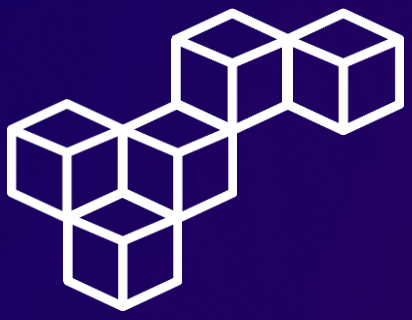
AWS re:Start LAB

Troubleshooting an Amazon VPC

WEEK 10

aws re/start



Overview

Troubleshooting an Amazon Virtual Private Cloud (VPC) involves diagnosing and resolving issues related to network configurations, connectivity, and resource accessibility. Key areas to review include route tables, security groups, and Network ACLs to ensure they allow necessary traffic. Common problems often stem from misconfigured routes, such as missing routes to an internet gateway for public subnets or a NAT gateway for private subnets. Verifying that security groups and Network ACLs are not blocking required inbound or outbound traffic is also crucial.

Additionally, checking connectivity between instances and other AWS services is essential. This includes ensuring instances are in the correct subnets, elastic IP addresses are properly assigned, and there are no IP address conflicts. If using VPN connections or Direct Connect, verify their configurations and statuses. Tools like AWS CloudWatch logs, VPC Flow Logs, and network reachability tools can help identify traffic flow issues and pinpoint connectivity problems.

Topics covered

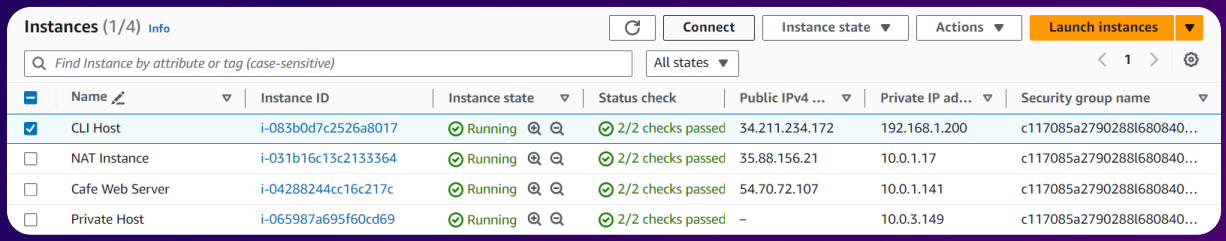
- Creating an Amazon Simple Storage Service (Amazon S3) bucket to hold VPC Flow Log data
- Creating a flow log to capture all IP traffic passing through network interfaces in the VPC
- Troubleshooting the VPC configuration issues to allow access to the resources
- Downloading and analyzing the flow log data

Task 1

Connecting to the CLI Host instance

Step 1: Connect to the CLI Host instance

On the EC2 Management Console, navigate to the **Instances** section, select the **CLI Host** instance, and connect to the instance using EC2 Instance Connect.



Instances (1/4) Info

Connect | Instance state | Actions | Launch instances

Find Instance by attribute or tag (case-sensitive) | All states | 1

Name	Instance ID	Instance state	Status check	Public IPv4 ...	Private IP ad...	Security group name
CLI Host	i-083b0d7c2526a8017	Running	2/2 checks passed	34.211.234.172	192.168.1.200	c117085a2790288l680840...
NAT Instance	i-031b16c13c2133364	Running	2/2 checks passed	35.88.156.21	10.0.1.17	c117085a2790288l680840...
Cafe Web Server	i-04288244cc16c217c	Running	2/2 checks passed	54.70.72.107	10.0.1.141	c117085a2790288l680840...
Private Host	i-065987a695f60cd69	Running	2/2 checks passed	–	10.0.3.149	c117085a2790288l680840...

