**Lab Manual- Backup Exiting VM with Terraform**

**Prepared for**:

**Date:** 18th Nov 2024

**Prepared by:** Bipin Sinhaa

Document Name: Lab Manual

**Document Number** DevOpsLab308

**Contributor:**

Shruti Sinhaa

Table of Contents

[1 OBJECTIVE 3](#_Toc160709859)

[2 Create New VM 4](#_Toc160709860)

[3 Create Main.tf 4](#_Toc160709861)

[4 Execute Terraform 6](#_Toc160709862)

[5 Verift the Backup detail 9](#_Toc160709863)

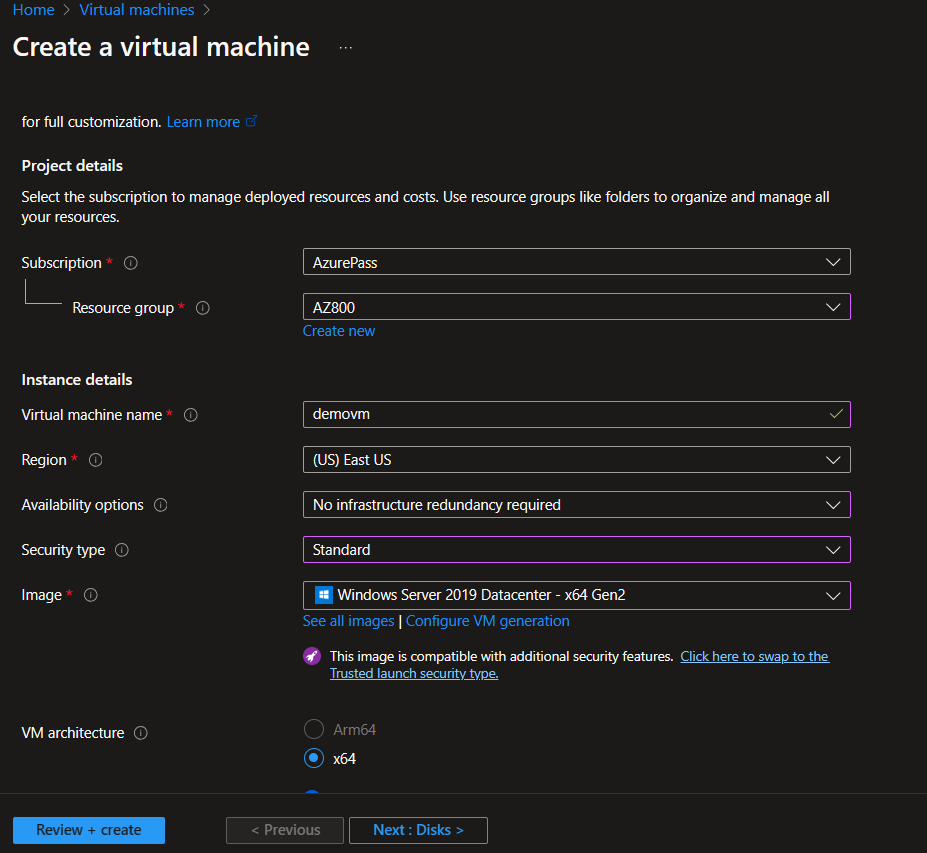
[6 Delete the Resources 13](#_Toc160709864)

# OBJECTIVE

This lab shows you how to enable Azure Backup on Linux VMs.. Terraform AzureRM provider has three relevant resources:

* [azurerm\_ \_virtual\_machine](https://registry.terraform.io/providers/hashicorp/azurerm/latest/docs/resources/linux_virtual_machine): parameters provision\_vm\_agent and allow\_extension\_operations should be true or enabling backups will fail (with or without Terraform)
* [azurerm\_virtual\_machine\_extension](https://registry.terraform.io/providers/hashicorp/azurerm/latest/docs/resources/virtual_machine_extension): this can be used to run/enable a VM extension on a VM
* [azurerm\_backup\_protected\_vm](https://registry.terraform.io/providers/hashicorp/azurerm/latest/docs/resources/backup_protected_vm): this resource enables backups on a VM
* [azurerm\_recovery\_services\_vault](https://registry.terraform.io/providers/hashicorp/azurerm/latest/docs/resources/recovery_services_vault): the recovery services vault where backups will be stored
* [azurerm\_backup\_policy\_vm](https://registry.terraform.io/providers/hashicorp/azurerm/latest/docs/resources/backup_policy_vm): backup schedule and retainment rules

