

## Desafío # 8

**Realizado por: Joselin Teixeira**

*Fecha de entrega: 14/08/2024*

### Escenario:

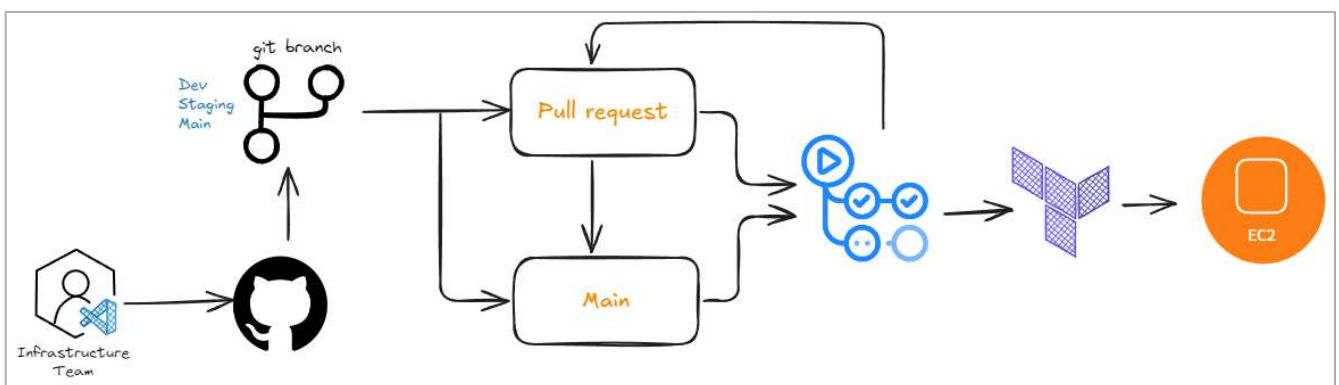
En nuestro sprint anterior trabajamos en modularizar nuestro proyecto de Ansible. Luego del éxito de este cambio el equipo identificó que existen playbook que solo quedan en el entorno local de los desarrolladores y es por esto por lo que necesitamos crear un pipeline CI/CD para forzar que la única manera de ejecutar e interactuar de ejecutar los playbook en los distintos entornos sea desde un código que sea leído de un repo y ejecutado desde un controlador de Jenkins.

Esta mejora en el proceso busca:

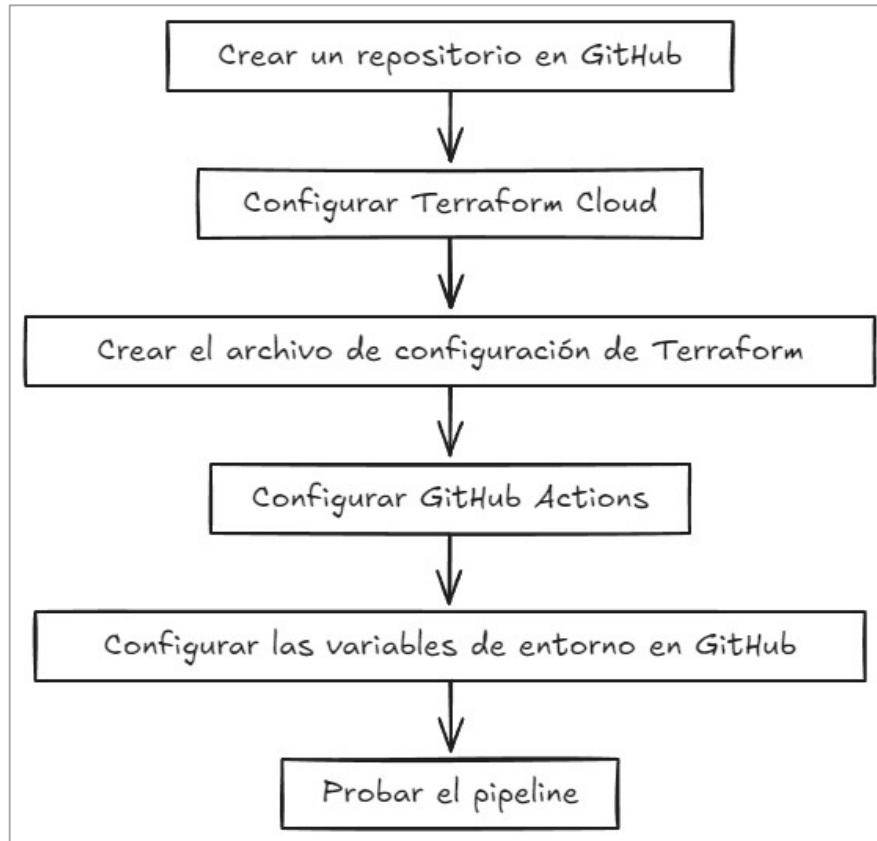
- Aumentar la seguridad ya que las credenciales de acceso solo estarán disponibles en el controlador de Jenkins y no será necesario que un desarrollador disponga de las credenciales de forma local.
- Permite forzar a los desarrolladores a crear sus entornos de trabajo local evitando que el proceso de desarrollo se ejecute en entornos compartidos.
- Todos los cambios podrán ser gestionados como una pieza de software y podrán pasar por un proceso de revisión de PR.
- Mantener la estrategia de branches que vienen utilizando:
  - DEV: entorno de desarrollo, es un entorno donde se llevan cambios frecuentes y los miembros del equipo tienen mayor libertad para ver los cambios en equipos reales.
  - STAGING: es un entorno donde se integran y prueban todos los cambios, los miembros del equipo no cuentan con acceso a los equipos y solo reciben feedback mediante las herramientas CI/CD y los cambios son aplicados solo mediante un PR (pull request).
  - MAIN: es nuestro entorno productivo, no contamos acceso y los cambios

### Requisitos:

1. Realizar la PoC aplicando todos los pasos detallados en el post propuesto.
2. Configurar un repositorio propio para que pueda alojar el proyecto.
3. Crear unas credenciales en nuestra sandbox de AWS Academy
4. Crear una cuenta en Terraform Cloud (no requiere tarjeta de crédito)

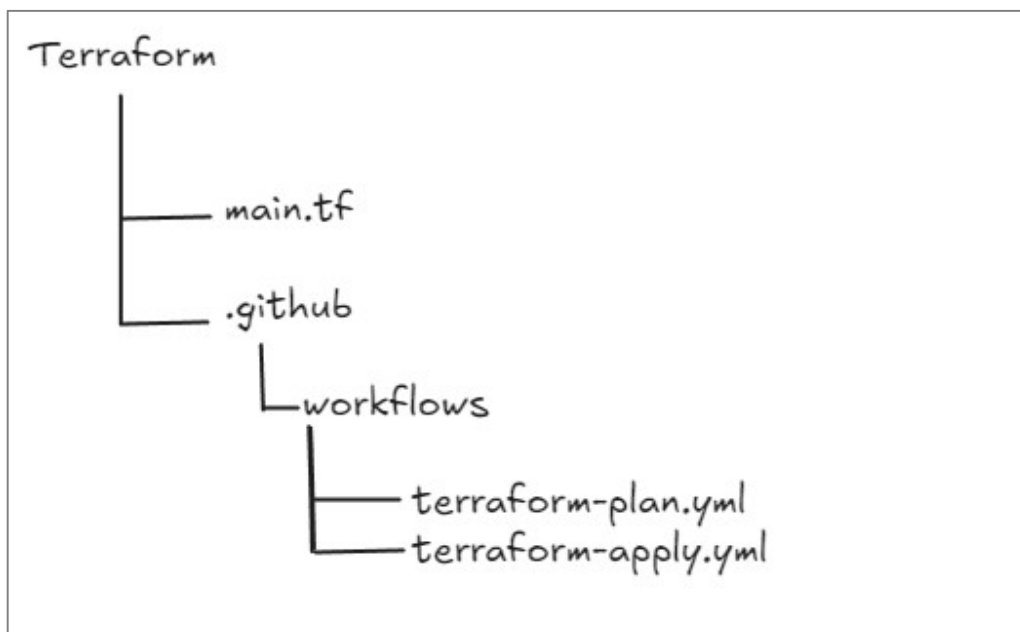


## Diagrama de flujo



### Crear Repositorio en GitHub

Crearemos un flujo de trabajo utilizando GitHub Actions y Terraform para implementar un servidor web de acceso público, crearemos la siguiente estructura en nuestro repositorio de GitHub:

