Working with Amazon VPC Network Access Analyzer

**SPL-TF-100-NWVNAA-1 - Version 1.0.5**

© 2023 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 demonstrates how to use the Network Access Analyzer feature in your Amazon VPC (Virtual Private Cloud). This feature helps you understand, verify, and improve your network security posture.

OBJECTIVES

By the end of this lab, you will be able to:

* Understand, verify, and improve a network security posture.
* Demonstrate a network configuration meets compliance requirements.

DURATION

This lab requires approximately **60 minutes** to complete.

PREREQUISITES

Knowledge about basic networking concepts (such as IP Addressing, CIDR notation, and routing), an understanding with basic cloud operations, and familiarity with navigating the AWS Management Console.

**Start lab**

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

 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.

**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.

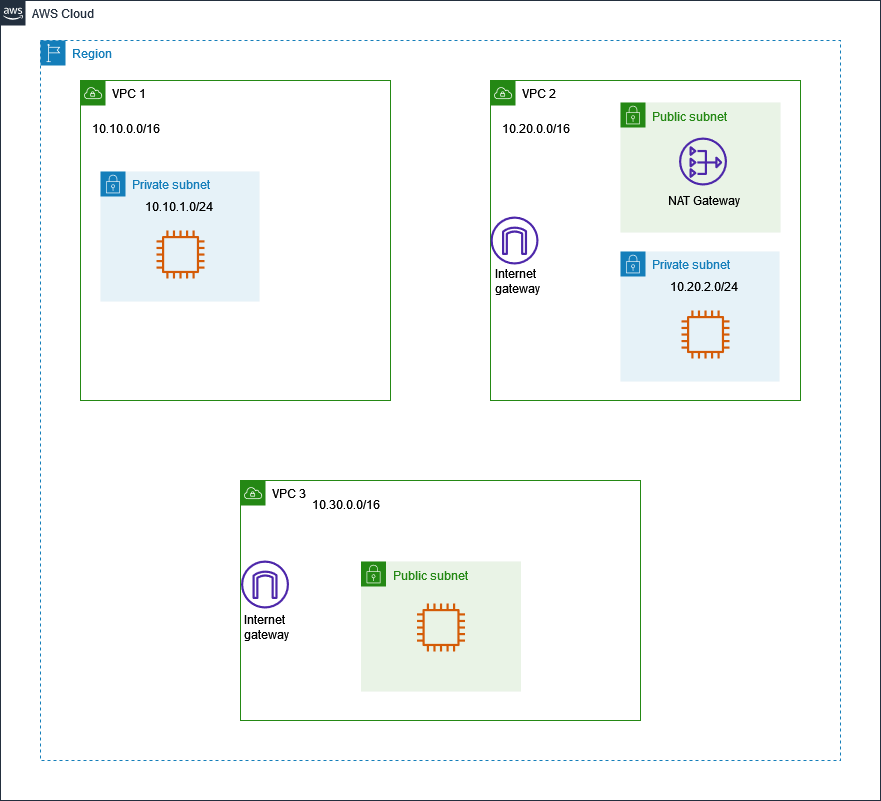
**Error: Choosing Start Lab has no effect**

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.

**Task 1: Understanding the architectures you are working with**

In this initial task, you explore the lab environment. This lab is pre-configured with 3 VPCs, each containing an EC2, and network elements like subnets, route tables, NACLs(Network Access Control Lists), security groups etc. Each VPC has unique architecture that generalizes common network designs.



EXPLORE THE PRE-CREATED RESOURCES IN YOUR ENVIRONMENT

1. At the top of the page, in the unified search bar, search for and choose

VPC

.

1. In the left navigation pane, choose **Your VPCs**.

You see the pre-configured VPCs, namely: vpc1, vpc2, and vpc3. You can choose vpcs to see details such as the CIDR and other attributes.

