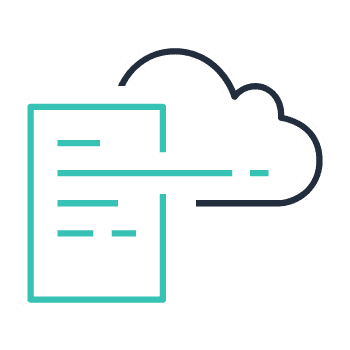
**What does CloudFormation do?**



You can use CloudFormation to treat your infrastructure as code. It gives you a way to model a collection of related AWS and third-party resources, provision them quickly and consistently, and manage them throughout their lifecycles.

You define your AWS resources in a structured text format, either YAML or JSON, called a **CloudFormation template.** Then you can create a **CloudFormation stack** in AWS, which contains the resources created. You can then manage these resources by updating the template.

CloudFormation tracks what changes need to be performed and makes all the changes while keeping your resources in a consistent state. CloudFormation can also create **Change Sets** for approval before making the changes, if you choose.



**By treating infrastructure as code, CloudFormation gives you a way to:**

* **Model a collection of related AWS and third-party resources**
* **Provision them quickly and consistently**
* **Manage them throughout their lifecycles**

**What problem does CloudFormation solve?**

CloudFormation can help you manage your AWS resources, especially resources that depend on each other. You can use CloudFormation to group your resources into **stacks**, using declarative **templates**. CloudFormation can also help you manage creating, updating, and deleting the resources within a stack. You can create resources in parallel, if possible, or create them in specific orders, if they depend on each other.

**What are the benefits of CloudFormation?**

CloudFormation benefits include the following:

**Automate best practices**

+

**Scale your infrastructure worldwide**

+

**Integrate with other AWS services**

+

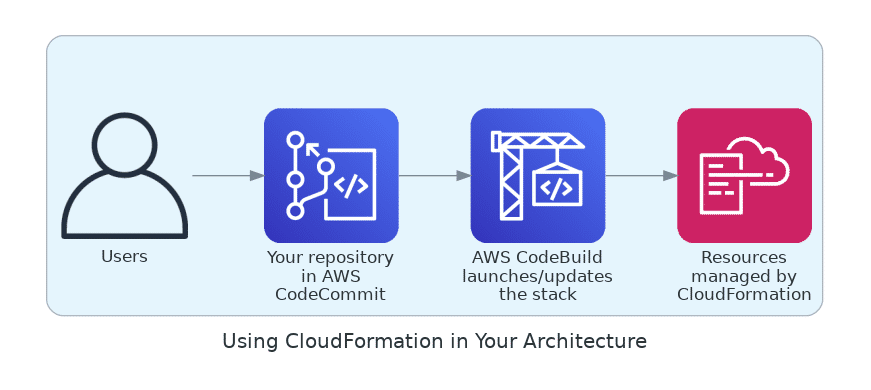
**Manage third-party and private resources**

+

**Extend CloudFormation with the community**

+

**How can I architect a cloud solution using CloudFormation?**



**Using CloudFormation, you can manage all your infrastructure with code.**

Since your infrastructure (your AWS resources) are created with code, you should manage it as code.

Different users can create CloudFormation templates and submit them to your code repository. After code reviews, the code is approved and merged into your main branch. This merging initiates a build process that will create your AWS resources.

**How can I use CloudFormation?**

**Single Devs – Avoid costs**

+

**Enterprise – Infrastructure as code**

+

**Disaster Recovery**

+

**What else should I be aware of when using CloudFormation?**

One important issue is how to create the CloudFormation stacks. You can create them manually on the console, but a better approach is to create an integration pipeline. You can create this so that merging changes to your templates into the main branch creates or modifies the stacks, then use your standard code review practices to manage the templates.

Another issue to keep in mind, when you get to specific resources, is how the lifecycle of each resource is managed. For example, if a change to a resource requires replacing the resource, you should be more careful when using it while it is changing.

Finally, manually updating resources that belong to a CloudFormation stack is strongly discouraged. Do not make changes to stack resources outside of CloudFormation.

**How much does CloudFormation cost ?**

The cost structure of CloudFormation is simple. CloudFormation is free for managing AWS resources. You are only charged for the resources you create and the API calls that CloudFormation performs on your behalf.

If you manage third-party resources using CloudFormation, there is a small charge per operation. For full pricing details, see [AWS CloudFormation Pricing documentation](https://aws.amazon.com/cloudformation/pricing/).

**[Lesson 2 -](https://explore.skillbuilder.aws/files/a/w/aws_prod1_docebosaas_com/1669759200/7DoXcbh2YWY8FFlHcyuBUg/tincan/0d5f303807209ed734ff1f5c24534bb642a72e18/index.html?enhanced_signature=L-fj8RbDB3Cd_AqmChFPWTOaG-jwtvoMnFr-pfu9Cf0&endpoint=https%3A%2F%2Fexplore.skillbuilder.aws%2Flrs-api%2F&auth=Basic%20ZGMxYmE1YWUtNDhiMy00ZWVkLTgzMmQtMTJkMTYzYTM2MWU0OmVjY2FhZGQwMTQ0OGE2N2M5YmM4YTRmNDJlZWZjNThl&actor=%7B%22name%22%3A%22bidemi+bakare%22%2C%22mbox%22%3A%22mailto%3Abidemi460%40gmail.com%22%7D&registration=3336177b-0bd3-4948-b9ab-a72e982581c3&activity_id=http%3A%2F%2FLuMdxFCDWHtT30MMtNnaRaldBfCBoRKd_rise&Accept-Language=en&course_id=3627&content_token=3336177b-0bd3-4948-b9ab-a72e982581c3&session_context=lms&host=kfase3bzbc.execute-api.us-east-1.amazonaws.com&path=/v1/xApi/&rs=638662f4ed604&crct=53effb7a204d0deb39764abd2e58dcd56385401775aebcd64f3c952a01b056ae2442d28c54a88ca40b012ec5f907132b858b1b5ab3bd406d23e7de542a3f373a&course_code=TCAA-DIG-100-MGCFGS-0100-EN-US&course_id=3627&username=9912fc45-754b-49c0-b815-7f038e3e8e75&user_id=2687294&hash=b5d11c5fe38be8bb515b2825d4a777440c0a3571f50ea8bfa9bd0e9b4da71401" \l "/lessons/3CS64qH0MKTn4g5Jm1IBEfdLBREvocq4)**

