Security

Amazon VPC provides features that we can use to increase and monitor the security for our VPC:

- Security groups — Act as a firewall for associated Amazon EC2 instances, controlling both inbound and outbound traffic at the instance level
- Network access control lists (ACLs) — Act as a firewall for associated subnets, controlling both inbound and outbound traffic at the subnet level
- Flow logs — Capture information about the IP traffic going to and from network interfaces in our VPC

When we launch an instance in a VPC, we can associate one or more security groups that we have created. Each instance in our VPC could belong to a different set of security groups. If we do not specify a security group when we launch an instance, the instance automatically belongs to the default security group for the VPC.

We can secure our VPC instances using only security groups; however, we can add network ACLs as a second layer of defense.

We can monitor the accepted and rejected IP traffic going to and from our instances by creating a flow log for a VPC, subnet or individual network interface. Flow log data is published to CloudWatch Logs and can help us diagnose overly restrictive or overly permissive security group and network ACL rules.

We can use AWS Identity and Access Management to control who in our organization has permission to create and manage security groups, network ACLs and flow logs. For example, we can give only our network administrators that permission, but not personnel who only need to launch instances.

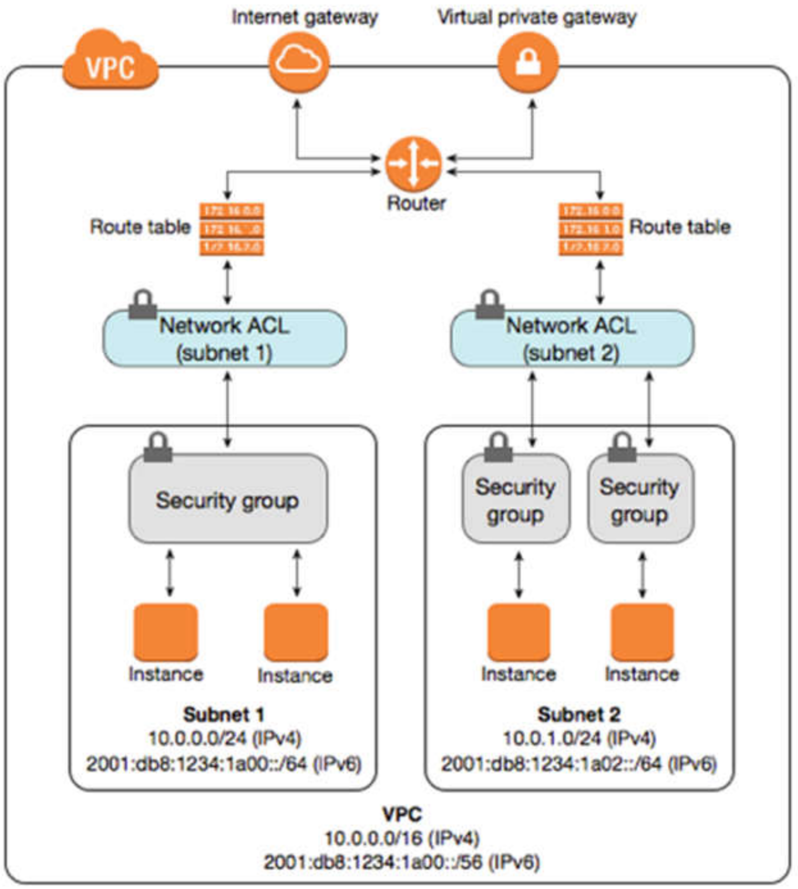
Amazon security groups and network ACLs do not filter traffic to or from link-local addresses or AWS-reserved IPv4 addresses—these are the first four IPv4 addresses of the subnet (including the Amazon DNS server address for the VPC). Similarly, flow logs do not capture IP traffic to or from these addresses. These addresses support the services: Domain Name Services (DNS), Dynamic Host Configuration Protocol (DHCP), Amazon EC2 instance metadata and routing in the subnet. We can implement additional firewall solutions in our instances to block network communication with link-local addresses.