# Create New VM



# Create Main.tf

terraform {

  required\_providers {

    azurerm = {

      source = "hashicorp/azurerm"

      version = "~>3.0"

    }

  }

}

provider "azurerm" {

  features {}

  subscription\_id   = "49c56ee8-d443-4854-a62c-3a0aae84ac6f"

  tenant\_id         = "be04fbd5-6b00-412c-a86c-ca105b5cce90"

  client\_id         = "0b381472-3197-49d4-a324-f1a96a23c8a7"

  client\_secret     = "t8P8Q~Yb4L8Dn~oj9t5GPzFOROUqJM4VJXTTBbLH"

}

# Define variables

variable "resource\_group\_name" {

  description = "Name of the resource group where resources will be created"

  type        = string

  default     = "Az800"  # Setting default value to "Az800"

}

variable "location" {

  description = "Azure region where resources will be deployed"

  type        = string

  default     = "East US"  # Setting default value to "East US"

}

variable "vm\_name" {

  description = "Name of the Azure VM to backup and restore"

  type        = string

  default     = "demovm"  # Setting default value to "demovm"

}

variable "backup\_policy\_name" {

  description = "Name of the backup policy to apply to the VM"

  type        = string

  default     = "default-backup-policy"  # Setting default value to "default-backup-policy"

}

# Retrieve VM ID using its name

data "azurerm\_virtual\_machine" "example" {

  name                = var.vm\_name

  resource\_group\_name = var.resource\_group\_name

}

# Create a recovery services vault

resource "azurerm\_recovery\_services\_vault" "example" {

  name                = "${var.resource\_group\_name}-recovery-vault"

  location            = var.location

  resource\_group\_name = var.resource\_group\_name

  sku = "Standard"  # Provide appropriate SKU here

}

resource "azurerm\_backup\_policy\_vm" "example" {

   name                = var.backup\_policy\_name

  resource\_group\_name = var.resource\_group\_name

  recovery\_vault\_name = azurerm\_recovery\_services\_vault.example.name

  backup {

    frequency = "Daily"

    time      = "23:00"

  }

  retention\_daily {

    count = 10

  }

}

# Enable backup for the existing VM

resource "azurerm\_backup\_protected\_vm" "example" {

  resource\_group\_name = var.resource\_group\_name

  recovery\_vault\_name = azurerm\_recovery\_services\_vault.example.name

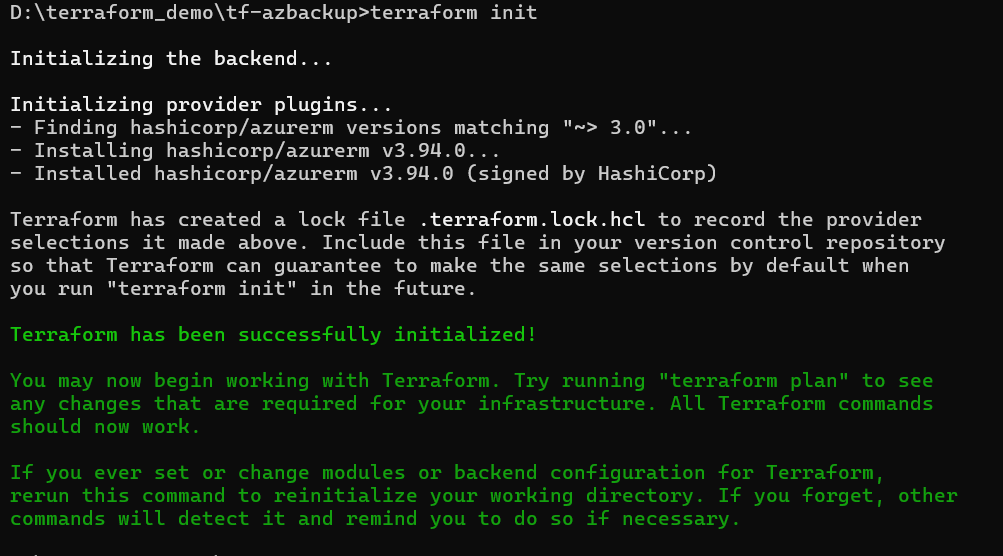
  source\_vm\_id        = data.azurerm\_virtual\_machine.example.id

  backup\_policy\_id    = azurerm\_backup\_policy\_vm.example.id  # Corrected argument for backup policy

