**DevOps SRE Challenge -1**

**Designing the ELK cluster on AWS**

We can have multiple ways to achieve the same, like creating EC2 instances within VPC with proper subnets and install various components manually. However, I will like to consume ElasticSearch service offered on AWS as it will save time to setup everything from the scratch, and will ensure the scalability and load balancing and other critical design aspects on its own.

**Initial steps:**

Create the VPC with subnet so that I can place ElasticSearch nodes for internal setup consumption only. If required, will setup the NAT Gateway for Internet access.

Create the IAM role for the access on cluster, will have separate role for admin access.

* How to provision the cluster in 30 minutes

Will go to AWS, and select the ElasticSearch service, and use the Production deployment, so that I can cover multiple AZ.

I will choose the size of master node, elastic search node and will also ensure encryption for security of data in transmits.

Will use Amazon Cognito for Authentication if no constraints are there, else organization’s AD accounts can also serve the purpose

* How to scale up or down the cluster

ELK stack will be continuously monitored using cloudwatch and enhanced monitoring. I will setup the action on the basis of cloud watch alarm and use step up scaling. So if load is tremendous, based on the CPU/Memory(through additional monitoring) will trigger the scaleup and eventually scale down.

* How to prevent unauthorized access to the cluster

I will select domain access template to specific IP only, so this will ensure that only business users can access the dashboard. However, for that they need to ensure the static IP and that could be a cumbersome.

So, I may consider the IAM roles and give access to specific IAM groups who are authorized. These IAM users could be AD members which are integrated with AWS using SAML

* How to ship logs from application to the elasticsearch cluster

We will install logstash agent on all Ec2 instances or even RDS or any service we are consuming on AWS. Once the logstash agent is installed it could be configured to push data to ElasticSearch server.

**Few points in my design:**

If log stash agent is installed on Ec2, same could be leveraged for shipping all logs to ElasticSearch, be it OS logs likes /var/log/messages or the application logs or the container logs running on said node. (We have container logs on /var/lib/docker/..) so this will save the efforts for installing the agent on containers.

Agent installation is through yum and manual configuration, but I will put these steps in post build action in my Ec2 build template(or in CloudFormation template) so that whenever any new Ec2 instance being launched it will automatically have all the required logs configured along with agent.

* How to monitor the performance of the cluster

Using cloud metrics dashboard, I will use the cloud watch will monitor the cluster from resource utilization perspective. Also, as I will be planning kinesis as a data stream so its size will be another key aspect of monitoring.

Replication, crosszone availability, at each and every layer there will be redundancy to avoid single point of failure.

* What about if the applications are running in multiple regions, (one system in us-east-1 and one system in us-west-2)

Ideal solution will be to have dedicated ELK cluster for single region. However, if situation demands, I will send all the data to S3 storage, which could leveraged by the services on other region and this way data from multiple region could be merged and we can consolidated view on KIBANA dashboard. Below diagram depicts my idea

S3 Storage

Region 1

ELK cluster will also receive data from multiple other sources like logstash of API maybe

Kibana dashboard

ELK cluster

Kinesis work stream

Region2