1. What is Docker, and how does it differ from virtual machines?

Docker is a containerization platform that packages applications and their dependencies into containers, ensuring consistent functionality across environments.

Differences:-

-> Docker containers share the host OS kernel, making them lightweight, while VMs have their own OS, making them resource-heavy.

-> Containers start in seconds, whereas VMs can take minutes.

2. What are the key components of Docker architecture?

Docker Client: The user interface to interact with Docker (e.g., docker commands).

Docker Daemon (dockerd): Runs on the host machine and manages containers.

Docker Images: Read-only templates used to create containers.

Docker Registry: Stores Docker images (e.g., Docker Hub).

Docker Containers: Runtime instances of images

3. How is a Docker image different from a container?

Docker Image: A blueprint or template used to create containers.

Docker Container: A running instance of a Docker image, with its own file system, network, and resources.

4. Explain Docker’s Union File System (UnionFS).

Answer: Docker uses UnionFS to stack multiple layers of images, making them lightweight and fast. UnionFS allows Docker to:

Use layers efficiently by reusing unchanged layers across images.

Make images immutable, ensuring consistency.

5. How do you persist data in Docker containers?

Answer: Use Docker Volumes or Bind Mounts:

Volumes: Managed by Docker and stored under /var/lib/docker/volumes/. They are more portable and secure.

Bind Mounts: Maps host files or directories to containers. It’s less portable and relies on host paths.

example:-

docker run -v /host/path:/container/path myimage

6. What are multi-stage builds in Docker, and why are they used?

Multi-stage builds allow you to use multiple FROM instructions in a single Dockerfile. They are used to:

Minimize image size by copying only the necessary artifacts into the final image.

Separate build and runtime environments.

7. How do you troubleshoot a container that is not starting?

Use docker logs <container\_id> to view logs.

Inspect the container with docker inspect <container\_id>.

Check the exit code using docker ps -a and docker inspect.

Verify network issues, resource limits, or entrypoint commands.

8. How does Docker manage networking?

Answer: Docker provides the following network modes:

Bridge: Default mode; containers share a virtual bridge.

Host: Binds the container to the host’s network stack.

None: Disables networking for the container.

Custom Networks: User-defined networks with custom subnets, IPs, etc.

commands:-

docker network ls

docker network create my\_network

docker run --network=my\_network my\_image

9.What is the difference between ENTRYPOINT and CMD in Docker?

ENTRYPOINT: Defines the container's executable that cannot be overridden during docker run.

CMD: Provides default arguments to the ENTRYPOINT or acts as the main command if ENTRYPOINT is absent.

Example:

ENTRYPOINT ["python"]

CMD ["app.py"]

10. Explain the concept of namespaces and cgroups in Docker.

Answer:Namespaces: Isolate resources (process IDs, file systems, etc.) for containers, ensuring each container runs independently.

Cgroups: Manage resource allocation (CPU, memory) to prevent containers from over-consuming host resources.

KUBERNETES

1. What is Kubernetes, and why is it used?

Answer:Kubernetes is an open-source container orchestration platform that automates the deployment, scaling, and management of containerized applications.

Why Use Kubernetes?

Automates container scheduling and scaling.

Provides self-healing capabilities (e.g., restarts failed containers).

Manages service discovery and load balancing.

Ensures efficient resource utilization.

2. What are the main components of Kubernetes architecture?

Answer:Master Node:

API Server: Serves as the front-end for Kubernetes.

Controller Manager: Manages controllers (e.g., replication controller).

Scheduler: Assigns pods to nodes.

etcd: Stores cluster state.

Worker Node:

Kubelet: Communicates with the API server and manages pods on the node.

Kube-proxy: Handles networking and load balancing for pods.

Container Runtime: Executes containers (e.g., Docker, containerd).

3. What is a Pod in Kubernetes?

Answer:A Pod is the smallest deployable unit in Kubernetes. It encapsulates:

One or more containers.

Storage resources (volumes).

Unique network IP.

Configuration options for containerized applications.

4. What is the difference between a Deployment and a StatefulSet?

Answer:Deployment: Used for stateless applications. It provides features like rolling updates and rollback.

StatefulSet: Used for stateful applications requiring stable identities and persistent storage (e.g., databases).

5. What are ConfigMaps and Secrets?

ConfigMaps: Store non-sensitive configuration data (e.g., application properties).

Secrets: Store sensitive data (e.g., passwords, API tokens). Secrets are encoded in Base64 for security.

