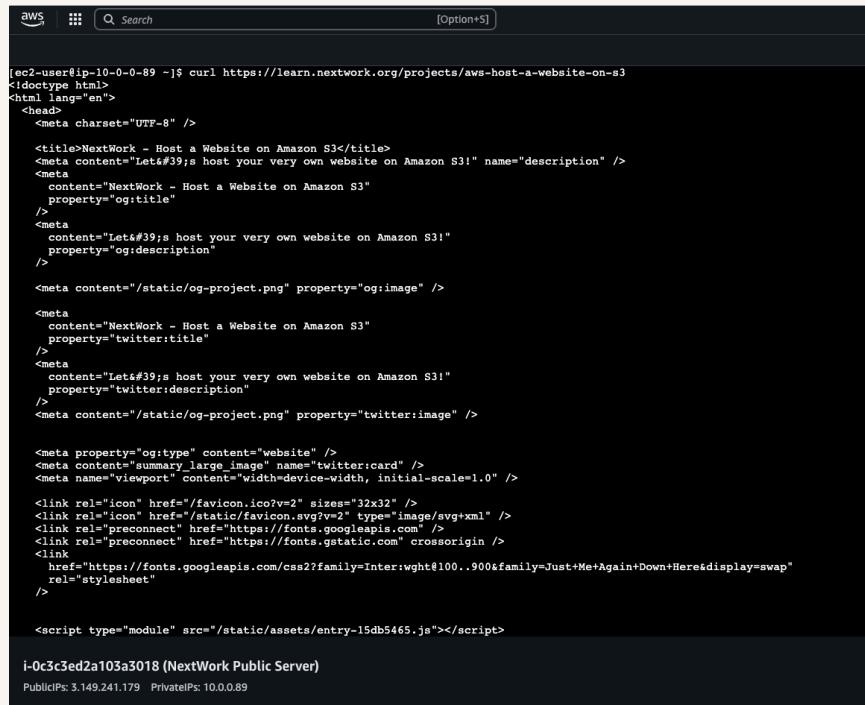




# Testing VPC Connectivity

R

Roshan Shrestha



```
[ec2-user@ip-10-0-0-89 ~]$ curl https://learn.nextwork.org/projects/aws-host-a-website-on-s3
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <title>NextWork - Host a Website on Amazon S3</title>
    <meta content="Let's host your very own website on Amazon S3!" name="description" />
    <meta content="NextWork - Host a Website on Amazon S3" property="og:title" />
    <meta content="Let's host your very own website on Amazon S3!" property="og:description" />
    <meta content="/static/og-project.png" property="og:image" />
    <meta content="NextWork - Host a Website on Amazon S3" property="twitter:title" />
    <meta content="Let's host your very own website on Amazon S3!" property="twitter:description" />
    <meta content="/static/og-project.png" property="twitter:image" />
    <meta property="og:type" content="website" />
    <meta content="summary_large_image" name="twitter:card" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
<link rel="icon" href="/favicon.ico?v=2" sizes="32x32" />
<link rel="icon" href="/static/favicon.svg?v=2" type="image/svg+xml" />
<link rel="preconnect" href="https://fonts.googleapis.com" />
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin />
<link href="https://fonts.googleapis.com/css2?family=Inter:wght@100..900&family=Just+Me+Again+Down+Here&display=swap" rel="stylesheet" />
<script type="module" src="/static/assets/entry-15db5465.js"></script>
```

i-0c3c3ed2a103a3018 (NextWork Public Server)  
PublicIPs: 3.149.241.179 PrivateIPs: 10.0.0.89



# Introducing Today's Project!

## What is Amazon VPC?

Amazon VPC is a virtual space allocated to us in the whole Amazon cloudspace. It is useful since it helps us be unique and be found easily, similar to an address in a zip code in a city.

## How I used Amazon VPC in this project

In today's project, I used Amazon VPC to launch a public and a private EC2 instance. Then I used tools like ping and curl to check network connectivity between the two instances and between the public instance and the internet as well.

