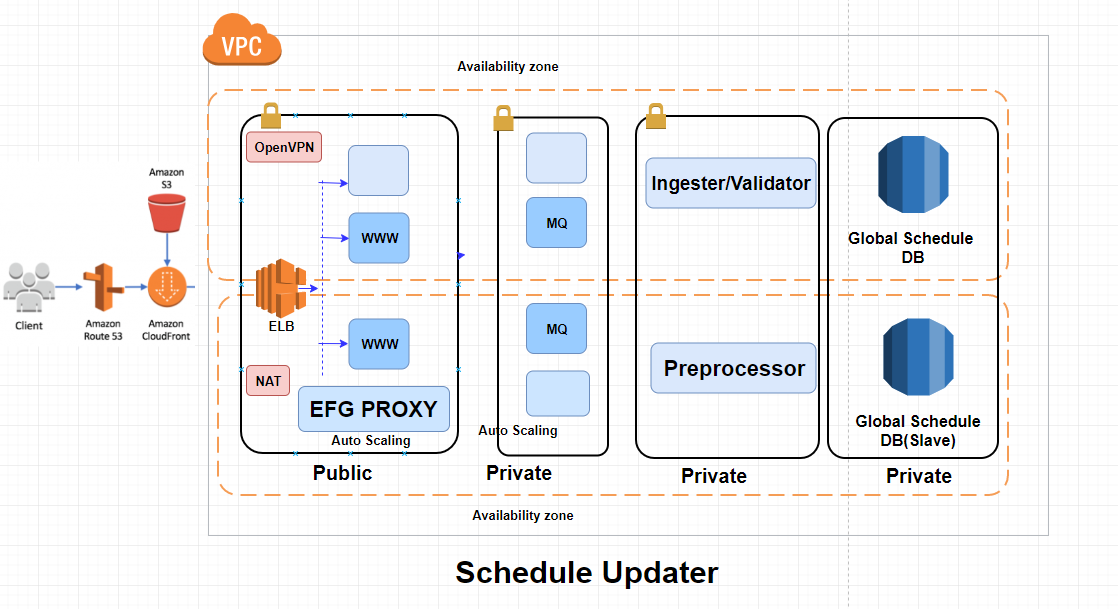
**How is your project Architecture?**



* I am working on a project named as Schedule Update and its infrastructure is on AWS.
* Schedule updater systems are dealing with flight schedules, that we receive from Airlines. We receive content in the form of file like SSM (Standard schedule message) files.
* EFG PROXY is a service which receive the files and do some preprocessing and publish it MQ server.
* Then the Ingester service takes the files from MQ and perform some validation for SSM file content and create instructions. Then these instructions are pass to pre-processor for some pre-processing.
* Then the pre-processor service will compare the new schedule with local schedule and generate the final instruction and push it to Global schedules database.
* SSM: Is a standard schedule message. It’s an industry format defined. Any Airlines who wants to publish their schedule, so they sent it an SSM format.
* It’s a pure text. So, Airlines FTP these files to our Sabre System.

**Where you are deploying?**

As of now they have deployed on physical hardware.

So, do you deploy the application?

Yes, we do, we know the recipe to deploy the application. So, when developer face issue after troubleshooting in order to verify everything we deploy the application.

**Where you are using Ansible.**

We use Ansible for deployment of application. We have a tool called SSPT when we integrate this tool with many application We have to modify the ansible playbook **project\_specific.yaml**.

And we have to write playbooks and shell script as per application requirement and call it in main playbook, like

- Checking Application status

- Deploying the build

- Starting the test

**What s chpasswd**

passwd: Changes the password for an existing user.

chpasswd: Reads a file of login name and password pairs and updates the passwords.

RUN echo "root:abc.123!" | chpasswd

Q) If dev team want to change the payload, then we are updating the Jmeter Image and pushing it to respective openshift project repo.

**Q3. How to Create Docker Image:**

- We have standard Docker file which install all required packages and mount cfg and logs directory.

- from Docker file we create an Image after that a container.

- Then we copy the required cfg file for new carrier as well as the bash script to the config directory.

- commit the container with carrier name.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Pull base image

FROM centos:latest

#Install Package

RUN yum update -y

RUN yum install openssh\* nfs\* nis\* java\* -y

# Install Java.

RUN yum install -y openjdk-6-jdk

#Mount Volume

VOLUME ["/project/plab/ssi/cfg", "/data"]

#Mount Logs Volume

VOLUME ["/project/plab/ssi/logs", "/logs"]

# Define working directory.

WORKDIR /data

# Define commonly used JAVA\_HOME variable

ENV java="./java"

#Set the NIS Server Name on Client

RUN echo "NISDOMAIN=sabrenis.com" >> /etc/sysconfig/network && echo "domain thegeekstuff.com server prod-db" >> /etc/yp.conf