**[Using CloudFormation](https://explore.skillbuilder.aws/files/a/w/aws_prod1_docebosaas_com/1669759200/7DoXcbh2YWY8FFlHcyuBUg/tincan/0d5f303807209ed734ff1f5c24534bb642a72e18/index.html?enhanced_signature=L-fj8RbDB3Cd_AqmChFPWTOaG-jwtvoMnFr-pfu9Cf0&endpoint=https%3A%2F%2Fexplore.skillbuilder.aws%2Flrs-api%2F&auth=Basic%20ZGMxYmE1YWUtNDhiMy00ZWVkLTgzMmQtMTJkMTYzYTM2MWU0OmVjY2FhZGQwMTQ0OGE2N2M5YmM4YTRmNDJlZWZjNThl&actor=%7B%22name%22%3A%22bidemi+bakare%22%2C%22mbox%22%3A%22mailto%3Abidemi460%40gmail.com%22%7D&registration=3336177b-0bd3-4948-b9ab-a72e982581c3&activity_id=http%3A%2F%2FLuMdxFCDWHtT30MMtNnaRaldBfCBoRKd_rise&Accept-Language=en&course_id=3627&content_token=3336177b-0bd3-4948-b9ab-a72e982581c3&session_context=lms&host=kfase3bzbc.execute-api.us-east-1.amazonaws.com&path=/v1/xApi/&rs=638662f4ed604&crct=53effb7a204d0deb39764abd2e58dcd56385401775aebcd64f3c952a01b056ae2442d28c54a88ca40b012ec5f907132b858b1b5ab3bd406d23e7de542a3f373a&course_code=TCAA-DIG-100-MGCFGS-0100-EN-US&course_id=3627&username=9912fc45-754b-49c0-b815-7f038e3e8e75&user_id=2687294&hash=b5d11c5fe38be8bb515b2825d4a777440c0a3571f50ea8bfa9bd0e9b4da71401" \l "/lessons/3CS64qH0MKTn4g5Jm1IBEfdLBREvocq4)**

**What are the benefits of CloudFormation?**

CloudFormation benefits include the following:

**Automate best practices**

–

With CloudFormation, you can apply DevOps and GitOps best practices using widely adopted processes such as starting with a Git repository and deploying through a continuous integration and continuous delivery (CI/CD) pipeline. You can also simplify auditing changes and trigger automated deployments with pipeline integrations such as GitHub Actions and AWS CodePipeline

**Scale your infrastructure worldwide**

–

Manage resource scaling by sharing CloudFormation templates for use across your organization to meet safety, compliance, and configuration standards across all AWS accounts and Regions. Templates and parameters help simplify scaling so you can share best practices and company policies. Additionally, you can use CloudFormation StackSets to create, update, or delete stacks across multiple AWS accounts and Regions with a single operation.

**Integrate with other AWS services**

–

To further automate resource management across your organization, you can integrate CloudFormation with other AWS services, including AWS Identity and Access Management (IAM) for access control, AWS Config for compliance, and AWS Service Catalog for turnkey application distribution and additional governance controls. Integrations with AWS CodePipeline and other builder tools give you the ability to implement the latest DevOps best practices and improve automation, testing, and controls.

**Integrate with other AWS services**

–

To further automate resource management across your organization, you can integrate CloudFormation with other AWS services, including AWS Identity and Access Management (IAM) for access control, AWS Config for compliance, and AWS Service Catalog for turnkey application distribution and additional governance controls. Integrations with AWS CodePipeline and other builder tools give you the ability to implement the latest DevOps best practices and improve automation, testing, and controls.

**Manage third-party and private resources**

–

Model, provision, and manage third-party application resources (such as monitoring, team productivity, incident management, CI/CD, and version control applications) alongside your AWS resources. Use the open source CloudFormation CLI to build your own CloudFormation resource providers (native AWS types published as open source).

**Extend CloudFormation with the community**

–

The CloudFormation GitHub organization offers open source projects that extend CloudFormation capabilities. You can use the CloudFormation registry and CloudFormation CLI to define and create resource providers to automate the creation of resources safely and systematically. Using CloudFormation GitHub projects, you can do things like check CloudFormation templates for policy compliance (using cfn-guard), or validate use of best practices (using cfn-lint).

HOW TO USE CLOUDFORMATION

**Single Devs – Avoid costs**

–

CloudFormation can help you quickly create and destroy a group of related resources, which is useful when you are learning about a new AWS service. You can quickly spin down a stack with all its resources when you are not using it and re-create it later. As you transition into using the services in production, you can start from those templates and scale up as needed.

**Enterprise – Infrastructure as code**

–

Many companies use CloudFormation to manage all of their AWS resources, with CI/CD pipelines creating the stacks from code. Some companies even manage resources outside AWS using

**Disaster Recovery**

–

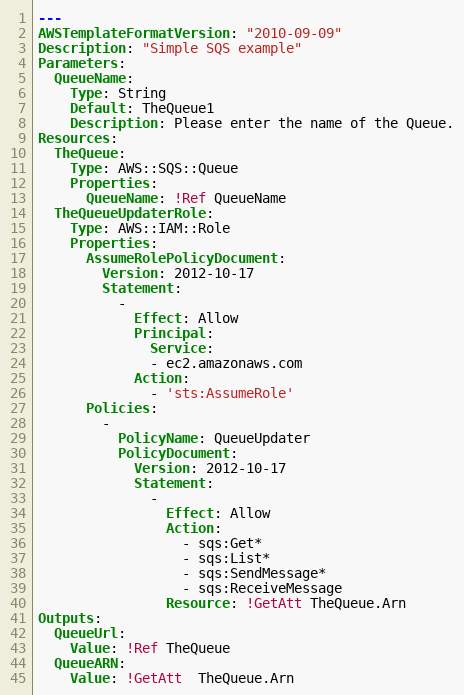
If you create your infrastructure with CloudFormation, you can quickly re-create it in a different Region or account, enabling disaster recovery and business continuity.

