

# *Gerenciando configurações com Parameter Store e efetuando o deploy com CloudFormation*

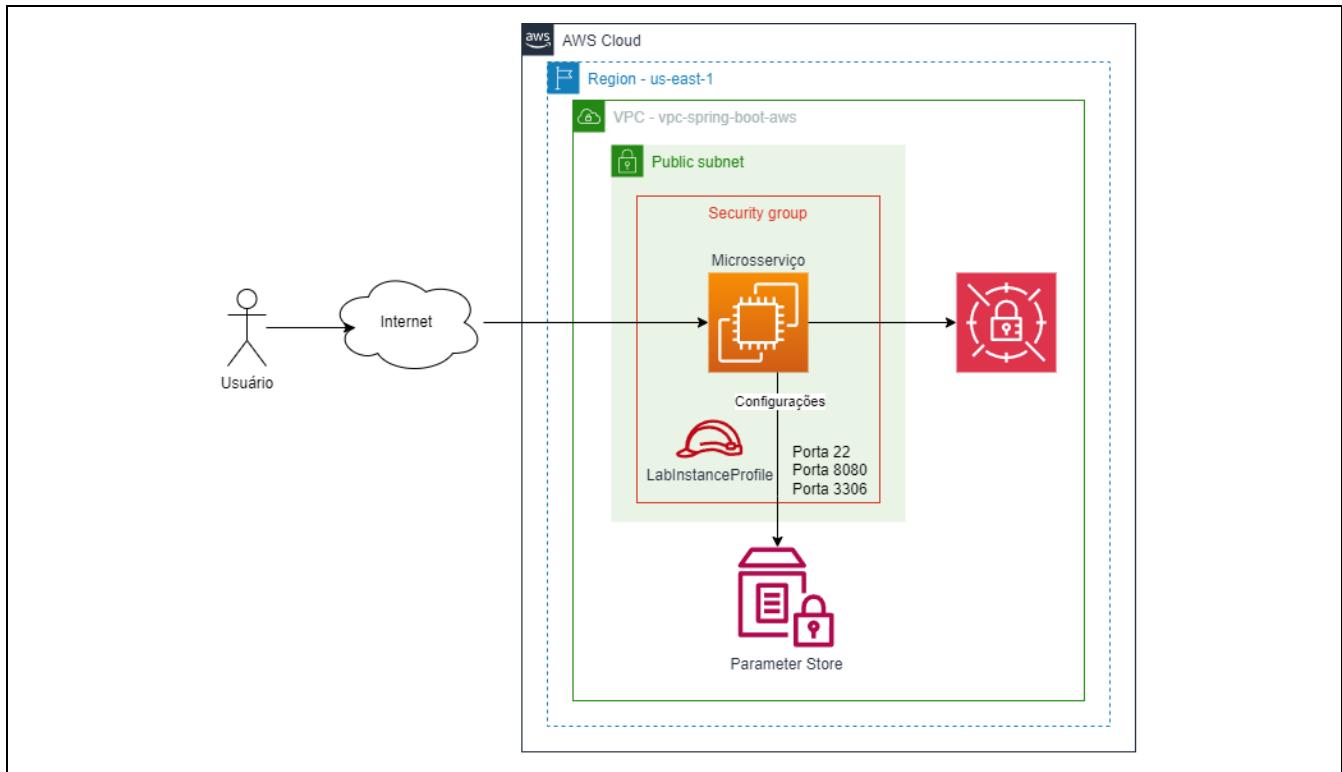


Prof. Thomás da Costa - <https://thomasdacosta.com.br>

## Objetivo:

Implantar os parâmetros de configuração do microsserviço em Spring Boot usando um script do CloudFormation e o Parameter Store.

## Desenho da Solução:

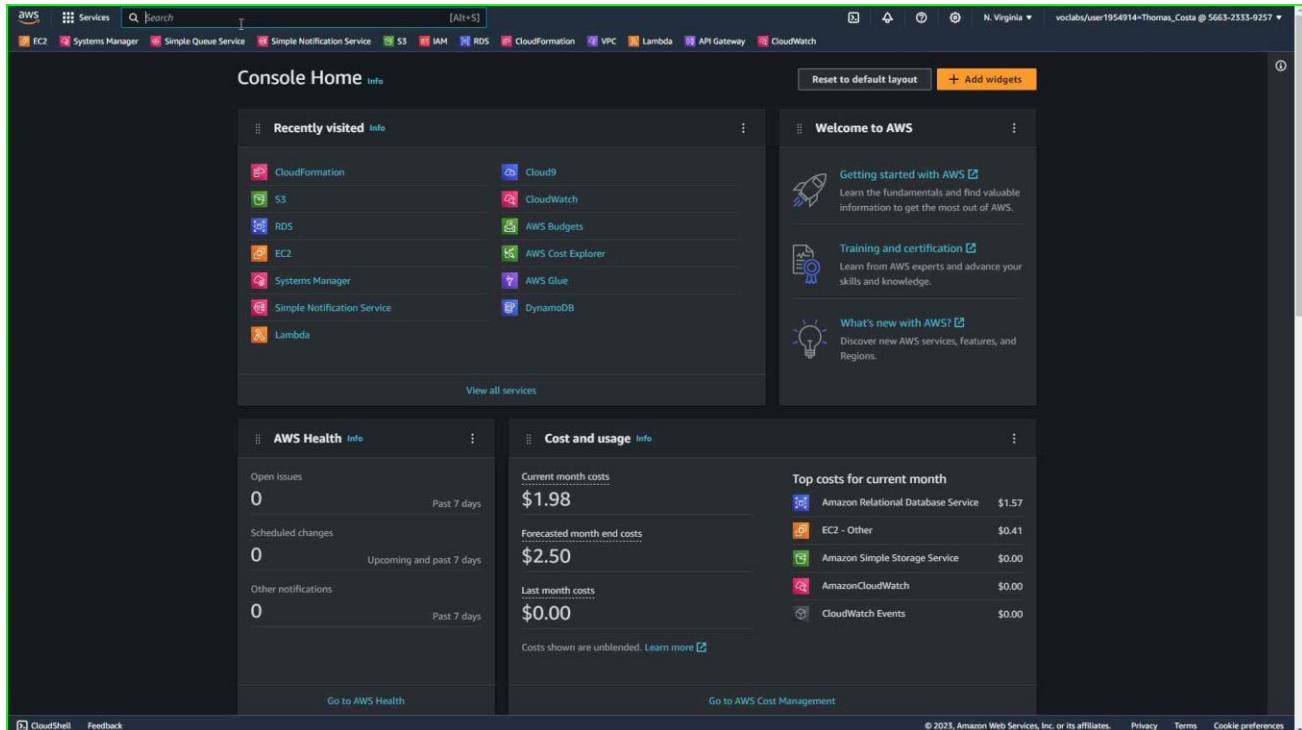


## Premissas:

- Nenhuma premissa para esse tutorial.