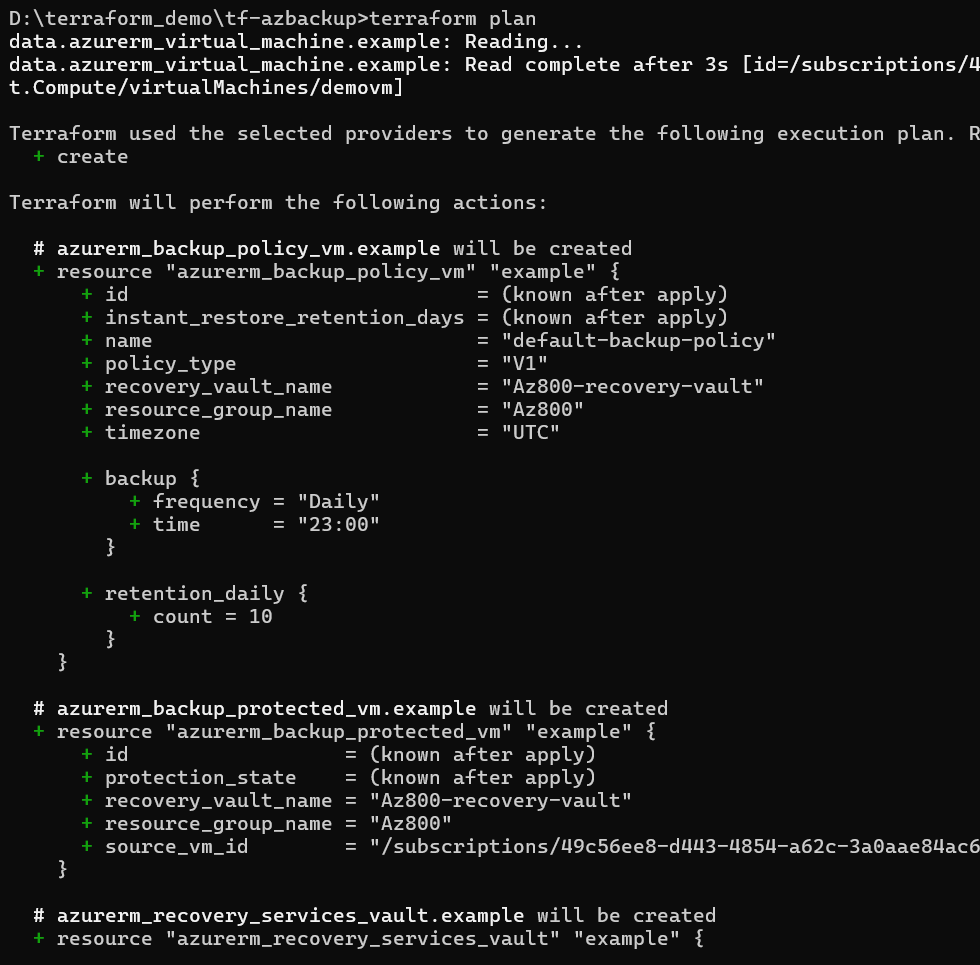
}

# Execute Terraform

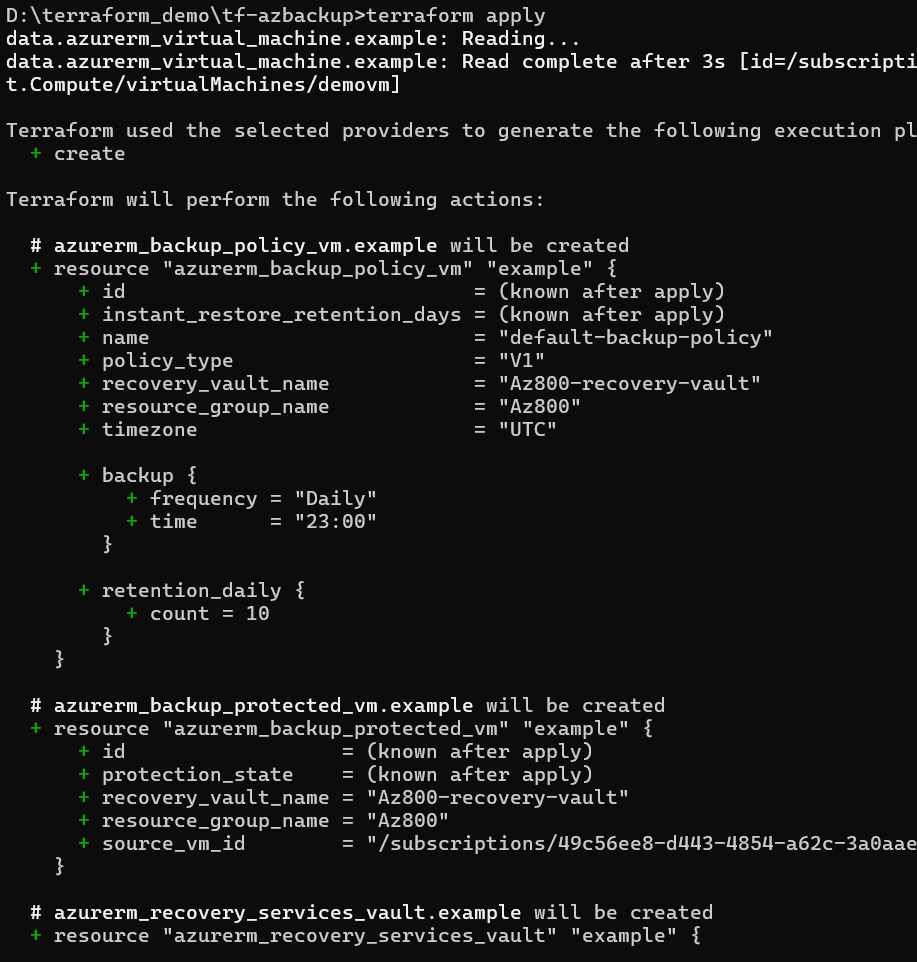