**Using CloudFormation**

*Lesson 2 of 3*

**What are the basic technical concepts of CloudFormation?**

* **Resources –**Any of the things you can create within AWS, which includes things like Amazon Simple Storage Service (Amazon S3) buckets, Amazon Elastic Compute Cloud (Amazon EC2) instances, or Amazon Simple Queue Service (Amazon SQS) queues.
* **Templates**– Text-based (JSON or YAML) descriptions of CloudFormation stacks that you can use to define all of your resources, including which resources depend on each other.
* **Stack** – A collection of AWS resources that you can manage as a single unit.
* **StackSet** – A named set of stacks that use the same template, but applied across different accounts and Regions. You can create, update, or delete stacks across multiple accounts and Regions with a single operation.



**Sample template**

The accompanying CloudFormation template creates an Amazon SQS queue and illustrates parameters and outputs.

It creates two resources: an SQS queue, starting in line 10, and an IAM role, starting in line 14.

It defines one parameter in line 5, which would need to be provided when creating a stack with this template (and which you could use to create different stacks with the same template, by providing different parameters).

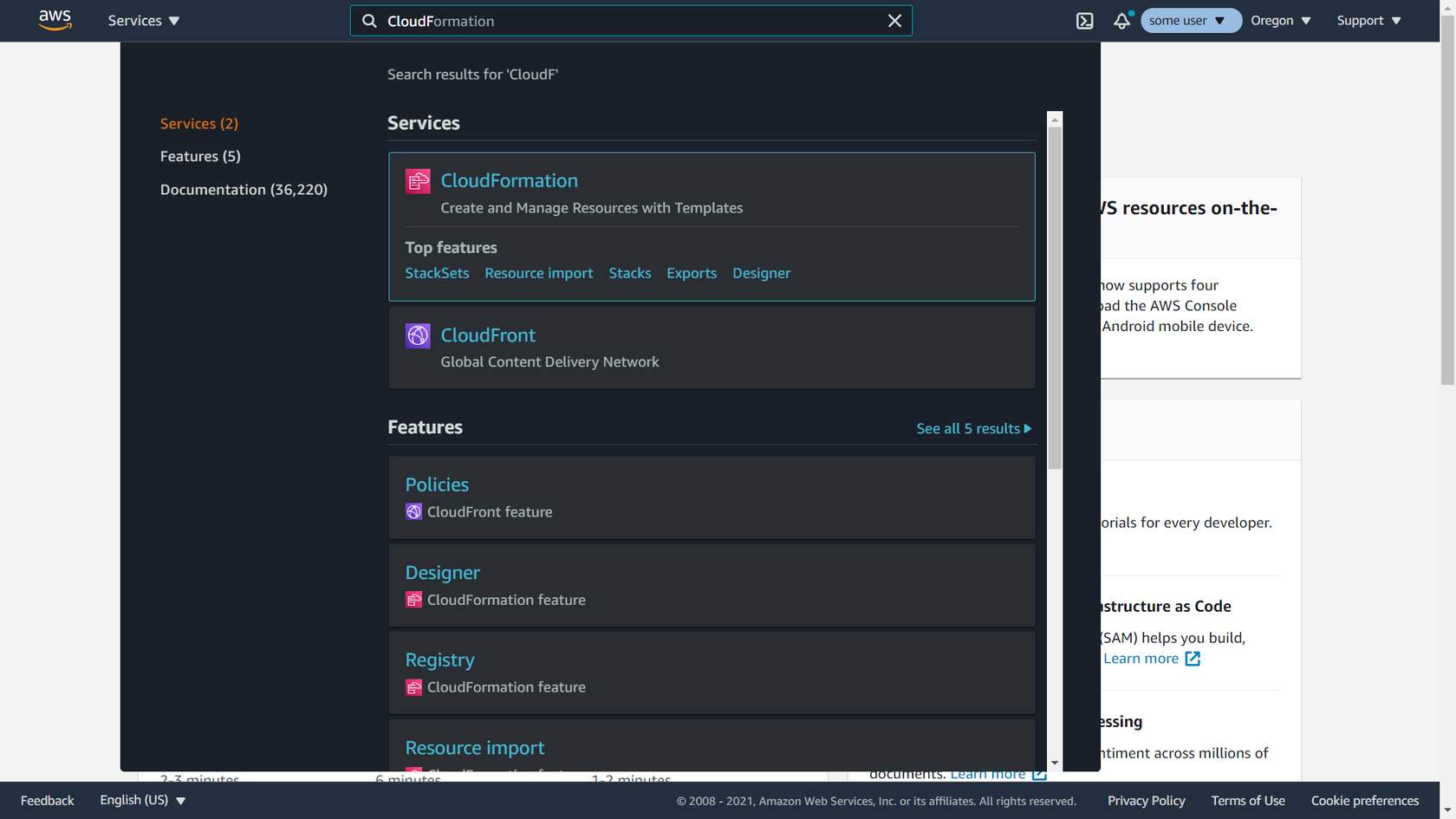
It defines two *outputs*, values that get attached to the stack, and can be viewed in the console or accessed with a program.

**How can I create CloudFormation stacks using the AWS Management Console?**

The following screenshots illustrate how you can create a CloudFormation stack using the console. If you want to follow along, you can download the YAML template provided.

