PHANI INTERVIEW:

Team: Regu, Paul, Maverick

Int: start off with brief description of your background.

P: I can lot of the work that I’m doing here in has been role of Senior DevOps Automaton Engineer I’ll be supporting the works of automation cicd and deployment across xxxx 5.04 and management we have and most of those applications being java xxxx 5.09, web applications, spring-based applications and lot of batch programs. My responsibilities are to be supporting their build processing applications fitting cicd automating their deployments, integrating the test reach, and orchestrating the executing and deployments of the application across the QA, SIP, xxx 5.32 performance and production and lot of that work involves using some of the key technology tool stack like Jenkins for ci, Git for version control, java gt application itself. I come from strong java gt background coding so that makes it easy for me to support the application and containerization is that I’ll be supporting docker and kubernetes that I’m very hands on and especially do the application that I was checking xxxx 6.01 in the system to give it around the jvm’s that are setting on tomcat, jboss and MySQL, oracle and some of the key automation scripting like python, bash and shell and lot of orchestration of containers using docker, kubernetes are my key expertise and especially also some of the application infrastructure that we have is actually running on aws cloud platform that I’m supporting well where I have good experience on AWS wherein some of the key services like ec2, s3, cloud watch, lambda, ELB, auto scaling, etc.,

Int: you said you worked on docker so what kind of orchestration is it deployed on.

P: it is 1.10 version of kubernetes

Int: is it on AWS or where is the kubernetes running?

P: right now, the kubernetes platform the delta microservice is not on cloud yet we have on premises vm ware latex systems where the kubernetes clusters are sitting where we have a plan to migrate to AWS probably take manage of xxxx 7.41 service but this time we are not on cloud on the microservices and kubernetes 1.10 version running on vm ware so it is not accepting our firms.

Int: could you talk about what would a pipeline look like to get your application into docker and then into kubernetes and for using it to further process.

P: 1st sort of pipeline used to start of with git. We use git, bitbucket, jira and to start with the configuration part of pipeline will be I would be dealing with lot of analysis on designing the branching strategies of git workflow to start with filing application deployment ci, so in the ci configuration the 1st part of the configuration describe on the name of the xxx 8.28 branching to be downloaded so in this case the xxxx 8.33 stack is apparently git and followed by certain web hooks that kick offs the Jenkins jobs that would have written for configuring Jenkins and we are actually doing the configuration written in pipeline as go like build apps using xxxx 8.45 so 1st part of configuration I would describe step of build in which I define xxxx 8.50 name, branch name and git publish code needs to be downloaded and next step is actual build script that will be used with its application so for e.g. it could be maven build ,or ant build using which it xxxxxx 9.06 pom.xml, may have to be extended based on xxxxxx 9.09 involved in that application and then followed by it’s a test reach that we would usually consider xxxx 9.14 gi on which it could be using plugins likes mogito plugin, junit plugin probably unit test phase followed by function test using celery plugin using celery server also in certain cases lot of web apps also booked load runner plugin to automate some of the performance test reach of the part of ci. And the next part is code coverage part where we use SonarQube plugin to automate the code that is getting covered and to put a threshold on how much of code is getting covered as part of test reach within the pipeline where lot of application required 95%, 98% of code coverage as which we can’t accomplish that much of code coverage to SonarQube beside more test reach complete the level of threshold the team is looking up for within SonarQube and then followed by security analysis on code coverage by using tool “xxxxx” 10.9 to identify security vulnerabilities followed by test reach that would require how we are going to compose the artifact and stage it on some artifactory location. During all these tasks with regards to microservice one another xxxxx 10.33 that we do is using the docker plugin. I would write docker file and put it in the same place where the git code is located on application branch and docker file will be sitting wherever Jenkins file is there of the application. When application code is downloaded from directory of Jenkins the docker file will also be downloaded so the docker plugin knows where to lookup and integrate docker file to build the image for the xxxx 10.55 so the artifact is mapped and sitting the docker file that will be referred to. And once the docker plugins kicks of the docker build image by looking the docker file is going to push the docker image to be docker trusted registry and forward by stamp process called switched log used to stand image itself to identify any security vulnerability then followed by there is a webhook between the docker trusted registry and kubernetes platform that’s used to kick off the deployment of the image at the container based on the deployment of yaml file that should define between the different platform so that this container will be actually running as a service the way you intend thru to and the way you want to orchestrate it based on the definition of the pods with that you define in the kubernetes using deployment object yaml file so that kind of interval pipeline a lot of stack is driven thru lot for the xxxxx 11.41 whether you want to deploy it on to QA/SIP/UAT in which case you send the namespaces or workspaces to identify the configuration that could vary but we meet that those configuration parameters are loosely coupled to the pipeline so we don’t have to monitor the pipeline code to deploy to different environments.

Int: how do you handle the configuration differences between the environments?