1. In the left navigation pane, choose **Subnets**.

You see the pre-configured subnets, namely:

* vpc1-PRIVATE-subnet
* vpc2-PRIVATE-subnet
* vpc2-PUBLIC-subnet
* vpc3-PUBLIC-subnet

You can choose the subnets to see more details.

1. In the left navigation pane, choose **Route tables**.

You see the pre-configured Route tables, namely:

* vpc1-PRIVATE-RouteTable
* vpc2-PRIVATE-RouteTable
* vpc2-PUBLIC-RouteTable
* vpc3-PUBLIC-RouteTable

You can choose the route tables to see more details.

1. In the left navigation pane, choose **Internet gateways**.

You see the pre-configured internet gateways, namely:

* vpc2-internet-gateway
* vpc3-internet-gateway

You can choose the internet gateways to see more details.

1. In the left navigation pane, choose **NAT gateways**.

You see the pre-configured NAT gateway, namely:

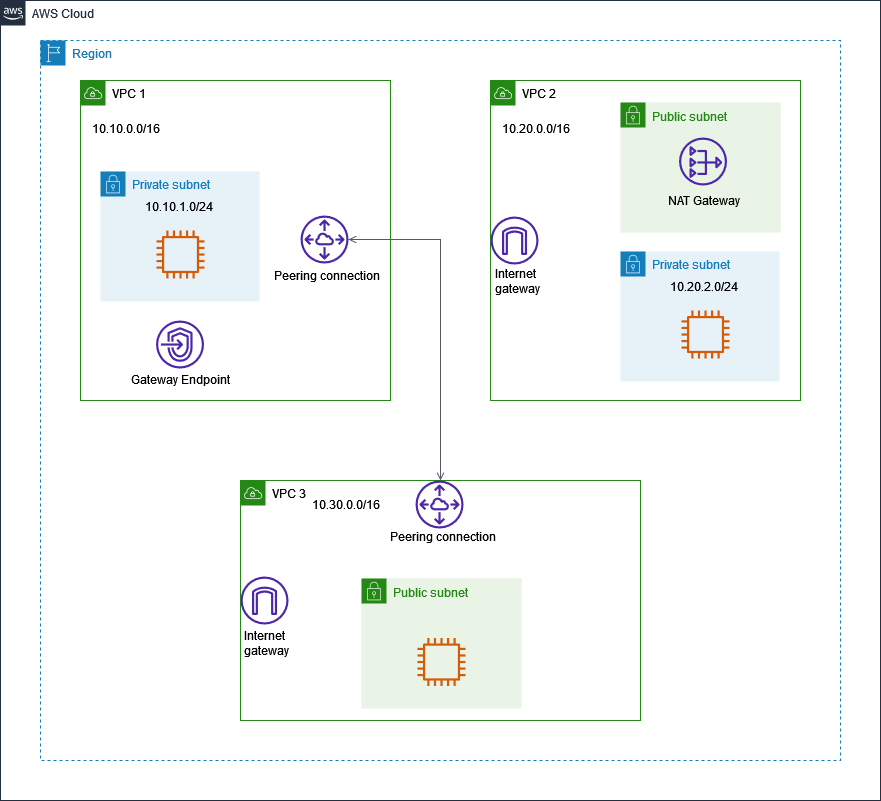
* vpc2-NatGateway-for-PrivateSubnet

You can choose the NAT gateway to see more details.

FINAL ENVIRONMENT

This image shows the configuration to be added. You will create a:

* S3 Gateway endpoint in vpc1.
* VPC peering connection between vpc1 and vpc3.



**Task 2: Use a Network Access Scope template to analyze ingress traffic**

Network Access Analyzer uses automated reasoning algorithms to analyze the network paths that a packet can take between resources in an AWS network. The key concepts of the Network Access Analyzer are:

* Network Access Scope
* Findings

A **Network Access Scope** determines the types of findings that the analysis produces. You add entries to **MatchPaths** to specify the types of network paths to identify. You add entries to **ExcludePaths** to specify the types of network paths to exclude.