**[BudgetWithParams.yaml](https://explore.skillbuilder.aws/files/a/w/aws_prod1_docebosaas_com/1669759200/7DoXcbh2YWY8FFlHcyuBUg/tincan/0d5f303807209ed734ff1f5c24534bb642a72e18/assets/6d_x6X3HBIlE8yCu_Swd6t2MoFNtQuEtC-BudgetWithParams.yaml" \t "_blank)**

[989 B](https://explore.skillbuilder.aws/files/a/w/aws_prod1_docebosaas_com/1669759200/7DoXcbh2YWY8FFlHcyuBUg/tincan/0d5f303807209ed734ff1f5c24534bb642a72e18/assets/6d_x6X3HBIlE8yCu_Swd6t2MoFNtQuEtC-BudgetWithParams.yaml" \t "_blank)

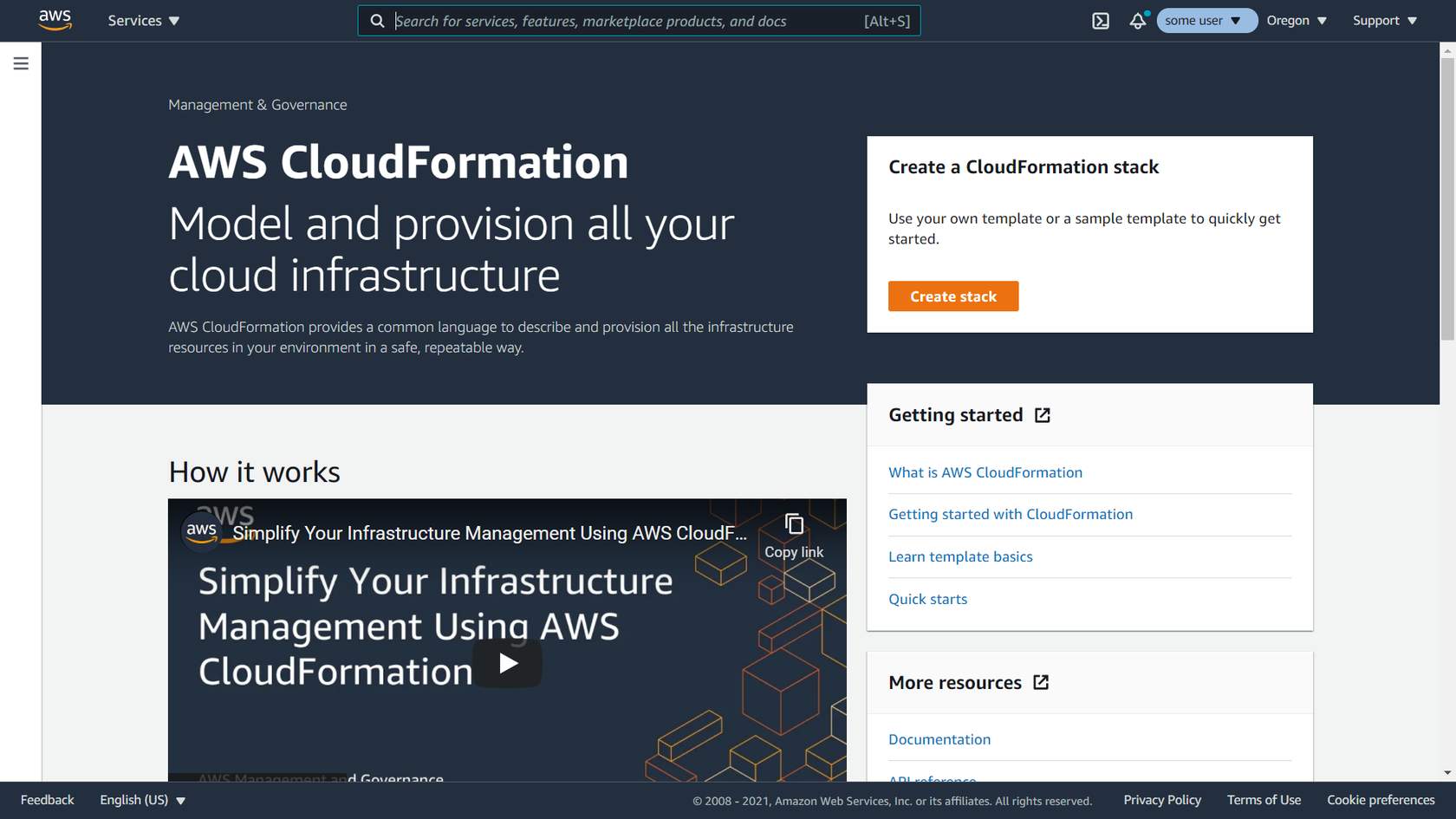


