

**클라우드 서비스(AWS)**



# Cloud Service

## 클라우드 서비스 유형

추가로 소프트웨어를 다운로드하지 않고도 인터넷을 통해 사용자가 액세스하는 모든 **인프라**, 플랫폼, 소프트웨어 또는 기술은 **클라우드 컴퓨팅** 서비스라고 볼 수 있으며 다음의 서비스형(as-a-Service) 솔루션이 포함됩니다.

서비스로서의 인프라(Infrastructure-as-a-Service, IaaS)는 사용자에게 컴퓨팅, 네트워킹 및 **스토리지** 리소스를 제공합니다.

서비스로서의 플랫폼(Platforms-as-a-Service, PaaS)은 애플리케이션을 실행할 수 있는 플랫폼과 플랫폼 실행에 필요한 IT 인프라를 제공합니다.

서비스로서의 소프트웨어(Software-as-a-Service, SaaS)는 **클라우드 애플리케이션**과 해당 애플리케이션을 실행하는 플랫폼, 플랫폼의 기반 인프라를 제공합니다.

서비스로서의 기능(Function-as-a-Service, FaaS)은 **이벤트 기반** 실행 모델로서, 개발자가 인프라를 유지관리하지 않고도 애플리케이션 패키지를 기능으로 구축, 실행 및 **관리**할 수 있도록 지원합니다.

# AWS 사이트 가입

URL : <https://aws.amazon.com/ko/>

The screenshot shows the AWS Management Console interface in Korean. At the top, there's a navigation bar with the AWS logo, a '서비스' (Services) menu, a search bar with the text '검색' (Search), and a keyboard shortcut '[알트+S]' (Alt+S). On the right side of the navigation bar, there are icons for help, notifications, settings, and a user profile 'kevin @ amicokb'. Below the navigation bar, there's a row of service icons: EC2, IAM, Route 53, and VPC. The main content area is titled '콘솔 홈' (Console Home) with a sub-link '정보' (Info). On the right side of this header, there are two buttons: '기본 레이아웃으로 재설정' (Reset to default layout) and '+ 위젯 추가' (Add widget). The main content is divided into two columns. The left column is titled '최근에 방문한 서비스' (Recently visited services) and lists various services with their icons: EC2, IAM, Route 53, 결제 및 비용 관리 (Billing and cost management), CloudWatch, Certificate Manager, S3, RDS, AWS Organizations, IAM Identity Center, CloudFront, and AWS Cloud Map. The right column is titled '애플리케이션 (0)' (Applications (0)) and shows the region '리전: Asia Pacific (Seoul)'. It includes a dropdown menu for the region (currently 'ap-northeast-2(현재 리전)'), a search bar with the text '애플리케이션 찾기' (Find application), and a table header with columns: 이름 (Name), 설명 (Description), 리전 (Region), and 최초 계정 (First account). Below the header, it says '애플리케이션 없음' (No applications) and '애플리케이션을 만들어 시작해 보세요.' (Create an application to get started.). There is a button '애플리케이션 생성' (Create application) at the bottom of this section. At the bottom of the page, there's a footer with 'CloudShell 의견' (CloudShell feedback) on the left and copyright information '© 2024, Amazon Web Services, Inc. 또는 계열사.' (© 2024, Amazon Web Services, Inc. or its affiliates.) on the right, followed by links for '개인 정보 보호' (Privacy), '약관' (Terms), and '쿠키 기본 설정' (Cookie settings).

aws | 서비스 | 검색 | [알트+S]

EC2 | IAM | Route 53 | VPC

콘솔 홈 정보

기본 레이아웃으로 재설정 | + 위젯 추가

최근에 방문한 서비스 정보

- EC2
- IAM
- Route 53
- 결제 및 비용 관리
- CloudWatch
- Certificate Manager
- S3
- RDS
- AWS Organizations
- IAM Identity Center
- CloudFront
- AWS Cloud Map

애플리케이션 (0) 정보

리전: Asia Pacific (Seoul)

ap-northeast-2(현재 리전) | 애플리케이션 찾기

< 1 >

이름	설명	리전	최초 계정
애플리케이션 없음 애플리케이션을 만들어 시작해 보세요.			

