

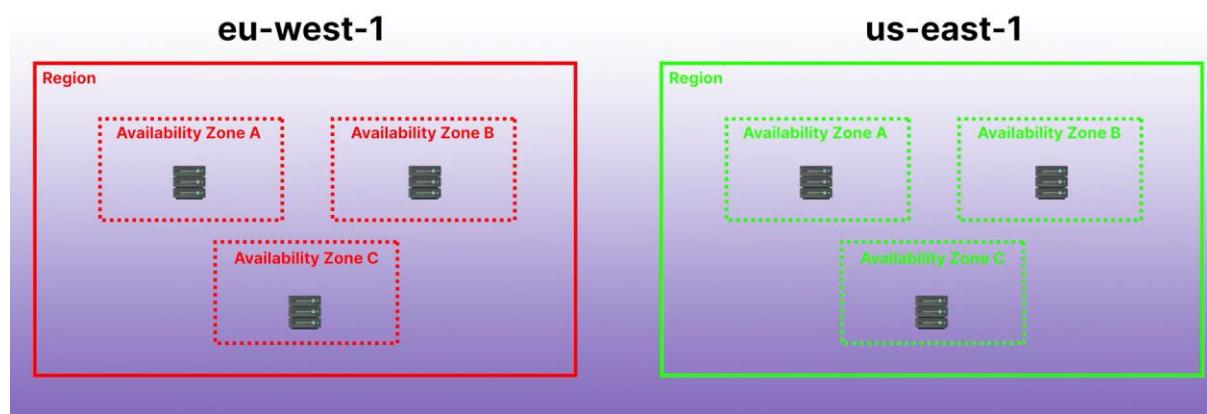
Week 5. AWS Fundamentals

제 1강. AWS Global Infrastructure

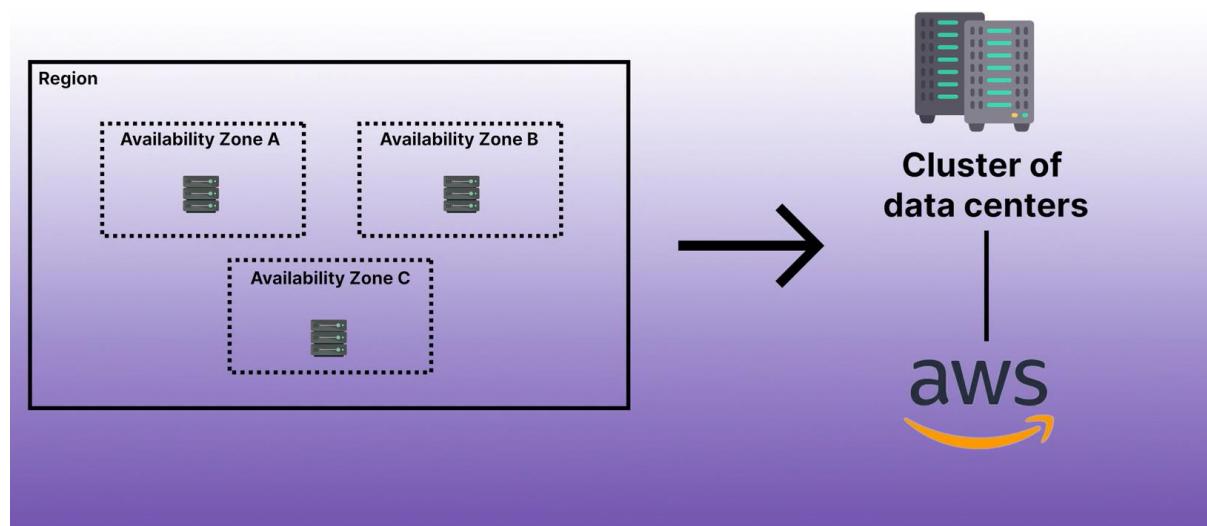
- AWS Global Infrastructure enables highly-available, scalable and reliable services to millions of users worldwide.
- Digital services and applications running on AWS must run in 24 hours no matter where you are.

1. AWS regions

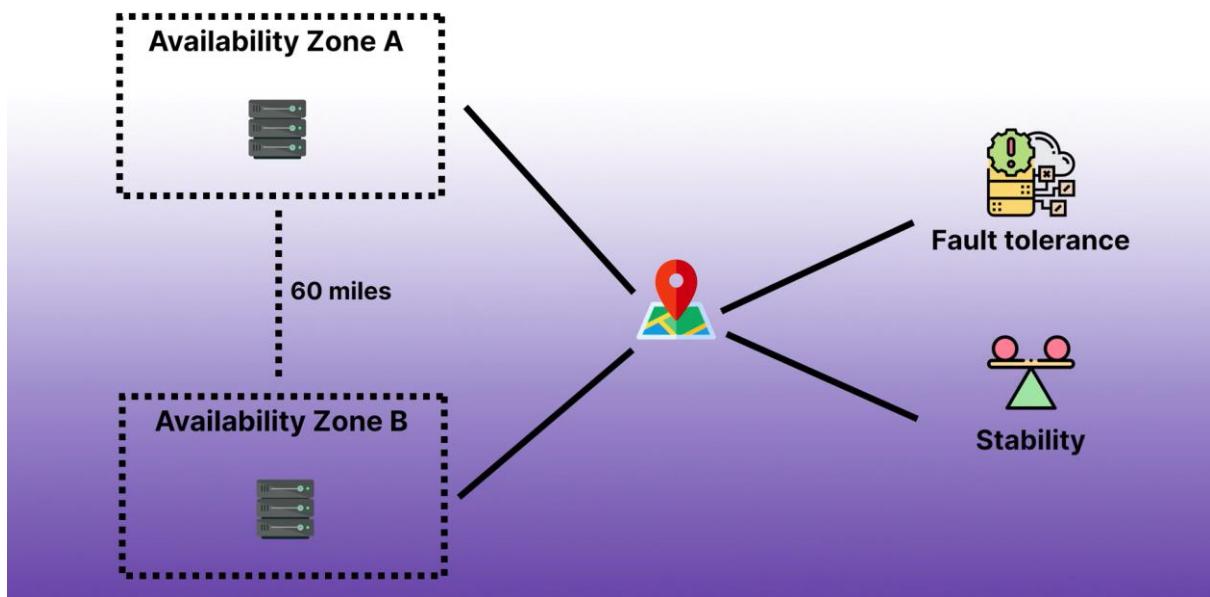
- AWS regions are global network of power stations, serving a specific geographical area which are independent one another. For example, eu-west-1 region does not affect us-east-1.



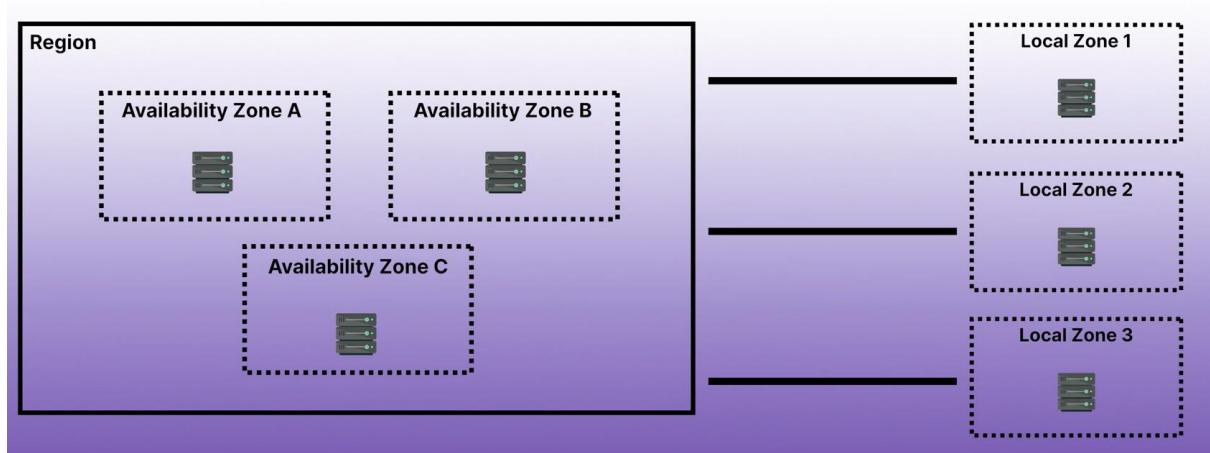
- This independence enables AWS to provide resilient services for Netflix.
- Each region is composed of the cluster of data centers.



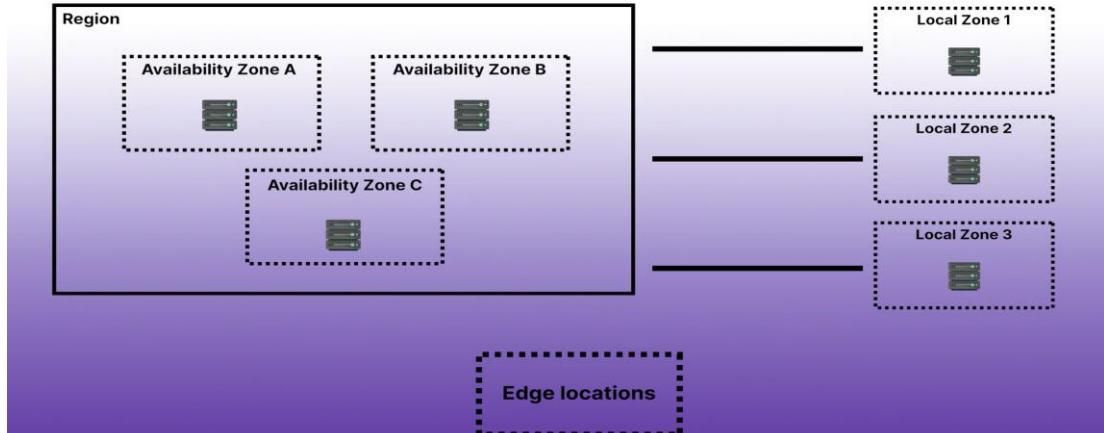
- There are multiple availability zones which are power generators within a power station. Each AZ has its own data centers with redundant power, networking and connectivity, allowing for seamless and reliable services, even if there is any issue.
- Availability zones are robust engines operating a power station, even if one engine encounters a problem.
- Each AZ is 60 miles apart from each other, providing geographical dispersions for fault tolerance and stability.



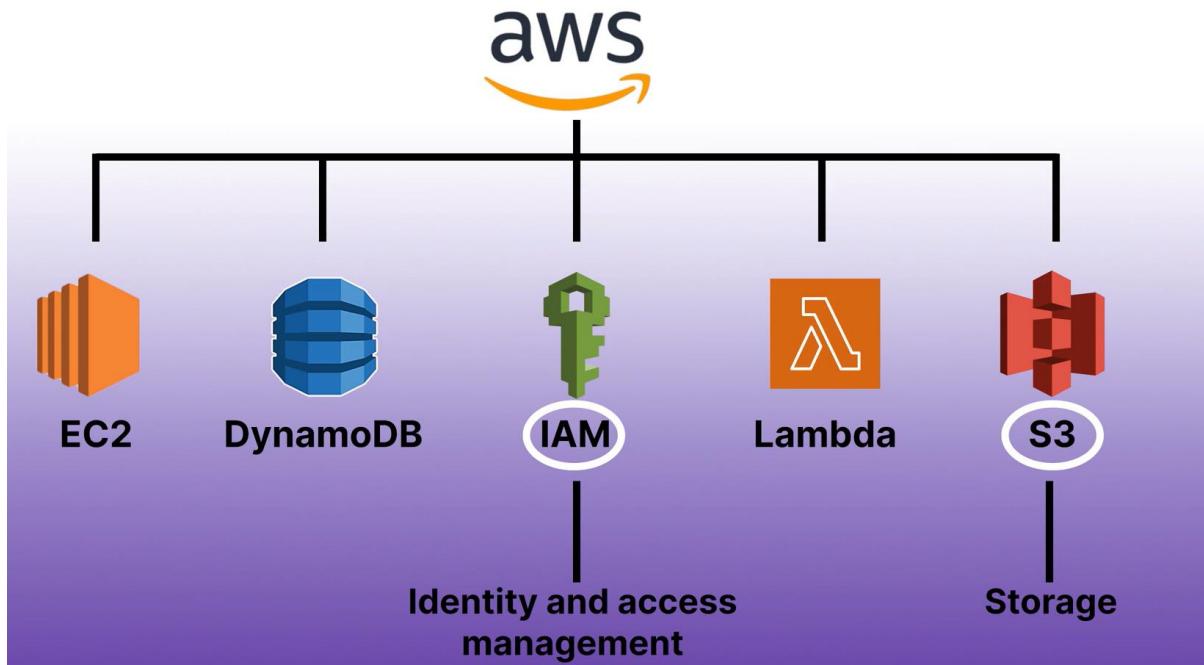
- We also have smaller and localized local zones, which are kinds of power stations extended from the main power station, region to bring services closer to users.



- Local zones are designed for latency-sensitive applications, ensuring users to access core AWS services fast and reliably.
- AWS also has edge locations, which is a kind of final transformer and distribution line of electricity for homes or businesses.



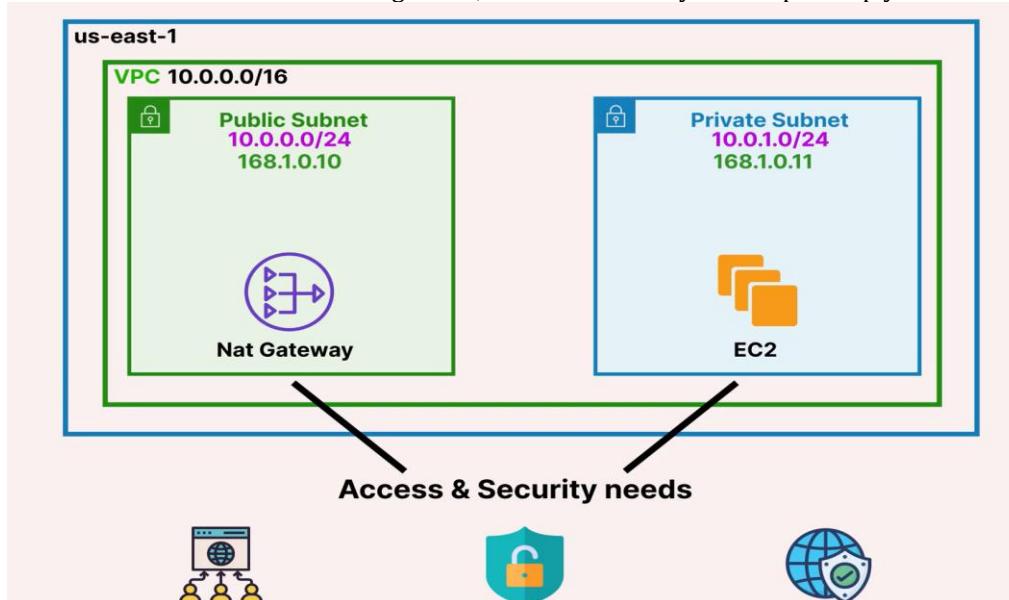
- Edge locations cache data closer to users for even lower latency, related to Amazon CloudFront, which is the AWS's last-mile delivery for users, allowing for the quickest access to data as needed.



- Some AWS services are confined to specific regions but for example, IAM and S3 are global services not constrained by geographic locations.
- AWS regions have a specific naming convention such as eu-west-2, identifying a specific geographic locations and corresponding available services.
- AWS clients are able to create an application, highly available, fault-tolerance and scaling globally by utilizing AWS global infrastructure such as regions, multi-AZs deployment, local zones and edge locations.

제 2강. Public vs Private Subnets

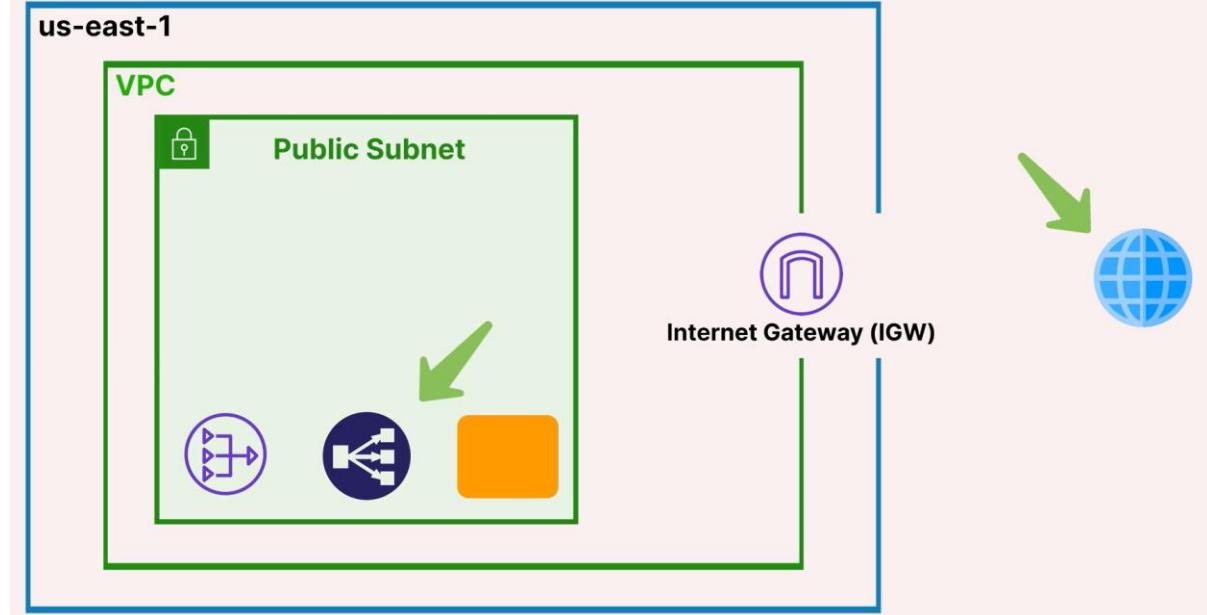
- There are public subnets and private subnets in VPC, which are designated areas serving specific roles of functions. They sub-divisions of your VPC IP address ranges, that allows segments and organize cloud resources based on varying levels of access and security needs.
- This allows efficient traffic management, enhances security and helps comply with network policy.