#Start the ypbind on Client

RUN service ypbind start && service ypxfrd start && chkconfig ypbind on && chkconfig ypxfrd on

# Define default command.

CMD ["bash"]

USER hybfun

ENTRYPOINT ["/bin/bash", "--", "nohup $java -jar ldfw.jar cfg/test\_Main\_2\_Avlxml2\_NEW.cfg > logs/test\_Main\_2\_Avlxml2\_NEW.new.log 2>&1 &"]

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Q4. Issue1: When running docker image fails reporting a “bad\_alloc” message**

**Description:** when the docker image is running a command fails reporting a bad\_alloc error message.

**Solution:** the problem likely occurs because the image is running with less memory than the one it requires. You can increase the memory by configuring the dev or default virtual machine on the Virtual Box application.

**docker logs –tail CONTAINERID**

--tail 🡺 Number of lines to show from the end of the logs

**Q5. Ctrl+C takes long time to stop the containers**

That’s because signals from the shell are not properly sent to your docker container. Just use dumb-init. It is a great, small wrapper over your executable, and properly forwards all signals to your command.

# Simply Change:

CMD ["npm", "start"]

# To:

CMD ["dumb-init", "npm", "start"]

**1) How you are taking Backup of your RDS**

Automated backup is configured which is taking snapshot every day at 8:30 AM.

Retention period: How many days backup you keep.

Here we are having retention period 1 Month.

**2) How you are restoring Backup.**

If any DB get corrupt just restore it from snapshot by specifying the instance Type and keep the DB instance name same as it was, so that it will create same end point.

**3) What kind of issue you are facing.**

Previously we used to take backup in every 6 hrs. When we increased the frequency of taking back up to 4 hr then we started getting an error **"Snapshot quota exceeded".**

After googling we come to know that by default AWS allow 100 snapshots per instance. So, we reduce retention period from 30 to 15.

**Q 4) In all of our production server we have enable cloud watch to monitor memory available, disk space use and memory use.**

**Q 5) What major issue you have faced till now.**

**Issue:** Recently we faced an issue with our SSL certificate. PROD Application websites were showing security message while opening through Google Chrome 70.

**Reason:** Google Chrome version 70 was not trusting Symantec SSL/TLS certificate and don't allow the Application site to open unless the new certificate is installed. We were using Symantec SSL Certificate.

Then we created new CSR and passed it to Network team. Then they contacted Symantec support and regenerated the certificate. They returned "Root certificate, Intermediate, Certificate chain"

Then we updated the certificate in Listener of ELB.

* + - 1. Certificate Body
      2. Certificate Private
      3. keyCertificate chain

**Q 6) How you are deploying the application build.**

Blue/Green Deployment

**Q 7) If any deployment is getting failed, then how you are reverting back.**

**Q 8) What is the common problem which you face in the day to day activity.**

The instance will go out of service all of sudden. In this case we have to do the 3 checks. That is

1. Check whether your instance is UP and Running or not.
2. Check on which port Health is happening. Suppose if it is 80 then go and check if the port 80 is open in security group or not.
3. Check within the server services are running or not.

**Day to Day responsibility**

**Q) We get a request that a website is down.**

* 1. First, we check the certificate validity in certificate manager. If all are good, then
  2. We go to the target group of the respective load balancer of the URL and reset the stickiness to 1 day.

By default, a Classic Load Balancer routes each request independently to the registered instance with the smallest load. However, you can use the *sticky session* feature (also known as *session affinity*), which enables the load balancer to bind a user's session to a specific instance.

**Q) Getting request to assign IAM Roles for instance in Bucket policy.**

* Go to the Bucket --> Policy --> Edit the the json file with IAM roles and instance id

"iam::S3access::instanceid"

**Q) Getting request to provide access to Bucket of one Account to Bucket of another account.**

- S3 --> Permission --> ACL --> mention the canonical ID and provide the specific access permission (List, write, read)

**Note:** There is an 12 digit AWS account ID which we call canonical ID

**Q) Checking Logs in cloud watch**

cloudwatch --> Logs --> log group --> create log group

**Note:** Mention the log group name in "Log Configuration" section of ECS to log

**Note:** In Log group there is a log stream which keep the logs minute wise

**Q) We get a request from Director to create SNS topic.**

**Q) Getting request to create SQS. We create SQS with below details and give the URL to requester.**

- Queue Type: FIFO / standard

- Message Retention period: 4 days

- Max message size: 256 KB

**Q) Sometimes we get request in change of SQS like retention period and message.**

**Q) some time we get a request to purge(empty) a Queue.**

- We get Queue name from the requester

- Go to SQS --> Queue --> action --> purge Queue

**Q) Create user in prod DB for read-only access.**

**Q) create IAM roles for a user.**

We get a request to create a IAM user and mirror the roles from existing user.

So, we check the exiting user and confirm which role has assigned to it.

Create a user and assign same role to him.