Step 2: Configure the AWS CLI

To configure the AWS CLI profile with credentials, in the EC2 Instance Connect terminal, run the aws configure command. At the prompts, enter the following information.

```
[ec2-user@cli-host ~]$ aws configure
AWS Access Key ID [None]: AKIA3FLD5ETGYRWBMRH4
AWS Secret Access Key [None]: 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
Default region name [None]: us-west-2
Default output format [None]: json
[ec2-user@cli-host ~]$
```

aws re/start

Task 2

Creating VPC Flow Logs

Step 1: Create the S3 bucket

To create the S3 bucket where the flow logs will be published, run the following aws s3api create-bucket command.

```
[ec2-user@cli-host ~]$ aws s3api create-bucket --bucket flowlog181017 \
> --region 'us-west-2' \
> --create-bucket-configuration LocationConstraint='us-west-2'
{
    "Location": "http://flowlog181017.s3.amazonaws.com/"
}
[ec2-user@cli-host ~]$
```

Step 2: Get the VPC ID

To get the VPC ID for VPC1 to create VPC Flow Logs, run the following aws ec2 describe-vpcs command.

```
[ec2-user@cli-host ~]$ aws ec2 describe-vpcs \
> --query 'Vpcs[*].[VpcId,Tags[?Key==`Name`].Value,CidrBlock]' \
> --filters "Name=tag:Name,Values='VPC1'"
[
    [
        "vpc-09a5d2a92b9bd2252",
        [
            "VPC1"
        ],
        "10.0.0.0/16"
    ]
]
[ec2-user@cli-host ~]$
```

aws re/start

Task 2

Creating VPC Flow Logs

Step 3: Create VPC Flow Logs

To create VPC Flow Logs on VPC1, run the following aws ec2 create-flow-logs command.

```
[ec2-user@cli-host ~]$ aws ec2 create-flow-logs \
> --resource-type VPC \
> --resource-ids vpc-09a5d2a92b9bd2252  \
> --traffic-type ALL \
> --log-destination-type s3 \
> --log-destination arn:aws:s3:::flowlog181017
{
    "Unsuccessful": [],
    "FlowLogIds": [
        "fl-0d55bd41bd349aea0"
    ],
    "ClientToken": "T4+hkUVCLu6Fn2rHJJm5Sc+CFxjbBD0G/tlT38ZKo9M="
}
[ec2-user@cli-host ~]$
```

Step 4: Confirm flow log creation

To confirm that the flow log was created, run the aws ec2 describe-flow-logs command.

```
[ec2-user@cli-host ~]$ aws ec2 describe-flow-logs
{
    "FlowLogs": [
        {
            "LogDestinationType": "s3",
            "Tags": [],
            "ResourceId": "vpc-09a5d2a92b9bd2252",
            "CreationTime": "2024-05-29T20:03:22.101Z",
            "TrafficType": "ALL",
            "FlowLogStatus": "ACTIVE",
            "LogFormat": "${version} ${account-id} ${interface-id} ${srcaddr} ${dstaddr} ${srcport}
 ${dstport} ${protocol} ${packets} ${bytes} ${start} ${end} ${action} ${log-status}",
            "FlowLogId": "fl-0d55bd41bd349aea0",
            "MaxAggregationInterval": 600,
            "LogDestination": "arn:aws:s3:::flowlog181017",
            "DeliverLogsStatus": "SUCCESS"
        }
    ]
}
[ec2-user@cli-host ~]$
```

aws re/start

Task 3

Troubleshooting VPC configuration issues to allow access to resources

Step 1: Find details about an instance

In the CLI Host terminal, to find details about the web server instance, run the following aws ec2 describe-instances command.

```
[ec2-user@cli-host ~]$ aws ec2 describe-instances \
> --filter "Name=ip-address,Values='54.70.72.107'"
{
    "Reservations": [
        {
            "Instances": [
                {
```