Comparison of Security Groups and Network ACLs

The following table summarizes the basic differences between security groups and network ACLs.

Security Group	**Network ACL**
Operates at the instance level (first layer of defense)	Operates at the subnet level (second layer of defense)
Supports allow rules only	Supports allow rules and deny rules
Is stateful: Return traffic is automatically allowed, regardless of any rules	Is stateless: Return traffic must be explicitly allowed by rules
We evaluate all rules before deciding whether to allow traffic (based on PORT number)	We process rules in number order when deciding whether to allow traffic
Applies to an instance only if someone specifies the security group when launching the instance, or associates the security group with the instance later on	Automatically applies to all instances in the subnets it's associated with (backup layer of defense, so we don't have to rely on someone specifying the security group)

The following diagram illustrates the layers of security provided by security groups and network ACLs. For example, traffic from an Internet gateway is routed to the appropriate subnet using the routes in the routing table. The rules of the network ACL associated with the subnet will control which traffic is allowed to the subnet. The rules of the security group associated with an instance will control which traffic is allowed to the instance.



Security Groups for our VPC

A *security group* acts as a virtual firewall for our instance to control inbound and outbound traffic. When we launch an instance in a VPC, we can assign up to five security groups to the instance. Security groups act at the instance level, not the subnet level. Therefore, each instance in a subnet in our VPC could be assigned to a different set of security groups. If we do not specify a particular group at launch time, the instance is automatically assigned to the default security group for the VPC.

For each security group, we add *rules* that control the inbound traffic to instances and a separate set of rules that control the outbound traffic. This section describes the basic things we need to know about security groups for our VPC and their rules.

We can set up network ACLs with rules similar to our security groups in order to add an additional layer of security to our VPC.

Security Group Basics

The following are the basic characteristics of security groups for our VPC:

- We have limits on the number of security groups that we can create per VPC, the number of rules that we can add to each security group, and the number of security groups we can associate with a network interface.
- We can specify allow rules, but not deny rules.
- We can specify separate rules for inbound and outbound traffic.
- When we create a security group, it has no inbound rules. Therefore, no inbound traffic originating from another host to our instance is allowed until we add inbound rules to the security group.
- By default, a security group includes an outbound rule that allows all outbound traffic. We can remove the rule and add outbound rules that allow specific outbound traffic only. If our security group has no outbound rules, no outbound traffic originating from our instance is allowed.
- Security groups are stateful — if we send a request from our instance, the response traffic for that request is allowed to flow in regardless of inbound security group rules. Responses to allowed inbound traffic are allowed to flow out, regardless of outbound rules.
- Instances associated with a security group cannot talk to each other unless we add rules allowing it (exception: the default security group has these rules by default).
- Security groups are associated with network interfaces. After we launch an instance, we can change the security groups associated with the instance, which changes the security groups associated with the primary network interface (eth0). We can also change the security groups associated with any other network interface.
- When we create a security group, we must provide it with a name and a description. The following rules apply:
    - Names and descriptions can be up to 255 characters in length.

- Names and descriptions are limited to the following characters: a-z, A-Z, 0-9, spaces, and ._-:/()#,@[]+=&;{}!$*.
- A security group name cannot start with sg-.
- A security group name must be unique within the VPC.

Default Security Group for our VPC

Our VPC automatically comes with a default security group. Each EC2 instance that we launch in our VPC is automatically associated with the default security group if we do not specify a different security group when we launch the instance.

The following table describes the default rules for a default security group.

Inbound			
Source	**Protocol**	**Port Range**	**Comments**
The security group ID (sg-*xxxxxxxx*)	All	All	Allow inbound traffic from instances assigned to the same security group.
Outbound			
Destination	**Protocol**	**Port Range**	**Comments**
0.0.0.0/0	All	All	Allow all outbound IPv4 traffic.
::/0	All	All	Allow all outbound IPv6 traffic. This rule is added by default if we create a VPC with an IPv6 CIDR block or if we associate an IPv6 CIDR block with our existing VPC.