- The primary distinctions between public subnet and private subnet are accessibility from the Internet and role in network architecture.

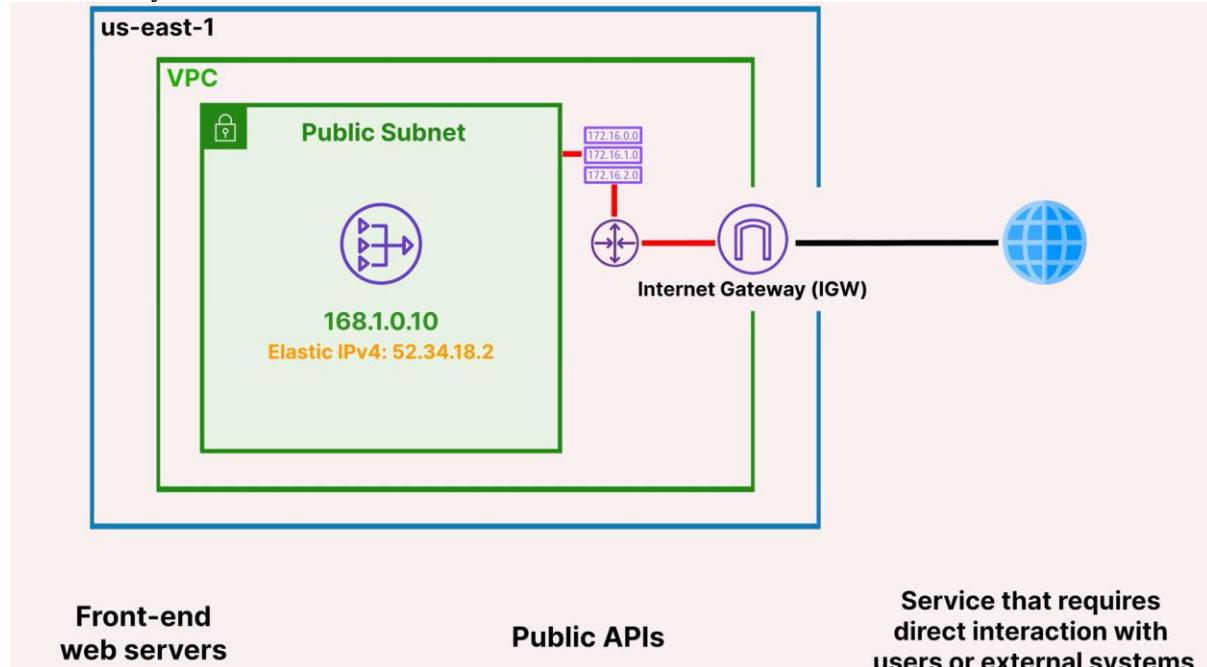
(1) Public subnet

- Resources in a public subnet can directly access through the internet. This accessibility is enabled by Internet Gateway. VPC provides a route for communication between resources in the public subnet and the internet.



- Public IP address or Elastic IPv4 must be assigned to a public subnet or resource in the subnet in order for them to be accessed by the internet.

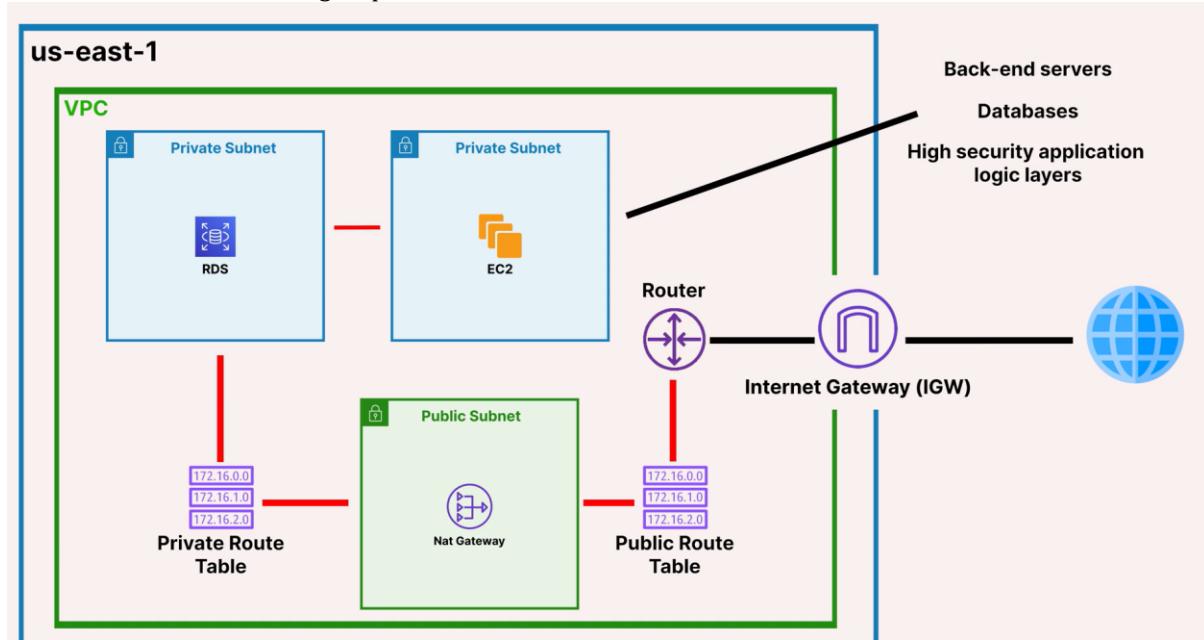
- This public subnet must have a route table containing a route connected to the Internet Gateway. A public subnet is ideal for front-end web servers, public APIs and service that requires direct interaction with users or external systems.



(2) Private subnet

- A private subnet is ideal for resources that should not be accessed directly from the internet.
- Resources in each private subnet and in a public subnet can communicate with each other but this private route table cannot access the Internet Gateway directly. Instead, there is a router allowing public resources

to access the Internet through a public route table.



- This router only allows outbound traffic from resources but not allow any inbound traffic from the internet.
- A private subnet is used for back-end servers, databases and high-security application logic layers that can be accessed via the public subnet indirectly.

(1) Enhance security

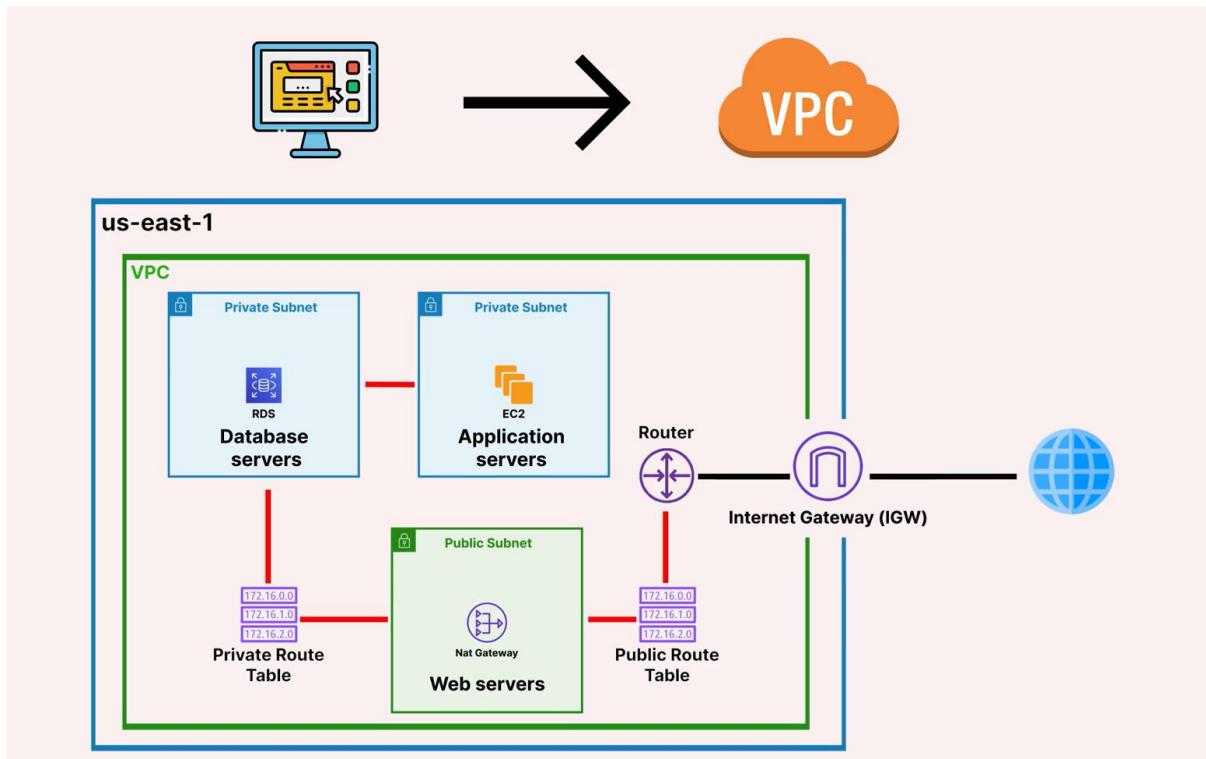
- The organized VPC structure design is crucial to enhance security for sensitive and critical workloads and make them separate from public-facing services to prevent unauthorized access.

(2) Improve performance

- VPC can localize traffic to minimize latency and maximize throughput by making applications servers to communicate with closely-related services such as databases.

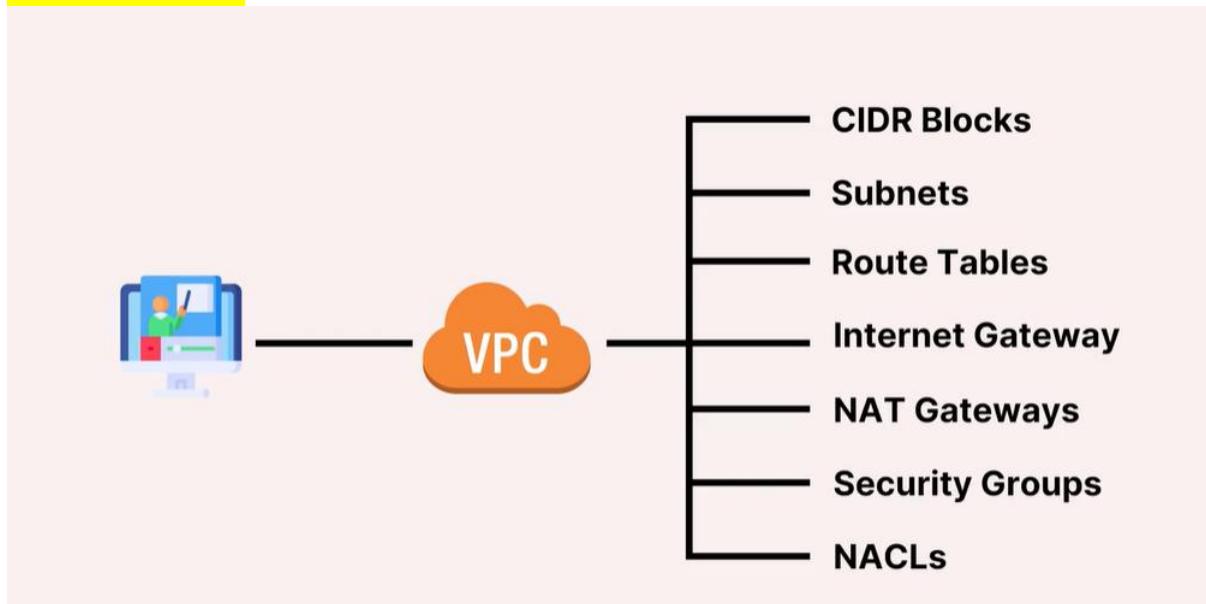
(3) Simplify Management

- We can adhere clear roles and policies to each subnet to facilitate the management of access controls, monitoring traffic and data governance requirements.

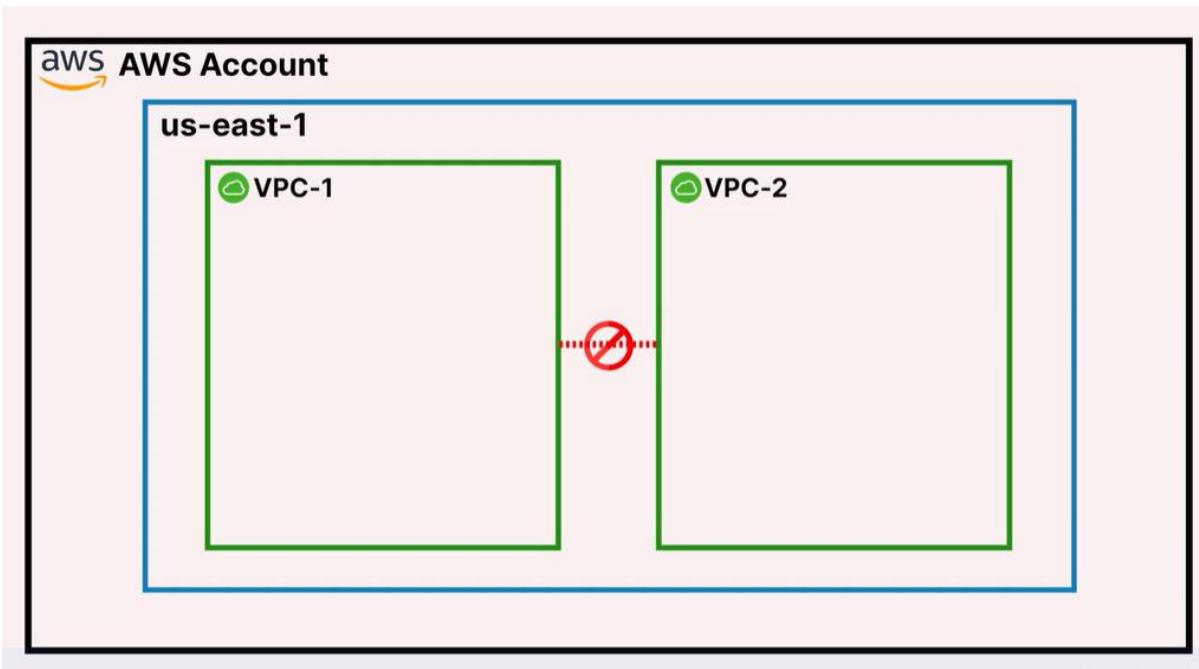


- For example, web servers are placed in a public subnet managing users' requests from the internet and place sensitive database servers and business logics (application servers) in private subnets to prevent unauthorized external access.

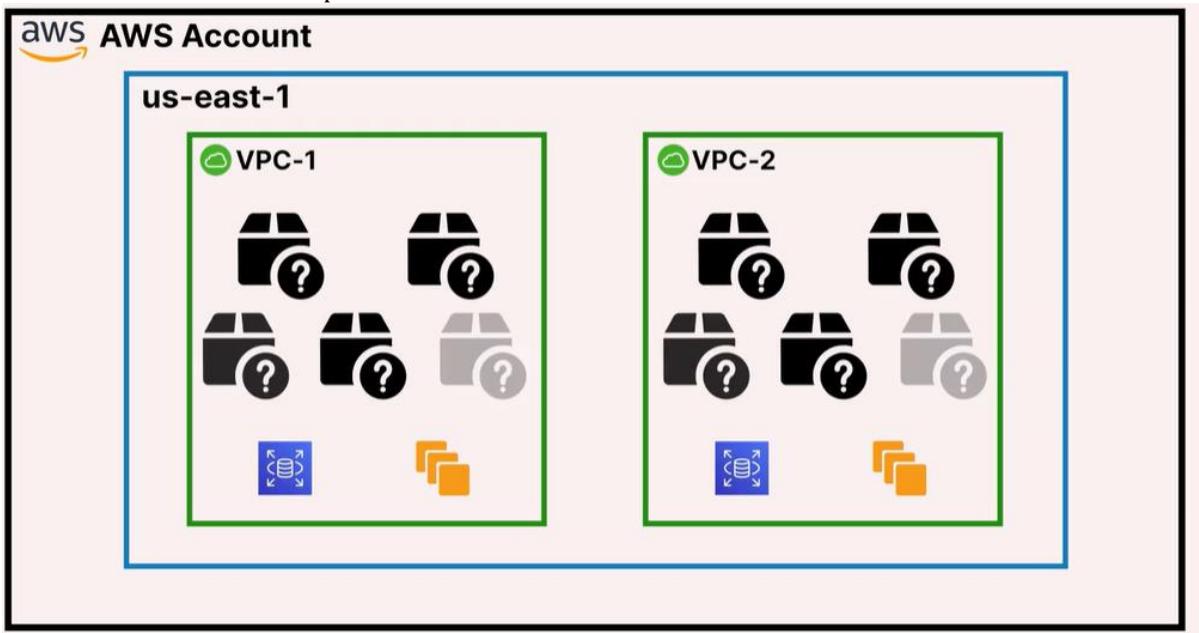
제 3강. Amazon VPC



- VPC is a core service that is provided by AWS and logically isolates private network for users and applications within AWS.
- VPC can create separate spaces within AWS cloud network where you can launch various resources in a virtual network (VPC).
- VPC is isolated from other virtual VPCs where you can fully control AWS services such as Amazon RDS and Amazon EC2 and these services are connected via network provided by VPC.