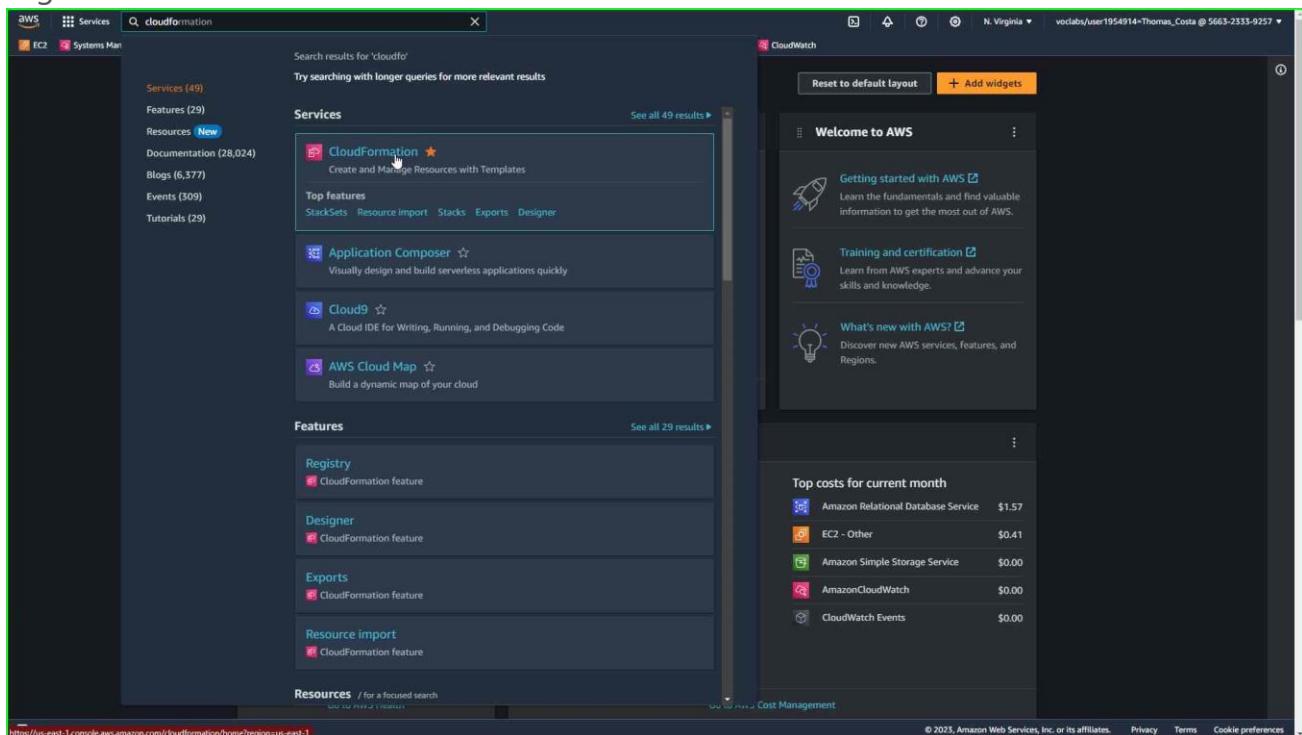
## Gerenciando configurações com Parameter Store e efetuando o deploy com CloudFormation

Entre na opção "Search":



The screenshot shows the AWS Console Home page. At the top, there's a navigation bar with links for EC2, Systems Manager, Simple Queue Service, Simple Notification Service, S3, IAM, RDS, CloudFormation, VPC, Lambda, API Gateway, and CloudWatch. Below the navigation bar is the "Console Home" header with a "Info" link. On the left, there's a sidebar titled "Recently visited" with links to CloudFormation, S3, RDS, EC2, Systems Manager, Simple Notification Service, and Lambda. To the right of the sidebar is a "Welcome to AWS" section with links to "Getting started with AWS", "Training and certification", and "What's new with AWS?". The main content area has two sections: "AWS Health" (with 0 open issues, 0 scheduled changes, and 0 other notifications) and "Cost and usage" (showing current month costs of \$1.98, forecasted end costs of \$2.50, and last month costs of \$0.00). A "Top costs for current month" table is also present. At the bottom, there are links to "Go to AWS Health" and "Go to AWS Cost Management". The footer includes links for CloudShell, Feedback, and various AWS services.

