structure record which encourages data doubts and indexing both S3 and EC2are called SimpleDB.

35) Explain Amazon ElasticCache

Amazon Elasticcache is a web service which makes it easy to deploy, scale and store data in the cloud.

36) What is AWS Lambda?

Lambda is an Amazon compute service which allows you to run code in the AWS Cloud without managing servers.

37) Name the types of AMI provided by AWS

The types of AMI provided by AWS are:

Instance store backed

EBS backed

38) Name the AWS service that exists only to redundantly cache data and images?

AWS Edge locations are services that redundantly cache data and images.

39) Explain Geo Restriction in CloudFront

A Geo-restriction feature helps you to prevent users of specific geographic locations from accessing content which you’re distributing through a CloudFront web distribution.

40) What is Amazon EMR?

EMR is a survived cluster stage which helps you to interpret the working of data structures before the intimation. Apache Hadoop and Apache Spark on the Amazon Web Services help you to investigate a large amount of data. You can prepare data for the analytics goals and marketing intellect workloads using Apache Hive and using other relevant open-source designs.

41) What is the boot time taken for the instance stored backed AMI?

The boot time for an Amazon instance store-backend AMI is less than 5 minutes.

42) Do you need an internet gateway to use peering connections?

Yes, the Internet gateway is needed to use VPC (virtual private cloud peering) connections.

43) How to connect EBS volume to multiple instances?

We can’t be able to connect EBS volume to multiple instances. However, you can connect various EBS Volumes to a single instance.

44) List different types of cloud services

Various types of cloud services are:

1. Software as a Service (SaaS),
2. Data as a Service (DaaS)
3. Platform as a Service (PaaS)
4. Infrastructure as a Service (IaaS).

45) State the difference between An Instance and AMI

AMI is a template consisting of software configuration part. For example Operating systems, applications, application servers if you start an instance, a duplicate of the AMI in a row as an attendant in the cloud.

46) What are the different types of Load Balancers in AWS services?

Two types of Load balancers are:

1. Application Load Balancer
2. Classic Load Balancer

47) In which situation you will select provisioned IOPS over Standard RDS storage?

You should select provisioned IOPS storage over standard RDS storage if you want to perform batch-related workloads.

48) What are the important features of Amazon cloud search?

Important features of the Amazon cloud are:

Boolean searches

Prefix Searches

Range searches

Entire text search

AutoComplete advice

49) Can vertically scaling is allowed in Amazon Instance?

Yes, you can vertically estimate one Amazon instance.

50) What is the use of lifecycle hooks in Autoscaling?

Lifecycle hooks are used for autoscaling to put an additional wait time to a scale in or scale out event.

51) What are the various layers of Cloud Architecture explained in AWS training?

Different layers of cloud architecture are:

1. Cloud controller
2. Cluster controller
3. Storage Controller
4. Node Controller

52) What are the storage class available in Amazon s3?

Storage classes available with Amazon s3 are:

1. Amazon S3 standard
2. Amazon S3 standard-infrequent Access
3. Amazon S3 Reduced Redundancy Storage
4. Amazon Glacier

53) Name some of the DB engines which can be used in AWS RDS

1. MS-SQL DB
2. MariaDB
3. MYSQL DB
4. OracleDB
5. PostgreDB