

DON'T JUST DEPEND ON THESE QUESTIONS ONLY, DO MORE PRACTICALS, SO THAT YOU CAN ANSWER ANY QUESTIONS

SAIKIRAN PINAPATHRUNI: www.youtube.com/@SaiKirannPinapathruni

Interviewer : You lost both PEM and PPK file how will you try to connect to your Instance ?

Answer :-

Create a role > select Ec2 > search for “managed instance core” select it > again search for “ssmf “ > and click on create a role

Attach the iam role to the instances you want to login

Got to management and governance > click on “AWS systems manager” > under drop down select “Sessions Manager ” > start session > here you will see your server > Select the server you wanted to login and start session.

Run the below commands

- Bash
- Sudo su –
- Nano etc/ssh/sshd_config
- Come down and search for “password authentication disabled “
- change it to "PasswordAuthentication yes
- Save and Exit
- sudo service ssh restart
- Set a password for the user you want to use to log in: sudo passwd username

Connect to the instance using SSH: ssh username@instance_ip. When prompted, enter the password you set for the user.

Connect / Subscribe : <https://lnkd.in/gE5kc3T8>

Interview question:

My client has a highly sensitive application deployed in an Amazon [hashtag#VPC](#) (Virtual Private Cloud) within their AWS account. The application needs to securely access AWS services such as Amazon S3 and Amazon DynamoDB without the public internet. Additionally, these services are restricted to only the resources within their VPC. How will you achieve this, and could you explain step by step?

VPC endpoints will allow the servers inside the VPC to communicate with other AWS services internally without going through the internet. We have two types of endpoints:

Gateway endpoints and Interface Endpoints.

Gateway endpoints are used for S3 and DynamoDB, while Interface endpoints will create a network interface on a corresponding subnet. By implementing VPC Endpoints, we can avoid the cost of NAT gateway which is \$0.045/hour and S3 \$0.045/GB, along with the data transmission prices.

Interviewer :

How can you diagnose latency between AWS resources?

Initially il check with ping command followed by the IP address or hostname of the target resource to check connectivity and measure round-trip latency.

ping IP_address_or_hostname

Interviewer : What was that round trip ? How would you measure ?

When you use the ping command in Linux, it sends a small packet of data to the target server and waits for a response. The time it takes for the packet to travel to the server and back to your computer is the round trip latency.

Interviewer: Fair enough, please proceed.

Il also use the traceroute command followed by the IP address or hostname of the target resource to trace the path packets from your machine to the target, showing latency at each hop.

traceroute IP_address or hostname

Netcat Command, il Use the nc command to test network connectivity and measure latency between two machines by establishing a TCP connection.

nc -z IP_address or hostname port_number

Interviewer what is -z ?

So when you use -z you're telling nc to check if the specified port on the target machine is open or reachable. If the port is open, you'll typically receive a response indicating that it's open, and if it's closed, you won't receive any response.

Subscribe : <https://lnkd.in/g2cDJnRU>

Interviewer: I'll only ask one simple question, can you walk me through it?

Yes :)

How do your VPCs communicate in your environment?

So basically, we have multiple clouds, and we follow multiple environments. Our VPCs were deployed in multiple regions.

Previously, we followed a peering strategy but found that we were creating a lot of peering connections. Unfortunately, we found it was cost-effective during analysis and decided to move to TSG (Transit Gateway). We use three different accounts for three different environments, so all these VPCs will be managed with the TSG.

Interviewer: VPC Peering also does the same? Correct?

Correct. But it needs a lot of peering connections to be created regardless of different regions. However, TSG acts like a hub between these VPCs.

Interviewer: So it means it acts as a VPN connection ?

No, instead, it facilitates communication between multiple VPCs, VPN connections, and Direct Connect connections.

Interviewer: Can you please walk me through the steps?

So we follow this entire process with Terraform and...

Interviewer: Before deploying it with Terraform, I believe you are well aware of manual steps:)

Please watch the video below video for detailed steps : <https://lnkd.in/g3BydQtg>

Interviewer : How good you are with the Instances and Cost optimizations ?

i can rate myself as 3.5 out of 5

Interviewer : So We are using On demand EC2 instance (m4.xLarge Windows) in our environment any other instance you suggest ?

If we knew the project term is more than 3 years, Then its not good to use EC2 instances. Instead you can go with reserved instance.

Interviewer : I interviewed 6 members before you, no one suggested Reserved instances, But am Impressed to know why you suggest Reserved Instance ?

Example Ec2 - Windows m4.xLarge charge 0.386\$ for hour Roughly on a year Its somewhere around 3k-\$ But if you knew that your project is long term its a best decision to move with Reserved cuz in reserved instance charges 21% less on Long-term orders.

Interviewer : Am gonna ask you one last question, Il select you if you answer.

:) okay .

Interviewer : So we are using SPOT instance for our DB what are the better recommendations you do?

its not good to keep DB servers in spot instances but its okay keep our web Servers or for testing purpose.

Interviewer : why ? Spot are very cheap right ?

Yes , But AWS Terminates our instance with 2 minutes Notice period if the ON-DEMAND EC2 Traffic is high. So it snot good to keep the DB servers in SPOT Instance, But for Practice or testing . Yes we Can .

Detailed video on Ec2 is here : <https://lnkd.in/gJJWB8z8>

Subscribe to my channel Please :)

<https://lnkd.in/gJJWB8z8>

Interviewer: So, we need an instance on a daily basis just for testing and development environments. What do you suggest? An instance daily? If yes, what instance? Could you please walk me through it?

Creating an instance on a daily basis solely for development and testing purposes might not be necessary. Instead, it's advisable to deploy an AMI (Amazon Machine Image) for efficiency and consistency. This allows you to replicate a pre-configured environment easily.

Interviewer: Do we have to manually install approximately 6-7 including some networking tools in the instance?

No, we can streamline the process using HashiCorp's tool called Packer. It automates the entire process of creating AMIs, allowing us to specify the tools we need to install. We can define the installation steps using JSON format scripts and even incorporate shell scripts to complete our configurations seamlessly. This ensures consistency and efficiency in our environment setup

Subscribe Below for more :)

<https://lnkd.in/gcV92A-7>

Interviewer: In resume you have you have mentioned you are worked on S3, Correct ? See i might stop you in the middle if i wanted to ask anything. okay ?

Yes, Okay :)

interviewer : Can explain the storage types and advantages and disadvantages ?

So basically we do have 3 storages in AWS

Block Storage > File Storage > Object Storage .

Block Storage does contains Instance storage and EBS, But instance storage was not a good option unless its something personal work or for testing, Cuz the data saved inside the instance will be gone post restarting the server . But EBS is like a persistence storage Can be increased to 16TB and Available in different types SSD - HDD . .

interviewer : Sorry interrupting, What were the types ?