## One thing I didn't expect in this project was...

Getting to troubleshoot setup configs like the NACL and security group rules.

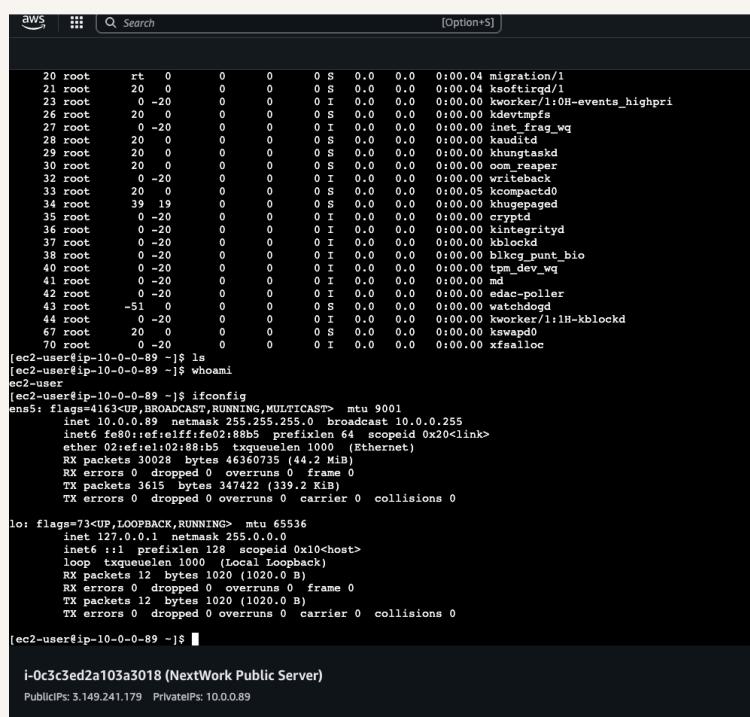
## This project took me...

A little over an hour.

# Connecting to an EC2 Instance

Connectivity means how well different parts of your network talk to each other and with other external networks.

My first connectivity test was whether I could connect to the NextWork Public Server



The screenshot shows a terminal window with the AWS logo at the top. The terminal displays several commands and their outputs:

```
[ec2-user@ip-10-0-0-89 ~]$ ls
[ec2-user@ip-10-0-0-89 ~]$ whoami
ec2-user
[ec2-user@ip-10-0-0-89 ~]$ ifconfig
ens5: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9001
        inet 10.0.0.89  netmask 255.255.255.0  broadcast 10.0.0.255
                inet6 fe80::efef:fffe%eth0:  prefixlen 64  scopid 0x20<link>
                    ether 02:ef:el:02:88:b5  txqueuelen 1000  (Ethernet)
                    RX packets 30028  bytes 46360735 (44.2 MiB)
                    RX errors 0  dropped 0  overruns 0  frame 0
                    TX packets 3615  bytes 347422 (339.2 KiB)
                    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
                inet6 ::1  prefixlen 128  scopid 0x10<host>
                    loop  txqueuelen 1000  (Local Loopback)
                    RX packets 12  bytes 1020 (1020.0 B)
                    RX errors 0  dropped 0  overruns 0  frame 0
                    TX packets 12  bytes 1020 (1020.0 B)
                    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
[ec2-user@ip-10-0-0-89 ~]$ 

i-0c3c3ed2a103a3018 (NextWork Public Server)
PublicIPs: 3.149.241.179  PrivateIPs: 10.0.0.89
```

# EC2 Instance Connect

I connected to my EC2 instance using EC2 Instance Connect, which is a shortcut way to get direct SSH access to your EC2 instance.

My first attempt at getting direct access to my public server resulted in an error, because the security group being used only had HTTP requests enabled whereas we use SSH for direct access.

I fixed this error by adding a rule for accepting SSH requests to the public security group for the public EC2 instance in the public subnet.



# Connectivity Between Servers

Ping is a common computer network tool used to check whether your computer can communicate with another computer or device on a network. I used ping to test the connectivity between my public and private EC2 instances.

