**DevOps Understanding – Ethans Pimple Saudagar- Prakash**

**Roles and Responsibility, DevOps Day to Day Activity:**

1. Currently working with abc company as DevOps engineer.

2. In my project we have java based applications and for these applications we have created CI and CD pipeline using open source such as git, jenkins, maven, docker,kubernetes,ansible,chef,aws, nagios, nexus, sonar, junit etc.

We are responsible for implementing DevOps CI/CD pipeline. Evaluation of tools for enhancement of pipeline. Setting up infrastructure automation, securing infrastructure by implementing devops and cloud best practices. smooth and faster build and deployments.

**3. day to day activity:**

We interact with developers and operations team on daily/weekly basis to get resolved their issues such as build and deployment failure. Access management issues. Knowledge sharing for adding new tools/technology/process.

Actively monitoring and troubleshooting of any Jenkins job failure. adding/removing/updating build agent (slave). Setting up new jobs. Integration of tools such as JIRA, SonarQube, Ansible, Docker, Nexus with Jenkins.

docker file creation and modifications for applications. Docker Container and image management.

Management of docker registry, docker secrets. Microservices deployment.

We use Kubernetes for docker container management. We are responsible for pod creation and deployment. Managing replica sets for autoscaling by setting up CPU usage in deployment file.

Exploring Kubernetes for monolithic application.

We use Ansible as configuration management purpose. We are responsible for playbook creation, infrastructure provision through ansible. We use it as deployment tool we have integrated it with Jenkins. We also use it as patching tool so we have written playbook for linux/windows patching.

Monitoring of AWS Cloud based resources such as EC2, VPC, S3 through Nagios, we are responsible for plugin creation and take necessary action.

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

**Git:**

1. Difference between git pull & git fetch & git clone.

Q 2. Difference between git merge and rebase

Q 3. What is version control, types, diff b/w distributed version control & centralized version control system and examples.

Q 4. Difference between git reset and revert.

Q 5. How many branched are there in your project:

Ans: for application1-5 (master, development, feat_devops, feature_dev_1, feature_dev_2)

Q 6. How can we revert changes for particular commit.

Q7. Tell me something about git strategy in your project.

Q 8. Who can merge changes to master branch.

Ans: lead developers/managers, devops engineer.

9.What is git remote repository?

Ans: A remote in Git is a common repository that all team members use to   exchange their changes. Using remote repository all the developers can work in   collaboration, we use Gitlab as source code manage repository.

Q 10. GIT revert Command?

Q 11.GIT reset Command?

Q 12. Define your Git strategy?

Q 13.hard reset vs soft reset?

Q 14. How to protect Git Branch?

Ans: it prevents its creation, if not already created, from everybody except users with Maintainer permission

It prevents pushes from everybody except users with Maintainer permission

It prevents anyone from force pushing to the branch

It prevents anyone from deleting the branch

1. Navigate to your project's Settings ➜ Repository
2. Scroll to find the Protected branches section.
3. From the Branch dropdown menu, select the branch you want to protect and click Protect

Q 15. How can we commit without adding files in staging area?

Ans: git commit -a -m "your message"

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

**AWS:**

Q. what is internet gate?

Ans: https://aws.amazon.com/vpc/faqs/

Q.NAT gateway?

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

Q. is there any cost for aws versioning?

Q. if one ec2 is in private subnet then how to install patches from internet without providing internet connectivity?

Ans: Nat Gateway/Nat Instance

Q.IAM User and Role difference?

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

Q.AWS life cycle management?

https://docs.aws.amazon.com/AmazonS3/latest/user-guide/create-lifecycle.html

Q.s3 encryption?

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

s3 property- Versioning, Static website, Events, Cross region replication.