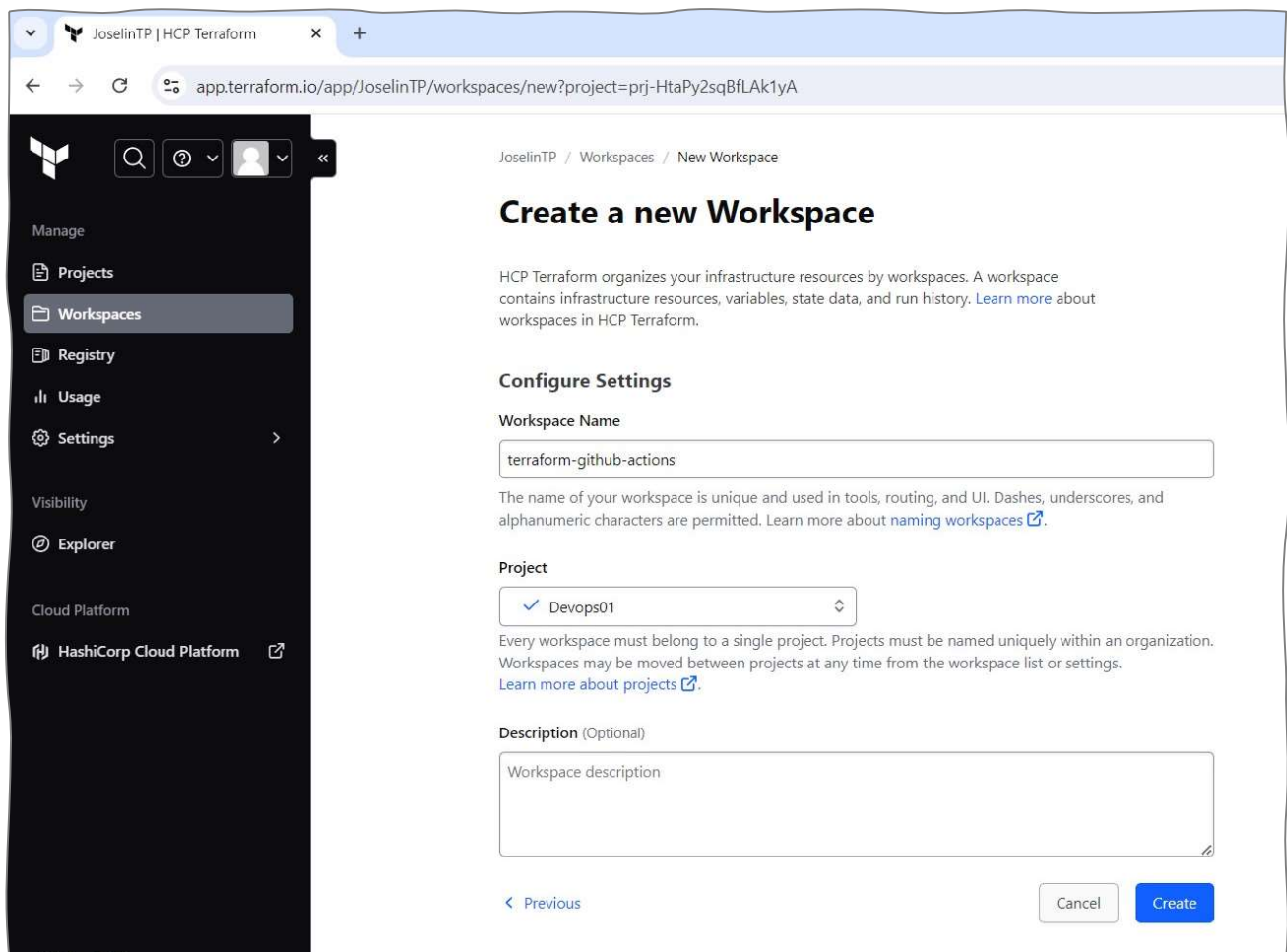


## Configurar Terraform Cloud

Crearemos un Nuevo Espacio de Trabajo **[terraform-github-actions]**:

Una vez que nuestra cuenta esté configurada, podemos comenzar a crear "espacios de trabajo" **[workspace]** para gestionar las configuraciones de Terraform.

- Vamos a la sección de "Workspaces" (Espacios de Trabajo) en la barra lateral.
- Hacemos clic en el botón "New Workspace" (Nuevo Espacio de Trabajo).
- Elegimos un nombre para nuestro espacio de trabajo **[terraform-github-actions]** y presionamos el botón [Create]



The screenshot shows the Terraform Cloud web interface. On the left is a dark sidebar with navigation links: Manage, Projects, Workspaces (highlighted), Registry, Usage, Settings, Visibility, Explorer, and Cloud Platform. The main content area is titled 'Create a new Workspace' and includes a brief description of workspaces. Below this is the 'Configure Settings' section with three fields: 'Workspace Name' (containing 'terraform-github-actions'), 'Project' (a dropdown menu showing 'Devops01'), and 'Description (Optional)' (a text area with 'Workspace description'). At the bottom are 'Previous', 'Cancel', and 'Create' buttons.

Una vez creado el Workspace (Espacios de Trabajo) configuramos las variables necesarias para la configuración de Terraform en el espacio de trabajo.

Para crear las variables de entorno `AWS_ACCESS_KEY_ID` y `AWS_SECRET_ACCESS_KEY` seguiremos estos pasos:

*Paso 1: Seleccionamos el Workspace*

1. En el panel de navegación, selecciona **Workspaces**.
2. Hacemos clic en el workspace **[terraform-github-actions]**.

*Paso 2: Configuramos Variables de Entorno*

1. Dentro del workspace, optamos por la pestaña **Variables**.
2. Aquí seleccionamos: **Environment Variables**.

### Paso 3: Agregar Variables de Entorno

1. En la sección **Environment Variables**, hacemos clic en **+ Add Variable**.
2. Para la primera variable:
  - **Key:** AWS\_ACCESS\_KEY\_ID.
  - **Value:** Ingresamos la Access Key ID de AWS.
  - Marca la opción **Sensitive** para que no se muestre en la interfaz.
  - Haz clic en **Save variable**.
3. Repetimos el proceso para la segunda variable:
  - **Key:** AWS\_SECRET\_ACCESS\_KEY.
  - **Value:** Ingresamos la Secret Access Key de AWS.
  - Marcamos la opción **Sensitive** y guardamos la variable.

**Workspace variables (0)**

Variables defined within a workspace always override variables from variable sets that have the same type and the same key. Learn more about variable set precedence.

Key	Value	Category
There are no variables added.		

Select variable category

☐ Terraform variable  
These variables should match the declarations in your configuration. Click the HCL box to use interpolation or set a non-string value.

☒ Environment variable  
These variables are available in the Terraform runtime environment.

Key:

Value:  ☒ Sensitive ⓘ

Description (Optional):

**Workspace variables (1)**

Variables defined within a workspace always override variables from variable sets that have the same type and the same key. Learn more about variable set precedence.

Key	Value	Category
AWS_ACCESS_KEY_ID	<i>Sensitive - write only</i>	env

Select variable category

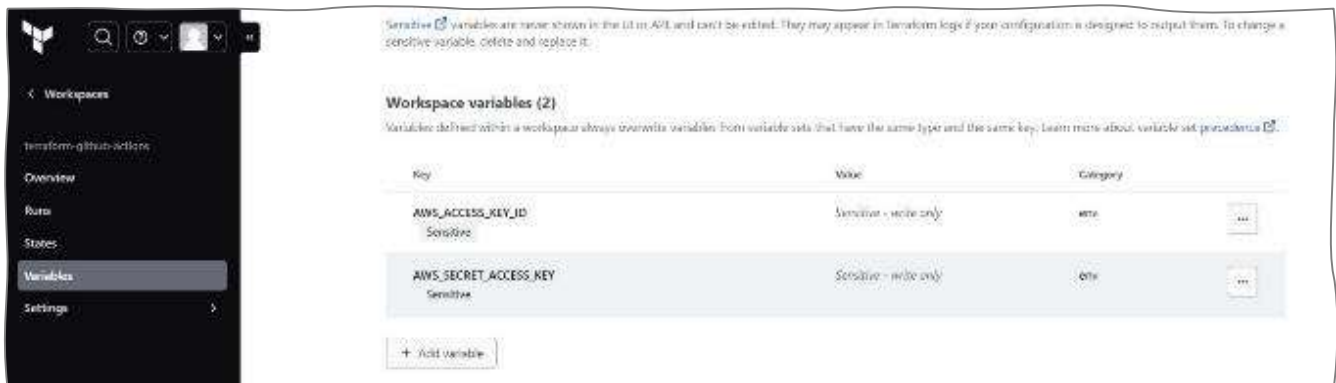
☐ Terraform variable  
These variables should match the declarations in your configuration. Click the HCL box to use interpolation or set a non-string value.

☒ Environment variable  
These variables are available in the Terraform runtime environment.

Key:

