

Cloud 기본 개념

목차

1. IAM

1-1 IAM이란?

1-2 IAM 액세스 관리

1-3 IAM 계정 생성

2. 키 생성

2-1 Access Key

2-2 Temporary Security Credentials(임시 자격 증명)

2-3 키 직접 생성하기

3. VPC

3-1 실전 구성하기

4. 라우팅 테이블

5. 인터넷 게이트웨이 (Internet Gateway)

5-1 인터넷 게이트웨이 실습

1. IAM

1-1 IAM이란?

AWS에는 두 가지 유형의 권한 계정이 있다. 하나는 루트 사용자(Root User)이고 다른 하나는 IAM(Identity Access Management) 사용자이다. 루트 사용자는 AWS 계정을 생성할 때 기본적으로 생성되는 첫 번째 ID로, 계정의 모든 것을 제어할 수 있다.

IAM(AWS Identity and Access Management)은 리소스에 대한 액세스를 안전하게 제어할 수 있는 웹 서비스이다. IAM은 AWS 리소스를 제어하는 권한을 관리하며, AWS 계정에 대한 인증 및 권한 부여를 제어하는 데 필요한 인프라를 제공한다.

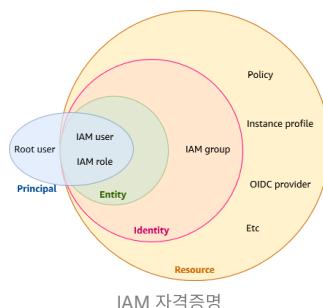
AWS 계정은 루트 사용자라는 단일 로그인 자격 증명으로 시작한다. 보안상의 이유로 일상적인 작업에 루트 사용자를 사용하지 않을 것을 강력히 권장한다. 루트 사용자는 루트 권한이 필요한 작업에만 사용하고, 이후에는 IAM을 통해 관리자, 분석가, 개발자 등 역할에 맞는 사용자 ID를 생성하여 필요한 권한만 부여한다.

IAM을 사용하면 다음과 같은 작업을 수행할 수 있다.

- AWS 계정에 대한 공유 액세스 권한 부여
- Amazon EC2에서 실행되는 애플리케이션을 위한 보안 AWS 리소스
- 멀티 팩터 인증(MFA)
- 아이덴티티 페더레이션을 통한 외부 사용자 인증

또한, IAM은 세분화된 권한을 할당할 수 있어, 각 사용자가 특정 리소스에서 수행할 수 있는 작업을 정확히 제어할 수 있다. 이러한 액세스 제어는 AWS 환경의 보안을 유지하는 데 중요한 역할을 한다.

1-2 IAM 액세스 관리



1) IAM User(사용자)

AWS 계정에서 생성하는 엔터티로, IAM을 사용하는 인간 사용자 또는 워크로드를 의미한다.

2) IAM Group(그룹)

IAM 사용자의 집합으로, 다수의 사용자들에 대한 권한을 지정함으로써 해당 사용자들에 대한 권한을 더 쉽게 관리할 수 있다. (ex. Admins라는 사용자 그룹이 있으면, 그 그룹의 모든 사용자는 자동으로 Admins라는 그룹 권한을 갖게 됨)

3) IAM Role(역할)

계정에 생성할 수 있는, 특정 권한을 지닌 IAM 자격증명으로, 한 사람하고만 연관되지 않고 해당 역할이 필요한 사람이라면 누구든지 맡을 수 있어야 한다. (역할을 맡은 사람에게는 해당 역할 세션을 위한 임시 보안 자격 증명이 제공)

4) IAM Policy(정책)

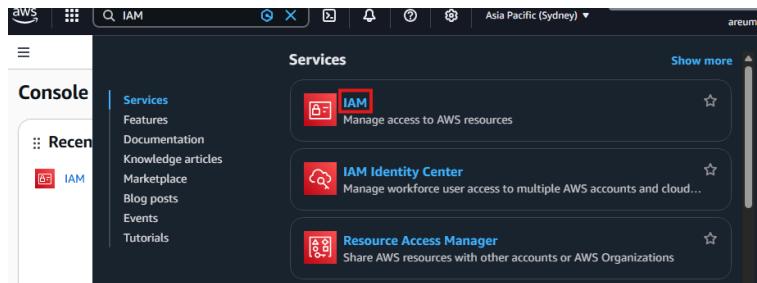
요청이 허용되거나 거부되는지를 결정한다. 대부분의 정책은 AWS에서 JSON 문서로 저장된다.

1-3 IAM 계정 생성

1. 대시보드 접속

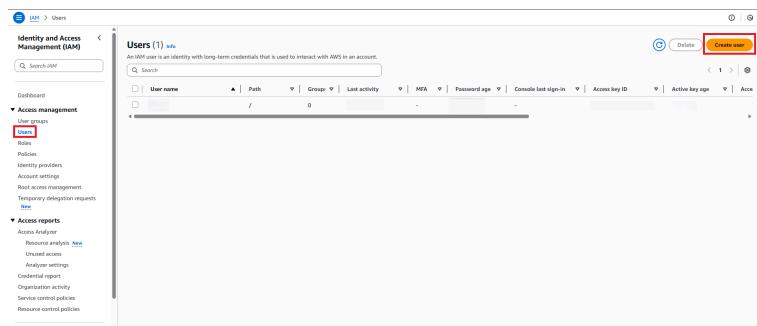
IAM 콘솔에 접속하여 대시보드 화면으로 이동한다.

<https://us-east-1.console.aws.amazon.com/iam/home?region=ap-southeast-2#/home>



2. Access management → Users → Create User

IAM 사용자 목록 화면에서 Create User를 클릭하여 신규 IAM 사용자 생성을 시작한다.



3. 사용자 세부 정보 지정

- 사용자 이름 : 신규 생성하려는 IAM 사용자명을 작성한다.
- AWS Management Console 체크한다. (웹 콘솔로 로그인을 가능하도록 설정)

→ 이 설정을 통해 해당 사용자는 AWS 웹 콘솔 로그인 권한을 갖게 된다.

User details

User name

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = , . @ _ - (hyphen)

Provide user access to the AWS Management Console - optional
In addition to console access, users with SignInLocalDevelopmentAccess permissions can use the same console credentials for programmatic access without the need for access keys.

Console password

Autogenerated password
You can view the password after you create the user.

Custom password
Enter a custom password for the user.

• Must be at least 8 characters long
• Must include at least three of the following mix of character types: uppercase letters (A-Z), lowercase letters (a-z), numbers (0-9), and symbols ! @ # \$ % ^ & * () _ + - (hyphen) = [] { }

Show password

Users must create a new password at next sign-in - Recommended
Users automatically get the IAMUserChangePassword policy to allow them to change their own password.

If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keypairs, you can generate them after you create this IAM user. [Learn more](#)

Cancel **Next**

4. 사용자 그룹을 생성

IAM 사용자에게 직접 권한을 부여하는 대신, 사용자 그룹을 생성하여 권한을 관리한다. 여러 사용자에게 동일한 권한을 효율적으로 부여하고 관리하기 위함이다.

Set permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Permissions options

Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

Copy permissions
Copy all group memberships, attached managed policies, and inline policies from an existing user.

Attach policies directly
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

Get started with groups
Create a group and select policies to attach to the group. We recommend using groups to manage user permissions by job function, AWS service access, or custom permissions. [Learn more](#)

Create group

Set permissions boundary - optional

Cancel **Previous** **Next**

5. 사용자 그룹 권한 설정

- 사용자 그룹 이름은 Administrators로 설정한다.
- 권한 정책에서 AdministratorsAccess를 선택한다.

AdministratorAccess 정책은 모든 AWS 리소스에 대해 모든 작업을 허용하는 정책으로, 관리자 역할의 사용자에게 부여되는 최고 수준의 권한이다.

Create user group

Create a user group and select policies to attach to the group. We recommend using groups to manage user permissions by job function, AWS service access, or custom permissions. [Learn more](#)

User group name
Enter a meaningful name to identify this group.

Administrators

Maximum 128 characters. Use alphanumeric and '+', '@', '-' characters.

Permissions policies (1/1109)

Filter by Type

Policy name	Type	Use...	Description
<input checked="" type="checkbox"/>  AdministratorAccess	AWS managed	None	Provides full access
<input type="checkbox"/>  AdministratorAccess-Amplify	AWS managed	None	Grants account admini
<input type="checkbox"/>  AdministratorAccess-AWSElasti...	AWS managed	None	Grants account adm
<input type="checkbox"/>  AIOpsAssistantIncidentReportP...	AWS managed	None	Provides permission
<input type="checkbox"/>  AIOpsAssistantPolicy	AWS managed	None	Provides ReadOnly p
<input type="checkbox"/>  AIOpsConsoleAdminPolicy	AWS managed	None	Grants full access to
<input type="checkbox"/>  AIOpsOperatorAccess	AWS managed	None	Grants access to the
<input type="checkbox"/>  AIOpsReadOnlyAccess	AWS managed	None	Grants ReadOnly pe
<input type="checkbox"/>  AlexaForBusinessDeviceSetup	AWS managed	None	Provide device setup
<input type="checkbox"/>  AlexaForBusinessFullAccess	AWS managed	None	Grants full access to
<input type="checkbox"/>  AlexaForBusinessGatewayExecu...	AWS managed	None	Provide gateway exe

Create user group

6. 사용자 그룹 선택

생성된 Administrators 그룹을 선택하여, 해당 IAM 사용자가 그룹에 포함되도록 설정한다.

Administrators user group created.

Review and create

Step 4
Retrieval password

Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

Attach policies directly
Attach a managed policy directly to a user instead of attaching policies to a group instead. Then, add the user to the appropriate group.

Copy permissions
Copy all group memberships, attached managed policies, and inline policies from an existing user.

User groups (1/1)

Create group

Administrators

Set permissions boundary - optional

Next

7. 설정 내용 확인

지금까지 설정한 사용자 이름, 콘솔 접근 여부, 그룹 및 권한 정책 정보를 최종 확인한다. 설정이 올바른 경우 Create User를 클릭하여 사용자 생성을 완료한다.

Review and create

Review your choices. After you create the user, you can view and download the autogenerated password, if needed.

User details

User name	Custom password type	Require password reset
ruhy	Custom password	Yes

Permissions summary

Name	Type	Used as
Administrators	Group	Permissions group
AWSConfigChangePassword	AWS managed	Permissions policy

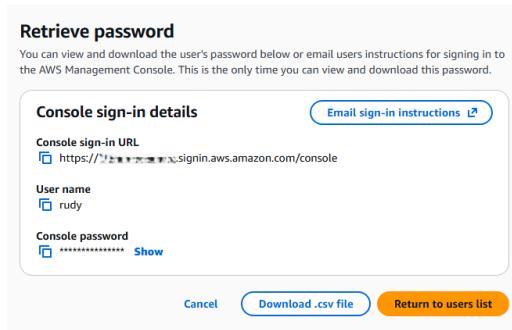
Tags - optional

Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.

Create user

8. 사용자 생성 완료

IAM 사용자가 정상적으로 생성되었으며, 해당 사용자는 부여된 권한 범위 내에서 AWS 리소스에 접근할 수 있다.



9. 최종 확인

IAM 사용자 목록 화면에서 새로 생성된 사용자가 표시되는 것을 확인한다.

이를 통해 IAM 사용자 생성 및 권한 설정이 정상적으로 완료되었음을 확인할 수 있다.

□ [rudy](#) / 1 - 4 minutes

2. 키 생성

2-1 Access Key

Access Key는 IAM 사용자 또는 AWS 계정 루트 사용자에 대한 장기 보안 인증으로, AWS CLI 또는 AWS API에 대한 프로그래밍 요청에 서명할 수 있다.

Access Key는 다음 두 가지 요소로 구성된다.

- Access Key ID
- Secret Access Key

이 두 값은 함께 사용되어 요청의 인증을 수행하며, Secret Access Key는 생성할 때만 가져올 수 있다.

IAM 사용자는 최대 2개의 Access Key를 가질 수 있으며, 사용 시에는 다음과 같은 보안 사항을 준수해야 한다.

