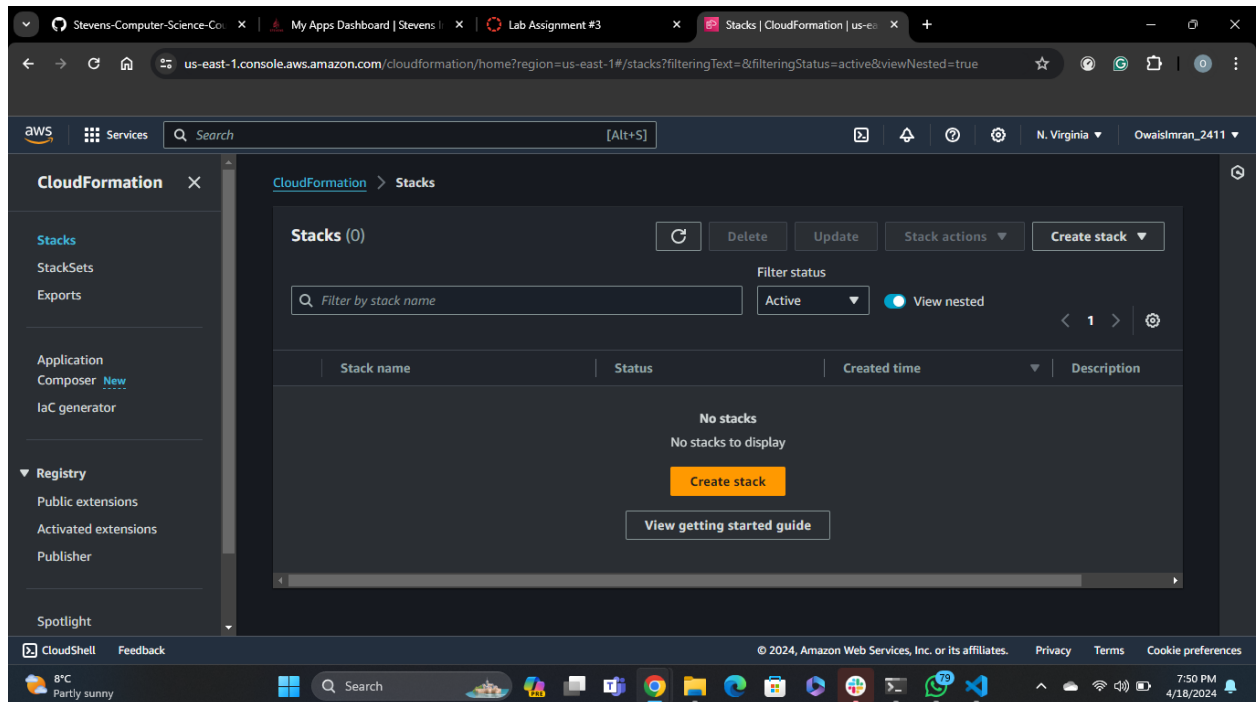
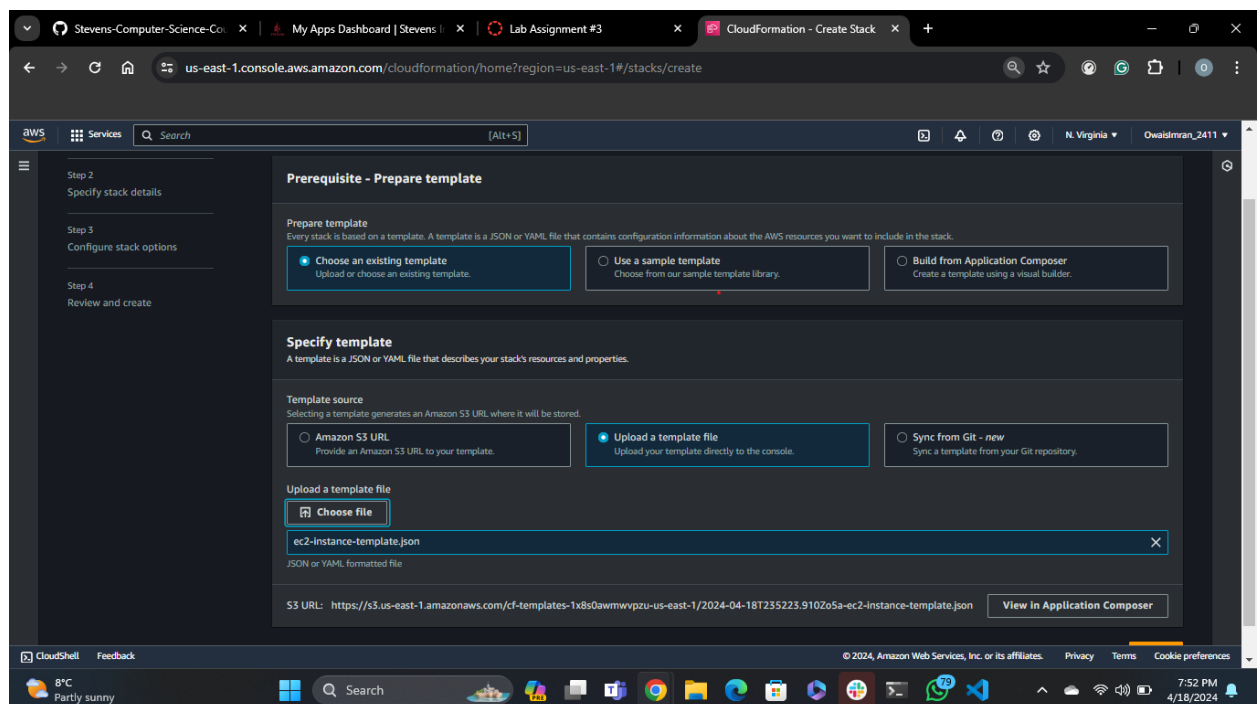