6. How does Kubernetes perform auto-scaling?

Answer: Kubernetes supports three types of scaling:

Horizontal Pod Autoscaler (HPA): Adjusts the number of pods based on CPU/memory usage or custom metrics.

Vertical Pod Autoscaler (VPA): Adjusts pod resource requests/limits.

Cluster Autoscaler: Scales the number of nodes in a cluster based on pending pods

7. What are Persistent Volumes (PVs) and Persistent Volume Claims (PVCs)?

PV: A storage resource provisioned by the admin.

PVC: A user's request for storage, bound to a PV.

8. How do you debug a failing pod?

Answer:Check pod status: kubectl get pods.

View logs: kubectl logs <pod-name>.

Describe the pod: kubectl describe pod <pod-name>.

Exec into the container for debugging: kubectl exec -it <pod-name> -- /bin/bash.

Inspect events in the namespace: kubectl get events.

9.How does Kubernetes ensure high availability?

Answer:

Master Node HA: Run multiple master nodes (control plane components) behind a load balancer.

Worker Node HA: Spread pods across multiple nodes using labels and affinity rules.

Self-Healing: Restarts failed pods automatically.

10.How does Kubernetes handle networking?

Answer: Kubernetes uses a flat networking model where all pods can communicate with each other without NAT.

CNI Plugins: Tools like Calico, Flannel, and WeaveNet implement networking in Kubernetes.

Service Types:

ClusterIP: Default, accessible only within the cluster.

NodePort: Exposes a service on a static port of each node.

LoadBalancer: Integrates with cloud providers for external access.

11.How do you secure a Kubernetes cluster?

Answer:

Use Role-Based Access Control (RBAC).

Enable pod security policies.

Use network policies for isolation.

Secure etcd with encryption and TLS.

Scan container images for vulnerabilities.

Use namespaces for multi-tenant isolation.

12.What are DaemonSets, and how are they used?

Answer:

A DaemonSet ensures that a copy of a specific pod runs on all (or some) nodes in a cluster. Common use cases include:

Running logging agents.

Monitoring tools (e.g., Prometheus Node Exporter).

JENKINS

1. What is Jenkins, and why is it used?

A.Jenkins is an open-source automation server used for continuous integration and continuous delivery (CI/CD).

Key Features:

Automates builds, testing, and deployments.

Extensible with plugins for various tools (e.g., Git, Docker, Kubernetes).

Supports pipelines for complex workflows.

2. What are the key components of Jenkins architecture?

Answer:

Jenkins Master:

Manages the configuration of jobs, plugins, and scheduling.

Delegates tasks to agents.

Jenkins Agent (Slave):

Executes jobs assigned by the master.

Installed on remote or local machines.

3. What is a Jenkins pipeline?

Answer:

A Jenkins pipeline is a set of steps defined in code to automate CI/CD processes.

Declarative Syntax: Simple and predefined structure.

Scripted Syntax: Flexible and complex scripting.

4. What are Jenkins plugins, and why are they important?

A.Plugins extend Jenkins' functionality.

Examples:

Git Plugin: Source code management.

Pipeline Plugin: Create CI/CD workflows.

Docker Plugin: Build and run containers.

5. How do you schedule a Jenkins job?

Answer:

Use the Build Triggers section and select "Build periodically."

Use CRON syntax for scheduling:

Example: H/5 \* \* \* \* (Run every 5 minutes).

6. What is Blue Ocean in Jenkins?

Answer:

Blue Ocean is a modern interface for Jenkins pipelines that provides:

Visualized pipelines.

Intuitive UI for configuring and managing jobs.

7. How do you secure Jenkins?

Answer:

Enable Role-Based Access Control (RBAC) using the Role Strategy plugin.

Use SSL for secure communication.

Regularly update Jenkins and plugins.

Disable anonymous access.

Integrate with external authentication (e.g., LDAP).

8.How do you manage secrets in Jenkins?

Answer:

Use the Credentials Plugin.

Store credentials (username, password, API keys).

Reference credentials in pipelines

9. How do you monitor Jenkins performance?

Answer:

Use the Monitoring Plugin or tools like Prometheus.

Enable logging and audit trails.

Monitor system resources (CPU, memory).

10. How do you handle failed builds in Jenkins?

Answer:

Configure Post-build Actions (e.g., email notifications).

Use retry logic in pipelines:

groovy

Copy

Edit

retry(3) {

sh 'build-command

}

AWS

1. If user’s can’t access an application hosted on ec2,what steps have to taken?
2. 1.Check security group 2.Verify Instance health
3. Network Configuration 4.Logs and Monitoring 5.DNS settings

2.DIfference between Elastic LoadBalancer and Autoscaling?