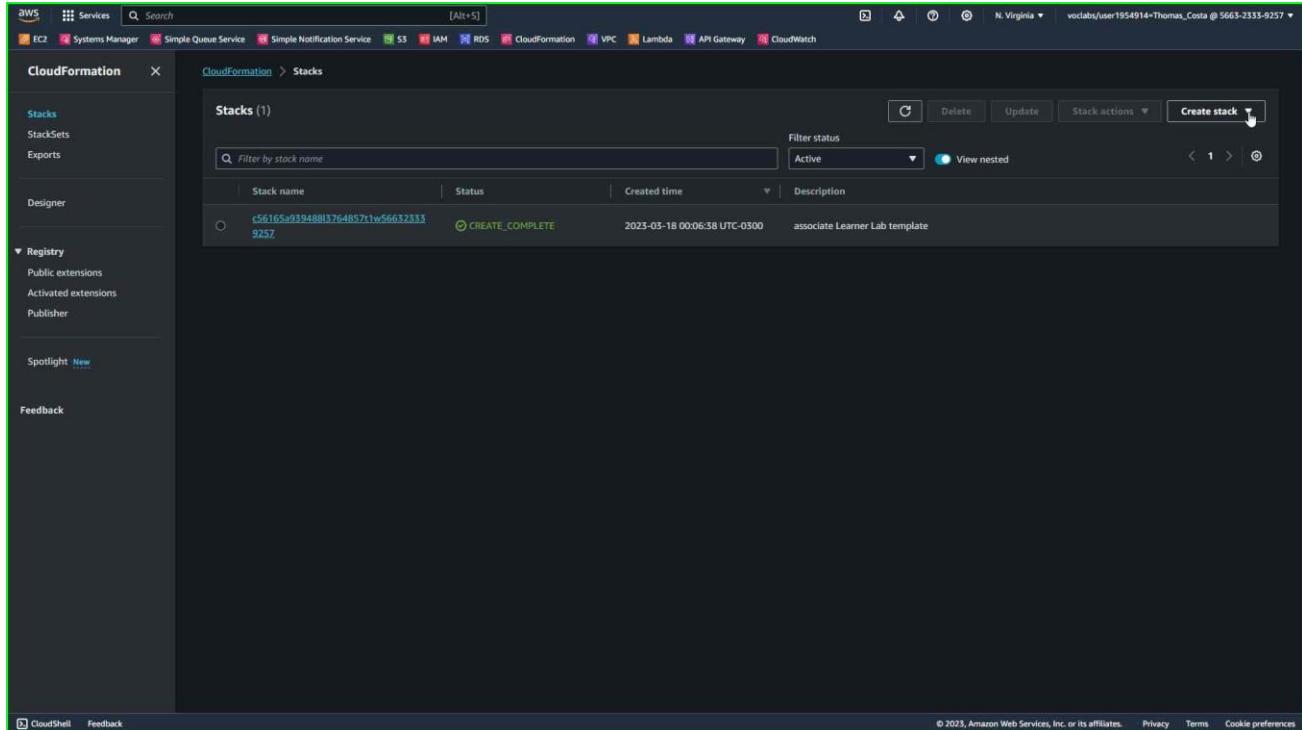
Digite "CloudFormation":



The screenshot shows the AWS search results for the query "cloudformation". The search bar at the top contains the text "cloudformation". The results are categorized into "Services" (49), "Features" (29), and "Resources" (New). In the "Services" category, the "CloudFormation" entry is highlighted, showing its description: "Create and Manage Resources with Templates" and listing "Top features" like StackSets, Resource Import, Stacks, Exports, and Designer. Other services listed include Application Composer, Cloud9, and AWS Cloud Map. In the "Features" category, there are entries for Registry, Designer, Exports, and Resource Import, all marked as "CloudFormation feature". The "Resources" category is shown with a note: "for a focused search". The right side of the screen shows the same "Welcome to AWS" and "Cost and usage" sections as the previous screenshot. The footer includes links for CloudShell, Feedback, and various AWS services.

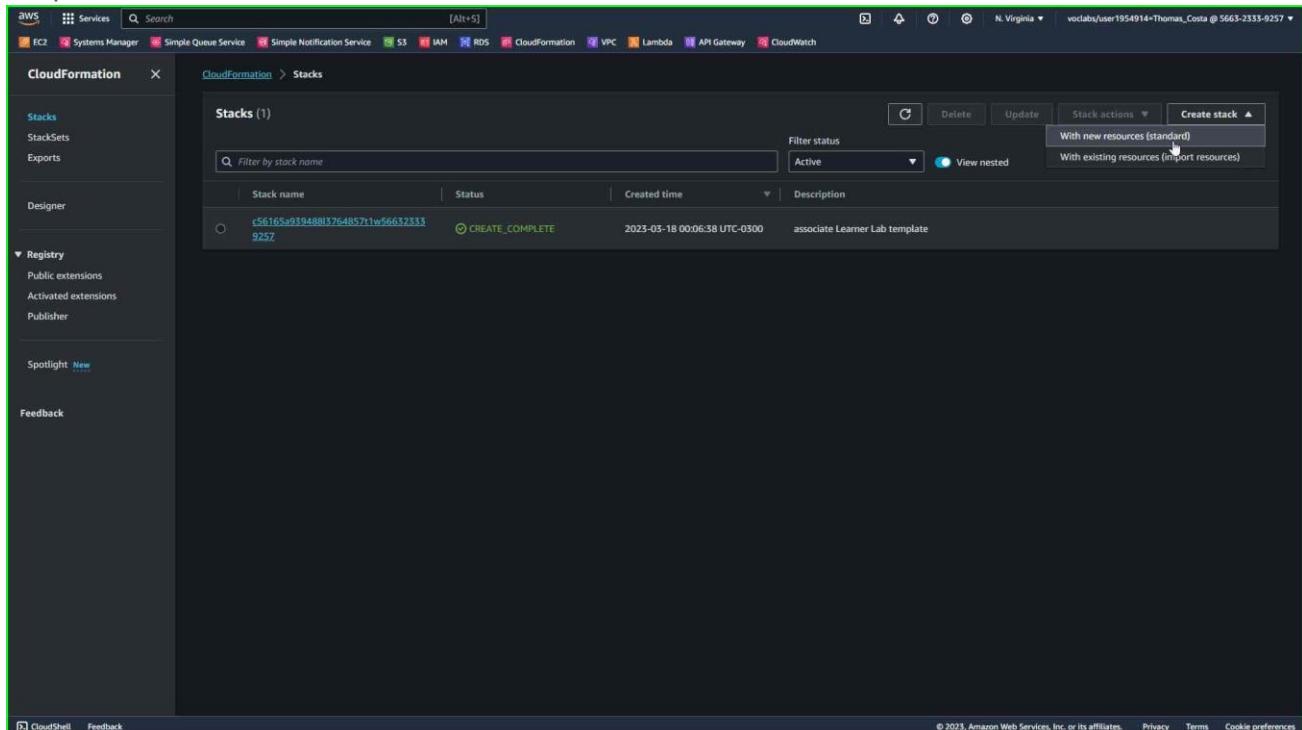
## Gerenciando configurações com Parameter Store e efetuando o deploy com CloudFormation

Clique em "Create stack":



The screenshot shows the AWS CloudFormation service in the AWS Management Console. The left sidebar has 'Stacks' selected. The main area displays a table titled 'Stacks (1)'. A single row is shown with the following details: Stack name 'c5e165a939488137648571w566373359257', Status 'CREATE\_COMPLETE', Created time '2023-03-18 00:06:38 UTC-0500', and Description 'associate Learner Lab template'. At the top right of the table, there is a 'Create stack' button with a small icon. The entire screenshot is framed by a green border.

