



Home / AWS / Guided Lab / Create a 2 tier Amazon VPC using Terraform

Create a 2 tier Amazon VPC using Terraform

Level: Fundamental

Amazon VPC Amazon Web Services Terraform

Required Points	10
Lab Duration	01:00:00
Average Start time	Less than a minute

Start Lab →

Need help?

- How to use Hands on Lab
- Troubleshooting Lab
- FAQs

Submit Feedback

Share

Lab Overview

- Cloud Architect
- Networking, Infrastructure

Lab Details

- This lab walks you through the steps to create a 2 tier Amazon VPC using Terraform.
- Duration: **60 minutes**

Privacy - Terms

3. AWS Region: **US East (N. Virginia) us-east-1.**

Introduction

What is VPC?

- VPC stands for Virtual Private Cloud.
- It's a custom-defined virtual network within the AWS Cloud.
- Users can logically create their personal network, designing and implementing a separate and independent network that would operate in the AWS Cloud.
- Primary components are Subnets, IP addresses, NAT Devices (Instances & Gateways), Route Tables, Internet & Virtual Private Gateways, Access Control Lists, Security groups, VPC Endpoints.
- A subnet is a segment of the VPC IP address range, where we can launch EC2 Instances, RDS, and other AWS resources.
- Subnet are further classified as Public and Private.

Basic Understanding before we start building VPC from scratch

- When you create an Amazon AWS VPC, you specify a set of IP addresses in the form of a **Classless Inter-Domain Routing (CIDR)** block (Ex: **10.0.0.0/16**).
- You can assign a single CIDR block to a VPC. The allowed block size is between a /28 netmask and /16 netmask. In other words, the VPC can contain from 16 to 65,536 IP addresses.

What is Terraform?

- It is an open-source IaC (Infrastructure as a code) software tool where you define and create resources using providers in the declarative configuration language example JSON.
- With Terraform, You can package and reuse the code in form of modules.
- It supports a number of cloud infrastructure providers such as AWS, Azure, GCP, IBM Cloud, OCI, etc.
- Terraform has four major commands:
 - terraform init
 - terraform plan
 - terraform apply

- terraform destroy

Prerequisite

- Install Terraform in your local machine using this official guide by Hashicorp.
 - To install Terraform using CLI, use this guide <https://learn.hashicorp.com/tutorials/terraform/install-cli>
 - To install Terraform by downloading, use this guide <https://www.terraform.io/downloads.html>
- Download and Install Visual Studio Code editor using this guide <https://code.visualstudio.com/download>

Architecture Diagram

Task Details

1. Sign in to the AWS Management Console
2. Setup Visual Studio Code
3. Create a variables file
4. Create VPC and its components in main.tf file
5. Confirm the installation of Terraform by checking the version
6. Apply terraform configuration
7. Check the resources in AWS Console
8. Validation of the lab
9. Delete resources

Launching Lab Environment

1. To launch the lab environment, Click on the **Start Lab** button.
2. Please wait until the cloud environment is provisioned. It will take less than a minute to provision.
3. Once the Lab is started, you will be provided with **IAM username, Password, Access Key, and Secret Access Key.**

Note : You can only start one lab at any given time

[About Us](#) [Subscription](#) [Instructions and Guidelines](#) [FAQ's](#) [Contact Us](#)



© 2023, Whizlabs Software Pvt. Ltd.