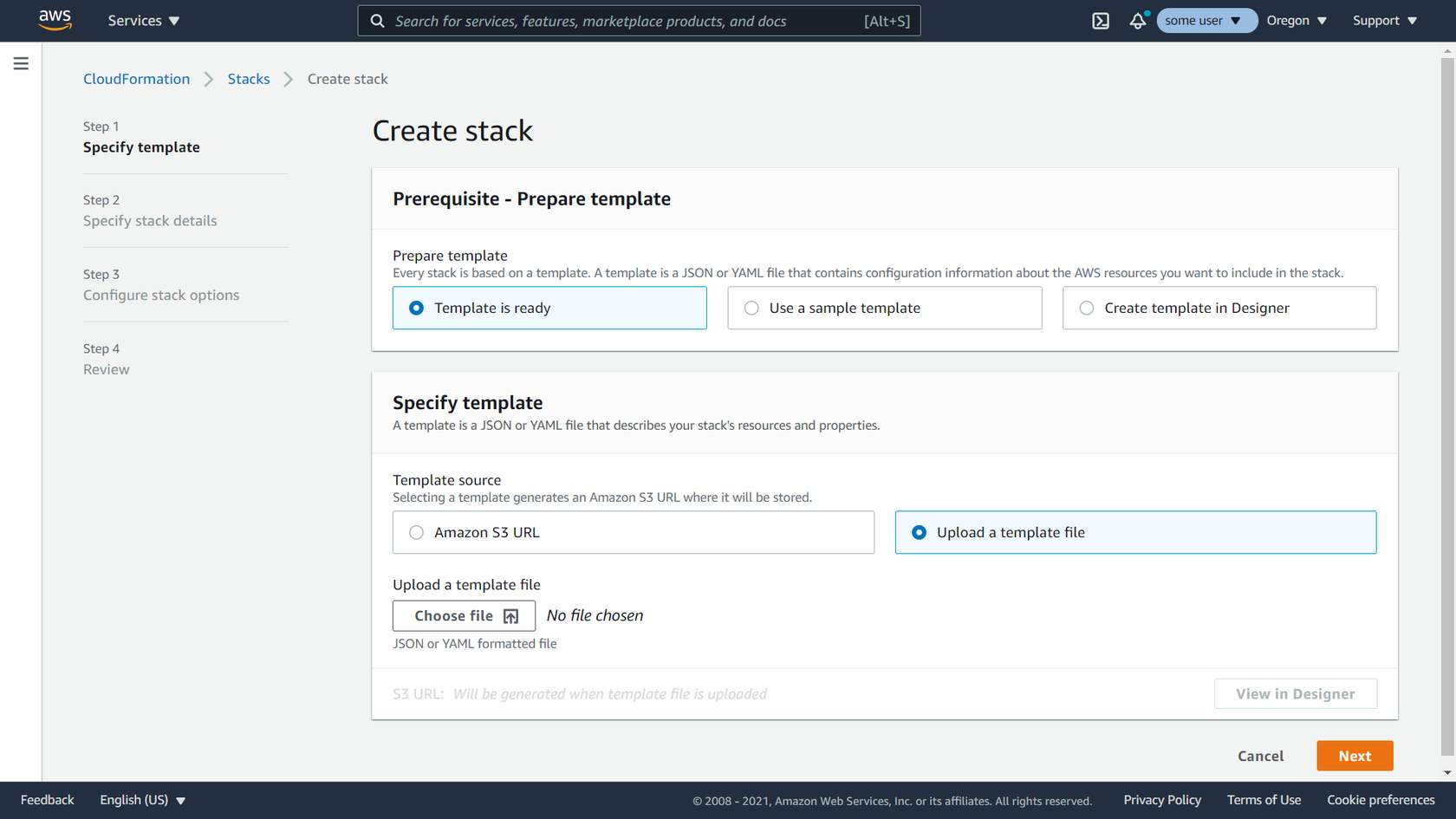
**Open the console and find CloudFormation**

Open the [console](https://console.aws.amazon.com/), search for CloudFormation on the services search bar, then select CloudFormation.

**Choose Create stack**

On the CloudFormation section, choose Create stack.





**Specify the template**

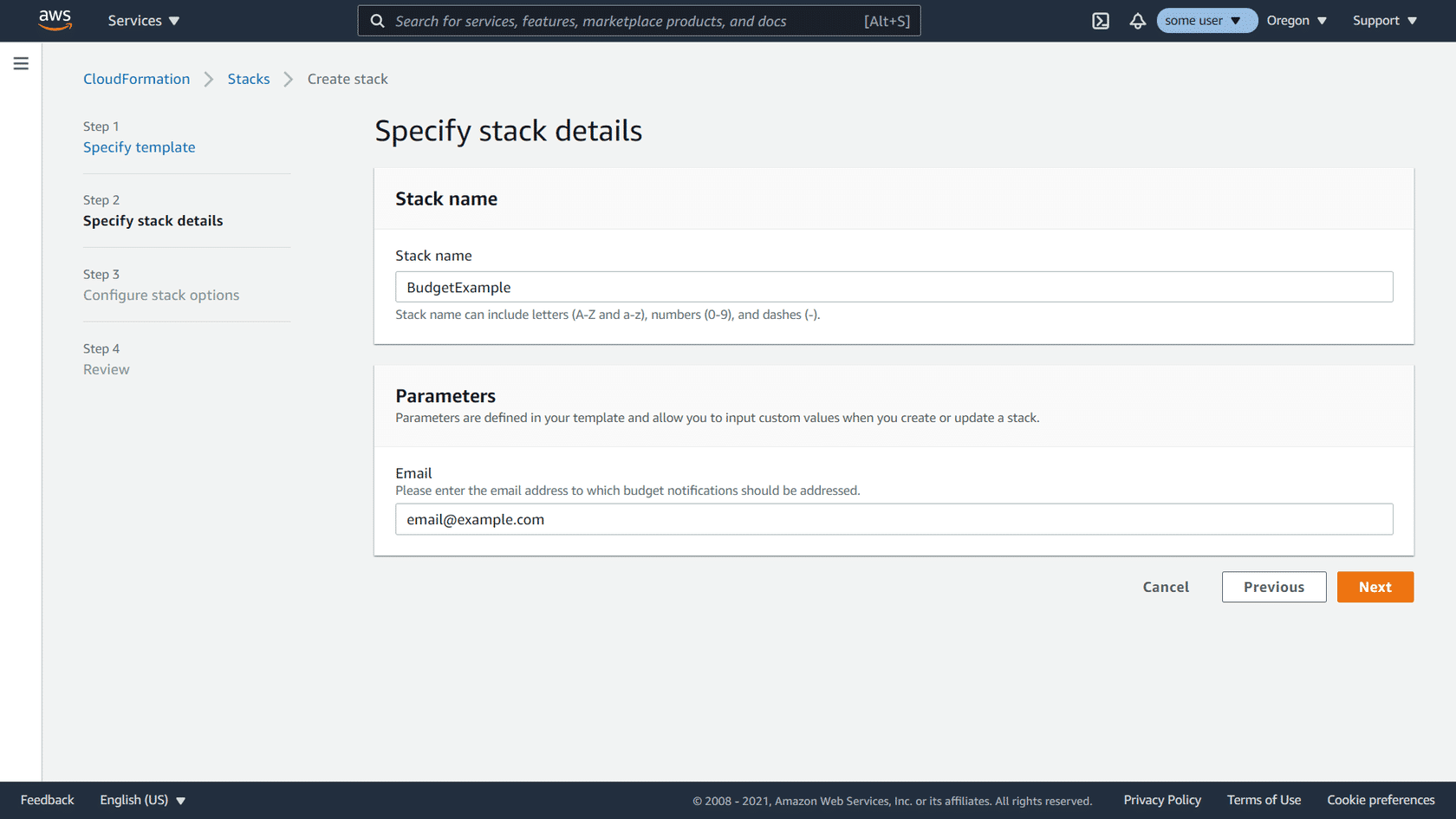
You can provide your own template, use a sample template, or use the template designer, which is a graphical interface for creating templates.

