

Introducing Today's Project!

What is Amazon VPC?

Amazon VPC is AWS's core networking service that enables resources to be private or public to the internet. It also allows setting up custom traffic flow and security rules, providing control over the network infrastructure.

How I used Amazon VPC in this project

In this project, I used Amazon VPC and its components via the VPC wizard, launched EC2 instances, and tested connectivity between resources within the network to ensure proper communication.

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

There was an unexpected challenge while troubleshooting connectivity issues related to security group and Network ACL configurations. It required careful attention to inbound and outbound rules to ensure proper traffic flow.

This project took me...

This project took me 2 hours to complete.

Connecting to an EC2 Instance

Connectivity refers to the ability of resources within a network to communicate and exchange data effectively. Without connectivity, resources cannot interact, resulting in issues like users being unable to access applications or services.

My first connectivity test was whether I could connect to my network's Public Server (an EC2 instance).

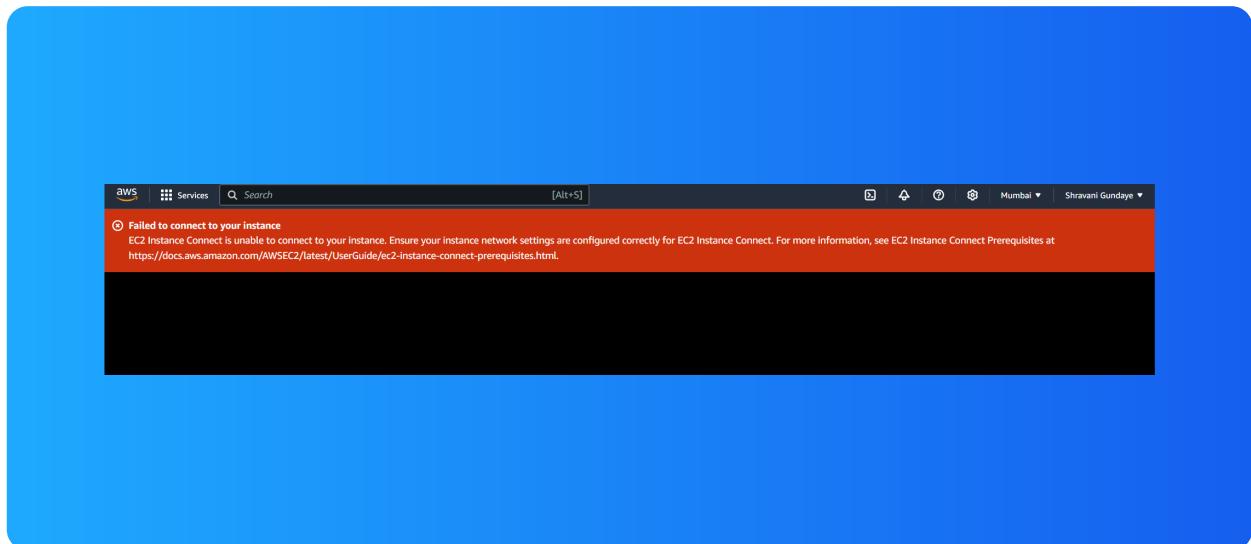


EC2 Instance Connect

I connected to my EC2 instance using EC2 Instance Connect, which is a tool that allows us to directly access an EC2 instance using the AWS Management Console. We no longer need to manage key pairs, or use an SSH client to connect to our EC2 instance.

My first attempt at getting direct access to my public server resulted in an error, because my Private Server had a security group that did not allow SSH traffic - it only allowed HTTP traffic i.e. a different protocol.

I fixed this error by adding a new inbound rule in my Private Server's security group that allows SSH traffic from anywhere.

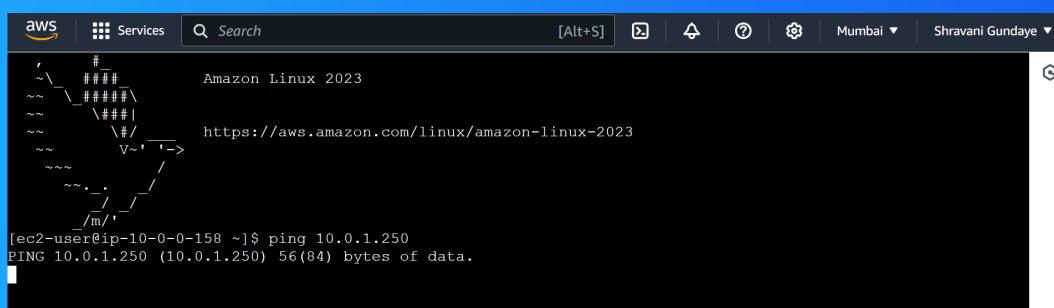


Connectivity Between Servers

Ping is a tool to test the connectivity between two servers and also the response time (i.e. the performance of the connection). I used ping to test the connectivity between my Public and Private Servers.