Step 2: Filter the results

To filter the results, run the following aws ec2 describe-instances command.

```
[ec2-user@cli-host ~]$ aws ec2 describe-instances \
> --filter "Name=ip-address,Values='54.70.72.107'" \
> --query 'Reservations[*].Instances[*].[State,PrivateIpAddress,InstanceId,SecurityGroups,SubnetId,KeyName]'
[
    [
        [
            {
                "Code": 16,
                "Name": "running"
            },
            "10.0.1.141",
            "i-04288244cc16c217c",
            [
                {
                    "GroupName": "c117085a279028816808407t1w767398061261-WebSecurityGroup-2HN5ETuG65Xh",
                    "GroupId": "sg-039b61712496fa5c1"
                }
            ],
            "subnet-0652e47641d560135",
            "vockey"
        ]
    ]
]
[ec2-user@cli-host ~]$
```

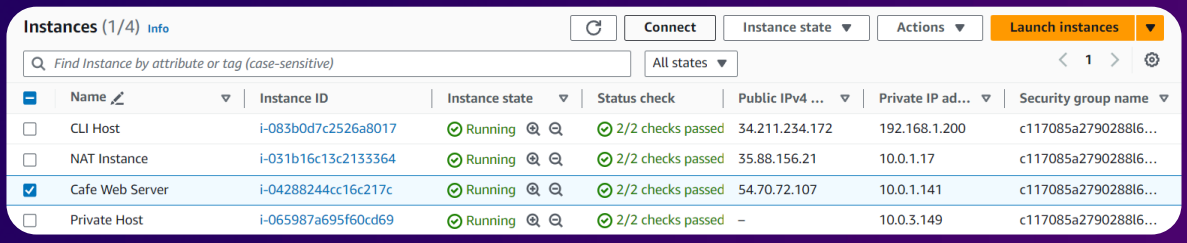
aws re/start

Task 3

Troubleshooting VPC configuration issues to allow access to resources

Step 3: Connect to the Cafe Web Server

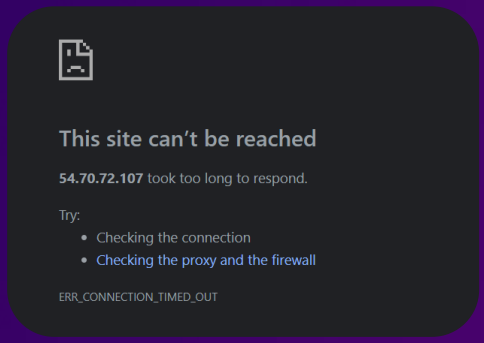
Attempt to connect to the **Cafe Web Server** instance using EC2 Instance Connect. The attempt to connect fails. You get an error that says Failed to connect to your instance.



Failed to connect to your instance
EC2 Instance Connect is unable to connect to your instance. Ensure your instance network settings are configured correctly for EC2 Instance Connect. For more information, see EC2 Instance Connect Prerequisites at https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-instance-connect-prerequisites.html.

Step 4: Access to the Cafe Website

Try to load the web server page. The page fails to load, and you receive a message indicating that the site can't be reached.



aws re/start

Task 3

Troubleshooting VPC configuration issues to allow access to resources

Step 5: Use the nmap utility

Use the nmap utility to check which ports are open on the web server EC2 instance. If nmap cannot find any open ports, there could be something else blocking access to the instance.

```
[ec2-user@cli-host ~]$ sudo yum install -y nmap
```

```
[ec2-user@cli-host ~]$ nmap 54.70.72.107

Starting Nmap 6.40 ( http://nmap.org ) at 2024-05-29 20:15 UTC
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.05 seconds
[ec2-user@cli-host ~]$
```

