**what are your daily responsibilities in your last project?**

I am done with my project. It's been two weeks since I rolled of my project.

I was a aws devops engineer. I am individually responsible about 1100 ec2 instances and 300 rds instances I have supported the production infrastructure and CI and CD mostly these are java j2ee applications and Hadoop based applications. so, I have experience automation using aws python boto sdk and templating languages like cloud formation, terraform creating servers in cloud in single shot. I did automation for that type of environment if a company need muti vm environment as a dev environment or qa environment if you run my script it creates in single shot.

some of things technology stack our java services are rewritten into microservices into ecs and node and ui bases applications.

we use message skills like rabbitmq, tibco, as a messaging layer. mostly oracle and mysql in datalayer . I have experience setting up replications in mutiple regions in aws as well. for hadoop and big data applications data from redshift we analyze data from redshift used on hands on level 20-25 aws services. mostly done automation.

**On docker** very comfortable using cluster management tools like swarm, compose. writing compose files to create multi container applications in single shot. I also created ecs clusters in aws. managed container services in aws. I also have experience putting them into the production. my most recent experience as cloud and containers. I started as linux administrator coming from a linux background. dev teams will come and click buttons on my dashboards it would create new aws account for them or I gave various aws permissions for the dev teams members in new dev like environment or qa environment for their application all that magic happens by python web services we built.

**what is devOps?**

DevOps is the combination of cultural philosophies, practices, and tools that increases an organization’s ability to deliver applications and services at high velocity: evolving and improving products at a faster pace than organizations using traditional software development and infrastructure management processes. This speed enables organizations to better serve their customers and compete more effectively in the market.

Under a DevOps model, development and operations teams are no longer “siloed.” Sometimes, these two teams are merged into a single team where the engineers work across the entire application lifecycle, from development and test to deployment to operations, and develop a range of skills not limited to a single function. Quality assurance and security teams may also become more tightly integrated with development and operations and throughout the application lifecycle.

In devops there are many tools it may be aws, azure, or google cloud etc.

Infrastructure as code tools like chef,puppet,salt stack, or ansible too.( Configuration tools)

Containers like kubernetes or mesos and put docker containers into production.

In general company willing to stream line the more releases to happen things like that all of this constitutes emphasize modern defintion of devops.

I am comfortable with aws, ruby, shell scrpting , ansible, docker, and kubernetes.

**Tell me something about your aws experience? what you have done?**

I am a devops and cloud engineer and support teams for the matured model for example I do migrate on prem applications to cloud and comprise whole lot of things includes monolitic applications into dedicated three tier applications you know frond end and backend rather than lift and shift. vpc, ec2, cloud formation, route53.

deploying applications to be the cloud first. deploying directly. my other area of expertise is devops chef, ansible, puppet. I come from both development and infrastructure background. to concentrate on premise and concentrate on docker containers.

I have worked 80 and 20 split in linux and windows.

**can you explain me about chef? ( configuration tools)**

•Proficient level of experience on DevOps essential tools like Chef, Puppet, Ansible, Docker, Subversion (SVN), GIT, Hudson, Jenkins, Ant, Maven.

•Written Chef Cookbooks for various DB configurations to modularize and optimize product configuration.

•Implemented Chef Recipes for Deployment on build on internal Data Centre Servers. Also re-used and modified same Chef Recipes to create a Deployment directly into Amazon EC2 instances.

•Development and version control of Chef Cookbooks, testing of Cookbooks using Food critic and Test Kitchen and running recipes on nodes managed by on premise Chef Server.

•Collaborate in the automation of AWS infrastructure via Terraform and Jenkins - software and services configuration via chef cookbooks.

.Integration of Automated Build with Deployment Pipeline. Currently installed Chef Server and clients to pick up the Build from Jenkins repository and deploy in target environments (Integration, QA, and Production).

**Can you explain you work experience with puppet?**

•Hands on Experience on Puppet server and workstation to manage and configure nodes, experience in writing puppet manifests to automate configuration of a board range of services.

•Deployed puppet for configuration management to existing infrastructure.

•Implemented puppet modules for server housekeeping.

•Actively involved in architecting the puppet infrastructure to manage servers in different environments.

**ANSIBLE**: •Used Ansible and Ansible Tower as Configuration management tool, to automate repetitive tasks, quickly deploys critical applications, and proactively manages change.