- There are a number of components within VPC.

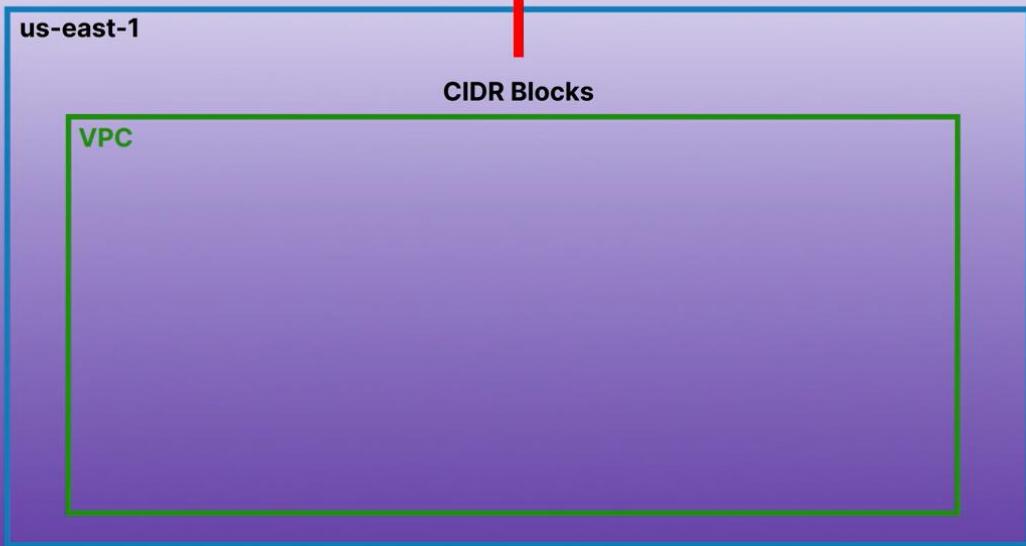


(1) CIDR Blocks

- CIDR Blocks is an IP address range assigned to your VPC. When you create a VPC, we must specify a specific IP address using CIDR notation.
- CIDR Blocks decide a IP address range and the number of resources launched within VPC.

CIDR Notation — 192.168.1.0/22

IP address range assigned to your VPC



- CIDR Notation is a method of signing and managing IP addresses within a network. For instance, if you start with a VPC with a specific IP address, 192.168.1.0 and ends with 192.168.1.255. It has totally 255 IP addresses.

- Every device or resource within a VPC such as servers, databases or containers must have each IP address within this range to ensure proper network communications and identifications.



— 192.168.1.200



— 192.168.1.105

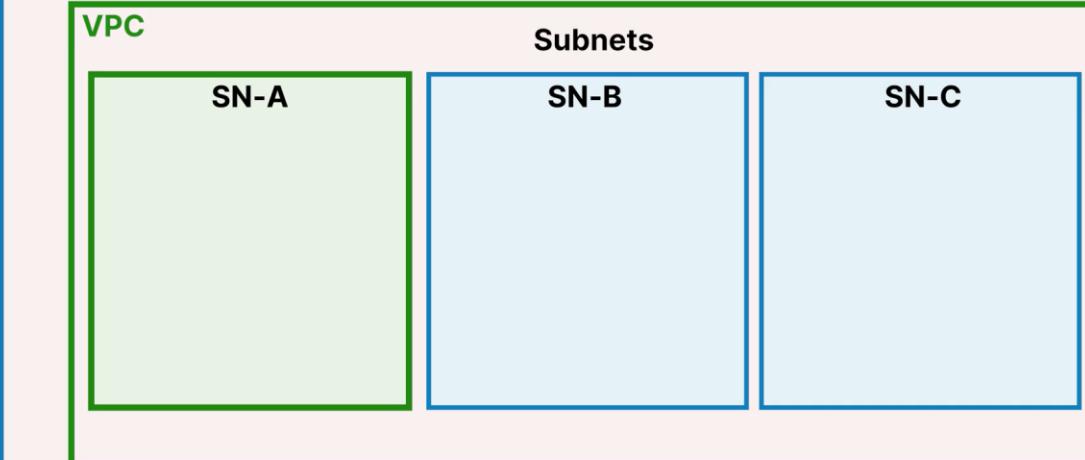


— 192.168.1.158

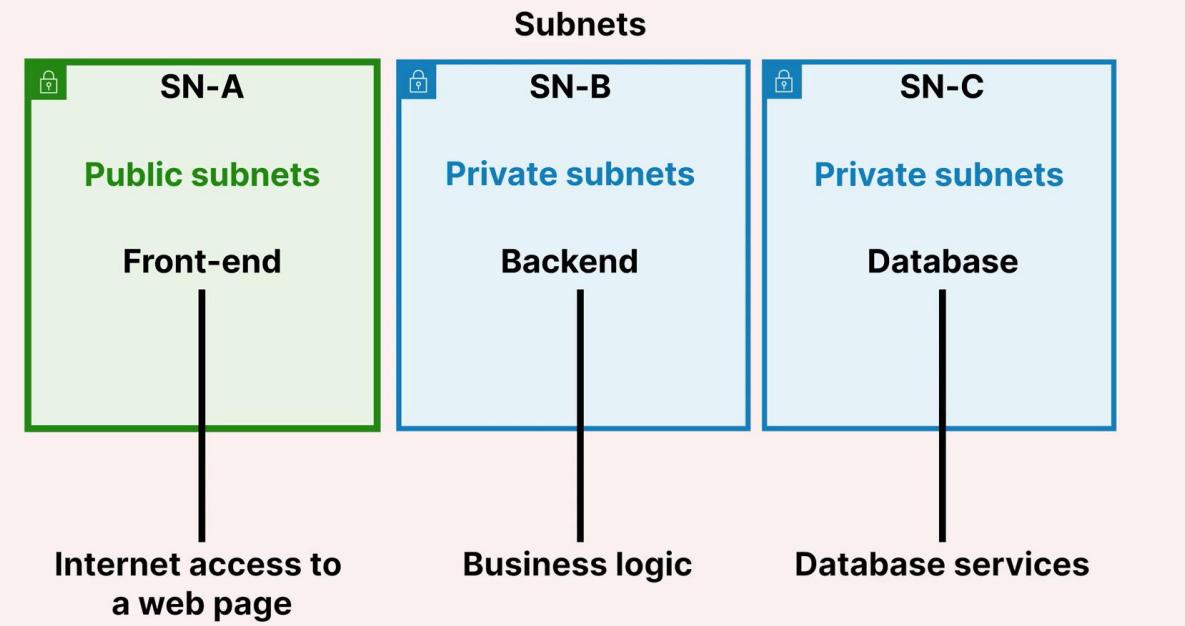
- Using this IP address range can organize a network infrastructure in a scalable and manageable way.
(2) Subnets

- Subnets divide a VPC into smaller sections. Each subnet can serve a specific purpose such as public subnets and private subnets.

us-east-1



- Subnets serve different purposes. For example, public subnets serve front-end for the internet access to a web page. Private subnets serve back-end servers such as business logic and database for database services.

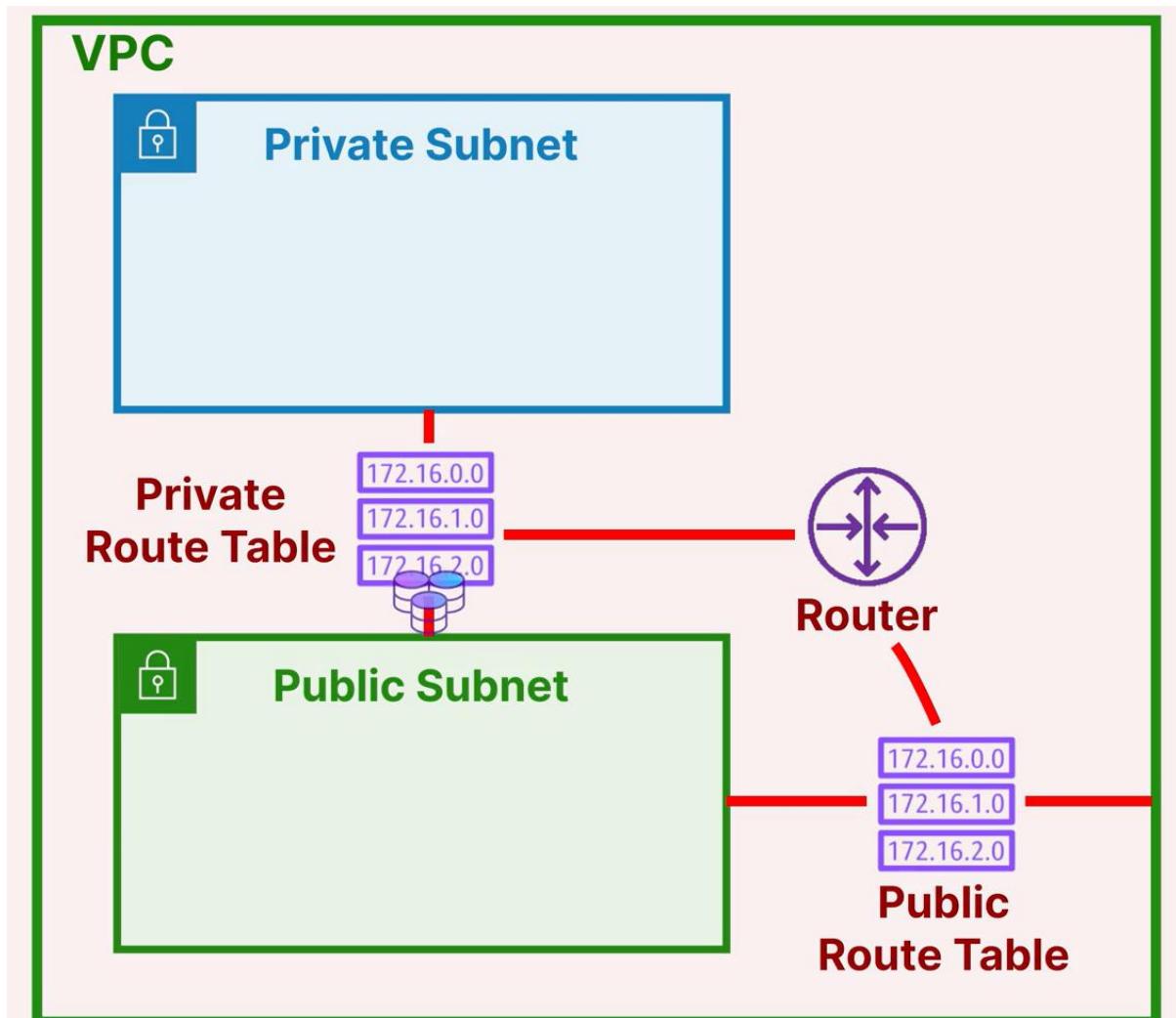


- Subnets isolate public resources from private resources. Most AWS services can be served by private subnets. You can only use public subnets under controlled access and only when it is needed.

(3) Route table

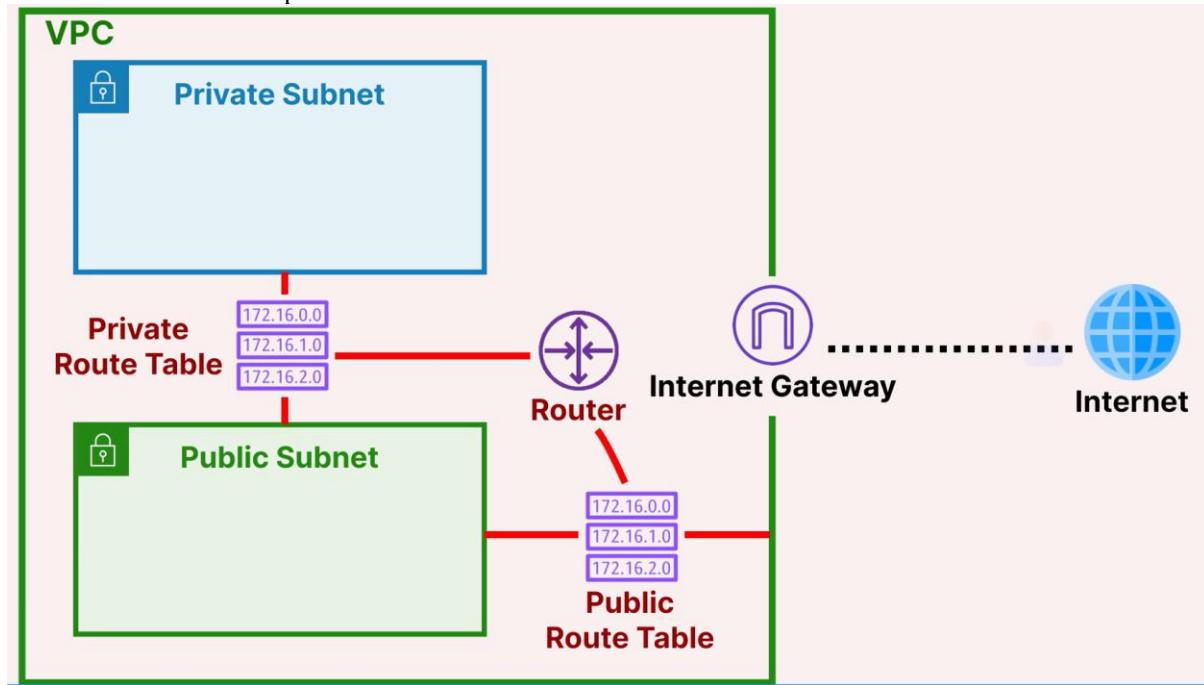
- Think a route table as a street sign guiding traffic on a road. A Route table guides data packet within your VPC and also help VPC access the Internet and other destinations. The route table consists of a set of rules called route determining directions of traffic flow. Each subnet has its own routing tables but you can also make a customized route table and assign it to a subnet.

- Manually customized route table enhances security controlling each subnet for where they need to go and ensure they only go where they are supposed to.



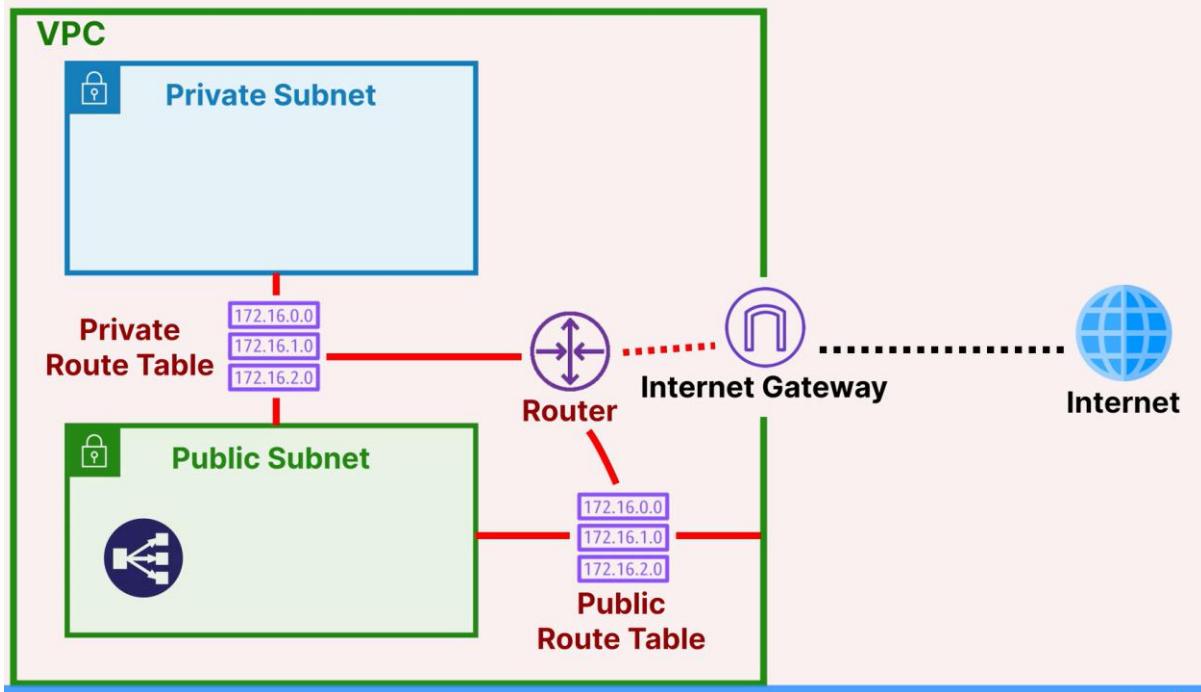
(4) Internet Gateway

- It is a main connection point between VPC and the internet.



