Table of Contents

Using CloudFormation in AWS Console

AWS CloudFormation Designer

- `AWS CloudFormation Designer` is a graphic tool for creating, viewing, and modifying AWS CloudFormation templates.

- With Designer you can diagram your template resources using a `drag-and-drop` interface.

- You can edit their details using the `integrated JSON and YAML editor`.

- AWS CloudFormation Designer can help you see the relationship between template resources.

- Navigate to CloudFormation Service > `Create Stack` > On the Select Template page > `Upload a template file` > `Select the YAML/JSON Template file`.

- Once you upload the template, this template files get uploaded in a default S3 bucket.

- To view the resources that will be created using this Template, you can click on the `View in Designer`.

- Review the graphical representation of the environment that will be created including the template in the JSON/YAML format.

> This Editor can also be use to convert existing Template from **JSON to YAML** and vice versa.

- Select the `Create Stack` icon > choose `Next`.
- In the Specify Details section, define a `Stack name`, provide an appropriate name.
- In the `Parameters` section:
  - Enter the necessary parameters provided in `Parameters` section in the CF Template
  - There can be some `Default` value set within the template
  - Specify the `EnvironmentName` either as `dev|qa|prod` > choose `Next`
  - On the Options page under `Tags`, specify Key Value for Tags.
  - To create Stack with all default options, Scroll to the bottom and choose Next.
  - On the Review page, review your choices and then choose Create.
  - On the CloudFormation console page, select the specific Stack that just got created.
  - Verify details under **Events , Resources , Output and Template** Tabs to see the activity log from the creation of your CloudFormation stack.