Value:  ☒ Sensitive ⓘ

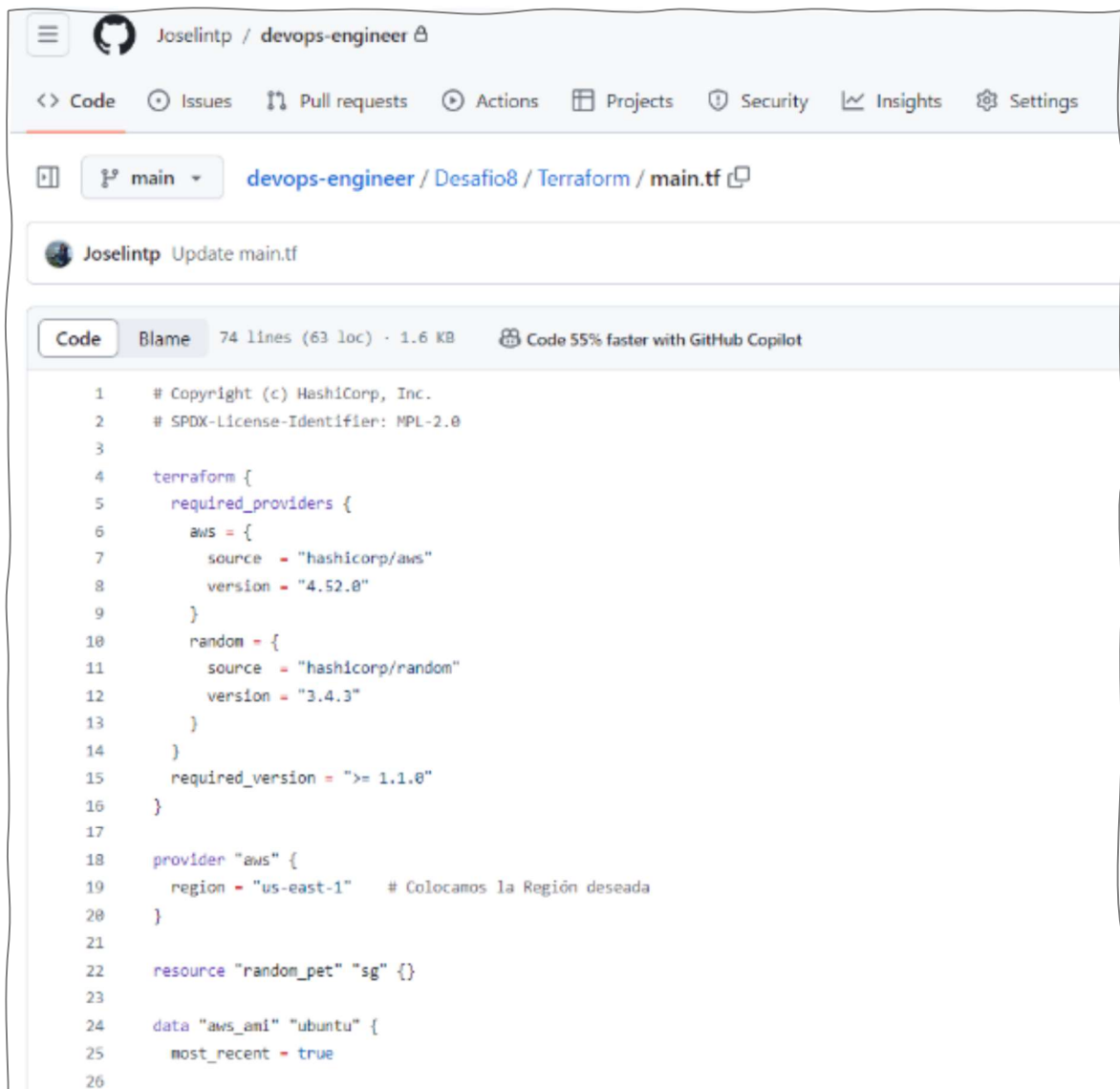
Description (Optional):



## Crear archivo de configuración Terraform

Añadiremos el archivo a nuestro repositorio de GitHub: [Desafio8/Terraform/main.tf](#)

Este archivo contiene la configuración de Terraform para implementar una instancia EC2 de acceso público

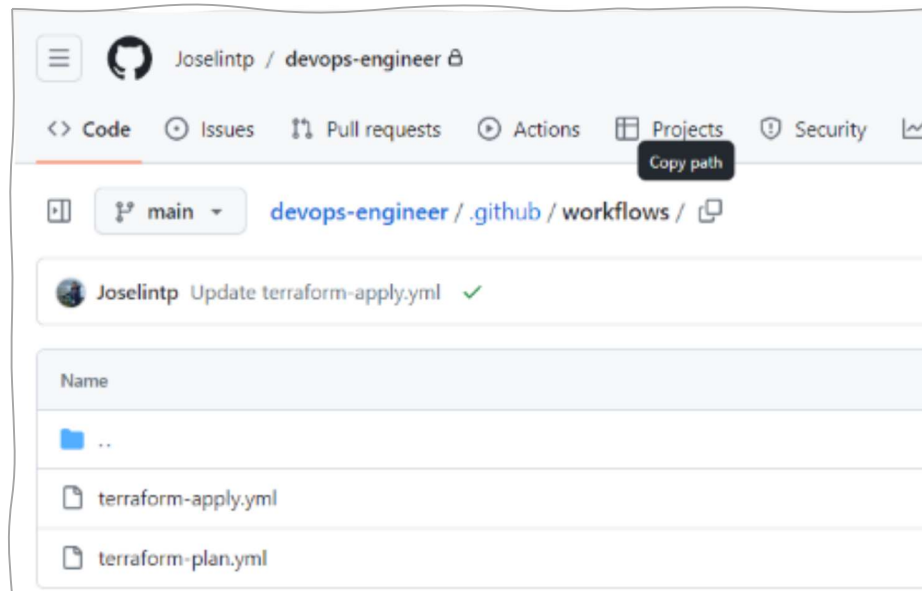


## Configurar Github Actions

Estableceremos los archivos de flujo de trabajo:

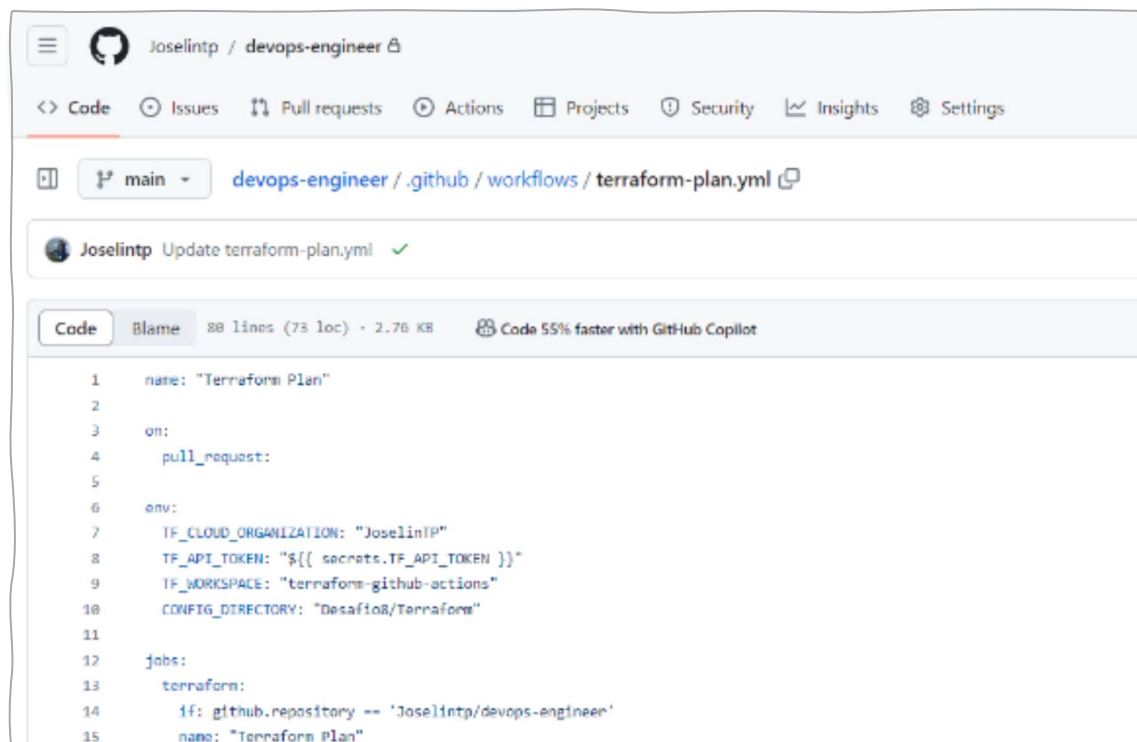
`.github/workflows/terraform-plan.yml` (que ejecuta el plan de Terraform)

`.github/workflows/terraform-apply.yml` (que ejecuta terraform apply)



## Archivo terraform-plan

El archivo `terraform-plan.yml` define un flujo de trabajo de GitHub Actions que ejecuta un plan de Terraform en un espacio de trabajo de Terraform Cloud cada vez que se crea un pull request en el repositorio.



A continuación, se detalla cada sección de dicho archivo:

- a) **name:** Define el nombre del flujo de trabajo como "Terraform Plan".
- b) **on:** Especifica que el flujo de trabajo se activará cuando se cree un pull request en el repositorio.
- c) **env:** Define variables de entorno utilizadas en el flujo de trabajo:
  - **TF\_CLOUD\_ORGANIZATION:** El nombre de la organización de Terraform Cloud. "JoselinTP"
  - **TF\_API\_TOKEN:** El token de API de Terraform Cloud, que se obtiene de los secrets de GitHub.
  - **TF\_WORKSPACE:** El nombre del espacio de trabajo de Terraform Cloud donde se ejecutará el plan. "terraform-github-actions"
  - **CONFIG\_DIRECTORY:** La ruta al directorio que contiene la configuración de Terraform. "Desafio8/Terraform"
- d) **jobs:** Define un trabajo llamado "terraform" que se ejecutará cuando se active el flujo de trabajo.
- e) **if:** Agrega una condición para que se ejecute en el repositorio de 'Joselintp/devops-engineer'
- f) **name:** Establece el nombre del trabajo como "Terraform Plan".
- g) **runs-on:** Especifica que el trabajo se ejecutará en una máquina virtual de Ubuntu.
- h) **permissions:** Otorga permisos de lectura al contenido del repositorio y de escritura a los pull requests.
- i) **steps:** Define los pasos a seguir dentro del trabajo:
  - **Checkout:** Descarga el código fuente del repositorio.
  - **Upload Configuration:** Carga la configuración de Terraform al espacio de trabajo de Terraform Cloud.
  - **Create Plan Run:** Crea un plan de ejecución en Terraform Cloud.
  - **Get Plan Output:** Obtiene los resultados del plan de ejecución.
  - **Update PR:** Agrega un comentario al pull request con los resultados del plan.

En resumen, este archivo se encarga de cargar la configuración de Terraform en la Cloud, crear un plan de ejecución y luego obtener los resultados del plan.