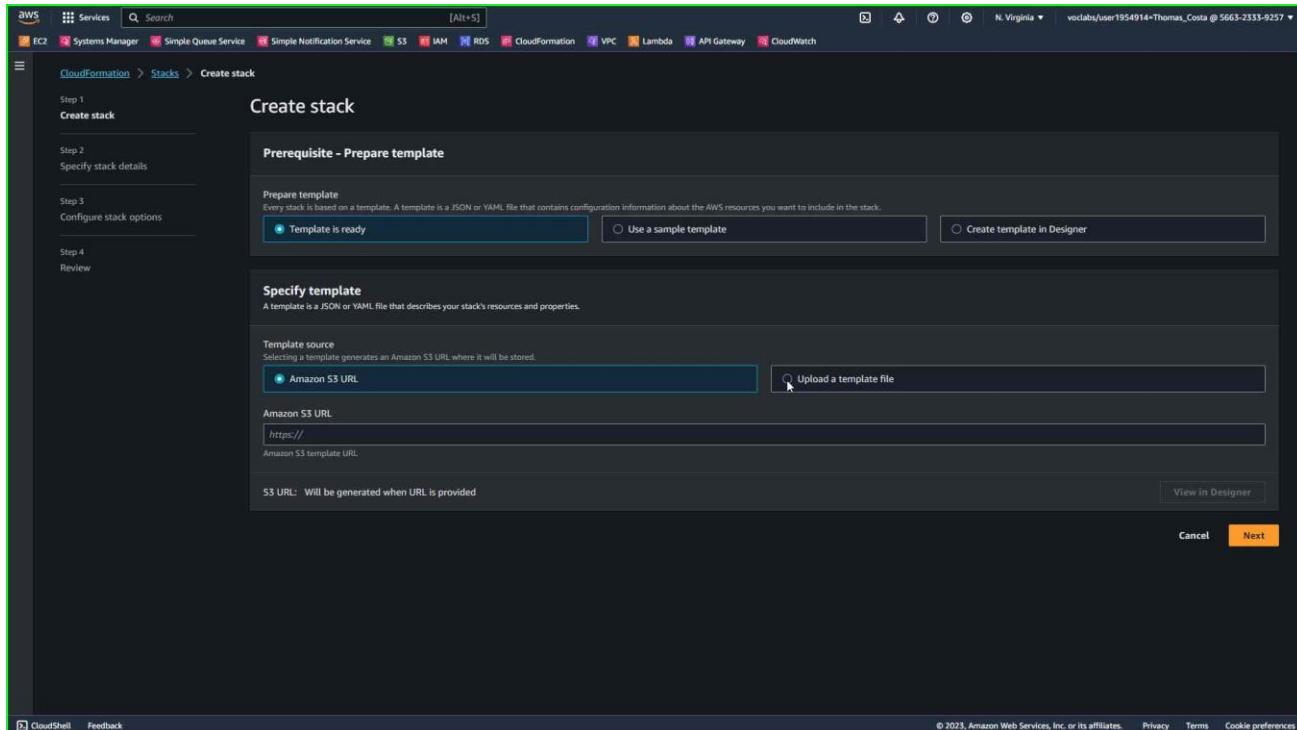
Clique em "With new resources (standard)":



This screenshot is identical to the one above, showing the AWS CloudFormation service in the AWS Management Console. The left sidebar is the same, showing 'Stacks' selected. The main area shows the 'Stacks (1)' table with the same row details. However, the 'Stack actions' dropdown at the top right is now open, revealing two options: 'With new resources (standard)' and 'With existing resources (import resources)'. The 'With new resources (standard)' option is highlighted with a gray background. The entire screenshot is framed by a green border.

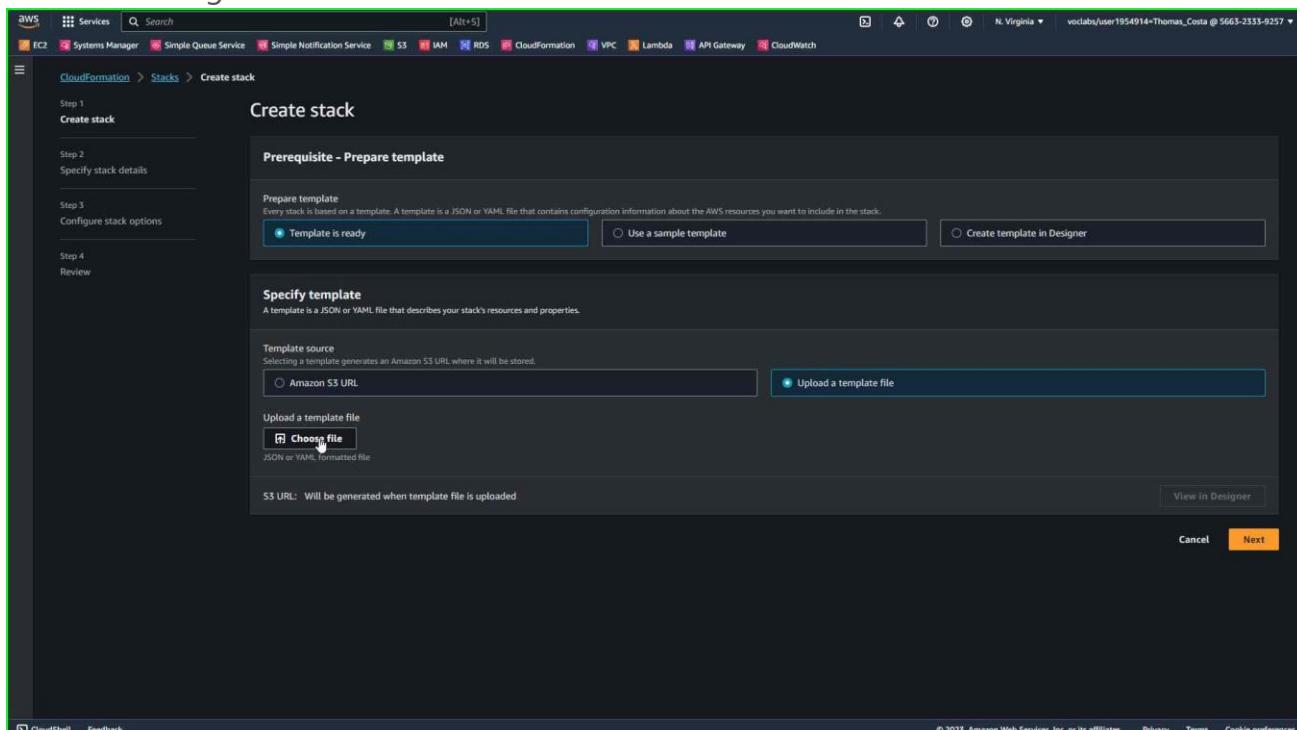
## Gerenciando configurações com Parameter Store e efetuando o deploy com CloudFormation