**Findings** are potential paths in your network that match any of the MatchPaths entries in your Network Access Scope, but do not match any of the ExcludePaths entries in your Network Access Scope.

In this task, use a pre-built template to analyze available traffic paths from an internet gateway.

1. At the top of the page, in the unified search bar, search for and choose

VPC

.

1. In the left navigation pane, scroll down to the bottom of the page, and choose **Network Manager**.
2. In the left navigation pane, choose **Network Access Analyzer**.
3. Choose **Get started** .

Here you see a few default Network Access Scopes. You may choose to use them directly. However, for this lab you will use templates to create new Network Access Scopes.

1. Choose **Create Network Access Scope** .

Here you see several Network Access Scope templates. Use one of the template to identify ingress traffic from the internet.

1. Choose  the template named **Identify access from Internet Gateways**.
2. Choose **Next** , then configure:

* **Name**:

ingress-paths

Leave the default configuration in the rest of the fields.

* Choose **Next** .

1. Review the template and conditions within the Network Access Scope.
2. Choose **Create Network Access Scope** .

**Sample output:**

 You successfully created < ID >

In the Network Access Scopes section, you see the newly created scope. Use this scope to analyze if there is any ingress path from an internet gateway.

1. Select  **ingress-paths**.
2. Choose **Analyze**.

**Sample output:**

 Analysis successfully started for Network Access Scope < ID >

* The analysis takes a few minutes or less to complete.
* After completion, the result is **Findings detected**.

**Note:** For this task, and the rest of the tasks in this lab, disregard any message similar to the following - *The analysis succeeded but it was unable to analyze all components. Reason: The request was missing the following optional permissions: [directconnect:DescribeVirtualInterfaces, directconnect:DescribeDirectConnectGateways]*

This page also presents a path display. The display presents the network elements within the ingress path from **Source** to **Destination**.

The ingress path analysis starts from the internet gateway of **vpc3** all the way up to the network interface of **vpc3-public-ec2**.

Review the network diagram provided at the top of this page. Both **vpc2** and **vpc3** have an internet gateway. You might ask - how come the analysis did not display an ingress path for vpc2?

The reason is - the **Network Access Scope definition** for this analysis has a source of **internet gateway** and a destination of **network interface**, in this case vpc2-private-ec2 is in a private subnet, and internet gateways do not have a direct path to network interfaces in a private subnet.

In this task, you used a pre-built template to analyze available traffic paths from an internet gateway. This analysis helps verify ingress traffic paths, and even demonstrate compliance in certain use cases.

**Task 3: Create and analyze a VPC endpoint path**

In this task, you create a VPC endpoint for Amazon S3 service and verify its path using the Network Access Analyzer. VPC endpoints increases the security posture of a VPC by enabling connectivity to services without the need of an Internet Gateway, hence your traffic stays within the AWS network.

1. At the top of the page, in the unified search bar, search for and choose

VPC

.

1. In the left navigation pane, choose **Endpoints**
2. Choose **Create endpoint** , then configure:

* **Name tag**:

vpc1-S3-endpoint

* **Services**: type ‘S3’ and press enter to search.
* Choose  the S3 service name with Type as **Gateway**.
* **VPC**: choose vpc1
* **Route tables**: choose  all route tables
* **Policy**: choose  Full access
* Choose **Create endpoint** .

**Sample output:**

 Successfully created VPC endpoint < ID >

Next, use the Network Access Analyzer to verify the S3 gateway endpoint path.

