



PROVISION THE CONTROL

What are we going to do?

We are going to perform the following steps:

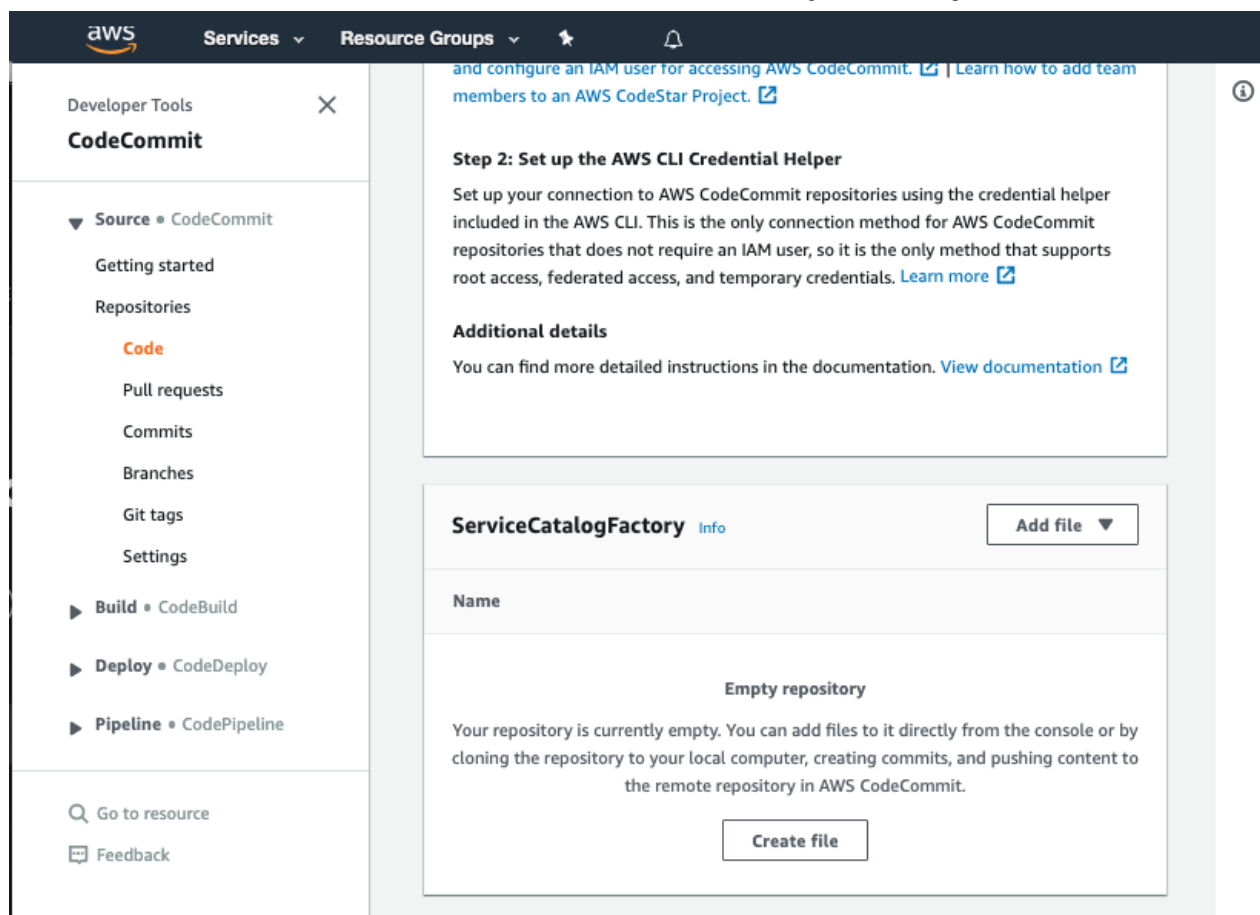
- create a manifest file with our account in it
- provision the product *aws-config-desired-instance-types* into our account

Step by step guide

Here are the steps you need to follow to provision the control. In the previous task, we created an AWS Service Catalog product but it has not yet been provisioned.