Clique em “Upload a template file”:



The screenshot shows the AWS CloudFormation 'Create stack' wizard. In Step 1: Prerequisite - Prepare template, the 'Template is ready' option is selected. Under 'Specify template', the 'Amazon S3 URL' input field contains 'https://...'. The 'Upload a template file' button is highlighted with a cursor.

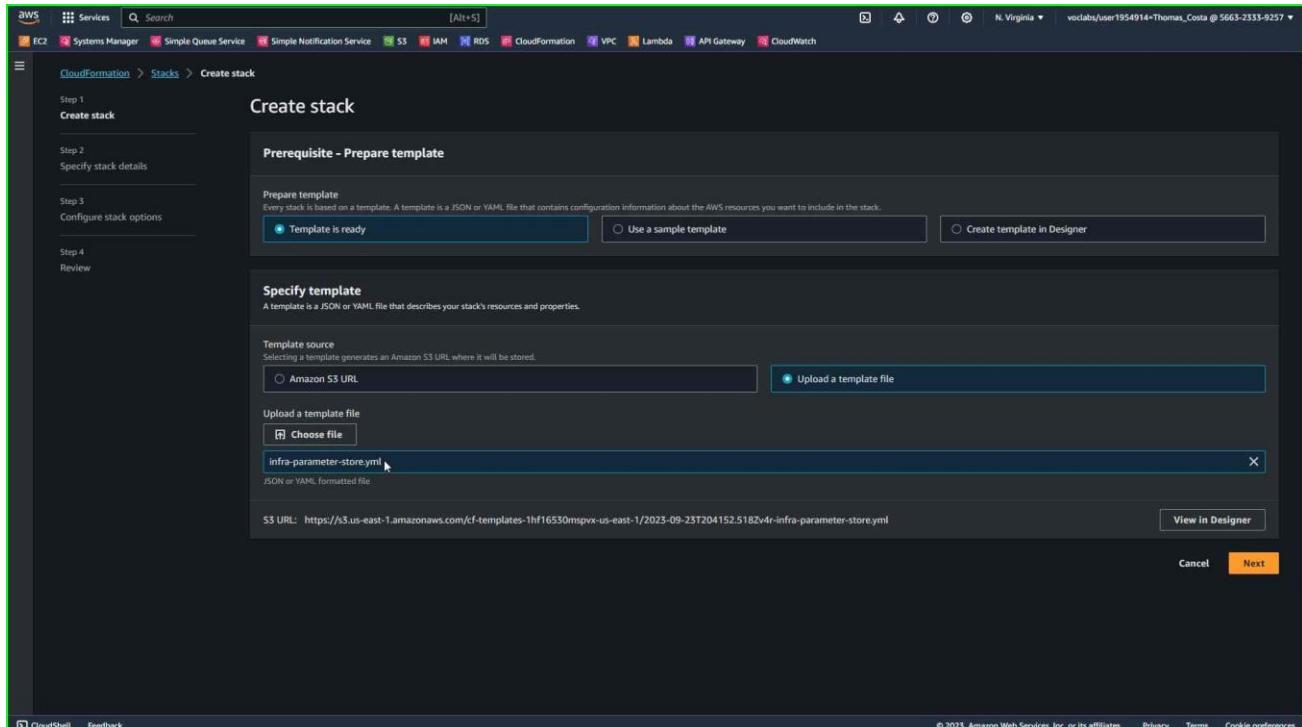
Crie um arquivo com extensão .yml do link [https://github.com/thomasdacosta/aula-aws/blob/main/aula\\_09\\_configuracao/infra-parameter-store.yml](https://github.com/thomasdacosta/aula-aws/blob/main/aula_09_configuracao/infra-parameter-store.yml) e faça o upload clicando no link da imagem abaixo:



The screenshot shows the AWS CloudFormation 'Create stack' wizard. In Step 1: Prerequisite - Prepare template, the 'Template is ready' option is selected. Under 'Specify template', the 'Upload a template file' section shows a 'Choose file' button with a cursor hovering over it.