1. In the left navigation pane, scroll down to the bottom of the page, and choose **Network Manager**.
2. In the left navigation pane, choose **Network Access Analyzer**.
3. Choose **Create Network Access Scope** .
4. Choose  the template named **Validate access from trusted networks**.
5. Choose **Next** , then configure:

* **Name**:

s3-endpoint-path

* Under **Source** section:
  + **Resource selection**: choose ‘Resource types’
  + **Resource types**: choose  EC2 Instances
* Under **Destination** section:
  + **Resource selection**: choose ‘Resource types’
  + **Resource types**: choose  VPC Endpoints
* Choose **Next** .

1. Review the template and conditions within the Network Access Scope.
2. Choose **Create Network Access Scope** .

**Sample output:**

 You successfully created < ID >

In the Network Access Scopes section, you see the newly created scope. Use this scope to analyze if there is any egress path through a VPC endpoint.

1. Select  **s3-endpoint-path**. (If any other Network Access Scope is already selected, de-select it.)
2. Choose **Analyze** .

**Sample output:**

 Analysis successfully started for Network Access Scope < ID >

* The analysis takes a few minutes or less to complete.
* After completion, the result is **Findings detected**.

This page also presents a path display. The display presents the network elements within the egress path from **Source** to **Destination**.

The egress path analysis starts from the network interface of the EC2 in vpc1 all the way up to the S3 endpoint. This analysis helps verify traffic paths, and even demonstrate compliance in certain use cases.

In this task, you created a VPC endpoint for Amazon S3 service and verified its path using the Network Access Analyzer.

**Task 4: Create a custom network access scope to analyze a private subnet**

In this task, you define a custom network access scope to verify that a private subnet does not have internet access. For a subnet to be private, there shouldn’t be a route associated with an Internet Gateway.

1. In the left navigation pane, choose **Network Access Analyzer**.
2. Choose **Create Network Access Scope** .
3. Choose  **Empty template.**
4. Choose **Next** , then configure:

* **Name**:

verify-private-subnet

* Choose **Add match condition** .
* Under **Source** section:
  + **Resource selection**: choose ‘Resource types’
  + **Resource types**: choose  Internet Gateways
* Under **Destination** section:
  + **Resource selection**: choose ‘Resource IDs’
  + **Resource types**: choose  Subnets
  + **Resource IDs**: choose  vpc1-PRIVATE-subnet,  vpc2-PRIVATE-subnet
* Choose **Next** .

1. Review the Network Access Scope definition.
2. Choose **Create Network Access Scope** .

**Sample output:**

 You successfully created < ID >

In the Network Access Scopes section, you see the newly created custom scope. Use this scope to analyze if there is any direct path from an Internet Gateway to private subnets.

1. Choose  verify-private-subnet. (If any other Network Access Scope is already selected, de-select it.)
2. Choose **Analyze** .

**Sample output:**

 Analysis successfully started for Network Access Scope < ID >

When the analysis is complete, you see the results under the ‘Latest analysis’ tab.

* The result is:  No findings detected

This verifies that the expected configuration is in place, the Internet gateway does not have direct access to the private subnet.

In this task, you used a custom network access scope to verify that a private subnet does not have internet access.

**Task 5: Create a custom network access scope to analyze VPC segmentation**

In this task, you define a custom network access scope to verify VPC segmentation. Assume a use case where **vpc1** & **vpc3** require a private connection through VPC peering.

First, use the Network Access Analyzer to verify there isn’t an existing VPC peering connection.

1. In the left navigation pane, choose **Network Access Analyzer**.
2. Choose **Create Network Access Scope** .
3. Choose  **Empty template.**
4. Choose **Next** , then configure:

* **Name**:

verify-vpc-segmentation

* Choose **Add match condition** .
* Under **Source** section:
  + **Resource selection**: choose ‘Resource IDs’
  + **Resource types**: choose  VPCs
  + **Resource IDs**: choose  vpc1,  vpc2,  vpc3
* Under **Destination** section:
  + **Resource selection**: choose ‘Resource types’
  + **Resource types**: choose  Peering Connections
* Choose **Next** .

