Lesson 10 Demo 07

Managing Sensitive Data in Terraform State

Objective: To manage sensitive data within Terraform state files by viewing them in raw format and suppressing sensitive information

Tools required: Visual Studio Code

Prerequisites: Ensure you have created and implemented the AWS access key and secret key before starting this demo. Refer to Lesson 08 Assisted Practice 02 for detailed steps

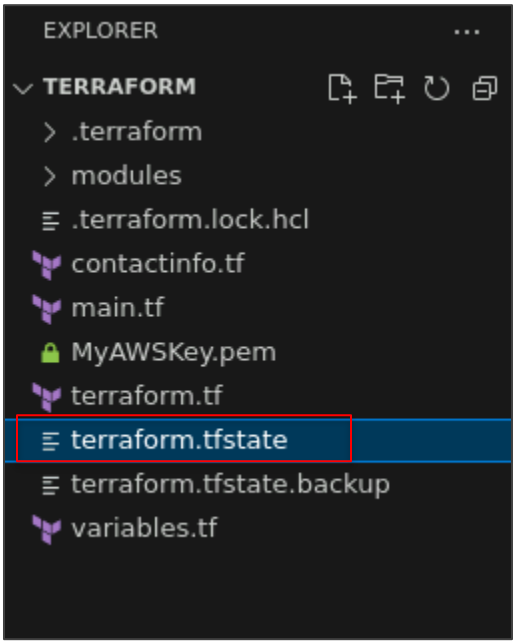
Note: The folder structure created in the previous demos is used here. It is also included in the resources section of LMS. Please refer Lesson_10_demo_01

Steps to be followed:

1. View Terraform state in raw format
2. Suppress sensitive information
3. View the Terraform state file

Step 1: View Terraform state in raw format

1.1 Open the **terraform.tfstate** file in your working directory

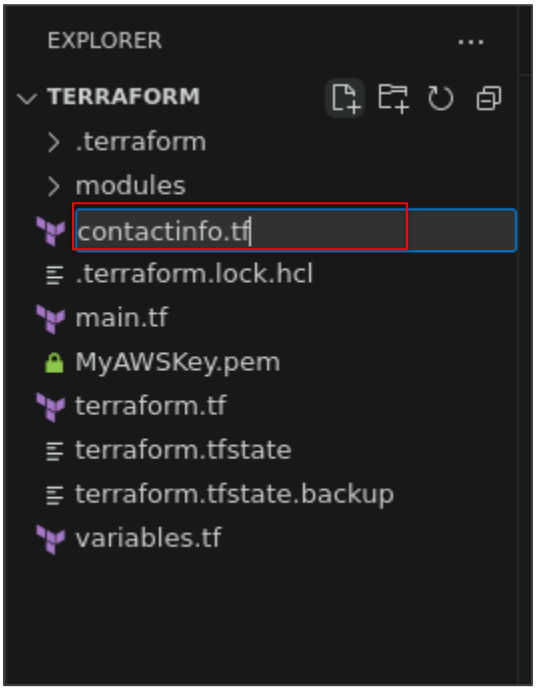


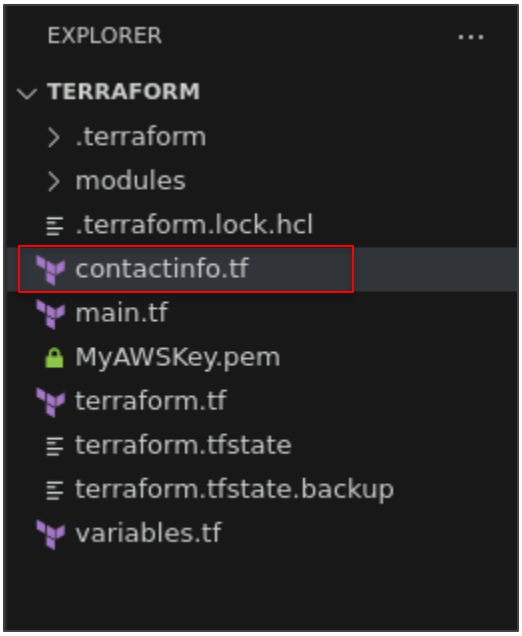
terraform.tfstate

```
{
  "version": 4,
  "terraform_version": "1.1.6",
  "serial": 1,
  "lineage": "40adafd2-dc49-e971-e06d-64f9e72d9dcc",
  "outputs": {
    "public_dns": {
      "value": "ec2-3-237-172-137.compute-1.amazonaws.com",
      "type": "string"
    },
    "public_dns_server_subnet_1": {
      "value": "ec2-34-233-121-225.compute-1.amazonaws.com",
      "type": "string"
    },
    "public_ip": {
```