Create a manifest file with our account in it

- Navigate to the [ServiceCatalogPuppet CodeCommit repository](#)
- Scroll down to the bottom of the page and hit the *Create file* button

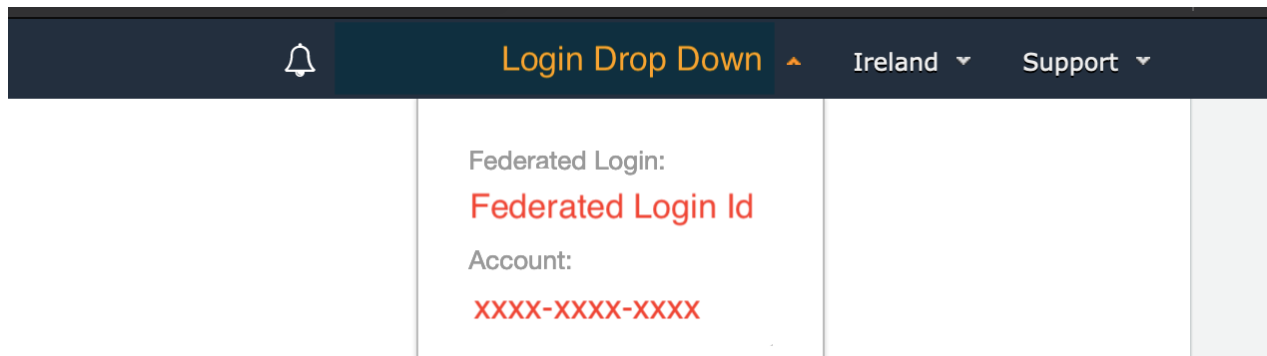


- Copy the following snippet into the main input field and replace account_id to show your account id on the highlighted line:

```
accounts:
- account_id: "<YOUR_ACCOUNT_ID_WITHOUT_HYPHENS>"
  name: "puppet-account"
  default_region: "eu-west-1"
  regions_enabled:
    - "eu-west-1"
    - "eu-west-2"
  tags:
    - "type:prod"
    - "partition:eu"
```

Note

You can find your Account Number in the **top right** drop down of the AWS Console page. It is a 12 digit number. Please **do not** include the hyphens ('-') in the manifest code.



Provision the product *aws-config-desired-instance-types* into a spoke account

- Append the following snippet to the end of the main input field:

```
launches:
  aws-config-desired-instance-types:
    portfolio: "reinvent-cloud-engineering-governance"
    product: "aws-config-desired-instance-types"
    version: "v1"
    parameters:
      InstanceType:
        default: "t2.medium, t2.large, t2.xlarge"
    deploy_to:
      tags:
        - tag: "type:prod"
          regions: "default_region"
```

Note

The CloudFormation template we used to create this product had a parameter named *InstanceType*. The highlighted lines show how we can use the framework to set a value for that parameter when provisioning it.

- The main input field should look like this (remember to set your `account_id`):

```
accounts:
  - account_id: "<YOUR_ACCOUNT_ID_WITHOUT_HYPHENS>"
    name: "puppet-account"
    default_region: "eu-west-1"
    regions_enabled:
```

```
- "eu-west-1"
- "eu-west-2"
tags:
  - "type:prod"
  - "partition:eu"
launches:
  aws-config-desired-instance-types:
    portfolio: "reinvent-cloud-engineering-governance"
    product: "aws-config-desired-instance-types"
    version: "v1"
    parameters:
      InstanceType:
        default: "t2.medium, t2.large, t2.xlarge"
    deploy_to:
      tags:
        - tag: "type:prod"
          regions: "default_region"
```

Committing the manifest file

Now that we have written the manifest file we are ready to commit it.