1. ELB:ELB automatically distributes incoming application traffic across multiple target’s

such as EC2 instance,IP address,containers to ensure no single resource is overwhelmed.

Autoscaling:Scalein and scaleout based on the load.

1. If I lost my pemfile how can I connect my ec2 instance?
2. Use AWS Systems Manager session manager

Ensure that your instance has an IAM role attached with permissions for SSM.

This method doesn’t require SSH access and is very secure.

Use EC2 Connect

Allows you to connect to your instance without needing the original PEM file,provided by that the instance configured correctly.

Instance have public IPV4 address and that EC2 instance connect is supported for your Instance.

This method temporarily pushes a public key to the Instance,allowing you to connect without the PEM file.

1. Difference between AMI,EBS volumes and EBS snapshots?
2. AMI: An AMI is a template that includes operating system,application server and application necessary to launch EC2 Instance.

EBS Volume:EBS Volume is a durable,block-level storage device that can be attached to EC2 Instance.

EBS Snapshot:An EBS Snapshot is a point in time backup of an EBS volume.It captures all the data on the volume at the moment snapshot is taken.

1. When will happen statefile locking in terraform?How will you rectify it?
2. Statefile locking in terraform prevents multiple users or process from simultaneously accessing the same statefile.This ensures that only one user or process can make a change at a time.

Terraform uses remote backends like AWS S3,Google Cloud Storage or Azure Blob Storage to manage state locks.

1. Difference between Terraform and Cloudformation?
2. Terraform
3. Supports multiple cloud providers and on-premises environments.
4. Ues HCL language.
5. Supports wide range of third-party providers and modules.
6. Allows strong statefiles externally in various backends like S3 or Consul.

Cloudformation

1. EXclusively focus on managing AWS resources.
2. Supports JSON and YAML formats for templates.
3. Integrates deeply with AWS services and resources.
4. Manages state of stack internally,necessitating custom sharing solutions.

1. What are components of ECS?
2. Cluster:A logical grouping of container instances where tasks and services run.
3. Task Definition:A blue print for your application.
4. Task:An instantiation of a task definition running on a container on a container instance.

Tasks can be run as standalone entities or as part of a service.

1. Service:Ensures that a specified number of task instances are running simultaneously within a cluster.Sevices can also manage loadbalancing and scaling.
2. Explain IAM role,policy and group?

IAM role:A role is a set of permissions that define what actions are allowing on specific AWS resources.Temporarily access without using credentials.

IAM policy:Policies are JSON documents that define permissions associated with IAM identities.

IAM group:An IAM group is a collection of IAM users.

1. How would you monitor AWS infrastructure and troubleshoot performance issues?
2. 1.Amazon Cloudwatch 2.Cloudtrail 3.Config 4.GuardDuty these are cloud native tools.

Third-party tools

1.Datadog 2.Dynatrace 3.signoz

1. What is meant by Drift?Common causes of Drift?
2. Drift happens when the actual state of your infrastructure no longer matches the desired state defined in your terraform files.

Common cause of drift

1. manual changes in the cloud console or cli.
2. External tools/scripts modifying the resource
3. Statefile issues like lost or outdated.
4. What is vpc,and why it is uesd ?
5. vpc is dedicated to your aws account.vpc are used to host resources without exposing them to the public internet.

Why it is used

1. security 2.Scalability 3.Control 4.Disaster recovery.
2. What is NAT Gateway and how it is differ from NAT instance?

NAT Gateway

1. fully managed by AWS.
2. High availability across Azs.
3. No security groups needed.
4. Hourly based cost+data processing fee.
5. Bandwidth upto 100gbps.

NAT Instance

1. It is self managed.
2. Requires manual setup.
3. Requires SG setup.
4. Cheaer but requires management overhead.
5. Depends on instance type.
6. What is aws privatelink and how is differ from vpc peering?
7. AWS privatelink allows private connnectivity between vpcs and aws services without exposing to the public internet.It is unidirectional.Uses interface endpoints.

vpc peering establishes a direct ,bidirectional connection between two vpcs,allowing resources in both vpcs to communicate with one another as if they were part of the same network.It allows direct communication between vpcs.

1. define vpc endpoints ? Explain types of vpc endpoints?
2. vpc endpoints are used to allow vpc to other aws services.

There are mainly two types of vpc endpoints.

Interface Endpoint:These are powered by aws privatelink and allow connections to services over private IP addresses.

Gateway Endpoint:These are specially designed for S3 and DynamoDB.Gateway endpoints allow you to route traffic directly to these services without needing a NAT device or public IP addresses.