SSD: gp2,Gp3,io1,io2

HDD: SC1 , SCT, Standard., We can attach and detach to other instance also .

Interviewer : Fair enough!!

So another one is file storage it contains AWS-EFS and AWS FSX which is mainly used for Linux and windows based operating systems .

and last we do have Object storage, which is S3 simple storage service which stores all unstructured data and Glacier stores historical data .

Interviewer: Il ask you one last question, I want to deploy a High-performance MS SQL Db server? What do you suggest?

^-^

So for Operating system , Post installation only read memory is required so i suggest to go with GP2 and GP3.

For Media and Apps HDD-ST1 is required, which gives a minimal data retrieval speed.

For Virtual memory / Page files or for TEMP-DB il use Instance Storage. However we do have backup for rest, So i believe it will not hamper anything.

Interviewer : What if i need 50TB hard disk?

For logs il allocate 20TB and for backup il allocate 30TB.

Interviewer : Thanks :)

Subscribe to my channel for more :)

https://lnkd.in/gda_nhRD

Interviewer: Sorry for not starting with an introduction, just want to jump into technical stuff without wasting time. 😊

Me: It's okay. 😊

So, I need 3 subnets, private only, from NV AZ 1A 1B 1C. When someone tries to access my web application, it should communicate in a secure way. You have 1 hour 10 minutes; can you please configure using only AWS services? our machine in front you which does have Aws access .

Yes, I can do that !!

Interviewer: Before jumping onto it, can you illustrate the entire structure.

Imagine it's a three-tier architecture: Web - App - Db.

First, I'll create a VPC with 3 private subnets and 3 public subnets. Am not going to keep anything in public subnets, instead il use it for NATs and NLB's and make sure all VPC components are configured appropriately just to avoid the miss communication between private and NAT :)

Post that, with the help of ACM Amazon Certificate Manager, I'll make sure my domain is updated with SSL for secure communication.

Then, I'll create target groups and select only private servers. Post that , I'll create a load balancer and make sure I'm selecting public subnets for it., I'll also select TLS under listeners with our ACM which were initially issued by AWS. Once done, I'll configure Route53 for domain forwarding.

Interviewer: Impressive!! Your plan looks good overall

Please do subscribe my YouTube channel here —!!

<https://lnkd.in/gW-7M9ti>

Interviewer : What do you know about load balancers ?

Load balancers are of two types , network and application **XXXXXXXXXXXX**

We do have two load balancers types , regional load balancer and VPC based load balancers .

In region we have global accelerator and in VPC we do have Network load balancer and application load balancer . The major difference between Network load balancer and application load balancer is application load balancer supports only http and https and network load balancer support tcp and udp protocol. Even http and https are tcp protocols but application load balancer doesn't support UDP .
udp is mainly used for streaming or gaming application or online application. that is not supported by ALB this is the major difference between these two.

Interviewer : Fair enough, Can you walk me through steps as per my requirement.

So, I need 3 subnets, private only, from NV AZ 1A 1B 1C. When someone tries to access my web application, it should communicate in a secure way.

Imagine it's a three-tier architecture: Web - App - Db.

First, I'll create a VPC with 3 private subnets and 3 public subnets. Am not going to keep anything in public subnets, instead il use it for NATs and NLB's and make sure all VPC components are configured appropriately just to avoid the miss communication between private and NAT :)

Post that, with the help of ACM Amazon Certificate Manager, I'll make sure my domain is updated with SSL for secure

communication.

Then, I'll create target groups and select only private servers. Post that , I'll create a load balancer and make sure I'm selecting public subnets for it., I'll also select TLS under listeners with our ACM which were initially issued by AWS. Once done, I'll configure Route53 for domain forwarding.

Interviewer: Impressive!!

Please do subscribe my YouTube channel here and watch the detailed video on what interviewer asked.

<https://lnkd.in/gEgqftTk>

Interviewer: Can you elaborate on the variances you've observed between a Network Load Balancer and an Application Load Balancer?

Network Load balancer doesn't supports HTTP and HTTPS redirection, also it doesn't supports URL/PATH based routing and it cannot be used as an Ingress controller for Kubernetes. Ultimately WAF was also not supported with NLB.

Interviewer : Okay, Imagine you've created a load balancer, What was the very next step you do ?

I will create a record in route53 with our domain using associate to ALB.

Interviewer : Great! Now explain. How does the packet retrieves the information when a user hits our domain name?

Sure, Route 53 resolves the domain to the public IP address associated with the ALB and the packet is then routed to the ALB in one of the public subnets. The ALB serves as the entry point for incoming traffic and then ALB forwards the packet to one of the instances registered with the associated target group. This occurs within the private subnets . Finally, the packet reaches the specific instance within the private subnet that is determined by the ALB's routing algorithm and the health of the instances in the target group.

Interviewer : How do you ensure seamless integration between Autoscaling and other AWS services, such as Launch Templates, Target Groups, Load Balancers, and Route 53, to guarantee optimal performance and scalability within the AWS environment?

Are there any prerequisites or specific application requirements that need consideration before selecting the OS IMAGE during Launch Template setup to avoid deployment failures when configuring the Autoscaling Group?

Interviewer : Imagine Its a spring boot application. . :)

Okay, For the traffic across healthy instances within your Auto Scaling group we required target groups and Load balancers. While we dont involve the ALB DNS directly to the public endpoints so we can follow with Route53 records.

Interviewer : So, is that you who watch whole environment ?

Am sorry its good to create SNS and Cloud watch metrics for visualizations of our metrics and logs.

Interviewer : Fair Enough!!

Just shared a real-time demo of Autoscaling on my youtube <https://lnkd.in/g98Bx-r2>

In the video, I guide you through the step-by-step setup process using Launch Templates, Target Groups, Load Balancers (LB), Route 53 (R53), and more. Also I explained how to implement Simple Notification Services (SNS), Ansible Playbooks, and set up alarms for Dynamic Scaling.

Title: Set up NGINX Server on AWS Test Environment

Description: Deploy an NGINX server for the test environment using our AMI. The server should display HTML content on the website with the domain name cloudvishwakarma.in, utilizing subdomains accessible in AP-MUMBAI-1 and US-EAST-1 regions. Ensure HTTPS is enabled using Let's Encrypt. Implement highly available and scalable web servers using ROUTE53 policies.

Configure the EC2 instance with a size of T3-nano and a 20GB disk in the eu-west-2 region. Allow public HTTP/HTTPS access without using a load balancer.