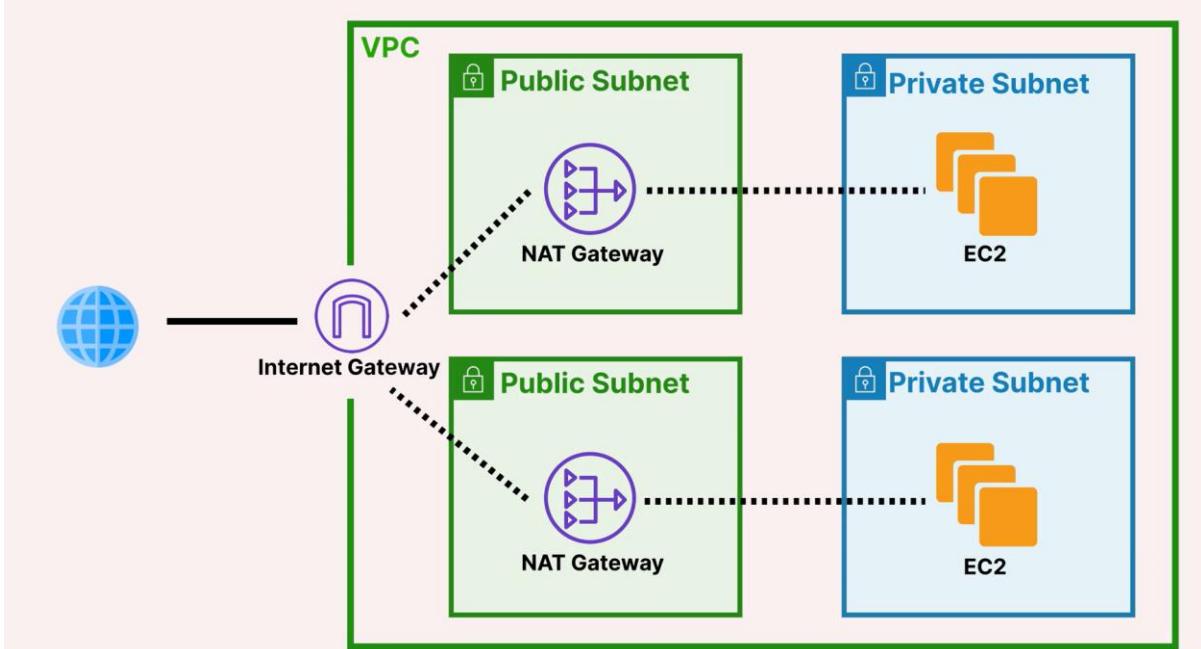
- Internet gateway allows traffic in and out of your VPC. Without the Internet gateway, resources in VPC is

totally isolated from public network. The internet gateway needs to be attached to a public subnet through a public route table defining the roles of the internet gateway to allow the public network to access a public subnet within VPC and without it, it is not allowed to access the public subnet.



(5) NAT (Network Address Translation) gateway

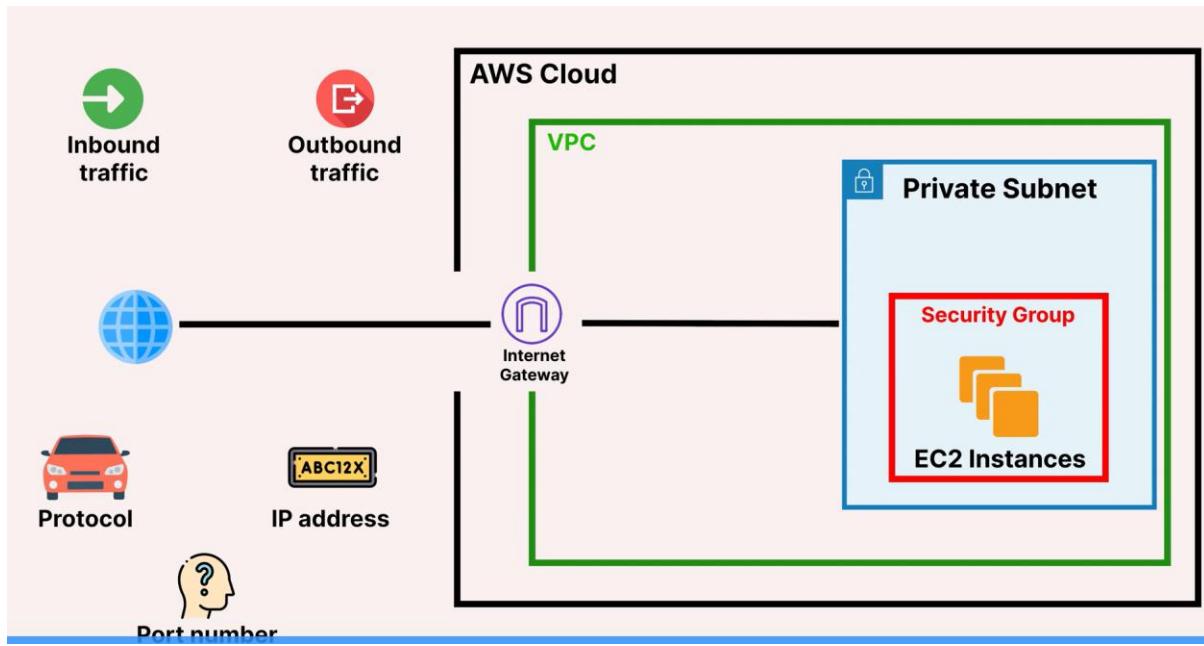
- NAT gateway allows outbound traffic from EC2 instances within private subnets toward the public network but prevent inbound traffic from the internet.



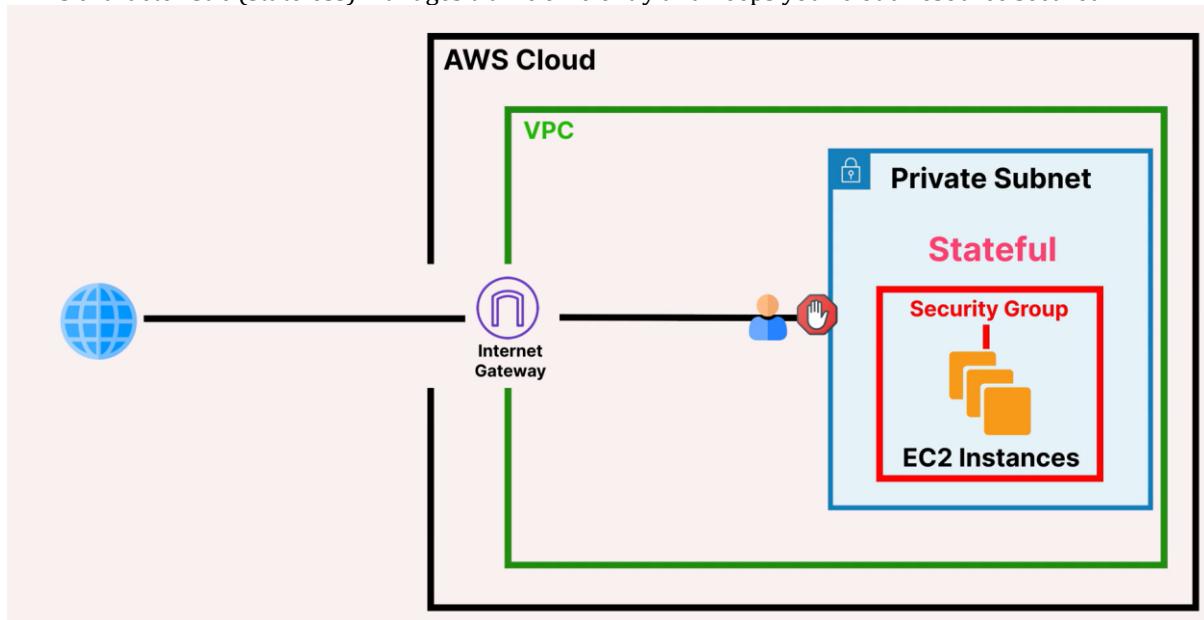
- Servers within private subnets need outgoing traffic to install software and security patch installation. Databases and applications should be installed within private subnets and have a route to NAT gateway routed to the Internet gateway directing toward the public internet.

(6) Security groups

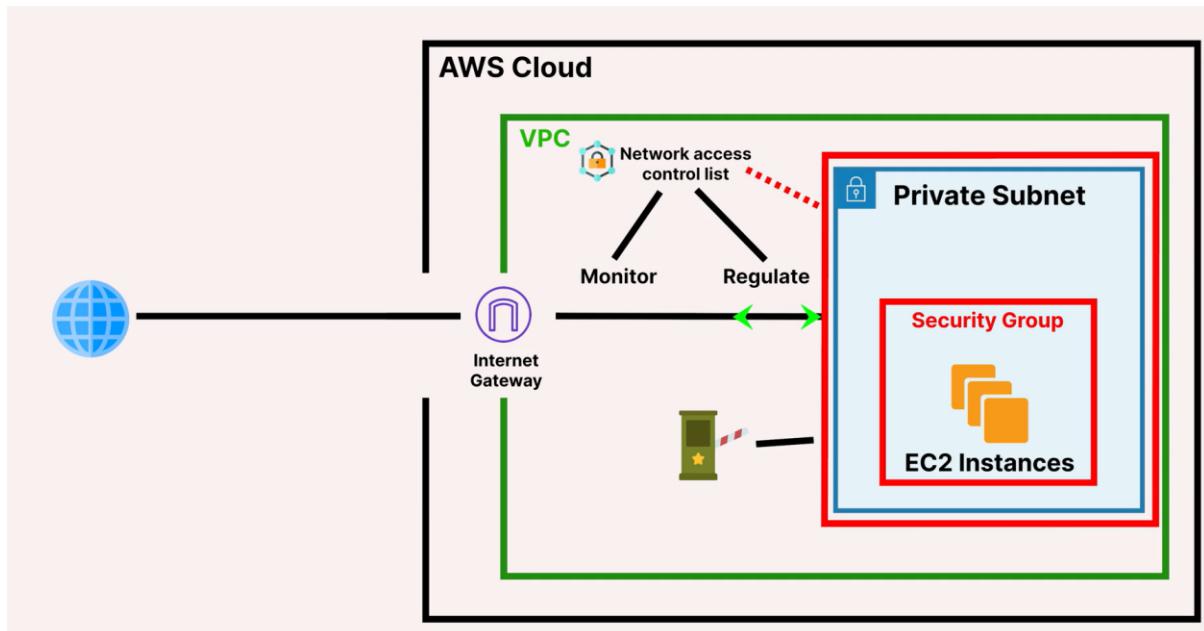
- Security groups consist of rules dictating inbound traffic, who can enter and outbound traffic, who can leave, including information such as protocols, IP address and port number (the reason to visit), which ensure only approved visitors can gain access.



- Security groups are stateless, meaning that once a visitor is allowed in based on their rules. They remember this traffic and when the visitor goes back again, the security groups does not need to check again which means that the visitor is automatically allowed to leave.
- This characteristic (stateless) manages traffic efficiently and keeps your cloud resource secured.



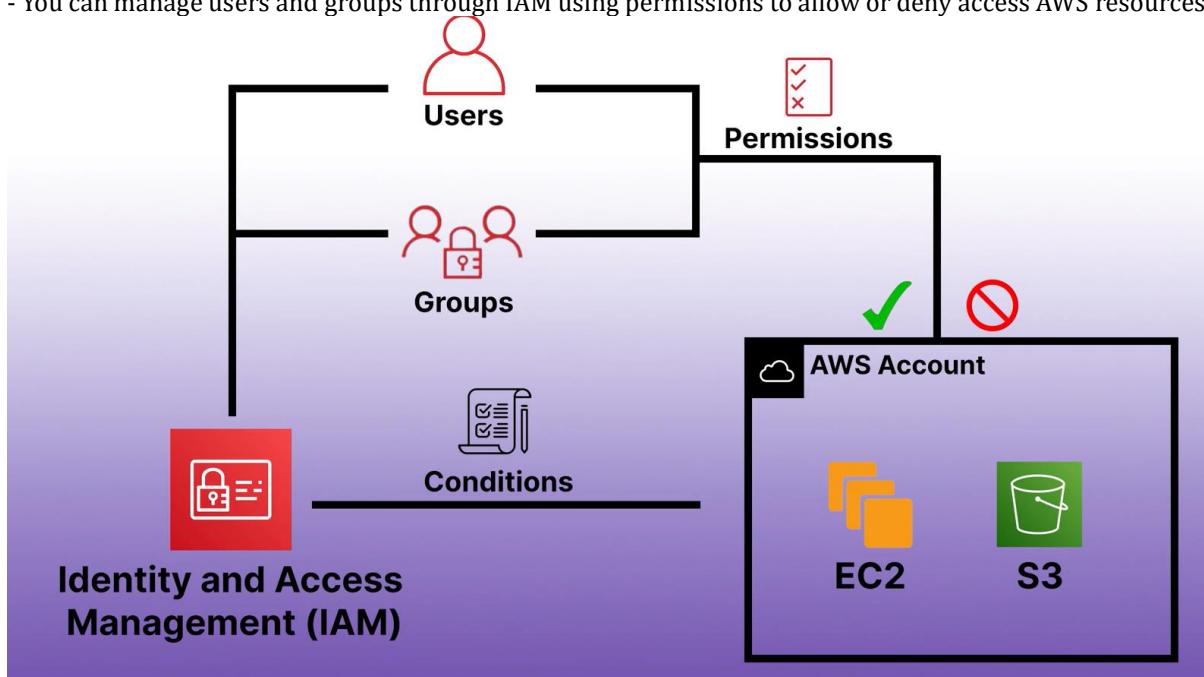
- In AWS, security groups are set up around EC2 instances and prevent unauthorized users based on rules. When configuring security rules, security groups can determine the types of traffic allowed.
- (7) Network Access Control Lists
- NACLs act as a security check point attached to each subnet monitoring and regulating traffic in and out of subnets.



- It controls data packets tried to get through a private VPC based on rules but it is not stateful but stateless, meaning that every datapacket in and out the VPC must be checked whenever they go in and out of the subnet against rules.
- You need separate rules used for making data packets in and out of the private subnet respectively because NACLs do not remember the previous allowances.
- NACLs are ideal for enforcing broader traffic rules at the subnet level such as denying access to certain IP addresses or ranges across the entire subnet.
- Security groups cannot explicitly block certain IP addresses but NACLs fill this gap. Security groups manage access to individual instances.

제 4강. AWS IAM (Identity Access Management)

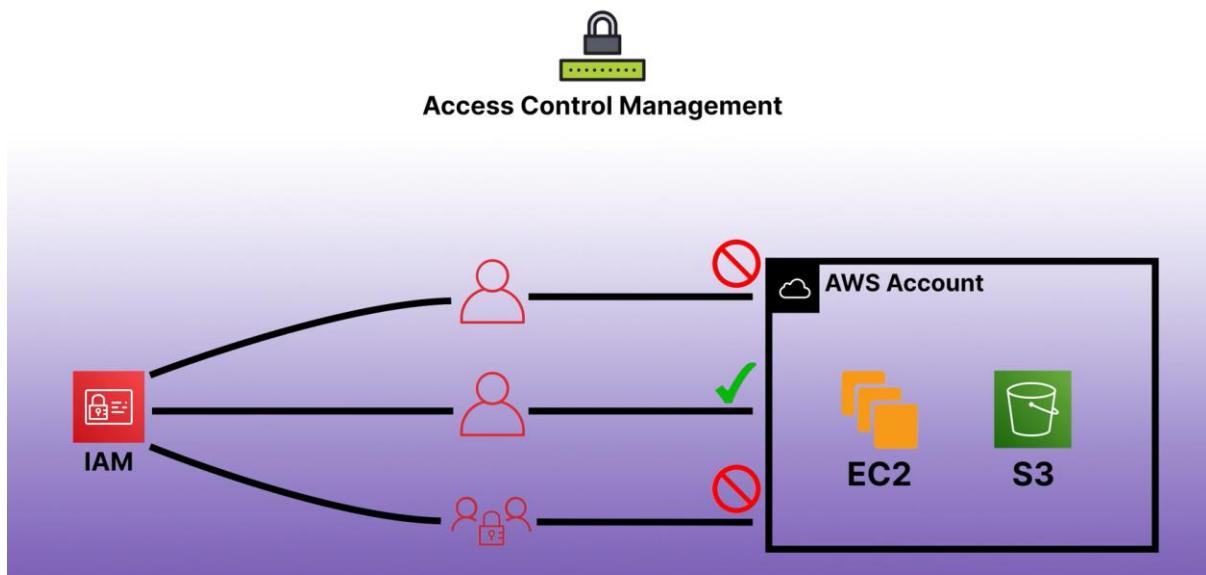
- AWS IAM ensures users to securely access AWS resources such as EC2 and S3 under specific conditions.
- You can manage users and groups through IAM using permissions to allow or deny access AWS resources.



- IAM is essential in managing cloud resources securely for several reasons.

(1) Access Control Management

- IAM can provide permissions to users and groups and specify what they can use and how they can use.



- Due to dynamic and scalable cloud nature, it becomes more complex to manage the access to cloud resources.