P: they are put together in the form of namespaces and environment variables so the key to each of the environment variables going to be the same references to the keys will stay the same within the pipeline code, but the values reflected based on the environment it’s going to be deployment. so, we have different tag inventory for QA/SIP/UAT and the pipeline will resolve those values for those environment keys with the xxxxxxx 12.39 for some of them its DB for some of them its namespaces and for some its property space like that.

Int: could you provide what the local preference would be for something running like kubernetes like what precedence for where some of your environments where would your settings come from? What’s the sequence?

P: thru the ENV-when your specifying the step: when your writing the pod schema you’re going to specify the xxxx 13.35 that containers and the image and all of the info and you have that ENV and in the ENV pod which is gong to refer to key and value pairs and some values like namespaces, each of those environment variables are declared by using config. Each value is referred to all variable names that is the key that is defined in the ENV

Int: can you talk about where do you inject it in kubernetes?

P: in kubernetes when we are deploying a container that run at the service on a pod, we would define a pod schema and deployment object yaml file. A pod schema would contain name, method data and speck that would contain specifics of the container, the image and tag and its going to have an ENB which is going to have references to config map that will reflect the actual key value pairs that you want the pod schema to refer to based on the environment in which the pod image is going to be executed. So the pods should map if you create the config map manually by scripts like kube-ctl command where you give the name of the map like data source to it for the key. For each key the bandwidth would be set of more dynamically outside of the yaml file itself but the pod schema that you write itself would be the config map that you define itself is going to be be referred with in the metadata by the so when you get a name to the particular config map we are going to give a kind of config & meta data and we are going to refer the data property . the data itself is data property & spring property to have reference within them to create the config map.

Int: how with you handle with XXX script?

P: something called secret objects in kubernetes. A large of the information is kind of concealed by an object of type secrete and they are pretty much integrated to old data base access ad user information access or anything involving as per of the code.

Int: in your Pipeline where does the secret originate from?

P: to the secret yaml file we can create a secret yaml file and we have reference that yaml file within the file for al the data you are pulling from the secret yaml file. 1st pull the yaml from secret and then you will have the reference from your secret object from yaml file. Reference from the key double flower bracket.

Int: let’s say you have a secret and how can a kubernetes will see as a yaml file with actual encrypted content in the pipeline from where it comes from typically how it’s done differently how its regenerated?

P: yaml file is specified within when I described webhook within the dtr and kubernetes platform that’s going to be fired, that is happening on the docker dtr- say every time push of an image happening on the dtr as per the pipeline that describes after docker plugin, git of the docker image by looking at the docker file and pushed the image dtr after the slot is complete its going to fire the web hook that will be actually pointing to the pod schema that you define within the kubernetes platform and that the pod schema is the one which will have the reference to the secret so within the pod schema of the kind pod specify the speck and the secret and in that secret it’s going to have secret name and actual name of the secret the key value pair in the yaml.

Int: In the git repo is it rested in plain text?

P: no, it’s nothing to do with git its something between dtr and webhooks between the dtr and kubernetes.

Int: let me know I’m understanding correctly that someone with secrets and person trading with secrets and therefore you know xxxxx 21.25 with kubernetes someone has the trade the state of it.

P: of course, someone already created it you’ll be using kube-ctl command but references to those are being made dynamically in the pipeline. The creation is not happening in the pipeline now.

Int: you have any experience working with something that xxxxxxx 21.45

P: I just used kubernetes platform itself, but I have experience with cyber alt for lot of secret management.

Int: describe your back ground on chef and ansible.

P: with ansible I am automating configuration management especially middleware and deployment on the tomcat servers and lot of the automation with regards to cicd is hooked up to apps to automate the deployment of java artifacts on group of inventories like 100’s of servers where certain application that are deployed on app servers and web servers. I used to write playbooks and ansible rules for the application they were to define the pipeline of cd and chef work part was more into infrastructure side, I can say its also used for configuration management but more of it is related to infrastructure specific patch management and writing cookbooks and most of it are also related to middleware more driven to infrastructure rather than application side where I used to maintain chef cookbooks and recipes to automate configurations for tomcat servers, apache and aws etc.., but more focused on other department as well but that was in previous project.

Int: how would you describe between ansible vs chef?

P: ansible is more of push-based tool it doesn’t need agent as long we have python, we can fetch, push the package and force the package to distribute and bring the node to desired stage that define to playbooks and coming to chef it requires agent called chef client. Here we need lot of processing and node converging techniques are kind of established by the chef client run between chef server and composes the manifests catalog and pushes back to the chef client to execute it and bring node to desired state. Performance wise when we compare chef is bit faster and to say chef has lot of good frameworks to support unit & integration testing of chef cookbooks like it supports test kitchen, inspec& chefspec for integration testing and unit testing and it targets way to compose own cicd and web by integrating chef code with git by using chef automate.