Tasks:

1. Provision an EC2 instance using our AMI in the AP-MUMBAI-1 region with the following specifications:
 - Instance type: T2.nano
 - Disk size: 20GB
2. Install NGINX and required packages on the EC2 instance according to the provided guidelines.
3. Set up HTTPS using Let's Encrypt Certbot for automatic redirection.
4. Create a custom AMI with the configured NGINX server and copy it to the AP-MUMBAI-1 and US-EAST-1 regions.
5. Deploy EC2 instances in AP-MUMBAI-1 and US-EAST-1 regions using the custom AMI.
6. Configure ROUTE53 health checks for the domain cloudvishwakarma.in with the following policies:
 - LATENCY
 - FAILOVER
 - WEIGHTED
7. Document the steps taken in the comments section for reference.
8. Once approved, proceed with the task.

Comments:

1. EC2 Instance Setup:
 - Created EC2 instance with T2.nano instance type and 20GB disk size in the eu-west-2 region.

2. NGINX Installation:

- Installed NGINX and required packages following the provided guidelines.

3. HTTPS Configuration:

- Set up HTTPS using Let's Encrypt Certbot for automatic redirection.

4. Custom AMI Creation:

- Created a custom AMI with the configured NGINX server and copied it to the AP-MUMBAI-1 and US-EAST-1 regions.

5. EC2 Deployment:

- Deployed EC2 instances in AP-MUMBAI-1 and US-EAST-1 regions using the custom AMI.

6. ROUTE53 Configuration:

- Configured ROUTE53 health checks for the domain cloudvishwakarma.in with LATENCY, FAILOVER, and WEIGHTED policies.

7. Steps documented for future reference.

8. Task ready for approval.

Please refer to the provided video for a detailed configuration guide. Don't forget to subscribe!

<https://lnkd.in/gUmpbHAU>

Interviewer: Hi let me ask you a scenario question , Even though we've set up firewalls, security groups, NACLs, and private subnets, and we're keeping our API and SSL updated, we're still worried about attackers. What else can we do to stay safe within AWS environment ?

Even with the utilization of AWS security groups or AWS Firewall, there are some security vulnerabilities which you cant stop Like SQL injection and Cross-Site Scripting attacks. Additionally, if there is a necessity to restrict access from specific IP addresses or block access to certain websites from particular countries or a whole country , Involving a Web Application Firewall (WAF) can be advantageous. Along with that we can also Secure our website with AWS Shield to defend DDOS Attacks.

Interviewer : :)

Watch Video below for More , Don't forget to subscribe !!

<https://lnkd.in/gTg5KMxk>

Instead of Introduction or introduce yourself, they might ask you like this!!

Interviewer : Hello, how are you doing?

I'm doing good, thanks.

Interviewer : Can you recall yourself what you did on April 16th after walking to your desk?

Sure. My day started with a heads-up call, where we discussed roadblocks and backlogs, set priorities, and ensured everyone was on the same page. After that, I received a task from a developer, which we track using Jira. Our developers are essentially my clients. Then, I monitored our 10 servers in the West and 10 servers in the East US using Prometheus to ensure everything was working fine. If any issues occurred within our range, we resolved them; otherwise, we escalated them to our on-site support team. Additionally, I wrote Dockerfiles as per our QA and development requirements. I also worked on a task in Groovy. Even if Sonar found vulnerabilities, the build stage passed and moved to the next stage, so I implemented quality gates to the existing Groovy script, ensured it was working as expected, and raised a PR. Apart from this, I conducted Level 1 interviews and conducted three tech rounds as scheduled before logging off from my desk.

Client: Thanks for detailed walk through !!

Interviewer : How to Copy a File to a Specific S3 Bucket from an EC2 Instance without Using Access Keys ?

1. Navigate to the IAM (Identity and Access Management) console.
2. Create a new IAM role. Avoid assigning the S3 full access policy; instead, opt for creating a custom role.
3. Within the role, under permissions, choose to add an inline policy.
4. Utilize the IAM custom Policy generator to create a policy as per your needs. Ensure that this policy is scoped specifically to the target S3 bucket using its Amazon Resource Name (ARN).
5. Once the policy is generated, apply it as an inline policy for the role.
6. Attach the newly created role to your EC2 instance.

Subscribe to my channel for more Realtime videos :

<https://lnkd.in/g2KUHNBA>

Interviewer : I'll ask you one simple question , Can you walk me through ?

Yes

Interviewer : explain the process of deploying an sample application using docker build image in AWS Elastic Beanstalk step by step ?

Sure

Initially we need to Install AWS CLI and Docker on your local machine / VM and have a sample application ready with a Dockerfile, Once done we run AWS configure for configuring access key and secret key or You can simply attach the role either way work fine!!

interviewer : Which one do you suggest and why ?

I suggest to go with attaching a role to instance , because End of the day access key and secret key will be saved under Meta-data in the instance.

Interviewer : Where ?

If you will find the keys along with the temporary token

interviewer : sounds good , please proceed.

Il make sure My sample application has a Dockerfile in its root directory, specifying the image and commands needed to run my application. Post that il Navigate to my application's directory in the terminal and run docker build -t dockerfilename:v1 . to build and tag the Docker image once it is completed, Il push it to my ECR on AWS.

Interviewer: Why ECR ? Why not dockerHUB

because when deploying an sample application on Elastic Beanstalk we need an prebuild image in AWS ECR, EBS doesn't support Dockerhub registry.

Interviewer : okay . .

Once done , Il route to Elastic beanstalk on AWS and Create Application and will provide the name and description of my application Post that il Click Create environment il select my Web server environment Select Docker as the platform and Upload my Docker image from ECR. Once done i will Set the necessary environment variables il Configure load balancer, scaling, and other settings as needed and il Deploy my sample application.

Once the deployment is successful, We can access our application using the provided URL, But i wanted to sync with route53 domain name with the provided DNS/URL .

Interviewer : Am impressed!!!!

Check out my mastering AWS playlist for more realistic content : <https://lnkd.in/ggCA95WG>

Interviewer: So, let's move into some Terraform questions. Imagine you have two developers deploying the same infrastructure on AWS. What's the challenge here?

With two developers working separately, there's a risk of duplicating resources, which could lead to unnecessary costs.

Interviewer: Absolutely! So, how would you avoid that duplication in Terraform?

One approach is to utilize Terraform's state file locking feature with DynamoDB. By configuring Terraform to use DynamoDB as a backend, it ensures that only one developer can make changes to the infrastructure at a time.

Interviewer: Good point! Anything else you'd consider?

Other way is by utilizing Terraform modules. By creating reusable modules for common infrastructure components, Both developers can use them instead of starting from scratch every time.

Interviewer: Great! It seems like you've got a good handle on avoiding multi-cost scenarios in Terraform deployments.