### El archivo terraform-apply.yml

Este **terraform-apply.yml** archivo se utiliza para automatizar la aplicación de configuraciones de Terraform en un entorno de nube.

El archivo está dividido en varias secciones clave:

#### 1. **Nombre del Workflow**

```
name: "Terraform Apply"
```

## 2. Eventos que lo disparan

```
on:  
  push:  
    branches:  
      - main
```

## 3. Variables de entorno

```
env:  
  TF_CLOUD_ORGANIZATION: "JoselinTP"  
  TF_API_TOKEN: "${{ secrets.TF_API_TOKEN }}"  
  TF_WORKSPACE: "terraform-github-actions"  
  CONFIG_DIRECTORY: "Desafio8/Terraform" # Ruta relativa al directorio que contiene main.tf
```

Aquí se definen varias variables de entorno que se utilizarán en los pasos del trabajo:

- **TF\_CLOUD\_ORGANIZATION:** El nombre de la organización en Terraform Cloud.
- **TF\_API\_TOKEN:** Un token de API que se almacena de forma segura en los secretos de GitHub para autenticar las solicitudes a Terraform Cloud.
- **TF\_WORKSPACE:** El espacio de trabajo de Terraform donde se aplicarán las configuraciones.
- **CONFIG\_DIRECTORY:** La ruta relativa al directorio que contiene el archivo main.tf, que es el archivo principal de configuración de Terraform.

## 4. Definición de trabajos (jobs)

```
jobs:  
  terraform:  
    if: github.repository == 'Joselintp/devops-engineer'  
    name: "Terraform Apply"  
    runs-on: ubuntu-latest
```

Aquí se define un trabajo llamado terraform que solo se ejecutará si el repositorio es [Joselintp/devops-engineer]. Este trabajo se ejecutará en un entorno de Ubuntu más reciente.

## 5. Pasos dentro de los trabajos

Los siguientes pasos son las acciones que se ejecutarán dentro del trabajo Terraform:

### Paso 1: Checkout

```
text  
- name: Checkout  
  uses: actions/checkout@v4
```

Este paso utiliza la acción de checkout para clonar el repositorio en el entorno de ejecución. Esto permite que los siguientes pasos tengan acceso a los archivos del repositorio.

### Paso 2: Upload Configuration

```
text  
- name: Upload Configuration
```



```
uses: hashicorp/tfc-workflows-github/actions/upload-configuration@v1.0.0
id: apply-upload
with:
  workspace: ${{ env.TF_WORKSPACE }}
  directory: ${{ env.CONFIG_DIRECTORY }}
```

Este paso sube la configuración de Terraform al workspace especificado en Terraform Cloud. Se utiliza la acción upload-configuration de HashiCorp, y se le pasa el directorio que contiene los archivos de configuración.

### Paso 3: Create Apply Run

```
text
- name: Create Apply Run
  uses: hashicorp/tfc-workflows-github/actions/create-run@v1.0.0
  id: apply-run
  with:
    workspace: ${{ env.TF_WORKSPACE }}
    configuration_version: ${{ steps.apply-upload.outputs.configuration_version_id }}
```

Se crea una ejecución de aplicación (apply run) en Terraform Cloud. Este paso utiliza la acción create-run y toma como entrada el ID de la versión de configuración que se obtuvo en el paso anterior.

### Paso 4: Apply

```
text
- name: Apply
  uses: hashicorp/tfc-workflows-github/actions/apply-run@v1.0.0
  if: fromJSON(steps.apply-run.outputs.payload).data.attributes.actions.isConfirmable
  id: apply
  with:
    run: ${{ steps.apply-run.outputs.run_id }}
    comment: "Apply Run from GitHub Actions CI ${{ github.sha }}"
```

Finalmente, este paso aplica la configuración de Terraform en el workspace. Se utiliza la acción apply-run y solo se ejecuta si la ejecución es confirmable, lo que se verifica mediante una condición. Se añade un comentario que incluye el SHA del commit actual para rastrear la ejecución.

Este archivo de configuración de GitHub Actions permite automatizar el proceso de aplicar configuraciones de Terraform en un entorno de nube, asegurando que los cambios en la rama main se reflejen automáticamente en la infraestructura definida por Terraform. Cada paso está diseñado para interactuar con Terraform Cloud, haciendo que el proceso sea eficiente y seguro.

### **Referencias:**

[Documentación oficial de actions/checkout](#)

### **Configurar Variables de Entorno GitHub**

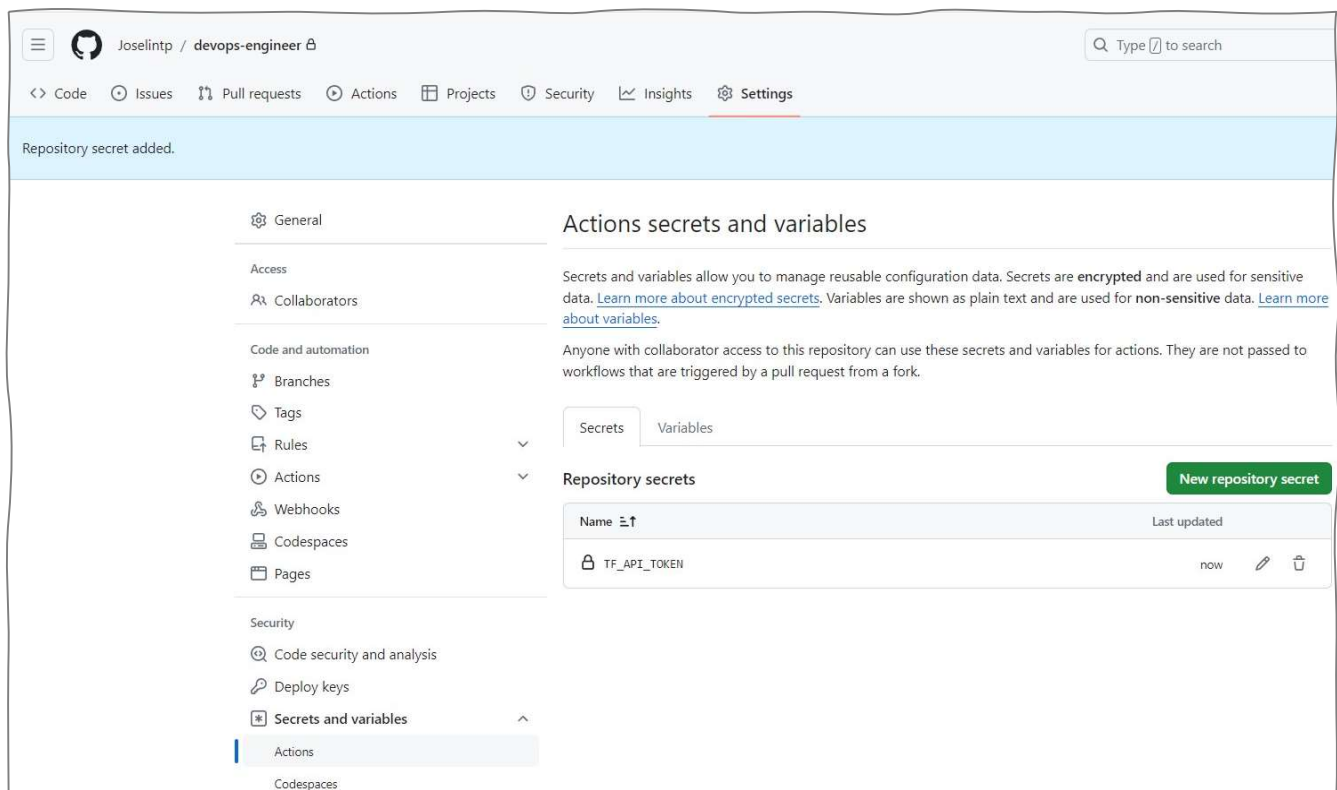
1. Vamos a nuestro repositorio en GitHub y seleccionamos **Settings**.
2. En el menú de la izquierda, seleccionamos **Secrets and variables > Actions**.

3. Hacemos clic en la Ficha **Secrets** y presionamos el botón: **New repository secret**. Luego crearemos la siguiente variable:

- TF\_API\_TOKEN: Será el token de API de Terraform Cloud.



Es importante guardar el token en un lugar seguro



## Probar el pipeline

1. Realizamos un commit en la rama main con cambios en archivo main.tf.
2. Creamos un pull request para fusionar los cambios.

- Finalmente verificamos que el pipeline de GitHub Actions se ejecute correctamente, realizando la inicialización, validación, planificación y aplicación de los cambios en Terraform Cloud.

### **Evidencia de pruebas exitosas**

✓ **Plan finished** 2 days ago

Resources: 3 to add, 0 to change, 0 to destroy ^

Started 2 days ago > Finished 2 days ago

+ 3 to create



Filter resources by address...



Filter by action ▾



☐ Show data sources

Terraform 1.9.4

Download raw log

> +  aws\_instance.web 

> +  aws\_security\_group.web-sg 

> +  random\_pet.sg 

> **Outputs** 1 planned to change

Download Sentinel mocks

ⓘ Sentinel mocks can be used for [testing your Sentinel policies](#)

✓ **Apply finished** 2 days ago

Resources: 3 added, 0 changed, 0 destroyed ^

Started 2 days ago > Finished 2 days ago



+ 3 created

Filter resources by address...



