Troubleshooting Network Connectivity in a Peered VPC (Assessment Version)

**SAL-TF-200-NWTNCV-1 - Version 1.0.0**

© 2024 Amazon Web Services, Inc. or its affiliates. All rights reserved. This work may not be reproduced or redistributed, in whole or in part, without prior written permission from Amazon Web Services, Inc. Commercial copying, lending, or selling is prohibited. All trademarks are the property of their owners.

Note: Do not include any personal, identifying, or confidential information into the lab environment. Information entered may be visible to others.

Corrections, feedback, or other questions? Contact us at [*AWS Training and Certification*](https://support.aws.amazon.com/#/contacts/aws-training).

**Lab overview**

This lab covers the connectivity concepts of Amazon Virtual Private Cloud (Amazon VPC). It demonstrates configuring VPC peering between two VPCs in the same account, and allowing client-server HTTP traffic between hosts in the VPCs.

AnyCompany needs to allow HTTP traffic between a client and a server. The HTTP client instance resides in one VPC while the HTTP server instance resides in a different VPC in the same AWS account. A junior member of your team already implemented the solution but it seems it was misconfigured.

As a member of the cloud team at AnyCompany, you have been assigned to check connectivity between the client and server. Your task is to troubleshoot the issue and fix the configuration. The configuration should allow HTTP traffic between the client and server via the VPC peering.

High-level guidance and references are provided to assist you in fixing the issue. The detailed solution instructions are provided in a collapsible section which you can expand.

OBJECTIVES

By the end of this lab, you should be able to do the following:

* Test the HTTP connection between the client and server and verify the issue.
* Identify the issues in the existing configuration and remediate the configuration to allow HTTP traffic between the client and server via the VPC peering.
* Use the Reachability Analyzer to troubleshoot VPC connectivity issues.

TECHNICAL KNOWLEDGE PREREQUISITES

To successfully complete this lab, you should have a basic knowledge of:

* Navigating through the AWS Management Console.
* Amazon Virtual Private Cloud (Amazon VPC).
* Amazon Elastic Compute Cloud (Amazon EC2).

DURATION

This lab requires approximately *60* minutes to complete.

ICON KEY

Various icons are used throughout this lab to call attention to different types of instructions and notes. The following list explains the purpose for each icon:

* **Command:** A command that you must run.
* **Expected output:** A sample output that you can use to verify the output of a command or edited file.
* **Note:** A hint, tip, or important guidance.
* **Knowledge check:** An opportunity to check your knowledge and test what you have learned.
* **Hint:** A hint to a question or challenge.
* **Answer:** An answer to a question or challenge.
* **Task complete:** A conclusion or summary point in the lab.
* **Consider:** A moment to pause to consider how you might apply a concept in your own environment or to initiate a conversation about the topic at hand.
* **Learn more:** Where to find more information.

**Start lab**

1. To launch the lab, at the top of the page, choose **Start lab**.

**Caution:** You must wait for the provisioned AWS services to be ready before you can continue.

1. To open the lab, choose **Open Console**.

You are automatically signed in to the AWS Management Console in a new web browser tab.

**WARNING:** **Do not change the Region unless instructed.**

COMMON SIGN-IN ERRORS

**Error: You must first sign out**



If you see the message, **You must first log out before logging into a different AWS account:**

* Choose the **click here** link.
* Close your **Amazon Web Services Sign In** web browser tab and return to your initial lab page.
* Choose **Open Console** again.

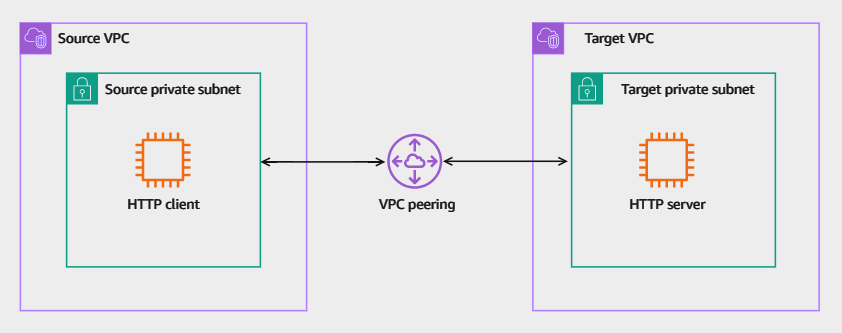
**Error: Choosing Start Lab has no effect**

In some cases, certain pop-up or script blocker web browser extensions might prevent the **Start Lab** button from working as intended. If you experience an issue starting the lab:

* Add the lab domain name to your pop-up or script blocker’s allow list or turn it off.
* Refresh the page and try again.

LAB ENVIRONMENT

The following diagram shows the lab scenario:



*Image description: The following list details the major resources in the lab:*

* *Two VPCs named as Source VPC and Target VPC.*
* *Each VPC has one private subnet.*
* *The HTTP client instance resides in the source private subnet.*
* *The HTTP server instance resides in the source private subnet.*
* *A VPC peering connection between the two VPCs.*

AWS SERVICES NOT USED IN THIS LAB

AWS service capabilities used in this lab are limited to what the lab requires. Expect errors when accessing other services or performing actions beyond those provided in this lab guide.

**Task 1: Testing HTTP client-server connectivity**

In this task, you test the HTTP connectivity between the client and server instances. You test the connectivity by initiating HTTP traffic from the HTTP client instance and observe the response.