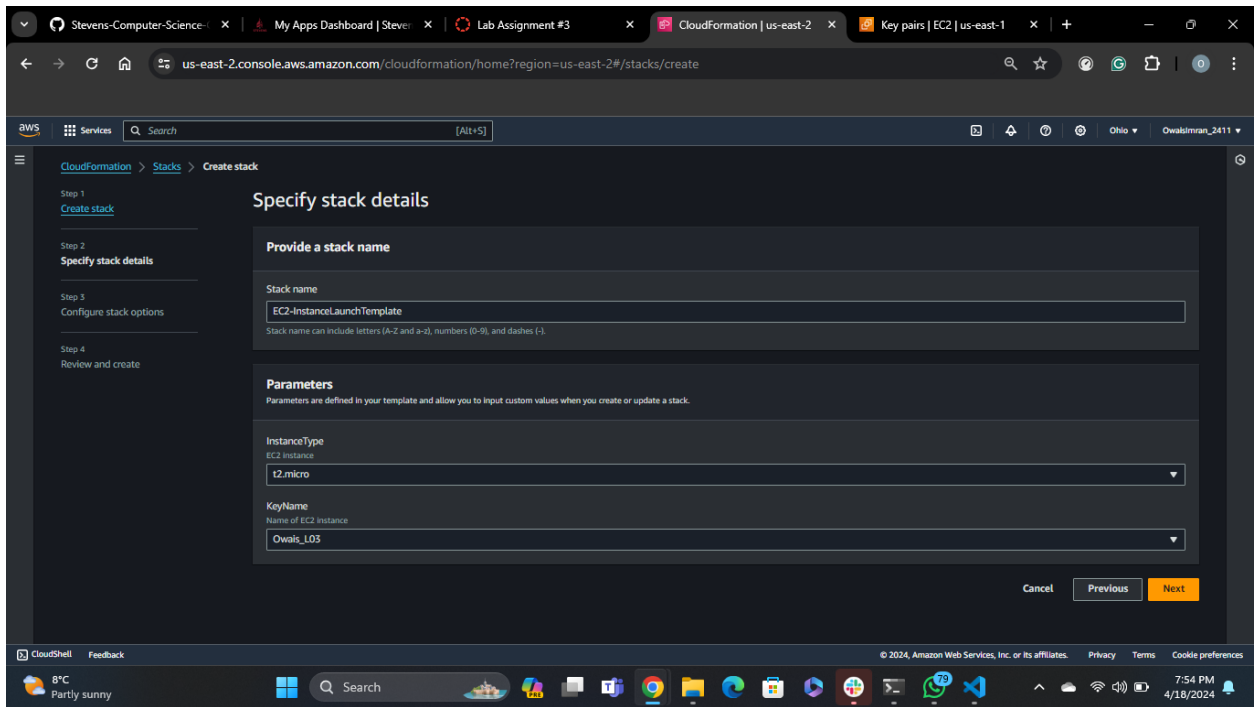
1. On your AWS Management Console, Search for Cloud Formation. Under stacks, click on Create Stack.



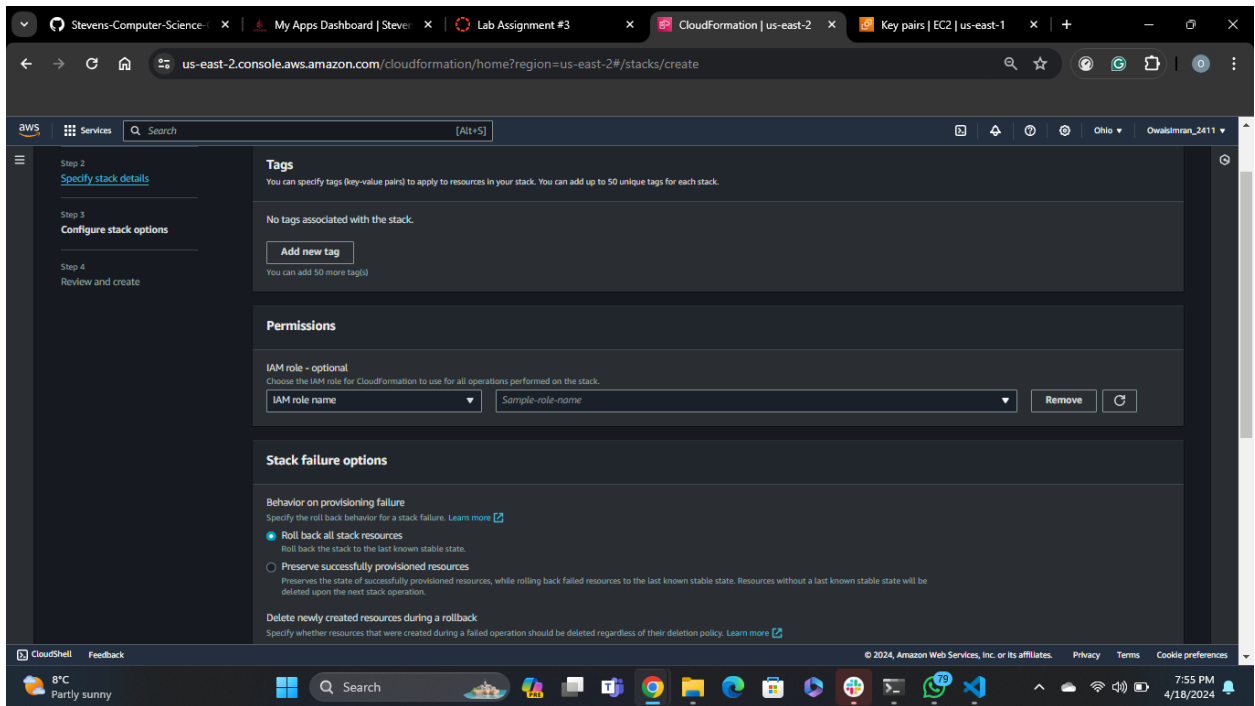
2. After clicking Create Stack, fill in the pre requisite to prepare a template, upload/reuse a template.