The ping command I ran was 'ping 10.0.1.250' where 10.0.1.250 is the private IPv4 address of my Private Server.

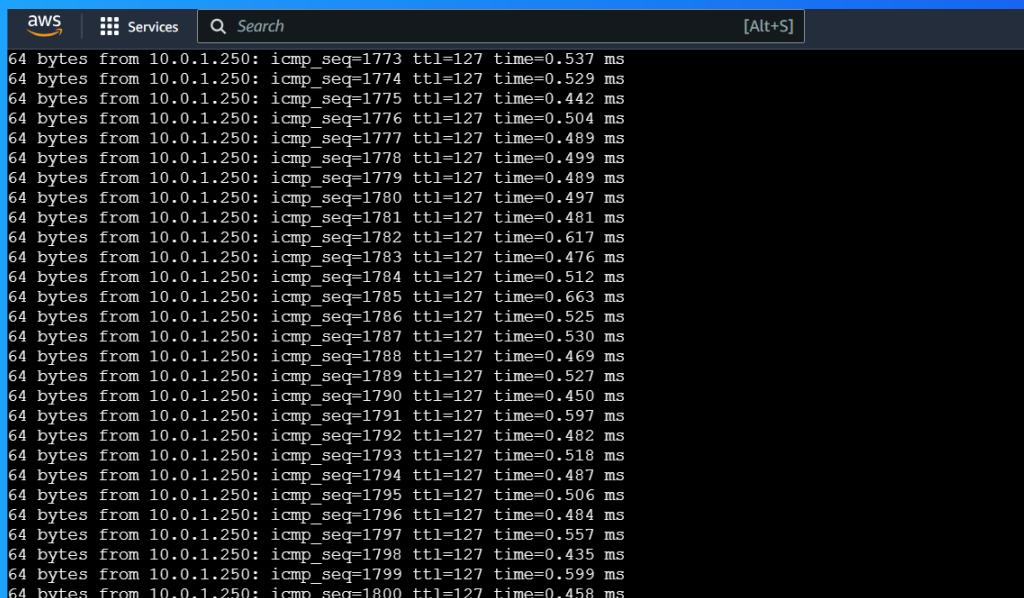
The first ping returned NO replies from the Private Server. This meant security settings with my private server was blocking inbound (and or outbound) ICMP traffic, which is the traffic type of ping messages.



```
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023
[ec2-user@ip-10-0-0-158 ~]$ ping 10.0.1.250
PING 10.0.1.250 (10.0.1.250) 56(84) bytes of data.
```

Troubleshooting Connectivity

I troubleshooted this by enabling ICMP traffic in my private server's network ACLs and security group rules.