Filter by action ▾

Terraform 1.9.4



Download raw log

> +  aws\_instance.web 

✓ Created id=i-02cd77f5d492cdddd

> +  aws\_security\_group.web-sg 

✓ Created id=sg-05dea16dfe87ba2b5

> +  random\_pet.sg 

✓ Created id=stirred-dinosaur

✓ **Outputs** 1 total

web-address : "ec2-52-90-200-49.compute-1.amazonaws.com:8080"

State versions created:  
[JoselinTP/terraform-github-actions#sv-S9XdZueMbkwDEbNH](#) (Aug 12, 2024 21:51:36 pm)

Features • GitHub Actions

Update terraform-plan.yml - Joselintp

github.com/Joselintp/devops-engineer/actions/runs/10361771971/job/28682811904

github.com/Joselintp / devops-engineer

CodeIssuesPull requests1ActionsProjectsSecurityInsightsSettings

Terraform Plan

Update terraform-plan.yml #5

Re-run all jobsLatest #2

Summary

Jobs

Terraform Plan

Run detailsUsageWorkflow file

Terraform Plan

succeeded 2 minutes ago in 30s

Search logs

Set up job1s

Pull hashicorp/terraform1.0.00s

Checkout1s

Upload Configuration2s

Create Plan Run21s

Get Plan Output0s

Update PR1s

Post Checkout0s

Complete job0s

Features • GitHub Actions

Update terraform-plan.yml - Joselintp

github.com/Joselintp/devops-engineer/actions/runs/10361771971/job/28682811904

github.com/Joselintp / devops-engineer

CodeIssuesPull requests1ActionsProjectsSecurityInsightsSettings

Terraform Plan

Update terraform-plan.yml #5

Re-run all jobsLatest #2

Summary

Jobs

Terraform Plan

Run detailsUsageWorkflow file

Terraform Plan

succeeded 2 minutes ago in 30s

Search logs

Set up job1s

1 Current runner version: "2.319.0"

2 Operating System

3 Runner Image

4 Runner Image Provisioner

5 GITHUB\_TOKEN Permissions

6 Secret source: Actions

7 Prepare workflow directory

8 Prepare all required actions

9 Getting action download info

10 Download action repository "actions/checkout@v3" (SHA:f43ae5ff28d294095638e18286ca9a3d1956744)

11 Download action repository "hashicorp/terraform@v1.0.0" (SHA:4c569ff81aef1b0da70939abb56bdc246748a)

12 Download action repository "actions/github-script@v6" (SHA:d7966e4a801822421a76ea35dca353c962f418)

13 Complete job name: Terraform Plan

Pull hashicorp/terraform1.0.00s

Checkout1s

Upload Configuration2s

Create Plan Run21s

Get Plan Output0s

Update PR1s

Post Checkout0s

Complete job0s

Features • GitHub Actions

Update terraform-plan.yml - Joel

github.com/josefintp/devops-engineer/actions/runs/10361771971/job/28682811904

SummaryJobsRun detailsUsageWorkflow file

Terraform Plan

succeeded 3 minutes ago in 30s

Search logs

Setup job11s

Pull hashicorp/terraform@v1.0.06s

1 Pull down action image 'hashicorp/terraform@v1.0.0'

Checkout15s

1 Run actions/checkout@v3

1 Syncing repository: josefintp/devops-engineer

2 Getting git version info

3 Temporarily overriding HOME='/home/runner/work/\_temp/3d3d084-515d-4f7f-9e5c-f978ad9917c' before making global git config changes

4 Adding repository directory to the temporary git global config as a safe directory

5 /usr/bin/git config --global --add safe.directory /home/runner/work/devops-engineer/devops-engineer

6 Deleting the contents of /home/runner/work/devops-engineer/devops-engineer

7 Initializing the repository

8 Disabling automatic garbage collection

9 Setting up auth

10 Fetching the repository

11 Determining the checkout info

12 Checking out the ref

13 /usr/bin/git log -1 --format=%H

14 7c4f143988ac62628c2130809f139a18

Upload Configuration25s

1 Run hashicorp/terraform-config@v1.0.0

1 /usr/bin/docker run --name hashicorp/terraform-config-v1.0.0 --label f6d5db --workdir /github/workspace --rm -e TF\_CLOUD\_ORGANIZATION -e TF\_API\_TOKEN -e TF\_WORKSPACE -e CONFIG\_DIRECTORY -e INPUT\_WORKSPACE -e INPUT\_DIRECTORY -e INPUT\_SPECULATIVE -e INPUT\_HOSTNAME -e INPUT\_TOKEN -e INPUT\_ORGANIZATION -e HOME -e GITHUB\_JOB -e GITHUB\_REF -e GITHUB\_SHA -e GITHUB\_REPOSITORY -e GITHUB\_REPOSITORY\_OWNER -e GITHUB\_REPOSITORY\_OWNER\_ID -e GITHUB\_RUN\_ID -e GITHUB\_RUN\_NUMBER -e GITHUB\_RETENTION\_DAYS -e GITHUB\_RUN\_ATTEMPT -e GITHUB\_REPOSITORY\_ID -e GITHUB\_ACTOR -e GITHUB\_TRIGGERING\_ACTOR -e GITHUB\_WORKFLOW -e GITHUB\_HEAD\_REF -e GITHUB\_BASE\_REF -e GITHUB\_EVENT\_NAME -e GITHUB\_SERVER\_URL -e GITHUB\_API\_URL -e GITHUB\_GRAPHQL\_URL -e GITHUB\_REF\_NAME -e GITHUB\_REF\_PROTECTED -e GITHUB\_REF\_TYPE -e GITHUB\_WORKFLOW\_REF -e GITHUB\_WORKFLOW\_SHA -e GITHUB\_WORKSPACE -e GITHUB\_ACTION -e GITHUB\_EVENT\_PATH -e GITHUB\_ACTION\_REPOSITORY -e GITHUB\_ACTION\_REF -e GITHUB\_PATH -e GITHUB\_ENV -e GITHUB\_STEP\_SUMMARY -e GITHUB\_STATE -e GITHUB\_OUTPUT -e RUNNER\_OS -e RUNNER\_ARCH -e RUNNER\_NAME -e RUNNER\_ENVIRONMENT -e RUNNER\_TOOL\_CACHE -e RUNNER\_TEMP -e RUNNER\_WORKSPACE -e ACTIONS\_RUNTIME\_URL -e ACTIONS\_RUNTIME\_TOKEN -e ACTIONS\_CACHE\_URL -e ACTIONS\_RESULTS\_URL -e GITHUB\_ACTIONS=true -e CI=true -v /var/run/docker.sock:/var/run/docker.sock -v /home/runner/work/\_temp/github/home:/github/home -v /home/runner/work/\_temp/github/workflow:/github/workflow -v /home/runner/work/\_temp/runner\_file\_commands:/github/file\_commands -v /home/runner/work/devops-engineer/devops-engineer:/github/workspace hashicorp/terraform-config-v1.0.0 --tfcli -hostname -token -organization -upload -workspace-terraform-github-actions -directory-desafios/terraform -speculative=true

Configuration version has been created: cv-916v98p97m2m

Uploading configuration...

Upload status: 'pending'

Upload status: 'uploaded'

16 {

17 'configuration\_version\_id': 'cv-916v98p97m2m',

18 'configuration\_version\_status': 'uploaded',

19 'status': 'Success'

20 }

Create Plan Run21s

Features • GitHub Actions

Update terraform-plan.yml - Joel

github.com/josefintp/devops-engineer/actions/runs/10361771971/job/28682811904

SummaryJobsRun detailsUsageWorkflow file

Terraform Plan

succeeded 4 minutes ago in 30s

Search logs

Checkout15s

Upload Configuration25s

Create Plan Run21s

1 Run hashicorp/terraform-config@v1.0.0

1 /usr/bin/docker run --name hashicorp/terraform-config-v1.0.0 --label f6d5db --workdir /github/workspace --rm -e TF\_CLOUD\_ORGANIZATION -e TF\_API\_TOKEN -e TF\_WORKSPACE -e CONFIG\_DIRECTORY -e INPUT\_WORKSPACE -e INPUT\_CONFIGURATION\_VERSION -e INPUT\_PLAN\_ONLY -e INPUT\_HOSTNAME -e INPUT\_TOKEN -e INPUT\_ORGANIZATION -e INPUT\_MESSAGE -e HOME -e GITHUB\_JOB -e GITHUB\_REF -e GITHUB\_SHA -e GITHUB\_REPOSITORY -e GITHUB\_REPOSITORY\_OWNER -e GITHUB\_REPOSITORY\_OWNER\_ID -e GITHUB\_RUN\_ID -e GITHUB\_RUN\_NUMBER -e GITHUB\_RETENTION\_DAYS -e GITHUB\_RUN\_ATTEMPT -e GITHUB\_REPOSITORY\_ID -e GITHUB\_ACTOR -e GITHUB\_TRIGGERING\_ACTOR -e GITHUB\_WORKFLOW -e GITHUB\_HEAD\_REF -e GITHUB\_BASE\_REF -e GITHUB\_EVENT\_NAME -e GITHUB\_SERVER\_URL -e GITHUB\_API\_URL -e GITHUB\_GRAPHQL\_URL -e GITHUB\_REF\_NAME -e GITHUB\_REF\_PROTECTED -e GITHUB\_REF\_TYPE -e GITHUB\_WORKFLOW\_REF -e GITHUB\_WORKFLOW\_SHA -e GITHUB\_WORKSPACE -e GITHUB\_ACTION -e GITHUB\_EVENT\_PATH -e GITHUB\_ACTION\_REPOSITORY -e GITHUB\_ACTION\_REF -e GITHUB\_PATH -e GITHUB\_ENV -e GITHUB\_STEP\_SUMMARY -e GITHUB\_STATE -e GITHUB\_OUTPUT -e RUNNER\_OS -e RUNNER\_ARCH -e RUNNER\_NAME -e RUNNER\_ENVIRONMENT -e RUNNER\_TOOL\_CACHE -e RUNNER\_TEMP -e RUNNER\_WORKSPACE -e ACTIONS\_RUNTIME\_URL -e ACTIONS\_RUNTIME\_TOKEN -e ACTIONS\_CACHE\_URL -e ACTIONS\_RESULTS\_URL -e GITHUB\_ACTIONS=true -e CI=true -v /var/run/docker.sock:/var/run/docker.sock -v /home/runner/work/\_temp/github/home:/github/home -v /home/runner/work/\_temp/github/workflow:/github/workflow -v /home/runner/work/\_temp/runner\_file\_commands:/github/file\_commands -v /home/runner/work/devops-engineer/devops-engineer:/github/workspace hashicorp/terraform-config-v1.0.0 --tfcli -hostname -token -organization -run -create -workspace-terraform-github-actions -configuration-version-cv-916v98p97m2m -speculative=true