S3 Management: Life cycle rules, analytics (aws analytics tool provide facility to manage transition of correct data to the correct class),

inventory: we can get report of our inventory in s3 like. bucket object, object, versioning details.

http://docs.aws.amazon.com/AmazonS3/latest/dev/analytics-storage-class.html

http://docs.aws.amazon.com/AmazonS3/latest/dev/storage-inventory.html

AWS REDSHIFT port: 5439

Q. Customer Managed policy and AWS Managed policy?

Q. How to mount file in linux?

Q. How you deploy your application to ec2 instance.?

Q. How to create aws lambda function?

Proficiency in designing, developing, and deploying cloud-based solutions using AWS

Q. in auto scaling which ec2 instance will terminate first?

Q. How to grant access to another user which is not in your aws account, means diff aws account right.?

you can do by creating aws role: -cross account authentication,

http://docs.aws.amazon.com/IAM/latest/UserGuide/tutorial_cross-account-with-roles.html

Q. Diff between s3 and elb?

Q. why s3 is 99.999999999 durable?

means if you have a file and, in that file, if you have 10000 objects. so, the chances of losing a single object from 10000 files in 1 million. means 9 ki power 11 times.

Q. how to deploy ssl application to aws?

Q. How can I set up HTTP to HTTPS redirections on ELB using Apache backend servers?

https://www.youtube.com/watch?v=hvqZV_50GlQ

Q. How to Set up SSL on EC2 instance?

Q. What is user data in ec2?

Q.in auto scaling which ec2 instance will terminate first.?

cloud watch does not provide metrics on basis of Memory/

VPC peering can't be does if ip overlaps. means when one vpc 1 is 10.0.0.0/16 and vpc2 is 10.0.0./16

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

**Jenkins**:

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

**Docker**:

Q.1 Docker secrets? should we keep docker secrets in source code?

Ans: secret is a blob of data, such as a password, SSH private key, SSL certificate, or another piece of data that should not be transmitted over a network or stored unencrypted in a Dockerfile or in your application's source code, Refer Docker folder google drive.

Q.2 Difference between docker Add and Copy? Which one you prefer and why?

Q.3 Blue Green deployment in Docker?

Ans: Blue-Green Deployment is a strategy to release new version of the app without downtime. The basic idea behind this technique involves using two identical production environments, named Blue and Green. At any time, only one of these environments is live and serving the production traffic. The other one is used to test newer version or for roll-back.
Let us assume that the current live production environment is Blue. When the new version is ready, we can deploy it to the non-production environment - Green. None of our users can see this new version as the live environment is still Blue. We can test the new version from the Green environment now. If this version is ready for release, we switch the production environment to Green and the users can now see the new release. Now the live version in at Green and staging can be done in Blue. If there is some error with the new release in Green, it is easy to roll-back to previous version by just switching the production environment back to Blue.
https://botleg.com/stories/blue-green-deployment-with-docker/

Q.4 Docker Architecture?

Q.5 Docker container vs VM ?

Q.6 How many networks are there in docker?

Q.7 How can we communicate in case of docker host cluster?

Ans: Overlay network

Q.8 Docker file components and there use case?

Q.9 How to do port mapping in docker? Difference between P (CAPITAL) and p (small) in docker run command?

Q.10 Have you written any dockerfile?

Q.11 Do you use docker hub or private docker registry?

Ans: private docker registry.

Q.12 Who manages docker registry in your organization?

Ans: We have global IT team they manage docker registry throughout organization. We have rights on our docker repository, we can push pull our docker images.

Q.13. What is docker

Q.14. What is diff between docker image and container

Q.15. How will you remove docker image

Q.16. How is docker container different from hypervisor

Q.17. Can we write compose file in json instead of yml

Q.18. What are the common use cases of docker?

Q.19. What are the main features of docker-compose

Q.20. What is diff between docker command up, start, and run

Q.21. What is docker swarm

Q.22. What is docker image

Q.23. What is docker machine

Q.24. Can we run more than one process in container

Q.25. What is docker hub

Q.26. What is the main security concerns with docker based containers

Q.27. How does docker simplify s/w development process

Q.28. How can docker run on different linux distribution

Q.29. What is the difference between docker ADD and COPY command

Q.30. What is build cache in docker

Q.31. What are the diff kinds of namespace available in docker

Q.32 How will you monitor docker in production

Q.33. Why does docker compose does not wait for container to be ready before moving on to

Q.34. How to build an image in docker?