Step 6: Check the security group

Run the following aws ec2 describe-security-groups command. The security group settings that are applied to the web server EC2 instance are allowing connectivity to port 22 and port 80.

```
[ec2-user@cli-host ~]$ aws ec2 describe-security-groups \
> --group-ids 'sg-039b61712496fa5c1'
{
    {
        "IpPermissions": [
            {
                "FromPort": 80,
                "IpRanges": [
                    {
                        "CidrIp": "0.0.0.0/0"
                    }
                ],
                "ToPort": 80,
                "IpProtocol": "tcp",
            },
            {
                "FromPort": 22,
                "IpRanges": [
                    {
                        "CidrIp": "0.0.0.0/0"
                    }
                ],
                "ToPort": 22,
                "IpProtocol": "tcp",
            }
        ],
```

aws re/start

Task 3

Troubleshooting VPC configuration issues to allow access to resources

Step 7: Check the route table

Run the following aws ec2 describe-route-tables commands. There isn't a route directing internet traffic to an internet gateway.

```
[ec2-user@cli-host ~]$ aws ec2 describe-route-tables \
> --filter "Name=association.subnet-id,Values='subnet-0652e47641d560135'"
{
    "RouteTables": [
        {
            "Associations": [
                {
                    "SubnetId": "subnet-0652e47641d560135",
                    "AssociationState": {
                        "State": "associated"
                    },
                    "RouteTableAssociationId": "rtbassoc-0306284c14729b3ee",
                    "Main": false,
                    "RouteTableId": "rtb-0690a6279988f3342"
                }
            ],
            "RouteTableId": "rtb-0690a6279988f3342",
            "VpcId": "vpc-09a5d2a92b9bd2252",
            "Tags": [
                {
                    "Value": "VPC1 Public Route Table",
                    "Key": "Name"
                },
```

```
[ec2-user@cli-host ~]$ aws ec2 describe-route-tables \
> --route-table-ids 'rtb-0690a6279988f3342' \
> --filter "Name=association.subnet-id,Values='subnet-0652e47641d560135'"
    "RouteTables": [
        {
            "Routes": [
                {
                    "GatewayId": "local",
                    "DestinationCidrBlock": "10.0.0.0/16",
                    "State": "active",
                    "Origin": "CreateRouteTable"
            ],
```

Step 8: Add a route

Run the aws ec2 describe-internet-gateways command to get the gateway-id. Run the following aws ec2 create-route command to add a route to the internet gateway.

```
[ec2-user@cli-host ~]$ aws ec2 describe-internet-gateways
{
    "InternetGateways": [
                {
                    "Value": "VPC1 Internet Gateway",
                    "Key": "Name"
                },
            ],
            "Attachments": [
                {
                    "State": "available",
                    "VpcId": "vpc-09a5d2a92b9bd2252"
                }
            ],
            "InternetGatewayId": "igw-072f2b9c385246948"
        },
```

```
[ec2-user@cli-host ~]$ aws ec2 create-route \
> --route-table-id 'rtb-0690a6279988f3342' \
> --gateway-id  'igw-072f2b9c385246948' \
> --destination-cidr-block '0.0.0.0/0'
{
    "Return": true
}
[ec2-user@cli-host ~]$
```

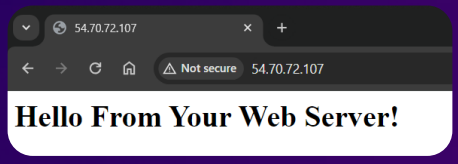
aws re/start

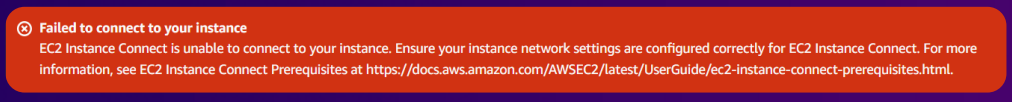
Task 3

Troubleshooting VPC configuration issues to allow access to resources

Step 9: Validate the solution

Try to load the web server page. It should display a message that says, "Hello From Your Web Server!". Try connecting to the **Cafe Web Server** instance using EC2 instance Connect. This attempt still fails.