Permissions and service roles

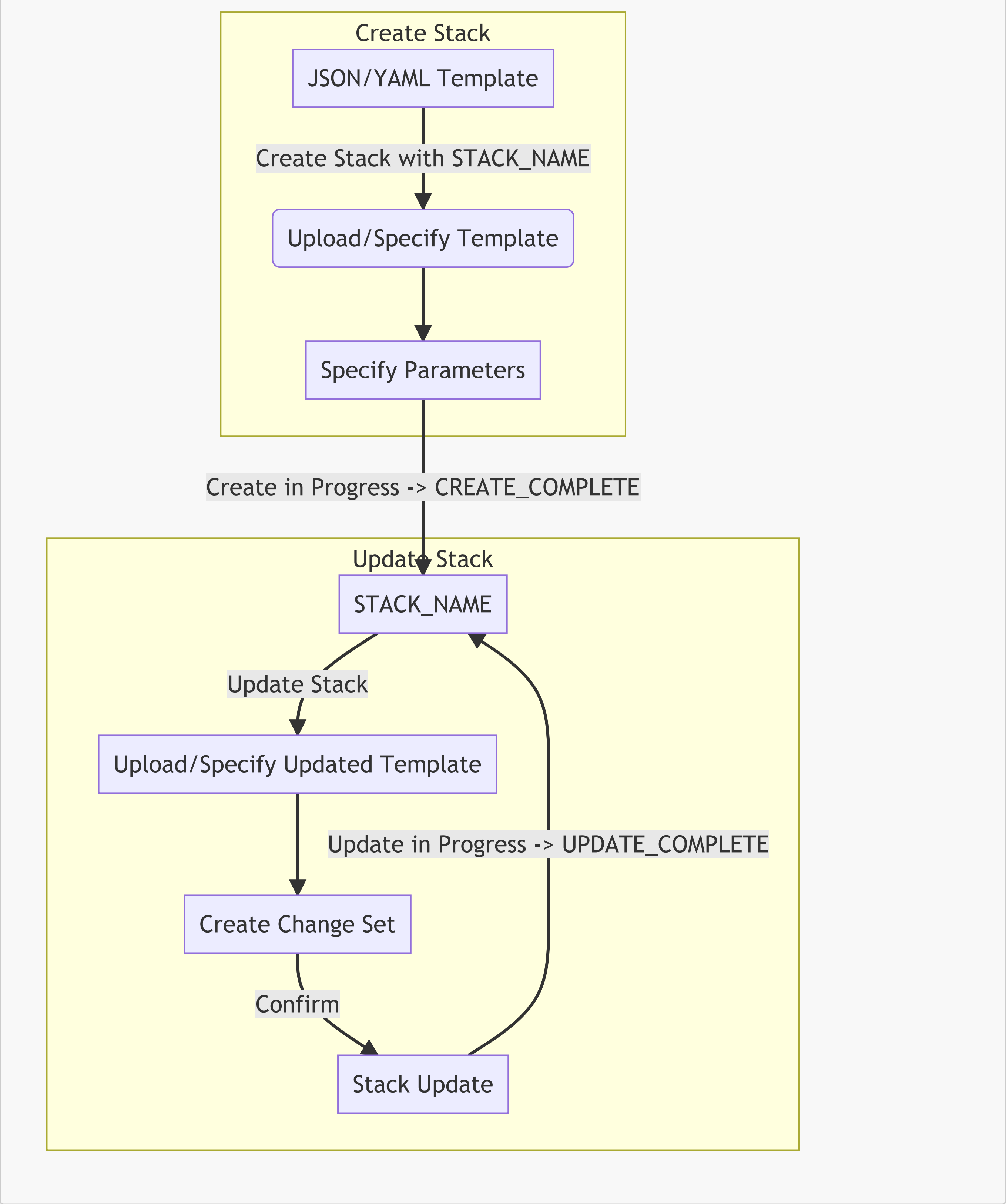
- When we create a Stack using CloudFormation Service, CloudFormation is just making API calls on your behalf.
- This means that CloudFormation will assume the very same permissions or role you use to execute your template.
  - If you don't have permission to create a new Bucket in S3, for example, any template you try to run that creates a S3 Bucket will fail.
- Thus anyone developing CloudFormation typically has a very elevated level of privileges, and these privileges are unnecessarily granted to CloudFormation each time a template is executed.
- If the CF template contains only one resource, which is a like a S3 Bucket,then there should be limited permissions to only S3 bucket instead of full admin privileges to AWS account.
- There should be granular set of permissions given to CloudFormation service to execute the template to limit extra permissions, if a bad template were to be executed. (i.e, a bad copy paste operation resulting in deleted resources).
- Service Roles help to define an IAM role and tell CloudFormation to use this role when your stack is being executed.

Stack Update

- In a scenario where we want to update an infrastructure created from a template?
- The first thing we do is update the template that we made our stack from.
- So, say we have a template with a security group that allows for HTTP traffic and we want to open up SSH traffic as well.
- The first step would be to add that security group rule change to our template file, pick our deployed stack from CloudFormation, and upload the updated template:
- Update the Template -> Upload it to the Same Stack.
- So, it's like the same process as before except we choose to `Update an existing stack` rather than `Create a new stack` like last time.
- `Update the Template` -> `Upload it to Same Stack` -> `Confirm Change Set`

Change Set

- When you update a stack with an updated template, it will generate a change set, and this will show you ALL the things that CloudFormation plans to do.
- Obviously this is incredibly useful for knowing what will be changed before its actually changed.
- Once you confirm the Change Set , cloudformation will go ahead and update the exisitng stack with the updated template.



Updating Resources-Drift

- One of the principles of IaC is that all changes should be represented as code for review and testing. This is especially important where CloudFormation is concerned.
- After creating a stack for you, the CloudFormation service is effectively hands off. If you make a change to any of the resources created by CloudFormation (in the web console,command line, or by some other method), you're effectively causing configuration drift.
- CloudFormation no longer knows the exact state of the resources in your stack.

- The correct approach is to make these changes in your CloudFormation template and perform an update operation on your stack.
- This ensures that CloudFormation always knows the state of your stack and allows you to maintain confidence that your infrastructure code is a complete and accurate representation of your running environments.