Ensure that you completed the instructions in the *Start lab* section above and you have the AWS Management Console page open before you proceed to the next step.

1. Copy the **HTTPClientTerminal** value that is listed to the left of these instructions. Paste the value into a new browser tab, and then press **Enter** to access the **HTTP Client** instance terminal.

Now, initiate an HTTP connection from the *HTTP-Client* to the *HTTP-Server*.

1. **Command:** Run the following command on the **HTTP-Client** prompt after replacing the **HTTPServerIPAddress** string with its respective value listed to the left of these instructions.

wget http://HTTPServerIPAddress

**Expected output:** Your values might differ from what is seen below.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\* This is OUTPUT ONLY. \*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

--2023-07-12 02:41:00-- http://10.2.2.112/Connecting to 10.2.2.112:80... failed: Connection timed out.

Retrying.

The HTTP client attempts to initiate the connection but it fails after couple of minutes.

1. Press CTRL-C to exit the connection retries.

The HTTP connection failed which verifies the reported issue.

**Task complete:** You tested the HTTP connection between the client and server and verified the issue.

**Task 2: Troubleshooting and remediating the issue**

In this task, you identify the issues in the existing configuration and remediate the configuration to allow the HTTP traffic between the client and server via the VPC peering.

Now that have you confirmed the issue, you need to identify the services and resources which were misconfigured and remediate them.

You can navigate through the AWS Management Console of the VPC, EC2, or any other service you need to identify the issues in the current configuration and determine how to remediate them.

**Consider:** Here are few things to consider when you attempt remediating the issue:

* You have the permissions to use the Reachability Analyzer for troubleshooting.
* The *httpd* service is up and running on the *HTTP-Server* instance so you do not need to troubleshoot it.
* The HTTP service is running on the standard HTTP port (TCP port 80).
* You do not need to create new resources to resolve the issue. However, you can modify existing resources.
* You have the required AWS Identity and Access Management (IAM) permissions to remediate the issues.

DO IT YOURSELF

**Hint:** Here are some references to assist you in solving the issue:

* [How Reachability Analyzer works](https://docs.aws.amazon.com/vpc/latest/reachability/how-reachability-analyzer-works.html).
* [Troubleshoot a VPC peering connection](https://docs.aws.amazon.com/vpc/latest/peering/troubleshoot-vpc-peering-connections.html).

SOLUTION

Expand the *Detailed instructions* section below for the full solution.

**Detailed instructions**

TEST THE SOLUTION

To verify that you resolved all the issues, you can simply re-initiate the HTTP traffic from the *HTTP-Client* instance to the *HTTP-Server* instance.

1. Choose the **HTTP-Client** terminal session browser tab that you used in Task 1.

**Note:** If you do not have the terminal session still open, 1. copy the **HTTPClientTerminal** value that is listed to the left of these instructions. Paste the value into a new browser tab, and then press **Enter** to access the **HTTP Client** instance terminal.

Now, initiate an HTTP connection from the *HTTP-Client* to the *HTTP-Server*.

1. **Command:** Run the following command on the **HTTP-Client** prompt after replacing the **HTTPServerIPAddress** string with its respective value listed to the left of these instructions.

wget http://HTTPServerIPAddress

**Expected output:** Your values might differ from what is seen below.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\* This is OUTPUT ONLY. \*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

--2023-07-12 23:21:43-- http://10.2.2.112/

Connecting to 10.2.2.112:80... connected.

HTTP request sent, awaiting response... 200 OK

Length: 51 [text/html]

Saving to: ‘index.html’

100%[=======================================================================================================>] 51 --.-K/s in 0s

2023-07-12 23:21:43 (9.10 MB/s) - ‘index.html’ saved [51/51]

If you get a 200 response as above, then the connection is successful.

**Consider:** Here is a quick summary of the requirements for connecting to a resource in a VPC from a resource in a peer VPC:

* For each resource in each VPC, verify that the route table for its subnet contains a route that sends traffic destined for the peer VPC via the VPC peering connection.
* For EC2 instances, verify that the security groups for the EC2 instances allow traffic from the peer VPC.
* For each resource in each VPC, verify that the network ACL for its subnet allows traffic from the peer VPC.

**Task complete:** You identified the issues in the existing configuration and remediated the configuration to allow HTTP traffic between the client and server via the VPC peering.

**Conclusion**

You have successfully done the following:

* Tested the HTTP connection between the client and server and verified the issue.
* Identified the issues in the existing configuration and remediated the configuration to allow HTTP traffic between the client and server via the VPC peering.
* Used the Reachability Analyzer to troubleshoot VPC connectivity issues.

**End lab**

Follow these steps to close the console and end your lab.

1. Return to the **AWS Management Console**.
2. At the upper-right corner of the page, choose **AWSLabsUser**, and then choose **Sign out**.
3. Choose **End lab** and then confirm that you want to end your lab.

**Additional resources**

* [Control traffic to subnets using network ACLs - Ephemeral ports](https://docs.aws.amazon.com/vpc/latest/userguide/vpc-network-acls.html#nacl-ephemeral-ports).
* [Connect VPCs using VPC peering](https://docs.aws.amazon.com/vpc/latest/userguide/vpc-peering.html).

For more information about AWS Training and Certification, see [*https://aws.amazon.com/training/*](https://aws.amazon.com/training/).

*Your feedback is welcome and appreciated.*  
If you would like to share any feedback, suggestions, or corrections, please provide the details in our [*AWS Training and Certification Contact Form*](https://support.aws.amazon.com/#/contacts/aws-training).