AWS Elastic Beanstalk

What is Elastic Beanstalk ?

- It help you to deploy and manage your application quickly
- No need to focus on the underlying infrastructure rather focusing on the application code development
- It's a developer centric PaaS service for deploying applications on AWS
- You just need to upload your application code and it will automatically deploy your application
- Reduces management complexity without compromising on choice and control
- Load balancing, scaling and application monitoring can be handle from a single view

Components of Elastic Beanstalk

- Application :
    - Logical collection of EB components which includes
        - Environments
        - Versions
        - Environment configurations
- Application Version :
    - It an application source bundle or a deployable code (e.g. war. zip)
    - An application can have multiple versions and each versions are unique
- Environment :
    - Collection of AWS resources running an application version
    - Each environment runs only one application version at a time
    - You can run same or different application version in multiple environments simultaneously
    - When you create an environment EB provisions the resources needed to run the application version you specified

Work flow of the EB

- Create application -> Upload code -> Launch environment -> Manage environment
- You can update version
- You can deploy a new version

Pricing :

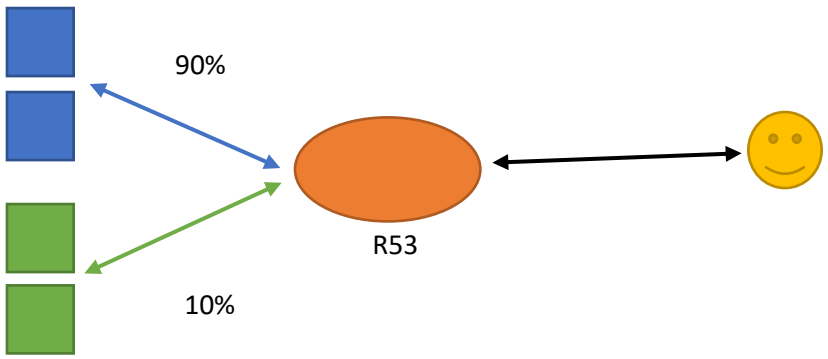
- There is no additional charge for using EB
- You will be charged only for the underlying resources that you provisioned to run your application

Deployment Options for Updates :

- All at once (deploy all in one go) :
    - Deployment will be faster but it require downtime as instances will be down for some time during the deployment
    - Suitable for lower environments

    - No additional cost
- Rolling :
    - Deployment will be done in batched with few instances at a time, and move to the next batches when first batch become healthy, so it doesn't required any downtime
    - Reduced the capacity or number of instances which may reduce the performance
    - Full deployment will take time
    - You can run both versions of application simultaneously
    - No additional cost
- Rolling with additional batches
    - It's similar to rolling but it first spin up new instances to deploy the updates so that it make sure that old application is still available
    - You won't lose capacity so no compromise on performance
    - Deployment will be longer
    - Both the application version can run simultaneously
    - Additional batch will be removed at the end of deployment
    - Good for production workload
    - Incur additional cost
- Immutable :
    - Spin up new instances in a new ASG, deploy new version to those instances and then swaps all the instances when everything is healthy
    - It will create a temporary ASG and launch same number of instances as existing, then it include new instances from temp ASG to existing ASG, then it will terminate old version instances and delete temp ASG
    - You can run both version simultaneously
    - Longest deployment among all
    - Zero downtime
    - Quick rollback in case of failure
    - Good for production environment
    - Incur additional cost

Blue/Green Deployment :



- Blue environment is the existing environment with v1 application
- Green environment is the new environment created with v2 application

- Swap the URLs between two environments
- Increase availability and reduce risk
- For doing the practical deployment you need to follow the below steps
    - Create one record set on route 53 for blue environment with weighted policy with 90% as value
    - Create second record set on route 53 for green environment with weighted policy with 10% as value
    - Then go to the green environment and swap the URL with blue environment and wait until the green env URL point to the blue env URL
    - Then terminate the blue environment, so entire traffic will go the green environment