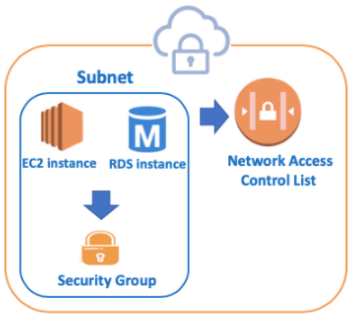
Creating Security Groups & Network ACLs

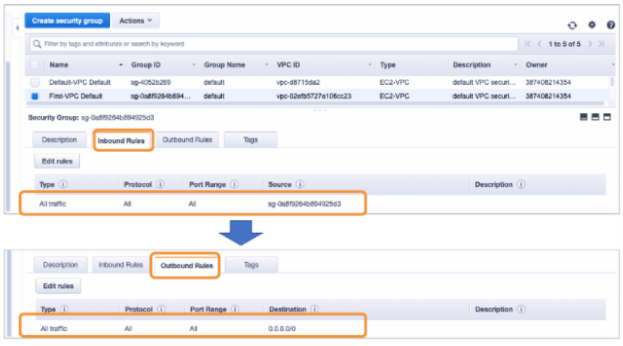
Security of Instance&Subnets



Security of Instance and Subnets

- Network ACLs and Security Groups are the rules created for a very similar purpose and we determine via them;
  - Which type of traffic to accept from which origin,
  - Which type of traffic to which destination to send.
- But, the main difference between them is that Network ACLs are assigned to **Subnets**, while Security Groups are assigned to **RDS and EC2 Instances**.
- So, we can say that Security groups are **instance-based** components. Network ACLs are **subnet-based security groups**.
- Network ACLs are also assigned to the **entire** subnet. Therefore, while Subnets are subject to the rules of Network ACLs, Instances are subject to the rules of both ACLs and Security Groups.

Structure of Default Security Groups



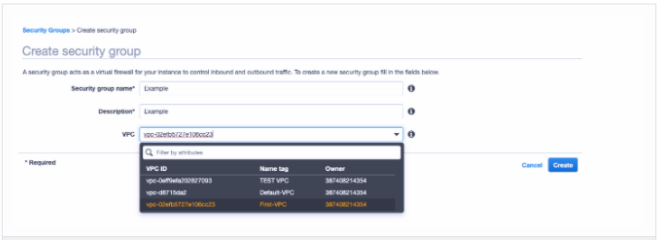
Default Security Group

Regardless of whether it is default or newly created, Security Groups **deny all inbound** traffic and **allow all outbound** traffic **until you add rules**.

Let's see the Structure of the Default Security Group seen in the picture above.

- First click **Default Security Group** of **First-VPC** created before,
- Select **inbound** tab at the bottom,
  - You'll see that Security Group's own ID is written as **Source**.
  - It means Security Group deliver the package **locally, not out of instance**,
- And after that, when we choose **outbound** we'll see one rule that means **all traffic is allowed**.
- So, any instance associated with the Default Security Group will not be accessible from **outside**.
- There will be also the same inbound and outbound rules if you create a new Security Group until you add rules.

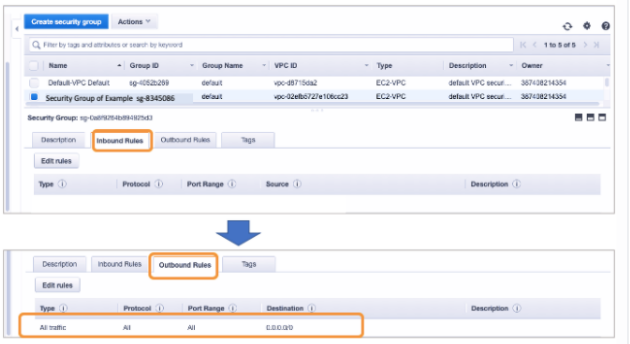
Create a New Security Group



Create a New Security Group

Now, let's create a new Security Group and modify it as we want by adding rules. But, while modifying we can add only **Allow Rules**.