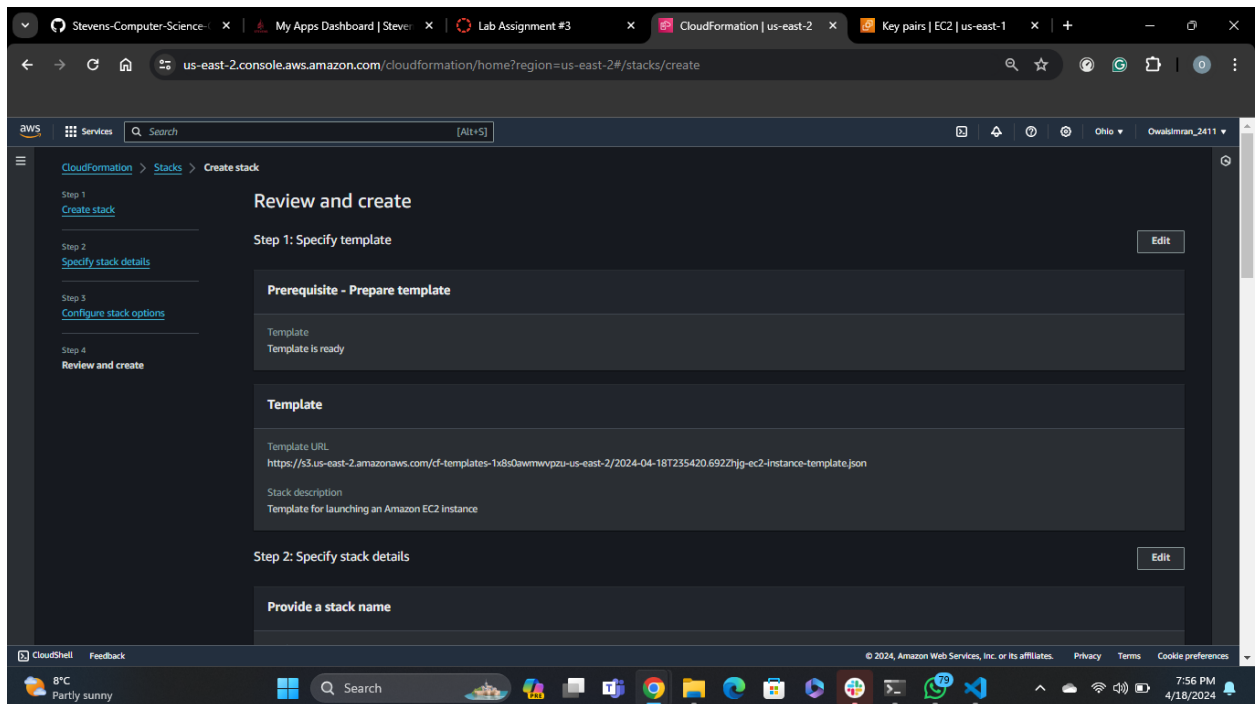
3. Specify the Stack Details, like naming the template, providing the required parameters.