Step 2: Suppress sensitive information

2.1 Create a new file **contactinfo.tf** in your working directory:





2.2 Add the following code in **contactinfo.tf** with sensitive variables:

```
variable "first_name" {
  type      = string
  sensitive = true
  default   = "Terraform"
}
variable "last_name" {
  type      = string
  sensitive = true
  default   = "Tom"
}
variable "phone_number" {
  type      = string
  sensitive = true
  default   = "867-5309"
}
locals {
  contact_info = {
    first_name   = var.first_name
    last_name    = var.last_name
    phone_number = var.phone_number
  }
  my_number = nonsensitive(var.phone_number)
}
output "first_name" {
  value = local.contact_info.first_name
  sensitive = true
}
output "last_name" {
  value = local.contact_info.last_name
  sensitive = true
}
output "phone_number" {
  value = local.contact_info.phone_number
  sensitive = true
}
output "my_number" {
  value = local.my_number
```

sensitive = true
}

contactinfo.tf

```
variable "first_name" {
  type      = string
  sensitive = true
  default   = "Terraform"
}
variable "last_name" {
  type      = string
  sensitive = true
  default   = "Tom"
}
variable "phone_number" {
  type      = string
  sensitive = true
  default   = "867-5309"
}
```

contactinfo.tf

```
variable "phone_number" {
}
locals {
  contact_info = {
    first_name   = var.first_name
    last_name    = var.last_name
    phone_number = var.phone_number
  }
  my_number = nonsensitive(var.phone_number)
}
output "first_name" {
  value = local.contact_info.first_name
  sensitive = true
}
output "last_name" {
  value = local.contact_info.last_name
  sensitive = true
}
```

contactinfo.tf

```
output "first_name" {
  sensitive = true
}
output "last_name" {
  value = local.contact_info.last_name
  sensitive = true
}
output "phone_number" {
  value = local.contact_info.phone_number
  sensitive = true
}
output "my_number" {
  value = local.my_number
  sensitive = true
}
```

2.3 Apply the configuration changes using the following command:
terraform apply

```
sakshiguptasimp@ip-172-31-22-2:~/Desktop/Terraform$ terraform apply
random_string.random: Refreshing state... [id=R#UiFcQoc3]
tls_private_key.generated: Refreshing state... [id=f85f14cc31180e45b9a481cc2c3d72a657b4ba40]
local_file.private_key_pem: Refreshing state... [id=3f3999456177b4194909cddcd7cec7b6057eb408]
aws_key_pair.generated: Refreshing state... [id=MyAWSKey]
aws_vpc.vpc: Refreshing state... [id=vpc-0607eef6018bfde4e]
```

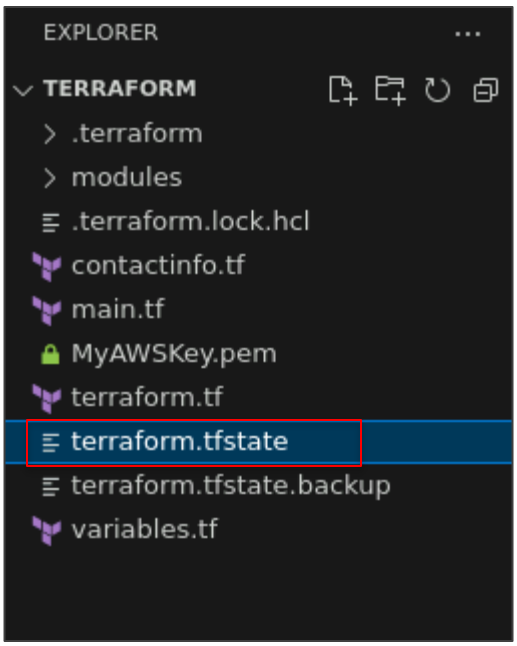
2.4 When prompted, approve the changes by typing **yes**

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS        terraform

  + my_number    = (sensitive value)
  + phone_number = (sensitive value)

You can apply this plan to save these new output values to the Terraform state, without changing any real
infrastructure.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes
```

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes


Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

first_name = <sensitive>
last_name = <sensitive>
my_number = <sensitive>
phone_number = <sensitive>
public_dns = "ec2-3-237-172-137.compute-1.amazonaws.com"
public_dns_server_subnet_1 = "ec2-34-233-121-225.compute-1.amazonaws.com"
public_ip = "3.237.172.137"
public_ip_server_subnet_1 = "34.233.121.225"
size = "t2.micro"
sakshiguptasimp@ip-172-31-22-2:~/Desktop/Terraform$
```

Step 3: View the Terraform state file

3.1 Open the updated **terraform.tfstate** file in your working directory to see the stored sensitive information



terraform.tfstate

```
{
  "version": 4,
  "terraform_version": "1.1.6",
  "serial": 2,
  "lineage": "40adafd2-dc49-e971-e06d-64f9e72d9dcc",
  "outputs": {
    "first_name": {
      "value": "Terraform",
      "type": "string",
      "sensitive": true
    },
    "last_name": {
      "value": "Tom",
      "type": "string",
      "sensitive": true
    },
```

terraform.tfstate

```
{
  "outputs": {
    },
    "last_name": {
      "value": "Tom",
      "type": "string",
      "sensitive": true
    },
    "my_number": {
      "value": "867-5309",
      "type": "string",
      "sensitive": true
    },
    "phone_number": {
      "value": "867-5309",
      "type": "string",
      "sensitive": true
    },
```

By following these steps, you have successfully managed sensitive data within Terraform state files by viewing them in raw format and suppressing sensitive information.