1. If you have 100 vpcs,how would you enable communication between all of them?
2. AWS transit gateway is helpful for connecting multiple vpcs in different AWS regions.

It reduces operational overhead,and allow for scalability.

1. What is vpc flow logs, and how do they help with monitoring?
2. vpc flow logs is a feature in aws that allows you to capture and log detailed information about IP traffic going to and from network within your vpc.This feature is essential for monitoring troubleshooting and secure your network environment.

Benefits of using vpc flow logs for monitoring

1. Traffic analysis 2.security Monitoring 3.Performance Monitoring 4.Compliance and Auditing 5.Troubleshooting connectivity issues
2. Define Route53?What are types of records supported by Route53?
3. Route53 is a highly scalable and available Domain Name System(DNS) service that translates human readable domain names into machine readable IP addresses.

Records supported by Route53

1. A record 2.AAAA record 3.CNAME record
2. MX record 5.NS record 6.Alias record
3. What is hosted Zone in Route53?What are the different types of routing policies?
4. A Hosted Zone is a container for DNS records for a specific domain.It can be public or private.

Routing policies

1. Simple routing 2.Failover routing 3.Weighted routing
2. Geographic routing 5.Latency-based routing 6.IP-based routing
3. What is the difference between an alias record and a CNAME record in Route53?
4. An alias record can point to aws resources like cloudfront distributions or load balancers without requiring a separate IP address,While a CNAME record points to another domain name.
5. A client needs backup website in case primary site goes down.How would you configure Route53 to backup website?
6. Configure failover routing policies in Route53,setting the primary site with a health check.If the health check fails,traffic will automatically redirect to the backup site.
7. You are running a web application with spiky traffic,and you need DNS to support load balancing across multiple endpoints.How would you configure Route53?
8. Use weighted routing policy or latency-based routing combined with health checks to distribute traffic across multiple endpoints,ensuring high availability during peak times.
9. You have multiple services under same domain, and each domain is hosted in a different AWS region.How would you route traffic to the correct service based on user’s location?
10. Use geolocation routing policies in Route53 that direct users to specific endpoints based on their geographic loaction,ensuring that they reach the nearest service.
11. How would you ensure high availability and disaster recovery in AWS?
12. High Availability:High Availability refers to the ability of a system to remain operational and accessible despite of failures or faults.
13. Design for no single point of failure 2.Autoscaling 3.Use Multiple availability zone
14. Health Monitoring 5.Decoupling 6.Managed services

DR: Focuses on restoring services after a major failure or disruption.

1. Backup and restore 2.Multi-Region Deployment 3.Regular Testing
2. Documentation and automation 5.Pilot Light Approach 6.Warm Stand by
3. How would you manage and automate aws infrastructure using cloudformation?
4. IAC service that allows you to define and provisioning aws resources using templates written in JSON or YAML.This approach automates creation,updation and managing your infrastructure,making it easier to maintain and reduce manual errors.

Key Components

1.Templates:These are text files that describe the resources needed for your application including their configurations and dependencies.

1. Stack:Stack is a collection of aws resources that you can manage as a single unit.You can create,update or delete stack based on the defined template.
2. Resources:These can include various aws services like EC2,VPC,SG etc.which are defined in your template.
3. Difference between Kubernetes and Amazon ECS?
4. Kubernetes:
5. Application can be deployed using a combination of pods,nodes and services.
6. Deployment is complex.
7. Two kinds of health checks liveliness and readiness.
8. Muti-cloud support.

Amazon ECS

1. Application can be deployed as tasks,that are docker containers running on EC2 instance.
2. Deployment is hard.
3. Health checks using cloudwatch.
4. Cloud natively support.
5. Define S3 and what are different types of S3 storage classes?
6. S3 is a scalable,high-speed,web-based cloud storage offered by AWS.It provides object storage through a web service interface,following users to store and retrieve any amount of data from anywhere on the internet.

S3 storage classes

1. Standard frequent access 2.Standard Infrequent access 3.Intelligent tier
2. One-zone IA 5.Archive and Deep Archive 6.Glacier
3. Explain difference between single upload and S3-multipart upload?
4. A single upload,also known as standard upload,involves sending an entire file as one complete request using the put operation.

You can upload files upto 5GB in size in a single PUT operation.

Best suited for smaller files <5GB where upload reliability is not a concern.

Multi-part upload allows you to upload a single object as a set of parts.Each part can be uploaded independently and in any order.