We can change the rules for the default security group.

We cannot delete a default security group. If we try to delete the default security group, we will get the following error: *Client.CannotDelete: the specified group: "sg-51530134" name: "default" cannot be deleted by a user.*

Note

If we have modified the outbound rules for our security group, we do not automatically add an outbound rule for IPv6 traffic when we associate an IPv6 block with our VPC.

Security Group Rules

We can add or remove rules for a security group (also referred to as *authorizing* or *revoking* inbound or outbound access). A rule applies either to inbound traffic (ingress) or outbound traffic (egress). We can grant access to a specific CIDR range or to another security group in our VPC or in a peer VPC (requires a VPC peering connection).

The following are the basic parts of a security group rule in a VPC:

- (Inbound rules only) The source of the traffic and the destination port or port range. The source can be another security group, an IPv4 or IPv6 CIDR block, or a single IPv4 or IPv6 address.
- (Outbound rules only) The destination for the traffic and the destination port or port range. The destination can be another security group, an IPv4 or IPv6 CIDR block, or a single IPv4 or IPv6 address.
- Any protocol that has a standard protocol number. If we specify ICMP as the protocol, we can specify any or all of the ICMP types and codes.
- An optional description for the security group rule to help us identify it later. A description can be up to 255 characters in length. Allowed characters are a-z, A-Z, 0-9, spaces, and ._-:/()#,@[]+=;{}!$*.

When we specify a security group as the source for a rule, this allows instances associated with the source security group to access instances in the security group. This does not add rules from the source security group to this security group. Incoming traffic is allowed based on the private IP addresses of the instances that are associated with the source security group (and not the public IP or Elastic IP addresses).

If we specify a single IPv4 address, specify the address using the /32 prefix length. If we specify a single IPv6 address, specify it using the /128 prefix length.

Some systems for setting up firewalls let us filter on source ports. Security groups let us filter only on destination ports.

When we add or remove rules, they are automatically applied to all instances associated with the security group.

The kind of rules we add may depend on the purpose of the instance. The following table describes example rules for a security group for web servers. The web servers can receive HTTP and HTTPS traffic from all IPv4 and IPv6 addresses, and send SQL or MySQL traffic to a database server.

Inbound			
Source	**Protocol**	**Port Range**	**Comments**
0.0.0.0/0	TCP	80	Allow inbound HTTP access from all IPv4 addresses
::/0	TCP	80	Allow inbound HTTP access from all IPv6 addresses
0.0.0.0/0	TCP	443	Allow inbound HTTPS access from all IPv4 addresses
::/0	TCP	443	Allow inbound HTTPS access from all IPv6 addresses

Our network's public IPv4 address range	TCP	22	Allow inbound SSH access to Linux instances from IPv4 IP addresses in our network (over the Internet gateway)
Our network's public IPv4 address range	TCP	3389	Allow inbound RDP access to Windows instances from IPv4 IP addresses in our network (over the Internet gateway)
Outbound			
Destination	**Protocol**	**Port Range**	**Comments**
The ID of the security group for our database servers	TCP	1433	Allow outbound Microsoft SQL Server access to instances in the specified security group
The ID of the security group for our MySQL database servers	TCP	3306	Allow outbound MySQL access to instances in the specified security group

A database server would need a different set of rules; for example, instead of inbound HTTP and HTTPS traffic, we can add a rule that allows inbound MySQL or Microsoft SQL Server access.

Modifying the Default Security Group

Our VPC includes a default security group whose initial rules are to deny all inbound traffic, allow all outbound traffic, and allow all traffic between instances in the group. We can't delete this group; however, we can change the group's rules. The procedure is the same as modifying any other security group.

Creating a Security Group

Although we can use the default security group for our instances, we might want to create our own groups to reflect the different roles that instances play in our system.