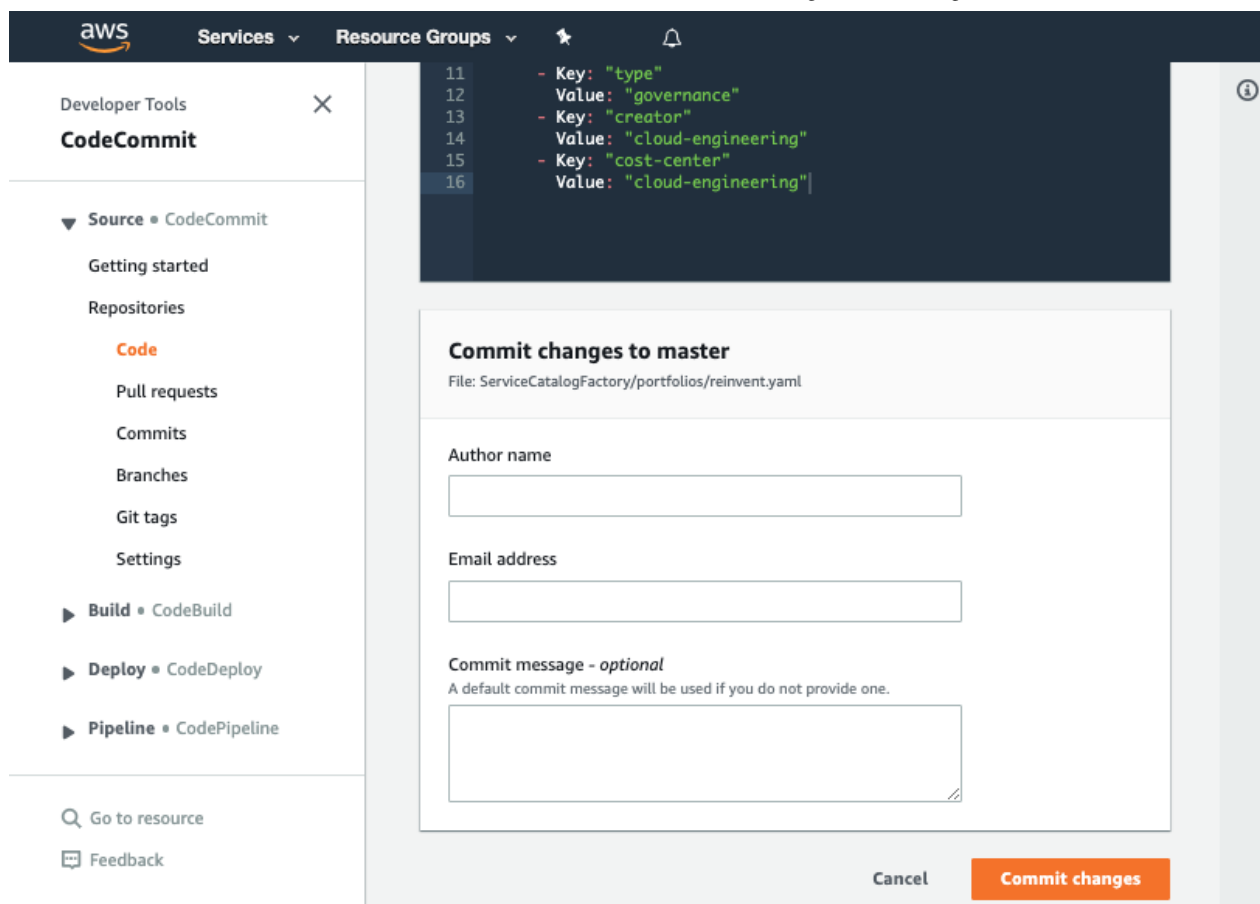
- Set the *File name* to `manifest.yaml`
- Set your *Author name*
- Set your *Email address*
- Set your *Commit message*



Tip

Using a good / unique commit message will help you understand what is going on later.

- Click the *Commit changes* button:



What did we just do?

The YAML file we created in the previous step told the framework to perform the following actions:

- provision a product named *aws-config-desired-instance-types* into each of the enabled regions of the account

When you added the following:

```
launches:
  aws-config-desired-instance-types:
    portfolio: "reinvent-cloud-engineering-governance"
    product: "aws-config-desired-instance-types"
    version: "v1"
    parameters:
      InstanceType:
        default: "t2.medium, t2.large, t2.xlarge"
    deploy_to:
      tags:
        - tag: "type:prod"
          regions: "default_region"
```

You told the framework to provision *v1* of *aws-config-desired-instance-types* from the portfolio *cloud-engineering-governance* into every account that has the tag *type:prod*

```
accounts:
- account_id: "<YOUR_ACCOUNT_ID_WITHOUT_HYPHENS>"
  name: "puppet-account"
  default_region: "eu-west-1"
  regions_enabled:
  - "eu-west-1"
  - "eu-west-2"
  tags:
  - "type:prod"
  - "partition:eu"
```

Within each account there will be a copy of the product provisioned into each of the regions listed in the `regions_enabled` section:

```
accounts:
- account_id: "<YOUR_ACCOUNT_ID_WITHOUT_HYPHENS>"
  name: "puppet-account"
  default_region: "eu-west-1"
  regions_enabled:
  - "eu-west-1"
  - "eu-west-2"
  tags:
  - "type:prod"
  - "partition:eu"
```

For this workshop, we are creating and provisioning the product into the same AWS Account, but in a multi-account setup, you might choose to create a product in a “hub” account and provision it only to “spoke” accounts.

Note

In the workshop, you will only have permission to view the products in eu-west-1.

Verifying the provisioned product

Once you have made your changes the [ServiceCatalogPuppet Pipeline](#) should have run. If you were quick in making the change, the pipeline may still be running. If it has not yet started feel free to hit the *Release change* button.