애플리케이션 생성

CloudShell 의견 | © 2024, Amazon Web Services, Inc. 또는 계열사. | 개인 정보 보호 | 약관 | 쿠키 기본 설정

# AWS 클라우드 소개

AWS 계정을 처음 생설할 때 본인 명의의 신용카드 필요

AWS 계정을 처음 생성하면 루트 유저와 기본 리소스(기본 VPC) 등이 생성됨

AWS 아이디가 부여됨(숫자)

- 추후 AWS 계정에 별명 지정 가능(문자)

루트 유저

- 생성한 계정의 모든 권한을 자동으로 가지고 있음
- 생성시 만든 이메일 주소로 로그인
- 복구가 매우 힘드므로 사용을 자제하고 관리용(계정 설정 변경, 빌링)으로만 이용
- AWS API 호출 불가

IAM User : AWS의 모든 권한을 관리하는 서비스. AWS의 관리를 제외한 모든 작업은 관리용 IAM User를 만들어 사용

# AWS 클라우드 소개

AWS Identity and Access Management(IAM)를 사용하면 AWS 서비스와 리소스에 대한 액세스를 안전하게 관리할 수 있으며 AWS 사용자 및 그룹을 만들고 관리하며 AWS 리소스에 대한 액세스를 허용 및 관리할 수 있다.

IAM 구성 : 사용자, 그룹, 정책(권한 부여), 역할(서비스 부여)

사용자 : 루트 사용자(모든 권한), IAM 사용자(IAM을 통해 생성해서 사용)

EC2 생명 주기

- 중지중에는 인스턴스 요금 미 청구. 단 EBS 요금, 다른 구성요소(Elastic IP등)은 청구
- 중지 후 재시작시 퍼블릭 IP 변경
- EBS를 사용하는 인스턴스만 중지 가능

# AWS 서비스

## 1. 컴퓨팅 (Compute)

### EC2 (Elastic Compute Cloud)

- 가상 서버 인스턴스를 제공하며 원하는 운영체제에서 애플리케이션을 실행할 수 있습니다.
- 수요에 따라 확장 및 축소가 가능하며 온디맨드 또는 예약 인스턴스로 사용할 수 있습니다.

### Lambda

- 서버리스 컴퓨팅 서비스로 서버를 직접 관리할 필요 없이 코드를 실행할 수 있습니다.
- 이벤트 기반으로 작동하며 사용한 만큼만 비용이 청구됩니다.

### ECS / EKS / Fargate

- ECS는 Docker 컨테이너를 관리하는 서비스입니다.
- EKS는 Kubernetes 클러스터를 쉽게 설정하고 운영할 수 있도록 지원합니다.
- Fargate는 서버리스 방식으로 컨테이너를 실행합니다.

# AWS 서비스

## 2. 스토리지 (Storage)

### S3 (Simple Storage Service)

- 확장 가능한 객체 스토리지로 정적 파일, 백업, 영상 등을 저장합니다.
- 버킷 단위로 데이터를 관리하며 높은 내구성과 가용성을 제공합니다.

### EBS (Elastic Block Store)

- EC2 인스턴스에 고성능 블록 스토리지를 제공합니다.
- 데이터베이스와 같은 고속 입출력이 필요한 애플리케이션에 적합합니다.

### Glacier

- 저렴한 비용으로 장기 데이터 보관에 적합한 아카이빙 스토리지입니다.

# AWS 서비스

## 3. 데이터베이스 (Database)

### **RDS (Relational Database Service)**

- 관리형 관계형 데이터베이스 서비스로 MySQL, PostgreSQL, SQL Server 등 여러 엔진을 지원합니다.
- 데이터베이스 관리의 복잡성을 줄여줍니다.

### **DynamoDB**

- NoSQL 데이터베이스로 빠른 읽기와 쓰기 성능을 제공합니다.
- 확장성과 고가용성이 뛰어나며 서버리스 환경에서 자주 사용됩니다.