To create a security group using the console

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/
2. In the navigation pane, choose **Security Groups**.
3. Choose **Create Security Group**.
4. Enter a name of the security group (for example, my-security-group) and provide a description. Select the ID of our VPC from the **VPC** menu and choose **Yes, Create**.

By default, new security groups start with only an outbound rule that allows all traffic to leave the instances. We must add rules to enable any inbound traffic or to restrict the outbound traffic.

Adding, Removing, and Updating Rules

When we add or remove a rule, any instances already assigned to the security group are subject to the change.

To add a rule using the console

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Security Groups**.
3. Select the security group to update. The details pane displays the details for the security group, plus tabs for working with its inbound rules and outbound rules.
4. On the **Inbound Rules** tab, choose **Edit**. Select an option for a rule for inbound traffic for **Type**, and then fill in the required information. For example, for a public web server, choose **HTTP** or **HTTPS** and specify a value for **Source** as 0.0.0.0/0.

   Note

   If we use 0.0.0.0/0, we enable all IPv4 addresses to access our instance using HTTP or HTTPS. To restrict access, enter a specific IP address or range of addresses.

5. Optionally provide a description for the rule and choose **Save**.
6. We can also allow communication between all instances associated with this security group. On the **Inbound Rules** tab, choose **All Traffic** from the **Type** list. Start typing the ID of the security group for **Source**; this provides us with a list of security groups. Select the security group from the list and choose **Save**.
7. If we need to, we can use the **Outbound Rules** tab to add rules for outbound traffic.

To delete a rule

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/
2. In the navigation pane, choose **Security Groups**.
3. Select the security group to update. The details pane displays the details for the security group, plus tabs for working with its inbound rules and outbound rules.
4. Choose **Edit**, select the role to delete, and then choose **Remove**, **Save**.

When we modify the protocol, port range, or source or destination of an existing security group rule using the console, the console deletes the existing rule and adds a new one for us.

To update a rule using the console

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/
2. In the navigation pane, choose **Security Groups**.
3. Select the security group to update, and choose **Inbound Rules** to update a rule for inbound traffic or **Outbound Rules** to update a rule for outbound traffic.
4. Choose **Edit**. Modify the rule entry as required and choose **Save**.

Changing an Instance's Security Groups

After we launch an instance into a VPC, we can change the security groups that are associated with the instance. We can change the security groups for an instance when the instance is in the running or stopped state.

Note

This procedure changes the security groups that are associated with the primary network interface (eth0) of the instance.

To change the security groups for an instance using the console

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/
2. In the navigation pane, choose **Instances**.
3. Open the context (right-click) menu for the instance and choose **Networking**, **Change Security Groups**.
4. In the **Change Security Groups** dialog box, select one or more security groups from the list and choose **Assign Security Groups**.

Deleting a Security Group

We can delete a security group only if there are no instances assigned to it (either running or stopped). We can assign the instances to another security group before we delete the security group. We cannot delete a default security group.

We can delete more than one security group at a time.

To delete a security group using the console

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/
2. In the navigation pane, choose **Security Groups**.
3. Select one or more security groups and choose **Security Group Actions**, **Delete Security Group**.
4. In the **Delete Security Group** dialog box, choose **Yes, Delete**.

Network ACLs

A *network access control list (ACL)* is an optional layer of security for our VPC that acts as a firewall for controlling traffic in and out of one or more subnets. We might set up network ACLs with rules similar to our security groups in order to add an additional layer of security to our VPC.