Scenario based question they might ask like below .

Interviewer: Hi there! Can you tell me about your experience with AWS account management?

I've worked with managing multiple AWS accounts using role switching and we also do have our AD on On-prem and Managed AWS AD both.

Interviewer: Interesting. Can you explain how you manage access between different accounts? How do you ensure security in this setup?

We have a setup with a Master account and two others: Staging and QA. We create roles in the Master account and grant access to Staging and QA from there. We're careful with permissions, only giving access to what's needed. Plus, we keep an eye on things with regular checks and security measures like multi-factor authentication.

Interviewer: And how do you set up trust between the Master and the other accounts?

We follow In-Line policies. We define who can access what and when, making sure only the right people can get in and also keeping track of who can do what was bit iterative work . But we sorted it out with clear tagging and regular reviews during our Audit call.

Interviewer: Thanks for sharing. It sounds like you've got a good handle on it.

Thanks :)

Today, I discussed implementing role switching and establishing access from the master account to multiple AWS accounts, namely Master, Staging, and QA.

please do checkout my video for detailed explanation : <https://lnkd.in/gAWxj3ja>

Interviewer : How good you are with docker ?

I can rate myself as 4/5 in docker :)

Interviewer : Okay , What strategies do you follow to keep your Docker environment clean and efficient?

In my experience, I ensure a clean and efficient Docker environment by regularly performing maintenance tasks such as removing dangling images, unused containers, and volumes.

Interviewer : What are dangling images ?

In Docker, dangling images refer to images that are not tagged and are not associated with any container. These images can occupy over time and consume disk space.

Interviewer : Okay, What is the command to remove all at a time ?

docker image prune - will prompt you to confirm the removal of all dangling images before executing the deletion. If you want to skip the confirmation prompt, you can use - docker image prune -f

Interviewer : thanks :)

Subscribe for more !!!

<https://lnkd.in/gAMw3gqs>

Interviewer: What are some common challenges faced when deploying containers in production?

One common challenge is to make sure that the containers are properly configured to work together parallelly . This involves setting up networking, storage, and security features.

Interviewer: Okay, can you elaborate on how you would address these challenges?

Inorder to address networking challenges, we can use container orchestration tools like Docker Swarm or Kubernetes.

Interviewer : sorry for interrupting, which one do you suggest ?

It depends on our environment and use case, if our project is smallscale then we can go with docker swarm, but the disadvantage with docker swarm is , it doesn't have auto scaling capabilities Whereas Kubernetes does have .

Interviewer : okay, please go ahead .

Coming to the storage part , we can use persistent volumes to ensure that data is retained even if a container crashes or gets replaced. And for security, we can implement access controls and encryption to protect sensitive information within the containers.

Fair enough thanks :)

Currently I'm posting Aws realtime practical stuff on YouTube . Later, I'll move on to DevSecOps and Kubernetes. It's perfect for career transitions and freshers looking to get into the cloud domain.

Check out here : <https://lnkd.in/gNurwWYT>

Interviewer : Il ask you one simple question can you walk me through ?

Yes

Interviewer : How did you handled broken pipeline in a live environment?

Recently when am working on a deployment , I found broken pipeline in a live environment, So initially I used our monitoring tool to identify the issue. Post that I found a connectivity problem with the database, I utilized logging and error tracking tools to pinpoint the root cause.

After resolving the issue all to gather with the support team , I implemented a fix by adjusting the database configuration.

To ensure stability, I continuously tested the solution in a staging environment before deploying it live. To avoid halting the live environment, we implemented rolling updates and utilized feature flags to selectively enable the fix without disrupting ongoing operations. Once validated, I deployed the fix during a scheduled maintenance window, minimizing downtime and ensuring smooth operations.

Interviewer : Scheduled maintenance window is really required ?

It's not partially required, any DevOps enthusiasts can put it in the comment section 😊 , we can follow two strategies 1 is CANARY, Blue Green .

Interviewer: let's say you're building a website for tax calculation, and users submit their tax information through a form. How does the data from the website get stored into the database?

So, when a user fills out the form on the website without using any instance , the data first goes to our API Gateway. Then, this data is sent to a Lambda function, the Lambda function saves this data into our DynamoDB database, where we keep all our information.

Interviewer : Okay, but how does this whole process help in saving costs and managing resources efficiently?

With this setup i recommend recommended to go with S3 hosted website , we don't need to worry about managing servers or paying for them when they're not being used. It's like having a power switch that we only turn on when EVENT is occurred . So, if our website gets fewer visitors, we don't waste money on unused servers. so we're always ready to handle any number of users without any extra effort.

Interviewer : Impressive , thanks :)

Checkout the full video on my YouTube channel : <https://lnkd.in/dWcxVqzA>

Interviewer : How good you are with Jenkins ?

I have a good grasp of Jenkins, I'd give myself a 4 out of 5 :)

Interviewer : Okay , Can you explain why you prefer Jenkins over Azure DevOps for our CI/CD pipeline?

Jenkins is an open-source automation server widely used for continuous integration and continuous delivery pipelines. It allows developers to automate the building, testing, and deployment of software projects, enabling faster and more reliable software delivery. and extensibility for various development workflows and environments.

Interviewer : Okay, but How do you integrate other tools with it and how do you secure your credentials and what strategies do you follow to avoid the downtime when 100 jobs are running ?

Jenkins has a vast plugin ecosystem, offering a wide range of integrations for various tools and platforms and to avoid security issues , one is storing our credentials under Manage credentials Tab , other way we follow HashiCorp Vault, a tool that keeps passwords and other sensitive information safe. It makes sure only authorized people can access them. To avoid downtime and we can implement the master-slave architecture in Jenkins, which is a common strategy to avoid downtime and improve availability.

Interviewer : Fair enough , thanks.

Don't forget to Subscribe here :)

<https://lnkd.in/gVKQcnqS>

Interviewer : Are you aware of S3 ?

Yes :)

Interviewer : Can you explain what we achieve when deploying a static website in S3 over traditional Hosting solution ?

After uploading our files to S3, we use CloudFront distribution. It improves performance by caching data, reducing latencies. It also secures our application with a Web Application Firewall and optimizes costs. Plus, it supports custom SSL certificates and integrates with ACM for easy certificate management.

But when following traditional way , we are supposed to configure auto scaling groups - load balancers and other services as well !!

Interviewer : fair enough, can you explain the basic steps you follow to create ?

- Create ACM
- Create Route53 hosted zones
- Create s3 bucket and upload all files
- Create Cloud front Distribution

Select HTTP-HTTPS routing , mention Default file, add SSL ACM, select S3 bucket static URL.

- Route to Route53 and call CFD by creating records , and add subdomain
- Try to access it from browser .