### **Aurora**

- RDS와 호환되지만 더 높은 성능을 제공하는 클라우드 네이티브 데이터베이스입니다.



# AWS 서비스

## 4. 네트워킹 (Networking)

### VPC (Virtual Private Cloud)

- AWS 내부에 격리된 네트워크 환경을 구성합니다.
- 서브넷, 라우팅 테이블, 게이트웨이 등 네트워크를 유연하게 설정할 수 있습니다.

### Route 53

- 도메인 이름을 관리하는 DNS 서비스입니다.
- 글로벌 트래픽 분산과 헬스체크 기능도 제공합니다.

### CloudFront

- 콘텐츠 배포 네트워크(CDN)로 전 세계에 데이터를 빠르게 전송합니다.
- S3와 연동해 웹사이트나 영상 콘텐츠를 제공합니다.

# AWS 서비스

## 5. 보안 및 관리 (Security & Management)

### **IAM (Identity and Access Management)**

- AWS 리소스에 대한 권한을 관리합니다.
- 사용자와 역할을 정의해 세밀한 접근 제어를 할 수 있습니다.

### **CloudWatch**

- AWS 리소스와 애플리케이션의 성능과 상태를 모니터링합니다.
- 경고 설정 및 지표 수집이 가능합니다.

### **CloudTrail**

- 계정 내 API 호출을 기록해 보안 감사와 문제 해결에 사용합니다.

# AWS에서 VPC 전체 구성 및 설정 프로세스

## 1단계: VPC 생성

VPC 생성: AWS Management Console에서 "VPC" 서비스로 이동합니다. "Your VPCs"에서 "Create VPC"를 클릭합니다. VPC 이름과 IPv4 CIDR 블록 (예: 10.0.0.0/16)을 입력합니다. "Create"를 클릭하여 VPC를 생성합니다.

## 2단계: 서브넷 생성

서브넷 생성: VPC 대시보드에서 "Subnets"를 선택합니다. "Create subnet"을 클릭합니다. 서브넷 이름, VPC 선택, 가용영역 선택 (예: us-east-1a), 서브넷의 IPv4 CIDR 블록 (예: 10.0.1.0/24)을 입력합니다. "Create"를 클릭하여 서브넷을 생성합니다. 필요 시 다른 가용영역에 대해서도 추가 서브넷을 생성합니다 (예: us-east-1b, CIDR: 10.0.2.0/24).

## 3단계: 인터넷 게이트웨이 및 라우팅 테이블 설정

인터넷 게이트웨이 생성 및 연결: "Internet Gateways"를 선택합니다. "Create internet gateway"를 클릭하고 이름을 입력한 후 "Create"를 클릭합니다. 생성된 인터넷 게이트웨이를 선택하고 "Attach to VPC"를 클릭하여 VPC에 연결합니다.

## AWS에서 VPC 전체 구성 및 설정 프로세스

라우팅 테이블 생성 및 설정:"Route Tables"를 선택합니다."Create route table"을 클릭하고 이름을 입력한 후 VPC를 선택합니다."Create"를 클릭하여 라우팅 테이블을 생성합니다.생성된 라우팅 테이블을 선택하고 "Routes" 탭에서 "Edit routes"를 클릭합니다.0.0.0.0/0 (모든 트래픽)을 인터넷 게이트웨이로 라우팅하도록 경로를 추가합니다."Subnet associations" 탭에서 "Edit subnet associations"를 클릭하고 서브넷을 선택하여 연결합니다.

4단계: EC2 인스턴스 생성 및 보안 그룹 설정보안 그룹 생성:"Security Groups"를 선택합니다."Create security group"을 클릭하고 이름과 설명을 입력합니다.VPC를 선택합니다."Inbound rules"와 "Outbound rules"를 설정합니다.Inbound rules: SSH (포트 22), HTTP (포트 80), HTTPS (포트 443) 등을 허용합니다.Outbound rules: 기본적으로 모든 트래픽을 허용합니다."Create"를 클릭하여 보안 그룹을 생성합니다. EC2 인스턴스 생성:AWS Management Console에서 "EC2" 서비스로 이동합니다."Launch Instance"를 클릭합니다.인스턴스의 이름을 입력하고, AMI (Amazon Machine Image), 인스턴스 유형 (예: t2.micro)을 선택합니다."Network settings"에서 VPC와 서브넷을 선택하고 보안 그룹을 선택합니다."Launch"를 클릭하여 인스턴스를 생성합니다.