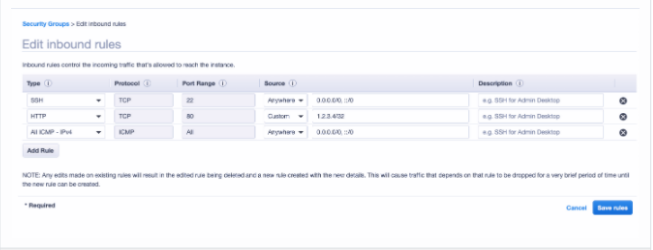
- First, go to the **Security Groups** section from the left-hand menu on VPC Dashboard,
- Then, Click **Create Security Group** tab.
  - Let's give the name of **Example** for Security Group Name
  - Enter also **Example** for Description,
  - Select the First-VPC that we created before,
  - Finally, click **Create**. It's done.
- Let's check our new Security Group. So, click the **Security Group of Example** on Menu:



New Security Group

- If you select the inbound tab, you'll see there is no rule.
- As for outbound traffic, all port is **allowed**. Because we haven't added any rule yet.

Modify New Security Group



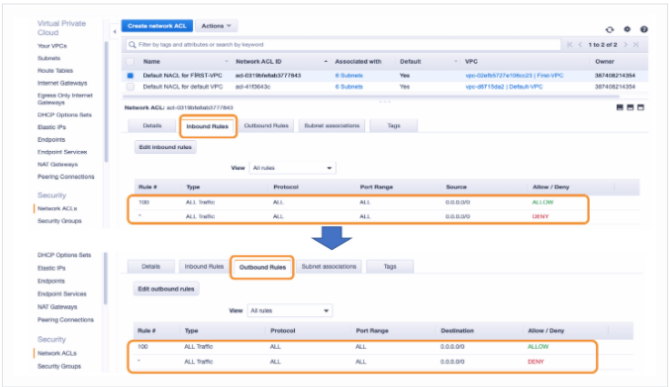
Editing Inbound Rules

So, Let's click **Edit Rules** while **inbound** tab of our new Security Group is selected to modify inbound traffic.

- As shown in the figure, we can customize our connection type according to our wishes.
- If we want to reach our instance from **Any IP** with **SSH Protocol** through **Port 22**,
  - For the Type Section : We select **SSH**,
  - For the Source Section : We choose **Any** and enter the IP of **0.0.0.0/0** which means anywhere.
- If we want to **ping** our instance only from **IP of 1.2.3.4/32 (e.g.)** with **All ICMP - IPv4 Protocol** through **All Port**,
  - For the Type Section : We select **All ICMP - IPv4**
  - For the Source Section : We choose **Custom** and enter the IP of **1.2.3.4/32**.

- If we want to reach our instance from **Any IP** with **HTTP Protocol** through **Port 80**,
  - For the Type Section : We select **HTTP**,
  - For the Source Section : We choose **Any** and enter the IP of **0.0.0.0/0** which means anywhere.
- Thus, virtual machines associated with the Security Group will be working according to the rules specified in this security group.

Structure of Default Network ACLs



Structure of Default Network ACL

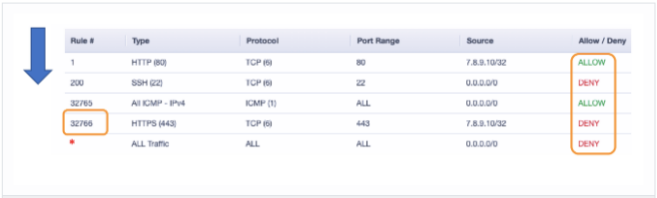
As we mentioned before Network ACL is somehow firewalls of the subnets.

First of all, let's look at the Network ACL menu and check the rules of **default** Network ACLs as you see in the picture above,

- Click **Default Network ACLs** of **First-VPC**,
- Then select **Inbound** tab, you'll see two rules as default.
  - The rule number 100 is at the top of the list. It determines that all inbound traffic is **allowed.**
  - The second rule with marked "*", **denies** all traffic.
- If you select **outbound tab**, you'll see the same rules and regulations also.
- According to the Network ACL working principle, all mentioned above means that **all traffic is allowed.**

But, How does it happen? Let's see.