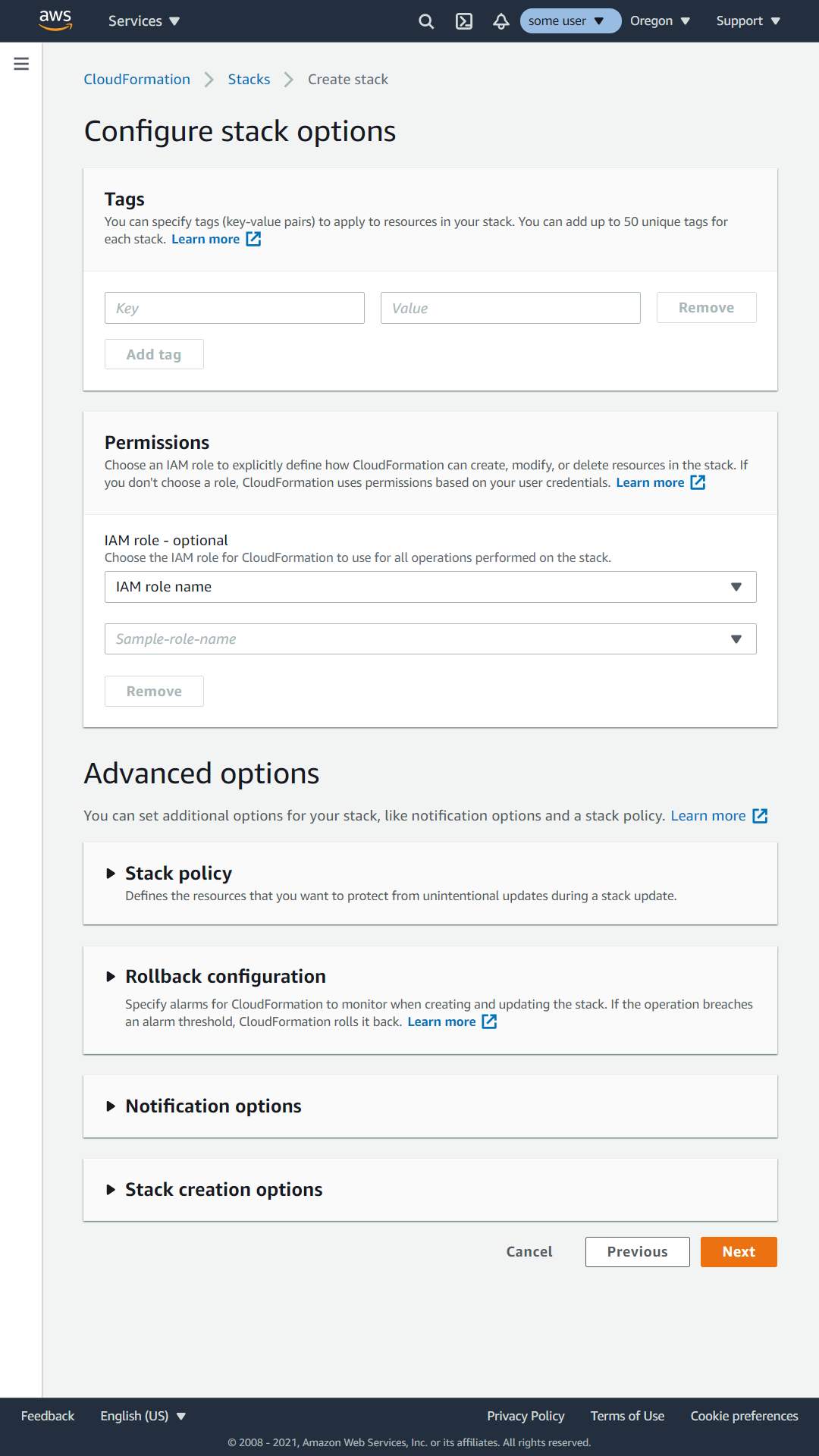
If you are providing the template, you can specify an Amazon S3 location or upload it from your computer.

**Specify details – name and parameters**

You need to provide a unique name for your stack (stacks in different accounts or different Regions can have the same name, but not in the same account and Region).

If your template has any parameters, you can specify them here.





**Configure more options**

On the next screen, you can configure more options.

You can add tags to your stack to make it easier to group and manage them.