Interviewer : awesome , thanks :)

subscribe here for more : <https://lnkd.in/dwi8cAfv>

Interviewer: Have you worked on S3 Policies and Accesspoints ?

yes :)

Interviewer: How would you configure the S3 bucket and manage permissions to ensure the testing team can access the required images without making the bucket publicly accessible? Explain the steps ?

After uploading artifacts to S3, we select the specific file the testing team needs. Then, we choose the "Action" option, where we can find the "Preassigned URL" feature. Using this, we can give the testing team access to the file for up to 7 days. After that, the access will automatically expire.

Interviewer: Fair Enough, But what if we have a s3 bucket and two folders and only 1 developer should have access ?

In order to achieve this , initially we need to create a user with IAM we set up an inline policy, ensuring that access is given to the specific bucket and its folders. This approach allows us to control access effectively, limiting it to the necessary areas.

Please do checkout my channel here : https://lnkd.in/dC-C2w_u

Interviewer : Can you explain how to mount an S3 bucket to a local directory on a server? What tools or methods would you use?

Yea, We can mount S3 bucket to a directory but can only possible with third party tools like S3FS .

Interviewer : what were the steps you follow?

```
Sudo apt install s3fs
Mkdir s3mount
s3fs bucketname s3mount
Cd s3mount
Df -h | grep -i s3
```


Interview : impressive !!

Please do check out the my aws playlist , don't forget to subscribe if interested :)

https://lnkd.in/gx8Z_mpn

Interviewer: Let me ask you a scenario where we need to set up two Jenkins servers such that any job run on the primary server is reflected on the secondary server, and all logs should be stored in a centralized location accessible only to the platform team. How would you achieve this?

First, I would create an NFS, which in AWS is referred to as EFS. I'll use Terraform to provision two Jenkins servers in us-east-1a and us-east-1b within existing VPC, security group, and subnets.

Once the servers are set up, I'll navigate to the primary Jenkins server and edit the `/etc/fstab` file to mount the EFS at the centralized location, typically `/var/lib/jenkins` is my centralized directory

https://lnkd.in/g_DssPVU: `/var/lib/jenkins/ nfs`

Next, I'll ensure the secondary server also mounts the EFS. After making the necessary changes, I'll use the command `'mount -a'` to mount the file system.

This configuration ensures that all primary jobs executed on the primary server are automatically reflected on the secondary server, with logs stored centrally in `/var/lib/jenkins`.

Interviewer: Thanks for the detailed explanation!

Am going to upload the video on same tomorrow.

Please do checkout out my Channel for real time practical's video, don't forget to subscribe If you love the content.

https://lnkd.in/dC-C2w_u

Interviewer: Imagine you're managing data storage in AWS S3 and have set up lifecycle rules to move data to Glacier automatically. If you delete some files, how can you retrieve them ?

I will make sure that i have enabled Bucket versioning so that the data inside it will be not loosen.

Interviewer: But what if the entire bucket is accidentally deleted? What approach do you follow to ensure data safety?

To safeguard against the accidental deletion of the entire bucket, I would configure cross-region replication. This rule

ensures that all data is replicated to another S3 bucket in a different region. By doing this, even if the primary bucket is deleted, a copy of the data remains secure in the replicated bucket, ensuring data availability.

Interviewer : During a critical system update, you realize that the stateFile, which contains essential configurations and states for your application, has been lost. How would you handle this situation ?

If we find backups, we simply restore them, which brings back all our configurations.

Interviewer : Okay, Let's say there are no backups .

Then it's good to turn to Terraform import. This means we manually tell Terraform about each piece of our setup. We create new instructions for Terraform based on what was in the stateFile. Then, we use Terraform import to connect these new instructions with the existing setup. But before this initiating a change request is typically a good practice before Making any significant changes to the infrastructure.

Interviewer : That make sense, thanks :)

Check my YouTube channel here for more realtime videos : <https://lnkd.in/g7AY8t3k>

Interviewer: Can you explain how to set up AWS FSX for Windows and integrate it with AWS Workspaces using Active Directory?

First, we'd install Active Directory Domain Services (AD DS) on our Windows instance, configure the domain, and create user accounts. Then, in AWS Workspaces, we'd register a directory, launch Workspaces, and configure user settings. For FSX, we'd create shared folders, set permissions using Active Directory groups, and disable inheritance for security.

For a detailed walkthrough, you can check out my YouTube video here: https://lnkd.in/gVBBii_4

Interviewer: How would you securely enable communication between three Kubernetes clusters without using external DNS?

To securely enable communication between three Kubernetes clusters without relying on external DNS, I would implement a service mesh such as Istio. Service mesh provide robust features for managing secure communication between services across clusters, including encryption and authentication. Additionally, I would configure network policies within each cluster to control traffic, ensuring that only authorized pods can communicate. While NodePort and ClusterIP are also options for communication within Kubernetes clusters, they may not provide the same level of security and flexibility as a service mesh like Istio

Interviewer:

Can you elaborate on how you would manage the certificates and authentication between the clusters?

For managing certificates and authentication, I would use a centralized certificate authority (CA) to issue and manage the certificates, utilizing tools like cert-manager to automate this process. This ensures all clusters trust the same CA. For authentication and authorization, I would implement Role-Based Access Control (RBAC) policies to define permissions for different services and users.

Interviewer: Fair Enough, Thanks.

! ! Please read the answer slowly ! !

Interviewer: How would you set up a Kubernetes pod with multiple containers, giving one container full S3 access and another container IAM access?

I would use Kubernetes service accounts and AWS IAM roles.

For the container that needs full S3 access, I would create a Kubernetes service account and associate it with an IAM role that has the necessary S3 permissions.

Then, I would annotate the pod specification to use this service account for the specific container

For the second container that requires IAM access, I would create another Kubernetes service account linked to a different IAM role with the required IAM permissions and annotate the pod accordingly

Interviewer: So as per your answer, we can't create two service accounts for one pod, right?

Correct, we can create an init or sidecar container to handle the specific permissions for the second service account.

The init container can fetch and configure the IAM credentials, storing them in a shared volume for the second container to use

This way, you can effectively give different permissions to each container within the same pod.

Interviewer : Alright !

Feel free to add any comments or suggestions in the comment section if you'd like to 😊

Interviewer: Can you describe a scenario where AWS Systems Manager was crucial for managing security patches on instances in a restricted environment?

In my previous project, our instances were within a highly secure environment with strict access controls. These instances were in private subnets with no direct internet access. Using SSM, I scheduled and automated security patch deployments via the Patch Manager.

Interviewer: What other specific tasks did you perform on these private instances using SSM?