Network ACL Working Principle-1



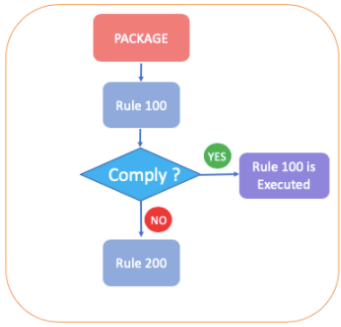
Rule #	Type	Protocol	Port Range	Source	Allow / Deny
1	HTTP (80)	TCP (6)	80	7.8.9.10/32	ALLOW
200	SSH (22)	TCP (6)	22	0.0.0.0/0	DENY
32765	All ICMP - IPv4	ICMP (1)	ALL	0.0.0.0/0	ALLOW
32766	HTTPS (443)	TCP (6)	443	7.8.9.10/32	DENY
*	ALL Traffic	ALL	ALL	0.0.0.0/0	DENY

Network ACL Working Principle

If we look Network ACL working principle:

- A Network ACL consists of numbered lists of rules as you see in the picture above.
- The number of rules can be selected by the user from 1 to 32766.
- You can specify both allow rules and deny rules.
- There is a hierarchy between the rules. The lowest-numbered rule at the top of the list has privileges.
- At the bottom of the list, there is a rule added by default for all Network ACL which denies all traffic.

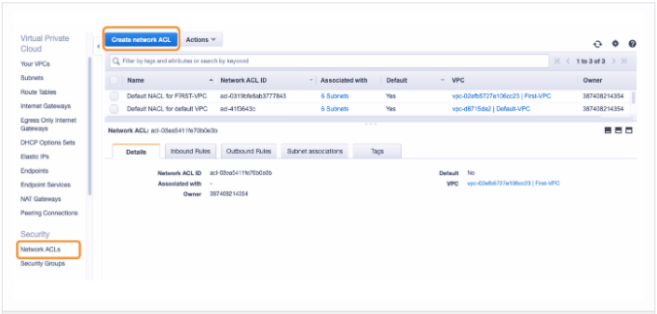
Network ACL Working Principle-2



Network ACL Working Principle

- So, when a package arrives;
  - The rules are viewed from top to bottom, in other words, from the smallest number to the highest number.
  - All the rules will be checking from the lowest numbered rule to the highest numbered rule respectively,
  - If one of these rules complies with the package, that rule is done.
  - If it does not comply with any rules, the package goes the bottom of the list.
  - At the bottom of the list, there is a default rule that doesn't allow the package to pass.

Create a Network ACL



Network ACL Menu

Let's create a new Network ACL and modify it according to our wishes.

- First, go to the **Network ACL** section from the left-hand menu on VPC Dashboard,
- Then, Click **Create Network ACL** tab on the top as you see in the picture above.

Create a Network ACL

- On the opening page;
  - Enter **Example Network ACL** as Name Tag,
  - Then select **First-VPC** or your VPC name as VPC
  - Finally click **Create**, and It's done. Our Network ACL is created.
- But, all inbound and outbound rules are **denied**. If you assign this Network ACL to any subnet in this state, there will be no connectivity for this subnet. So we need to modify it.

⚠ **Avoid ! :**
- As you remember, in Default Network ACL, all rules are allowed unlike newly created Network ACL

Modify the New Network ACL

Let's modify it according to our wishes. So;

- First select **inbound** tab and click **Edit Routes**. Then you'll see the page below and start to add a rule.