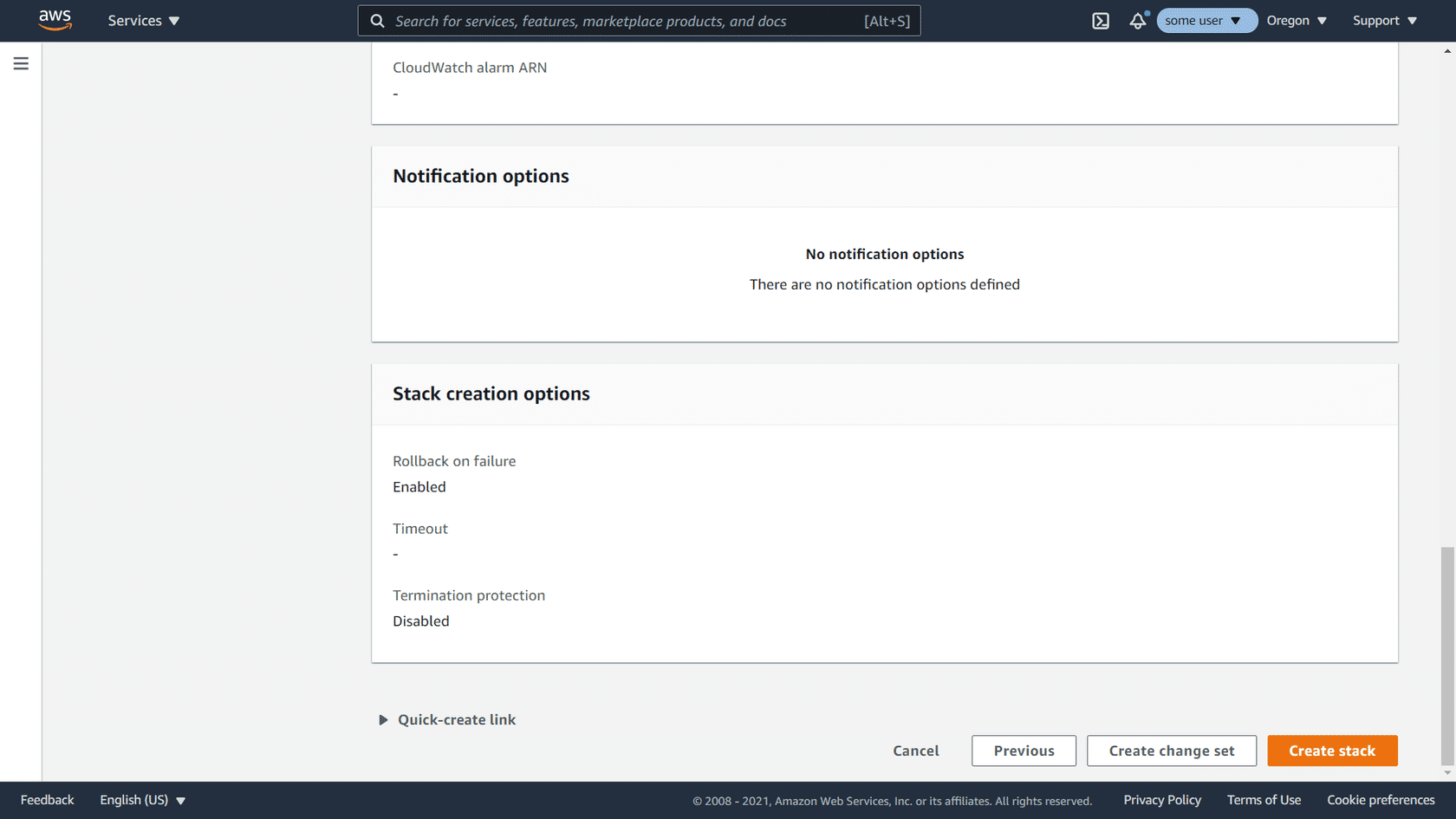
You can specify whether you want a special IAM role for CloudFormation to use to operate on the stack.

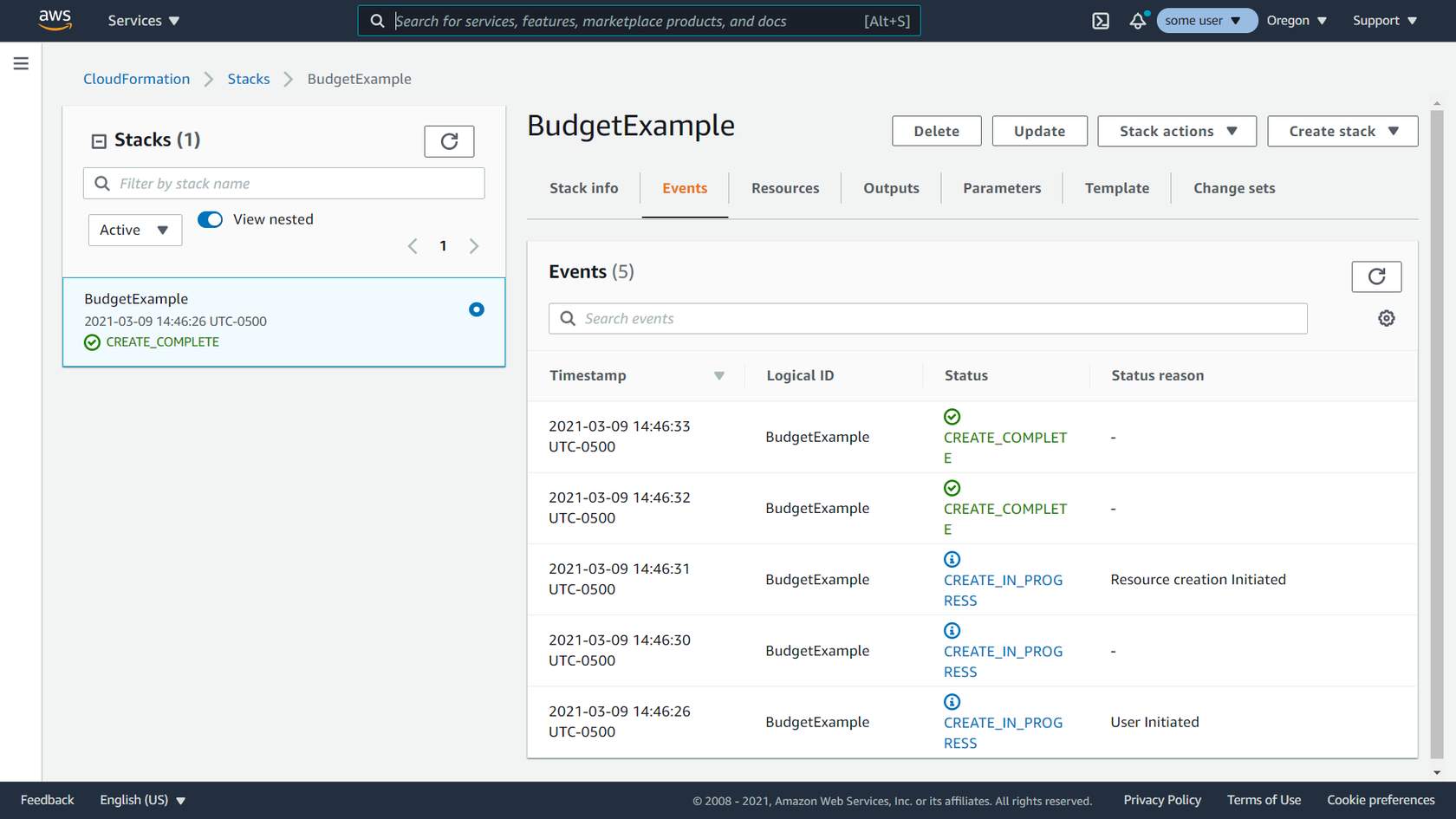
You can specify other advanced options, like enhanced protection with a stack policy, rollback and notification options, and other creation options.

