

AWS Certification – Management Tools – Cheat Sheet

CloudFormation

- gives developers and systems administrators an easy way to create and manage a collection of related AWS resources
- Resources can be updated, deleted and modified in an orderly, controlled and predictable fashion, in effect applying version control to the AWS infrastructure as code done for software code
- CloudFormation Template is an architectural diagram, in JSON format, and Stack is the end result of that diagram, which is actually provisioned
- template can be used to set up the resources consistently and repeatedly over and over across multiple regions and consists of
 - List of AWS resources and their configuration values
 - An optional template file format version number
 - An optional list of template parameters (input values supplied at stack creation time)
 - An optional list of output values like public IP address using the Fn::GetAtt function
 - An optional list of data tables used to lookup static configuration values *for e.g., AMI names per AZ*
- supports Chef & Puppet Integration to deploy and configure right down the application layer
- supports Bootstrap scripts to install packages, files and

services on the EC2 instances by simply describing them in the CF template

- automatic rollback on error feature is enabled, by default, which will cause all the AWS resources that CF created successfully for a stack up to the point where an error occurred to be deleted
- provides a *WaitCondition* resource to block the creation of other resources until a completion signal is received from an external source
- allows *DeletionPolicy* attribute to be defined for resources in the template
 - retain to preserve resources like S3 even after stack deletion
 - snapshot to backup resources like RDS after stack deletion
- *DependsOn* attribute to specify that the creation of a specific resource follows another
- Service role is an IAM role that allows AWS CloudFormation to make calls to resources in a stack on the user's behalf
- support Nested stacks that can separate out reusable, common components and create dedicated templates to mix and match different templates but use nested stacks to create a single, unified stack

Elastic BeanStalk

- makes it easier for developers to quickly deploy and manage applications in the AWS cloud.
- automatically handles the deployment details of capacity

provisioning, load balancing, auto-scaling and application health monitoring

- CloudFormation supports ElasticBeanstalk
- provisions resources to support
 - a web application that handles HTTP(S) requests or
 - a web application that handles background-processing (worker) tasks
- supports Out Of the Box
 - Apache Tomcat for Java applications
 - Apache HTTP Server for PHP applications
 - Apache HTTP server for Python applications
 - Nginx or Apache HTTP Server for Node.js applications
 - Passenger for Ruby applications
 - MicroSoft IIS 7.5 for .Net applications
 - Single and Multi Container Docker
- supports custom AMI to be used
- is designed to support multiple running environments such as one for Dev, QA, Pre-Prod and Production.
- supports versioning and stores and tracks application versions over time allowing easy rollback to prior version
- can provision RDS DB instance and connectivity information is exposed to the application by environment variables, but is NOT recommended for production setup as the RDS is tied up with the Elastic Beanstalk lifecycle and if deleted, the RDS instance would be deleted as well

OpsWorks

- is a configuration management service that helps to

configure and operate applications in a cloud enterprise by using Chef

- helps deploy and monitor applications in stacks with multiple layers
- supports preconfigured layers for Applications, Databases, Load Balancers, Caching
- OpsWorks Stacks features is a set of lifecycle events – Setup, Configure, Deploy, Undeploy, and Shutdown – which automatically runs specified set of recipes at the appropriate time on each instance
- Layers depend on Chef recipes to handle tasks such as installing packages on instances, deploying apps, running scripts, and so on
- OpsWorks Stacks runs the recipes for each layer, even if the instance belongs to multiple layers
- supports Auto Healing and Auto Scaling to monitor instance health, and provision new instances

CloudWatch

- allows monitoring of AWS resources and applications in real time, collect and track pre configured or custom metrics and configure alarms to send notification or make resource changes based on defined rules
- does not aggregate data across regions
- stores the log data indefinitely, and the retention can be changed for each log group at any time
- alarm history is stored for only 14 days
- can be used an alternative to S3 to store logs with the ability to configure Alarms and generate metrics, however

logs cannot be made public

- Alarms exist only in the created region and the Alarm actions must reside in the same region as well

CloudTrail

- records access to API calls for the AWS account made from AWS management console, SDKs, CLI and higher level AWS service
- support many AWS services and tracks who did, from where, what & when
- can be enabled per-region basis, a region can include global services (like IAM, STS etc), is applicable to all the supported services within that region
- log files from different regions can be sent to the same S3 bucket
- can be integrated with SNS to notify logs availability, CloudWatch logs log group for notifications when specific API events occur
- call history enables security analysis, resource change tracking, trouble shooting and compliance auditing