I used SSM to run scripts for software updates and to collect system logs for troubleshooting. This was done through the Run Command feature in SSM, which allowed me to execute commands on multiple instances simultaneously without needing direct SSH access. Also in my current project I have implemented enhanced monitoring for disk space , Detailed Transaction tracing and in depth log analysis .

Interviewer : Impressive , thanks :)

Am going to deploy a video on same, please check out my YouTube here : <https://lnkd.in/gese5mhk>

Don't forget to subscribe if interested 😊

Interviewer : Have you worked on terraform in your previous project ?

Yes , we used terraform for infrastructure provisioning and we do have multiple environments so we used terraform modules .

Interviewer : Sounds good, Will terraform directly deploy infra when you enter terraform apply ?

Terraform communicates with the cloud provider's API to create resources as defined in the configuration files, and then it prompts us to confirm with Yes or No.

Interviewer : Okay, So you said you worked on modules , I wanted to delete a particular service from a particular environment ? How can I do this with terraform ?

`terraform destroy -target=resource_type.resource_name`

Interviewer: Thanks :)

YouTube : <https://lnkd.in/gZdpvz6U>

Interviewer: Describe how you would configure AWS SSM and CloudWatch to monitor custom metrics like disk space utilization for EC2 instances ?

First, I log in to the AWS Management Console and navigate to the Systems Manager. Within SSM, I locate the "Run Command" feature. This is where I can execute commands on my EC2 instances without needing to log into each one

individually.

Next, I select the appropriate command document designed for custom metrics monitoring. This document contains predefined scripts and configurations needed for the task.

After that, I choose the EC2 instances I want to monitor. AWS allows me to select instances directly or use tags to filter and apply the command to multiple instances at once. Once the instances are selected, I configure any necessary parameters for the command. In this case, I would include scripts or commands specifically for collecting disk space utilization metrics. With everything set up, I initiate the execution of the command on the chosen EC2 instances. The command runs and collects the disk space utilization metrics as specified. Finally, I switch to the CloudWatch service in the AWS Management Console. Here, I look under the EC2 namespace for custom metrics related to disk space utilization. These metrics will be available for me to analyze and create dashboards as needed to monitor the health and performance of my EC2 instances.

Checkout the detailed video on same : <https://lnkd.in/gSNd8s5G>

Interviewer: How would you create custom workspaces for two Linux instances and install the necessary packages for monitoring custom metrics like Disk_Used, Innode_Percent & Cpu_Percent . What steps would you take?

I would start by setting up the AWS Systems Manager to manage the instances. First, I would ensure that the EC2 instances have the appropriate IAM roles with permissions for SSM and CloudWatch. Using SSM, I would then run commands to install the CloudWatch agent on both instances. After that, I would use the package manager on each instance to install necessary packages. I would configure the CloudWatch agent to collect the required metrics and log data from these instances. Let's say if our web servers are running on Nginx so it will be easy in future to apply custom metrics on nginx error logs and Access logs .

Interviewer: Okay, you mentioned Nginx logs. Can you explain how you copied Nginx error and access logs to your CloudWatch?

To copy Nginx error and access logs to CloudWatch, I first modified the CloudWatch agent configuration file on each EC2 instance to include the paths to the Nginx log files. I specified the log file locations in the configuration file, creating log streams for both the error and access logs(/etc/nginx/nginx.config) . Once the configuration was updated, I restarted the CloudWatch agent to apply the changes. This ensured that the logs were being collected and sent to CloudWatch. I then verified that the logs were appearing in CloudWatch by checking the log streams in the CloudWatch console.

Interviewer : Great :)

Interviewer : you've mentioned that you've worked on docker

Yes :)

Interviewer : what is the Docker's feature that helps you ensure that one application does not consume all the server's resources?

Docker's cgroups feature helps you by limiting how much CPU, memory, and other resources each application can use. This means you can set rules so one application cannot take up all the resources, making sure other applications still have enough to run properly.

Interviewer: okay , but How does Docker's use of ensure that each application has its own isolated environment?

Docker's namespaces feature ensures that each application runs in its own separate space. This means that the files, network settings, and processes of one application are hidden from others. It keeps everything isolated, so changes in one application won't affect any other applications.

Interviewer : Sounds Good, Thanks :)

Please checkout my YouTube here : <https://lnkd.in/gfwMssMN>

Interviewer : You need to automate the deletion of GP3 volumes immediately when someone creates, What steps you take ?

I would set up an EventBridge rule to trigger a Lambda function on a regular schedule. The Lambda function would have the necessary permissions to identify and delete unused volumes by following this any GP3 volumes created are automatically deleted .

Interviewer: Sounds good, There is an server and i wanted to take snapshot of the server for each and every 5hours, what steps you follow ?

To take a snapshot of a server every 5 hours, I would use a scheduled task in Amazon Event Bridge. I'd configure the job to run every 5 hours and ensure the volume ID is specified for the snapshots. This setup will automate regular backups, maintaining data integrity and availability.

Interviewer : Thank you :)

Please Do watch the detailed video on same here : https://lnkd.in/g_Ee7zU2

Interviewer : Imagine you are working on a Kubernetes cluster, and you need to deploy a logging service that collects logs from all nodes in the cluster. This service should always run on every node, even if new nodes are added later. How would you achieve this?

To deploy a logging service that runs on every node and automatically includes new nodes, I would use a DaemonSet. A DaemonSet ensures that a pod runs on all nodes in the cluster. When new nodes are added, the DaemonSet automatically adds the logging pod to these new nodes, ensuring logs are collected from everywhere.

Interviewer: If i wanted to deploy a database service where each instance requires a stable identity, What deployment strategy you follow ?

For the database service, I would use a StatefulSet. Each pod in a StatefulSet gets a stable, unique name and maintains

its data in persistent storage even if it's deleted or rescheduled.

I deployed a video on YouTube about using AWS Developer Tools. Check it out <https://lnkd.in/g3FVhyGU>

Interviewer: Imagine you are managing an AWS environment and need to monitor API activities and track changes to resources. Which AWS services would you use, and what are the key differences between them?

In AWS, we can monitor all API activities, such as logging from the console, logging from the CLI, or logging from any third-party application like Packer and Terraform. All these events will be recorded by CloudTrail, and we can also ship those logs to CloudWatch or S3 for long-term retention for governance and compliance purposes.

On the other hand, AWS Config is a service to monitor and track changes happening to your resources. For example, if you attach an Elastic IP to your instance, add tags to your instance, or make any other changes to your resources, AWS Config will show you an overview of the changes that have been performed.

Interviewer: Sounds good!! Thanks.