**terraform init**



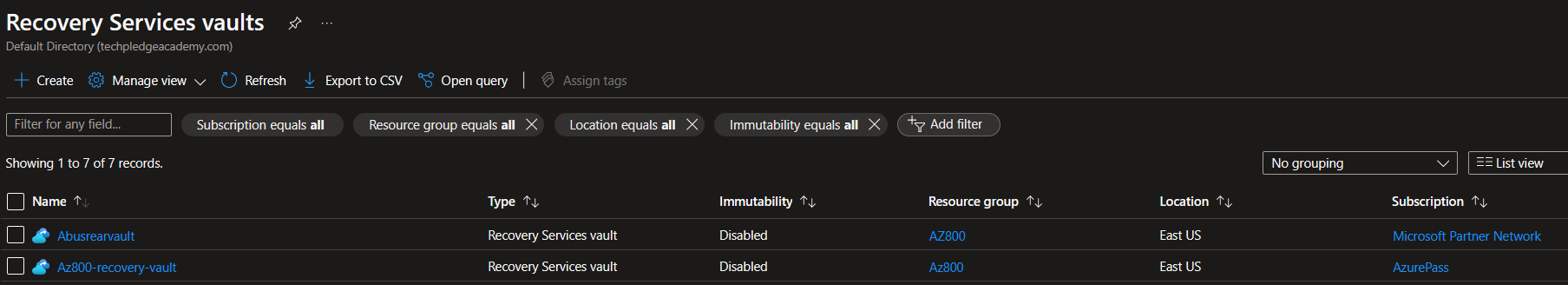
**terraform plan**

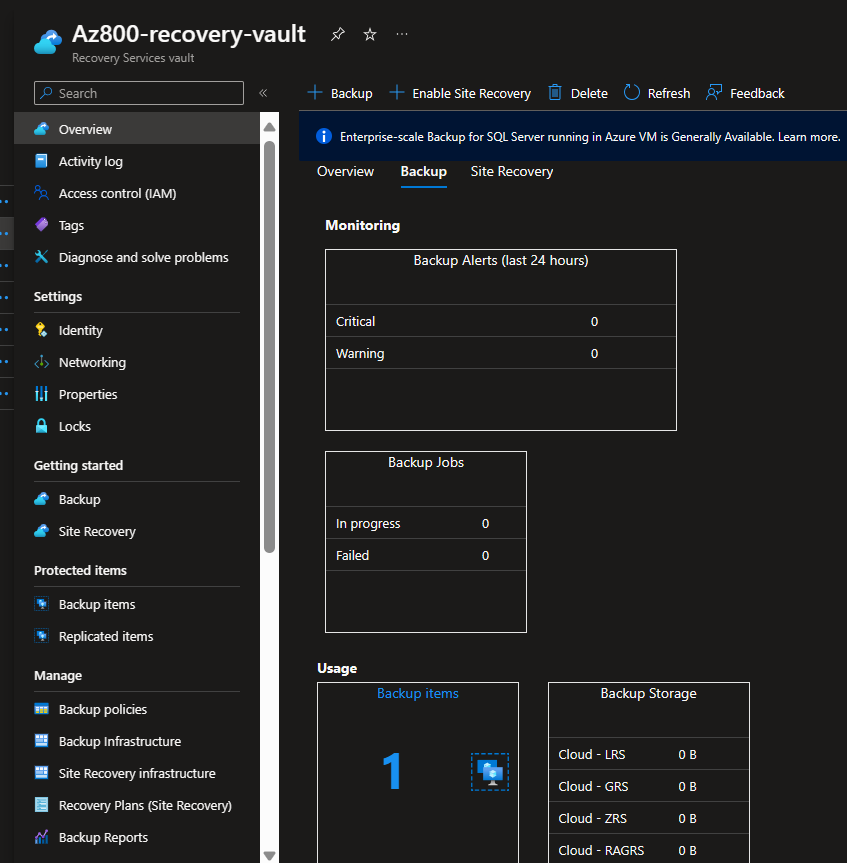