- Access Key를 생성할 때는 계정의 루트 자격 증명을 사용하면 안된다.
- 애플리케이션 파일에 액세스 키나 자격 증명 정보를 넣으면 안된다.
- 프로젝트 영역에 액세스 키나 자격 증명 정보가 포함된 파일을 포함하면 안된다.
- 액세스 키를 권한 없는 사람에게 제공하면 안된다.

또한, Access Key 생성 후에는 AWS CloudTrail을 사용하여 액세스 키 사용량을 모니터링함으로써, 무단 액세스 시도를 탐지할 수 있다.

이러한 이유로 AWS에서는 장기 Access Key 사용을 최소화하고, 가능한 경우 IAM Role 기반의 임시 자격 증명 사용을 권장한다.

2-2 Temporary Security Credentials(임시 자격 증명)

AWS Security Token Service(AWS STS)는 AWS 리소스에 접근이 필요한 사용자나 애플리케이션에 임시 보안 자격 증명 (Temporary Security Credentials)을 생성하여 제공하는 서비스이다. 이를 통해 신뢰할 수 있는 사용자에게 일시적인 권한을 부여함으로써 AWS 리소스에 대한 접근을 제어할 수 있다. 임시 보안 자격 증명은 장기 액세스 키 자격 증명과 유사하게 작동하지만, 다음과 같은 차이점이 있다.

- 임시 보안 자격 증명은 단기적으로 사용된다.

자격 증명이 만료되면, AWS는 해당 자격 증명을 더 이상 인식하지 않아 이를 사용한 API 요청은 허용되지 않는다.

- 사용자에게 영구적으로 저장되지 않는다.

임시 보안 자격 증명은 사용자 계정에 저장되지 않고, 요청 시 동적으로 생성되어 제공된다. 자격 증명이 만료되었거나 만료되기 전이라도, 사용자가 여전히 권한을 보유하고 있다면 새로운 자격 증명을 다시 요청할 수 있다.

이러한 특성으로 인해 임시 보안 자격 증명은 장기 자격 증명에 비해 다음과 같은 장점을 가진다.

- 애플리케이션 장기 AWS 보안 자격 증명을 배포하거나 포함할 필요가 없다.
- AWS 계정을 위한 별도의 IAM 사용자를 생성하지 않고도 사용자에게 AWS 리소스 접근 권한을 제공할 수 있다.
- 자격 증명의 유효 기간이 제한되어 있으므로, 더 이상 필요하지 않을 경우 별도로 업데이트하거나 명시적으로 철회할 필요가 없다.

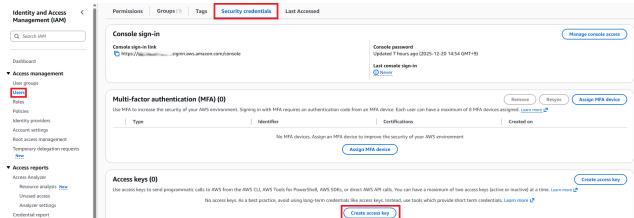
자격 증명이 만료되면 재사용이 불가능하며, 최대 유효 기간까지 직접 설정할 수 있다.

2-3 키 직접 생성하기

Access Key 생성

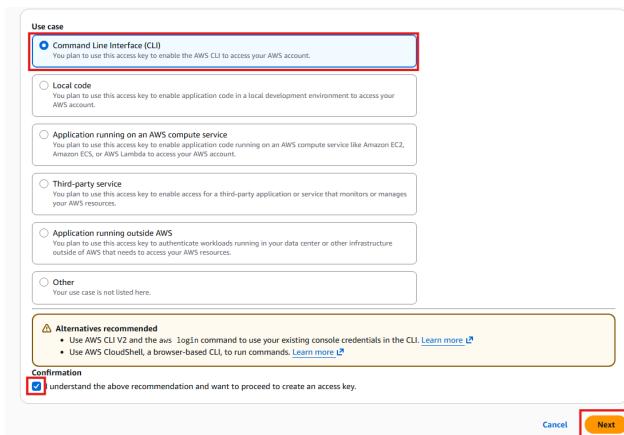
1. IAM 콘솔 → User → Security credentials → Create access Key

IAM 사용자에 대한 Access Key 생성을 위한 보안 자격 증명(Security credentials) 메뉴에서 Create access Key를 선택한다.



2. Use Case → Command Line Interface

본 실습에서는 AWS CLI를 활용한 리소스 접근을 확인하기 위해, Access Key 사용 목적을 Command Line Interface(CLI)로 선택하였다.



3. 설명 태그 설정

해당 access key는 실습 및 학습 목적으로 사용될 예정이므로, 설명 태그를 study라고 설정한 후 Create access Key를 클릭하여 키 생성을 진행한다.

Set description tag - optional info

The description for this access key will be attached to this user as a tag and shown alongside the access key.

Description tag value

Describe the purpose of this access key and where it will be used. A good description will help you rotate this access key confidently later.

Maximum 256 characters. Allowed characters are letters, numbers, spaces representable in UTF-8, and: _ . : / = + ~ @

Cancel
Previous
Create access key

4. Access Key 생성 완료

Access Key와 Secret Access Key가 생성되며, Secret Access Key는 이 단계에서만 확인할 수 있으므로 안전한 장소에 별도로 저장해야 한다.

⌚ This is the only time that the secret access key can be viewed or downloaded. You cannot recover it later. However, you can create a new access key any time. ×

Step 1
Access key best practices & alternatives

Step 2 - optional
Set description tag

Step 3
Retrieve access keys

Retrieve access keys info

Access key

If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.

Access key	Secret access key
	***** Show

Access key best practices

- Never store your access key in plain text, in a code repository, or in code.
- Disable or delete access key when no longer needed.
- Enable least-privilege permissions.
- Rotate access keys regularly.

For more details about managing access keys, see the [best practices for managing AWS access keys](#).

Download .csv file
Done

Temporary Security Credentials 생성

1. IAM User 생성

1-1 IAM 사용자 생성

위와 같이 IAM 사용자를 생성하되, 초기 단계에서는 별도의 권한을 부여하지 않고 생성한다. 이후 실습에서 필요한 권한만을 명시적으로 부여하기 위함이다.

Step 1 Specify user details Set permissions Review and create Step 4 Retrieve password

Set permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Permissions options

Add user to a group Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job functions.

Copy permissions Copy all group memberships, attached managed policies, and inline policies from an existing user.

Attach policy directly Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

User groups (1)

Group name	Attached... Created
Administrators	2025-12-20 (Yesterday)

Set permissions boundary - optional

Cancel
Previous
Next

1-2 사용자 정보 확인 후 생성

사용자 이름 및 접근 방식 정보를 확인한 뒤 IAM 사용자를 생성을 완료한다.

Review and create

User details

User name: dareum

Console password type: Custom password

Require password reset: Yes

Permissions summary

Name: IAMUserChangePassword

Type: AWS managed

Used as: Permissions policy

Tags - optional

Add new tag

Create user

1-3 생성된 사용자 선택

생성된 IAM 사용자를 클릭하여 권한 설정을 진행한다.

Users (3) info

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

Search

User name	Path	Group	Last activity	MFA	Password age	Console last sign-in	Access key ID	Active key age	Access
/	0	-	-	-	-	-	-	-	-
dareum	/	0	-	-	1 minute	-	-	-	-
rudy	/	1	-	-	Yesterday	-	-	22 hours	-

Delete Create user

2. Inline Policy를 통한 STS 권한 부여

2-1 Inline Policy 생성

IAM 사용자 메뉴에서 Permissions → Create Inline Policy를 선택한다.

2-2 Inline Policy란?

Inline Policy는 특정 IAM ID(사용자, 사용자 그룹 또는 역할)에 대해 하나에만 직접 연결되는 정책이다. 정책과 자격 증명이 1대 1로 결합되어 있어, 해당 자격 증명이 삭제되면 정책도 함께 삭제된다. 인라인 정책은 자격 증명을 생성할 때 함께 만들거나, 이미 존재하는 자격 증명에 나중에 추가할 수 있다.

본 실습은 특정 사용자에게만 STS AssumeRole 권한을 최소 법위로 부여하기 위한 목적이었기 때문에 인라인 정책을 사용하였다

Permissions

Groups Tags Security credentials Last Accessed

Permissions policies (1)

Permissions are defined by policies attached to the user directly or through groups.

Filter by Type

Search All types

Policy name	Type	Attached via
IAMUserChangePa...	AWS managed	Directly

Add permissions Create inline policy

2-3 STS 서비스 선택

정책 편집기에서 STS(Security Token Service) 서비스를 선택하고, STS 액션 중 AssumeRole만 허용하여 임시 보안 자격 증명 발급 권한만 부여한다. 리소스는 실습의 단순화를 위해 All로 설정하였다.

Specify permissions [info](#)
Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.

Policy editor

STS Action
Specify what actions can be performed on specific resources in STS.

Actions allowed
Specify actions from the service to be allowed.

Manual actions | Add actions | All STS actions (464+)

Access level
Read (S)
With selected (1/1)
All actions

AssumeRole [info](#)
 AssumeRoleWithWebIdentity [info](#)
 DecodeAuthorizationMessage [info](#)
 GetDelegatedCredentialsToken [info](#)
 SetContext [info](#)
 SetSourcedIdentity [info](#)

Effect
 Allow Deny

Tags (2)
Resources
Specify ARNs for these actions.

Specific

The 'all allowed' may be overly permissive for the selected actions. Allowing specific ARNs for these service resources can improve security.

2-4 Inline Policy 생성 완료

정책 이름을 입력하고, 설정된 내용을 확인한 후 인라인 정책을 생성한다. 이로써 해당 IAM 사용자는 STS AssumeRole 권한만을 가지게 된다.



2-5 IAM 사용자 인라인 정책 확인

IAM 사용자 dareum은 assume-role이라는 인라인 정책을 정상적으로 적용되었음을 확인한다.

Modify permissions in assume-role [info](#)
Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.

Policy editor

```
1 {
  "version": "2012-10-17",
  "Statement": [
    {
      "Sid": "visualizeditor",
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
      "Resource": "*"
    }
  ]
}
```

3. Role 생성 및 신뢰 정책 설정

3-1 Role 생성

STS AssumeRole 실습을 위해 새로운 IAM Role을 생성한다.

Identity & Access Management (IAM)

Roles (4) [info](#)
An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

Role name | Last activity

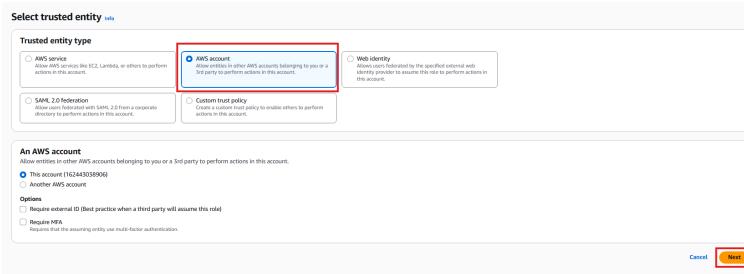
- AWSServiceRoleForResourceExplorer
- AWSServiceRoleForSupport
- AWSServiceRoleForThumbnailer
- visualizeditor

X.509 Standard [Edit](#) [Temporary credentials](#)

Temporary credentials
Use temporary credentials with ease and benefit from the enhanced security they provide.

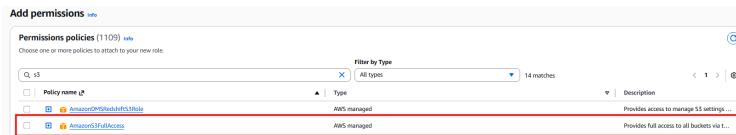
3-2 신뢰 주체 선택

Trusted entity type을 AWS account로 설정하여, 동일 AWS 계정 내 IAM 사용자가 해당 Role을 Assume할 수 있도록 구성한다.



3-3 Role에 권한 정책 부여

Role을 Assume한 이후 접근할 수 있는 리소스를 정의하기 위해 S3 접근 권한이 포함된 권한 정책을 Role에 연결한다.