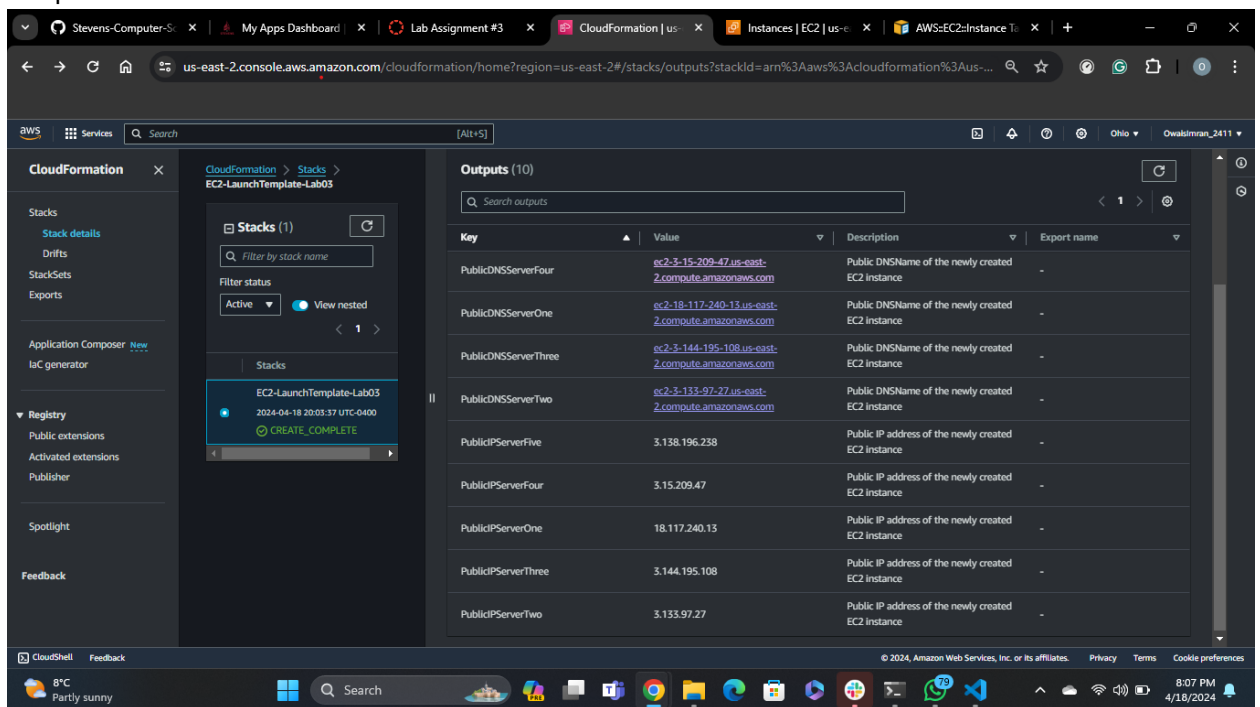
4. Configure other stack options (if needed), otherwise leave it on default.



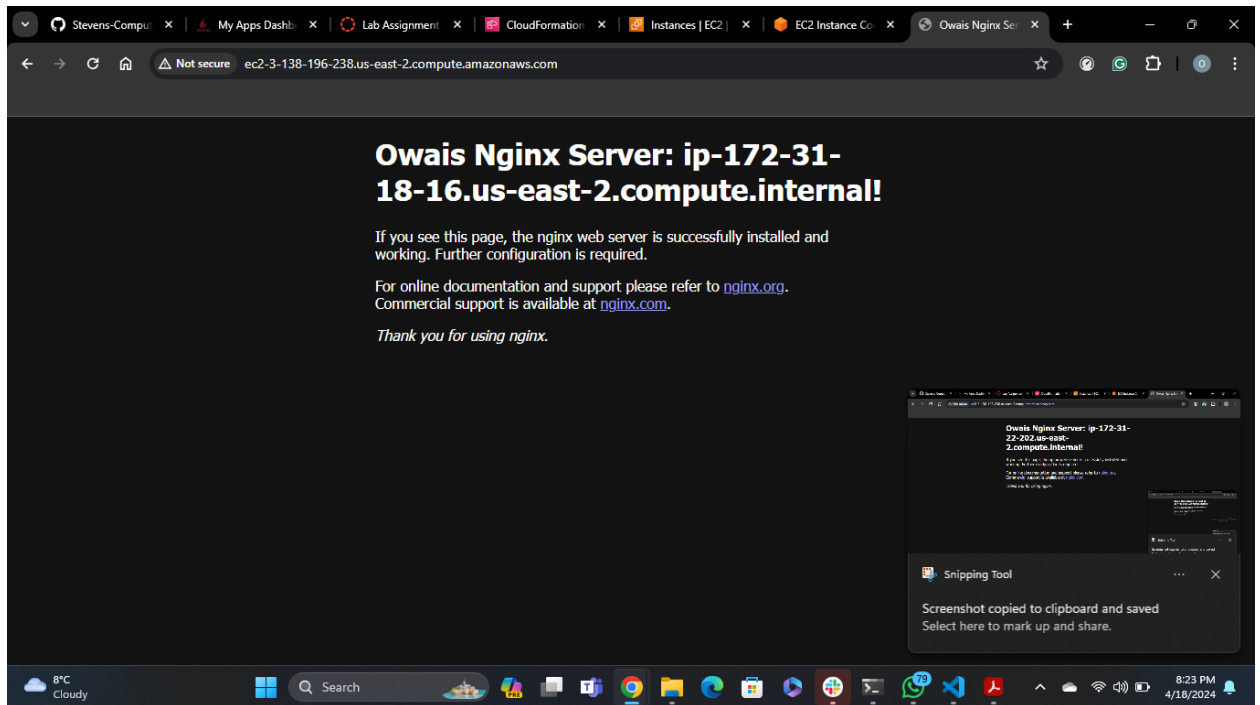
5. Review the information and create stack