Step 10: Check the network ACL

To check the network access control list settings, run the following aws ec2 describe-network-acls command. Rule Number 40 is denying inbound traffic on Port 22 (SSH).

```
[ec2-user@cli-host ~]$ aws ec2 describe-network-acls \
> --filter "Name=association.subnet-id,Values='subnet-0652e47641d560135'" \
> --query 'NetworkAcls[*].[NetworkAclId,Entries]'
[
    [
        "acl-0c854a830288b5be1",
        [
            {
            {
                "RuleNumber": 40,
                "Protocol": "6",
                "PortRange": {
                    "To": 22,
                    "From": 22
                },
                "Egress": false,
                "RuleAction": "deny",
                "CidrBlock": "0.0.0.0/0"
            },
```

aws re/start

Task 3

Troubleshooting VPC configuration issues to allow access to resources

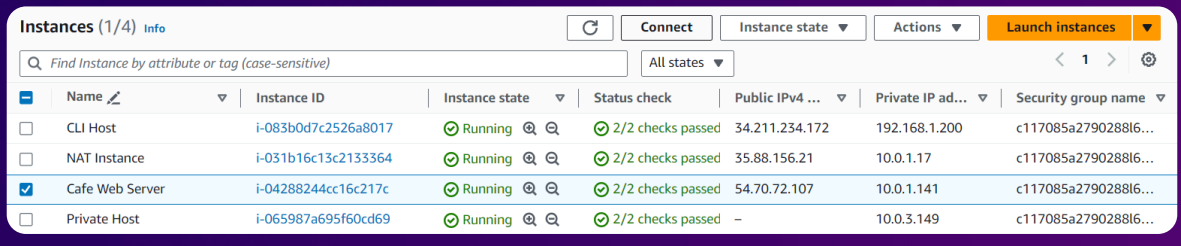
Step 11: Delete Network ACL entry

To delete the network ACL entry that is causing the issue, run the following aws ec2 delete-network-acl-entry command.

```
[ec2-user@cli-host ~]$ aws ec2 delete-network-acl-entry \
> --network-acl-id 'acl-0c854a830288b5be1' \
> --ingress \
> --rule-number 40
[ec2-user@cli-host ~]$
```

Step 12: Validate the solution

Connect to the **Cafe Web Server** instance using EC2 instance Connect. If you can connect, you have successfully resolved the issue. Run the hostname command after you are connected.




```
       _  #_
       ~\_  ####_        Amazon Linux 2
      ~~  \_#####\
      ~~     \###|       AL2 End of Life is 2025-06-30.
      ~~       \#/ ___
       ~~       V~' '->
        ~~~         /    A newer version of Amazon Linux is available!
          ~~._.   _/
             _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
           _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@web-server ~]$ hostname
web-server
[ec2-user@web-server ~]$
```

aws re/start

Task 4

Analyzing flow logs

Step 1: Downloading and extracting flow logs

Run the following commands to download and extract the flow log files.

```
[ec2-user@cli-host ~]$ mkdir flowlogs
[ec2-user@cli-host ~]$ cd flowlogs
[ec2-user@cli-host flowlogs]$ aws s3 ls
2024-05-29 20:03:23 flowlog181017
[ec2-user@cli-host flowlogs]$ aws s3 cp s3://flowlog181017/ . --recursive
download: s3://flowlog181017/AWSLogs/767398061261/vpcflowlogs/us-west-2/2024/05/29/767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2000Z_df16eb1f
.log.gz to AWSLogs/767398061261/vpcflowlogs/us-west-2/2024/05/29/767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2000Z_df16eb1f.log.gz
```

```
[ec2-user@cli-host flowlogs]$ cd AWSLogs/767398061261/vpcflowlogs/us-west-2/2024/05/29/
[ec2-user@cli-host 29]$ ls
767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2000Z_df16eb1f.log.gz
```

```
[ec2-user@cli-host 29]$ gunzip *.gz
[ec2-user@cli-host 29]$ ls
767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2000Z_df16eb1f.log
```