The ping command. I ran was "ping 10.0.1.157" where 10.0.1.157 is my private EC2 instances ip address.

The first ping returned in no results. This meant my ping request was not reaching its destination (the private EC2 instance in this case) or the response from the destination could also have been blocked from reaching me (public EC2 instance).

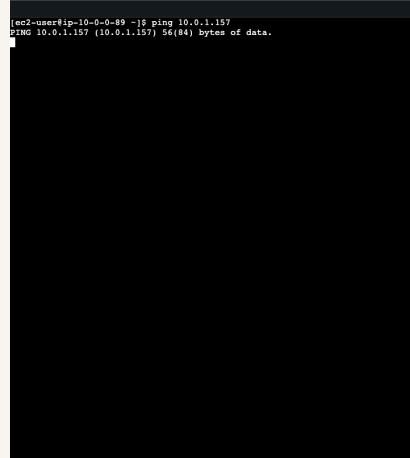
R

**Roshan Shrestha**  
NextWork Student

[nextwork.org](http://nextwork.org)

```
[ec2-user@ip-10-0-0-89 ~]$ ping 10.0.1.157
PING 10.0.1.157 (10.0.1.157) 56(84) bytes of data.

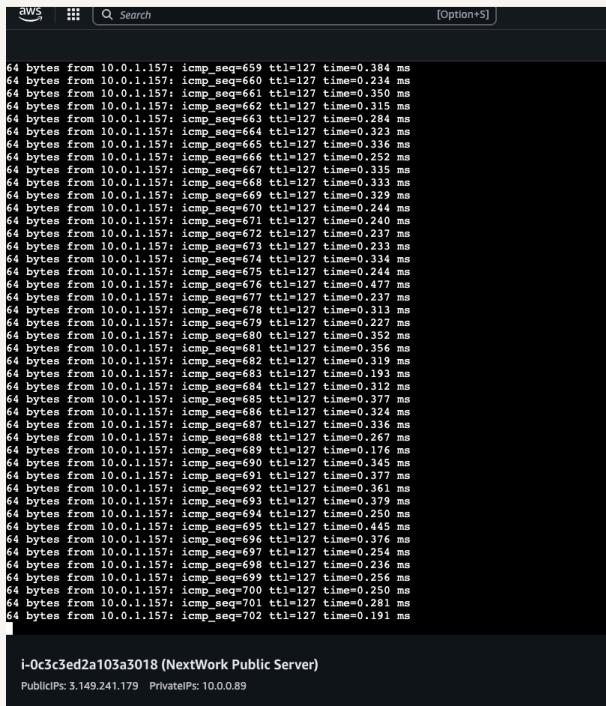
```



```
i-0c3c3ed2a103a3018 (NextWork Public Server)
PublicIP: 5.149.241.179 PrivateIP: 10.0.0.89
```

# Troubleshooting Connectivity

I troubleshooted this by looking into the Private NACL for the private EC2 instance. Added both inbound and outbound rules to allow ping messages (ICMP ipv4). Also added inbound rule for ping messages to the private security group.