(2) Principle of least privilege

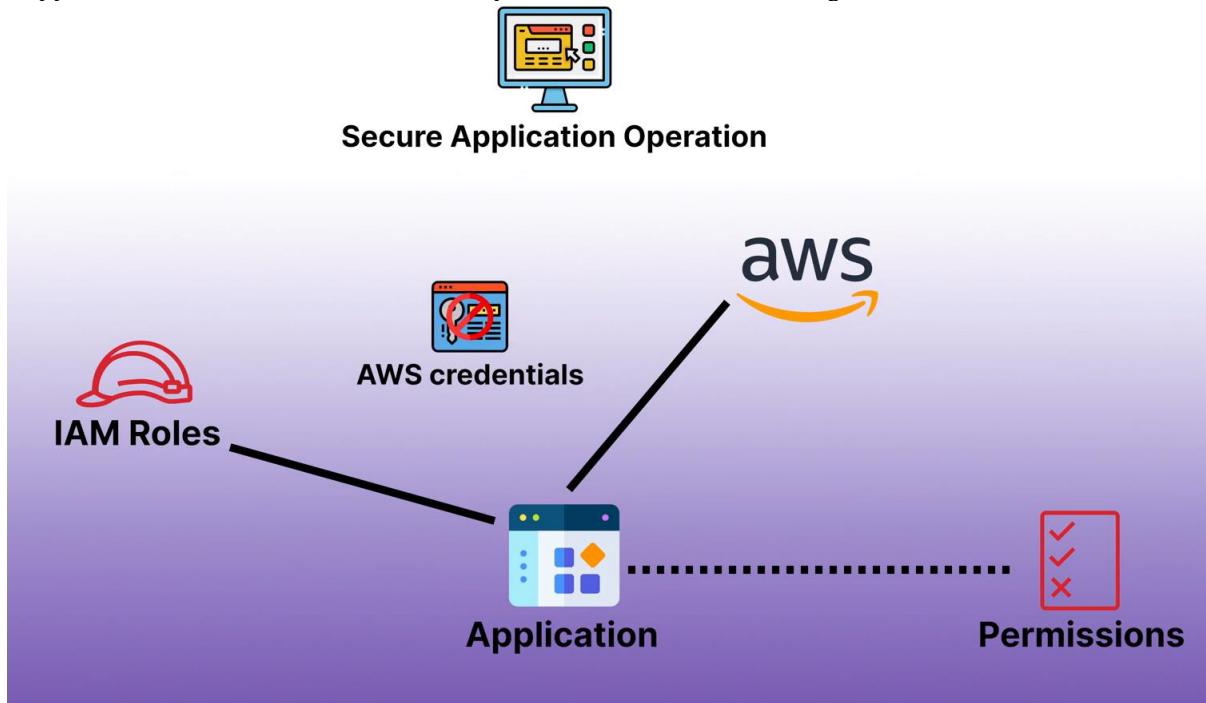
- IAM supports security best practices of granting the least privilege, providing minimum requirements for users and groups to complete tasks.
- This practice minimizes errors and malicious access to cloud resources by limiting access rights of users and groups, providing the bare minimum requirements to complete their tasks.

(3) Multi-Factor Authentication (MFA)

- IAM supports MFA, providing extra layers of security to request users to verify their identity using one or more authentication methods. This will significantly reduce unauthorized access AWS environment.

(4) IAM roles

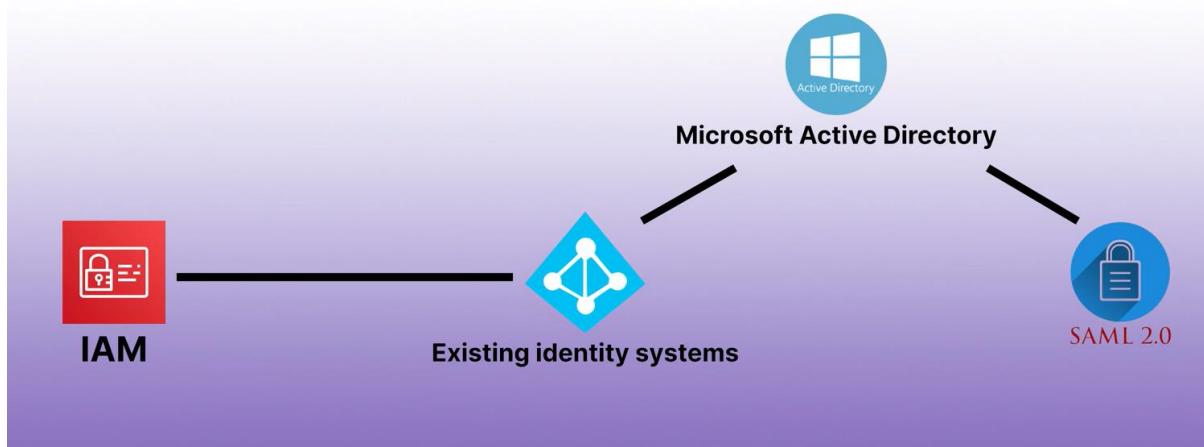
- IAM also ensures applications running on AWS to have only the permissions they need to operate.
- Applications with IAM Roles make API requests to AWS without needing AWS credentials.



(5) Integration and Federation

- IAM can integrate with existing identity systems such as Microsoft Active Directory or any other identity system supporting SAML 2.0

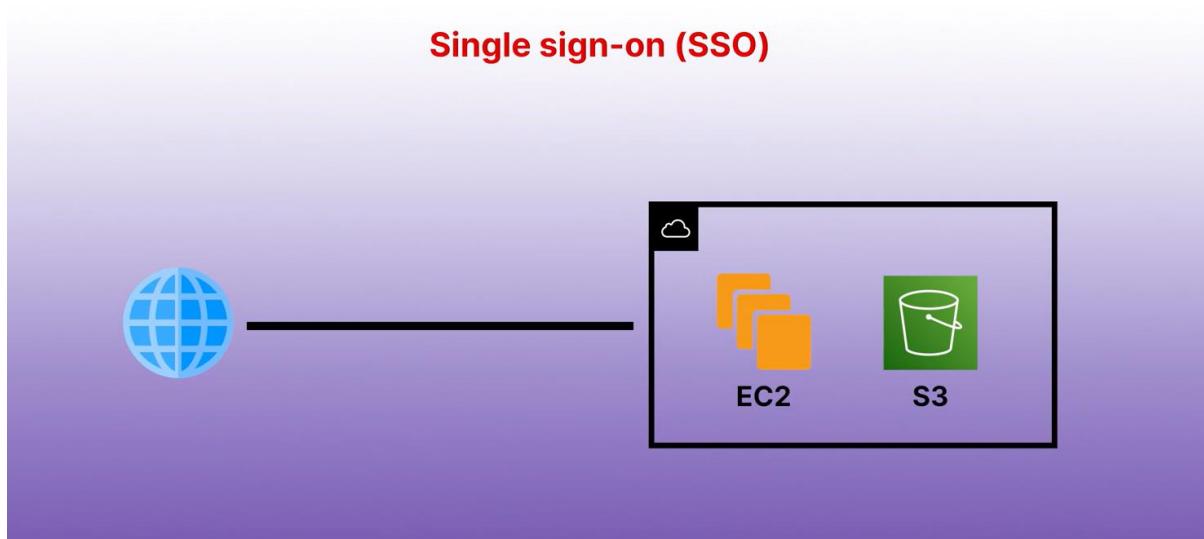
Integration and Federation



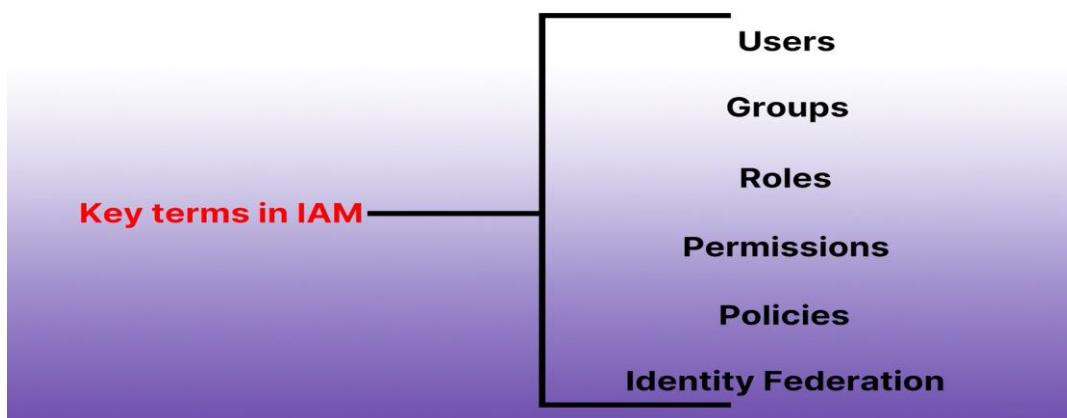
- You can use your own SSO (Single Sign-on) when accessing AWS resources, allowing users to utilize their own credentials that simplify the access management across organizations.

Integration and Federation

Single sign-on (SSO)



Key terms in IAM



(1) Users

- IAM users are individuals or services given permissions to access and interact with AWS resources.

(2) Groups

- IAM groups organize and group users by similar roles and access needs for easier management of permissions.

(3) Roles

- IAM roles is a temporary access badges for limited access for specific AWS resources. IAM roles provide permissions for AWS resources access not using permanent user credentials.

(5) Policies

- IAM policies define permissions or access rules about which AWS resources that users or roles can access and how they can interact with them.

(6) Identity Federation

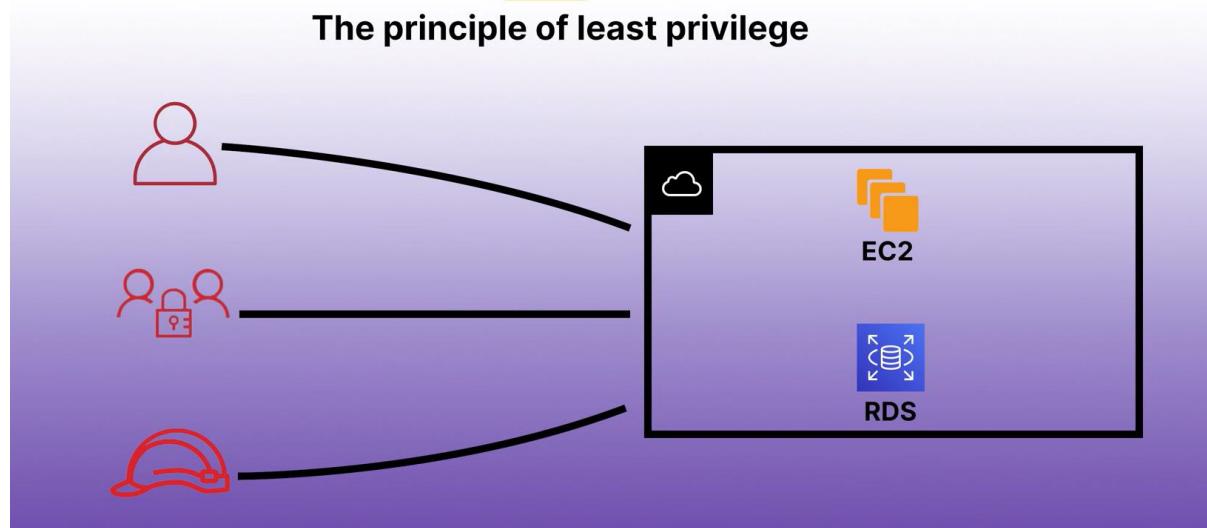
- Identity federation allows users to authenticate using their existing identity from external systems. For examples, corporate directories can access AWS resources without creating additional IAM user account.

(7) The principle of least privilege

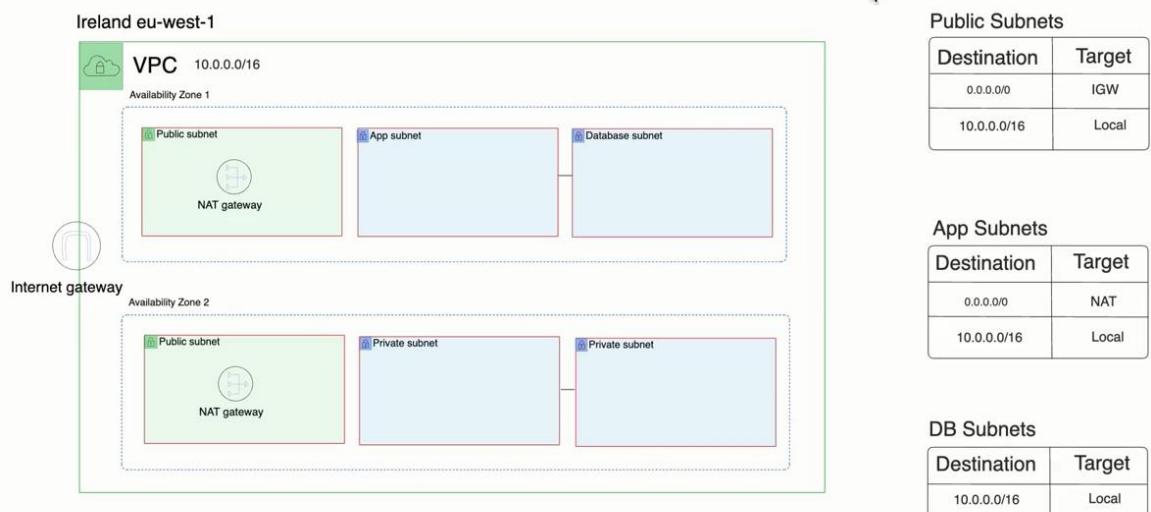
- Users, groups and roles can only have the minimum levels of access needed to perform their tasks, preventing unnecessary access to sensitive information.



The principle of least privilege



제 6강. VPC Diagram



- On the right side, it is route tables including destination and target. In the public subnets, all possible IP addresses (0.0.0.0/0) would be routed to IGW (Internet Gateway).
- It means that all resources within the public subnets will be directed to IGW.
- In the app subnets, all routes or all possible IP addresses are routed to NAT gateway. All resources or possible IP addresses in the app subnets will be directed to NAT gateway, which will be redirected to IGW.
- 10.0.0.0/16 is CIDR blocks, allowing for local routing within our VPC which means that resources within VPC can communicate with each other. For example, if database subnets do not have any CIDR block, other resources cannot be directed to the database subnet. If we add CIDR blocks into DB subnets, they will accept any traffic within VPC.
- There isn't any default route to the NAT gateway in the public subnets inside DB subnets. However, it can access the app subnet using CIDR blocks that can be accessed the NAT gateway through the app subnets.



- NAT gateway allows routes from private subnets to the Internet but prevent unauthorized access from the Internet to the private subnets. It allows only outbound traffic but not inbound traffic, enhancing security in AWS environment because it prevents resources within private subnets from being exposed to external attacks.
- There are 2 different AZs and each NAT gateway is placed within a respective AZ and resources within the app subnets will be routed through the NAT gateway in the same AZ because there will be extra charges for cross-AZ.

제 7강. Create your VPC

- When you click your VPCs, there is a default AWS VPC. There is a CIDR block which is the IP address of VPC.

The screenshot shows the AWS VPC dashboard. In the main pane, a table lists 'Your VPCs (1/1)'. The single entry is 'vpc-0faf378640ac09c0f', which is 'Available' with an IPv4 CIDR of 172.31.0.0/16. Below this, a detailed view for 'vpc-0faf378640ac09c0f' is shown, specifically the 'CIDRs' tab. It displays the IPv4 CIDR 172.31.0.0/16 as associated. The left sidebar includes sections for EC2 Global View, Virtual private cloud (Your VPCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, Peering connections), and Security.

The screenshot shows the VPC resource map for the 'vpc-0faf378640ac09c0f' VPC. The map illustrates the network structure: a central VPC node connected to two Subnet nodes ('us-west-1a' and 'us-west-1b'), which in turn connect to a single Route table node ('rtb-05b85f3d13322338b'). This route table is connected to a Network connection node ('igw-016add0a0ac6b722a'). The left sidebar shows the 'Resource map' tab is selected.

- In the resource map, there is a VPC with 2 different kinds of subnets in different AZs such as us-west-1a and us-west-1b, which are routed to a route table connected to the public internet.

The screenshot shows the 'Create VPC' wizard. On the left, the 'VPC settings' section includes fields for 'Name tag auto-generation' (set to 'Auto-generate' with 'project' as the tag), 'IPv4 CIDR block' (set to 10.0.0.0/16), 'IPv6 CIDR block' (set to 'No IPv6 CIDR block'), and 'Tenancy' (set to 'Default'). On the right, the 'Preview' section shows a hierarchical diagram of the proposed VPC structure: a 'VPC' node ('project-vpc') connected to four 'Subnets' ('us-west-1a' and 'us-west-1b', each containing two subnet types: 'project-subnet-public1-us-west-1a', 'project-subnet-private1-us-west-1a' in 'us-west-1a', and 'project-subnet-public2-us-west-1b', 'project-subnet-private2-us-west-1b' in 'us-west-1b'). These subnets connect to three 'Route tables' ('project-rtb-public', 'project-rtb-private1-us-west-1a', 'project-rtb-private2-us-west-1b') and a single 'Network connection' node ('project-lgw').

- When you create a VPC and more, there is a IPv4 CIDR block with 65,536 IP addresses.
- VPC endpoints connect internal connections to your S3 bucket or dynamoDB.

VPC settings

Preview

VPC Show details
Your AWS virtual network

Subnets (6)
Subnets within this VPC

- us-west-1a**
 - project-subnet-public1-us-west-1a
 - project-subnet-private1-us-west-1a
 - project-subnet-private3-us-west-1a
 - project-subnet-private4-us-west-1b
- us-west-1b**
 - project-subnet-public2-us-west-1b
 - project-subnet-private2-us-west-1b

Route tables (5)
Route network traffic to resources

- project-rtb-public
- project-rtb-private1-us-west-1a
- project-rtb-private2-us-west-1b
- project-rtb-private3-us-west-1a
- project-rtb-private4-us-west-1b

Network connections (4)
Connections to other networks

- project-igw
- project-1-igw-public2-us-west-1b
- project-vpc-s3

Name tag auto-generation **Info**
Enter a name for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.