****

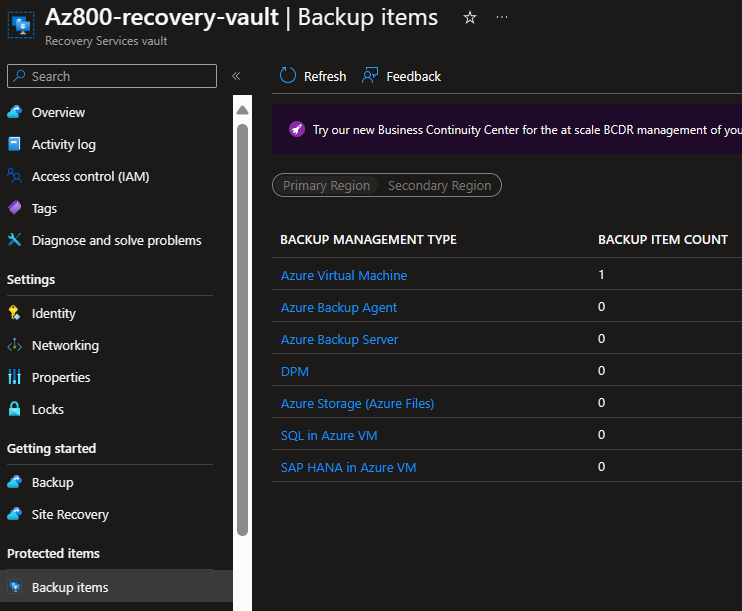
**terraform Apply**

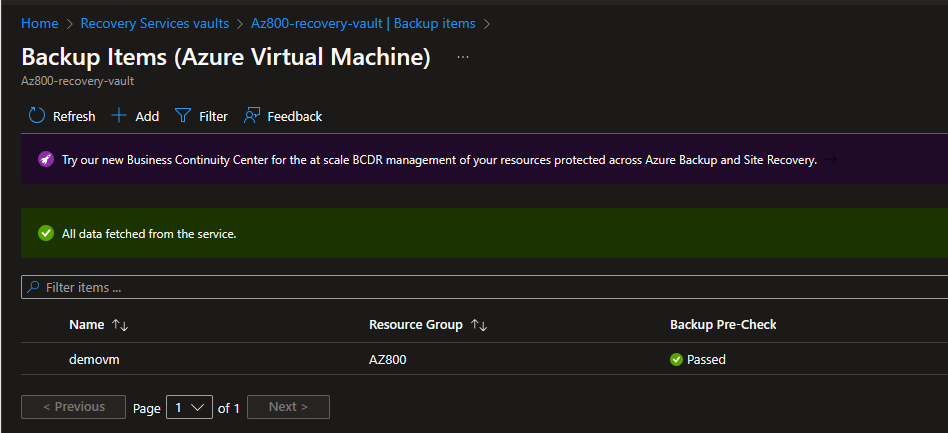