Am going to post the same video as above today, Along with Elastic BeanStalk. Please do checkout my channel below, Don't forget to subscribe If you like my videos 😊

YouTube : <https://lnkd.in/ghMzDF5Q>

Interviewer: Imagine you are setting up a Docker environment with multiple containers. One of these containers is a database container, and you don't want to isolate it from the other containers. How would you ensure that all containers can communicate with each other?

In that case, I would use a bridge network. By default, Docker containers use the bridge network, which allows them to communicate with each other on the same network. This way, the database container and the other containers can easily interact without being isolated.

Interviewer: Can you explain why you would choose a bridge network in this scenario?

bridge network is something which is straightforward to set up and manage. It make sures that all containers on the same network can communicate with each other using their container names as hostnames, which simplifies connectivity and configuration. This way is the is ideal when you want to maintain easy communication between your containers, including the database container.

Interviewer : Thanks !

Any container enthusiasts, your insights on overlay networks would be greatly appreciated/helpful 🙌

Subscribe to my channel if you like my videos : <https://lnkd.in/ggPycp3b>

Interviewer: How good are you with containers?

I'm good with containers. Please proceed with the technical discussion, and I'll answer as many questions as I can 😊

Interviewer: Okay, Imagine you need to store sensitive information like database passwords and API keys. How would you use Kubernetes to securely manage ?

I would use Kubernetes Secrets to securely store sensitive information. This allows us to separate sensitive data from the application code, making it more secure.

Interviewer: Sounds good. But my Questions is How would you ensure that the secrets are not exposed or mishandled?

I would control access using Kubernetes Role-Based Access Control. so that only specific resource and users can access the secrets. I would also enable encryption for the secrets within the cluster. Also its good to regularly reviewing audit logs to monitor who accesses the secrets would help in finding any unauthorized access. This way, the sensitive information remains secure and properly managed.

I Posted a Realtime task on ECR and ECS, Also explained what strategies does Org followed before virtualization and Containers were not in the game .

please checkout the video here : <https://lnkd.in/gp4e6uP5>

nstead of Introduction or introduce yourself, they might ask you like this!!

Interviewer : Hello, how are you doing?

I'm doing good, thanks :)

Interviewer : I see in your resume you have good exposure on multiple tools , Can you recall yourself what you did Yesterday after Logging-in ?

Usually, Daily my day starts with a heads-up call, where we discussed roadblocks and backlogs and set priorities, and ensured everyone was on the same page.

Post that, I monitored our 10 servers in the Us-West and Us-East US using Prometheus to ensure everything was working fine. basically If any issues occurs within our range, we resolved them else we escalated them to our on-site support team. Additionally, I wrote a Dockerfile to containerize an test web app as per our QA engineer requirement, We do track everything with Service now and DFM . I also worked on a Jenkins task. Our Sonar founds vulnerabilities, Even though the build stage passed and moved to the next stage, so I implemented quality gates to the existing Groovy script, ensured it was working as expected, and raised a PR.

Interviewer : Thanks for detailed walk through .

Interviewer: Have you worked on Docker Swarm?

Yes, but we have recently moved to a self-hosted management cluster.

Interviewer: Okay, but what challenges did you face with Docker Swarm that led you to move to Kubernetes?

Docker Swarm doesn't support TLS secrets, Namespaces, RBAC, StatefulSets in Deployments, or Liveness & Readiness Probes. It also lacks support for third-party plugins and environment variables as secrets. These features are missing in Docker Swarm.

Interviewer: In Entire Kubernetes Components, what's the speciality of Kubelet.

Kubelet runs as a Daemon Service, Rest all runs as a Pods. kubelet is a critical component on each node that ensures containers are running in a pod by interacting with the container runtime

please do checkout my Docker Voting app deployment on EKS here : <https://lnkd.in/gVbsTMyZ>

Interviewer: what are the best steps you take for cost optimization ?

- Document Which resources need to be run 24/7.
- Shut down or pause resource which are not necessary running off hours .
- Use Reserved instance for long term and Spot instances for reduce billing for temporary workloads.
- trusted advisors to check unutilized resources.

Interviewer: Sounds Good, What are the best practice you take for Security ?

- Deploy infra using IaC- Terraform
- Where every possible use serverless using lambda
- CloudTrail is enabled on all locations where infra is deployed.
- No Servers are deployed on public subnets.
- Backup are taken for Database Servers
- Autoscaling is used withstanding the performance hikes.
- All Servers are distributed across multiple AZ for HA.
- SG and NACL are used to restrict traffic.
- VPC Endpoints are used for internal connectivity to other servers from VPC.
- Microsoft AD is used for SSO.
- All services are monitored with cloud watch.

Guys We have one session left to wrap up our AWS series. Today and tomorrow, we'll discuss daily responsibilities and how to position yourself as a DevOps engineer. I encourage everyone to practice a lot. We'll be diving into Docker and Kubernetes From next week .

If you're new to containers, follow my videos closely, they'll prepare you to handle real-time scenario-based questions. We're focusing on tasks ranging from easy to hard, exclusively on a self-hosted 6-node Kubernetes cluster—not using Minikube or single nodes.

Check out my YouTube channel and subscribe if you find the content helpful

<https://lnkd.in/gHMzDF5Q>

Today marks the last Session in AWS series!

What new Exiting stuff i can get from your AWS series comparatively to paid/others?

Great Question!!

We have discussed about 36 Services but here are some you ain't get from paid/others. VPC Flow Logs - VPC EndPoints - EC2 AMI Images with hashiCorp Packer - IAM Identity Center (Not IAM but IAM identity Center where you can manage lakhs of users from AD) - RedShift - S3 Access Points - Glacier - AWS Workspaces - FSX-EFS - AWS Amplify, each service has been covered with detailed, hands-on projects. [hashtag#masteringaws](#)

For more details, check out the complete GitHub repo: <https://lnkd.in/gnwcYhvi>

And don't miss our latest video where I discuss how to introduce yourself as a DevOps engineer :

<https://lnkd.in/g86429C2>

AWS – DEVOPS – KUBERNETES BASIC QUESTIONS NOT SCENARIO BASED QUESTIONS , Check out the video here :

<https://lnkd.in/g6Vt-rPT>

Interviewer: During development or testing phases, if you want to ensure that Docker automatically removes a container once you stop it, how would you achieve this?

```
docker run --rm -d --name frontend -p 8001:80 containername
```

Interviewer: Okay, When deploying an application to a production environment, what considerations would you take into account?

I would ensure that Docker images used in production are lightweight and optimized for performance. Also i will make sure removing unnecessary dependencies and ensuring that only required components are included in each image. I will also try to avoid using official Docker images or images from trusted sources to minimize security risks.