Auto-generate
project

IPv4 CIDR block **Info**
Determine the starting IP and the size of your VPC using CIDR notation.
10.0.0.0/16 65,536 IPs
CIDR block size must be between /16 and /28

IPv6 CIDR block **Info**
 No IPv6 CIDR block
 Amazon-provided IPv6 CIDR block

Tenancy **Info**
Default

Number of Availability Zones (AZs) **Info**
Choose the number of AZs in which to provision subnets. We recommend at least two AZs for failover availability.
1 **2** **3** **4** **Customize AZs**

CloudWatch Feedback

Number of public subnets **Info**
The number of public subnets to add to your VPC. Use public subnets for web applications that need to be publicly accessible over the Internet.
0 **1** **2** **3** **4**

Number of private subnets **Info**
The number of private subnets to add to your VPC. Use private subnets to secure backend resources that don't need public access.
0 **1** **2** **3** **4**

Customize subnets CIDR blocks

Public subnet CIDR block in us-west-1a	4,096 IPs
10.0.0.0/20	4,096 IPs
Public subnet CIDR block in us-west-1b	4,096 IPs
10.0.16.0/20	4,096 IPs
Private subnet CIDR block in us-west-1a	4,096 IPs
10.0.128.0/20	4,096 IPs
Private subnet CIDR block in us-west-1b	4,096 IPs
10.0.144.0/20	4,096 IPs
Private subnet CIDR block in us-west-1a	4,096 IPs
10.0.160.0/20	4,096 IPs
Private subnet CIDR block in us-west-1b	4,096 IPs
10.0.176.0/20	4,096 IPs

NAT gateways (\$) **Info**
Choose the number of Availability Zones (AZs) in which to create NAT gateways. Note that there is a charge for each NAT gateway.
1 per AZ

NAT gateways (1) **Info**
Choose the number of Availability Zones (AZs) in which to create NAT gateways. Note that there is a charge for each NAT gateway.
None **In 1 AZ** **1 per AZ**

VPC endpoints **Info**
Endpoints can help reduce NAT gateway charges and improve security by accessing S3 directly from the VPC. By default, full access policy is used. You can customize this policy at any time.
None **S3 Gateway**

DNS options **Info**
 Enable DNS hostnames
 Enable DNS resolution

Additional tags

- There are 2 AZs closest to users by default such as us-west-1 and us-west-2. There are 2 different public subnets separated in each AZ and 4 different private subnets, among which 2 different private subnets are placed within each AZ.
- We can place each NAT gateway for each public subnet to make resources within private subnets connected to the public internet.
- 2 different public subnets are routed to the Internet gateway for the public internet connectivity.

EC2 Global View

Filter by IP/CIDR

VPC : vpc-02ae3f6fa855f115c

Clear filters

Subnets (6) info

ID	Available IPv4 addresses	Availability Zone	Availability Zone ID	Route table	Network ACL	Default subnet	Auto-assign public IPv4 add...	Auto-assign private IPv6 add...
4091		us-west-1b	usw1-az1	rtb-07ecab4b6ea0c48fe proj...	ad-007723936bddeb58ca	No	No	No
4091		us-west-1b	usw1-az1	rtb-0b142110a1435da1 proj...	ad-007723936bddeb58ca	No	No	No
4091		us-west-1b	usw1-az1	rtb-045442d407e5de2f proj...	ad-007723936bddeb58ca	No	No	No
4091		us-west-1a	usw1-az3	rtb-042063a614e2a6f2f proj...	ad-007723936bddeb58ca	No	No	No
4091		us-west-1a	usw1-az3	rtb-04442d407e5de2f proj...	ad-007723936bddeb58ca	No	No	No
4091		us-west-1a	usw1-az3	rtb-06e365c4c26378a48 proj...	ad-007723936bddeb58ca	No	No	No

Select a subnet

CloudShell Feedback

- Each subnet has its own route table. You can go to the subnets and filter the subnets.

EC2 Global View

Filter by VPC

VPC : vpc-02ae3f6fa855f115c

Clear filters

Subnets (1/6) info

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR	IPv6 CIDR association ID	Available IPv6 CIDR
project-subnet-private2-us-west-1b	subnet-0501a560b56c43bf	Available	vpc-02ae3f6fa855f115c proj...	10.0.144.0/20	-	-	4091
project-subnet-private1-us-west-1b	subnet-0876992b1d2727403	Available	vpc-02ae3f6fa855f115c proj...	10.0.176.0/20	-	-	4091
project-subnet-public2-us-west-1b	subnet-095c5e50ce1925a0f	Available	vpc-02ae3f6fa855f115c proj...	10.0.16.0/20	-	-	4091
project-subnet-private1-us-west-1a	subnet-00889554802ee3a8	Available	vpc-02ae3f6fa855f115c proj...	10.0.128.0/20	-	-	4091
project-subnet-public1-us-west-1a	subnet-0bf8d174d890d8e76	Available	vpc-02ae3f6fa855f115c proj...	10.0.0.0/20	-	-	4091
project-subnet-private3-us-west-1a	subnet-0ef5ceb830752y07	Available	vpc-02ae3f6fa855f115c proj...	10.0.160.0/20	-	-	4091

subnet-0bf8d174d890d8e76 / project-subnet-public1-us-west-1a

Details | Flow logs | **Route table** | Network ACL | CIDR reservations | Sharing | Tags

Route table: rtb-045442d407e5de2f / project-rtb-public

Routes (2)

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	gw-071a7b541994eb054

Edit route table association

CloudShell Feedback

- In a public subnet, you can see a IP address for local communication within VPC and all possible resources within the public subnet can be routed to the internet gateway for external connectivity.

The screenshot shows the AWS VPC Subnets page. On the left, there's a sidebar with navigation links like EC2 Global View, Filter by VPC, Virtual private cloud, Subnets, Route tables, Security, DNS firewall, and Network Firewall. The main area shows a table of subnets with columns for Name, Subnet ID, State, VPC, IPv4 CIDR, IPv6 CIDR, IPv6 CIDR association ID, and Available IPv6. One subnet is selected: project-subnet-private4-us-west-1b. Below the table, a detailed view for this subnet shows its route table association: rtb-0b142110a1435da1d / project-rtb-private4-us-west-1b. The route table has two routes: one to 0.0.0.0/16 (target: local) and another to 0.0.0.0/0 (target: nat-0b16890b34407585).

- In a private subnet, all possible resources can be routed to the NAT gateway in each public subnet in the same AZ for the external connectivity. And also, it is possible for internal communication with resources within VPC.

The screenshot shows the AWS Route Tables page. The sidebar includes EC2 Global View, Filter by VPC, Virtual private cloud, Subnets, Route tables, Internet gateways, Egress-only internet gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, Peering connections, Security, DNS firewall, Rule groups, and Network Firewall. The main area displays a table of route tables with columns for Name, Route table ID, Explicit subnet association, Edge associations, Main, VPC, Owner ID, and Last updated. One route table is selected: rtb-042063a614e2a6f2f / project-rtb-private1-us-west-1a. The details pane shows its routes: one to 0.0.0.0/0 (target: nat-0290af44-1-pu440a, status: Blackhole, propagated: No) and another to 0.0.0.0/16 (target: local, status: Active, propagated: No).

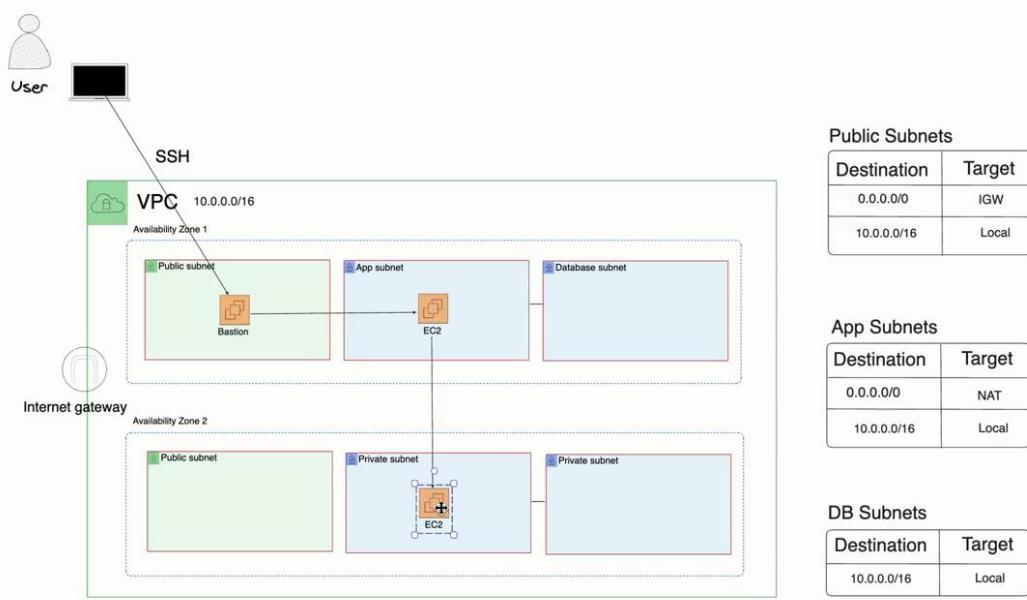
- In route tables and a private subnet, if there is any NAT gateway route table, we can edit routes and add a route table for routing to a NAT gateway in a public subnet.

제 11강. Updating Architecture

- Due to the fact that NAT gateway costs a lot of money, we replace it with Bastion host in a public subnet. This bastion host is connected to EC2 instances running in both app subnet and private subnet.
- We want to ping EC2 instances in the private subnet from the app subnet in AZ 1 while configuring security groups for EC2 instances.

제 12강. Setting up Bastion host

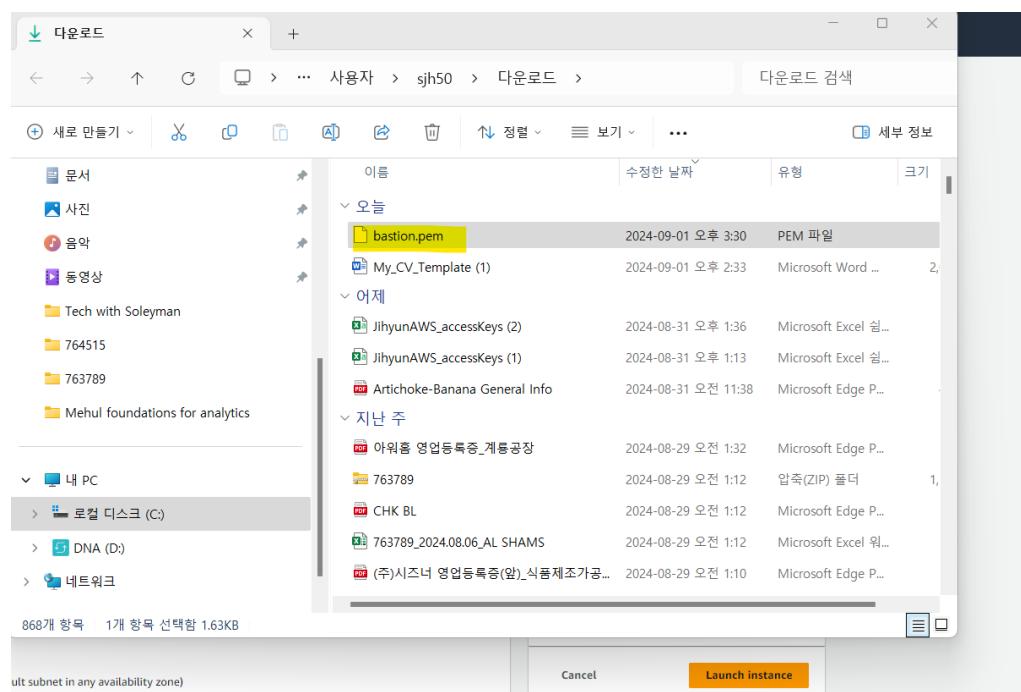
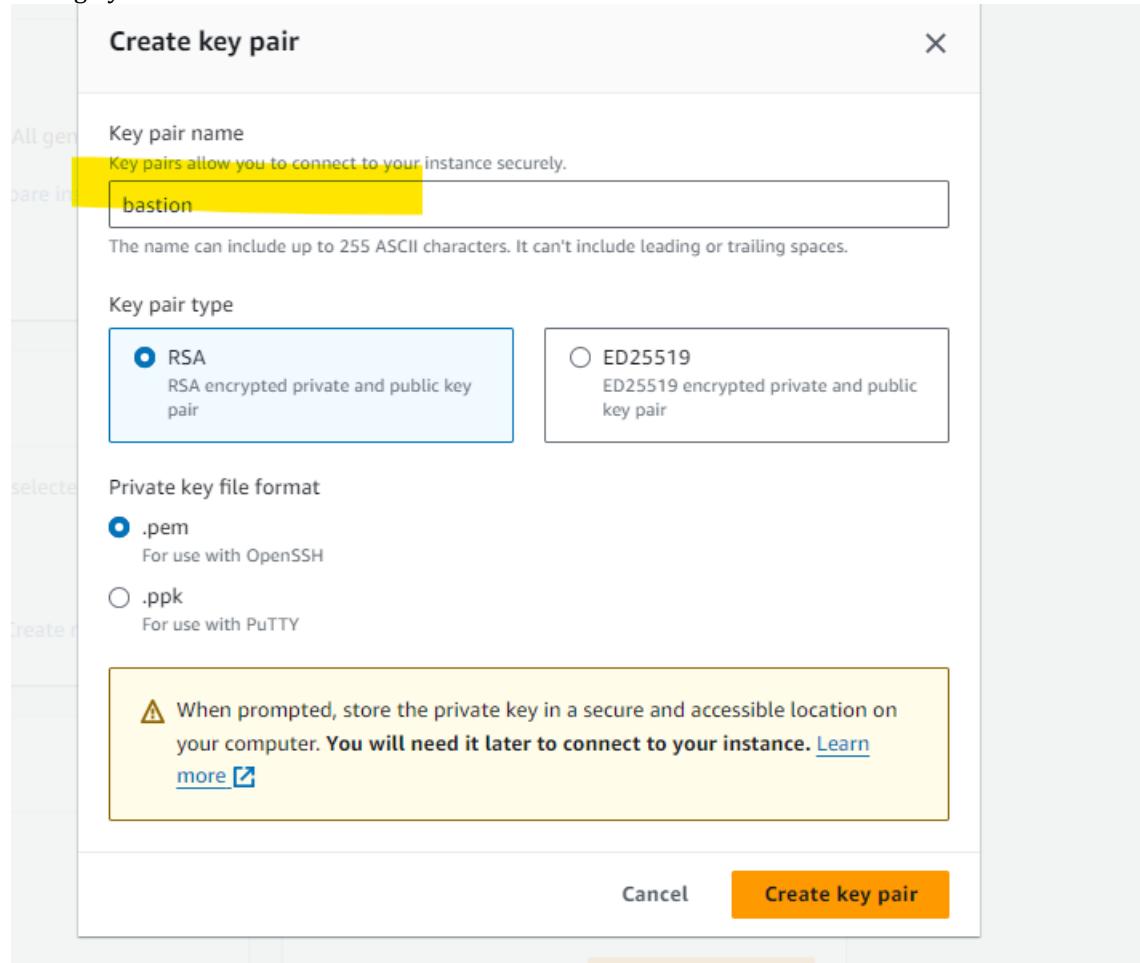
- A bastion host is attached using SSH by our local machines that can connect to EC instances in a app subnet and ping EC2 instances in the 2nd AZ from EC2 instance from the 1st AZ.



(1) Creating a bastion server.

- We can go to the EC2 instance dashboard and launch a new instance and name it as BastionHost.

- Scroll down to key pairs and create a key pair that will be used for security connection to the BastionHost through your local machine.



- And then moving down to the network settings and clicking edit.

Network settings

Network | [Info](#)
vpc-0faf378640ac09c0f

Subnet | [Info](#)
No preference (Default subnet in any availability zone)

Auto-assign public IP | [Info](#)
Enable
Additional charges apply when outside of free tier allowance

Firewall (security groups) | [Info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group Select existing security group

We'll create a new security group called 'launch-wizard-2' with the following rules:

Allow SSH traffic from Anywhere
Helps you connect to your instance

Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server

Allow HTTP traffic from the internet

[Cancel](#) [Launch instance](#) [Review command](#)

Network settings

VPC - required | [Info](#)
vpc-02ae3f6fa855f115c (project-vpc)
10.0.0.0/16

Subnet | [Info](#)
subnet-0bf8d174d890d8e76 project-subnet-public1-us-west-1a
VPC: vpc-02ae3f6fa855f115c Owner: 825765379748 Availability Zone: us-west-1a
Zone type: Availability Zone IP addresses available: 4091 CIDR: 10.0.0.0/20

[Create new subnet](#)

Auto-assign public IP | [Info](#)
Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) | [Info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group Select existing security group

Security group name - required
bastionSG

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and _:/()#-.@!\$*

[dShell](#) [Feedback](#)

- In the network setting, you can select customized VPC and subnet must be a public subnet where this bastion EC2 instance will be placed. Auto-assign public IP is enabled and create a security group for this bastion EC2 instance.

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 209.129.88.187/32, MyLocalIPOnly)

Type | Info Protocol | Info Port range | Info

ssh TCP 22

Source type | Info Name | Info Description - optional | Info

My IP Add CIDR, prefix list or security MyLocalIPOnly

209.129.88.187/32 X

Add security group rule Advanced network configuration

- In inbound security group rules, Source type should be My IP only because we want to make only our local machine to access the Bastion Host.

제 13강. SSH into Bastion host

Instances (1/2) Info

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
linux_ec2	i-089299f6f1ac6bd3a	Running	t2.micro	2/2 checks passed	View alarms +	us-west-1a	ec2-18-
BastionHost	i-0ce56617d0f18d1ba	Running	t2.micro	2/2 checks passed	View alarms +	us-west-1a	ec2-54-

i-0ce56617d0f18d1ba (BastionHost)

Details | Status and alarms | Monitoring | Security | Networking | Storage | Tags

Instance summary | Public IPv4 address copied

Instance ID: i-0ce56617d0f18d1ba (BastionHost) | 54.67.113.123 | open address

IPv6 address: - | Instance state: Running

Hostname type: IP name: ip-10-0-1-30.us-west-1.compute.internal | Private IP DNS name (IPv4 only): ip-10-0-1-30.us-west-1.compute.internal

Answer private resource DNS name | Instance type: t2.micro

Private IPv4 addresses: 10.0.1.30 | Public IPv4 DNS: ec2-54-67-113-123.us-west-1.compute.amazonaws.com | Elastic IP addresses:

- Copy the public IPv4 for Bastion Host.