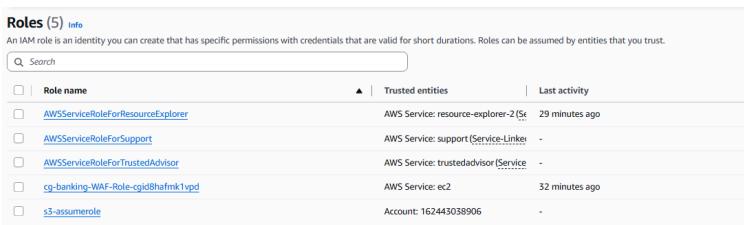
3-4 Role 이름 설정

생성할 IAM Role의 이름을 지정한다.



3-5 Role 생성 완료 확인

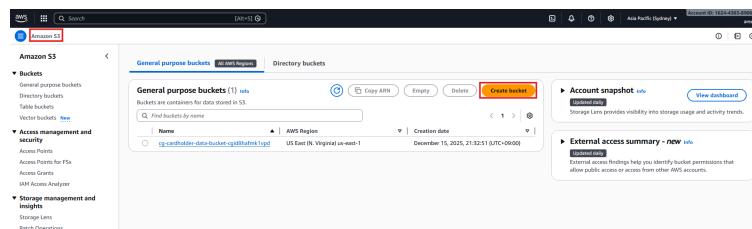
IAM Role이 정상적으로 생성되었음을 확인한다.



4. S3 버킷 만들기

4-1 S3 버킷 생성

STS AssumeRole을 통해 발급된 임시 보안 자격 증명이 실제 AWS 리소스 접근 권한으로 정상 동작하는지를 확인하기 위해 테스트용 S3 버킷을 생성한다.



4-2 버킷 기본 설정

버킷 이름, 리전 등 기본 설정을 지정하고 S3 버킷 생성을 완료한다.

Create bucket [Info](#)

Buckets are containers for data stored in S3.

General configuration

AWS Region
Asia Pacific (Seoul) ap-northeast-2

Bucket name [Info](#)
test-assume-01

Bucket names must be 3 to 63 characters and unique within the global namespace. Bucket names must also begin and end with a letter or number. Valid characters are a-z, 0-9, periods (.), and hyphens (-). [Learn more](#)

Copy settings from existing bucket - optional
Only the bucket settings in the following configuration are copied.

[Choose bucket](#)

Format: s3://bucket/prefix

4-3 객체 업로드

버킷 접근 여부를 확인하기 위해 테스트용 파일을 업로드한다.

General purpose buckets (2) [Info](#)

Buckets are containers for data stored in S3.

Find buckets by name

Name	AWS Region	Creation date
cg-cardholder-data-bucket-cgid8hafmk1vpd	US East (N. Virginia) us-east-1	December 15, 2025, 21:32:51 (UTC+09:00)
test-assume-01	Asia Pacific (Seoul) ap-northeast-2	December 21, 2025, 21:42:01 (UTC+09:00)

test-assume-01 [Info](#)

[Objects](#) [Metadata](#) [Properties](#) [Permissions](#) [Metrics](#) [Management](#) [Access Points](#)

Objects (0)

[Create folder](#) [Upload](#)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Find objects by prefix

Name	Type	Last modified	Size	Storage class
No objects				

You don't have any objects in this bucket.

[Upload](#)

Upload [Info](#)

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDKs or Amazon S3 REST API. [Learn more](#)

Drag and drop files and folders you want to upload here, or choose [Add files](#) or [Add folder](#).

Files and folders (1 total, 652.7 KB)

All files and folders in this table will be uploaded.

[Remove](#) [Add files](#) [Add folder](#)

<input type="text"/> Find by name	Folder	Type	Size
배경화면.png	-	image/png	652.7 KB

Upload succeeded
For more information, see the [Files and folders](#) table.

Upload: status

After you navigate away from this page, the following information is no longer available.

Summary

Destination	Succeeded	Failed
s3://test-assume-01	1 file, 652.7 KB (100.00%)	0 files, 0 B (0%)

Files and folders (1 total, 652.7 KB)

Name	Folder	Type	Size	Status	Error
배경화면.png	-	image/png	652.7 KB	Succeeded	-

5. EC2 인스턴스 생성 및 CLI 연결

5-1 EC2 인스턴스 생성

AWS CLI 실습을 수행하기 위해 EC2 인스턴스를 생성한다.

EC2 Instances

Instances

Launch instances

5-2 인스턴스 설정

인스턴스 이름을 지정하고 Amazon Linux 이미지를 선택한다.

Name and tags

Name: assume-instance

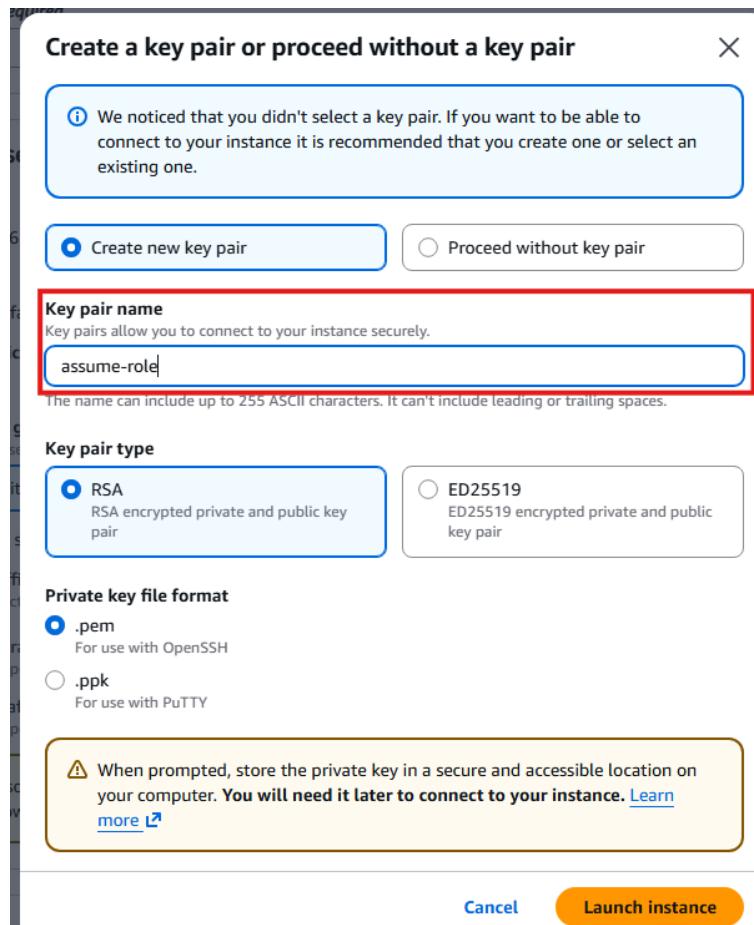
Application and OS Images (Amazon Machine Image)

Amazon Machine Image (AMI)

Amazon Linux 2023 kernel-6.1 AMI

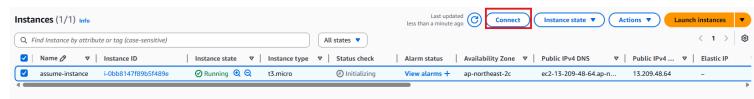
5-3 키 페어 설정

EC2 인스턴스 접근을 위한 Key Pair name을 설정한 후 인스턴스를 생성한다.



5-4 EC2 연결 완료 화면

생성된 EC2 인스턴스에 연결하여 CLI 환경을 구성한다.



5-5 EC2 연결 완료 화면

EC2 인스턴스에 정상적으로 접속된 화면이다.



6. STS AssumeRole 수행

6-1 S3 ARN 복사

접근 테스트에 사용할 S3 버킷의 ARN 정보를 확인한다.



6-2 AssumeRole 수행

AWS CLI에서 aws sts assume-role 명령을 실행하여 IAM Role에 대한 임시 자격 증명을 발급받는다.

```
aws sts assume-role \
--role-arn arn:aws:iam::162443038906:role/s3-assumerole \
--role-session-name MySessionName
```

```
[ec2-user@ip-172-31-33-12 ~]$ aws sts assume-role \
> --role-arn arn:aws:iam::162443038906:role/s3-assumerole \
> --role-session-name MySessionName
{
  "Credentials": {
    "AccessKeyId": "ASIA8LUS4CC5MGNYEP22",
    "SecretAccessKey": "VigKsA8Qf6tEl2f4dV2btPe6wvymYzDa2Yugt",
    "SessionToken": "IQoobo3JyZ2lXv2jvBuaMmWjN5vcnRo2WFzDc0yIkgrgIhAM5UUTwY6+OJW/FKBp3EINB+y7Weu
vSFVhpXhM#WaQQT/AiPAzt853u4dp2ZkDpTxV4J5gTdg400Ojg3SjolvBV7D7b3ggowII3w//////////ARAAgwxhNjI0NDMmzg5MDY
/DNUehpXh0i0hGVnqcbj3Aevi+0sR4ucy/Oo6Dl+XSwSm08kRfIKLvc1cDORYdtJ0x0n0sZtx+HedfRurY61Gz2X3kdv7uLyJ
7zLkcm0sAfpjueD06UQ0FRj15BwfwGSKUxGNiws5rczC19sxBh13ELzCjR3zpggWzKRdr2s7vrqGD2+msf4Fp01JESD8mNk11zu5
nxrh74ixDcPBEtr/3:1t35Igv8nME29Lcku9Wn98cRQsAeBa/+5tsKNU8sdH2nEMsjY0/vnVnJiuc+6os1pmQ4e25rwXXX3Ry0WJ1H
raTlWn7x2PjS8XMB9j0uqg+n0uBX/9VWW4AsGLb4tW3N2fyygY6naGPSS08gtOTGsyhHP2CBMuvQFxskJf65Uw96rOh+24ls
8UDx4aENwpVAnidw/bimpD4VsBrh0QeJyyKomVQEcC9x1xUfiiizF57lVvXWim9aveLb1lWMOIwVlWfrghvNyHxt/OytOYiKv/sid
2Bu15t5tJ8me7+rWnVxkoeH2cdKvRoed61xcpOz1wK/goOennWJWqzFixFaw=",
    "Expiration": "2025-12-21T13:50:04+00:00"
  },
  "AssumedRoleUser": {
    "AssumedRoleId": "AROASLUS4CC5NCFBG725:MySessionName",
    "Arn": "arn:aws:sts::162443038906:assumed-role/s3-assumerole/MySessionName"
  }
}
```

6-3 임시 자격 증명 적용 및 S3 접근 확인

발급받은 Access Key, Secret Key, Session Token을 AWS CLI에 적용하고 S3을 조회해보면 버킷명과 파일들이 나오는 것을 확인할 수 있다.

이는 AssumeRole을 통해 발급받은 임시 자격 증명에 S3 접근 권한이 포함되어 있기 때문이다.

```
aws configure set aws_access_key_id <AccessKeyId 문자열>
aws configure set aws_secret_access_key <SecretAccessKey 문자열>
aws configure set aws_session_token <SessionToken 문자열>

aws s3 ls #버킷 나열
aws se ls <버킷명> #해당 버킷에 있는 파일들 나열
```

```
[ec2-user@ip-172-31-33-12 ~]$ aws configure set aws_access_key_id ASIA8LUS4CC5MGNYEP22
[ec2-user@ip-172-31-33-12 ~]$ aws configure set aws_secret_access_key VigKsA8Qf6tEl2f4dV2btPe6wvymYzDa2Yugt
[ec2-user@ip-172-31-33-12 ~]$ aws configure set aws_session_token IQoobo3JyZ2lXv2jvBuaMmWjN5vcnRo2WFzDc0yIkgrgIhAM5UUTwY6+OJW/FKBp3EINB+y7Weu
[ec2-user@ip-172-31-33-12 ~]$ aws s3 ls
[ec2-user@ip-172-31-33-12 ~]$ aws s3 ls s3://test-assume-01
[ec2-user@ip-172-31-33-12 ~]$
```

3. VPC

Amazon Virtual Private Cloud(Amazon VPC)를 사용하면 정의한 논리적으로 격리된 가상 네트워크에서 AWS 리소스를 시작할 수 있다.

