# **Solution: Provisioning a Windows VM Using Terraform**

This Terraform configuration provisions a **Windows EC2 instance** on AWS with **RDP (port 3389) access**, assigns a **Name tag**, and allows remote connectivity.

## **Step 1: Create a Terraform Configuration File**

**Create a new directory and navigate to it:** mkdir terraform-windows-vm && cd terraform-windows-vm

**Create a new Terraform file:** touch main.tf

**Open main.tf in a text editor and paste the following code:**

### **Terraform Configuration (main.tf)**

terraform {

required\_providers {

aws = {

source = "hashicorp/aws"

version = "~> 4.16"

}

}

required\_version = ">= 1.2.0"

}

provider "aws" {

region = "us-west-2" # Change this to your preferred region

}

resource "aws\_security\_group" "windows\_sg" {

name = "windows-rdp-sg"

description = "Allow RDP access"

ingress {

from\_port = 3389

to\_port = 3389

protocol = "tcp"

cidr\_blocks = ["0.0.0.0/0"] # Open to all (not recommended for production)

}

egress {

from\_port = 0

to\_port = 0

protocol = "-1"

cidr\_blocks = ["0.0.0.0/0"]

}

}

# Windows EC2 Instance

resource "aws\_instance" "windows\_vm" {

ami = "ami-00cf40ed19c0fca69" # Windows AMI ID (Verify for your region)

instance\_type = "t3.micro" # Windows requires more memory

key\_name = "Arjun" # Replace with your AWS key pair name

security\_groups = [aws\_security\_group.windows\_sg.name]

tags = {

Name = "WindowsVM-Terraform"

}

}

## **Step 2: Initialize Terraform**

Before using Terraform, initialize the working directory:

terraform init

✔️ This downloads necessary provider plugins and prepares the working directory.

## **Step 3: Format and Validate Terraform Configuration**

**Format Terraform Code:** terraform fmt

1. ✔️ Ensures that the Terraform file is properly structured.

**Validate Terraform Configuration:** terraform validate

1. ✔️ Checks for syntax errors and misconfigurations.

## **Step 4: Plan the Deployment**

Before applying, review the planned changes:

terraform plan

✔️ This command shows what Terraform will create or change.

## **Step 5: Apply the Configuration**

Run the following command to create the Windows EC2 instance:

terraform apply

✔️ Terraform will prompt for confirmation. Type **yes** and press Enter.

Terraform will now:

* Provision a Windows EC2 instance in **us-west-2**.
* Assign it the **ami-0b2f6494ff0b07a0e** (Replace with a valid Windows AMI).
* Use a **t2.medium** instance type.
* Attach a **security group** allowing RDP access.
* Assign a **tag: Name = "WindowsVM-Terraform"**.

## **Step 6: Verify the EC2 Instance**

Once the deployment is complete, run:

aws ec2 describe-instances --filters "Name=tag:Name,Values=WindowsVM-Terraform"

✔️ This will display details about your Windows EC2 instance.

Alternatively, you can:

* Go to **AWS Console → EC2 Dashboard**
* Locate the instance under **Instances**
* Copy the **Public IP Address**

### **Connect to the Windows VM via RDP:**

1. Open **Remote Desktop Connection** (mstsc on Windows).
2. Enter the **Public IP Address** from the AWS EC2 instance.
3. Use **Administrator** as the username and the **password retrieved from AWS Console** under "Connect to Windows Instance."

## **Step 7: View Terraform State**

After deployment, inspect Terraform’s state using:

terraform show

✔️ This will display all managed resources, including the Windows VM, security groups, and associated metadata.

## **Step 8: Destroy the EC2 Instance (Optional)**

If you want to delete the instance and free up resources, run:

terraform destroy

✔️ Terraform will ask for confirmation. Type **yes** to proceed.

## **Submission Requirements:**

✅ **Submit your main.tf file** ✅ **Attach a screenshot of your Windows VM running in AWS** ✅ **Provide the output of terraform show after deployment**

## **Bonus Task (Optional):**

🔹 **Create an Elastic IP** and attach it to your Windows VM.  
 🔹 **Use Terraform’s user\_data to run a PowerShell script on startup** (e.g., install updates, enable IIS).

## **Expected Outcome:**

By completing this assignment, you will:  
 ✔ Learn how to provision a Windows VM using Terraform.  
 ✔ Understand security group configuration for RDP access.  
 ✔ Practice Terraform’s core workflow (init, fmt, validate, plan, apply, show, destroy).

🚀 **Well Done! You've successfully deployed a Windows VM using Terraform!** 🚀