Deploy a PHP Application using AWS Elastic Beanstalk

Ananda D. R.

Intro!

- Cloud Engineer @ Btech
- Software Engineering Student @ UGM
- AWS Community Builder
- FLOSS Enthusi ast
- https://linktr.ee/misskecupbung

Agenda

- Intro!
- AWS Elastic Beanstalk
- Components
- Environment Tiers
- Deployment Options
- Monitoring and Health Checks
- Diagram
- Hands-On Lab
  - Create a new App
  - Manage and Monitor
  - Update the app, v1 → v2
- References

Pre-Requisites

- Fundamental, Cloud + AWS
- Terms: instance, load balancing, security group, monitoring, logs,
- AWS Services: EC2, VPC, Route53
- PHP

AWS Elastic Beanstalk

- *“Easy way to deploy and scale applications”*
- Compatible with:
  - Package builder
  - Single container Docker
  - Multicontainer Docker
  - Pre-configured Docker
  - GO
  - Java SE
  - Java with TomCat
  - .NET on Windows Server with IIS
  - Node.js
  - PHP
  - Python
  - Ruby
- FREE

Components

- **Application version**
  - Reference to section of deployable code.
- **Environment**
  - Refers to an app version that has been deployed on AWS Resources
  - Comprised of ALL the resources created by Elastic Beanstalk
- **Environment configurations**
  - A collection of parameters and settings. Using YAML/JSON, saved with .config and stored within the .ebextenstions.
- **Environment tier**
  - Reflects on how Elastic Beanstalk provisions resources based on what application is designed to do

Components

- **Configuration template**
  - This is the template that provides the baseline for creating a new, unique, environment configurations
- **Platform**
  - Culmination of components in which you can build your application upon using Elastic Beanstalk. E.g: OS, Language, service type, and Elastic Beanstalk itself.
- **Applications**
  - A collection of different elements, such as environments, environment configurations and application versions.

Environment Tiers - Web Server Tier

- **Route 53**
  - Direct web traffic to the right servers > URL
- **Elastic Load Balancer**
  - Automatically distributes incoming application traffic and scales resources to meet traffic demands.
- **Auto Scaling**
  - Manage the capacity planning of your applications based on the load received.
- **EC2 Instances**
  - Part of the auto scaling group.
- **Security Groups**
  - Allows port 80 to be open everyone.

-

Environment Tiers - Worker Tier

- **AWS SQS Queue**
- **Auto Scaling**
  - To ensure that performance isn't impacted based on load
- **IAM Service Role**
  - To allow your EC2 instances to monitor queue activity in the SQS Queue

Deployment Options

- **All at once (Default option)**
  - If you needed to update your app, using the all at once option will simply roll out the application to your resources all at the same time.
- **Rolling**
  - Minimise the amount of disruption that is caused
  - Deploy your app in batches
  - Perform the update on the next batch

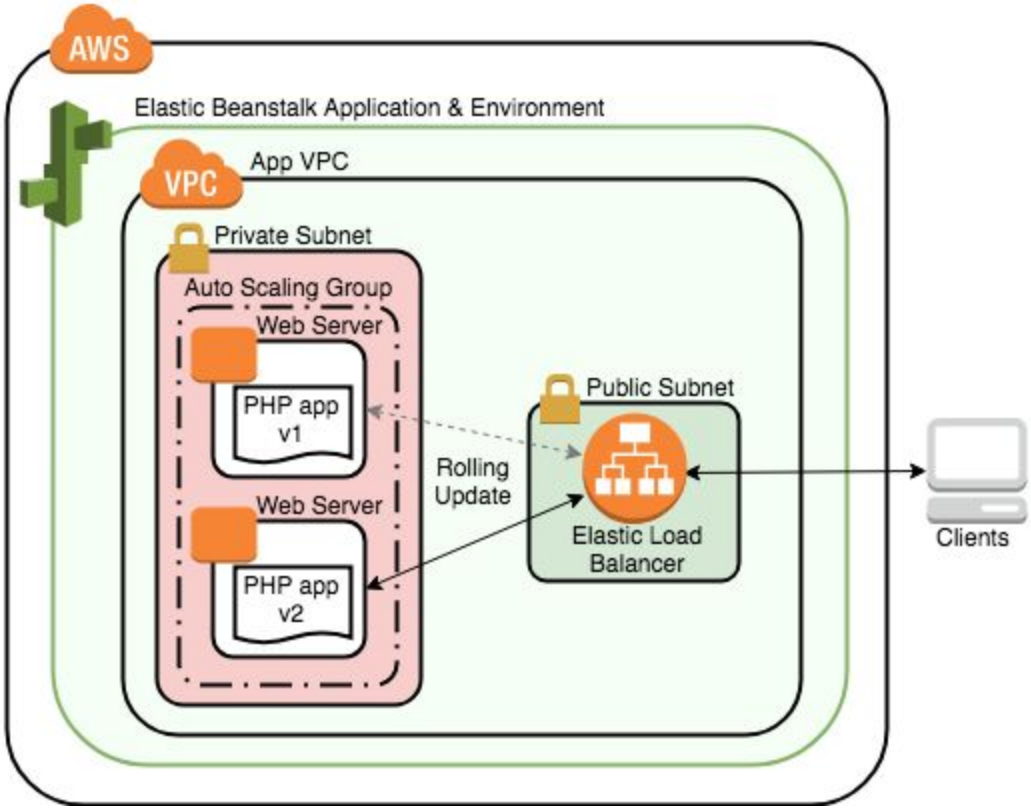
Deployment Options

- **Rolling with additional batch**
  - Environment is updated in batches until all your resources have the new update
  - Adds another batch of instances within your environment to your resource pool to ensure application availability is not impacted
- **Immutable**
  - Create an entirely new set of instances
  - Served through a temporary auto scaling group behind your ELB
  - The old env would be removed and the autoscaling group updated

Monitoring and Health Checks

- **Basic Health Reporting**
  - Any resources running in your environment will send metrics to Amazon Cloudwatch in five minute intervals
  - Health check every 10 sec
  - System status check & Instance status check
  - Common issues: incorrect network configuration, corrupt file systems, exhausted memory, and incompatible kernel
- **Advanced Health Reporting**
  - Messages: OK, Warning, Degraded, Pending, Unknown, Suspended
  - the health agent probes the instance at a deeper level and more frequently over Cloudwatch than the basic

Diagram


AWS
Elastic Beanstalk Application & Environment
VPC
App VPC
Private Subnet
Auto Scaling Group
Web Server
PHP app v1
Web Server
PHP app v2
Rolling Update
Public Subnet
Elastic Load Balancer
Clients

Hands-On Lab: https://s.id/OYNN

References

- https://docs.aws.amazon.com/elastic-beanstalk/index.html
- https://docs.aws.amazon.com/elasticbeanstalk/latest/dg/Welcome.html

Thank you