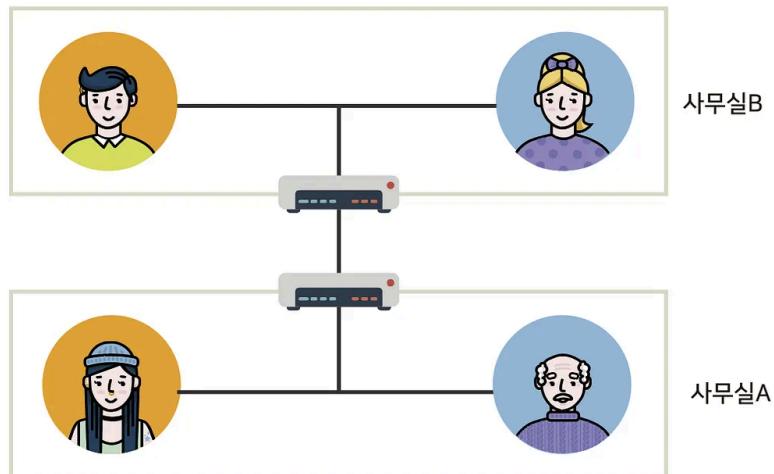
사용자는 자기가 원하는대로 IP 주소 범위 선택, 서브넷 생성, 라우팅 테이블 및 네트워크 게이트웨이 구성 등 가상 네트환경을 구성해 VPC를 구성할 수 있다.

한 마디로, AWS용 나만의 개인 네트워크 망 데이터센터라고 이해하면 된다.

VPC를 알아보기 전 알아야 할 지식들이 있다. 앞으로 VPC를 설정할 때 등장하는 개념들과 용어들이니 잘 익혀두도록 하자.

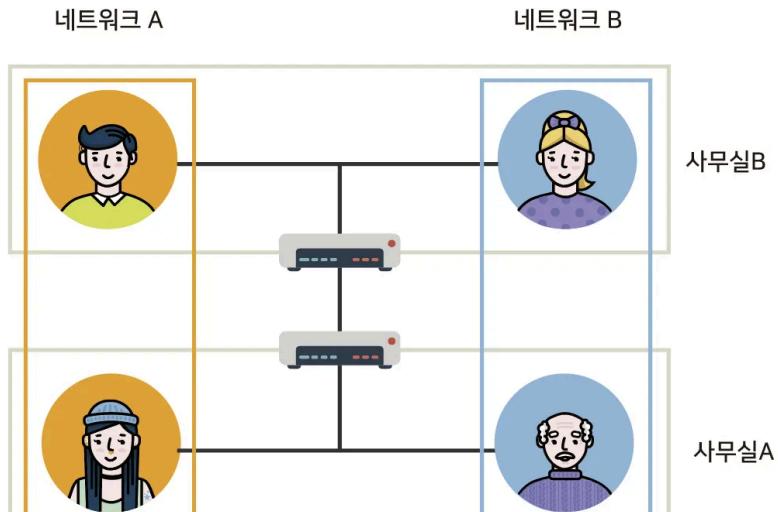
첫 번째로 VPN이다.

VPN(Virtual Private Network)은 가상사설망으로, 사용자가 사설망에 연결된 것처럼 인터넷에 엑세스할 수 있도록 하는 인터넷 보안 서비스이다. 인터넷 통신을 암호화할 뿐만 아니라 익명성 또한 제공한다.



VPN이 안되어 있는 경우

위 예시를 보면, 회사의 네트워크와 다음 그림과 같이 구성되어 있다고 가정하고 보안상의 이유로 직원간 네트워크를 분리하고자 할 때, 기존 인터넷선도 다시하고 다시 전용선을 깔아주어야 한다는 단점이 있다.



VPN이 있는 경우

하지만 VPN을 설치해주면 네트워크 A와 네트워크 B가 실제로 같은 네트워크상에 있지만 논리적으로 다른 네트워크인것처럼 동작한다. 이를 가상사설망이라고 한다.

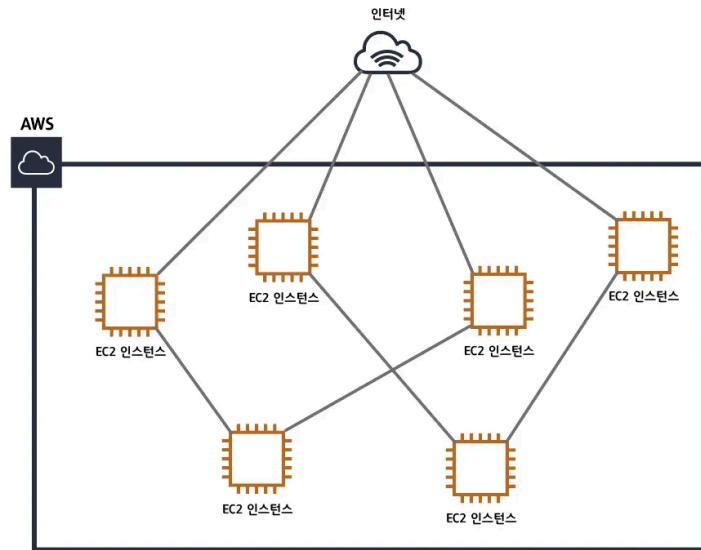
두 번째 개념들은 좀 길게 작성될 예정으로 새로운 페이지에서 다뤄보겠다.

Network 용어 정리

CIDR(Classless Inter-Domain Routing)

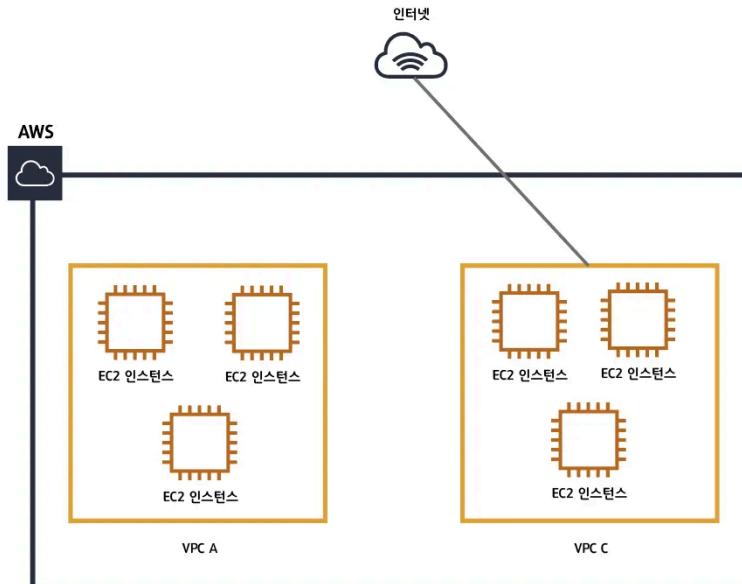
AWS 웹 서비스 구조

그렇다면 VPC는?



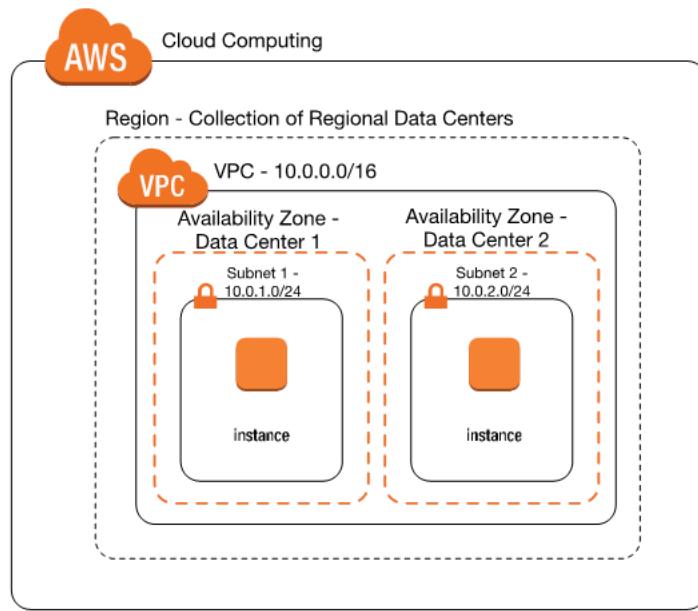
VPC가 없는 경우

VPC가 없다면 인스턴스들이 거미줄처럼 연결되고 인터넷과 연결된다. 이런 구조는 시스템의 복잡도를 끌어올릴뿐만 아니라 하나의 인스턴스만 추가되도 모든 인스턴스를 수정해야하는 불편함이 생긴다.



VPC가 있는 경우

VPC를 적용하면 VPC별로 네트워크를 구성할 수 있고 각각의 VPC에 따라 다르게 네트워크 설정을 줄 수 있다. 또한 각각의 VPC는 완전히 독립된 네트워크처럼 작동하게 된다.



위 그림을 살펴보면, AWS Cloud 내에는 IDC의 집합인 Region이 존재하며, region은 2개의 AZ로 이루어져 있다.

IDC란 인터넷 데이터 센터로의 준말로, 인터넷 연결의 핵심이 되는 서버를 모아 집중시킬 필요가 있을 때 설립하는 시설을 말한다.

그중에서 VPC는 region에 상응하는 규모의 네트워크라는 것을 알 수 있다.

즉, VPC는 독립된 하나의 네트워크를 구성하기 위한 가장 큰 단위이며, 특정 region에 종속되어 생성된다. 하나의 region에는 여러 개의 VPC를 생성할 수 있지만, 하나의 VPC를 여러 region에 걸쳐 확장하는 것은 불가능하다.

VPC는 각 region에 종속되어 RFC1918이라는 사설 IP 대역에 맞추어 설계해야 한다.

VPC에서 사용하는 사설 IP 대역은 아래와 같다.

- 10.0.0.0 ~ 10.255.255.255 (10/8 prefix)
- 172.16.0.0 ~ 172.31.255.255 (172.16/12 prefix)
- 192.168.0.0 ~ 192.168.255.255 (192.168/16 prefix)

AWS VPC는 On-premise(전통적인 인프라)와 동일한 대역의 사설 IP를 이용해 범위 내에서 VPC CIDR 영역을 설정한다. (한 리전안에 VPC를 여러 개 생성할 때 서로 아이피는 겹치지 안된다)

예를 들어, 기본 VPC가 172.31.0.0/16으로 생성되어 있다면, 우리는 172.31 네트워크 IP 주소를 피해서 생성해야 한다.

그리고 유의할 점이 있는데, 원래 규정된 사설 IP 범위와는 다르게 AWS에서는 [/16~/28 비트의 서브넷 마스크만을 허용한다는 것이다.](#)

즉, AWS VPC를 생성할 수 있는 가장 큰 IP 대역은 /16이며, 가장 작은 대역은 /28인 셈이다.

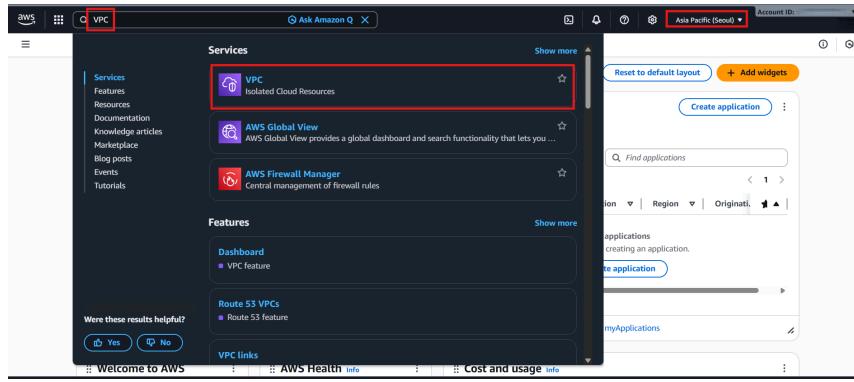
이 범위내에서만 CIDR를 설정해야 된다.

그리고 VPC에서는 한 번 설정된 IP 대역은 수정할 수 없으며, 각각의 VPC는 독립적이기 때문에 서로 통신할 수 없다.