Network ACL Basics

The following are the basic things that we need to know about network ACLs:

- Our VPC automatically comes with a modifiable default network ACL. By default, it allows all inbound and outbound IPv4 traffic and, if applicable, IPv6 traffic.
- We can create a custom network ACL and associate it with a subnet. By default, each custom network ACL denies all inbound and outbound traffic until we add rules.
- Each subnet in our VPC must be associated with a network ACL. If we don't explicitly associate a subnet with a network ACL, the subnet is automatically associated with the default network ACL.
- We can associate a network ACL with multiple subnets; however, a subnet can be associated with only one network ACL at any given time. When we associate a network ACL with a subnet, the previous association is removed.
- A network ACL contains a numbered list of rules that we evaluate in order, starting with the lowest numbered rule, to determine whether traffic is allowed in or out of any subnet associated with the network ACL. The highest number that we can use for a rule is 32766. It is recommended that we start by creating rules with rule numbers that are multiples of 100, so that we can insert new rules where we need to later on.
- A network ACL has separate inbound and outbound rules, and each rule can either allow or deny traffic.
- Network ACLs are stateless; responses to allowed inbound traffic are subject to the rules for outbound traffic (and vice versa).

Network ACL Rules

We can add or remove rules from the default network ACL, or create additional network ACLs for our VPC. When we add or remove rules from a network ACL, the changes are automatically applied to the subnets 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 may contradict it.
- Protocol. We can specify any protocol that has a standard protocol number.
- [Inbound rules only] The source of the traffic (CIDR range) and the destination (listening) port or port range.
- [Outbound rules only] The destination for the traffic (CIDR range) and the destination port or port range.
- Choice of ALLOW or DENY for the specified traffic.

Default Network ACL

The default network ACL is configured to allow all traffic to flow in and out of the subnets to which it is associated. Each network ACL also includes a rule whose rule number is an asterisk. This rule ensures that if a packet doesn't match any of the other numbered rules, it's denied. We can't modify or remove this rule.

The following is an example default network ACL for a VPC that supports IPv4 only.

Inbound					
Rule #	**Type**	**Protocol**	**Port Range**	**Source**	**Allow/Deny**
100	All IPv4 traffic	All	All	0.0.0.0/0	ALLOW
*	All IPv4 traffic	All	All	0.0.0.0/0	DENY
Outbound					
Rule #	**Type**	**Protocol**	**Port Range**	**Destination**	**Allow/Deny**
100	All IPv4 traffic	all	all	0.0.0.0/0	ALLOW
*	All IPv4 traffic	all	all	0.0.0.0/0	DENY

If we create a VPC with an IPv6 CIDR block or if we associate an IPv6 CIDR block with our existing VPC, we automatically add rules that allow all IPv6 traffic to flow in and out of our subnet. We also add rules whose rule numbers are an asterisk that ensures that a packet is denied if it doesn't match any of the other numbered rules. We can't modify or remove these rules. The following is an example default network ACL for a VPC that supports IPv4 and IPv6.

Note

If we have modified our default network ACL's inbound rules, we do not automatically add an ALLOW rule for inbound IPv6 traffic when we associate an IPv6 block with our VPC. Similarly, if we have modified the outbound rules, we do not automatically add an ALLOW rule for outbound IPv6 traffic.

Inbound					
Rule #	**Type**	**Protocol**	**Port Range**	**Source**	**Allow/Deny**
100	All IPv4 traffic	All	All	0.0.0.0/0	ALLOW
101	All IPv6 traffic	All	All	::/0	ALLOW
*	All traffic	All	All	0.0.0.0/0	DENY
*	All IPv6 traffic	All	All	::/0	DENY
Outbound					

Rule #	Type	Protocol	Port Range	Destination	Allow/Deny
100	All traffic	all	all	0.0.0.0/0	ALLOW
*	All traffic	all	all	0.0.0.0/0	DENY

Ephemeral Ports

The example network ACL in the preceding section uses an ephemeral port range of 32768-65535. However, we might want to use a different range for our network ACLs depending on the type of client that we are using or with which we are communicating.

