CloudFormation Section 3 Stack-In-Depth

**Introduction to Stacks:**

**Stacks:**

* A collection of AWS resources that you can manage as a single unit
* You can create, update, or delete a stack
* Resources in a stack are defined by the stack’s CloudFormation template
* All of a stack’s resources must be created or deleted successfully for the stack to be created or deleted
* If a resource can not be created, a stack will be rolled back
* If a resource can not be deleted, any remaining resources are retained until the stack can be deleted successfully
* Work with stack by using the CloudFormation Console, CLI, or API

**CloudFormation Console:**

* Create, monitor, update, and delete stacks
* Open the AWS CloudFormation console directly with the URL or from the console services menu
* If there are no stacks running, you are presented with the option to create a stack. Otherwise, you see a list of your current running stacks.

**Creating a stack:**

* Start the wizard
* You can select a template
* Input the templates parameters
* Set stack options
* Review your stack
* Then create the stack
* After creating a stack, you can monitor the stack’s progress, view the stack’s resources and outputs, update the stack, and delete it.

**Protecting your stacks:**

**Production stacks need protection:**

* You have production stacks, multiple architects with admin access to those stacks. How do you protect them from accidental deletion or update?
* 4 ways
* Termination protection, stack level policies, resource level policies, IAM policies

**Termination Protection:**

* For stacks that hold critical resources you can apply termination protection
* Enable termination protection while creating a new stack
* Delete actions against the stack will be denied
* You can enable this via the management console, cli, or api
* After you create your stack, termination protection is in the overview section of your stack
* For nested stacks, termination protection cascades down to the sub-stacks
* Termination protection can be removed so manage this privilege with IAM

**Stack Level Policies:**

* When you create a stack, all update actions are allowed on all resources
* Stack policies can prevent accidental update or delete of stacks
* A stack policy is a JSON document that defines the update actions that can be performed on designated resources
* After you set a stack policy, all of the resources in the stack are protected by default
* Set an explicit allow on resources that you want to allow updates on
* You can define only one stack policy per stack
* This policy applies to all users trying to use that stack. Update that stack

**Resource Level Policies:**

* Deletion Policy- Used to preserve or backup a resource when its stack is deleted
* Specify a DeletionPolicy attribute for each resource that you want to control
* A resource is deleted by default without the deletion policy
* Use Retain to keep a resource after the stack is deleted
* Specify snapshot to have CloudFormation create a snapshot before deleting the resources
* DeletionPolicy options – Delete, Retain, Snapshot

**Stacks in Detail:**

**Review Stacks and Estimate Cost:**

* The final step before creating your stack is to review the stack
* You can click the Cost link to estimate the cost of your stack
* Create your stack – initially will display CREATE\_IN\_PROGRESS
* Use the stack detail pane to vie details of stack creation
* It is instructive to watch stack creation to understand the step by step process
* After the stack had been successfully created, its status changes to CREATE\_COMPLETE
* Click the Outputs tab to view your stacks outputs

**Change Sets:**

* Preview how a stack will be configured before creating the stack by creating a change set
* Review your stack and make corrections and changes to your stack before executing the change set
* Change set is basically a template
* You can also specify new input parameters
* More on change sets later

**Viewing Stack Data:**

* After stack creation, use the Management Console to view its data and resources
* Outputs that were declared in the stack
* Resources
* Events- operations that are tracked when you create, update, or delete the stack
* Template
* Parameters
* Tags
* Stack policy – stack resources that are protected against stack update. To update these resources, they must be explicitly allowed during a stack update.

**Stack Deletion:**

* Stack status proceeds to the DELETE\_IN\_PROGRESS state
* After deletion starts you cannot abort it
* Stack status changes to DELETE\_COMPLETE after deletion finishes
* Stacks in the DELETE\_COMPLETE state are not displayed in the console by default
* You can view deleted stacks, but not by default

**Rollback Triggers:**

Integrate CloudWatch into your stack so you can monitor your stacks and setup alarms for your stacks during update to stacks. If these alarms are triggered, a rollback starts.

**Rollback Trigger Details:**

* Allows you to integrate application- and resource-level alarms from Amazon CloudWatch into the update process for your stacks
* If a change to the stack causes any of the registered alarms to fire, CloudFormation immediately stops the update and rolls back to the last good state.
* You can include a monitoring window after all updates are complete to allow additional time for the change to stabilize
* This window happens prior to the CloudFormation cleanup phase, allowing attribute changes and replaced resources to be quickly restored
* When you create a rollback trigger, you need to specify the CloudWatch alarm that CloudFormation should monitor
* CloudFormation monitors the specified alarms during the stack create or update operation
* You specify the amount of time after all resources have been deployed for the rollback triggers to be in effect
* If an alarm goes to ALERT state during the stack operation or the monitoring period, CloudFormation rolls back the entire stack operation
* If there are no alarms that go to ALERT state, CloudFormation proceeds to dispose of old resources as usual
* By default, CloudFormation only rolls back stack operations if an alarm goes into an ALERT state

**Using AWS Config to Monitor Stacks:**

Track the current and historical configurations of your CloudFormations.