3-1 실전 구성하기

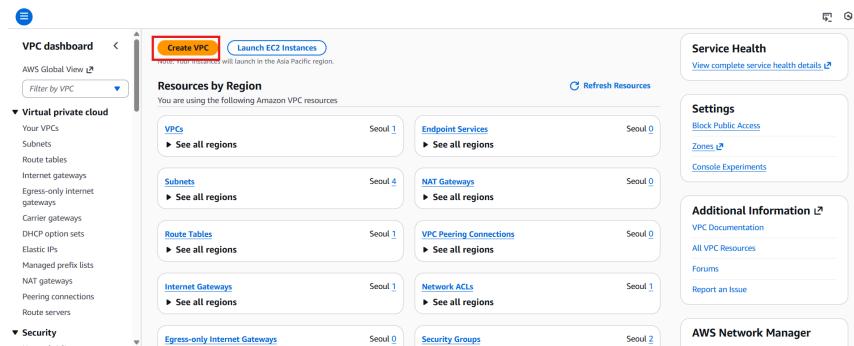
1. VPC 콘솔 진입 화면

AWS 콘솔에서 VPC 서비스를 검색하여 Virtual Private Cloud 관리 화면으로 이동한다.



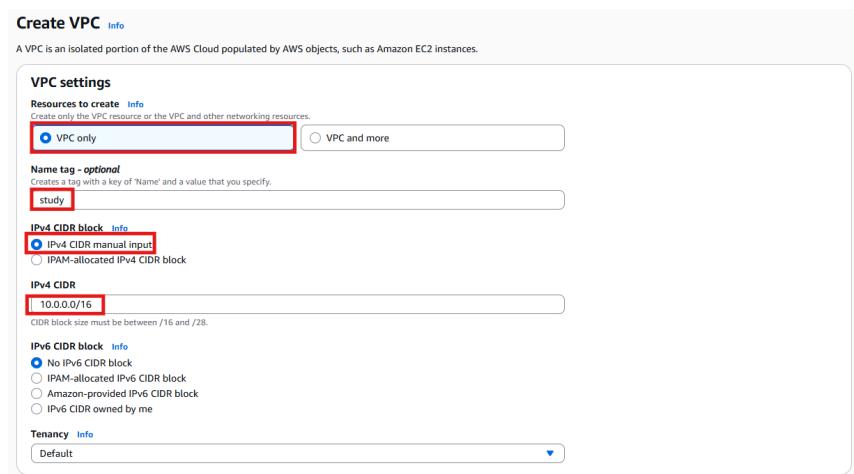
2. Create VPC

새로운 VPC 생성을 위해 Create VPC 버튼을 선택한다.



3. VPC 설정 화면

본 실습에서는 단일 VPC 화면을 구성하기 위해 VPC only 옵션을 선택하고, IPv4 CIDR 블록을 10.0.0.0/16으로 직접 지정하여 VPC를 생성하였다.



4. VPC 생성 결과 확인

설정한 값에 따라 study라는 이름의 VPC가 정상적으로 생성된 것을 확인할 수 있다.

Your VPCs (2) Info						
<input type="text" value="Find VPCs by attribute or tag"/>		VPC ID	State	Block Public...	IPv4 CIDR	IPv6 CIDR
<input type="checkbox"/>	Name	–	Available	<input type="radio"/> Off	172.31.0.0/16	–
<input type="checkbox"/>	study	–	Available	<input type="radio"/> Off	10.0.0.0/16	–

5. DNS Hostname 설정 화면

EC2 인스턴스에 퍼블릭 DNS 이름이 자동으로 할당되도록 VPC의 DNS Hostnames 옵션이 활성화되어 있는지 확인한다.



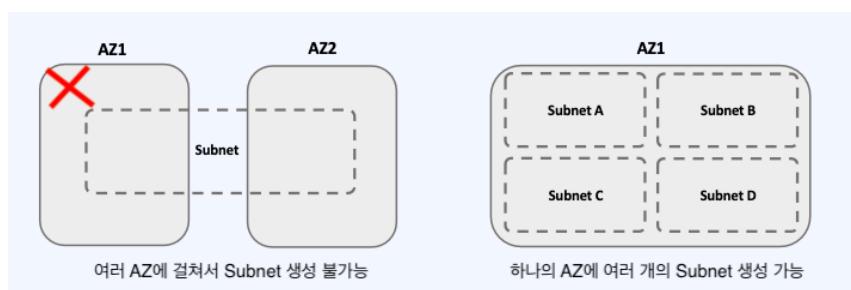
다음으로 서브넷을 생성할 것이다.

서브넷은 VPC의 IP 주소를 나누어 리소스가 배치되는 물리적인 주소 범위를 뜻한다.

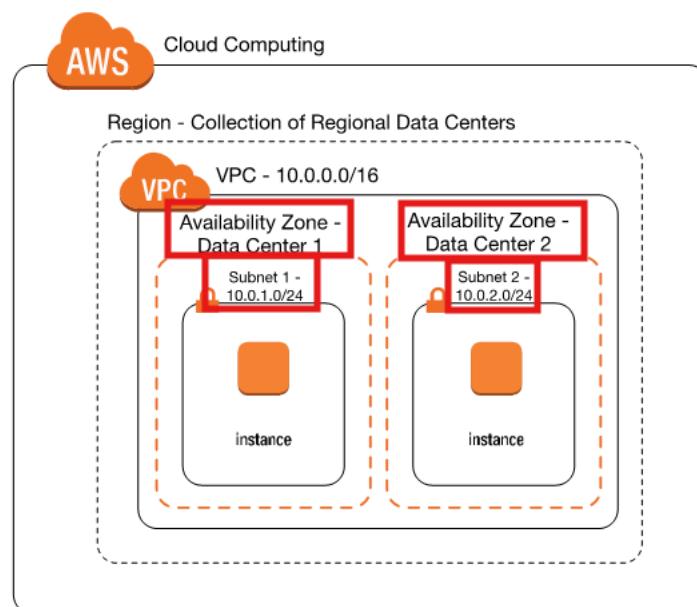
VPC가 논리적인 범위를 의미한다면, 서브넷은 VPC 안에서 실제로 리소스가 생성될 수 있는 네트워크 영역이라고 생각하면 된다. (EC2, RDS와 같은 리소스를 생성할 수 있다.)

하나의 VPC에 N개의 서브넷을 가질 수 있지만 하나의 AZ에서만 생성이 가능하다.

다음 사진처럼 여러 AZ에 걸쳐서 서브넷을 생성할 수 없다는 의미이다.



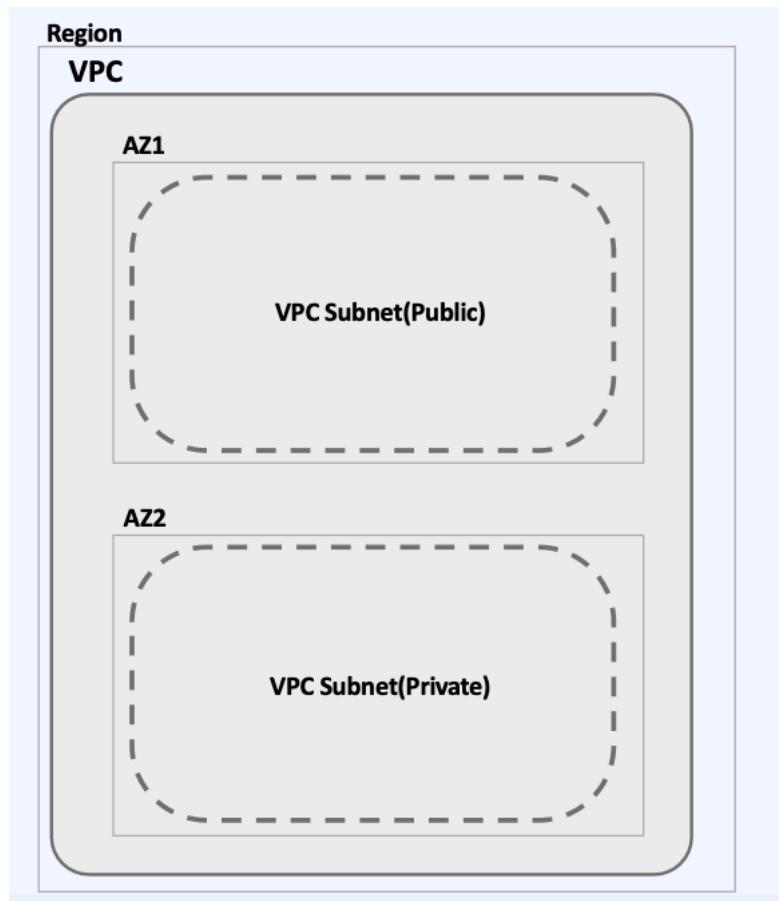
서브넷 역시 VPC처럼 CIDR 범위는 /16 ~ /28을 사용할 수 있으며, VPC CIDR 블록 범위에 속하는 CIDR 블록을 지정할 수 있다.



위 사진을 보면, VPC(10.0.0.16) 내에서 다시 크기를 두 개로 쪼개어 사용하게 되는데 VPC를 쪼갠 조각들이 바로 서브넷이다. 그리고 하 나의 AZ에 서브넷이 각각 할당되어 있는 것을 볼 수 있다.

VPC를 잘게 나눈 것이 서브넷이기 때문에 당연히 VPC보다 대역폭이 낮아야 한다.

서브넷은 다시 Public Subnet / Private Subnet으로 나뉜다.



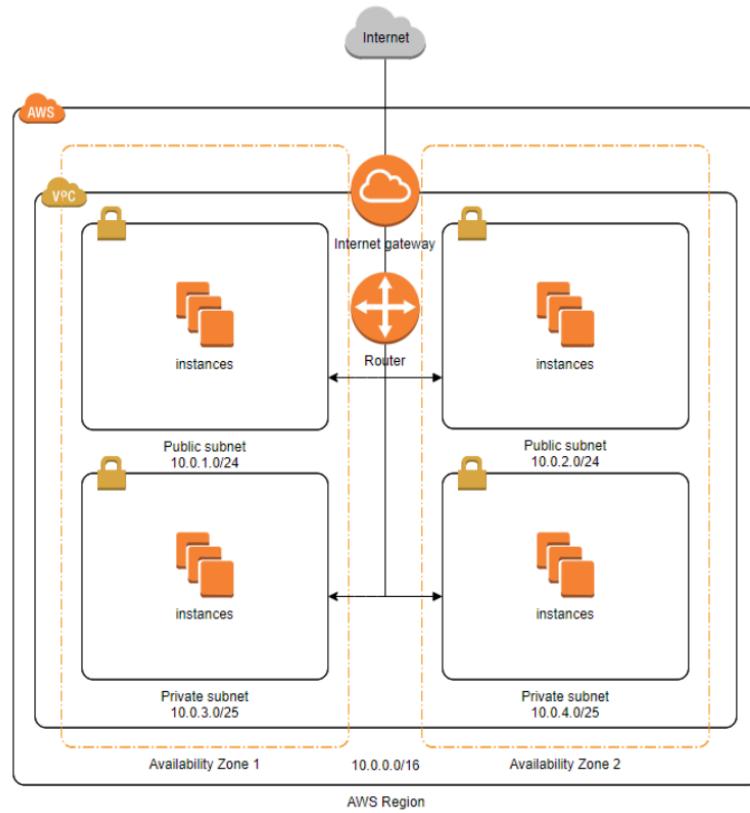
인터넷에 연결되어 있는 서브넷을 public subnet이라고 하고, 인터넷과 연결되어 있지 않은 서브넷을 private subnet이라고 한다.

- Private Subnet - 인터넷에 접근 불가능한 Subnet (VPC 내부에서만 통신)
- Public Subnet - 인터넷에 접근 가능한 Subnet (VPC 외부/내부와 통신)

따라서, public Subnet에 존재하는 인스턴스는 Public IP와 Elastic IP를 보유하여 인터넷에 연결되어 아웃바운드, 인바운드 트래픽을 주고받을 수 있는 반면, Private Subnet은 외부에 노출이 되어 있지 않아 접근할 수 없다.

(인바운드는 외부 → 인스턴스로 오는 트래픽, 아웃바운드는 인스턴스 → 외부로 가는 트래픽)

그래서 Private Subnet에 민감한 데이터 정보들을 저장해 보안을 강화하는 식으로 설계를 하는 편이다. 이처럼 Private는 서브넷 내부의 인스턴스들은 오직 다른 서브넷과만 연결이 가능한데, 만일 데이터를 업데이트하는데 연결이 필요라면 NAT Instance를 통해서 Private 내부의 인스턴스들이 인터넷을 가능하게 만들 수 있다.