Q.35. how can you check if, some webservice in running container not working as expected?

Q.36. can you delete an image, if container associate with it is in running state?

Q.37. can you delete an image, if container associate with it has stopped?

Q.38. what is difference between softlink and hardlink ?

Q.39. how can you expose port in docker ?

Q.40. how can you mount volumes in docker ?

Q.41. how you using docker in your project ?

Q.42. In pom.xml , can you use docker plug-in ?

Q.43. what is docker registry , how you using in your project?

Q.44. have you used kubernetes

Q.45. how can you go inside a login container

Q.45. Write docker code to deploy ngnix webserver and it should listen on port 9090

Q.46. What is the difference between Docker and docker compose

Q.47. How do I transfer a Docker image from one machine to another one without using a

repository, no matter private or public?

**Ans**: You will need to save the Docker image as a tar file

docker save -o <path for generated tar file> <image name>

Then copy your image to a new system with regular file transfer tools such as cp or

scp. After that you will have to load the image into Docker:

docker load -i <path to image tar file>


docker save -o c:/myfile.tar centos:16

Q.48: How to rename running docker container?

**Ans**: docker rename <current container name> <new name>

Q.49: How to check layers in docker image?

**Ans**: docker history <docker image>

this command will help you in any situation where the content of your image isn't what you

expect, this will help you to solve issue.

Q.50. What is docker export?

**Ans**: Export is used to persist a container (not an image). So we need the container id which we

can see like this:

docker ps -a

To export a container, we simply do:

sudo docker export <CONTAINER ID> > /home/export.tar

Q.51. If you have a running container that you would like to move to another host, how can you do that?

**Ans**: save the container to an image, save it to a tar file, move it to your new host and load the image into the new docker server.

step1: commit a container and create new image

docker commit <container ID> mynewimage

docker save mynewimage > /tmp/mynewimage.tar

Copy the mynewimage.tar file to your new Docker instance using whatever method works in your environment, for example FTP, SCP, etc.

Run the docker load command on your new Docker instance and specify the location of the image tar file.

docker load < /tmp/mynewimage.tar

then type docker images

Q.52. What is Difference between save and export in Docker?

**Ans**: **Export**

Export is used to persist a container (not an image). So we need the container id which we can see like this:

docker ps -a

To export a container, we simply do:

sudo docker export <CONTAINER ID> > /home/export.tar

The result is a TAR-file which should be around 2.7 MB big (slightly smaller than the one from save).

**Save**

Save is used to persist an image (not a container). So we need the

image name which we can see like this:

sudo docker save <image id> > /home/save.tar

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

**Kubernetes**:

Q. why Kubernetes over aws ecs/Docker Swarm?

Q. Difference between Kubernetes and Docker Swarm?

Q. Docker Architecture?

Q. What is Pod, Replica sets, Kubeadm, Kubelet, Kubectl in Kubernetes?

Q. How many pods you have?

Ans: You can answer in between 50-100 pods, make sure you remember your applications names.

Q. how do you manage cluster in Kubernetes?

Q. Can you brief on the working of the master node in Kubernetes?

Q. What are the different types of services in Kubernetes?

| Cluster IP | Node Port | Load Balancer | External Name |
|---|---|---|---|
| • Exposes the service on a cluster-internal IP. | • Exposes the service on each Node's IP at a static port. | • Exposes the service externally using a cloud provider's load balancer. | • Maps the service to the contents of the External Name field by returning a CNAME record with its value. |
| • Makes the service only reachable from within the cluster. | • A Cluster IP service to which Node Port service will route, is automatically created. | • Services, to which the external load balancer will route, are automatically created. | • No proxying of any kind is set up. |
| • This is the default Service Type. | | | |

https://www.edureka.co/blog/interview-questions/kubernetes-interview-questions/

https://www.onlineinterviewquestions.com/kubernetes-interview-questions/

https://www.zeolearn.com/interview-questions

https://www.zeolearn.com/interview-questions/kubernetes


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

Ansible:


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


Nagios:


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