NIO

1. Tell me about yourself?

I have been working as a devops engineer and build engineer for more than 8 years. I started my career as a python developer. I have worked with many tools and technologies like ant & maven to automate the build process, also worked with several version control tools like GIT and SVN and also implemented CI by using Jenkins, integrated Jenkins with different tools. I have also worked with configuration management tools like chef, Ansible, also migrated some applications from my own data center to cloud by using AWS, and I also have knowledge on container based applications like Docker.

Additional:

and migrated from SVN to GIT by using “SUBGIT” which helps you to do one-time import from SVN to GIT, once we push to GIT or commit to SVN, SUBGIT will take care of synchronization. I have also worked with several application servers like tomcat, weblogic, jboss, and websphere.

2.day to day tasks:

I am in a team of 6 members, I worked in a devops team. I will work with one of the application team, I will try to analyze their current process by seeing whether they are using any manual steps for the purpose of building a project. I will try to adopt their project and try to implement CI, CD methodology for them. If they are doing build and deployment in their own data centers I will try to migrate those to AWS.

I will write chef cookbooks and ansible playbooks to automate infrastructure related tasks. Like purging the nodes, installing and configuring different application servers , so all these things I will try to automate with chef and ansible.

3. What are the services you used in AWS

EC2, S3, VPC, EBS, ELB. I have used most of the services to bootstrap the nodes. Now a days we are trying to use terraform (also similar to AWS CFT used to create, update and version AWS infrastructure, an APN technology) also to provision the nodes. It is giving more flexibility to use AWS services. We can divide variables and environments.

4. Give me one example of which application you have migrated from data centric framework to AWS framework?

I have migrated one of our company’s application to AWS but I am not only the person responsible for migrating the entire application, but I have written cookbooks and recipes and also using cloud formation templates with different resources and invoking the chef cookbook recipes, and also when we kickoff the CFT it should be able to create instances and monitor those instances by using cloud watch. I have also written cookbook for installing tomcat and I have invoked that through the cloud formation template.

4.what did you do for security purposes

I have created IAM roles and created users, groups, roles and provide access based on the roles. Other security is like creating VPC for our own environment. And also by creating security groups and restricting inbound and outbound access as well. Other kind is by using NACL. It acts as a firewall at the network level.

5. What is instance profile? How do you create a role?

It’s a container for IAM role and you can pass this role information to a EC2-instance when the instance starts. This role gives access permissions to S3 buckets and other repositories where your applications are stored.

We can create IAM instance profile by using AWS CLI

Command to create instance profile is ---

Aws iam create -instance-profile - -instance-profile-name

6.What is the difference between NACL and SG?

NACL are applicable at the subnet level, if any instance in the subnet associated with NACL has to follow the NACL rules whereas security groups are applicable at the instance level.

In NACL we can set both allow and deny rules for instance, whereas in SG we can only set allow rules for instance, by default everything is denied.

SG evaluates everything before allowing the traffic. But its not the same case while coming to NACL. It has to first check the deny rules and the check the allow rules.

7. What is cfn-init ?

It is a script that reads the template metadata from AWS::CloudFormation::init key and used to

1. Fetch data from cloud formation
2. Install packages
3. Write files to disk
4. Enable/disable, start/stop service

If you use cfn-init to update an existing file it creates a backup file with .bak extension.

8. Any experience on docker, Benefits of Docker

One of the application team wanted to implement docker. So I have build images, pushed them to docker registry, and integrated with jenkins. So after the build they can create the containers, they can install and configure AEM (Adobe experience manager) they can build the code and run test cases, if everything is good they can destroy those containers. We use docker for non-prod environment. Just to check the build and deploy and also build is successful and all the test cases are run successfully are not. Only for that purpose we use docker.

9. Can you tell me which project u have used docker?

Currently we are using docker. Here we will implement lots of tools and technologies. One of the application team had approached me to implement docker in which they want to implement CI, CD using docker. So for that purpose I have created a docker file and created docker images and whenever they wanted to build through jenkins they kicks off the container run test cases and also sends email whether build is success or not or deployment is success or not. So after everything is done we will push the docker images to docker registry. As because AWS is expensive they have chosen docker containers which is an open source and light weight so I have implemented it for them.

**9. Nginx** is an open source web server that can also be used as a proxy. Some of the largest trafficked websites use it as their web server of choice as it is known to be efficient and fast in the way in handles concurrent connections. Furthermore, certain Apache web server users take advantage of Nginx in combination with Apache by using it as a reverse proxy.

**Varnish** on the other hand, is not a web server at all. Rather, its purpose is to act as a front end accelerator or reverse proxy. However, unlike Nginx, it is **not a standalone solution** to run an entire application. You still require a dedicated web server in order to use Varnish.  [VCL](https://varnish-cache.org/docs/trunk/users-guide/vcl.html), or Varnish Configuration Language, is the language used in Varnish in order to specify certain request handling rules and caching policies.

In terms of comparable features, Nginx and Varnish are quite similar. For instance, both solutions can be used as a **reverse proxy** and **load balancer**. They also allow you to configure cache and can help protect against [DDoS attacks](https://www.keycdn.com/support/ddos-attack/). However, there are also a couple differences such that Varnish supports [ESI](https://www.varnish-cache.org/docs/3.0/tutorial/esi.html) where Nginx does not, however, Nginx supports SSL where Varnish does not. Additionally, it was not until recently (September 2016) that Varnish just started [supporting HTTP/2](https://info.varnish-software.com/blog/how-its-made-varnish-5.0-and-http/2).

1. What happens at the boot up process?

* When you turn on the power, the power is distributed to all parts of the system.

The 6 high level stages of a typical Linux boot process:

1. BIOS: Basic I/P, O/P system executes MBR.

Searches, loads, and executes the boot loader program.

1. MBR: Master Boot Record executes GRUB.

MBR contains the information about GRUB. So, MBR loads and executes the GRUB boot loader.

1. GRUB: Grand Unified Bootloader loads and executes kernel and initrd images.
2. Kernel: Kernel executes /sbin/init.

Since init was the 1st program to be executed by Linux Kernel, it has the process id (PID) of 1. Do a ‘ps -ef | grep init’ and check the pid.

1. Init: Init executes runlevel programs

Looks at the /etc/inittab file to decide the Linux run level.

0 – halt

1 – Single user mode

2 – Multiuser, without NFS

3 – Full multiuser mode

4 – unused

5 – X11

6 – reboot

1. Runlevel: Runlevel programs are executed from /etc/rc.d/rc\*.d/

* POST – Power on self test

>> BIOS initializes tests to make sure devices connected are working all right like printer, keyboard, mouse, speakers

* The third step is BIOS check the hard drive and Boot disk / OS. Once it is confirmed that there is OS, This OS is loaded into RAM. When this process is done we will see the screen.