A screenshot of a terminal window showing a log of ICMP traffic. The log consists of numerous entries, each showing a packet from source IP 10.0.1.157 to destination IP 10.0.1.157. The entries are timestamped and show varying sequence numbers (seq) and times. The log ends with the identifier "i-0c3c3ed2a103a3018 (NextWork Public Server)" and the message "PublicIPs: 3.149.241.179 PrivateIPs: 10.0.0.89".

```
64 bytes from 10.0.1.157: icmp_seq=659 ttl=127 time=0.384 ms
64 bytes from 10.0.1.157: icmp_seq=660 ttl=127 time=0.234 ms
64 bytes from 10.0.1.157: icmp_seq=661 ttl=127 time=0.350 ms
64 bytes from 10.0.1.157: icmp_seq=662 ttl=127 time=0.315 ms
64 bytes from 10.0.1.157: icmp_seq=663 ttl=127 time=0.284 ms
64 bytes from 10.0.1.157: icmp_seq=664 ttl=127 time=0.323 ms
64 bytes from 10.0.1.157: icmp_seq=665 ttl=127 time=0.336 ms
64 bytes from 10.0.1.157: icmp_seq=666 ttl=127 time=0.252 ms
64 bytes from 10.0.1.157: icmp_seq=667 ttl=127 time=0.334 ms
64 bytes from 10.0.1.157: icmp_seq=668 ttl=127 time=0.333 ms
64 bytes from 10.0.1.157: icmp_seq=669 ttl=127 time=0.329 ms
64 bytes from 10.0.1.157: icmp_seq=670 ttl=127 time=0.244 ms
64 bytes from 10.0.1.157: icmp_seq=671 ttl=127 time=0.240 ms
64 bytes from 10.0.1.157: icmp_seq=672 ttl=127 time=0.237 ms
64 bytes from 10.0.1.157: icmp_seq=673 ttl=127 time=0.233 ms
64 bytes from 10.0.1.157: icmp_seq=674 ttl=127 time=0.334 ms
64 bytes from 10.0.1.157: icmp_seq=675 ttl=127 time=0.244 ms
64 bytes from 10.0.1.157: icmp_seq=676 ttl=127 time=0.477 ms
64 bytes from 10.0.1.157: icmp_seq=677 ttl=127 time=0.237 ms
64 bytes from 10.0.1.157: icmp_seq=678 ttl=127 time=0.314 ms
64 bytes from 10.0.1.157: icmp_seq=679 ttl=127 time=0.297 ms
64 bytes from 10.0.1.157: icmp_seq=680 ttl=127 time=0.352 ms
64 bytes from 10.0.1.157: icmp_seq=681 ttl=127 time=0.356 ms
64 bytes from 10.0.1.157: icmp_seq=682 ttl=127 time=0.319 ms
64 bytes from 10.0.1.157: icmp_seq=683 ttl=127 time=0.193 ms
64 bytes from 10.0.1.157: icmp_seq=684 ttl=127 time=0.312 ms
64 bytes from 10.0.1.157: icmp_seq=685 ttl=127 time=0.377 ms
64 bytes from 10.0.1.157: icmp_seq=686 ttl=127 time=0.324 ms
64 bytes from 10.0.1.157: icmp_seq=687 ttl=127 time=0.336 ms
64 bytes from 10.0.1.157: icmp_seq=688 ttl=127 time=0.267 ms
64 bytes from 10.0.1.157: icmp_seq=689 ttl=127 time=0.345 ms
64 bytes from 10.0.1.157: icmp_seq=690 ttl=127 time=0.345 ms
64 bytes from 10.0.1.157: icmp_seq=691 ttl=127 time=0.377 ms
64 bytes from 10.0.1.157: icmp_seq=692 ttl=127 time=0.361 ms
64 bytes from 10.0.1.157: icmp_seq=693 ttl=127 time=0.379 ms
64 bytes from 10.0.1.157: icmp_seq=694 ttl=127 time=0.250 ms
64 bytes from 10.0.1.157: icmp_seq=695 ttl=127 time=0.445 ms
64 bytes from 10.0.1.157: icmp_seq=696 ttl=127 time=0.376 ms
64 bytes from 10.0.1.157: icmp_seq=697 ttl=127 time=0.254 ms
64 bytes from 10.0.1.157: icmp_seq=698 ttl=127 time=0.236 ms
64 bytes from 10.0.1.157: icmp_seq=699 ttl=127 time=0.256 ms
64 bytes from 10.0.1.157: icmp_seq=700 ttl=127 time=0.250 ms
64 bytes from 10.0.1.157: icmp_seq=701 ttl=127 time=0.281 ms
64 bytes from 10.0.1.157: icmp_seq=702 ttl=127 time=0.191 ms
```

i-0c3c3ed2a103a3018 (NextWork Public Server)  
PublicIPs: 3.149.241.179 PrivateIPs: 10.0.0.89



# Connectivity to the Internet

Curl is a tool to test connectivity in a network. Where ping checks if one computer can contact another (and how long messages take to travel back and forth), curl is used to transfer data to or from a server.