Step 2: Analyze the structure of the logs

Run the following head command to analyze the structure of the logs. The header row indicates the kind of data that each log entry contains. Each entry contains information, such as the IP address of the source of the event (in the fourth column), the destination port (seventh column), start and end timestamps (in Unix timestamp format), and the action that resulted (ACCEPT or REJECT).

```
[ec2-user@cli-host 29]$ head 767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2000Z_df16eb1f.log
version account-id interface-id srcaddr dstaddr srcport dstport protocol packets bytes start end action log-status
2 767398061261 eni-0a1469cce7034a10f 185.242.226.45 10.0.1.141 54111 52001 6 1 44 1717013041 1717013062 REJECT OK
2 767398061261 eni-0a1469cce7034a10f 152.42.198.222 10.0.1.141 52156 1006 6 1 44 1717013041 1717013062 REJECT OK
2 767398061261 eni-0a1469cce7034a10f 79.110.62.185 10.0.1.141 56968 11917 6 1 40 1717013041 1717013062 REJECT OK
2 767398061261 eni-0a1469cce7034a10f 162.216.149.220 10.0.1.141 55952 19222 6 1 44 1717013041 1717013062 REJECT OK
2 767398061261 eni-0a1469cce7034a10f 79.110.62.185 10.0.1.141 56968 12096 6 1 40 1717013041 1717013062 REJECT OK
2 767398061261 eni-0a1469cce7034a10f 162.216.149.12 10.0.1.141 56817 9095 6 1 44 1717013041 1717013062 REJECT OK
2 767398061261 eni-0a1469cce7034a10f 79.110.62.185 10.0.1.141 56968 55964 6 1 40 1717013041 1717013062 REJECT OK
2 767398061261 eni-0a1469cce7034a10f 172.105.36.98 10.0.1.141 48203 5777 6 1 44 1717013041 1717013062 REJECT OK
2 767398061261 eni-0a1469cce7034a10f 78.128.112.114 10.0.1.141 55683 59786 6 1 40 1717013005 1717013034 REJECT OK
[ec2-user@cli-host 29]$
```

