Deploy Threat Defense Virtual in an Existing VPC on AWS

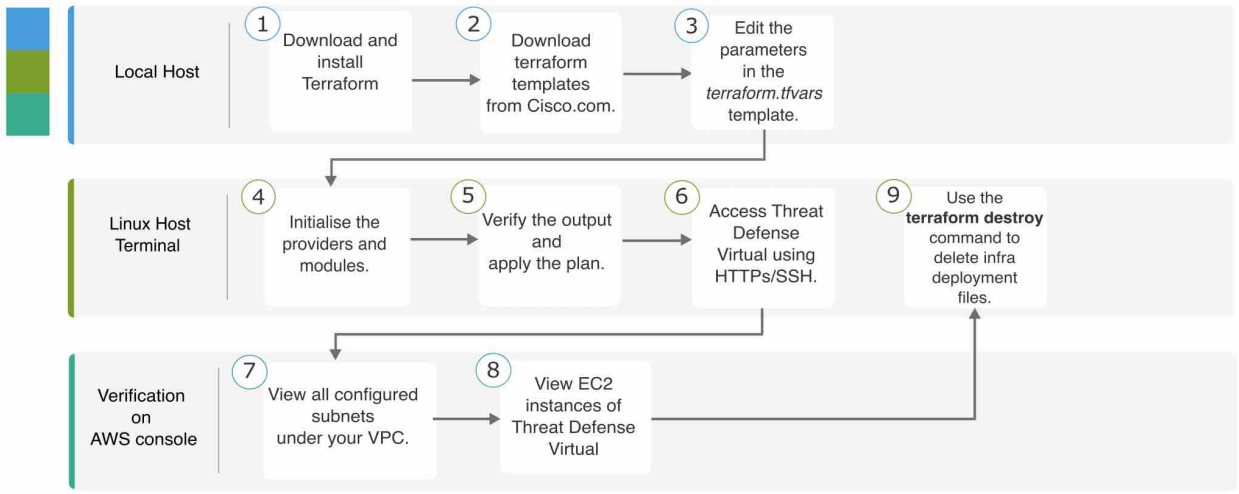
First Published: 2024-01-11

Introduction

This document describes the procedure of deploying Cisco Secure Firewall Threat Defense Virtual and other network components on AWS using a terraform script. This procedure creates all the required resources inside an existing VPC on your AWS account. If you want to deploy the Threat Defense Virtual on AWS in a new VPC, see Deploy Threat Defense Virtual in a New VPC on AWS.