<https://aws-hyoh.tistory.com/entry/VPC-쉽게-이해하기-1>

AZ 안에 Public Subnet과 Private Subnet을 배치했다.

Public Subnet인 10.0.0.1/24와 10.0.2.0/24은 EC2 다수를 탑재해야하니 다수의 사설 IP가 필요하므로 CIDR 값을 낮게 줘서 /24 Subnet 범위를 크게 나누었다.

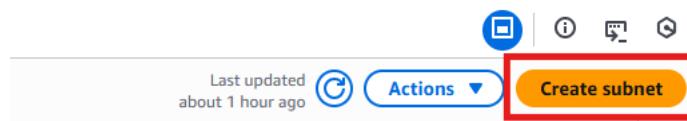
Private Subnet인 10.0.3.0.25와 10.0.4.0/25는 RDS와 같이 소수의 서비스를 탑재하므로 많이 필요하지 않아 CIDR 값을 높게 /25를 주었다.

그렇다면 실제로 구성을 해보자.

위에 그림처럼 AZ 2개의 Public Subnet, Private subnet을 구성해볼 것이다.

1. Subnet 생성 메뉴 진입 화면

VPC 내부에 서브넷을 생성하기 위해 좌측 메뉴에서 Subnet을 선택한 후 Create subnet 버튼을 클릭한다.



2. VPC 선택화면

서브넷을 생성할 대상 VPC로 앞서 생성한 study VPC를 선택한다.

서브넷 이름과 가용 영역(AZ)을 지정하여 각 서브넷이 서로 다른 AZ에 생성되도록 설정한다. Public Subnet은 다수의 EC2 인스턴스를 배치하기 위해 상대적으로 넓은 IP 범위를 제공하는 /24 CIDR 블록으로 지정하고, Private은 소수의 내부 리소스를 배치하기 위해 /25 CIDR 블록을 사용하여 IP 범위를 설정한다.

Create subnet Info

VPC

VPC ID
Create subnets in this VPC.

Select a VPC

Select a VPC first to create new subnets.

[Add new subnet](#)

Subnet settings
Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 4

Subnet name
Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

IPv4 subnet CIDR block
 256 IPs

Tags - optional
Key Value - optional [Remove](#)

Subnet 2 of 4

Subnet name
Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

IPv4 subnet CIDR block
 128 IPs

Tags - optional
Key Value - optional [Remove](#)

Subnet 3 of 4

Subnet name
Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

IPv4 subnet CIDR block
 256 IPs

Tags - optional
Key Value - optional [Remove](#)

Subnet 4 of 4

Subnet name
Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Availability Zone Info
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 VPC CIDR block Info
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

IPv4 subnet CIDR block
 128 IPs

Tags - optional
Key Value - optional [Remove](#)

3. 서브넷 생성 완료 화면

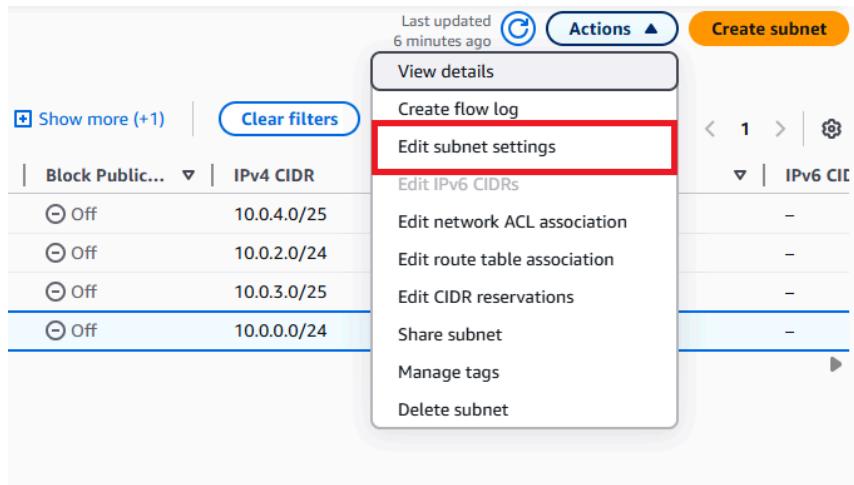
설정한 값에 따라 Public Subnet과 Private Subnet이 정상적으로 생성된 것을 서브넷 목록에서 확인할 수 있다.

Available IPv4 addresses	
123	
251	
123	
251	

* 서브넷은 한 번 만들면 옵션 설정이 불가능하므로, 설정을 바꿔야 한다면 지우고 새로 만들어야 한다.

4. Public Subnet 선택 화면

외부 인터넷과 통신이 필요한 Public Subnet을 선택하여 세부 설정을 변경한다.



이를 설정하지 않으면 인터넷과 통신을 할 수 없다.

5. Public IPv4 자동 할당 설정 화면

Edit subnet → settings에 들어가 Auto-assign IP settings에서 Enable auto assign public IPv4 address에 체크를 해준다.



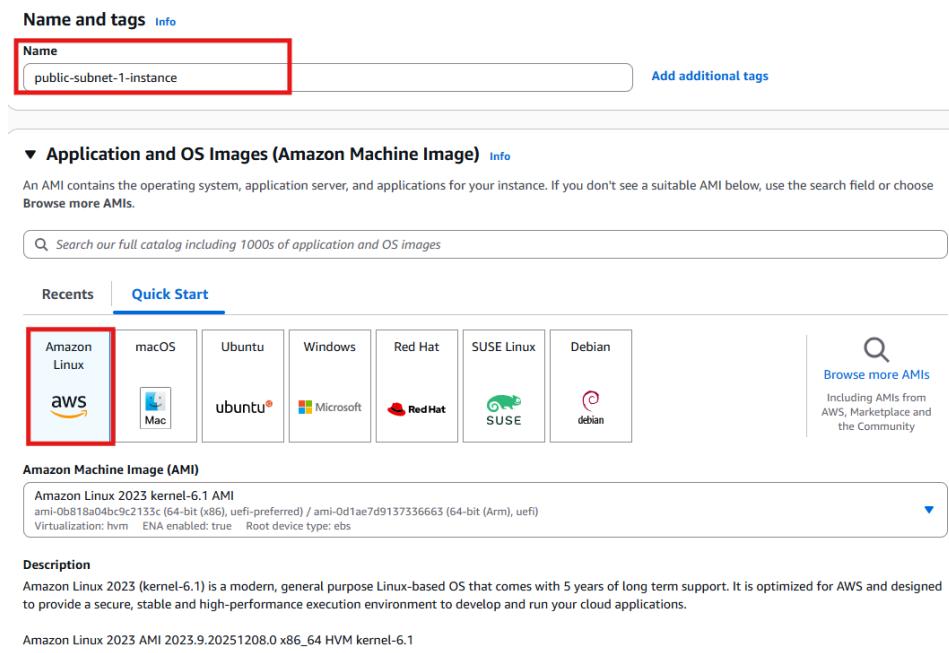
6. Subnet에 EC2 인스턴스 생성

EC2 인스턴스를 생성하고 설정을 진행해준다.

6-1. EC2 - Public Subnet

네트워크와 서브넷에 위에서 생성한 VPC와 퍼블릭 서브넷을 선택하고, 퍼블릭 IP 자동할당은 활성화를 선택한다.

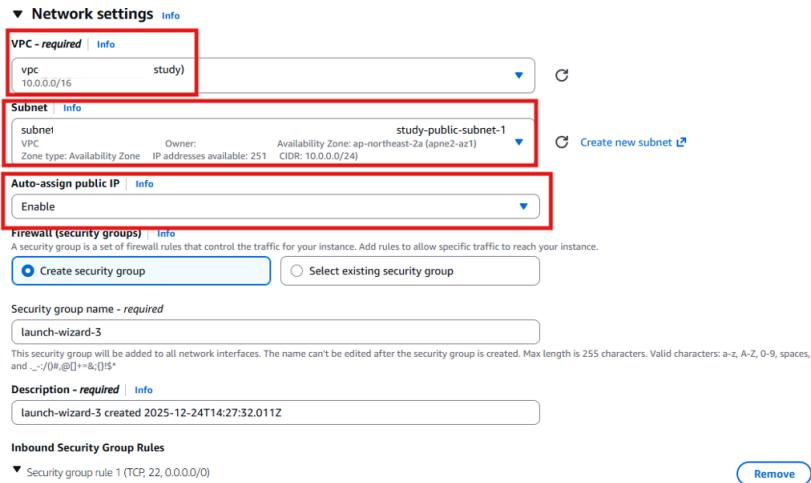
이름을 입력하고, OS Image는 Amazon Linux로 설정한다.



Instance type은 t3.micro로 설정한다.



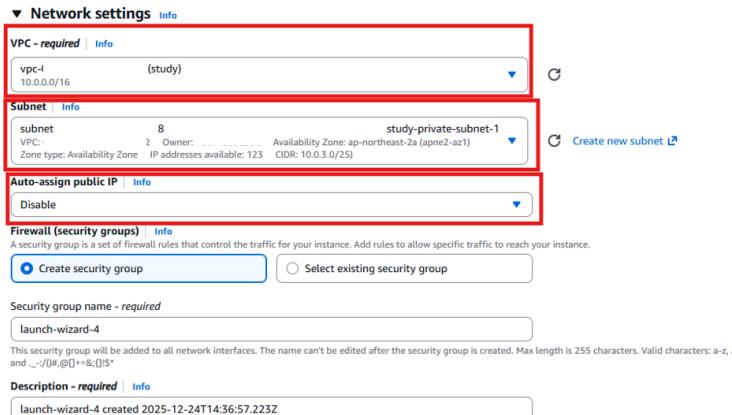
이후, 네트워크 설정에서는 study VPC를 선택하고, Subnet은 study-public-1을 지정하며 Auto-assign public IP은 Enable인지 확인한다.



6-2 EC2 - Private Subnet

1. Private Subnet도 위에 Public과 같이 생성해주면 된다.

Private Subnet에 생성되는 인스턴스는 외부 노출을 방지하기 위해 퍼블릭 IP 자동 할당을 비활성화하여 설정한다.



2. EC2 생성 결과 화면

Public Subnet과 Private Subnet에 각각 EC2 인스턴스가 생성되어 네트워크 분리가 정상적으로 이루어진 것을 확인할 수 있다.



4. 라우팅 테이블

Subnet은 각각 서로 다른 네트워크 영역을 가지고 있다. 한 subnet이 다른 Subnet으로 가기 위해서는 'Routing'이 필요하게 된다.

라우팅 테이블은 VPC 안에서 발생한 네트워크 요청을 처리하기 위해 **어디로 트래픽을 전송해야 하는지** 알려주는 표지판 역할을 수행한다.

각 서브넷들은 네트워크 트래픽 전달 규칙에 해당하는 라우팅 테이블을 가지고 있으며, 요청이 발생하면 이 라우트 테이블을 사용해서 목적지를 찾을 수 있게 된다.



* IP 0.0.0.0/0

0.0.0.0 - 보통 서브넷 마스크 0.0.0.0과 같이 사용되며 모든 IP를 의미한다.

이 문구를 보면, 0.0.0.0은 local machine의 모든 IPv4 address를 의미하기 때문에 0.0.0.0로 접근하면 로컬 호스트의 모든 IPv4로 되어있는 호스트에 접근이 가능하다는 것을 뜻한다.

호스트에 정확한 address가 할당되어 있지 않다면, 각각의 host는 그 address를 자신이라고 주장하게 되고 이에 따라 웹 서비스에서 0.0.0.0을 지정하면 자신의 IP를 그 address로 지정하게 되어 local로 접근이 되는 원리이다.