The client that initiates the request chooses the ephemeral port range. The range varies depending on the client's operating system. Many Linux kernels (including the Amazon Linux kernel) use ports 32768-61000. Requests originating from Elastic Load Balancing use ports 1024-65535. Windows operating systems through Windows Server 2003 use ports 1025-5000. Windows Server 2008 and later versions use ports 49152-65535. A NAT gateway uses ports 1024-65535. For example, if a request comes into a web server in our VPC from a Windows XP client on the Internet, our network ACL must have an outbound rule to enable traffic destined for ports 1025-5000.

If an instance in our VPC is the client initiating a request, our network ACL must have an inbound rule to enable traffic destined for the ephemeral ports specific to the type of instance (Amazon Linux, Windows Server 2008, and so on).

In practice, to cover the different types of clients that might initiate traffic to public-facing instances in our VPC, we can open ephemeral ports 1024-65535. However, we can also add rules to the ACL to deny traffic on any malicious ports within that range. Ensure that we place the DENY rules earlier in the table than the ALLOW rules that open the wide range of ephemeral ports.

Working with Network ACLs

Determining Network ACL Associations

We can use the Amazon VPC console to determine the network ACL that's associated with a subnet. Network ACLs can be associated with more than one subnet, so we can also determine the subnets that are associated with a network ACL.

To determine which network ACL is associated with a subnet

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/
2. In the navigation pane, choose **Subnets**, and then select the subnet.

   The network ACL associated with the subnet is included in the **Network ACL** tab, along with the network ACL's rules.

To determine which subnets are associated with a network ACL

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/
2. In the navigation pane, choose **Network ACLs**. The **Associated With** column indicates the number of associated subnets for each network ACL.

3. Select a network ACL.
4. In the details pane, choose **Subnet Associations** to display the subnets associated with the network ACL.

Creating a Network ACL

We can create a custom network ACL for our VPC. By default, a network ACL that we create blocks all inbound and outbound traffic until we add rules, and is not associated with a subnet until we explicitly associate it with one.

To create a network ACL

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Network ACLs**.
3. Choose **Create Network ACL**.
4. In the **Create Network ACL** dialog box, optionally name our network ACL, and then select the ID of our VPC from the **VPC** list, and choose **Yes, Create**.

Adding and Deleting Rules

When we add or delete a rule from an ACL, any subnets associated with the ACL are subject to the change. We don't have to terminate and relaunch the instances in the subnet; the changes take effect after a short period.

To add rules to a network ACL

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Network ACLs**.
3. In the details pane, choose either the **Inbound Rules** or **Outbound Rules** tab, depending on the type of rule that we need to add, and then choose **Edit**.
4. In **Rule #**, enter a rule number (for example, 100). The rule number must not already be used in the network ACL. We process the rules in order, starting with the lowest number.

   Tip

   It is recommended that we leave gaps between the rule numbers (such as 100, 200, 300), rather than using sequential numbers (101, 102, 103). This makes it easier add a new rule without having to renumber the existing rules.

5. Select a rule from the **Type** list. For example, to add a rule for HTTP, choose **HTTP**. To add a rule to allow all TCP traffic, choose **All TCP**. For some of these options (for example, HTTP), we fill in the port for us. To use a protocol that's not listed, choose **Custom Protocol Rule**.
6. (Optional) If we are creating a custom protocol rule, select the protocol's number and name from the **Protocol** list. For more information, see IANA List of Protocol Numbers.
7. (Optional) If the protocol we have selected requires a port number, enter the port number or port range separated by a hyphen (for example, 49152-65535).

8. In the **Source** or **Destination** field (depending on whether this is an inbound or outbound rule), enter the CIDR range that the rule applies to.
9. From the **Allow/Deny** list, select **ALLOW** to allow the specified traffic or **DENY** to deny the specified traffic.
10. (Optional) To add another rule, choose **Add another rule**, and repeat steps 4 to 9 as required.
11. When we are done, choose **Save**.

To delete a rule from a network ACL

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Network ACLs**, and then select the network ACL.
3. In the details pane, select either the **Inbound Rules** or **Outbound Rules** tab, and then choose **Edit**. Choose **Remove** for the rule we want to delete, and then choose **Save**.