1. Running SSH commands to connect to the Bastion host and also make sure where you download the key pair for this instance.

```
% -i bastion.pem ec2-user@54.67.113.123
```

- write down the public IPv4 for the instance after @ but the CLI is asking to lockdown the bastion.pem file because it is a secret key that allows for the secure access to Bastion EC2 instance.



A screenshot of a terminal window titled "Downloads — zsh — 80x24". The window shows the following text:

```
[soleyman@Soleymans-MBP Downloads % ssh -i bastion.pem ec2-user@3.8.236.121
@@@@@@@@@@@WARNING: UNPROTECTED PRIVATE KEY FILE! @
Permissions 0644 for 'bastion.pem' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
Load key "bastion.pem": bad permissions
ec2-user@3.8.236.121: Permission denied (publickey,gssapi-keyex,gssapi-with-mic)
.
soleyman@Soleymans-MBP Downloads %
```

2. We need to change the permissions but please make sure that the bastion key file should be located in the same directory you designated.

```
% chmod 400 bastion.pem
```



A screenshot of a terminal window titled "Downloads — zsh — 80x24". The window shows the following text:

```
[soleyman@Soleymans-MBP Downloads % ssh -i bastion.pem ec2-user@3.8.236.121
@@@@@@@@@@@WARNING: UNPROTECTED PRIVATE KEY FILE! @
Permissions 0644 for 'bastion.pem' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
Load key "bastion.pem": bad permissions
ec2-user@3.8.236.121: Permission denied (publickey,gssapi-keyex,gssapi-with-mic)
.
soleyman@Soleymans-MBP Downloads % chmod 400 bastion.pem
```

3. After changing the permissions, we rerun the same command again.

```
% -i bastion.pem ec2-user@3.8.236.121
```

```

[soleyman@Soleymans-MBP Downloads % ssh -i bastion.pem ec2-user@3.8.236.121
@@@@@@@@@@@WARNING: UNPROTECTED PRIVATE KEY FILE!
Permissions 0644 for 'bastion.pem' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
Load key "bastion.pem": bad permissions
ec2-user@3.8.236.121: Permission denied (publickey,gssapi-keyex,gssapi-with-mic)

[soleyman@Soleymans-MBP Downloads % chmod 400 bastion.pem
[soleyman@Soleymans-MBP Downloads % ssh -i bastion.pem ec2-user@3.8.236.121
#_
###_ Amazon Linux 2023
###_#####
###_ \###|
###_ \#/ __ https://aws.amazon.com/linux/amazon-linux-2023
###_ \~' '-->
###_ /
###_.--/_/
/m/'

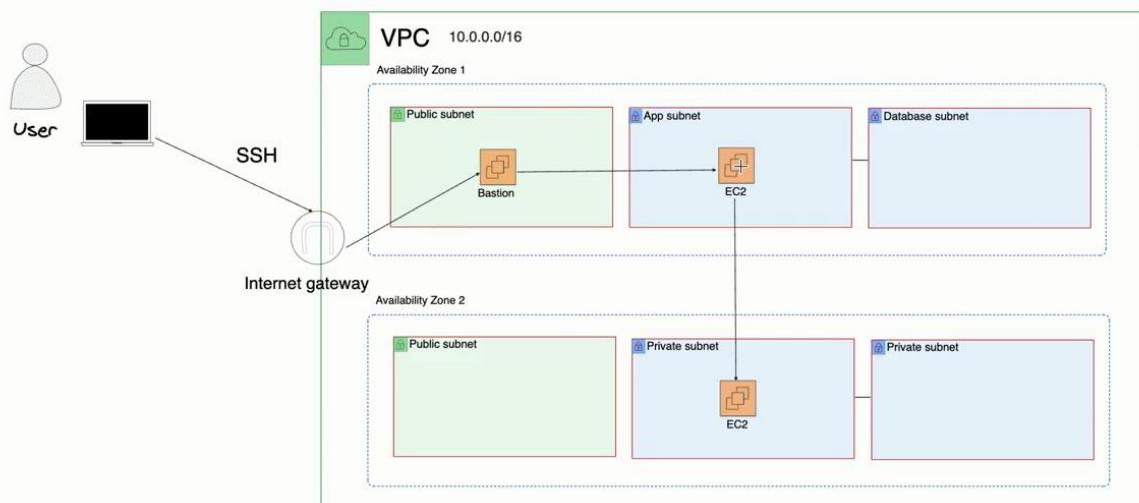
[ec2-user@ip-10-0-8-95 ~]$ 

```

- And now we can successfully access the bastion host and 10-0-8-95 is the private IP address for this instance and successfully get into the Bastion host via our local machine.

제 14강. Routing in Action

- We need the public internet connectivity if we want to SSH into EC2 instances in private subnets. We can use NAT gateway but we need to pay a lot of money. Instead, we are using the Bastion Host to help private EC2 instances connect to the public internet via SSH from your local machine.



- However, the actual route is that SSH is connected to the Internet gateway to connect to the Bastion Host. If we remove the Internet gateway, we no longer access the Bastion Host via SSH with local IP. The bastion Host has its route to the Internet gateway.

What about removing the route for the Internet gateway?

```

Last login: Sat Mar 9 17:59:24 on ttys000
[soleyman@Soleymans-MBP ~ % cd Downloads
soleyman@Soleymans-MBP Downloads % ssh -i bastion.pem ec2-user@3.8.236.121
#
#_
~\_ #####_      Amazon Linux 2023
~~ \#####\
~~  \###|
~~   \#/ ___ https://aws.amazon.com/linux/amazon-linux-2023
~~   V~' '-->
~~   /
~~ .-' /-
~- /_/
/_m'_
Last login: Sat Mar 9 14:00:19 2024 from 2.51.107.15
[ec2-user@ip-10-0-8-95 ~]$

```

1. Connected to the bastion host via SSH once again.

Name	Subnet ID	State	VPC	IPv4 CIDR
project-subnet-private2-us-west-1b	subnet-0501a560bd56c45bf	Available	vpc-02ae3f6fa855f115c proj...	10.0.144.0/20
project-subnet-private4-us-west-1b	subnet-0876992b21d273403	Available	vpc-02ae3f6fa855f115c proj...	10.0.176.0/20
project-subnet-public2-us-west-1b	subnet-0b5c5e59ce1925a0f	Available	vpc-02ae3f6fa855f115c proj...	10.0.16.0/20
project-subnet-private1-us-west-1a	subnet-00d689554802ee3a8	Available	vpc-02ae3f6fa855f115c proj...	10.0.128.0/20
project-subnet-public1-us-west-1a	subnet-0bf8d174d890d8e76	Available	vpc-02ae3f6fa855f115c proj...	10.0.0.0/20
-	subnet-04a08564f470ebcb0	Available	vpc-0faf378640ac09c0f	172.31.0.0/20
-	subnet-0017cd180f404e7e	Available	vpc-0faf378640ac09c0f	172.31.16.0/20
project-subnet-private3-us-west-1a	subnet-0ef38ceb830752c02	Available	vpc-02ae3f6fa855f115c proj...	10.0.160.0/20

subnet-0bf8d174d890d8e76 / project-subnet-public1-us-west-1a

Details Flow logs Route table Network ACL CIDR reservations Sharing Tags

Details

- Moving onto the subnets in VPC and click the public subnet where the Bastion host is placed.

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR	IPv6 CIDR association ID	Avail
project-subnet-private2-us-west-1b	subnet-0501a560bd56c45bf	Available	vpc-02ae3f6fa855f115c proj...	10.0.144.0/20	-	-	409
project-subnet-private4-us-west-1b	subnet-0876992b21d273403	Available	vpc-02ae3f6fa855f115c proj...	10.0.176.0/20	-	-	409
project-subnet-public2-us-west-1b	subnet-0b5c5e59ce1925a0f	Available	vpc-02ae3f6fa855f115c proj...	10.0.16.0/20	-	-	409
project-subnet-private1-us-west-1a	subnet-00d689554802ee3a8	Available	vpc-02ae3f6fa855f115c proj...	10.0.128.0/20	-	-	409
project-subnet-public1-us-west-1a	subnet-0bf8d174d890d8e76	Available	vpc-02ae3f6fa855f115c proj...	10.0.0.0/20	-	-	409
-	subnet-04a08564f470ebcb0	Available	vpc-0faf378640ac09c0f	172.31.0.0/20	-	-	409
-	subnet-0017cd180f404e7e	Available	vpc-0faf378640ac09c0f	172.31.16.0/20	-	-	409
project-subnet-private3-us-west-1a	subnet-0ef38ceb830752c02	Available	vpc-02ae3f6fa855f115c proj...	10.0.160.0/20	-	-	409

Route table: rtb-045442d407e3d0e2f / project-rtb-public

Routes (2)

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	igw-071679543994ebd54

- All possible routes in the public subnet can be accessed the internet gateway and click the actual route

table.

The screenshot shows two consecutive pages from the AWS VPC console:

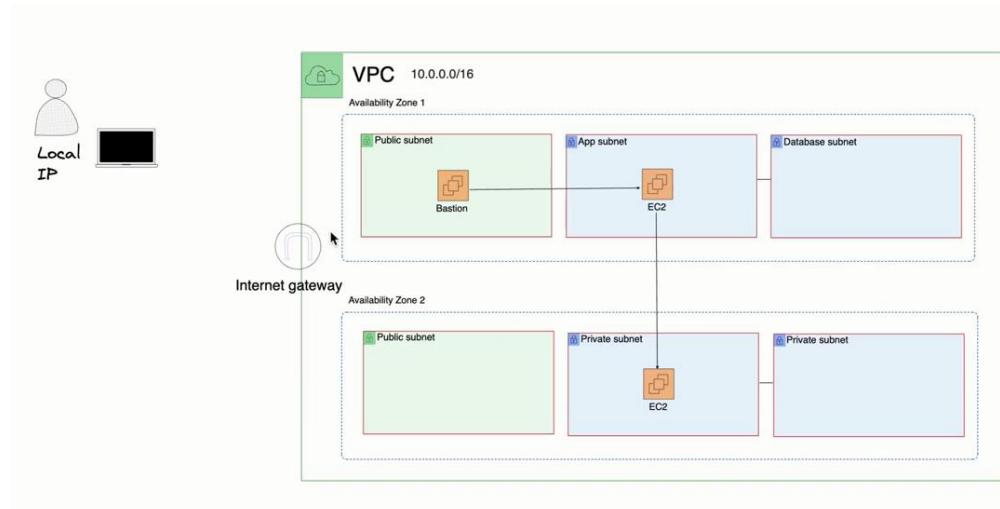
- RouteTables (1/1) info**: A list of route tables. One route table is selected: **rtb-045442d407e3d0e2f / project-rtb-public**. It has 2 subnets associated with it. The status is "No" and the VPC is "vpc-02ae36fa855f115c".
- Routes (2)**: A detailed view of the routes in the selected route table. There are two routes:

Destination	Target	Status	Propagated
0.0.0.0/0	igw-071679543994ebd54	Active	No
10.0.0.0/16	local	Active	No
- Edit routes**: A form to modify routes. It shows two routes:

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	Internet Gateway	Active	No

Buttons at the bottom include "Remove" (highlighted with a yellow arrow), "Cancel", "Preview", and "Save changes" (highlighted with a yellow arrow).

- After removing all possible routes for the Internet gateway, there isn't any route to the IGW from the public subnet. We did not detach the IGW from the VPC but there isn't any remaining route toward the IGW.



```

Downloads — ssh -i bastion.pem ec2-user@3.8.236.121 — 94x32
Last login: Sat Mar  9 18:03:47 on ttys000
[soleyman@Soleymans-MBP ~ % cd Downloads
[soleyman@Soleymans-MBP Downloads % ssh -i bastion.pem ec2-user@3.8.236.121

```

- After changing the directory to Downloads file where the key pair is located, we try again to SSH into the public subnet but it is not allowed due to the removal of all possible routes to the IGW from the public subnet.

VPC > Route tables > rtb-045442d407e3d0e2f > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	Internet Gateway	-	No
	igw-071679543994eb54	-	

Add route Remove Cancel Preview Save changes

- We are now adding the route for IGW once again with the correct VPC ID.

VPC dashboard > Updated routes for rtb-045442d407e3d0e2f / project-rtb-public successfully

Details

rtb-045442d407e3d0e2f / project-rtb-public

Details Info

Route table ID	Main	Explicit subnet associations	Edge associations
rtb-045442d407e3d0e2f	No	2 subnets	-
VPC	Owner ID		
vpc-02ae3f6fa85ff115c project-vpc	8257655379748		

Routes Subnet associations Edge associations Route propagation Tags

Routes (2)

Destination	Target	Status	Propagated
0.0.0.0/0	igw-071679543994eb54	Active	No
10.0.0.0/16	local	Active	No

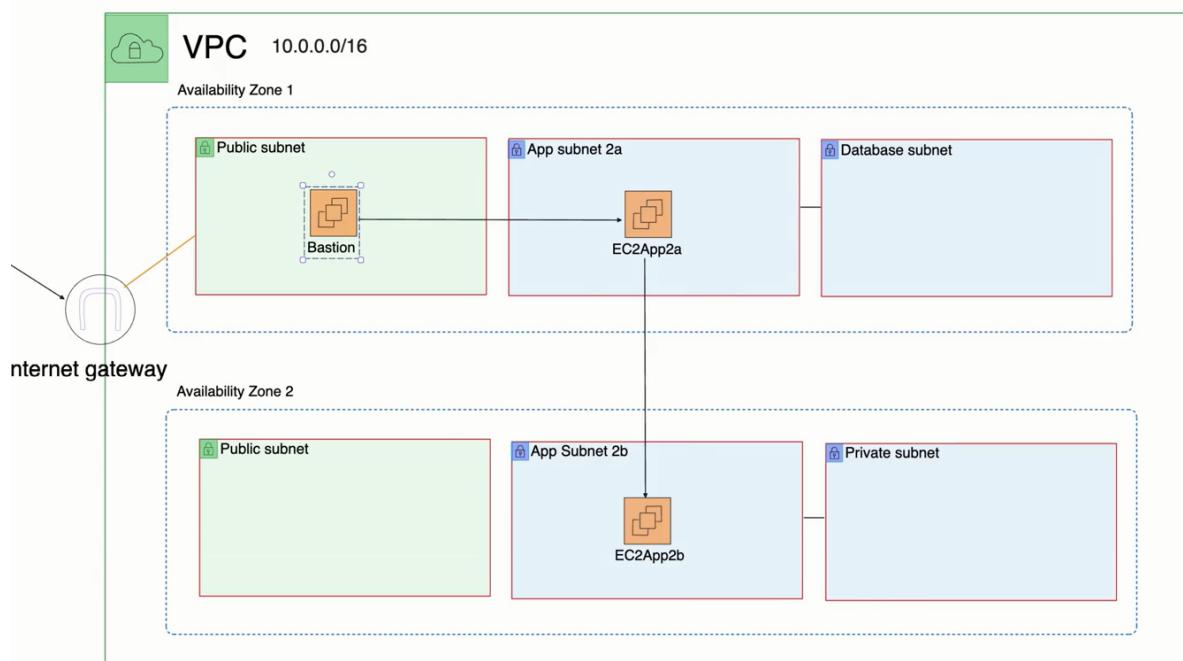
Both Edit route < 1 >

```

Downloads — ec2-user@ip-10-0-8-95:~ — ssh -i bastion.pem ec2-user@3.8.236.121 — 94x32
Last login: Sat Mar  9 18:03:47 on ttys000
[soleyman@Soleymans-MBP ~ % cd Downloads
[soleyman@Soleymans-MBP Downloads % ssh -i bastion.pem ec2-user@3.8.236.121
ssh: connect to host 3.8.236.121 port 22: Operation timed out
[soleyman@Soleymans-MBP Downloads % ssh -i bastion.pem ec2-user@3.8.236.121
#_
~\_ #####_          Amazon Linux 2023
~~ \#####\
~~   \###|
~~     \#/ ___ https://aws.amazon.com/linux/amazon-linux-2023
~~       V~' '-->
~~     /
~~.._./
~/_/
/_m/
Last login: Sat Mar  9 14:04:29 2024 from 2.51.107.15
[ec2-user@ip-10-0-8-95 ~]$ I

```

v



The screenshot shows the AWS CloudShell interface with the following details:

- EC2 Dashboard**: Shows 1 instance.
- Instances**: Sub-options include Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, and Reservations.
- Images**: Sub-options include AMIs and AMI Catalog.
- Elastic Block Store**: Sub-options include Volumes, Snapshots, and Lifecycle Manager.
- Network & Security**: Sub-options include Security Groups, Elastic IPs, Placement Groups, Key Pairs, and Network Interfaces.
- Load Balancing**: Sub-options include CloudFront and Application Load Balancers.

