Laboratorio

PRACTICA

TERRAFORM

CONTROL DE VERSIONES

Elaborado por:	Jonatan Stiven Gutierrez	**No. de Versión:**	1.0.0
Revisado por:		**Fecha de revisión:**	
Aprobado por:		**Fecha de Aprobación:**	

Historia de Modificaciones

No. de Versión	Fecha de Versión	Autor	Revisado por	Aprobado por	Descripción
1.0.0	21/02/2024	Jonatan Stiven Gutierrez			Documento Original

Lista de distribución

Para	Acción*	Empresa	Firma/Medio de Entrega

** Tipos de acción: Aprobar, Revisar, Informar, Archivar, Complementar, Asistir a junta, Otras (por favor especificar)*

Contenido

INTRODUCCION

El siguiente documento proporciona una introducción detallada a los ejercicios realizados en el laboratorio.

PRERREQUISITOS

- Tener entendimiento de los temas vistos anteriormente.

EJERCICIO 1:

En este ejercicio usaremos el Provider "local_file".

1. Estando en consola creamos un directorio.
   - mkdir ejercicio1

2. Ejecutamos el siguiente comando
   - cd ejercicio1

3. Ejecutamos el siguiente comando
   - code .
     Este comando "code ." nos lleva directamente a Visual Studio Code.

4. Estando en visual Studio Code, verificamos que estamos en la ruta previamente conocida.

5. Creamos un archivo

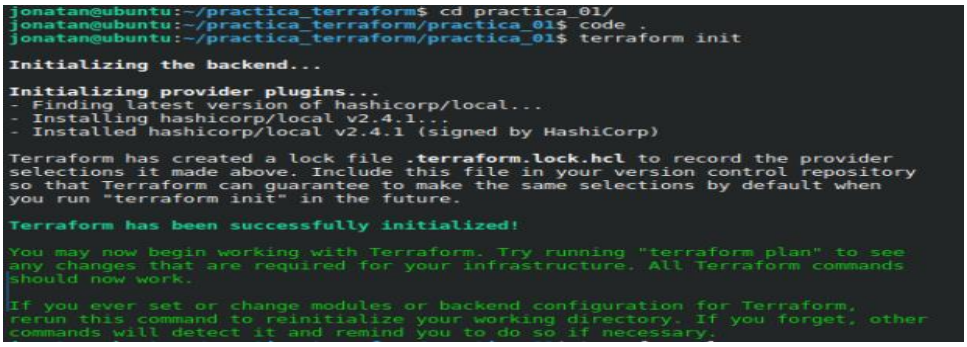
   - Terraform.tf
     Dentro de él, copiamos el siguiente código.

```
resource "local_file" "cali" {
    content = "Hola, grupo devops"
    filename = "seti.txt"
}
```

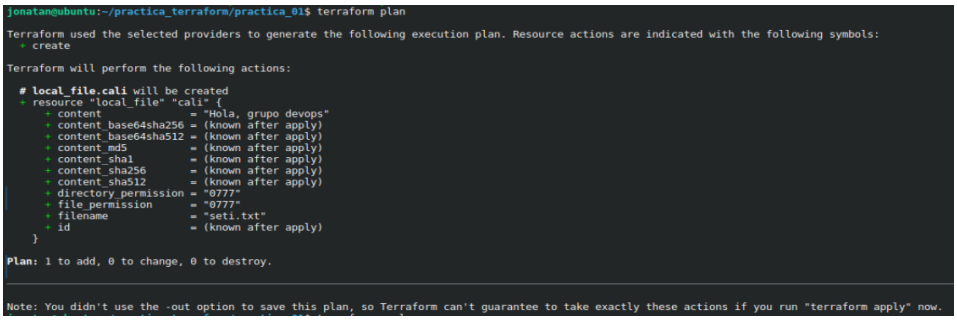
6. Volvemos a la consola.