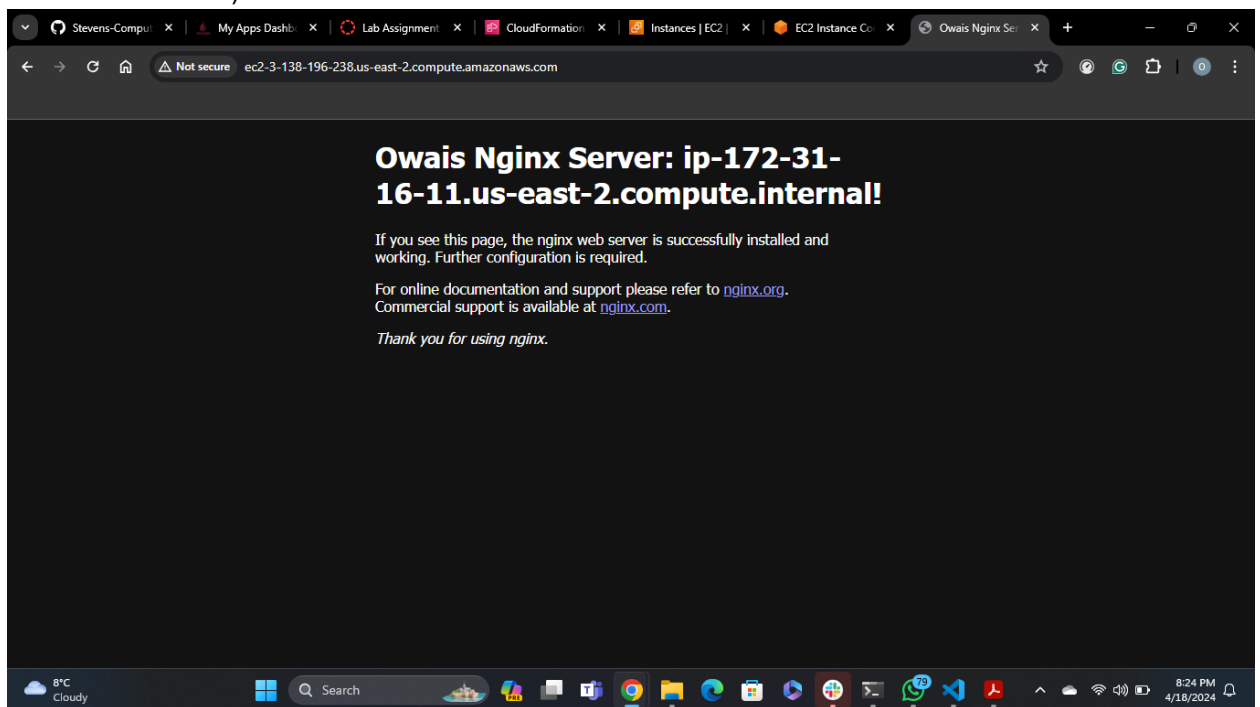
6. Output screenshot.



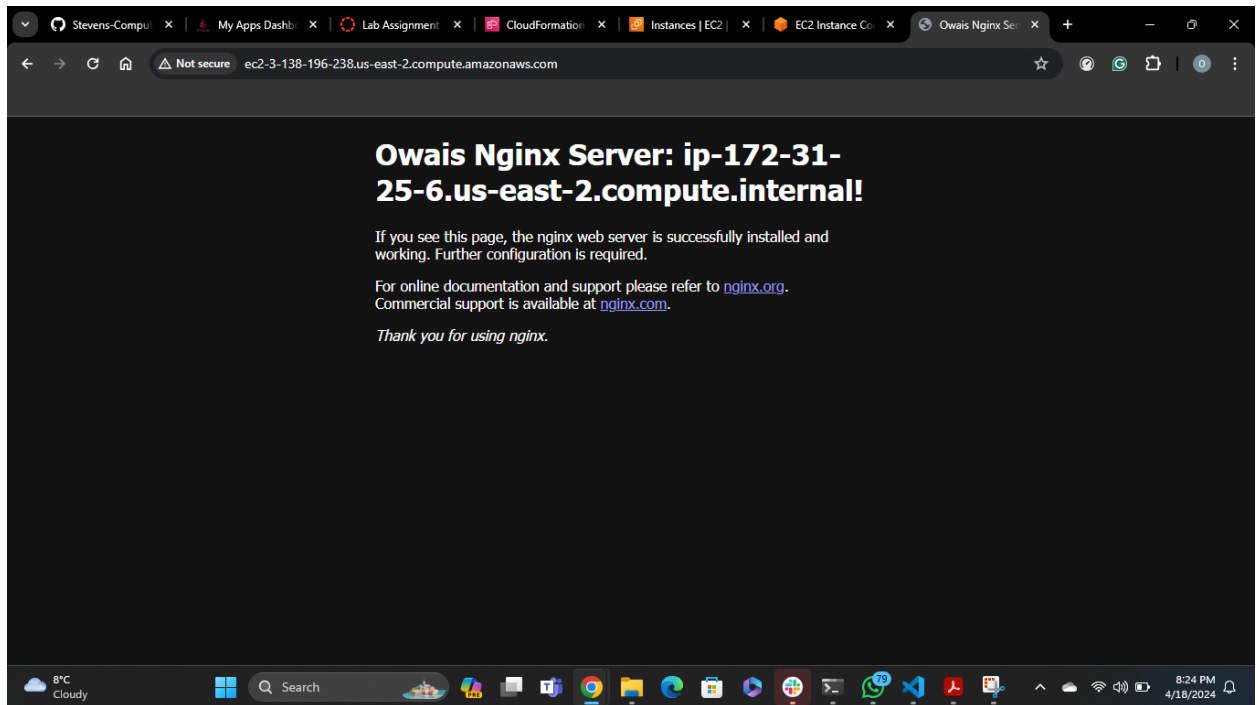
7. Output of Web Server 1 (Served from LoadBalancer Public DNS, PublicDNSServerFive in above screenshot)