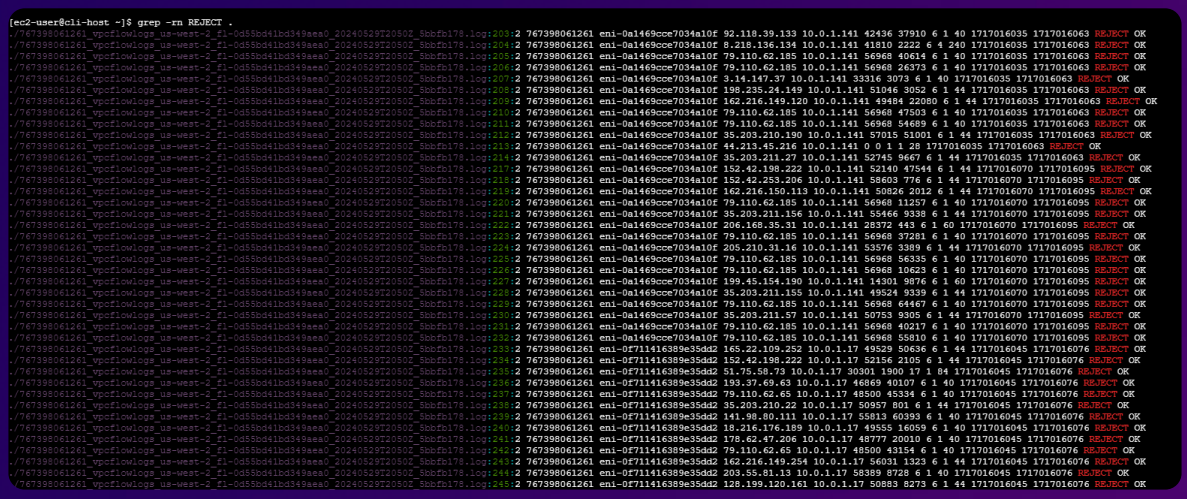
aws re/start

Task 4

Analyzing flow logs

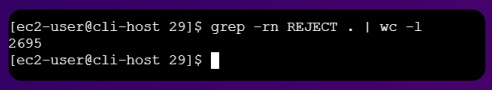
Step 3: Search each log file

To search each log file in the current directory and return lines that contain the word REJECT, run the following command. This command should return a large dataset because it includes every event where the VPC settings rejected the request.



Step 4: Count records

To find out how many records were returned, run the following command. The results show the number of lines in your result set.



aws re/start

Task 4

Analyzing flow logs

Step 5: Refine your search

To refine your search by looking for only lines that contain 22 (which is the port number where you attempted to connect to the web server when access was blocked), run the following command. This command should return a smaller number of results.

```
[ec2-user@cli-host 29]$ grep -rn 22 . | grep REJECT
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2005Z_a6a7184b.log:6:2 767398061261 eni-0a1469cce7034a10f 162.216.149.88 10.0.1.141 52221 9096 6 1 44 1717013126 1717013155 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2005Z_a6a7184b.log:16:2 767398061261 eni-0f711416389e35dd2 152.42.198.222 10.0.1.17 52156 5634 6 1 44 1717013136 1717013164 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2005Z_a6a7184b.log:18:2 767398061261 eni-0f711416389e35dd2 165.22.109.252 10.0.1.17 49529 5817 6 1 44 1717013136 1717013164 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2005Z_a6a7184b.log:38:2 767398061261 eni-0f711416389e35dd2 185.242.226.54 10.0.1.17 33458 8569 6 1 44 1717013199 1717013225 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2005Z_a6a7184b.log:39:2 767398061261 eni-0f711416389e35dd2 162.216.150.57 10.0.1.17 55972 500 17 1 1276 1717013199 1717013225 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2005Z_a6a7184b.log:40:2 767398061261 eni-0f711416389e35dd2 143.198.206.247 10.0.1.17 49521 50903 6 1 44 1717013199 1717013225 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2005Z_a6a7184b.log:41:2 767398061261 eni-0f711416389e35dd2 35.203.211.164 10.0.1.17 51562 47512 6 1 44 1717013199 1717013225 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2005Z_a6a7184b.log:42:2 767398061261 eni-0f711416389e35dd2 162.216.150.14 10.0.1.17 57246 1688 6 1 44 1717013199 1717013225 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2005Z_a6a7184b.log:43:2 767398061261 eni-0f711416389e35dd2 141.98.9.20 10.0.1.17 43661 52538 6 1 40 1717013199 1717013225 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2005Z_a6a7184b.log:44:2 767398061261 eni-0f711416389e35dd2 194.26.229.174 10.0.1.17 22 49699 6 1 44 1717013199 1717013225 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2005Z_a6a7184b.log:45:2 767398061261 eni-0f711416389e35dd2 152.42.198.222 10.0.1.17 52140 4303 6 1 44 1717013199 1717013225 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2005Z_a6a7184b.log:46:2 767398061261 eni-0f711416389e35dd2 79.110.62.65 10.0.1.17 48500 43464 6 1 40 1717013199 1717013225 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2005Z_a6a7184b.log:47:2 767398061261 eni-0f711416389e35dd2 183.101.147.115 10.0.1.17 46645 41439 6 1 44 1717013199 1717013225 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2005Z_a6a7184b.log:48:2 767398061261 eni-0f711416389e35dd2 79.110.62.65 10.0.1.17 48500 49087 6 1 40 1717013199 1717013225 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2005Z_a6a7184b.log:49:2 767398061261 eni-0f711416389e35dd2 35.203.210.22 10.0.1.17 50433 9364 6 1 44 1717013199 1717013225 REJECT OK
```