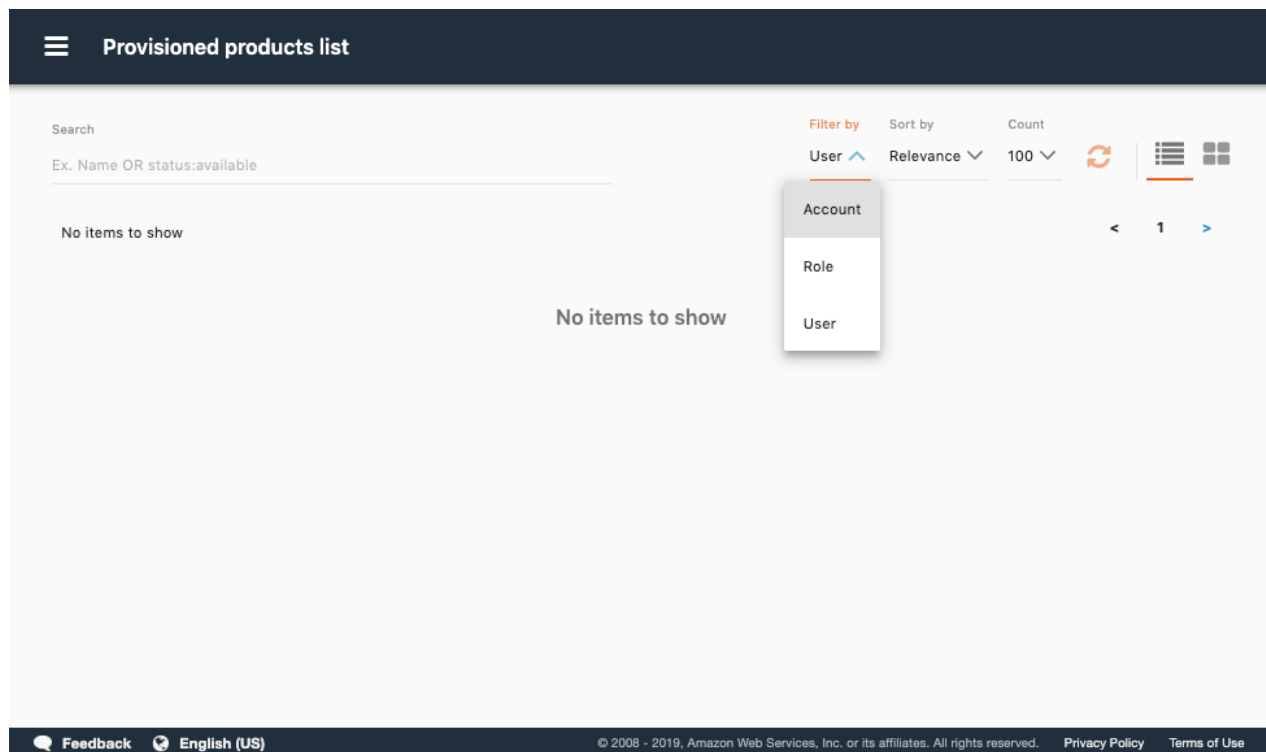
Once it has completed it should show the *Source*, *Generate* and *Deploy* stages in green to indicate they have completed successfully:

The screenshot displays the AWS Service Catalog console interface. The top navigation bar includes the AWS logo, 'Services', 'Resource Groups', and user information. The left sidebar shows the 'CodePipeline' section with a list of stages: Source (CodeCommit), Build (CodeBuild), and Deploy (CodeDeploy). The main content area shows the pipeline details for 'CodeCommit'. The pipeline is marked as 'Succeeded - 1 day ago'. The stages are: Source (CodeCommit), Generate (GenerateShares), and Deploy (Deploy). Each stage is marked as 'Succeeded - 1 day ago'. The right sidebar shows three green checkmarks indicating successful completion.

Note

If this is failing please raise your hand for some assistance

Once you have verified the pipeline has run you can go to [Service Catalog provisioned products](#) to view your provisioned product. Please note when you arrive at the provisioned product page you will need to select account from the filter by drop down in the top right:



Note

If you cannot see your product please raise your hand for some assistance

You have now successfully provisioned a product

Verify that the AWS Config rule is enabled

To see the AWS Config rule enabled, navigate to [AWS Config rules](#). Once there you should see the following:

aws

Services ▾

Resource Groups ▾

★

🔔

Support ▾

AWS Config

Dashboard

Rules

Resources

Advanced query

Settings

Authorizations

Aggregated view

Rules

Resources

Aggregators

What's new

Learn More

Documentation ↗

Partners ↗

FAQs ↗

Pricing ↗

📘

The redesigned AWS Config console is now available for use.

We've completely redesigned the console to improve the overall look and feel.

[Try it out now.](#)

Rules

Rules represent your desired configuration settings. AWS Config evaluates whether your resource configurations comply with re
results in the following table.

+

Add rule

Manage remediation

View details

Edit

Compliance status ▾

Filter

| | Rule name | Compliance | Remediation a |
|-----------------------|-------------------------------------|------------|---------------|
| <input type="radio"/> | Your config rule should appear here | | |

https://service-catalog-tools-workshop.com/40-reinvent2019/100-task-1/200-provision-the-control.html

9/9