Created Run ID: run-50p97m2m

Run status: 'pending'

Run status: 'plan\_queued'

Run status: 'planning'

Run status: 'planning'

Run status: 'planned\_and\_finished'

----- Plan Log -----

19 Terraform v1.9.4

20 on linux\_amd64

21 Initializing plugins and modules...

22 [0level] info: [message] "Terraform 1.9.4", [module]: "terraform.ui", [timestamp]: "2024-08-13T01:04:57.488963Z", [terraform]: "1.9.4", [type]: "version", [ui]: "1.2"

23 [0level] info: [message] "data.aws\_ami.ubuntu: Refreshing...", [module]: "terraform.ui", [timestamp]: "2024-08-13T01:04:58.002480Z", [hook]: [resource]:

24 [addr]: "data.aws\_ami.ubuntu", [module]: "", [resource]: "data.aws\_ami.ubuntu", [implied\_provider]: "aws", [resource\_type]: "aws\_ami", [resource\_name]: "ubuntu", [resource\_key]: null, [action]: "read", [type]: "apply\_start"

25 [0level] info: [message] "data.aws\_ami.ubuntu: Refresh complete after 0s [id=ami-030425a3efaaad179]", [module]: "terraform.ui", [timestamp]: "2024-08-13T01:04:58.255282Z", [hook]: [resource]:

26 [addr]: "data.aws\_ami.ubuntu", [module]: "", [resource]: "data.aws\_ami.ubuntu", [implied\_provider]: "aws", [resource\_type]: "aws\_ami", [resource\_name]: "ubuntu", [resource\_key]: null, [action]: "read", [id\_key]: "id", [id\_value]: "ami-030425a3efaaad179", [elapsed\_seconds": "0", [type]: "apply\_complete"]

27 [0level] info: [message] "random\_pet.sg: Plan to create", [module]: "terraform.ui", [timestamp]: "2024-08-13T01:04:58.276713Z", [change]: [resource]:

28 [addr]: "random\_pet.sg", [module]: "", [resource]: "random\_pet.sg", [implied\_provider]: "aws", [resource\_type]: "aws\_security\_group", [resource\_name]: "sg", [resource\_key": null, [action]: "create", [type]: "planned\_change"

29 [0level] info: [message] "aws\_security\_group.web-sg: Plan to create", [module]: "terraform.ui", [timestamp]: "2024-08-13T01:04:58.276794Z", [change]: [resource]: [addr]: "aws\_security\_group.web-sg", [module]: "", [resource]: "aws\_security\_group.web-sg", [implied\_provider]: "aws", [resource\_type": "aws\_security\_group", [resource\_name": "web-sg", [resource\_key": null, [action]: "create", [type]: "planned\_change"

30 [0level] info: [message] "aws\_instance.web: Plan to create", [module]: "terraform.ui", [timestamp]: "2024-08-13T01:04:58.276812Z", [change]: [resource]:

31 [addr]: "aws\_instance.web", [module]: "", [resource]: "aws\_instance.web", [implied\_provider": "aws", [resource\_type": "aws\_instance", [resource\_name": "web", [resource\_key": null, [action": "create", [type": "planned\_change"

32 [0level] info: [message] "Plan: 3 to add, 0 to change, 0 to destroy", [module]: "terraform.ui", [timestamp]: "2024-08-13T01:04:58.276872Z", [changes]: [addr]: "3", [change": "0", [import": "0", [remove": "0", [operation": "plan", [type": "change\_summary"

33 [0level] info: [message] "Outputs: 1", [module]: "terraform.ui", [timestamp]: "2024-08-13T01:04:58.276926Z", [outputs]: [web\_address": [sensitive": false, [action": "create", [type": "outputs"

34 View Run in Terraform Cloud: <https://app.terraform.io/app/josefintp/workspaces/terraform-github-actions/runs/run-50p97m2m>

35 {

36 'configuration\_version\_id': 'cv-916v98p97m2m',

37 'plan\_id': 'plan-80q9m2cc0cgr3',

38 'plan\_status': 'finished',

39 'run\_id': 'run-50p97m2m',

Features • GitHub Actions

Update terraform-plan.yml - J...

github.com/joselintp/devops-engineer/actions/runs/10361771971/job/28682811904

Summary

Jobs

Terraform Plan

Run details

Usage

Workflow file

Terraform Plan

succeeded 5 minutes ago in 30s

Search logs

Create Plan Run

21s

Get Plan Output

0s

Update PR

1s

Post Checkout

0s

Complete job

0s

1 ▶ Run hashicorp/terraform-provider-aws@v1.0.0

9 /usr/bin/docker run --name hashicorp/terraform-provider-aws\_770461 --label f6dcb /github/workspace --rm -e "TF\_CLOUD\_ORGANIZATION" -e "TF\_API\_TOKEN" -e "TF\_WORKSPACE" -e "CONFIG\_DIRECTORY" -e "INPUT\_PLAN" -e "INPUT\_HOSTNAME" -e "INPUT\_TOKEN" -e "INPUT\_ORGANIZATION" -e "HOME" -e "GITHUB\_JOB" -e "GITHUB\_REF" -e "GITHUB\_SHA" -e "GITHUB\_REPOSITORY" -e "GITHUB\_REPOSITORY\_OWNER" -e "GITHUB\_RUN\_ID" -e "GITHUB\_RUN\_NUMBER" -e "GITHUB\_RETENTION\_DAYS" -e "GITHUB\_RUN\_ATTEMPT" -e "GITHUB\_REPOSITORY\_ID" -e "GITHUB\_ACTOR\_ID" -e "GITHUB\_ACTOR" -e "GITHUB\_TRIGGERING\_ACTOR" -e "GITHUB\_WORKFLOW" -e "GITHUB\_HEAD\_REF" -e "GITHUB\_BASE\_REF" -e "GITHUB\_EVENT\_NAME" -e "GITHUB\_SERVER\_URL" -e "GITHUB\_API\_URL" -e "GITHUB\_GRAPHQL\_URL" -e "GITHUB\_REF\_NAME" -e "GITHUB\_REF\_PROTECTED" -e "GITHUB\_REF\_TYPE" -e "GITHUB\_WORKFLOW\_REF" -e "GITHUB\_WORKFLOW\_SHA" -e "GITHUB\_WORKSPACE" -e "GITHUB\_ACTION" -e "GITHUB\_EVENT\_PATH" -e "GITHUB\_ACTION\_REPOSITORY" -e "GITHUB\_ACTION\_REF" -e "GITHUB\_PATH" -e "GITHUB\_ENV" -e "GITHUB\_STEP\_SUMMARY" -e "GITHUB\_STATE" -e "GITHUB\_OUTPUT" -e "RUNNER\_OS" -e "RUNNER\_ARCH" -e "RUNNER\_NAME" -e "RUNNER\_ENVIRONMENT" -e "RUNNER\_TOOL\_CACHE" -e "RUNNER\_TEMP" -e "RUNNER\_WORKSPACE" -e "ACTIONS\_RUNTIME\_URL" -e "ACTIONS\_RUNTIME\_TOKEN" -e "ACTIONS\_CACHE\_URL" -e "ACTIONS\_RESULTS\_URL" -e GITHUB\_ACTIONS=true -e Ci=true -v "/var/run/docker.sock":"/var/run/docker.sock" -v "/home/runner/work/\_temp/github\_home":"/github/home" -v "/home/runner/work/\_temp/github\_workflow":"/github/workflow" -v "/home/runner/work/\_temp/runner\_file\_commands":"/github/file\_commands" -v "/home/runner/work/devops-engineer/devops-engineer":"/github/workspace" hashicorp/terraform-provider-aws:1.0.0 "tfcl" "-hostname" "-token" "-organization" "plan" "output" "-planplan-80xqwdCCx0q3"

10 {

11 "add": "j",

12 "change": "0",

13 "destroy": "0",

14 "plan\_id": "plan-80xqwdCCx0q3",

15 "plan\_status": "finished",

16 "status": "success"

17 }

1 ▶ Run actions/github-script@v6

1 Post job cleanup.

2 /usr/bin/git version

3 git version 2.46.0

4 Temporarily overriding HOME="/home/runner/work/\_temp/632a6b7-c8d4-40us-b3d3-2d78usczas80" before making global git config changes

5 Adding repository directory to the temporary git global config as a safe directory

6 /usr/bin/git config --global --add safe.directory /home/runner/work/devops-engineer/devops-engineer

7 /usr/bin/git config --local --name-only --get-regexp core.sshCommand

8 /usr/bin/git submodule foreach --recursive sh -c "git config --local --name-only --get-regexp 'core.sshCommand' && git config --local --unset-all 'core.sshCommand' || :"

9 /usr/bin/git config --local --name-only --get-regexp http.https://github.com/V/.extraheder

10 http.https://github.com/V/.extraheder

11 /usr/bin/git config --local --unset-all http.https://github.com/V/.extraheder

12 /usr/bin/git submodule foreach --recursive sh -c "git config --local --name-only --get-regexp 'http.https://github.com/V/.extraheder' && git config --local --unset-all 'http.https://github.com/V/.extraheder' || :"

1 Cleaning up orphan processes

Features • GitHub Actions

Update terraform-plan.yml - J...

github.com/joselintp/devops-engineer/actions/runs/10361771971/job/28682811904

Joselintp / devops-engineer

Code Issues Pull requests 1 Actions Projects Security Insights Settings

Terraform Plan

Update terraform-plan.yml #5

Re-run all jobs Latest #2 ...