## AWS에서 VPC 전체 구성 및 설정 프로세스

5단계: ALB (Application Load Balancer) 생성  
ALB 생성: "Load Balancers"를 선택합니다. "Create Load Balancer"를 클릭하고 "Application Load Balancer"를 선택합니다. ALB의 이름을 입력하고, 인터넷 연결 여부 (Internet-facing), 그리고 리스너 및 가용영역을 설정합니다. 리스너: 기본적으로 HTTP (포트 80)를 선택할 수 있습니다. 가용영역: 사용할 서브넷을 선택합니다. 보안 그룹을 선택합니다. 타겟 그룹을 설정합니다 (EC2 인스턴스를 타겟으로 추가). 타겟 그룹 생성 시, 이름을 입력하고 VPC를 선택합니다. "Targets" 탭에서 "Edit"를 클릭하고 EC2 인스턴스를 추가합니다. ALB를 생성합니다.

# AWS에서 VPC 전체 구성 및 설정 프로세스

## 6단계: Route 53 설정

Route 53 도메인 설정: AWS Management Console에서 "Route 53" 서비스를 선택합니다. "Hosted zones"에서 "Create hosted zone"을 클릭합니다. 도메인 이름을 입력하고, "Public hosted zone"을 선택합니다. "Create hosted zone"을 클릭합니다. Route 53 레코드 설정: 생성된 호스티드 존을 선택하고 "Create record"를 클릭합니다. ALB의 DNS 이름을 사용하여 A 레코드를 생성합니다. Alias로 ALB를 선택하여 연결합니다.

## 7단계: HTTPS 연결 설정 (SSL/TLS)

ACM (AWS Certificate Manager) 인증서 요청: AWS Management Console에서 "Certificate Manager"로 이동합니다. "Request a certificate"를 클릭하고, "Request a public certificate"를 선택합니다. 도메인 이름을 입력하고, "Next"를 클릭하여 인증서 요청을 완료합니다. 이메일을 통한 도메인 소유권 검증을 완료합니다. ALB에 HTTPS 리스너 추가: ALB의 설정으로 돌아가서 "Listeners" 탭을 선택합니다. "Add listener"를 클릭하고 HTTPS (포트 443)을 선택합니다. 인증서 설정에서 ACM에서 요청한 인증서를 선택합니다. HTTPS 리스너에 HTTP 리스너와 동일한 타겟 그룹을 추가합니다.

# AWS에서 VPC 전체 구성 및 설정 프로세스

## 8단계: 테스트 및 확인

테스트:Route 53에서 설정한 도메인을 사용하여 웹 브라우저에서 접속하여 설정이 올바르게 작동하는지 확인합니다.HTTPS를 통해 접속하여 인증서가 올바르게 적용되었는지 확인합니다.SSH를 통해 EC2 인스턴스에 접속하여 네트워크 설정 및 보안 그룹이 올바르게 적용되었는지 확인합니다.이렇게 단계별로 설정을 완료하면, VPC, 서브넷, 인터넷 게이트웨이, 라우팅, EC2 인스턴스, 보안 그룹, ALB, Route 53, HTTPS 연결이 모두 설정되어 안전하고 확장 가능한 웹 애플리케이션을 운영할 수 있는 환경이 구성됩니다.