The minimum size for an individual object is 5TB,making multipart uploads necessary for files larger than 5GB.Each part must be least 5MB.

1. What is the purpose of S3 object lock?How it is differ from versioning?  
   A.S3 object lock is a feature designed to protect objects stored in S3 by preventing them from being deleted or overwritten for a specified retention period.It uses WORM model,ensuring that once data is written,it cannot be modified or deleted until the retention period expires.

By preventing accidental deletions or overwrites,object serves as additional layer of critical backups and sensitive information.

Versioning allows multiple versions of an object to exits in the same bucket.When an object is updated or deleted, anew versioning is created instead of overwritten the existing one.

It is primarily used to recover previous versions of objects in case of accidental deletions or modification.

1. How do you access an S3 bucket from account A in account B’s AWS account?
2. Cross account Management.
3. A mobile application serves scientific articles from an S3 bucket.Articles older than 30 days are rarely read,and those older than 60 days no longer need to be available through the application but should be kept for historical purposes.What is the best cost-effective solution?
4. Implement an S3 Life-cycle policy to transition articles older than 30 days to the S3 standard-IA class and those older than 60 days to S3 glacier,ensuring cost efficiency while retaining historical data.
5. A bank’s fraud detection team uses S3 to store historical transcational data and runs analytic using Redshift spectrum.They want to trigger realtime fraud detection process when new data is upload.How can they achieve this?
6. Set up an S3 event notifications to trigger an aws lambda function whenever new data is uploaded to the S3 bucket,allowing for real-time processing and fraud detection without manual intervention.
7. What is aws lambda?Benefits of using aws lambda?  
   A.AWS lambda is a serverless compute service that allows developers to run code without managing servers.It’s a function as a service(faas) offerded by AWS.

Benefits of using aws lambda

1. No servers to manage 2.automatic scaling 3.Cost Efficiency
2. Increased innovation 5.Integration with AWS services 6.Built in fault tolerance.
3. How can you monitor AWS lambda functions?
4. Monitoring AWS lambda functions is a crucial for maintaining performance,identifying issues,and optimizing resource usage.Methods and tools for effectively monitoring AWS lambda.
5. Cloudwatch 2.X-rays 3.Lambda insights 4.Third party tools 5.Cloudtrail
6. How would you use AWS lambda to delete old logs from an S3 bucket everyday?
7. Create a cloud watch event rule to trigger the lambda function everyday.

Use boto3 to list objects in the S3 bucket.

Delete objects older than a certain date.

1. How do you deploy AWS lambda using CI/CD pipelines?
2. AWS codepipeline+codebuild=Automates lambda deployments.

Serverless framework-simplifies deployments with yaml-based configurations.

AWS SAM(Serverless Aplication Model)-Uses sam build and sam deploy.

Terraform/cloud formation-IAC for automated deployments.

1. How would you process large files efficiently using lambda?
2. Lambda has a timeout limit of 15 minutes,so processing large files requires.

S3 event triggers-triggers lambda on file upload.

Parallel processing-split the files and process in minutes lambda executions.

AWS step functions-Break large processing tasks into smaller steps.

Use external storage-store intermediate results in DynamoDB,S3 or EFS.

1. You need to automatically resize an image whenever a user uploads it to an S3 bucket.How would you implement this using AWS lambda?
2. Setup an S3 event trigger to invoke the lambda function when a new image is uploaded.

Use AWS SDK(Boto3 in python)to read uploaded image.

Utilize python or Node.js to resize the image.

Save the resized image back to the S3 bucket.

1. What are inodes in Linux?
2. An inode is a data-structure in unix-style file systems that store metadata about files and directories.Each file and directory is allocated a unique i node,which is identified by an inode number.

Inodes do not store the file name or actual file content.

Each inode is unique within its filesystem,through inode numbers can be reused across different filesystems.

1. Define top and htop in linux?
2. The top command is powerful linux utility for real-time system monitoring.It provides a dynamic interactive view of running processes and system resource utilization.

Key features

1. CPU usage 2.Memory consumption 3.Running processes 4.System load

Htop is an advanced,interactive process viewer for linux that enhances the traditional top command with powerful monitoring capabilities.

Real-time system resource monitoring color-coded interface

Interactive process management Interactive process management

1. Define df-h and free-m?
2. df-h is a command in linux displays disk usage in a human readable format.

It shows files system disk free space information.

Presents in sizes in:GB,MB,KB.

Makes disk space information easily understandable.

Free-m command in linux displays system memory usage statistics in megabytes(MB).It provides a quick overview of memory allocation across different categories.

Purpose:Shows memory utilization information.