Launch an instance (Info)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags (Info)

Name: EC2App02a

Application and OS Images (Amazon Machine Image) (Info)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

Search: Search our full catalog including 1000s of application and OS images

Recent AMIs: Amazon Linux, Ubuntu, Windows, Red Hat, SUSE Linux, Debi

Browse more AMIs: Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI: ami-04fdeaa8c25817cd69 (64-bit (x86), uefi-preferred) / ami-0ca3c47f559a03429 (64-bit (Arm), uefi)

Virtualization type: EMR enabled: true Root device type: ebs

Summary

Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.5.2...read more

Virtual server type (instance type): t2.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel | Launch instance | Review commands

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy

Key pair (login) (Info)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name: required

Bastion

Create new key pair

Network settings (Info)

VPC - required: vpc-02ae316fa855f115c (project-vpc)

Subnet: subnet-02ae316fa855f115c project-subnet-private1-us-west-1a

Auto-assign public IP: Disable

Firewall (security groups): Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group | Select existing security group

Security group name: required

EC2App02aSG

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 64 characters.

Summary

Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.5.2...read more

Virtual server type (instance type): t2.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel | Launch instance | Review commands

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy

- We are going to create an EC2 instance in a private subnet in us-west-1a. Auto-assign public IP is disabled and in case of a key pair, a new key pair should be created for the new EC2 instance in the private subnet. However, in this case, we simply attach the same key pair that we used for the bastion host.

- We are going to revise inbound security group rules soon.
And also we should create a new EC2 instance in a different private subnet in a different AZ. In this case, we do not need to attach a key pair.

Instance type

t2.micro
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand RHEL base pricing: 0.0282 USD per Hour
On-Demand SUSE base pricing: 0.0138 USD per Hour
On-Demand Windows base pricing: 0.0184 USD per Hour
On-Demand Linux base pricing: 0.0138 USD per Hour

Key pair (login)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Network settings

VPC - required

vpc-02ae3f6fa855f115c (project-vpc)
10.0.0.16

Subnet

subnet-0501a560bd56c45bf project-subnet-private2-us-west-1b
VPC: vpc-02ae3f6fa855f115c Owner: 825765379748 Availability Zone: us-west-1b
Zone type: Availability Zone IP addresses available: 4091 CIDR: 10.0.144.0/20

Summary

Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.5.2...read more
ami-04fd8e25817cd69

Virtual server type (instance type): t2.micro

Firewall (security group): New security group

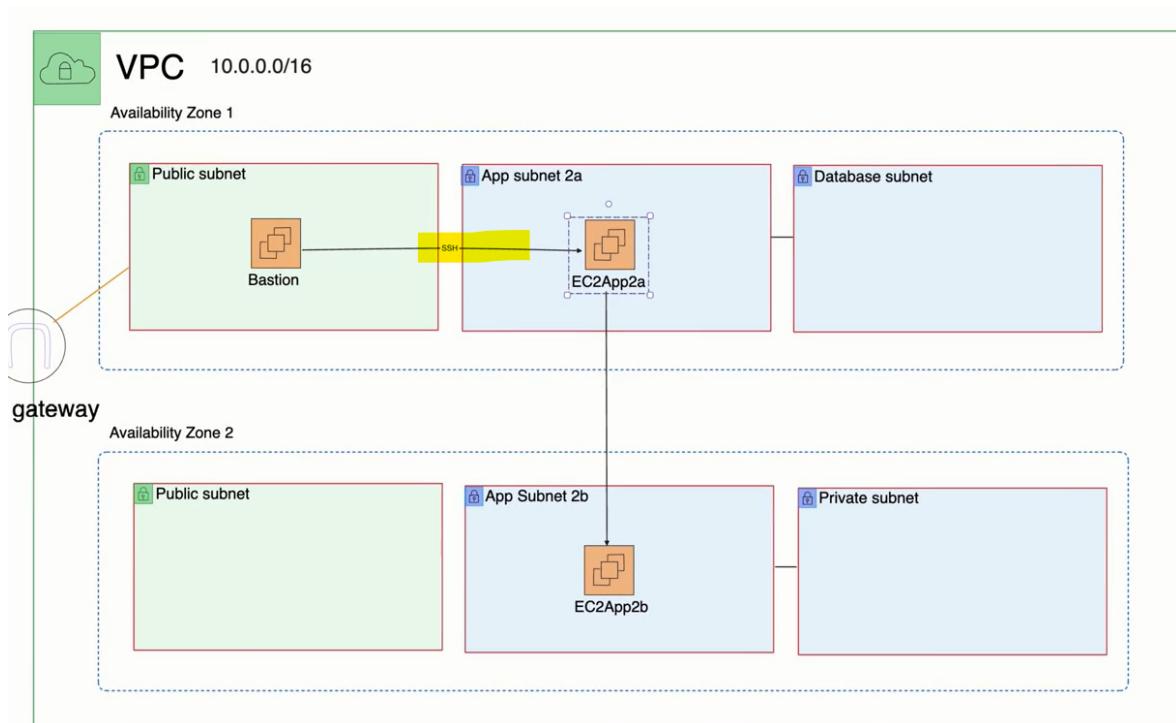
Storage (volumes): 1 volume(s) - 8 GiB

Free tier: in your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.

Cancel Launch instance Review commands

제 16강. Updating SGs

- We need to allow SSH connection from the Bastion Host to EC2App2a instance in a private subnet.



Screenshot of the AWS EC2 Instances page showing the status of EC2 instances:

Instances (1/4) info

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs
linux_ec2	i-08929ff6fac6bd3a	Stopped	t2.micro	-	View alarms +	us-west-1a	-	-	-	-
BastionHost	i-0ce566170f0f18d1ba	Running	t2.micro	2/2 checks passed	View alarms +	us-west-1a	ec2-54-67-113-123.us...	54.67.113.123	-	-
EC2App2a	i-08cc5b84a0a95daac	Running	t2.micro	2/2 checks passed	View alarms +	us-west-1a	-	-	-	-
EC2App2b	i-01a880c38fd90469	Running	t2.micro	2/2 checks passed	View alarms +	us-west-1b	-	-	-	-

i-08cc5b84a0a95daac (EC2App2a)

Security details

- IAM Role: -
- Owner ID: 825765379748
- Launch time: Sun Sep 01 2024 19:23:49 GMT-0700 (한국 표준시)

Security groups

- sg-071dd2362630ed74f (EC2App2aSG)

Inbound rules

Name	Security group rule ID	Port range	Protocol	Source	Security groups	Description
-	sgr-02d7fb5ecf0a71615	22	TCP	0.0.0.0/0	EC2App2aSG	-

Outbound rules

Name	Security group rule ID	Port range	Protocol	Source	Destination	Description
-	-	-	-	-	-	-

The screenshot shows the AWS EC2 Dashboard with the 'Security Groups' section selected. A single security group, 'sg-071dd2362630ed74f - EC2App2aSG', is listed. The 'Inbound rules' tab is active, showing one rule: 'sgr-02d7fb3ecfa71615' (Type: SSH, Protocol: TCP, Port range: 22, Source: 0.0.0.0/0). The 'Details' tab is also visible.

- When editing inbound rules, the default mode is that any traffic 0.0.0.0/0 can SSH into the EC2App2a instance. Just delete this rule and add a new rule, including a security group of the Bastion Host or a private IP address for the Bastion Host.

The screenshot shows the 'Edit inbound rules' dialog for the security group 'sg-071dd2362630ed74f'. The 'Delete' button for the existing SSH rule is highlighted. Below it, a new rule is being added: Type: ICMP, Protocol: ICMP, Port range: All, Source: sg-071dd2362630ed74f (selected from a dropdown).

- In the 2nd EC2 instance in a different AZ, we do not need to SSH into the EC2App2b from the EC2APP2a. We just ping a request (IP address) to the EC2App2b and get a response from it. The security group type is ALL ICMP – Ipv4 and attach the security group of the EC2App2a.

The screenshot shows the 'Edit inbound rules' dialog for the security group 'sg-074130ca3df602984 - EC2AppBSG'. A new rule is being added: Type: ICMP, Protocol: ICMP, Port range: All, Source: sg-071dd2362630ed74f (selected from a dropdown).

제 17강. Copying keypairs and pinging requests.

- We pass the key file to the bastion host placed in the public subnet to SSH into it via the Internet gateway.
- We copy the key file and pass it over to the EC2App2a in a private subnet in the AZ 1 to connect to this private instance via the bastion host.

- We need to copy the public IP address for the Bastion Host and the private IP address for the EC2App2a.

Name	Instance ID	Instance state	Instance type	Status check
BastionHost	i-051d1dc54ca841f4	Running	t2.micro	2/2 checks passed
EC2App2a	i-0b542f86c49781a74	Running	t2.micro	2/2 checks passed
EC2App2b	i-0ae07ae39e9630610	Running	t2.micro	2/2 checks passed

(1) SSH into the Bastion host.

- We need to change the directory first where the key pair for the bastion host is placed.

```
soleyman@Soleymans-MBP Downloads % ssh -i bastion.pem ec2-user@35.178.154.182
The authenticity of host '35.178.154.182' (35.178.154.182) can't be established.
ED25519 key fingerprint is SHA256:NNUOehOiaIUEFYrMFG1TAuQsk2bjb36Uu8pb3cHv+SI.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:7: 3.8.236.121
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '35.178.154.182' (ED25519) to the list of known hosts.

      _#
     ~\_\#\#\#_          Amazon Linux 2023
     ~~\_\#\#\#\\
     ~~  \#\#\|
     ~~    \#/ -- https://aws.amazon.com/linux/amazon-linux-2023
     ~~    V~' T->
     ~~~   /
     ~~.~.  /_
     _/_ /_
     _/m/' 

Last login: Sat Mar  9 14:12:26 2024 from 2.51.107.15
[ec2-user@ip-10-0-8-95 ~]$ ls
[ec2-user@ip-10-0-8-95 ~]$ ls
[ec2-user@ip-10-0-8-95 ~]$ 
```

- We passed the key pair, bastion.pem to the EC2 user with the public IP address of the Bastion Host. And when we put `ls`, there is nothing in the Bastion Host. At this time, the personal local IP address must be added into the inbound rule of the Bastion Host.

- After logging into the Bastion Host, we need to copy the key pair placed in our local machine to the Bastion host, opening a new tab on CLI and type this command to copy the key pair in your local machine.

- `scp -i<your_bastion_key_file> <give a new key name "private.pem"> ec2-user@<bastion-public-ip>:~/`

- After this, get back to the original CLI and enter `ls` and there should be the copied key file on your Bastion host.

```

      _' '
 _/m/' 
Last login: Sat Mar  9 14:12:26 2024 from 2.51.107.15
[ec2-user@ip-10-0-8-95 ~]$ ls
[ec2-user@ip-10-0-8-95 ~]$ ls
[ec2-user@ip-10-0-8-95 ~]$ ls
bastion.pem
[ec2-user@ip-10-0-8-95 ~]$ 

```

- After this, we are going to log into the EC2App2a instance.

```

$ ssh -I bastion.pem ec2-user@<the private IP address for the private instance>
[ec2-user@ip-10-0-8-95 ~]$ ls
[ec2-user@ip-10-0-8-95 ~]$ ls
[ec2-user@ip-10-0-8-95 ~]$ ls
bastion.pem
[ec2-user@ip-10-0-8-95 ~]$ ssh -i bastion.pem ec2-user@10.0.137.18
The authenticity of host '10.0.137.18 (10.0.137.18)' can't be established.
ED25519 key fingerprint is SHA256:+qhLVyn5TJoZi/lkHdobj2F6yuPUfi7BItl6G+elH2g.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.137.18' (ED25519) to the list of known hosts.

      #_
 ~\_ #####_          Amazon Linux 2023
 ~~ \#####\
 ~~  \###|
 ~~   \#/ ___  https://aws.amazon.com/linux/amazon-linux-2023
 ~~    V~' '-->
 ~~~   /
 ~~..-./
 _/ _/
 _/m/' 
[ec2-user@ip-10-0-137-18 ~]$ 

```

- Next, we should ping the 3rd private EC2 instance IP address

\$ ping <the private IP address for the 3rd EC2 instance in another private subnet> and then we can see the responses back from the 3rd EC2 instance.

```

bastion.pem
[ec2-user@ip-10-0-8-95 ~]$ ssh -i bastion.pem ec2-user@10.0.137.18
The authenticity of host '10.0.137.18 (10.0.137.18)' can't be established.
ED25519 key fingerprint is SHA256:+qhLVyn5TJoZi/lkHdobj2F6yuPUfi7BItl6G+elH2g.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.137.18' (ED25519) to the list of known hosts.

      #_
 ~\_ #####_          Amazon Linux 2023
 ~~ \#####\
 ~~  \###|
 ~~   \#/ ___  https://aws.amazon.com/linux/amazon-linux-2023
 ~~    V~' '-->
 ~~~   /
 ~~..-./
 _/ _/
 _/m/' 
[ec2-user@ip-10-0-137-18 ~]$ ping 10.0.146.173
PING 10.0.146.173 (10.0.146.173) 56(84) bytes of data.
64 bytes from 10.0.146.173: icmp_seq=1 ttl=127 time=1.39 ms
64 bytes from 10.0.146.173: icmp_seq=2 ttl=127 time=1.04 ms
64 bytes from 10.0.146.173: icmp_seq=3 ttl=127 time=1.05 ms
64 bytes from 10.0.146.173: icmp_seq=4 ttl=127 time=1.14 ms
64 bytes from 10.0.146.173: icmp_seq=5 ttl=127 time=1.10 ms
64 bytes from 10.0.146.173: icmp_seq=6 ttl=127 time=1.24 ms
64 bytes from 10.0.146.173: icmp_seq=7 ttl=127 time=1.19 ms
64 bytes from 10.0.146.173: icmp_seq=8 ttl=127 time=1.02 ms
64 bytes from 10.0.146.173: icmp_seq=9 ttl=127 time=1.03 ms
64 bytes from 10.0.146.173: icmp_seq=10 ttl=127 time=1.05 ms
64 bytes from 10.0.146.173: icmp_seq=11 ttl=127 time=0.966 ms

```

If you want to end the ping process, press **ctrl+c**

What happens if we remove the inbound rule of the EC2App2b to prevent the ping?

Screenshot of the AWS EC2 Instances page showing three instances: BastionHost, EC2App2a, and EC2App2b. EC2App2b is selected.

Name	Instance ID	Instance state	Instance type	Status check
BastionHost	i-051d1dd54ca841f4	Running	t2.micro	2/2 checks passed
EC2App2a	i-0b542f86c49781a74	Running	t2.micro	2/2 checks passed
EC2App2b	i-0ae07ae39e9630610	Running	t2.micro	2/2 checks passed

Instance: i-0ae07ae39e9630610 (EC2App2b)

Security details

IAM Role: - Owner ID: 891377277583 Launch time: Mon Mar 11 2024 16:03:50 GMT+0400 (Gulf Standard Time)

Security groups: sg-04de21da067358bd4 (EC2AppBSG)

Inbound rules

Security group rule ID	Port range	Protocol	Source	Security groups	Description
sgr-0d06da4fe67970397	All	ICMP	sg-07947cd0d5b056351	EC2AppBSG	-
sgr-00177800ee38715fb	22	TCP	0.0.0.0/0	EC2AppBSG	-

Outbound rules

Screenshot of the AWS Security Groups page showing the inbound rules for security group sg-04de21da067358bd4 - EC2AppBSG.

Edit inbound rules

Inbound rules control the incoming traffic that's allowed to reach the instance.

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-0d06da4fe67970397	All ICMP - IPv4	ICMP	All	Custom	<input type="text"/> sg-07947cd0d5b056351 <input type="button" value="Delete"/>
sgr-00177800ee38715fb	SSH	TCP	22	Custom	<input type="text"/> 0.0.0.0/0 <input type="button" value="Delete"/>

Add rule

Warning: Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Buttons: Cancel, Preview changes, Save rules

- In the inbound rule of the security group for the EC2App2b, we removed the rule allowing the ping from the EC2App2a.

```

64 bytes from 10.0.146.173: icmp_seq=4 ttl=127 time=1.14 ms
64 bytes from 10.0.146.173: icmp_seq=5 ttl=127 time=1.10 ms
64 bytes from 10.0.146.173: icmp_seq=6 ttl=127 time=1.24 ms
64 bytes from 10.0.146.173: icmp_seq=7 ttl=127 time=1.19 ms
64 bytes from 10.0.146.173: icmp_seq=8 ttl=127 time=1.02 ms
64 bytes from 10.0.146.173: icmp_seq=9 ttl=127 time=1.03 ms
64 bytes from 10.0.146.173: icmp_seq=10 ttl=127 time=1.05 ms
64 bytes from 10.0.146.173: icmp_seq=11 ttl=127 time=0.966 ms
64 bytes from 10.0.146.173: icmp_seq=12 ttl=127 time=1.82 ms
64 bytes from 10.0.146.173: icmp_seq=13 ttl=127 time=0.988 ms
^C
--- 10.0.146.173 ping statistics ---
13 packets transmitted, 13 received, 0% packet loss, time 12017ms
rtt min/avg/max/mdev = 0.966/1.156/1.815/0.221 ms
[ec2-user@ip-10-0-137-18 ~]$ ping 10.0.146.173

```

- And in the CLI, we ping again to the 3rd EC2 instance due to the fact that the security group of the inbound rule for this process was removed.

```

13 packets transmitted, 13 received, 0% packet loss, time 12017ms
rtt min/avg/max/mdev = 0.966/1.156/1.815/0.221 ms
[ec2-user@ip-10-0-137-18 ~]$ ping 10.0.146.173
PING 10.0.146.173 (10.0.146.173) 56(84) bytes of data.
^C
--- 10.0.146.173 ping statistics ---
32 packets transmitted, 0 received, 100% packet loss, time 32215ms
[ec2-user@ip-10-0-137-18 ~]$

```

제 18강. Project Wrap UP and Terminate Instances