# SSH로 로컬에서 연결

aws 서비스 검색 [알트+S]

EC2 IAM Route 53 VPC

EC2 > 인스턴스 > i-033cec7f76856eb3c > 인스턴스에 연결

인스턴스에 연결 정보

다음 옵션 중 하나를 사용하여 인스턴스 i-033cec7f76856eb3c (bnvl\_server)에 연결

EC2 인스턴스 연결 Session Manager SSH 클라이언트 EC2 직렬 콘솔

인스턴스 ID

i-033cec7f76856eb3c (bnvl\_server)

1. SSH 클라이언트를 엽니다.

2. 프라이빗 키 파일을 찾습니다. 이 인스턴스를 시작하는 데 사용되는 키는 bnvl\_key.pem입니다.

3. 필요한 경우 이 명령을 실행하여 키를 공개적으로 볼 수 없도록 합니다.

chmod 400 "bnvl\_key.pem"

4. 퍼블릭 IP를(를) 사용하여 인스턴스에 연결:

3.38.82.7

예:

ssh -i "bnvl\_key.pem" ubuntu@3.38.82.7

참고: 대부분의 경우 추정된 사용자 이름은 정확합니다. 하지만 AMI 사용 지침을 읽고 AMI 소유자가 기본 AMI 사용자 이름을 변경했는지 확인하세요.

취소

```
root@DESKTOP-EG3AB1S:~# ssh -i "./bnvl_aws/bnvl_key.pem" ubuntu@3.38.82.7
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-1015-aws x86_64)
```

```
* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/pro
```

System information as of Thu Oct 24 15:33:13 UTC 2024

```
System load:  0.01          Processes:            106
Usage of /:   8.2% of 28.89GB Users logged in:     1
Memory usage: 27%          IPv4 address for eth0: 10.0.6.20
Swap usage:   0%
```

\* Ubuntu Pro delivers the most comprehensive open source security and compliance features.

<https://ubuntu.com/aws/pro>

Expanded Security Maintenance for Applications is not enabled.

20 updates can be applied immediately.  
18 of these updates are standard security updates.  
To see these additional updates run: `apt list --upgradable`

Enable ESM Apps to receive additional future security updates.  
See <https://ubuntu.com/esm> or run: `sudo pro status`

New release '24.04.1 LTS' available.  
Run 'do-release-upgrade' to upgrade to it.

```
*** System restart required ***
Last login: Thu Oct 24 15:33:13 2024 from 58.234.212.51
ubuntu@ip-10-0-6-20:~$
```



# AWS 에서 인스턴스 연결

## 인스턴스에 연결 정보

다음 옵션 중 하나를 사용하여 인스턴스 i-033cec7f76856eb3c (bnvl\_server)에 연결

EC2 인스턴스 연결

Session Manager

SSH 클라이언트

EC2 직렬 콘솔



**EC2 인스턴스 연결 서비스 IP 주소가 승인되지 않음**

포트 22(SSH)는 [보안 그룹](#)에서 승인됩니다. 그러나 EC2 인스턴스 연결을 사용하려면 13.209.1.56/29 리전의 EC2 인스턴스 연결 서비스 IP 주소에 대해 포트 22도 승인하는 것이 좋습니다. [자세히 알아보기](#).

인스턴스 ID

i-033cec7f76856eb3c (bnvl\_server)

연결 유형



**EC2 Instance Connect를 사용하여 연결**

Connect using the EC2 Instance Connect browser-based client, with a public IPv4 or IPv6 address.



**EC2 인스턴스 연결 엔드포인트를 사용하여 연결**

프라이빗 IPv4 주소 및 VPC 엔드포인트가 있는 EC2 인스턴스 연결 브라우저 기반 클라이언트를 사용하여 연결합니다.

☒ 퍼블릭 IPv4 주소

3.38.82.7

☐ IPv6 주소

-

사용자 이름

인스턴스를 시작하는 데 사용되는 AMI에 정의된 사용자 이름을 입력합니다. 사용자 지정 사용자 이름을 정의하지 않은 경우 기본 사용자 이름인 ubuntu를(을) 사용합니다.

ubuntu



**참고:** 대부분의 경우 기본 사용자 이름 ubuntu은(는) 정확합니다. 하지만 AMI 사용 지침을 읽고 AMI 소유자가 기본 AMI 사용자 이름을 변경했는지 확인하세요.

취소

연결