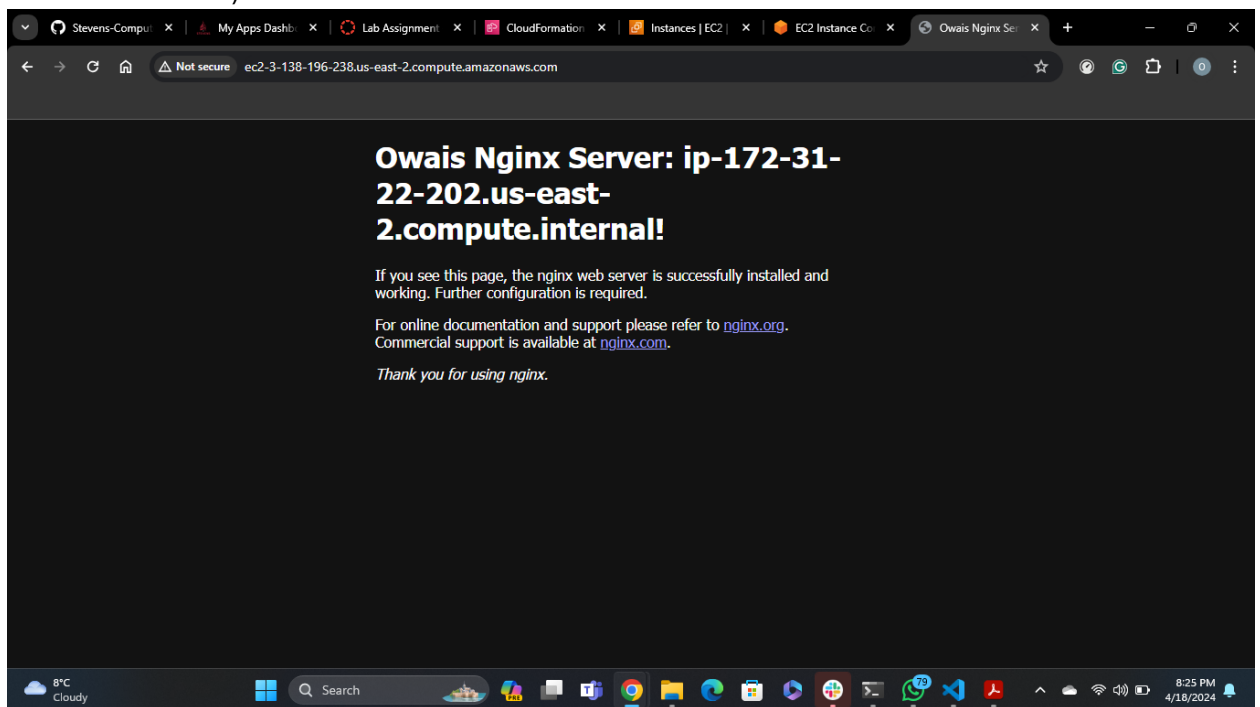
8. Output of Web Server 2 (Served from LoadBalancer Public DNS, PublicDNSServerFive in above screenshot)



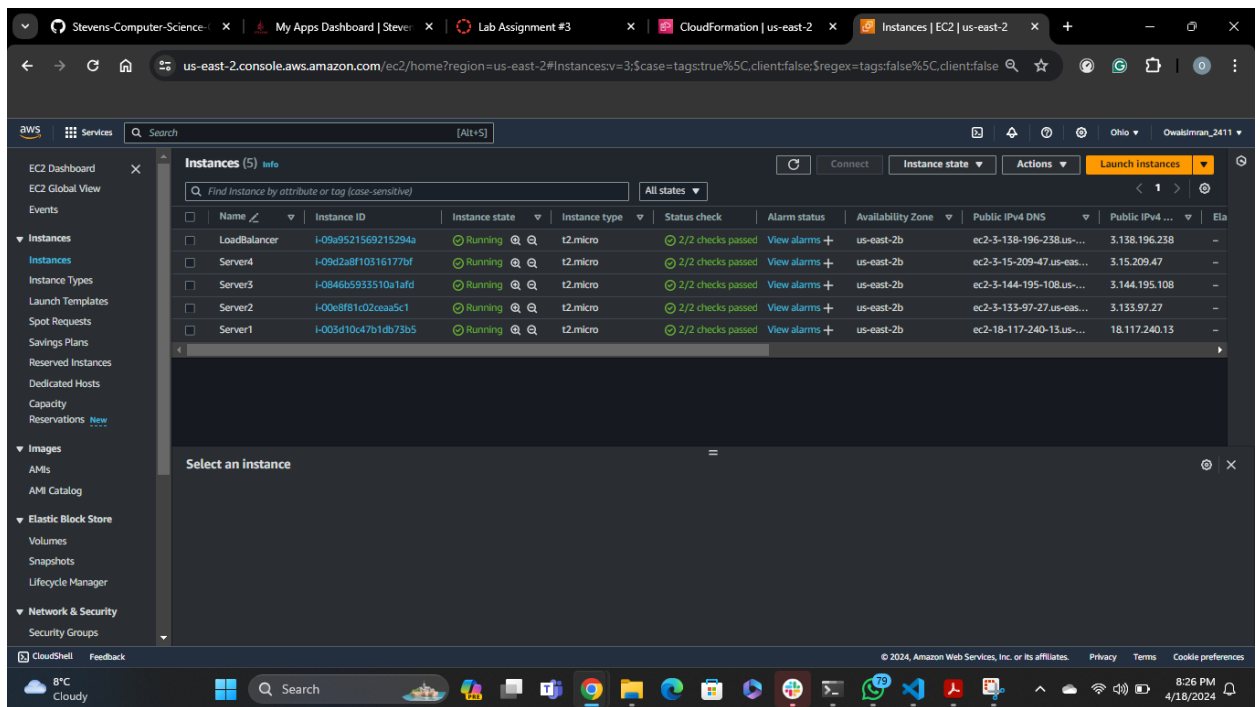
9. Output of Web Server 3 (Served from LoadBalancer Public DNS, PublicDNSServerFive in above screenshot)