0.0.0.0/0 정리

- 모든 IPv4 주소 지정 방법
- 라우팅 테이블에서 사용되는 경우 기본 게이트웨이를 식별하는데 0.0.0.0에 대한 경로가 기본 경로
- 컴퓨터, 모뎀 및 네트워크 카드에서 수신 대기 중임을 의미
- 네트워크에 액세스할 수 있는 곳이면 어디에서나 접근 가능을 의미

::/0 - 0.0.0.0의 IPv6 버전을 의미함

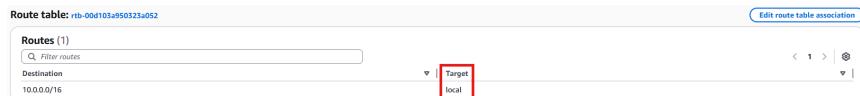
127.0.0.1 - 로컬 호스트 즉 자기 스스로를 가리키는 IP

- 동일한 기계에서만 액세스할 수 있음
- 포트가 인터넷이나 네트워크가 아닌 PC 자체의 연결만을 수신

255.255.255.255 - 브로드캐스트 용도로 사용하기 위해 예약된 IP 주소

VPC 내부(모든 Subnet)에 대해서는 디폴트로 Routing이 자동으로 생성되어 있으므로 별도의 설정 없이 한 Subnet에서 다른 Subnet으로 통신이 가능하다는 말이다.

VPC 대역인 10.0.0.0/16에 대해 Local Routing이 지정되어 있음을 알 수 있다.



이렇게 각 서브넷에 라우팅 테이블이 있음으로서, VPC 내 Subnet에 할당된 Resource라면 어느 Subnet이든 다른 Subnet의 Resource와 자유롭게 통신할 수 있게 된다.

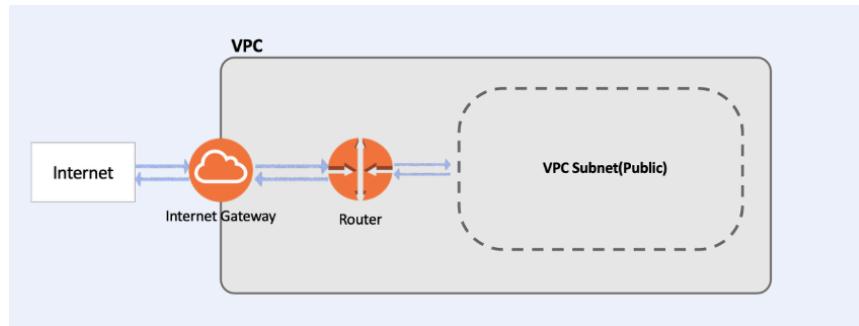
또한, 라우팅 테이블은 내부에서만 통신하는 Local Routing 뿐만 아니라 외부 인터넷망으로 나갈 수 있는 Routing을 가질 수 있다. 이를 인터넷 게이트웨이 (Internet Gateway)라고 한다.

5. 인터넷 게이트웨이 (Internet Gateway)

VPC는 기본적으로 격리된 네트워크 환경이다. 따라서 VPC에서 생성된 리소스들은 기본적으로 인터넷을 사용할 수 없다.

이러한 환경에서, 인터넷 게이트웨이는 VPC의 인스턴스와 인터넷간에 통신을 할 수 있게 해준다.

인터넷 게이트웨이는 VPC의 리소스와 인터넷 간의 통신을 활성화하기 위해 VPC에 연결하는 관문으로 VPC 리소스가 인터넷으로 나가기 위한 통로이다.



앞서서 위에 서브넷을 private 2개, public 2개를 만들어줬다.

하지만 실제로는 private 서브넷 4개를 만든 것이다.

서브넷을 진짜 퍼블릭으로 바꾸기 위해선 인터넷 게이트웨이를 라우팅 테이블에 설정하고 등록해줘야 하기 때문이다.

이렇게 서브넷이 인터넷 게이트웨이로 향하는 라우팅이 있는 경우 퍼블릭 서브넷이라고 부르게 되는 것이다. 반대로 어떤 서브넷이 인터넷 연결을 할 필요가 없다면 이 서브넷은 프라이빗 서브넷이라고 한다.

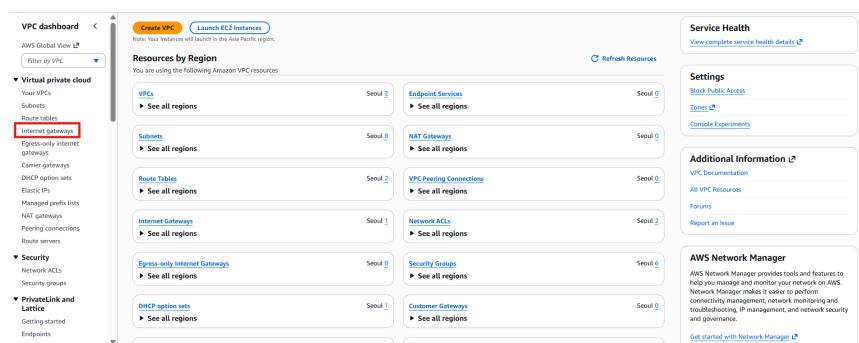
VPC에서 Resource (EC2 등)가 외부 인터넷과 통신하고자 할 경우 갖추어야 할 조건이 다음과 같다.

- Internet 통신하고자 하는 Resource가 공인 IP를 보유할 것
- Resource가 소속된 Subnet의 Routing Table에 '0.0.0.0/0' 목적지로 갖는 Routing 'Internet Gateway'이 있을 것
- Network ACL과 Security Group 규칙에서 허용할 것

5-1 인터넷 게이트웨이 실습

1. Internet Gateway 메뉴 진입 화면

VPC와 외부 인터넷 간 통신을 설정하기 위해 VPC 콘솔에서 Internet Gateways 메뉴로 이동한다.



2. Internet Gateway 생성 화면

새로운 Internet Gateway를 생성하기 위해 Create Internet Gateway 버튼을 선택한다.



3. Internet Gateway 이름 설정 화면

생성할 Internet Gateway의 이름을 지정한 후 게이트웨이를 생성한다.

Create internet gateway Info

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

Internet gateway settings

Name tag
Creates a tag with a key of 'Name' and a value that you specify.
study-igw

Tags - optional
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="study-igw"/> X Remove

Add new tag
You can add 49 more tags.

Cancel Create internet gateway

Internet Gateway가 정상적으로 생성되었으며, 아직 VPC에는 연결되지 않은 상태임을 알 수 있다.



4. VPC에 인터넷 게이트웨이 연결하기

생성한 Internet Gateway를 선택한 후 Actions 메뉴에서 Attach To VPC를 클릭한다.



5. VPC 연결 설정 화면

Internet Gateway를 연결한 대상 VPC로 앞서 생성한 study VPC를 생성한다.

VPC

Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.

Available VPCs
Attach the internet gateway to this VPC.

AWS Command Line Interface command

Cancel Attach internet gateway

6. Internet Gateway 연결 완료 화면

Internet Gateway는 하나의 VPC에만 연결 가능하며, Attached 상태를 통해 정상적으로 연결되었음을 확인할 수 있다.



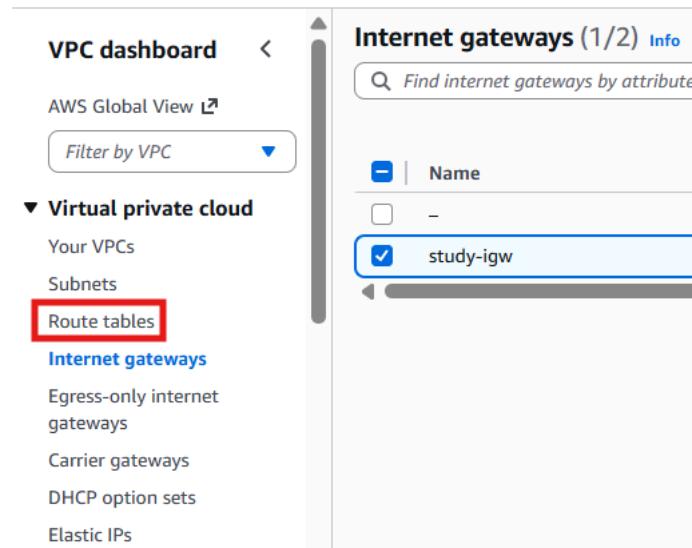
The screenshot shows the AWS VPC Internet Gateways list. It has a header with columns: Name, Internet gateway ID, State, VPC ID, and Owner. A search bar at the top says 'Find internet gateways by attribute or tag'. There is one item listed: 'study-igw' with 'Attached' status. The 'Actions' button is visible at the top right.

설정이 잘 완료됨을 알 수 있다.

라우팅 테이블 만들기

7. Route Tables 메뉴 화면

서브넷의 트래픽 흐름을 제어하기 위해 Route tables 메뉴로 이동한다.

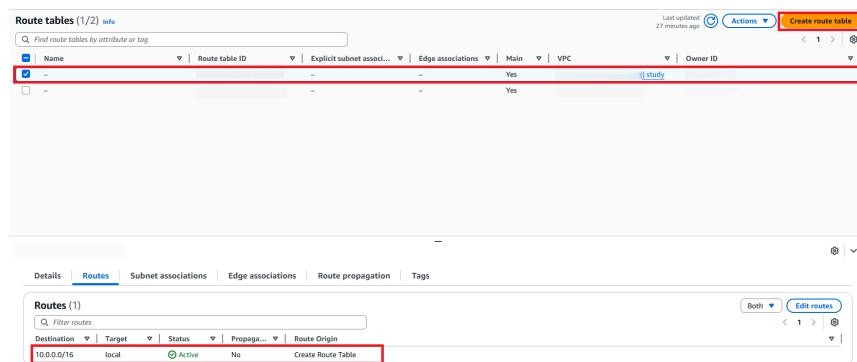


The screenshot shows the AWS VPC dashboard. On the left, there's a sidebar with 'Virtual private cloud' sections: Your VPCs, Subnets, Route tables (which is highlighted with a red box), Internet gateways, Egress-only internet gateways, Carrier gateways, DHCP option sets, and Elastic IPs. On the right, there's a list of 'Internet gateways (1/2)'. The 'study-igw' gateway is selected, indicated by a checked checkbox. The 'Actions' button is at the top right of the gateway list.

8. Public 라우팅 테이블 생성 화면

외부 인터넷 통신을 위한 별도의 Public 라우팅 테이블을 생성한다.

기본으로 생성된 라우팅 테이블은 내부 통신 용도로 사용한다.



The screenshot shows the AWS Route tables list. It has a header with columns: Name, Route table ID, Explicit subnet assoc..., Edge associations, Main, VPC, Owner ID. A search bar at the top says 'Find route tables by attribute or tag'. There is one item listed: a route table with a single route entry: '10.0.0.0/16' to 'local' with 'Active' status. The 'Actions' button is visible at the top right.

9. 라우팅 테이블 설정

생성된 Public 라우팅 테이블을 선택하여 세부 라우팅 규칙을 설정한다.

Create route table Info

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

Route table settings

Name - optional
Create a tag with a key of 'Name' and a value that you specify.

VPC
The VPC to use for this route table.

Tags
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key **Value - optional** You can add 49 more tags.

Route tables (1/3) Info

Last updated less than a minute ago

Name	Route table ID	Explicit subnet assoc...	Edge associations	Main	VPC	Owner ID
<input checked="" type="checkbox"/> study-private-table	-	-	-	<input checked="" type="checkbox"/> Yes	study	
<input type="checkbox"/> -	-	-	-	<input type="checkbox"/> Yes		
<input checked="" type="checkbox"/> study-public-table	-	-	-	<input type="checkbox"/> No	study	

이렇게 설정하면, 기본(예)으로 생성된 서브넷들은 기본적으로 private-table을 사용해서 인터넷에 접근 못하는 보안성을 확보하고, 외부와 연결이 필요한 서브넷들만 따로 public-table을 추가로 만들어서 넣어주면 된다.

퍼블릭 라우팅 테이블에 서브넷 등록하기

10. 서브넷 연결 설정 화면

Route tables (1/3) Info

Last updated 9 minutes ago

Name	Route table ID	Explicit subnet assoc...	Edge associations	Main	VPC	Owner ID
<input checked="" type="checkbox"/> study-private-table	-	-	-	<input checked="" type="checkbox"/> Yes	study	
<input type="checkbox"/> -	-	-	-	<input type="checkbox"/> Yes		
<input checked="" type="checkbox"/> study-public-table	-	-	-	<input type="checkbox"/> No	study	

study-public-table

Explicit subnet associations (0)

No subnet associations
You do not have any subnet associations.