1. Review the Network Access Scope definition.
2. Choose **Create Network Access Scope** .

**Sample output:**

 You successfully created < ID >

In the Network Access Scopes section, you see the newly created custom scope. Use this scope to analyze if there are any VPC peering connections.

1. Choose  verify-vpc-segmentation. (If any other Network Access Scope is already selected, de-select it.)
2. Choose **Analyze** .

**Sample output:**

 Analysis successfully started for Network Access Scope < ID >

When the analysis is complete, you see the results under the ‘Latest analysis’ tab.

* The result is:  No findings detected

This verifies that there aren’t any VPC peering connections.

CREATE A VPC PEERING CONNECTION BETWEEN VPC1 AND VPC3.

1. At the top of the page, in the unified search bar, search for and choose

VPC

.

1. In the left navigation pane, choose **Peering connections**
2. Choose **Create peering connection** , then configure:

* **Name**:

vpc1-vpc3-peering

* **VPC ID (Requester)**: vpc1
* **Account**:  My account.
* **Region**:  This Region
* **VPC ID (Accepter)**: vpc3

1. Choose **Create peering connection**.

**Sample output:**

 A VPC peering connection (ID) / vpc1-vpc3-peering has been requested…

The state of the connection request transitions to **Pending acceptance**.

1. From the **Actions** button, choose ‘Accept request’.

Review the connection request information.

1. Choose **Accept request**.

**Sample output:**

 Your VPC peering connection (ID) | vpc1-vpc3-peering has been established…

The connection state is  Active. To send and receive traffic across this VPC peering connection, you add the CIDR of the peered VPC in the route table.

1. In the left navigation pane, choose **Route tables.**
2. Select  the route table named **vpc1-PRIVATE-RouteTable**.
3. Choose **Routes** tab.

**Note:** There are two route entries.

* A destination route of 10.10.0.0/16, with Target as local. This is a default entry, it is automatically created during the VPC launch.
* A destination route to S3, with Target as a VPC endpoint. This route was added in Task 3, when you created a VPC endpoint configuration for S3 access.

1. Choose **Edit routes**.
2. Choose **Add route**.

Add vpc3’s CIDR 10.30.0.0/16 to enable vpc1 to reach it through the VPC peering connection.

* **Destination**- 10.30.0.0/16
* **Target**- select **Peering Connection** from the drop-down list.
  + Choose **vpc1-vpc3-peering**

1. Choose **Save changes**.

**Sample output:**

**Updated routes for < rtb-ID > / vpc1-PRIVATE-RouteTable successfully**

Now, the **vpc1-PRIVATE-RouteTable** has 3 routes. Configure routing for vpc3 as well.

1. In the left navigation pane, choose **Route tables.**
2. Select  the route table named **vpc3-PUBLIC-RouteTable**.
3. Choose **Routes** tab.

**Note:** There are two route entries.

* A destination route of 10.30.0.0/16, with Target as local. This is a default entry, it is automatically created during the VPC launch.
* A destination route of 0.0.0.0/0, with Target as an Internet gateway.

1. Choose **Edit routes**.
2. Choose **Add route**.

Add vpc1’s CIDR 10.10.0.0/16 to enable vpc1 to reach it through the VPC peering connection.

* **Destination**- 10.10.0.0/16
* **Target**- select **Peering Connection** from the drop-down list.
  + Choose **vpc1-vpc3-peering**

1. Choose **Save changes**.

**Sample output:**

**Updated routes for < rtb-ID > / vpc3-PUBLIC-RouteTable successfully**

Now, **vpc3-PUBLIC-RouteTable** has 3 routes.

Use the VPC peering Network Access Scope you created previously to re-analyze if VPC segmentation exists.

1. In the left navigation pane, scroll down to the bottom of the page, and choose **Network Manager**.
2. In the left navigation pane, choose **Network Access Analyzer**.
3. Choose  verify-vpc-segmentation.
4. Choose **Analyze** .