Associating a Subnet with a Network ACL

To apply the rules of a network ACL to a particular subnet, we must associate the subnet with the network ACL. We can associate a network ACL with multiple subnets; however, a subnet can be associated with only one network ACL. Any subnet not associated with a particular ACL is associated with the default network ACL by default.

To associate a subnet with a network ACL

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Network ACLs**, and then select the network ACL.
3. In the details pane, on the **Subnet Associations** tab, choose **Edit**. Select the **Associate** check box for the subnet to associate with the network ACL, and then choose **Save**.

Disassociating a Network ACL from a Subnet

We can disassociate a custom network ACL from a subnet — by doing so, the subnet is then automatically associated with the default network ACL.

To disassociate a subnet from a network ACL

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Network ACLs**, and then select the network ACL.
3. In the details pane, choose the **Subnet Associations** tab.
4. Choose **Edit**, and then deselect the **Associate** check box for the subnet. Choose **Save**.

Changing a Subnet's Network ACL

We can change the network ACL that's associated with a subnet. For example, when we create a subnet, it is initially associated with the default network ACL. We might want to instead associate it with a custom network ACL that we have created.

After changing a subnet's network ACL, we don't have to terminate and relaunch the instances in the subnet; the changes take effect after a short period.

To change a subnet's network ACL association

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Subnets**, and then select the subnet.
3. Choose the **Network ACL** tab, and then choose **Edit**.
4. Select the network ACL to associate the subnet with from the **Change to** list, and then choose **Save**.

Deleting a Network ACL

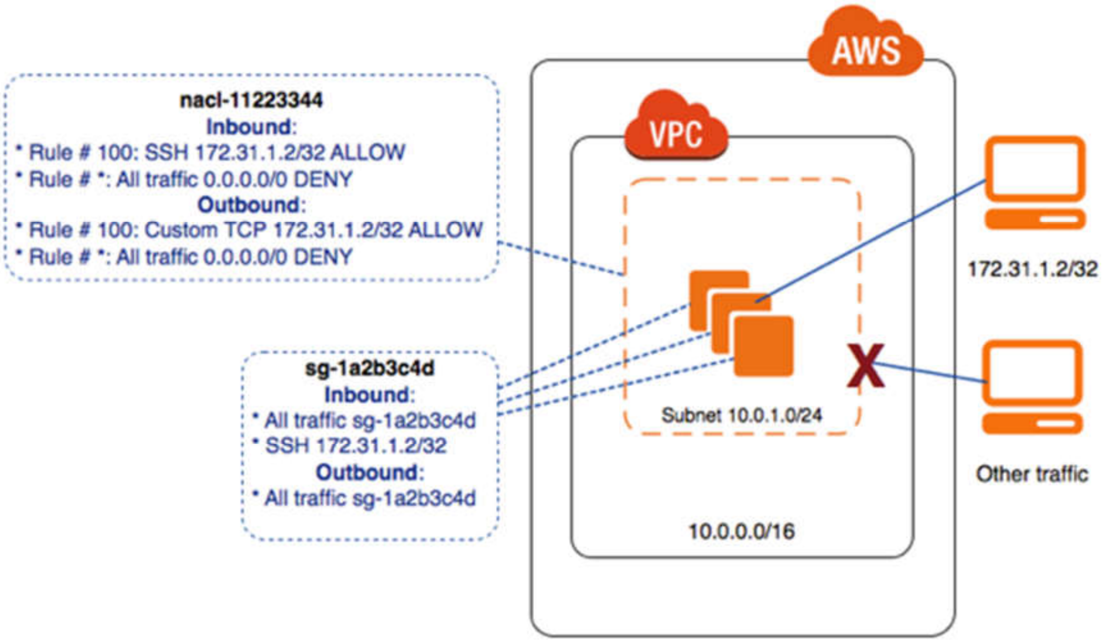
We can delete a network ACL only if there are no subnets associated with it. We can't delete the default network ACL.