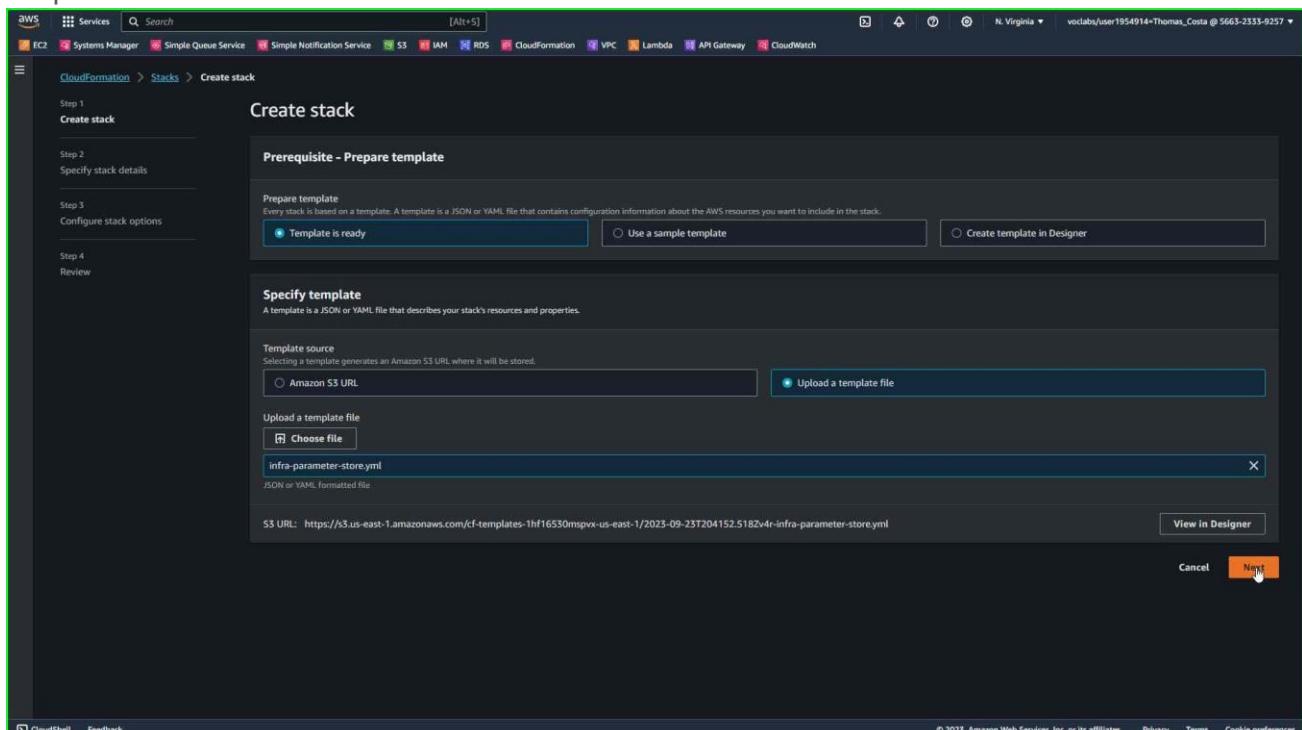
## Gerenciando configurações com Parameter Store e efetuando o deploy com CloudFormation

Arquivo carregado com sucesso:



The screenshot shows the AWS CloudFormation 'Create stack' wizard. The current step is 'Prerequisite - Prepare template'. Under 'Prepare template', the 'Template is ready' radio button is selected. In the 'Specify template' section, the 'Upload a template file' button is highlighted, and the file 'infra-parameter-store.yml' is listed. The bottom right corner shows the 'Next' button.

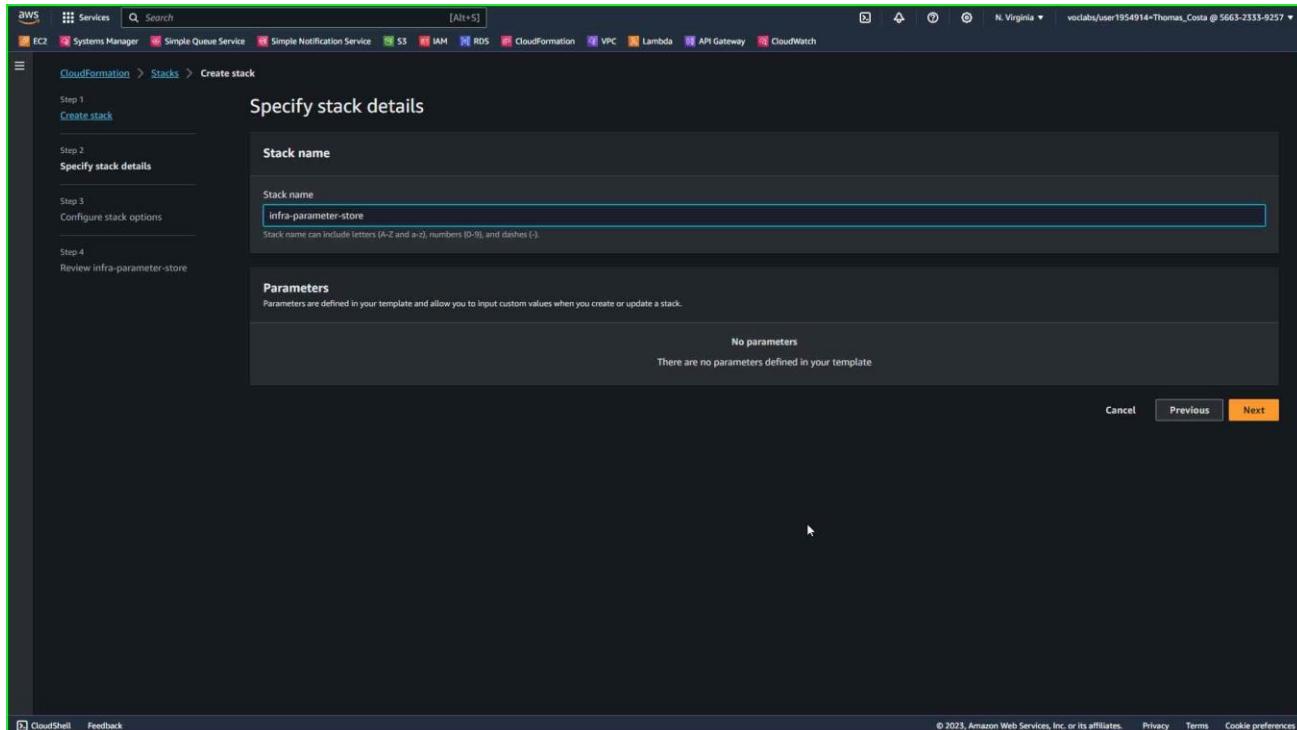
Clique no botão "Next":