10. Output of Web Server 4 (Served from LoadBalancer Public DNS, PublicDNSServerFive in above screenshot)



11. Screenshots of EC2 instance launched.



12. Commands executed on LoadBalancer instance to configure a load balancer.

```
[root@ip-172-31-16-119 ec2-user]# history
 1 dnf install nginx -y
 2 clear
 3 vi /etc/nginx/nginx.conf
 4 systemctl start nginx
 5 ls
 6 vi /etc/nginx/nginx.conf
 7 sed -i "s/SERVER1/ec2-18-117-240-13.us-east-2.compute.amazonaws.com/g" /etc/nginx/nginx.conf
 8 cat /etc/nginx/nginx.conf
 9 sed -i "s/SERVER2/ec2-3-133-97-27.us-east-2.compute.amazonaws.com/g" /etc/nginx/nginx.conf
10 sed -i "s/SERVER3/ec2-3-144-195-108.us-east-2.compute.amazonaws.com/g" /etc/nginx/nginx.conf
11 sed -i "s/SERVER4/ec2-3-15-209-47.us-east-2.compute.amazonaws.com/g" /etc/nginx/nginx.conf
12 clear
13 cat /etc/nginx/nginx.conf
14 clear
15 systemctl start nginx
16 systemctl enable nginx
17 clear
18 history
[root@ip-172-31-16-119 ec2-user]#
```

## EC2-CloudFormation Template:

```
{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Description": "Template for launching an Amazon EC2 instance",
  "Parameters": {
    "KeyName": {
      "Description": "Name of EC2 instance",
      "Type": "AWS::EC2::KeyPair::KeyName",
      "ConstraintDescription": "must be the name of an existing EC2 KeyPair."
    },
    "InstanceType": {
      "Description": "EC2 instance",
      "Type": "String",
      "Default": "t2.micro",
      "AllowedValues": [
        "t2.micro"
      ],
      "ConstraintDescription": "must be a valid EC2 instance type."
    }
  },
  "Resources": {
    "EC2Instance1": {
      "Type": "AWS::EC2::Instance",
      "Properties": {
        "InstanceType": {
          "Ref": "InstanceType"
        },
        "SecurityGroups": [
          "launch-wizard-2"
        ],
        "KeyName": {
          "Ref": "KeyName"
        },
        "Tags": [
          {
            "Key": "Name",
            "Value": "Server1"
          }
        ],
        "ImageId": "ami-0900fe555666598a2",
        "UserData": "IyFiaW4vYmFzaApkbyYgaw5zdGFsbCBuZ2lueCAteQpleHBvcnQgSE9TVE5BTUU9JChob3N0bmFtZSAteZikKc2VkIC1pICJzL1dlbGNvbWUgdG8gbmdpbngvT3dhaXMgTmdpbnggU2VydmlVY0iAkSE9TVE5BTUUvZyIgLS9Vzci9zaGFyZS9uZ2lueC9odG1sL2luZGV4Lmhh0bWwKc3lzZdGVtY3R5IHN0YXJ0IG5naW54CnN5c3RlbWNB0bCBlbmFibGUgbmdpbng="
      }
    },
    "EC2Instance2": {
      "Type": "AWS::EC2::Instance",
      "Properties": {
        "InstanceType": {
          "Ref": "InstanceType"
        },
        "SecurityGroups": [
          "launch-wizard-2"
        ],
        "KeyName": {
          "Ref": "KeyName"
        },
        "Tags": [
          {
            "Key": "Name",
            "Value": "Server2"
          }
        ]
      }
    }
  }
}
```

```

    }
  ],
  "ImageId": "ami-0900fe555666598a2",
  "UserData":
"IyFiaW4vYmFzaApkbmYgaW5zdGFsbCBuZ2lueCAtQpleHBvcnQgSE9TVE5BTUU9JChob3N0bmFtZSAtdlBGNvbWUgdG8gbmdpbngvT3dhaXMgTmdpbnggU2VydMvY0iAkSE9TVE5BTUUvZyIgL3Vzci9zaGFyZS9uZ2lueC9odG1sL2luZGV4Lmh0bWwKc3lzdGVtY3R5IHNOYXJ0IG5naW54CnN5c3RlbWN0bCBlbmFibGUgbmdpbng="
}
},
"EC2Instance3": {
  "Type": "AWS::EC2::Instance",
  "Properties": {
    "InstanceType": {
      "Ref": "InstanceType"
    },
    "SecurityGroups": [
      "launch-wizard-2"
    ],
    "KeyName": {
      "Ref": "KeyName"
    },
    "Tags": [
      {
        "Key": "Name",
        "Value": "Server3"
      }
    ]
  },
  "ImageId": "ami-0900fe555666598a2",
  "UserData":
"IyFiaW4vYmFzaApkbmYgaW5zdGFsbCBuZ2lueCAtQpleHBvcnQgSE9TVE5BTUU9JChob3N0bmFtZSAtdlBGNvbWUgdG8gbmdpbngvT3dhaXMgTmdpbnggU2VydMvY0iAkSE9TVE5BTUUvZyIgL3Vzci9zaGFyZS9uZ2lueC9odG1sL2luZGV4Lmh0bWwKc3lzdGVtY3R5IHNOYXJ0IG5naW54CnN5c3RlbWN0bCBlbmFibGUgbmdpbng="
}
},
"EC2Instance4": {
  "Type": "AWS::EC2::Instance",
  "Properties": {
    "InstanceType": {
      "Ref": "InstanceType"
    },
    "SecurityGroups": [
      "launch-wizard-2"
    ],
    "KeyName": {
      "Ref": "KeyName"
    },
    "Tags": [
      {
        "Key": "Name",
        "Value": "Server4"
      }
    ]
  },
  "ImageId": "ami-0900fe555666598a2",
  "UserData":
"IyFiaW4vYmFzaApkbmYgaW5zdGFsbCBuZ2lueCAtQpleHBvcnQgSE9TVE5BTUU9JChob3N0bmFtZSAtdlBGNvbWUgdG8gbmdpbngvT3dhaXMgTmdpbnggU2VydMvY0iAkSE9TVE5BTUUvZyIgL3Vzci9zaGFyZS9uZ2lueC9odG1sL2luZGV4Lmh0bWwKc3lzdGVtY3R5IHNOYXJ0IG5naW54CnN5c3RlbWN0bCBlbmFibGUgbmdpbng="
}
},
"EC2Instance5": {
  "Type": "AWS::EC2::Instance",
  "Properties": {
    "InstanceType": {
      "Ref": "InstanceType"
    }
  }
}