Summary

Jobs

Terraform Plan

Run details

Usage

Workflow file

Terraform Plan

succeeded 2 minutes ago in 30s

Search logs

Set up job

1s

Pull hashicorp/terraform-provider-aws@v1.0.0

0s

Checkout

0s

Upload Configuration

2s

Create Plan Run

21s

Get Plan Output

0s

Update PR

1s

Post Checkout

0s

Complete job

0s





Features • GitHub Actions

Update terraform-plan.yml · Jo · +

github.com/joselintp/devops-engineer/actions/runs/10361771971/job/26682811904

Summary

Jobs

Terraform Plan

Run details

Usage

Workflow file

Terraform Plan

succeeded 4 minutes ago in 30s

Search logs

Checkout

25s

Upload Configuration

25s

Create Plan Run

21s

1 ▶ Run hashicorp/terraform-github-actions/create-run@v1.0.0

11 /usr/bin/docker run --name hashicorp/terraform-github-actions-10361771971-job-26682811904 --workdir /github/workspace --rm -e "TF\_CLOUD\_ORGANIZATION" -e "TF\_API\_TOKEN" -e "TF\_WORKSPACE" -e "CONFIG\_DIRECTORY" -e "INPUT\_WORKSPACE" -e "INPUT\_CONFIGURATION\_VERSION" -e "INPUT\_PLAN\_ONLY" -e "INPUT\_HOSTNAME" -e "INPUT\_TOKEN" -e "INPUT\_ORGANIZATION" -e "INPUT\_MESSAGE" -e "HOME" -e "GITHUB\_JOB" -e "GITHUB\_REF" -e "GITHUB\_SHA" -e "GITHUB\_REPOSITORY" -e "GITHUB\_REPOSITORY\_OWNER" -e "GITHUB\_REPOSITORY\_OWNER\_ID" -e "GITHUB\_RUN\_ID" -e "GITHUB\_RUN\_NUMBER" -e "GITHUB\_RETENTION\_DAYS" -e "GITHUB\_ATTEMPT" -e "GITHUB\_REPOSITORY\_ID" -e "GITHUB\_ACTOR\_ID" -e "GITHUB\_ACTOR" -e "GITHUB\_TRIGGERING\_ACTOR" -e "GITHUB\_WORKFLOW" -e "GITHUB\_HEAD\_REF" -e "GITHUB\_BASE\_REF" -e "GITHUB\_EVENT\_NAME" -e "GITHUB\_SERVER\_URL" -e "GITHUB\_API\_URL" -e "GITHUB\_GRAPHQL\_URL" -e "GITHUB\_REF\_NAME" -e "GITHUB\_REF\_PROTECTED" -e "GITHUB\_REF\_TYPE" -e "GITHUB\_WORKFLOW\_REF" -e "GITHUB\_WORKFLOW\_SHA" -e "GITHUB\_WORKSPACE" -e "GITHUB\_ACTION" -e "GITHUB\_EVENT\_PATH" -e "GITHUB\_ACTION\_REPOSITORY" -e "GITHUB\_ACTION\_REF" -e "GITHUB\_PATH" -e "GITHUB\_ENV" -e "GITHUB\_STEP\_SUMMARY" -e "GITHUB\_STATE" -e "GITHUB\_OUTPUT" -e "RUNNER\_OS" -e "RUNNER\_ARCH" -e "RUNNER\_NAME" -e "RUNNER\_ENVIRONMENT" -e "RUNNER\_TOOL\_CACHE" -e "RUNNER\_TEMP" -e "RUNNER\_WORKSPACE" -e "ACTIONS\_RUNTIME\_URL" -e "ACTIONS\_RUNTIME\_TOKEN" -e "ACTIONS\_CACHE\_URL" -e "ACTIONS\_RESULTS\_URL" -e "GITHUB\_ACTIONS=true" -e "CI=true" -v "/var/run/docker.sock":"/var/run/docker.sock" -v "/home/runner/work/\_temp/\_github\_home":"/github/home" -v "/home/runner/work/\_temp/\_github\_workflow":"/github/workflow" -v "/github/file\_commands":"/github/file\_commands" -v "/home/runner/work/devops-engineer/devops-engineer":"/github/workspace" hashicorp/terraform-github-actions:1.0.0 "tfci" "-hostname" "-token" "-organization" "run" "create" "-workspace-terraform-github-actions" "-configuration-version-cv-916v78sp97eztr" "-message" "-plan-only=true"

12 Created Run ID: run-SqPspDzisywmyq

13 Run status: 'pending'

14 Run status: 'plan\_queued'

15 Run status: 'planning'

16 Run status: 'planning'

17 Run status: 'planned\_and\_finished'

18

19

20 ----- Plan Log -----

21 Terraform v1.9.4

22 on linux\_amd64

23 Initializing plugins and modules...

24 ["@level":"info","@message":"terraform.ui","@timestamp":"2024-08-13T01:03:57.488065Z","terraform":"1.9.4","type":"version","ui":"1.2"]

25 ["@level":"info","@message":"data.aws\_ami.ubuntu: refreshing...","@module":"terraform.ui","@timestamp":"2024-08-13T01:04:00.255262Z","hook":{"resource":{"addr":"data.aws\_ami.ubuntu","module":"","resource":{"aws","resource\_type":"aws\_ami","resource\_name":"ubuntu","resource\_key":null,"action":"read"},"type":"apply\_start"},"@level":"info","@message":"data.aws\_ami.ubuntu: refresh complete after 0s [id=ami-03b425a3faad1791],"@module":"terraform.ui","@timestamp":"2024-08-13T01:04:00.255262Z","hook":{"resource":{"addr":"data.aws\_ami.ubuntu","module":"","resource":{"aws","resource\_type":"aws\_ami","resource\_name":"ubuntu","resource\_key":null,"action":"read","id\_key":"id","id\_value":"ami-03b425a3faad1791","elapsed\_seconds":0},"type":"apply\_complete"},"@level":"info","@message":"random\_pet.sg: Plan to create","@module":"terraform.ui","@timestamp":"2024-08-13T01:04:00.276732Z","change":{"resource":{"addr":"random\_pet.sg","module":"","resource":{"random\_pet.sg","implied\_provider":"aws","resource\_type":"random\_pet","resource\_name":"sg","resource\_key":null,"action":"create"},"type":"planned\_change"},"@level":"info","@message":"aws\_security\_group.web.sg: Plan to create","@module":"terraform.ui","@timestamp":"2024-08-13T01:04:00.276794Z","change":{"resource":{"addr":"aws\_security\_group.web.sg","module":"","resource":{"aws\_security\_group.web.sg","implied\_provider":"aws","resource\_type":"aws\_security\_group","resource\_name":"web.sg","resource\_key":null,"action":"create"},"type":"planned\_change"},"@level":"info","@message":"aws\_instance.web: Plan to create","@module":"terraform.ui","@timestamp":"2024-08-13T01:04:00.276836Z","change":{"resource":{"addr":"aws\_instance.web","module":"","resource":{"aws\_instance.web","implied\_provider":"aws","resource\_type":"aws\_instance","resource\_name":"web","resource\_key":null,"action":"create"},"type":"planned\_change"},"@level":"info","@message":"Plan: 3 to add, 0 to change, 0 to destroy","@module":"terraform.ui","@timestamp":"2024-08-13T01:04:00.276922Z","changes":{"addr":"3","change":"0","import":"0","remove":"0","operation":"plan"},"type":"change\_summary"},"@level":"info","@message":"Outputs: 1","@module":"terraform.ui","@timestamp":"2024-08-13T01:04:00.276922Z","outputs":{"web-address":{"sensitive":false,"action":"create"},"type":"outputs"}]

26

27

28

29

30

31

32 View Run in Terraform Cloud: <https://app.terraform.io/app/3ops101tf/workspaces/terraform-github-actions/runs/run-SqPspDzisywmyq>