To delete a network ACL

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. In the navigation pane, choose **Network ACLs**.
3. Select the network ACL, and then choose **Delete**.
4. In the confirmation dialog box, choose **Yes, Delete**.

Example: Controlling Access to Instances in a Subnet

In this example, instances in our subnet can communicate with each other, and are accessible from a trusted remote computer. The remote computer may be a computer in our local network or an instance in a different subnet or VPC that we use to connect to our instances to perform administrative tasks. Our security group rules and network ACL rules allow access from the IP address of our remote computer (172.31.1.2/32). All other traffic from the Internet or other networks is denied.


nacl-11223344
Inbound:
• Rule # 100: SSH 172.31.1.2/32 ALLOW
• Rule # *: All traffic 0.0.0.0/0 DENY
Outbound:
• Rule # 100: Custom TCP 172.31.1.2/32 ALLOW
• Rule # *: All traffic 0.0.0.0/0 DENY
sg-1a2b3c4d
Inbound:
• All traffic sg-1a2b3c4d
• SSH 172.31.1.2/32
Outbound:
• All traffic sg-1a2b3c4d
AWS
VPC
Subnet 10.0.1.0/24
10.0.0.0/16
X
172.31.1.2/32
Other traffic

All instances use the same security group (sg-1a2b3c4d), with the following rules.

Inbound Rules				
Protocol Type	**Protocol**	**Port Range**	**Source**	**Comments**
All traffic	All	All	sg-1a2b3c4d	Enables instances associated with the same security group to communicate with each other.
TCP	SSH	22	172.31.1.2/32	Allows inbound SSH access from the remote computer. If the instance is a Windows computer, then this rule must use the RDP protocol for port 3389 instead.
Outbound Rules				
Protocol Type	**Protocol**	**Port Range**	**Destination**	**Comments**
All traffic	All	All	sg-1a2b3c4d	Enables instances associated with the same security group to communicate with each other.

The subnet is associated with a network ACL that has the following rules.

Inbound Rules						
Rule #	**Type**	**Protocol**	**Port Range**	**Source**	**Allow/Deny**	**Comments**

100	SSH	TCP	22	172.31.1.2/32	ALLOW	Allows inbound traffic from the remote computer. If the instance is a Windows computer, then this rule must use the RDP protocol for port 3389 instead.
*	All traffic	All	All	0.0.0.0/0	DENY	Denies all other inbound traffic that does match the previous rule.
Outbound Rules						
Rule #	**Type**	**Protocol**	**Port Range**	**Destination**	**Allow/Deny**	**Comments**
100	Custom TCP	TCP	1024-65535	172.31.1.2/32	ALLOW	Allows outbound responses to the remote computer. Network ACLs are stateless, therefore this rule is required to allow response traffic for inbound requests.
*	All traffic	All	All	0.0.0.0/0	DENY	Denies all other outbound traffic that does not match the previous rule.

This scenario gives us the flexibility to change the security groups or security group rules for our instances and have the network ACL as the backup layer of defense. The network ACL rules apply to all instances in the subnet, so if we accidentally make our security group rules too permissive, the network ACL rules continue to permit access only from the single IP address. For example, the following rules are more permissive than the earlier rules — they allow inbound SSH access from any IP address.

Inbound Rules				
Type	**Protocol**	**Port Range**	**Source**	**Comments**
All traffic	All	All	sg-1a2b3c4d	Enables instances associated with the same security group to communicate with each other.
SSH	TCP	22	0.0.0.0/0	Allows SSH access from any IP address.
Outbound Rules				
Type	**Protocol**	**Port Range**	**Destination**	**Comments**
All traffic	All	All	0.0.0.0/0	Allows all outbound traffic.

However, only other instances within the subnet and our remote computer are able to access this instance. The network ACL rules still prevent all inbound traffic to the subnet except from our remote computer.