**What Is Kubernetes? (containers)**

Kubernetes is an open-source system for automating deployment, scaling, and management of containerized applications. It groups containers that make up an application into logical units for easy management and discovery.

**Can you explain me about JENKINS? ( CICD)**

I am definetly very comfortable with the jenkins . the rest api, automation using jenkins build pipelines in jenkins.I have integrated jenkins with various other tools that need to part of release process for applications that code quality analysis like sonar, hd45 for security analysis. issue management tools like jenkins and jera github on occasion when there is no readymade plugin available.

**what is Docker?( containers)**

In the early stages guys who did poc wrote alot of docker files, understanding docker, shared volumes, compose, swarm, setting custom network configurations for dockerv. my work mainly writing docker files and setting up continuous integration using docker.

With docker I started using poc if docker is good. I configured docker repository and artifactory good with using docker commands and swarm, compose, creating custom network with docker.

==>we use spring boot (Frameworks) inside the docker container.

==>we need to configure spring.io repository, antifactory all the dependency can be pulled. springboot, docker, jar. ==>we need to pass java-jar and entry point.

**Can you explain monitoring framework you worked with or supported? Monitoring tools (Nagios, Splunk, cloud watch)**

I worked on multiple tools using Nagios. on cloud, I have extensive experience on cloud watch for example I wrote Perl scripts that upload custom matrix to cloud watch. for production purpose, we used detailed monitoring in cloud watch. I have experience writing metrics in system level and application using various Nagios plugins. also worked and rollout Splunk in the production environments in previous engagements too. These are the tools I am very comfortable with writing custom queries to mind the logs that are uploaded to Splunk. you I can install and configure.

spunk cluster to search heads, forwarders, and indexers setting all that up for production grade spunk infrastructure.

**Most challenging project in last couple of years? ( kubernetes)**

Basically, one effort we want to use a container framework instead of swarm in docker. and microservices in production so basically, I have setup Kubernetes cluster with core os nodes and all of this five noded cd cluster and all that(KUBERNETES) was in production and gluster like clusters and deployed a few microservices. These all yaml files only but kubernetes have notion, nomenclature like services pods ingres and outgres rules to route traffic to specific container running on a server. we have to configure rules and replication container. It is a challenging project and there is lot of learning for me interms of how the kubernetes works, how service discovery tools work and also how do we apply it for out company that sort of stuff. If you are implementing kubernetes why do you need swarm. swarm and ecs are already in the production now.

**Kubernetes:**

we are trying to see can better up our production rollout evaluate both kubernetes and mesos. poc is considered successful. These are very few technologies. we are investing on our teams to write kubernetes deployments skills and understand integral in writing a kube cluster in production.

we also have actual hardware devices that act as storage cache to applications running in the cloud. If the applications running in the datacentre trying to talk with this running cloud which is already migrated to cloud storage cached item is cached the datacentre the applications are speeded up certain portions of datacentre applications to use storage gateways and storage cache called as gateway cached volumes. so, I have experience setting all those things up.and I also created open vpn between cloud infrastructure and network infrastructure between vm that act like a gateway.

If the service is running in a data centre services in the cloud it would request to this proxy open vpn implemented correctly. These proxy would forward the requests services running in the cloud. once proxy receives responses it would forward back to the datacentre service experiecne setting those type of services.

**what kind of alerts you can setup?**

It greatly depends on the level cloud or datacenter which type applications we are dealing with. I will give a few examples. traditional metrics includes cpu utilization, load averages and ram utilization, disk space utilization.It merely depends on the utilization of data bases.

Most recently I have written a function in **aws lamda** that would connect to a database, which will run a query on that mysql database and we measure the query response time and if the response time more than threshold it would trigger cloudwatch which would in return trigger auto scaling group a new mysql instances would be added from the snapshot of the running rds instance. It is kind of horizontal scaling for databases but involves monitoring for long running quries to the databases. If it is like a webservice have any ui you on occasions I have written custom monitors in nagios that would make restful calls to the api and make sure i am getting a 200 response to the request. If I am getting 200 we know that sure that api is running daemon under service.I have written that sort of stuff it greatly depends on what we are monitoring and where.

I haven't done musch of java development. I have done python webdevelopment. for example I build a dashboard kind of self service for dev in the cloud using django, apache lit, celery, and bunch of other libraries like paramico.

But I been supporting deployment automation java applications I understand what head dums are what heat dums are. read the exception and pinpoint which lines or package jvm is complainning about . and helping debugging java applications.