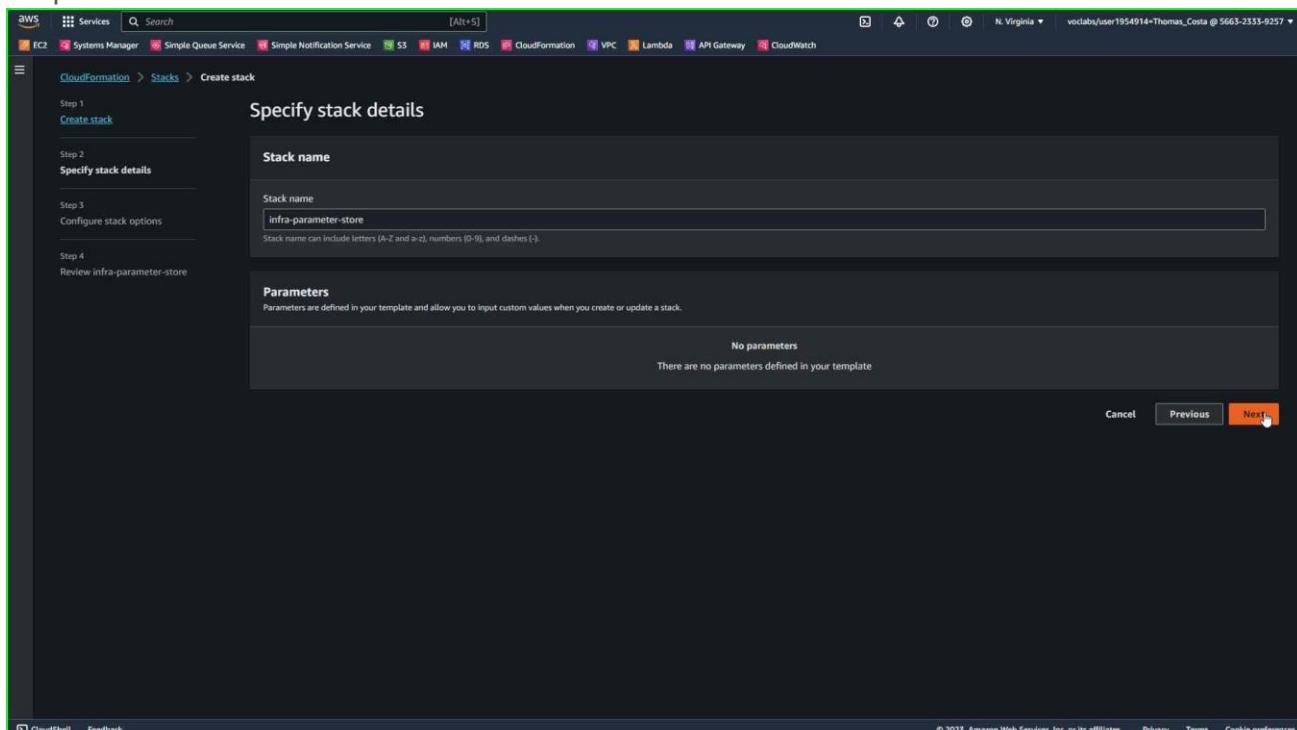
The screenshot shows the AWS CloudFormation 'Create stack' wizard. The current step is 'Prerequisite - Prepare template'. Under 'Prepare template', the 'Template is ready' radio button is selected. In the 'Specify template' section, the 'Upload a template file' button is highlighted, and the file 'infra-parameter-store.yml' is listed. The bottom right corner shows the 'Next' button, which is currently being clicked, indicated by a mouse cursor over it.

## **Gerenciando configurações com Parameter Store e efetuando o deploy com CloudFormation**

Coloque no nome da stack conforme imagem abaixo:



Clique no botão “Next”:



## Gerenciando configurações com Parameter Store e efetuando o deploy com CloudFormation

Selecione o combo da imagem abaixo:

The screenshot shows the AWS CloudFormation 'Create stack' wizard at Step 3: Configure stack options. In the 'Permissions' section, the 'IAM role - optional' dropdown is set to 'Sample-role-name'. Under 'Stack failure options', the radio button for 'Roll back all stack resources' is selected. The 'Advanced options' section is collapsed.

Selecione a opção "LabRole":

The screenshot shows the AWS CloudFormation 'Create stack' wizard at Step 3: Configure stack options. In the 'Permissions' section, the 'IAM role - optional' dropdown is open, showing 'LabRole' selected from a list. Under 'Stack failure options', the radio button for 'Roll back all stack resources' is selected. The 'Advanced options' section is collapsed.

## Gerenciando configurações com Parameter Store e efetuando o deploy com CloudFormation

Opção selecionada:

The screenshot shows the 'Configure stack options' step of the CloudFormation wizard. In the 'Permissions' section, an IAM role named 'LabRole' is selected. A note below states: 'AWS CloudFormation will use this role for all stack operations. Other users that have permissions to operate on this stack will be able to use this role, even if they don't have permission to pass it. Ensure that this role grants least privilege.'

Clique no botão "Next":

The screenshot shows the 'Review infra-parameter-store' step of the CloudFormation wizard. The 'Advanced options' section is expanded, showing four sections: 'Stack policy', 'Rollback configuration', 'Notification options', and 'Stack creation options'. The 'Next' button at the bottom right is highlighted.

## Gerenciando configurações com Parameter Store e efetuando o deploy com CloudFormation

Não modifique as opções abaixo:

Review infra-parameter-store

Step 1: Specify template

Template

Template URL: https://s3.us-east-1.amazonaws.com/cf-templates-1hf16530mpvx-us-east-1/2023-09-23T204152.518Zv4r-infra-parameter-store.yml

Stack description: Criando os parametros da aplicacao Spring Boot

Step 2: Specify stack details

Stack name: Infra-parameter-store

Parameters

No parameters

There are no parameters defined in your template

Step 3: Configure stack options

Clique em "Submit":

Stack policy

No stack policy

There is no stack policy defined

Rollback configuration

Monitoring time

CloudWatch alarm ARN

Notification options

SNS topic ARN

No notification options

There are no notification options defined

Stack creation options

Timeout

Termination protection: Deactivated

Quick-create link

Create change set

Cancel Previous Submit

## Gerenciando configurações com Parameter Store e efetuando o deploy com CloudFormation

O CloudFormationa começará a criar os parametros no Parameter Store:

The screenshot shows the AWS CloudFormation console with the stack 'infra-parameter-store' in progress. The 'Events' tab displays a single event: 'CREATE\_IN\_PROGRESS' for the stack, initiated by 'User Initiated'.

Parametros criados com sucesso:

The screenshot shows the AWS CloudFormation console with the stack 'infra-parameter-store' successfully created. The 'Events' tab lists 32 events, all of which are 'CREATE\_COMPLETE' status, indicating successful parameter creation.