7. Ejecutamos el siguiente comando

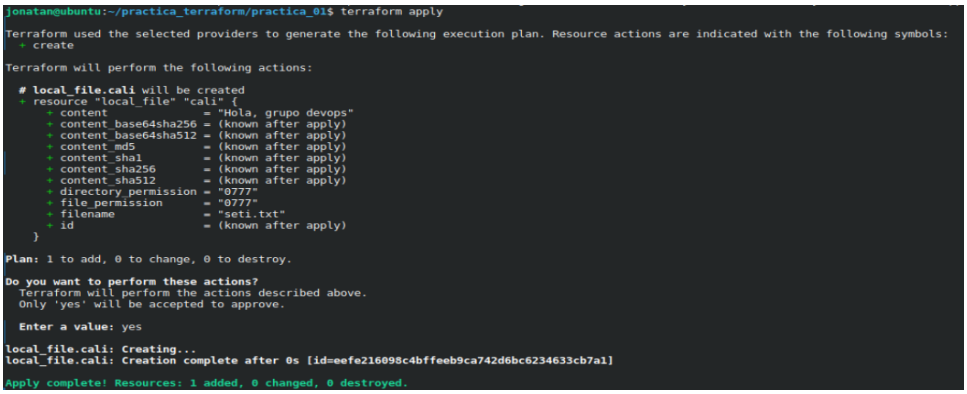
   - terraform init



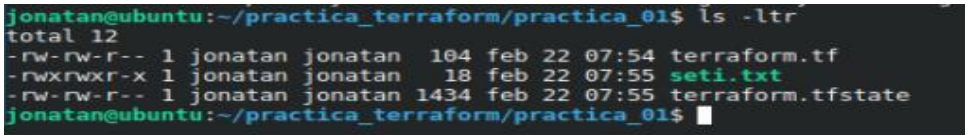
8. Ejecutamos el siguiente comando
    - terraform plan



9. Ejecutamos el siguiente comando
    - terraform apply
    - En "Enter a value" escribe: **yes**
    - Presiona enter



10. Ejecutamos el siguiente comando para ver los archivos del directorio
    - ls-ltr

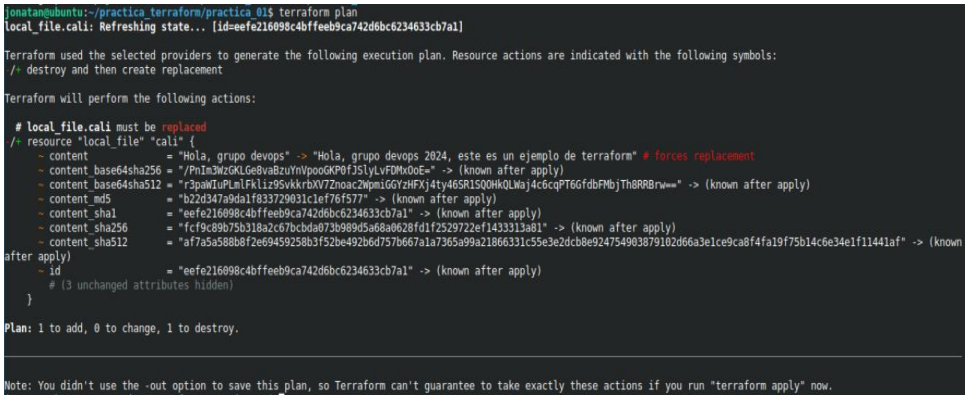


11. Ejecutamos el siguiente comando para ver el contenido del archivo .txt
    - cat seti.txt

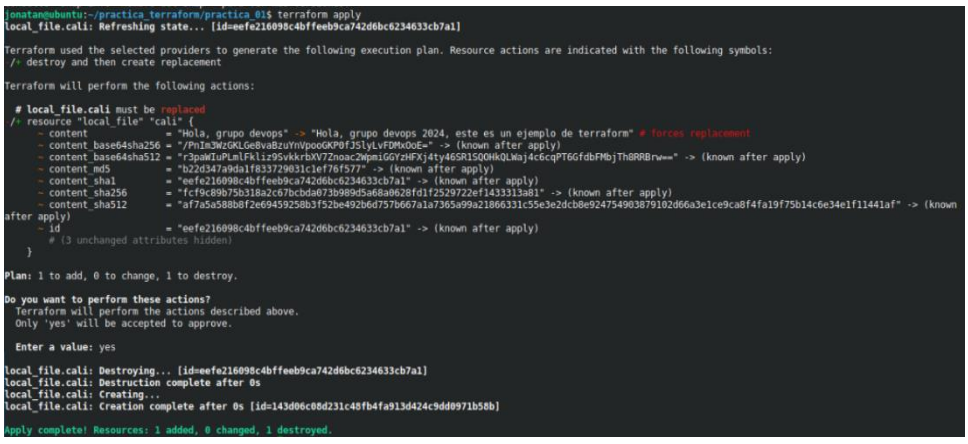
12. Ahora vamos a editar el archivo seti.txt
    - Volvemos al Visual Studio Code
    - Modificamos el archivo terraform.tf

```
resource "local_file" "cali" {
    content = "Hola, grupo devops 2024, este es un ejemplo de terraform"
    filename = "seti.txt"
}
```