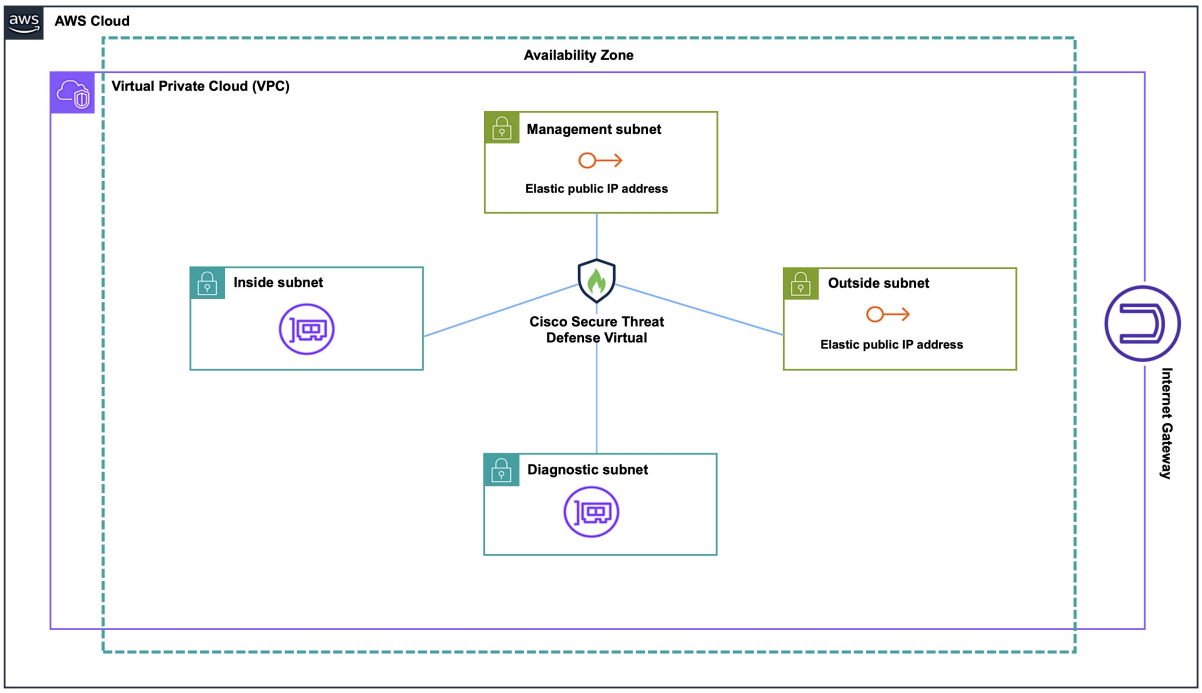
End-to-End Process

The following flowchart illustrates the workflow for deploying Threat Defense Virtual in an existing VPC on AWS.



Sample Topology

The following network topology is deployed on AWS.



Prerequisites

- Download and install Terraform on your local machine. For more information, see Install Terraform.
- An AWS account with proper permissions for creating VPCs and EC2 instances. For more information, see Amazon VPC policy examples.

Procedure

Perform the following steps to deploy the required infrastructure in a VPC that already exists in your AWS account.

Procedure

Step 1 Download the terraform scripts from here.

Step 2 Extract the zip file and open the folder.

Step 3 Open the **terraform.tfvars** file by using a code editor or "vim" and provide inputs.

Step 4 Add your **aws_access_key**, **aws_secret_key** and **region** in the space provided between double quotes. For example, **region = "us-east-1"**. For information on how to fetch access keys and secret access key for your account, see Managing access keys for IAM users.

Step 5 Optionally, add a password for admin in the **admin_password** field. By default, the password is Admin123.

Step 6 If required, change the version of the Threat Defense Virtual in the **"FTD_version"** field.

Step 7 Provide the name of the VPC and the CIDR block in the **vpc_name** and **vpc_cidr** fields.

Step 8 If your VPC does not have an internet gateway attached to it, set the **create_igw** field to **true**. Otherwise, set it to **false**.

Step 9 Enter the appropriate subnet CIDR blocks for the four different subnets - **mgmt_subnet, outside_subnet, inside_subnet**, and **diag_subnet**.

Note Ensure that you only add subnet CIDRs that are not already present in the VPC to avoid any potential conflicts.

Step 10 Enter the private IP addresses corresponding to the respective subnet's CIDR blocks in the **ftd01_mgmt_ip, ftd01_outside_ip, ftd01_inside_ip** and **ftd01_diag_ip** fields.

Step 11 Initialize the providers and modules by using the following command:

```
terraform init
```

Step 12 Submit the terraform plan by using the following command:

```
terraform plan --out filename
```

Step 13 Verify the output of the plan in the terminal and then apply the plan by using the following command:

```
terraform apply filename
```

Step 14 The terraform output displays the IP address of the management interface and the command to SSH into the firewall. Use these to access the Threat Defense Virtual over HTTPs/SSH.

Step 15 Open the AWS console after the deployment is complete. Go to your provided region and validate the final configuration.

a) Go to **Service > VPC** to view all the configured subnets under your VPC.
b) Go to **Service > EC2** to view the EC2 instance of Threat Defense Virtual with the name - Cisco Threat Defense Virtual.

Note Do not delete the **.terraform** folder and **terraform.tfstate** files as they are required for the clean-up process.

Clean-Up

We recommend that you delete the infrastructure deployment once it's not needed to prevent unnecessary billing on your AWS account.

To delete the infrastructure deployment that was created by terraform, enter the **terraform destroy** command from the same directory in which you entered the **terraform apply** command.

```
terraform destroy
Type "yes" to delete the infrastructure deployment.
```

Note The **terraform destroy** command does not delete anything that was manually configured on your AWS account.

After entering the command, verify that all the resources are deleted from your AWS account.