```



```

    },
    "SecurityGroups": [
        "launch-wizard-2"
    ],
    "KeyName": {
        "Ref": "KeyName"
    },
    "Tags": [
        {
            "Key": "Name",
            "Value": "LoadBalancer"
        }
    ],
    "ImageId": "ami-0900fe555666598a2"
}
}
},
"Outputs": {
    "PublicDNSServerOne": {
        "Description": "Public DNSName of the newly created EC2 instance",
        "Value": {
            "Fn::GetAtt": [
                "EC2Instance1",
                "PublicDnsName"
            ]
        }
    },
    "PublicIPServerOne": {
        "Description": "Public IP address of the newly created EC2 instance",
        "Value": {
            "Fn::GetAtt": [
                "EC2Instance1",
                "PublicIp"
            ]
        }
    },
    "PublicDNSServerTwo": {
        "Description": "Public DNSName of the newly created EC2 instance",
        "Value": {
            "Fn::GetAtt": [
                "EC2Instance2",
                "PublicDnsName"
            ]
        }
    },
    "PublicIPServerTwo": {
        "Description": "Public IP address of the newly created EC2 instance",
        "Value": {
            "Fn::GetAtt": [
                "EC2Instance2",
                "PublicIp"
            ]
        }
    },
    "PublicDNSServerThree": {
        "Description": "Public DNSName of the newly created EC2 instance",
        "Value": {
            "Fn::GetAtt": [
                "EC2Instance3",
                "PublicDnsName"
            ]
        }
    },
    "PublicIPServerThree": {

```

```

        "Description": "Public IP address of the newly created EC2 instance",
        "Value": {
            "Fn::GetAtt": [
                "EC2Instance3",
                "PublicIp"
            ]
        }
    },
    "PublicDNSServerFour": {
        "Description": "Public DNSName of the newly created EC2 instance",
        "Value": {
            "Fn::GetAtt": [
                "EC2Instance4",
                "PublicDnsName"
            ]
        }
    },
    "PublicIPServerFour": {
        "Description": "Public IP address of the newly created EC2 instance",
        "Value": {
            "Fn::GetAtt": [
                "EC2Instance4",
                "PublicIp"
            ]
        }
    },
    "PublicDNSServerFive": {
        "Description": "Public DNSName of the newly created EC2 instance",
        "Value": {
            "Fn::GetAtt": [
                "EC2Instance5",
                "PublicDnsName"
            ]
        }
    },
    "PublicIPServerFive": {
        "Description": "Public IP address of the newly created EC2 instance",
        "Value": {
            "Fn::GetAtt": [
                "EC2Instance5",
                "PublicIp"
            ]
        }
    }
}
}

```

UserData Script is mentioned in base64 encoded:

```

#!/bin/bash
dnf install nginx -y
export HOSTNAME=$(hostname -f)
sed -i "s/Welcome to nginx/Owais Nginx $HOSTNAME/g" /usr/share/nginx/index.html
systemctl start nginx
systemctl enable nginx

```

HTML Code for website:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Owais Nginx Server: ip-172-31-16-11.us-east-2.compute.internal!</title>
    <style>
      html {
        color-scheme: light dark;
      }
      body {
        width: 35em;
        margin: 0 auto;
        font-family: Tahoma, Verdana, Arial, sans-serif;
      }
    </style>
  </head>
  <body>
    <h1>Owais Nginx Server: ip-172-31-16-11.us-east-2.compute.internal!</h1>
    <p>If you see this page, the nginx web server is successfully installed and
    working. Further configuration is required.</p>

    <p>For online documentation and support please refer to
    <a href="http://nginx.org/">nginx.org</a>.<br/>
    Commercial support is available at
    <a href="http://nginx.com/">nginx.com</a>.</p>

    <p><em>Thank you for using nginx.</em></p>
  </body>
</html>
```