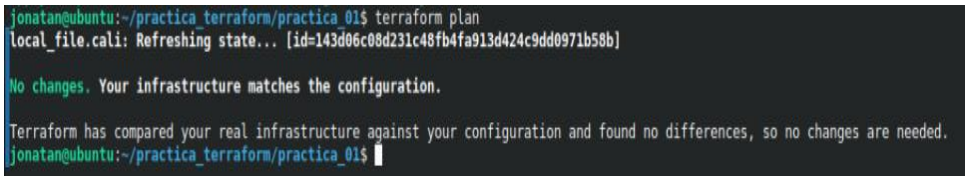
13. Una vez editado el archivo, vamos a actualizar el estado de archivo seti.txt.
    - Volvemos a la consola.
    - Ya no ejecutaríamos primero el comando terraform init, si no "terraform plan".
    - Esto nos muestra que hubo un cambio en el contenido, pero también nos dice que el archivo fue remplazado (porque no puede editar el archivo), entonces el destruye el archivo anterior y agrega el archivo de nuevo.



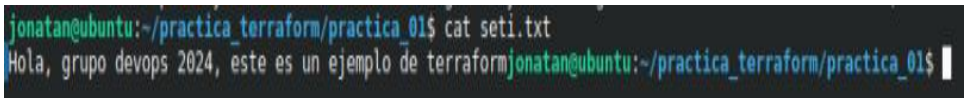
14. Ejecutamos el siguiente comando para aplicar los cambios
    - terraform apply



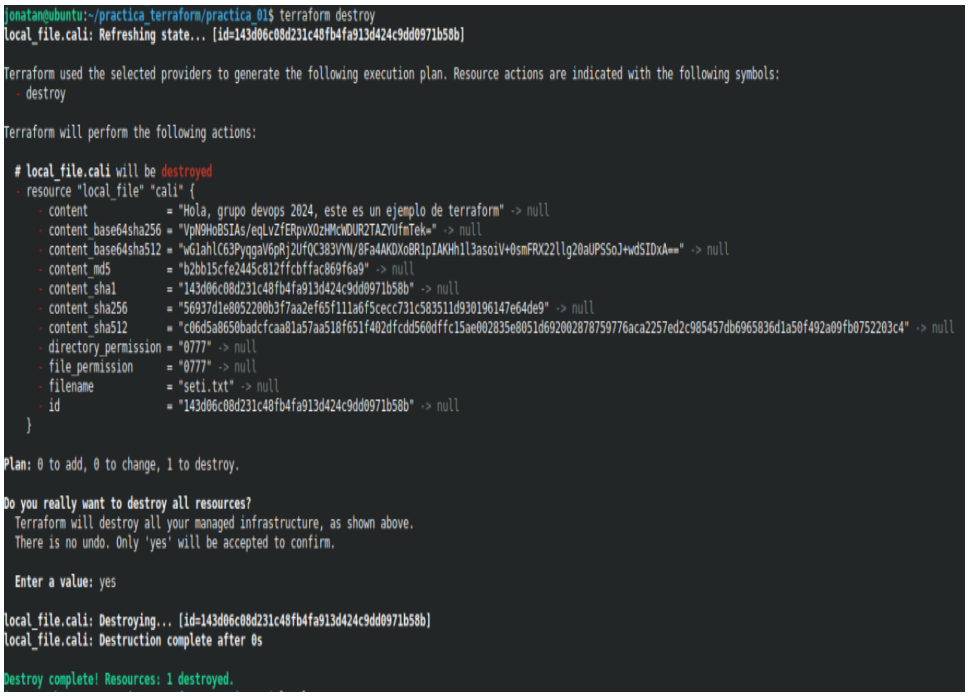
15. Ejecutamos el siguiente comando para validar si hay cambios pendientes por aplicar.
    - terraform plan
    - Como vemos no hay cambios pendientes.



16. Ejecutamos el siguiente comando para ver los cambios en el archivo.
    - cat seti.txt



17. Ahora queremos eliminar la infraestructura del código
    - terraform destroy



18. Ejecutamos el siguiente comando para ver los archivos del directorio
    - ls –ltr
    - Como vemos se generó un archivo "terraform.tfstate.backup" que ya más adelante veremos su función.

```
jonatan@ubuntu:~/practica_terraform/practica_01$ ls -ltr
total 12
-rw-rw-r-- 1 jonatan jonatan  142 feb 22 09:30 terraform.tf
-rw-rw-r-- 1 jonatan jonatan 1472 feb 22 10:21 terraform.tfstate.backup
-rw-rw-r-- 1 jonatan jonatan  180 feb 22 10:21 terraform.tfstate
jonatan@ubuntu:~/practica_terraform/practica_01$
```