A screenshot of a AWS CloudWatch Log Stream window. The window has a dark theme with white text. At the top, there are tabs for 'aws' (highlighted in orange), 'Services' (with a grid icon), and 'Search' (with a magnifying glass icon). To the right of the search bar is the text '[Alt+S]'. The main area contains a log of ICMP traffic. Each log entry consists of '64 bytes from 10.0.1.250:' followed by an 'icmp_seq' value ranging from 1773 to 1800, 'ttl=127', and a 'time' value in milliseconds (e.g., 0.537 ms, 0.529 ms, etc.).

```
64 bytes from 10.0.1.250: icmp_seq=1773 ttl=127 time=0.537 ms
64 bytes from 10.0.1.250: icmp_seq=1774 ttl=127 time=0.529 ms
64 bytes from 10.0.1.250: icmp_seq=1775 ttl=127 time=0.442 ms
64 bytes from 10.0.1.250: icmp_seq=1776 ttl=127 time=0.504 ms
64 bytes from 10.0.1.250: icmp_seq=1777 ttl=127 time=0.489 ms
64 bytes from 10.0.1.250: icmp_seq=1778 ttl=127 time=0.499 ms
64 bytes from 10.0.1.250: icmp_seq=1779 ttl=127 time=0.489 ms
64 bytes from 10.0.1.250: icmp_seq=1780 ttl=127 time=0.497 ms
64 bytes from 10.0.1.250: icmp_seq=1781 ttl=127 time=0.481 ms
64 bytes from 10.0.1.250: icmp_seq=1782 ttl=127 time=0.617 ms
64 bytes from 10.0.1.250: icmp_seq=1783 ttl=127 time=0.476 ms
64 bytes from 10.0.1.250: icmp_seq=1784 ttl=127 time=0.512 ms
64 bytes from 10.0.1.250: icmp_seq=1785 ttl=127 time=0.663 ms
64 bytes from 10.0.1.250: icmp_seq=1786 ttl=127 time=0.525 ms
64 bytes from 10.0.1.250: icmp_seq=1787 ttl=127 time=0.530 ms
64 bytes from 10.0.1.250: icmp_seq=1788 ttl=127 time=0.469 ms
64 bytes from 10.0.1.250: icmp_seq=1789 ttl=127 time=0.527 ms
64 bytes from 10.0.1.250: icmp_seq=1790 ttl=127 time=0.450 ms
64 bytes from 10.0.1.250: icmp_seq=1791 ttl=127 time=0.597 ms
64 bytes from 10.0.1.250: icmp_seq=1792 ttl=127 time=0.482 ms
64 bytes from 10.0.1.250: icmp_seq=1793 ttl=127 time=0.518 ms
64 bytes from 10.0.1.250: icmp_seq=1794 ttl=127 time=0.487 ms
64 bytes from 10.0.1.250: icmp_seq=1795 ttl=127 time=0.506 ms
64 bytes from 10.0.1.250: icmp_seq=1796 ttl=127 time=0.484 ms
64 bytes from 10.0.1.250: icmp_seq=1797 ttl=127 time=0.557 ms
64 bytes from 10.0.1.250: icmp_seq=1798 ttl=127 time=0.435 ms
64 bytes from 10.0.1.250: icmp_seq=1799 ttl=127 time=0.599 ms
64 bytes from 10.0.1.250: icmp_seq=1800 ttl=127 time=0.458 ms
```

Connectivity to the Internet

Curl is a connectivity tool used for testing network connectivity and transferring data between servers. It can be employed to retrieve data from various web servers.

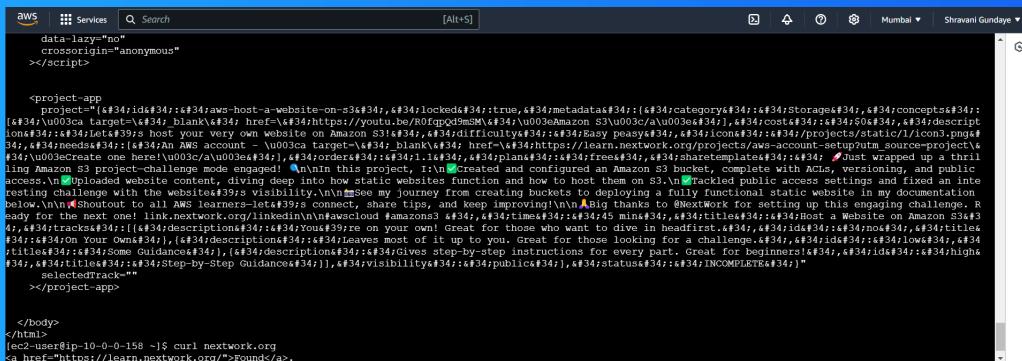
I used curl to test the connectivity between my network's Public Server with the public internet. A successful connectivity test indicates that the internet gateway, network ACLs, security groups, and route tables are configured correctly.

Ping vs Curl

Ping and curl differ in their responses: ping provides a report on the connectivity performance between servers (e.g. public to private), while curl retrieves data (e.g. HTML) from a target server, such as another public server.

Connectivity to the Internet

I ran the curl command 'curl https://learn.nextwork.org/projects/aws-host-a-website-on-s3' which returned the HTML content of NextWork's first project guide.



A screenshot of a browser window displaying the source code of a web page. The browser interface includes tabs for AWS, Services, and Search, along with a status bar showing 'Mumbai' and 'Shravani Gunday'. The main content area shows a large amount of HTML code, starting with a script tag and ending with a body tag containing a href link to 'https://learn.nextwork.org/'. The code is heavily escaped with numerous backslashes and double quotes, making it difficult to read directly but representing the raw HTML output of the curl command.



NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