**Sample output:**

 Analysis successfully started for Network Access Scope < ID >

When the analysis is complete, you see the results under the ‘Latest analysis’ tab.

* The result is: **Findings detected**.
* There are two findings with the following starting point:
  + vpc1-private-ec2
  + vpc3-public-ec2

This page also presents a path display. The display presents the network elements that have a **VPC peering** connection within their path. The result only shows vpc1 and vpc3 are peered. Therefore, vpc2 has no peering connection to any other VPC.

This analysis helps verify VPC segmentation, and even demonstrate compliance in certain use cases.

In this task, you created a VPC peering connection between two VPCs and verified network segmentation exists using the Network Access Analyzer.

**Task 6: Verify the use of a NAT gateway for internet traffic**

A common use case for having a NAT gateway in a VPC is to enable internet access for a private subnet. There are some use cases where you have a private subnet that does not require access to the internet.

Review the **private subnets** in the architecture diagram provided in the beginning section of this lab. Both vpc1 and vpc2 have private subnets. However, only vpc2 contains a NAT gateway. The diagram depicts a use case where the private subnet in vpc2 requires access to the internet; while the private subnet in vpc1 does not require internet access.

In this task, use the Network Access Analyzer to validate that the private subnet in vpc2 uses a NAT gateway when accessing the internet. At the same time, validate that the private subnet in vpc1 does not have access to the internet.

1. In the left navigation pane, choose **Network Access Analyzer**.
2. Choose **Create Network Access Scope** .
3. Choose  **Empty template.**
4. Choose **Next** , then configure:

* **Name**:

verify-NAT-use

* Choose **Add match condition** .
* Under **Destination** section:
  + **Resource selection**: choose ‘Resource types’
  + **Resource types**: choose  Internet Gateways

**Note:** No conditions configured under **Source** section.

* Choose **Next** .

1. Review the Network Access Scope definition.
2. Choose **Create Network Access Scope** .

**Sample output:**

 You successfully created < ID >

In the Network Access Scopes section, you see the newly created custom scope.

1. Choose  verify-NAT-use. (If any other Network Access Scope is already selected, de-select it.)
2. Choose **Analyze** .

**Sample output:**

 Analysis successfully started for Network Access Scope < ID >

When the analysis is complete, the result is: **Findings detected**.

As expected, there are two **Findings**:

* A **private** EC2 Instance in vpc2 accessing the internet gateway.
* A **public** EC2 Instance in vpc3 accessing the internet gateway.

The findings do not include the private EC2 instance in vpc1.

Toggle  between the two findings and review the display path for each finding. You see a NAT gateway within the path for **vpc2** instance. You can identify the NAT gateway by finding the label prefixed with **nat-** followed by a string of characters.

This validates the architecture diagram, showing internet access is enabled for the private subnet in vpc2 through the NAT gateway; while the private subnet in vpc1 does not have access to the internet.

In this task, you used the Network Access Analyzer to validate a network architecture diagram, in particular, the use of a NAT gateway for internet traffic.

**Task 7: Duplicate and modify a Network Access Scope**

Network Access Scopes can be duplicated, modified, then used to run a new analysis. This can help save time from creating a new Network Access Scope.

In this task, duplicate the **verify-NAT-use** Network Access Scope created in the previous task, and use the new scope to check any internet access path that doesn’t include a NAT gateway.

1. In the left navigation pane, choose **Network Access Analyzer**.
2. Choose  verify-NAT-use.
3. From the **Actions** button, choose ‘Duplicate and modify’.
4. In the Duplicate Network Access Scope page, configure:

* **Name**:

verify-no-NAT-use

* In the **Exclusion conditions** section, choose **Add exclusion condition** .
* Under **Through** section, choose **Resource types**.
* **Resource Types**:  NAT Gateways
* Choose **Duplicate and analyze Network Access Scope** .