****

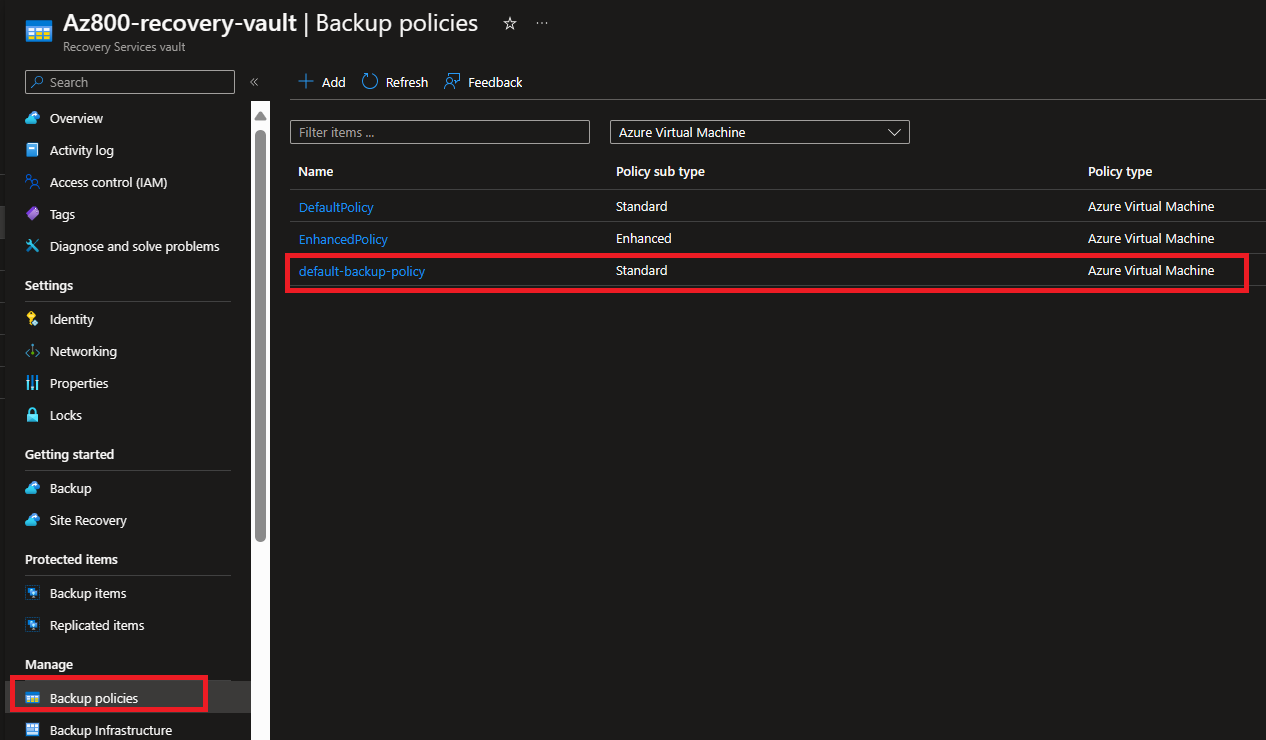
# Verift the Backup detail

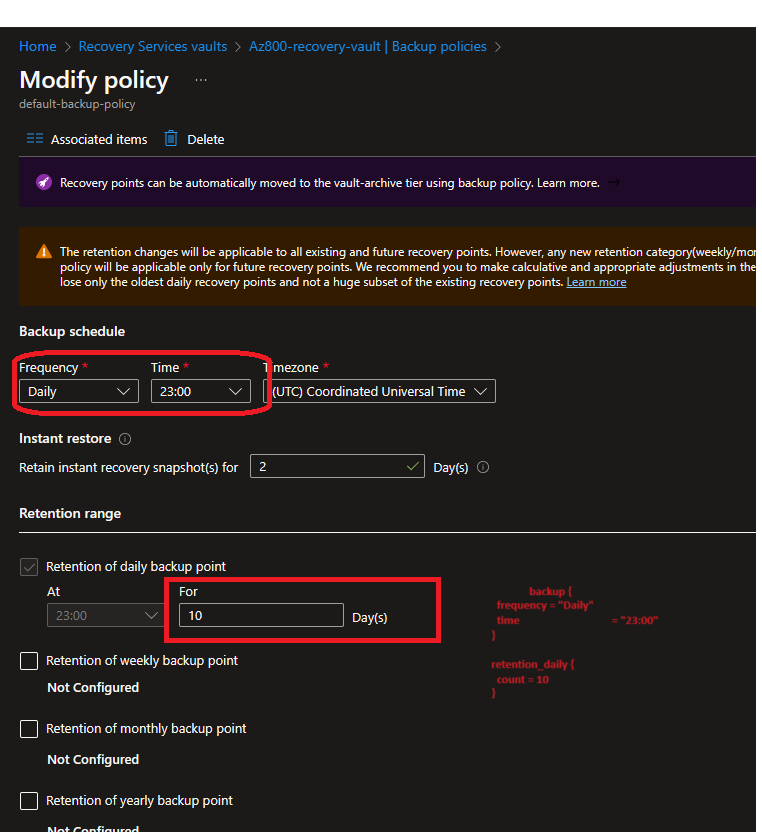