**Review and confirm**

The console will display all options that will be used for your stack, so you can review and verify they are all as intended.

Choosing Create stack will initiate the stack creation.





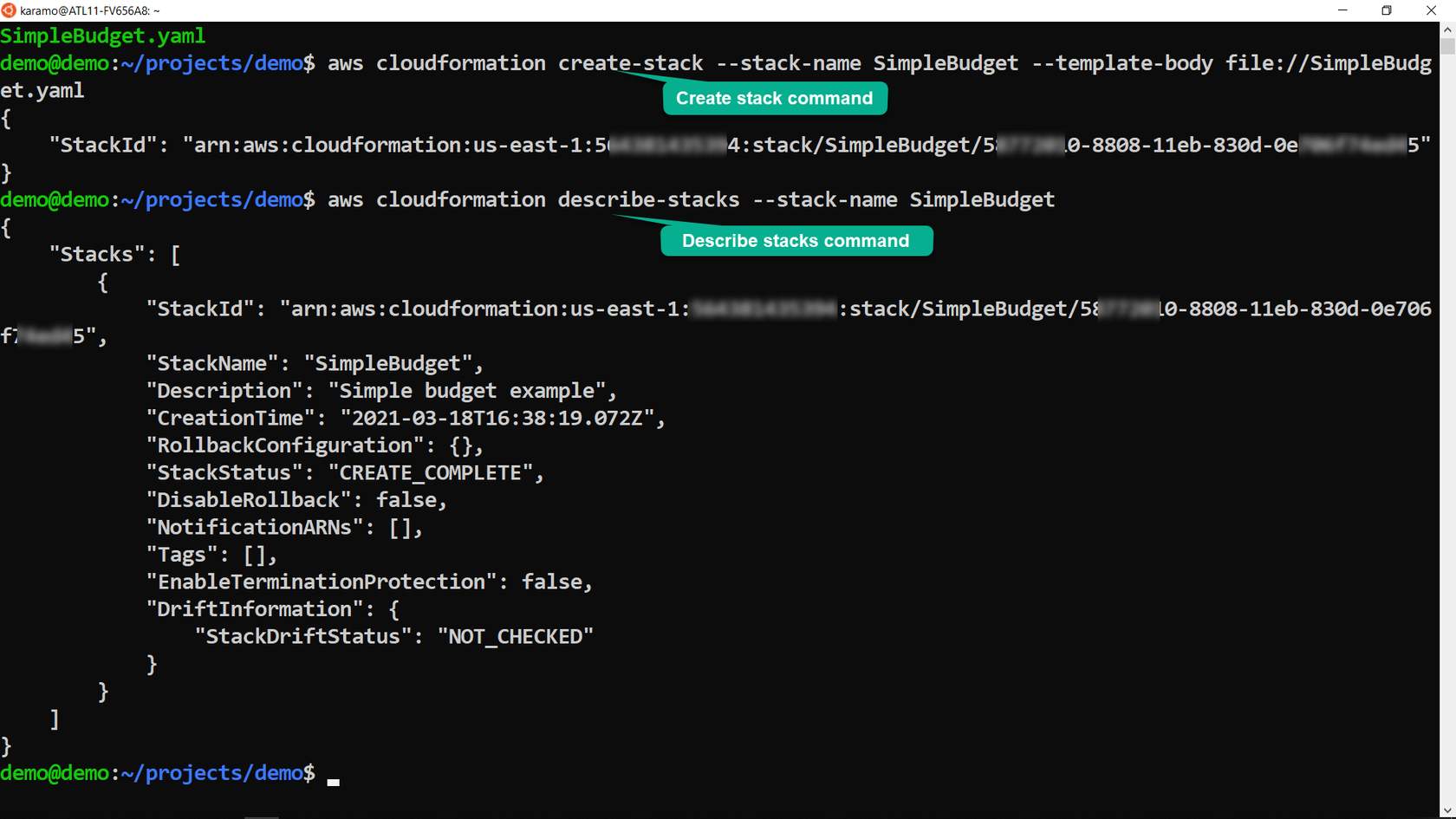
**Stack list and events**

While the stack is creating, you can see a list of events updating as new resources are created. The other tabs have different information about your stack.

Eventually, your stack and all its resources should be in CREATE\_COMPLETE state.

**How can I create CloudFormation stacks using the AWS Command Line Interface?**

If you have the [AWS Command Line Interface (AWS CLI)](https://aws.amazon.com/cli/) installed and configured, you can create a stack using the CloudFormation [create-stack](https://awscli.amazonaws.com/v2/documentation/api/latest/reference/cloudformation/create-stack.html) command, You need to pass it the stack name and the template, which can be a local file or on S3. You can use the CloudFormation [describe-stacks](https://awscli.amazonaws.com/v2/documentation/api/latest/reference/cloudformation/describe-stacks.html) command to obtain basic information about your stack.



This screenshot shows the use of that command and the describe-stacks command used to obtain information about the stack, which you can use to monitor its progress.

SAMPLE BUSINESS PROBLEMS

An education company delivers course materials to their students as PDF downloads. Recently, there have been more downloads than students in their classes, and the company wants to limit each PDF to being downloaded to class members.

The current architecture has students log in to a website that runs on EC2 instances. The website provides links to publicly accessible files in Amazon S3. The company wants to use CloudFront to limit the downloads.