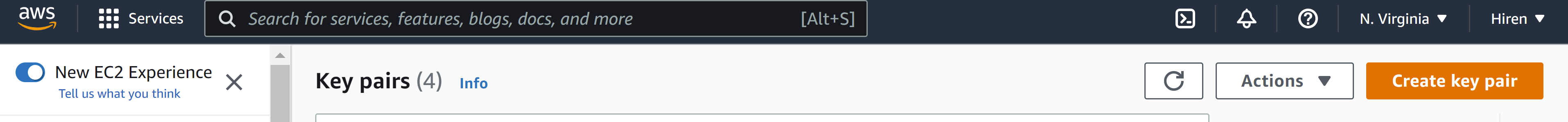
**Name:** Hiren Shah

**Purpose:** Using the AWS Consol and Terraform… create a custom VPC with Internet Gateway, custom Route Tables, a Subnet, associate the subnet with the Route Table, SG to allow ports 22,80,443, create a Network Interface with IP in the subnet, assign an Elastic IP to the network interface, and create a Ubuntu server and install/enable Apache. This will let us have a live running web server on AWS using Terraform

1. The first thing you want to do is create a Key Pair in AWS Console. Go to the “EC2” and create a new key pair and save it on your laptop.



1. Create a VPC in Terraform:

Text

Description automatically generated

1. Create an Internet Gateway **and** make sure to pass it the vpc\_id:

Text

Description automatically generated

1. Create a custom route table and define the routes to a default route so all ipv4 traffic where it points:

Text

Description automatically generated

1. Create a subnet where the web-server will reside on. Make sure to specifiy the AZ:

A screenshot of a computer

Description automatically generated with medium confidence

1. Make sure to associate the subnet to the route table that was created:

Text

Description automatically generated

1. Create a Security Group:

Text

Description automatically generated

Text

Description automatically generated

1. Create a Network Interface, assign the NIC to the subnet created, and specify any IP a proper IP address and pass in the Security Group id:

Text

Description automatically generated

1. Create an elastic IP so that anybody on the internet can access it. The AWS EIP may require to IGW to exist prior to association. Use the depends\_on to set dependency on IGW:

Text

Description automatically generated

1. Creat an Ubuntu server and install apache. Make sure to grab the AMI from the AWS Console ubuntu “launch instance” section. Make sure to hard code the AZ. In addition, make sure to reference the key pair that we downloaded in step 1! Write a small bash script to show to launch and enable apache and make sure to echo something out: (see screenshot on next page)

Text

Description automatically generated

1. Next we will make sure everything looks good on the console and the web page loads with the IP address:

Table

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

**Connect to that device via SSH:**

1. Open up Putty and under host name type in “ubuntu@[ip adress goes here] on Port 22 of course:

Graphical user interface, text, application

Description automatically generated

1. Next go to the “SSH” and then “Auth” section on the left side navigation. In the field for “Private key file for authentication” 🡪 browse for you Private Key where you saved it. Make sure it is saved as a ppk. If not you will have to use puttygen to convert it to ppk.

Graphical user interface, text, application

Description automatically generated

1. Voila! You are now logged in using putty.. You can check if apache is properly installed (see screenshot below):

Graphical user interface, text

Description automatically generated

1. Finally, make sure you destroy the project so you don’t get any unexpected bills.

Text

Description automatically generated