Timestamp	Logical ID	Status	Status reason
2023-09-23 17:42:49 UTC-0300	infra-parameter-store	CREATE_COMPLETE	-
2023-09-23 17:42:48 UTC-0300	DbUrlParameter	CREATE_COMPLETE	-
2023-09-23 17:42:48 UTC-0300	SnsNotificationNameParameter	CREATE_COMPLETE	-
2023-09-23 17:42:48 UTC-0300	S3BucketParameter	CREATE_COMPLETE	-
2023-09-23 17:42:48 UTC-0300	Valor3Parameter	CREATE_COMPLETE	-
2023-09-23 17:42:48 UTC-0300	Valor2Parameter	CREATE_COMPLETE	-
2023-09-23 17:42:48 UTC-0300	DbPassParameter	CREATE_COMPLETE	-
2023-09-23 17:42:47 UTC-0300	SqQueueNameParameter	CREATE_COMPLETE	-
2023-09-23 17:42:47 UTC-0300	DbUserParameter	CREATE_COMPLETE	-
2023-09-23 17:42:47 UTC-0300	HelloWorldParameter	CREATE_COMPLETE	-
2023-09-23 17:42:47 UTC-0300	Valor1Parameter	CREATE_COMPLETE	-
2023-09-23 17:42:47 UTC-0300	DbUrlParameter	CREATE_IN_PROGRESS	Resource creation initiated
2023-09-23 17:42:47 UTC-0300	SnsNotificationNameParameter	CREATE_IN_PROGRESS	Resource creation initiated
2023-09-23 17:42:47 UTC-0300	Valor3Parameter	CREATE_IN_PROGRESS	Resource creation initiated
2023-09-23 17:42:47 UTC-0300	Valor2Parameter	CREATE_IN_PROGRESS	Resource creation initiated
2023-09-23 17:42:47 UTC-0300	S3BucketParameter	CREATE_IN_PROGRESS	Resource creation initiated

## Gerenciando configurações com Parameter Store e efetuando o deploy com CloudFormation

Procure por "Systems Manager":

The screenshot shows the AWS CloudSearch interface with a search bar at the top containing 'system manager'. Below the search bar, there is a message: 'Try searching with longer queries for more relevant results'. The main area displays search results under three categories: Services, Features, and Resources.

- Services:** Includes 'Systems Manager' (highlighted with a star), 'Incident Manager', 'EFS', and 'FSx'.
- Features:** Includes 'Inventory', 'Maintenance Windows', 'Run Command', and 'Application Manager'.
- Resources:** A link for a focused search.

On the right side of the interface, there is a CloudWatch Metrics viewer showing a table of metrics with columns for 'Status' and 'Status reason'. Most entries show 'CREATE\_COMPLETE' status, while some show 'CREATE\_IN\_PROGRESS' with a note about resource creation initiated.

Selecione a opção "Parameter Store":

The screenshot shows the AWS Systems Manager Parameter Store page. The left sidebar contains a navigation menu with several collapsed sections. The 'Parameter Store' section is expanded, showing its sub-options: 'Group your resources', 'View insights', and 'Take action'. The main content area features a heading 'AWS Systems Manager' with the subtext 'Gain Operational Insight and Take Action on AWS Resources.' Below this is a 'Get Started with Systems Manager' button and a brief description of operational data groups. The 'How it works' section includes three cards: 'Group your resources', 'View insights', and 'Take action'. The 'Features' section lists 'Remote connect' and 'Resource grouping' with their respective descriptions. On the right, there is a 'More resources' sidebar with links to 'Documentation', 'API reference', and 'FAQs'.

## Gerenciando configurações com Parameter Store e efetuando o deploy com CloudFormation

Configurações foram criadas:

The screenshot shows the AWS Systems Manager Parameter Store interface. The left sidebar has 'AWS Systems Manager' selected, and the main area shows 'Parameter Store'. Below that are tabs for 'My parameters', 'Public parameters', and 'Settings'. The 'My parameters' tab is selected. A search bar at the top has 'Search' entered. Below the search bar is a table with columns: Name, Tier, Type, and Last modified. There are 10 entries in the table, all of which are of type String and were last modified on Saturday, September 23, 2023, at 20:42:47 GMT. The entries are: /config/spring-boot-localstack\_production/dbPass, /config/spring-boot-localstack\_production/dbUrl, /config/spring-boot-localstack\_production/dbUser, /config/spring-boot-localstack\_production/helloWorld, /config/spring-boot-localstack\_production/s3Bucket, /config/spring-boot-localstack\_production/snsNotificat..., /config/spring-boot-localstack\_production/sqlQueueN..., /config/spring-boot-localstack\_production/valor1, /config/spring-boot-localstack\_production/valor2, and /config/spring-boot-localstack\_production/valor3.

Selecione uma configuração qualquer, somente para verificar o valor:

This screenshot is identical to the one above, showing the AWS Systems Manager Parameter Store interface. The 'My parameters' table is displayed with the same 10 entries. However, the entry '/config/spring-boot-localstack\_production/dbUrl' is now highlighted with a red box around its row, indicating it is selected for viewing. The rest of the interface remains the same, including the search bar and the table headers.

## Gerenciando configurações com Parameter Store e efetuando o deploy com CloudFormation

Clique no botão "Edit":

The screenshot shows the AWS Parameter Store Overview page for the path `/config/spring-boot-localstack_production/dbUrl`. The parameter details are as follows:

- Name:** /config/spring-boot-localstack\_production/dbUrl
- Tier:** Standard
- Type:** String
- Last modified date:** Sat, 23 Sep 2023 20:42:47 GMT
- Description:** -
- Data type:** text
- Last modified user:** arn:aws:sts::566323339257:assumed-role/LabRole/AWSCloudFormation
- Version:** 1
- Value:** jdbc:mysql://localhost:3306/dbspringbootlocalstack?createDatabaseIfNotExist=true

The "Edit" button is highlighted with a mouse cursor.

Valor da propriedade é mostrado abaixo:

The screenshot shows the AWS Parameter Store Edit parameter page for the path `/config/spring-boot-localstack_production/dbUrl`. The parameter details are as follows:

- Name:** /config/spring-boot-localstack\_production/dbUrl
- Description — Optional:** -
- Tier:** Standard  
Parameter Store offers standard and advanced parameters.  
Limit of 10,000 parameters. Parameter value size up to 4 KB.  
Parameter policies are not available. No additional charge.
- Advanced:** Can create more than 10,000 parameters. Parameter value size up to 8 KB. Parameter policies are available. Charges apply.
- Type:** String
- Data type:** Text
- Value:** jdbc:mysql://localhost:3306/dbspringbootlocalstack?createDatabaseIfNotExist=true

The "Save changes" button is highlighted with a mouse cursor.