****

****

****

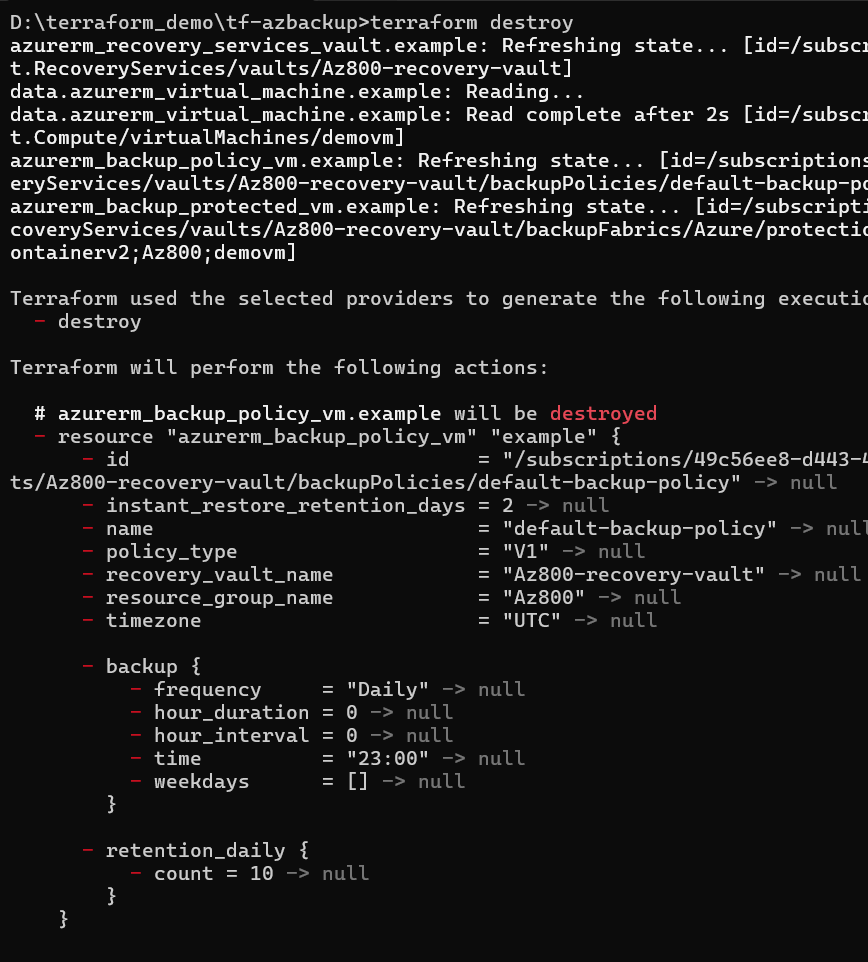
****

****

****

# Delete the Resources

**terraform Destroy**

****