**Sample output:**

 You successfully created and started an analysis for Network Access Scope < ID >

After a few moments the ‘Analysis status’ for the newly created scope transitions to  Complete. You might need to refresh the page to see the state change.

1. Choose  verify-no-NAT-use. (If any other Network Access Scope is already selected, de-select it.)
2. Choose the link under the **Network Access Scope ID** column.

There is one **Finding** -

* EC2 Instance (vpc3-public-ec2) in vpc3 accessing the internet gateway.
* The instance is in a public subnet.

This confirms what you observed in the architecture diagram: vpc3 can access the internet directly, without the need of a NAT gateway.

In this task, you duplicated and modified a previously created Network Access Scope. You used the new Network Access Scope to validate an internet access path that doesn’t include a NAT gateway.

**Task 8: Validate a compliance requirement**

In this final task, you configure a network change and demonstrate a compliant configuration.

Assume a use case whereby the private instance in vpc2 is required to access a specific IP address and port number. The requirement is:

* Allow egress traffic to destination IP address - 205.251.242.103 *(this is an Amazon.com IP address)*.
* Destination port number - 443.

Before making changes, use the Network Access Analyzer to review the current internet bound configuration path for vpc2-private-ec2.

1. In the left navigation pane, choose **Network Access Analyzer**.
2. Choose **Create Network Access Scope** .
3. Choose  **Empty template.**
4. Choose **Next** , then configure:

* **Name**:

vpc2-private-outbound

* Choose **Add match condition** .
* Under **Source** section:
  + **Resource selection**: choose ‘Resource IDs’
  + **Resource types**: choose  EC2 Instances
  + **Resource IDs**: choose  vpc2-private-ec2
* Under **Destination** section:
  + **Resource selection**: choose ‘Resource types’
  + **Resource types**: choose  Internet Gateways
* Choose **Next** .

1. Review the Network Access Scope definition.
2. Choose **Create Network Access Scope** .

**Sample output:**

 You successfully created < ID >

In the Network Access Scopes section, you see the newly created custom scope.

1. Choose  vpc2-private-outbound. (If any other Network Access Scope is already selected, de-select it.)
2. Choose **Analyze** .

**Sample output:**

 Analysis successfully started for Network Access Scope < ID >

When the analysis is complete, the result is: **Findings detected**.

Review the path analysis. Notice the second element in the path is the Security Group:

* The label is prefixed with **sg-** followed by a string of characters.
* The **Destination** address is 0.0.0.0/0 ; and the outbound port is 80

UPDATE THE SECURITY GROUP TO MATCH THE REQUIREMENT

1. Choose the security group label.
2. Choose the security group Resource ID.
3. Choose the **Outbound rules** tab.
4. Choose **Edit outbound rules** .
5. In the **HTTP** drop down list, choose **HTTPS**.
6. Replace the

0.0.0.0/0

 destination address with

205.251.242.103/32

1. Choose **Save rules** .
2. In the left navigation pane, scroll down to the bottom of the page, and choose **Network Manager**.
3. In the left navigation pane, choose **Network Access Analyzer**.
4. Choose  vpc2-private-outbound.
5. Choose **Analyze** .

**Sample output:**

 Analysis successfully started for Network Access Scope < ID >

When the analysis is complete, the result is: **Findings detected**.

Review the path analysis:

* The Security group **Destination** address is 205.251.242.103/32 ; and the outbound port is 443.

This analysis validates the current configuration is compliant with the requirement.

In this task, you configured a network change and used the Network Access Analyzer to demonstrate a compliant configuration.

**Conclusion**

 Congratulations! You now have successfully:

* Understood, verified, and improved a network security posture.
* Demonstrated a network configuration meets compliance requirements.

**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**

* For more information about how to use Network Access Analyzer, see [AWS Network Access Analyzer Documentation](https://docs.aws.amazon.com/vpc/latest/network-access-analyzer/what-is-network-access-analyzer.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).