33 {

34 "configuration\_version\_id": "cv-916v78sp97eztr",

35 "plan\_id": "plan-BqfQmmtCxcQcgr",

36 "plan\_status": "finished",

37 "run\_id": "run-SqPspDzisywmyq",

Features • GitHub Actions

Update terraform-plan.yml · Jo · +

github.com/joselintp/devops-engineer/actions/runs/10361771971/job/26682811904

Summary

Jobs

Terraform Plan

Run details

Usage

Workflow file

Terraform Plan

succeeded 5 minutes ago in 30s

Search logs

Create Plan Run

21s

Get Plan Output

0s

1 ▶ Run hashicorp/terraform-github-actions/plan-output@v1.0.0

9 /usr/bin/docker run --name hashicorp/terraform-github-actions-10361771971-job-26682811904 --workdir /github/workspace --rm -e "TF\_CLOUD\_ORGANIZATION" -e "TF\_API\_TOKEN" -e "TF\_WORKSPACE" -e "CONFIG\_DIRECTORY" -e "INPUT\_PLAN" -e "INPUT\_HOSTNAME" -e "INPUT\_TOKEN" -e "INPUT\_ORGANIZATION" -e "HOME" -e "GITHUB\_JOB" -e "GITHUB\_REF" -e "GITHUB\_SHA" -e "GITHUB\_REPOSITORY" -e "GITHUB\_REPOSITORY\_OWNER" -e "GITHUB\_REPOSITORY\_OWNER\_ID" -e "GITHUB\_RUN\_ID" -e "GITHUB\_RUN\_NUMBER" -e "GITHUB\_RETENTION\_DAYS" -e "GITHUB\_ATTEMPT" -e "GITHUB\_REPOSITORY\_ID" -e "GITHUB\_ACTOR\_ID" -e "GITHUB\_ACTOR" -e "GITHUB\_TRIGGERING\_ACTOR" -e "GITHUB\_WORKFLOW" -e "GITHUB\_HEAD\_REF" -e "GITHUB\_BASE\_REF" -e "GITHUB\_SERVER\_URL" -e "GITHUB\_API\_URL" -e "GITHUB\_GRAPHQL\_URL" -e "GITHUB\_REF\_NAME" -e "GITHUB\_REF\_PROTECTED" -e "GITHUB\_REF\_TYPE" -e "GITHUB\_WORKFLOW\_REF" -e "GITHUB\_WORKFLOW\_SHA" -e "GITHUB\_WORKSPACE" -e "GITHUB\_ACTION" -e "GITHUB\_EVENT\_PATH" -e "GITHUB\_ACTION\_REPOSITORY" -e "GITHUB\_ACTION\_REF" -e "GITHUB\_PATH" -e "GITHUB\_ENV" -e "GITHUB\_STEP\_SUMMARY" -e "GITHUB\_STATE" -e "GITHUB\_OUTPUT" -e "RUNNER\_OS" -e "RUNNER\_ARCH" -e "RUNNER\_NAME" -e "RUNNER\_ENVIRONMENT" -e "RUNNER\_TOOL\_CACHE" -e "RUNNER\_TEMP" -e "RUNNER\_WORKSPACE" -e "ACTIONS\_RUNTIME\_URL" -e "ACTIONS\_RUNTIME\_TOKEN" -e "ACTIONS\_CACHE\_URL" -e "ACTIONS\_RESULTS\_URL" -e "GITHUB\_ACTIONS=true" -e "CI=true" -v "/var/run/docker.sock":"/var/run/docker.sock" -v "/home/runner/work/\_temp/\_github\_home":"/github/home" -v "/home/runner/work/\_temp/\_github\_workflow":"/github/workflow" -v "/home/runner/work/\_temp/\_runner\_file\_commands":"/github/file\_commands" -v "/home/runner/work/devops-engineer/devops-engineer":"/github/workspace" hashicorp/terraform-github-actions:1.0.0 "tfci" "-hostname" "-token" "-organization" "plan" "output" "-plan-plan-BqfQmmtCxcQcgr"

10 {

11 "addr": "3",

12 "change": "0",

13 "destroy": "0",

14 "plan\_id": "plan-BqfQmmtCxcQcgr",

15 "plan\_status": "finished",

16 "status": "success"

17 }

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

Update PR

1s

Post Checkout

0s

1 Post job cleanup.

2 /usr/bin/git version

3 git version 2.46.0

4 Temporarily overriding HOME=/home/runner/work/\_temp/632a6b27-ca84-48a8-b1d5-2d785623a899 before making global git config changes

5 Adding repository directory to the temporary git global config as a safe directory

6 /usr/bin/git config --global --add safe.directory /home/runner/work/devops-engineer/devops-engineer

7 /usr/bin/git config --local --name-only --get-regexp core.sshCommand

8 /usr/bin/git submodule foreach --recursive sh -c 'git config --local --name-only --get-regexp 'core.sshCommand' && git config --local --unset-all 'core.sshCommand' || :'

9 /usr/bin/git config --local --name-only --get-regexp http.https://github.com/.extrahader

10 http.https://github.com/.extrahader

11 /usr/bin/git config --local --unset-all http.https://github.com/.extrahader

12 /usr/bin/git submodule foreach --recursive sh -c 'git config --local --name-only --get-regexp 'http.https://github.com/.extrahader' && git config --local --unset-all 'http.https://github.com/.extrahader' || :'

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

Complete job

0s

1 Cleaning up orphan processes



Creación grupo de seguridad / instancia EC2 en AWS

Panel de EC2

Vista global de EC2

Eventos

Console-to-Code

Vista previa

Instancias

Imágenes

AMI

Catálogo de AMI

Elastic Block Store

Volumenes

Instantáneas

Administrador del ciclo de vida

Red y seguridad

Security Groups

Direcciones IP elásticas

Grupos de ubicación

Pares de claves

Interfaces de red

Equilibrio de carga

Balancedadores de carga

Grupos de destino

Trust Stores

Nuevo

Auto Scaling

Grupos de Auto Scaling

Configuración

Instancias (1/2)

Información

Conectar

Estado de la instancia

Acciones

Lanzar instancias

Buscar instancia por atributo o etiqueta (case-sensitive)

Todos los e...

Name	ID de la instancia	Estado de la i...	Tipo de ...	Comprobación de	Estado de la al	Zona de ...	DNS de ...	Dirección IP...	IP elástica	Direccio...	Monitor...	Nombre del grupo de s
Bastion Host	i-00404c7442686b3ce	En ejecución	t2.micro	2/2 comprobador	Ver alarmas +	us-east-1a	ec2-3-237-...	3.237.174.78	-	-	disabled	Ec2SecurityGroup
	i-02cd77f5d492cddddd	En ejecución	t2.micro	2/2 comprobador	Ver alarmas +	us-east-1b	ec2-52-90-...	52.90.200.49	-	-	disabled	stirred-dinosaur-sg

i-02cd77f5d492cddddd

Detalles

Estado y alarmas

Monitoreo

Seguridad

Redes

Almacenamiento

Etiquetas

Resumen de instancia

Información

ID de la instancia

i-02cd77f5d492cddddd

Dirección IPv4 pública

52.90.200.49 | dirección abierta

Estado de la instancia

En ejecución

Nombre DNS de IP privada (solo IPv4)

ip-172-31-81-177.ec2.internal

Tipo de instancia

t2.micro

ID de VPC

vpc-0d1aa4f8e4000d5e8

ID de subred

subnet-05d9686d76fb3e436

ARN de instancia

IMDSv2

Direcciones IPv4 privadas

172.31.81.177

DNS de IPv4 pública

ec2-52-90-200-49.compute-1.amazonaws.com | dirección abierta

Direcciones IP elásticas

-

Hallazgo de AWS Compute Optimizer

Suscribirse a AWS Compute Optimizer para recibir recomendaciones. | Más información

Nombre del grupo de Auto Scaling

-

Panel de EC2

Vista global de EC2

Eventos

Console-to-Code

Vista previa

Instancias

Imágenes

AMI

Catálogo de AMI

Elastic Block Store

Volumenes

Instantáneas

Administrador del ciclo de vida

Red y seguridad

Security Groups

Direcciones IP elásticas

Grupos de ubicación

Pares de claves

Interfaces de red

Equilibrio de carga

Balancedadores de carga

Grupos de destino

Trust Stores

Nuevo

Auto Scaling

Grupos de Auto Scaling

Configuración

Instancias (1/2)

Información

Conectar

Estado de la instancia

Acciones

Lanzar instancias

Buscar instancia por atributo o etiqueta (case-sensitive)

Todos los e...

Name	ID de la instancia	Estado de la i...	Tipo de ...	Comprobación de	Estado de la al	Zona de ...	DNS de ...	Dirección IP...	IP elástica	Direccio...	Monitor...	Nombre del grupo de s
Bastion Host	i-00404c7442686b3ce	En ejecución	t2.micro	2/2 comprobador	Ver alarmas +	us-east-1a	ec2-3-237-...	3.237.174.78	-	-	disabled	Ec2SecurityGroup
	i-02cd77f5d492cddddd	En ejecución	t2.micro	2/2 comprobador	Ver alarmas +	us-east-1b	ec2-52-90-...	52.90.200.49	-	-	disabled	stirred-dinosaur-sg

i-02cd77f5d492cddddd

Detalles

Estado y alarmas

Monitoreo

Seguridad

Redes

Almacenamiento

Etiquetas

Detalles de seguridad

Rol de IAM

-

ID del propietario

492265689622

Grupos de seguridad

sg-05dea16df8b7ba2b5 (stirred-dinosaur-sg)

Reglas de entrada

Filtrar reglas

Nombre

ID de la regla del grupo d...

Intervalo de pu...

Protocolo

Origen

Grupos de seguridad

Descripción

-

sg-0b49ae382c6ca3189

8080

TCP

0.0.0.0/0

stirred-dinosaur-sg

Reglas de salida

Filtrar reglas