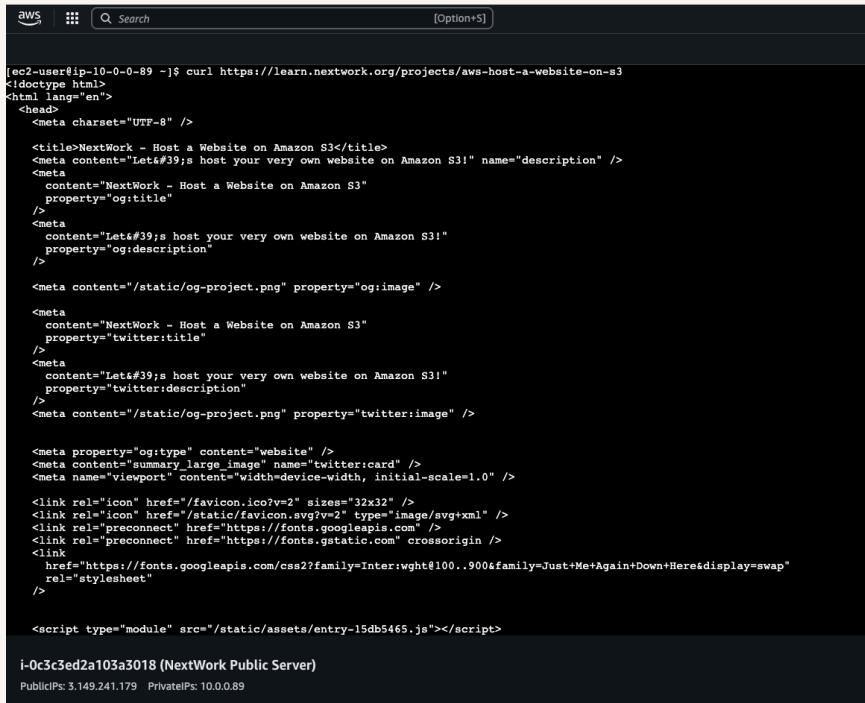
I used curl to test the connectivity between my public EC2 instance and the internet (google.com, learn.nextwork.org, etc.)

## Ping vs Curl

Ping and curl are different because curl is a tool to test connectivity in a network whereas ping checks if one computer can contact another (and how long messages take to travel back and forth).

# Connectivity to the Internet

I ran the curl command "curl learn.nextwork.org" which returned the HTML content of that website.



```
[ec2-user@ip-10-0-0-89 ~]$ curl https://learn.nextwork.org/projects/aws-host-a-website-on-s3
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>NextWork - Host a Website on Amazon S3</title>
    <meta content="Let's host your very own website on Amazon S3!" name="description" />
    <meta content="NextWork - Host a Website on Amazon S3" property="og:title" />
    <meta content="Let's host your very own website on Amazon S3!" property="og:description" />
    <meta content="/static/og-project.png" property="og:image" />
    <meta content="NextWork - Host a Website on Amazon S3" property="twitter:title" />
    <meta content="Let's host your very own website on Amazon S3!" property="twitter:description" />
    <meta content="/static/og-project.png" property="twitter:image" />
    <meta property="og:type" content="website" />
    <meta content="summary_large_image" name="twitter:card" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="icon" href="/favicon.ico?v=2" sizes="32x32" />
    <link rel="icon" href="/static/favicon.svg?v=2" type="image/svg+xml" />
    <link rel="preconnect" href="https://fonts.googleapis.com" />
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin />
    <link href="https://fonts.googleapis.com/css2?family=Inter:wght@100..900&family=Just+Me+Again+Down+Here&display=swap" rel="stylesheet" />
    <script type="module" src="/static/assets/entry-15db5465.js"></script>

```

i-0c3c3ed2a103a3018 (NextWork Public Server)  
PublicIPs: 3.149.241.179 PrivateIPs: 10.0.0.89



[nextwork.org](https://nextwork.org)

# The place to learn & showcase your skills

Check out [nextwork.org](https://nextwork.org) for more projects