Checkout my first video on Docker & Kubernetes Series : <https://lnkd.in/ggvTMKfg>

Interviewer : You have noticed that the default Docker directory `/var/lib/docker` is getting full as you continue downloading images and generating logs. What Precautions you take to avoid this and tell me the steps to achieve ?

In, Our previous environment we made sure that our `/` space is not exceeding more than 50% and we used to store all docker Appdata in separate directory By creating an EBS volume and attaching it to instance. post that,

```
lsblk
```

```
fdisk diskname
```

```
lsblk
```

```
adding filesystem - mkfs.ext4 /dev/xvdf1
```

once done we create a directory called `/dockerdata` with "`mkdir /dockerdata`" and By adding an entry to `/etc/fstab`, EBS volume is mounted every time the system starts. Once all done we do the changes to the docker service directory file accordingly .

Interviewer :Okay, Imagine you are managing Docker images on your system. How would you list all the dangling images, and then how would you delete ?

```
docker images --filter=dangling=true
```

 This command shows the unwanted none files

```
docker images -q --filter=dangling=true
```

 This command will delete unwanted none files

Interviewer : Great :)

Please do checkout my docker day 02 video here: <https://lnkd.in/gDA2i5Rh>

Don't forget to subscribe if you like my videos 🙏

Community tab : <https://lnkd.in/gKpUrV4A>

I cover some unique services in my AWS playlist. If you follow and practice what I did, you can easily crack any Question on AWS during interview!

No one motivated me on Monday, so Bharath came on Tuesday 😊 .

A friend told me that when the W* site got hacked, they didn't allowed vendors to use local tools. Instead, they used Cloud9 for development and other purposes. I explained Cloud9 in my videos too. And In organizations, lakhs of users' data are stored on-premises. When integrating with the any cloud, users get access through directory services and IAM Identity Center. I explained both.

I tried to explain proper migration from VMWARE ESXI server to AWS S3 , unfortunately ESXI pricing was quite out of my budget But in future il definitely do a real time migration video .

If you're really trying to learn AWS, check out my playlist. If interested, go ahead and complete it. Apologies for the first few videos; please watch them at a faster speed as I explained slowly due to YouTube beginners' mistakes.

Here is my AWS Playlist: <https://lnkd.in/d8hNSdzX>

Interviewer : How can you prevent your Docker build from using the previous cache when you run the build?

Docker build -t kiran236/customnginx:V2 -f dockerfile.dev . --no-cache

Interviewer : Great, can I overwrite the build ARG when running container , if not, what flag you pass to overwrite ?

No we cannot, we can pass or overwrite the build ARG only during the time of building the image not when running the container . But during the build time you can add --build-arg for overwriting .

Interviewer : Fair Enough , if I wanted to deleted all stopped containers, all network which were not used by any containers and all dangling images . What is that one command, I can run?

Docker system prune and prompt with yes . It's going to deleted all the unused network , images and build cache too .

I have explained Docker file line by line on my channel Here : <https://lnkd.in/g6aFwaa9>

Interviewer : Imagine you're building a Docker image for a Node.js application. What steps would you take to minimize the image size while ensuring all dependencies are included?

For such application I would use a multi-stage build. In the first stage, I would only take care of installing the dependencies by copying json files and then running npm install .

In the second stage, I would use a smaller runtime image like node:alpine to run the application, and then copy the necessary files and built artifacts from the first stage. Additionally, I would make sure to avoid adding extra commands that can increase the image size unnecessarily, like temporary files or unnecessary packages. This way, the final image will be as small and efficient as possible.

I did a same video on above along with Google distroless images and also explained what exactly a DevOps engineer do once developer done writing the code , please do check out here : <https://lnkd.in/gPrgk8Kd>

If you are looking for proper aws realtime projects on individual services checkout here : <https://lnkd.in/gUyUs29y>

Interviewer : What are Terraform null resources ?

A null resource in Terraform is a resource that doesn't actually do anything on its own but can be useful for various purposes such as running local-exec or remote-exec provisioners, creating dependencies between resources, or as a placeholder for future changes .

Interviewer : can you give one use case ?

Suppose you have an AWS EC2 instance and you want to run a script on your local machine every time the instance's tags are changed. You can use a `null_resource` with a `local-exec` provisioner and create a dependency on the EC2 instance tags.

For aws realtime tasks on 39 service , checkout my YouTube here : <https://lnkd.in/gmH4yDGY>

For Docker & Kubernetes , checkout here : <https://lnkd.in/g-3-HHj>

Interviewer : Have you worked on docker Swarm ?

Yes , I do but now we moved to self hosted management cluster .

Interviewer: Okay , what is routing mesh ?

Routing mesh in Docker Swarm ensures that any request sent to the cluster, regardless of which node it arrives at, it is directed to the appropriate service.

Interviewer : What's your relevant experience in DevOps ? Have you written any Jenkins scripts ? Or anyone developer writes and deploys ?

3.7 was my relevant experience and we do have our source code in our SCM and I do write Jenkins scripts from scratch too.

Interviewer: Okay, what are your CI stages and what are you CD stages ?

In CI we do have code commit, once developer done writing code , we ensure it meets the quality standard and we do have Build and automated tests to ensure there were no bugs and proper functionality .

In CD we do have staging and Development deployments along with monitoring deployment too for tracking application healths on regular basis .

Please do checkout my YouTube channel here for more realistic stuff: https://lnkd.in/gJFBa_pR

Interviewer: What are the different Ingress controllers we have apart from nginx?

KONG, AWS-NLB, and Traefik .

Interviewer: Fair enough. So my client requires all their websites should use SSL termination. Currently, the websites run on port 80, but they need to be automatically redirected to port 443 for secure connections. How would you achieve this?

To achieve SSL termination and redirection, I would configure Traefik as the Ingress controller. Traefik handles routing, SSL termination, and manages secure connections. Additionally, I would set up a Route 53 DNS zone with the master IP

and create a record named "traefik" to manage domain names. This will ensure that requests to our services are correctly routed through the Network Load Balancer configured with Traefik. I would create two target groups one for traffic on port 80 and another for traffic on port 443. For the Network Load Balancer, one for port 80 and one for port 443 to properly distribute incoming traffic.

In Route 53, I would create DNS records for our services and point them to the NLB.

Interviewer: Thanks.

I will be posting the same video with Docker Swarm cluster setup at 8:30 PM on my YouTube channel:

<https://lnkd.in/gMzDF5Q>

For those watching my posts for the first time, I post videos on AWS, DevOps, and Kubernetes on my channel .

Freshers or anyone looking to shift domains will find practical tutorials On my channel .

AWS Playlist: <https://lnkd.in/gigi68GY>

Docker + Kubernetes Playlist : <https://lnkd.in/gSNpdR-H>