Add Inbound Rules

- As the first rule; If we want to reach our instance from **IP of 7.8.9.10/32** with **HTTP Protocol** trough **Port 80**,
  - For the Number of Rule : We determine the number of the rule as **100**,
  - For the Type Section : We select **HTTP Protocol**,
  - For the Source Section : We enter the IP of **7.8.9.10/32**,
  - For Allow/Deny Section : We select **Allow**
- As the second rule; if we want to reach our instance from **Any IP** with **SSH Protocol** trough **Port 22**,
  - For the Number of Rule : We determine the number of the rule as **200**,
  - For the Type Section : We select **SSH Protocol**,
  - For the Source Section : We select **Any** and enter the IP of **0.0.0.0/0** which means anywhere.
  - For Allow/Deny Section : We select **Allow**
- As the third rule; If we want to **ping** our instance from **Any IP** with **All ICMP - IPv4 Protocol** trough **All Port**,
  - For the Number of Rule : We determine the number of the rule as **300**,
  - For the Type Section : We select **All ICMP - IPv4**
  - For the Source Section : We select **Any** and enter the IP of **0.0.0.0/0** which means anywhere.
  - For Allow/Deny Sectio : We select **Allow**
- As the fourth rule; if we **don't want our instance to reachable** from **IP of 7.8.9.10/32** with **HTTPS Protocol** trough **Port 443**,
  - For the Number of Rule : We determine the number of the rule as **400**,
  - For the Type Section : We select **HTTPS Protocol**,
  - For the Source Section : We enter the IP of **7.8.9.10/32**,
  - For Allow/Deny Section : We select **Deny**
- And then, click **Save** tab for saving rules.
- After that, we set the **outbound** rules as we did for inbound.
- After modifying, if you look at the **inbound** rule;
  - You can see 4 rules that we have just created.
  - At the bottom of the list, there is one more rule that created by **default** and **denies all traffic** as seen in the picture below.

Rule #	Type	Protocol	Port Range	Source	Allow / Deny
100	HTTP (80)	TCP (6)	80	7.8.9.10/32	ALLOW
200	SSH (22)	TCP (6)	22	0.0.0.0/0	ALLOW
300	All ICMP - IPv4	ICMP (1)	ALL	0.0.0.0/0	ALLOW
400	HTTPS (443)	TCP (6)	443	7.8.9.10/32	DENY
*	ALL Traffic	ALL	ALL	0.0.0.0/0	DENY

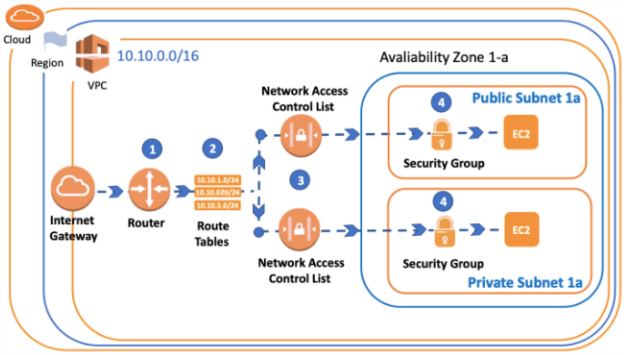
Inbound Rules

Conclusion 1: Security Groups vs. Network ACLs

	Security Group	Network Access Control List
Rules	It supports only **Allow Rules**	It supports **both Allow and Deny** rules
Default by AWS	By default, **inbound** rules are **Denied, Out bound** rules are **Allow**	By default, all the rules are **Allowed**
* **Newly Created by User**	By default, **inbound** rules are **Denied, Out bound** rules are **Allow**	By default, all the rules are **Denied*** until you add rules.
Add Rule	You need to add the rule which you'll **Allow**	You need to add the rule which you can **either Allow or Deny it**.
Stateful/Stateless	It is a **Stateful** means that any changes made in the inbound rule will be automatically reflected in the outbound rule	It is a **Stateless** means that any changes made in the inbound rule will not reflect the outbound rule
Association	1. It is **instance-based** 2. Instances can associate with **more than one** Security Groups	1. It is **subnet-based** 2. Subnets can **associate with only one** Network ACL

Security Groups vs. Network ACLs

Conclusion 2: Route Tables & Network ACLs & Security Groups



Route Tables & Network ACLs & Security Groups