외부 인터넷과 통신해야 하는 Public Subnet을 Public 라우팅 테이블에 명시적으로 연결한다.

Edit subnet associations
Change which subnets are associated with this route table.

Available subnets (2/4)

Filter subnet associations

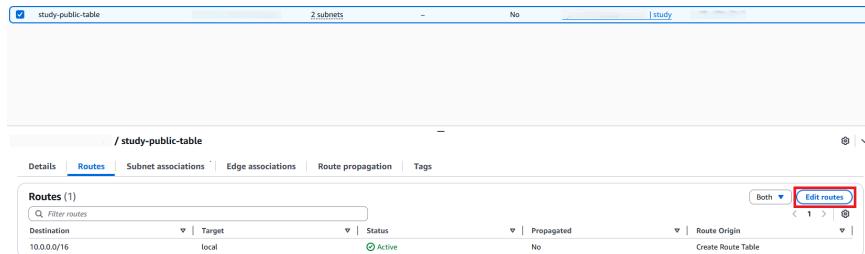
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input type="checkbox"/> study-private-subnet-2	10.0.4.0/25	-	-	
<input checked="" type="checkbox"/> study-public-subnet-2	10.0.2.0/24	-	-	
<input type="checkbox"/> study-private-subnet-1	10.0.3.0/25	-	-	
<input checked="" type="checkbox"/> study-public-subnet-1	10.0.0.0/24	-	-	

Selected subnets

라우팅 테이블에 인터넷 게이트웨이 연결하기

11. 라우팅 테이블 선택 화면

Internet Gateway와 연결할 Public 라우팅 테이블을 선택하고, 선택한 라우팅 테이블에서 Edit routes 버튼을 클릭하여 라우팅 규칙을 수정한다.



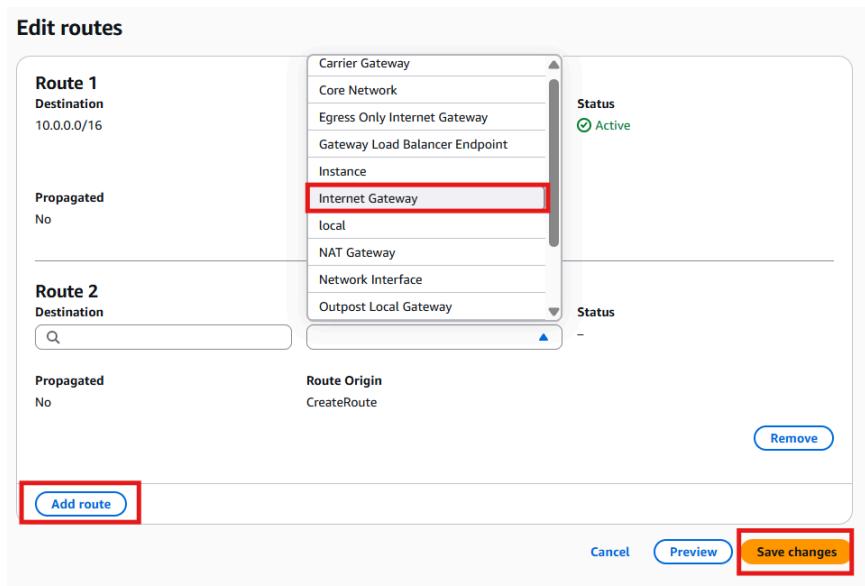
The screenshot shows the AWS CloudFormation console with a route table named 'study-public-table'. The 'Routes' tab is selected. A single route is listed with the following details:

Destination	Target	Status	Propagated	Route Origin
10.0.0.0/16	local	Active	No	Create Route Table

The 'Edit routes' button in the top right corner of the table is highlighted with a red box.

12. Internet Gateway 라우트 추가 화면

목적지 0.0.0.0/0에 대해 타깃을 Internet Gateway로 설정하여 외부 인터넷으로 향하는 트래픽이 IGW를 통해 전달되도록 구성한다.



The screenshot shows the 'Edit routes' dialog box. It contains two routes:

- Route 1**: Destination 10.0.0.0/16, Target Internet Gateway, Propagated No, Status Active.
- Route 2**: Destination 0.0.0.0/0, Target local, Propagated No, Status -.

At the bottom, there are 'Add route' and 'Save changes' buttons. The 'Add route' button is highlighted with a red box, and the 'Save changes' button is highlighted with a yellow box.

Route 1

Destination: 10.0.0.0/16

Target: local

Status: Active

Route 2

Destination: 0.0.0.0/0

Target: Internet Gateway (igw-045)

Status: In Progress

Propagated: No

Route Origin: CreateRouteTable

Propagated: No

Route Origin: CreateRoute

Save changes

13. 라우팅 테이블 설정 완료 화면

라우팅 규칙이 정상적으로 추가되어 Public Subnet에 속한 리소스가 Internet Gateway를 통해 외부 인터넷과 통신할 수 있게 된다.

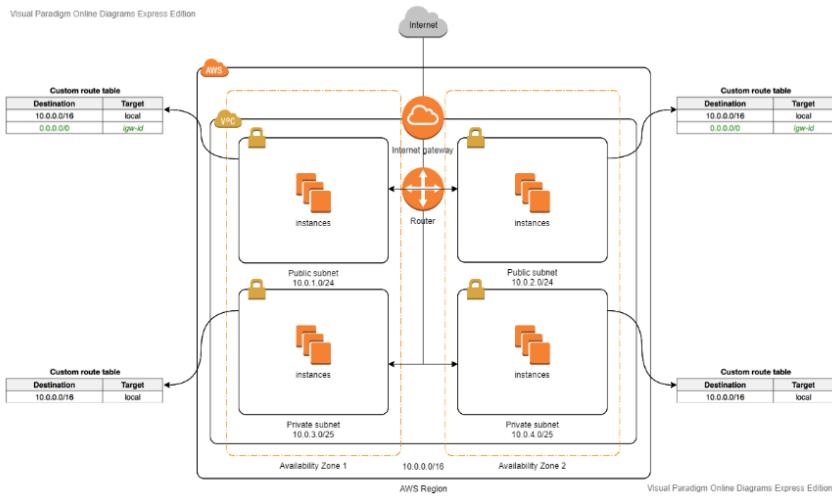
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
study-public-subnet-2	subnet	10.0.2.0/24	-
study-public-subnet-1	subnet	10.0.0.0/24	-

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
study-private-subnet-2	subnet	10.0.4.0/25	-
study-private-subnet-1	subnet	10.0.3.0/25	-

라우팅 테이블 연결 설정을 맞췄으면 이제야 정말로 Public Subnet과 Private Subnet을 완성시킨 것이다.

사진에서 보이듯이 서브넷을 명시적으로 인터넷 게이트웨이를 연결한 라우팅 테이블에 등록함으로서 외부에 연결이 되어, 인터넷 게이트웨이로 향하는 라우팅이 있는 경우 퍼블릭 서브넷으로 라우팅 테이블에 적혀있는대로 가는 곳이다.

지금까지의 구축 과정을 그림으로 표현하면 다음과 같다.



위 그림에서 Public Subnet 2개는 Local Routing과 함께 Internet Gateway에 연결되어 있으므로 내부와 Internet 통신이 가능하며, Private Subnet은 Local Routing만을 갖고 있으므로 내부 통신만 가능하다.

- 공인 인터넷과 통신 가능한 Subnet을 Public Subnet
- 공인 인터넷이 차단된 사설 IP만 할당된 Subnet을 Private Subnet
- Private Subnet은 IGW로 연결되어 있음
- 인터넷 게이트웨이는 VPC 내부가 아닌 VPC의 외부 통신에 관여함

Reference

1. <https://www.beyondtrust.com/blog/entry/aws-root-vs-iam-user-what-to-know-when-to-use-them>
2. <https://aws.amazon.com/ko/iam/details/manage-users/>
3. https://goddaehee.tistory.com/327#google_vignette
4. https://jhlee-developer.tistory.com/entry/AWS-AWS-CLI-%EC%84%A4%EC%B9%98-%EB%B0%8F-V2-%EC%97%85%EB%8D%B0%EC%9D%B4%ED%8A%B8#google_vignette
5. https://docs.aws.amazon.com/ko_kr/IAM/latest/UserGuide/introduction.html
6. https://docs.aws.amazon.com/ko_kr/IAM/latest/UserGuide/intro-iam-features.html
7. https://docs.aws.amazon.com/ko_kr/IAM/latest/UserGuide/when-to-use-iam.html
8. https://docs.aws.amazon.com/ko_kr/IAM/latest/UserGuide/introduction_identity-management.html
9. https://docs.aws.amazon.com/ko_kr/IAM/latest/UserGuide/introduction_access-management.html

10. <https://hyunki99.tistory.com/94>
11. https://docs.aws.amazon.com/ko_kr/IAM/latest/UserGuide/id_credentials_temp_use-resources.html
12. https://docs.aws.amazon.com/ko_kr/IAM/latest/UserGuide/id_credentials_access-keys.html
13. <https://velog.io/@chas369/aws-%EC%97%91%EC%84%B8%EC%8A%A4%ED%82%A4-%EC%8B%9C%ED%81%AC%EB%A6%BF%ED%82%A4-%EB%B0%9C%EA%B8%89%EB%B0%9B%EA%B8%80>
14. <https://inpa.tistory.com/entry/AWS-%F0%9F%93%9A-%EC%9E%A5%EA%B8%B0-%EC%9E%90%EA%B2%A9-%EC%A6%9D%EB%AA%85-%EC%9E%84%EC%8B%9C-%EC%9E%90%EA%B2%A9-%EC%A6%9D%EB%AA%85-Access-Key-Secret-Access-Key>
15. https://docs.aws.amazon.com/ko_kr/IAM/latest/UserGuide/access_policies_managed-vs-inline.html
16. https://inpa.tistory.com/entry/AWS-%F0%9F%93%9A-S3-%EB%B2%84%ED%82%8B7-%EC%83%9D%EC%84%B1-%EC%82%AC%EC%9A%A9%EB%B2%95-%EC%8B%A4%EC%A0%84-%EA%B5%AC%EC%B6%95#s3%EC%9D%84_%EC%82%AC%EC%9A%A9%ED%95%98%EB%8A%94_%EC%91%EA%80%9C%EB%85%90%EC%9E%A1%EA%B8%B0-71eef95a7098
17. <https://medium.com/harrythegreat/aws-%EA%B0%80%EC%9E%A5%EC%89%BD%EA%B2%8C-vpc-%EA%B0%9C%EB%85%90%EC%9E%A1%EA%B8%B0-71eef95a7098>
18. <https://www.cloudflare.com/ko-kr/learning/access-management/what-is-a-vpn/>
19. https://docs.aws.amazon.com/ko_kr/vpc/latest/userguide/how-it-works.html
20. <https://aws.amazon.com/blogs/apn/amazon-vpc-for-on-premises-network-engineers-part-one/>
21. <https://aws-hyoh.tistory.com/entry/VPC-%EC%89%BD%EA%B2%8C-%EC%9D%B4%ED%95%B4%ED%95%98%EA%B8%B0-1>
22. <https://inpa.tistory.com/entry/AWS-%F0%9F%93%9A-VPC-%EC%82%AC%EC%9A%A9-%EC%84%9C%EB%B8%8C%EB%84%B7-%EC%9D%B8%ED%84%80%EB%84%B7-%EA%B2%8C%EC%9D%B4%ED%8A%B8%EC%9B%A8%EC%9D%B4-NAT-%EB%B3%B4%EC%95%88%EA%B7%B8%EB%A3%B9-NACL-Bastion-Host>
23. https://docs.aws.amazon.com/ko_kr/vpc/latest/userguide/VPC_Internet_Gateway.html
24. https://docs.aws.amazon.com/ko_kr/vpc/latest/userguide/VPC_Internet_Gateway.html
25. <https://www.youtube.com/@AWSClassroom>
26. https://kimjingo.tistory.com/172#google_vignette