AWS Certified Cloud Practitioner Training Bootcamp

Design Principles – Disposable Resources

Disposable Resources Instead of Fixed Servers

- What exactly does *Disposable* mean ?
- Disposable resources mean temporary resources, easy available; one time use resources
- In traditional data centers environments, you work with fixed resources or servers; this translates to you as high upfront costs and a time to production high as well
- Within AWS, you launch as many servers as you need, use them as long as you need them and pay accordingly

Disposable Resources Instead of Fixed Servers

- Other aspects to note in regards to fixed servers :
  - configuration drift and immutable infrastructure

- Configuration drift – configuration changes and software patches can be applied inconsistently; this leads to different configurations (maybe untested) on your resources in the DC

- Immutable infrastructures – this can solve the previous issue; instead of patching and modifying initial configuration on your servers, just change the old server with a new one that has new software packages applied

Infrastructure Instantiating - Automation

- Manually setting up your infrastructure is time consuming and is also error prone
- Ideally, any new environment setup or scaling up existing infrastructure should be done automatically
- In AWS, you can use *Bootstrapping* or *Golden Images*, or both at the same time

Infrastructure Instantiating - Bootstrapping

- When you launch an EC2 AMI, the instance starts with default configuration

- Want to configure the EC2 instance as a web server ?
  - sudo su
  - service install httpd
  - service httpd start
  - cd /var/www/html
  - nano index.html

- Run a bootstrap script automatically for all of the above

Infrastructure Instantiating – Golden Image

- When your web server is ready, up and running, with all security and OS patches applied, you can create a snapshot of this EC2 instance

- The snapshot (golden image) may then be used in order to create an AMI

- The AMI could be used for example in an Auto Scaling group, so that resources sustaining your app can scale up or down, as needed

AWS Certified Cloud Practitioner Training Bootcamp

Thank you