Step 6: Isolate the result set

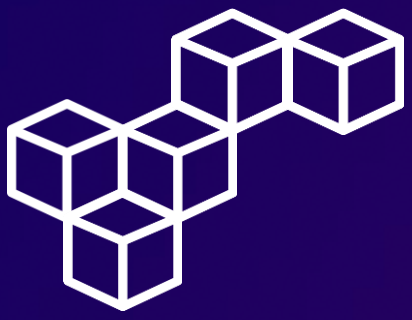
To isolate the result set so that it displays only the log entries that correspond to the failed SSH connection attempts that you made, filter the results using your own Public IP address. To confirm that the network interface ID that is recorded in the flow log matches the network interface that is assigned to the web server instance, run the following aws ec2 describe-network-interfaces command. Convert the start and end timestamps into a human-readable format, and compare the result to the current time using the date command.

```
[ec2-user@cli-host 29]$ grep -rn REJECT . | grep 190.117.66.167
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2050Z_5bbfb178.log:99:2 767398061261 eni-0a1469cce7034a10f 190.117.66.167 10.0.1.141 61888 443 6 5 260 1717015915 1717015945 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2050Z_5bbfb178.log:113:2 767398061261 eni-0a1469cce7034a10f 190.117.66.167 10.0.1.141 61889 443 6 5 260 1717015915 1717015945 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2055Z_a9ee4192.log:16:2 767398061261 eni-0a1469cce7034a10f 190.117.66.167 10.0.1.141 62043 443 6 5 260 1717016126 1717016153 REJECT OK
./767398061261_vpcflowlogs_us-west-2_fl-0d55bd41bd349aea0_20240529T2055Z_a9ee4192.log:23:2 767398061261 eni-0a1469cce7034a10f 190.117.66.167 10.0.1.141 62044 443 6 5 260 1717016126 1717016153 REJECT OK
[ec2-user@cli-host 29]$
```

```
[ec2-user@cli-host 29]$ aws ec2 describe-network-interfaces \
> --filters "Name=association.public-ip,Values='54.70.72.107'" \
> --query 'NetworkInterfaces[*].[NetworkInterfaceId,Association.PublicIp]'
[
    [
        "eni-0a1469cce7034a10f",
        "54.70.72.107"
    ]
]
[ec2-user@cli-host 29]$
```

```
[ec2-user@cli-host 29]$ date -d @1717015915 ; date -d @1717015945
Wed May 29 20:51:55 UTC 2024
Wed May 29 20:52:25 UTC 2024
[ec2-user@cli-host 29]$ date -d @1717016126 ; date -d @1717016153
Wed May 29 20:55:26 UTC 2024
Wed May 29 20:55:53 UTC 2024
[ec2-user@cli-host 29]$ date
Wed May 29 21:48:30 UTC 2024
[ec2-user@cli-host 29]$
```

aws re/start



Conclusions

Troubleshooting a VPC

Effective VPC troubleshooting requires checking route tables, security groups, and network ACLs to ensure correct traffic flow and access permissions.

VPC Flow Logs

VPC Flow Logs provide detailed information about IP traffic flowing to and from network interfaces in your VPC, aiding in identifying and resolving network issues.

Creating VPC Flow Logs

Setting up VPC Flow Logs involves specifying the VPC, subnet, or network interface to monitor, and choosing a log destination, such as an S3 bucket or CloudWatch Logs.

Downloading and extracting flow logs

Flow logs can be downloaded from their storage destination (e.g., S3) and extracted using tools like AWS CLI or SDKs to facilitate detailed examination and analysis.

Analyzing logs

Analyzing flow logs helps detect unusual patterns, such as unauthorized access attempts or traffic bottlenecks, enabling proactive network security and performance optimization.

aws re/start

Cristhian Becerra

cristhian-becerra-espinoza

+51 951 634 354

cristhianbecerra99@gmail.com

Lima, Peru