**Network Access Control List (Network ACL) in VPC.**

A network access control list (ACL) allows or denies specific inbound or outbound traffic at the subnet level. You can use the default network ACL for your VPC, or you can create a custom network ACL for your VPC with rules that are similar to the rules for your security groups in order to add an additional layer of security to your VPC.

The following diagram shows a VPC with two subnets. Each subnet has a network ACL. When traffic enters the VPC (for example, from a peered VPC, VPN connection, or the internet), the router sends the traffic to its destination. Network ACL A determines which traffic destined for subnet 1 is allowed to enter subnet 1, and which traffic destined for a location outside subnet 1 is allowed to leave subnet 1. Similarly, network ACL B determines which traffic is allowed to enter and leave subnet 2.


      A VPC with two subnets and a network ACL for each subnet.
    

**Network ACL rules**

You can add or remove rules from the default network ACL, or create additional network ACLs for your VPC. When you add or remove rules from a network ACL, the changes are automatically applied to the subnets that it's associated with.

**The following are the parts of a network ACL rule:**

* **Rule number**. Rules are evaluated starting with the lowest numbered rule. As soon as a rule matches traffic, it's applied regardless of any higher-numbered rule that might contradict it.
* **Type**. The type of traffic; for example, SSH. You can also specify all traffic or a custom range.
* **Protocol**. You can specify any protocol that has a standard protocol number. For more information, see [Protocol Numbers](http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml). If you specify ICMP as the protocol, you can specify any or all of the ICMP types and codes.
* **Port range**. The listening port or port range for the traffic. For example, 80 for HTTP traffic.
* **Source**. [Inbound rules only] The source of the traffic (CIDR range).
* **Destination**. [Outbound rules only] The destination for the traffic (CIDR range).
* **Allow/Deny**. Whether to allow or deny the specified traffic.

**Architecture:-**

A screenshot of a computer

Description automatically generated

Above image shows the added layer of security using VPC Network ACL. It works similar to the security group. With a few changes.

Security group applied to instance level, whereas Network ACL applied to Subnet level.

Security group can have only allow rules, Network ACL can have both allow and Deny rules.

We have 2 layers of protection. Now if we have to access the EC2 instance (or any other resources) we have to pass through 2 layers of protection. (first Network ACL and Second one is Security group).

Steps:-

1. **Create Security group with all traffic allowed.**
2. **Create VPC and Subnet.**
3. **Create Web Server EC2 Instance.**

As all the prerequisites are ready we can create the Network ACL now,

1. **Create Network ACL.**

Navigate to VPC🡪Network ACL. Click on Create network ACL.

A screenshot of a computer

Description automatically generated

Click on Create network ACL.

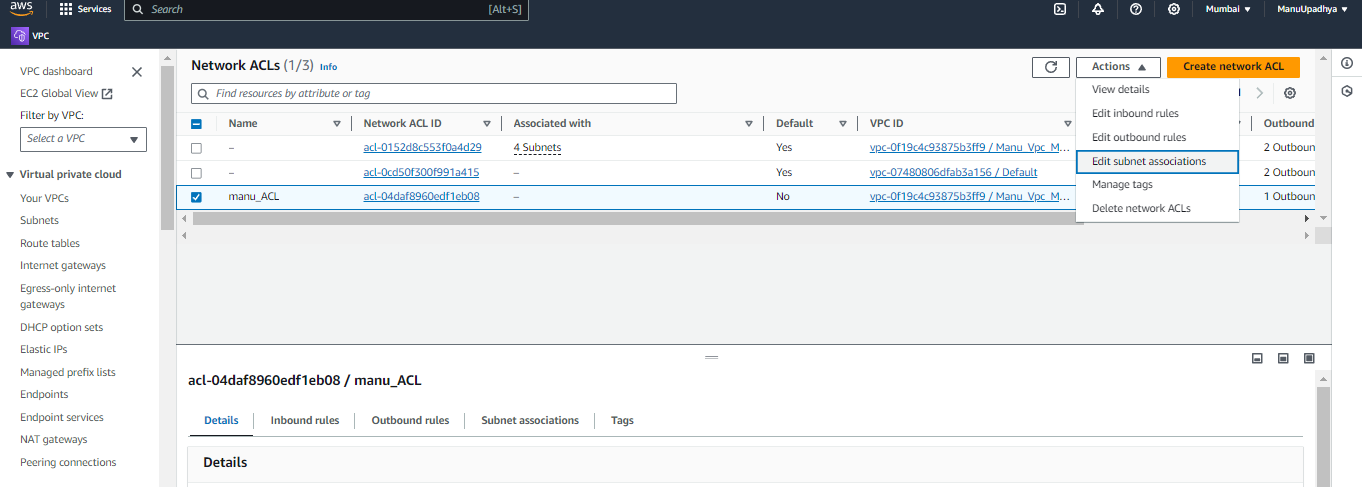
A screenshot of a computer

Description automatically generated

Network ACL is useless until we attach it with the subnet.

Now our already created EC2 instances in the AP-South-1A subnet we will attach the created network ACL to this subnet.

Select the created network ACL 🡪 Actions 🡪 edit subnet associations.



A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Click on Save changes.

When we are creating Network ACL, By default all inbound traffic are in deny state.

So now we are not able to access the website.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

The website is also not opening. If we use ping server also, we won’t get the response.

Now Navigate to Network ACL and add rule.   
Here Note that **Rule number is very important** because ACLs rules are applied based on it. (priority of rules are decided based on this rule number, Lower the rule number higher the priority.)

A screenshot of a computer

Description automatically generated

Now I Have added 3 rules, ICMP, HTTPS, SSH (used for ping, Web access, SSH respectively).

A screenshot of a computer

Description automatically generated

Still, we are not getting response because We have only added inbound rules. Network ACLs are Stateless, we need to add outbound rule to get response as of now outbound rule is all traffic deny so we are not getting the response.

Now we will add the outbound network ACL rule.

A screenshot of a computer

Description automatically generated

We have added all traffic allowed.

A screen shot of a computer

Description automatically generated

We can access the website now.

Same way we can add deny rules as well since it is an extra feature in the Network ACL, I am not going to perform lab on this because we have already tested for all traffic deny in the beginning, I think by that we can understand the deny concept easily.

Another example to test deny rule, we can add our PC IP address to deny rule and try to access website from the system we can’t access the website from the PC. If we try to access the same website using our mobile, we can access the website.

**Note:-**  Rule Number is Very Very important as priority of rule depends upon it.