

1. 온라인 학습 회사가 AWS 클라우드로 이전 중입니다. 이 회사는 PostgreSQL 데이터베이스에 학생 기록을 보관합니다. 이 회사는 여러 AWS 지역에서 항상 데이터를 사용할 수 있고 온라인 상태인 솔루션이 필요합니다.

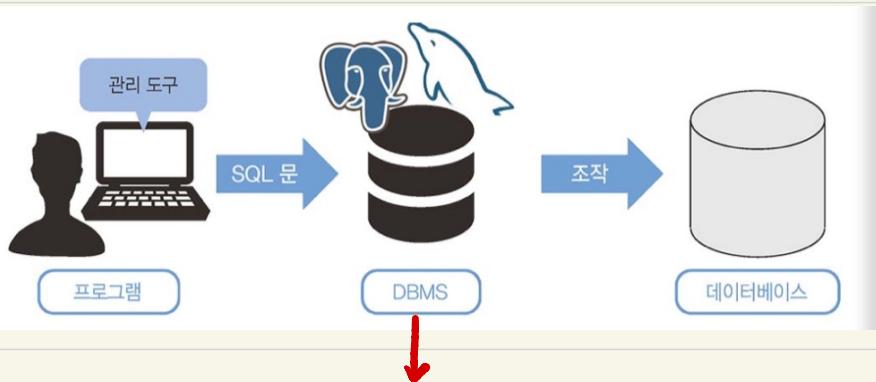
어떤 솔루션이 운영 오버헤드를 최소화하면서 이러한 요구 사항을 충족할까요?

↳ 가능 두통시 또는 지원

- A. PostgreSQL 데이터베이스를 Amazon EC2 인스턴스의 PostgreSQL 클러스터로 마이그레이션합니다.
- B. Multi-AZ 기능이 켜진 상태에서 PostgreSQL 데이터베이스를 Amazon RDS for PostgreSQL DB 인스턴스로 마이그레이션합니다.
- C. PostgreSQL 데이터베이스를 Amazon RDS for PostgreSQL DB 인스턴스로 마이그레이션합니다. 다른 Region에 읽기 복제본을 만듭니다.
- D. PostgreSQL 데이터베이스를 Amazon RDS for PostgreSQL DB 인스턴스로 마이그레이션합니다. 다른 Region으로 복사할 DB 스냅샷을 설정합니다

↳  
복제 / 복사

- 데이터 : 이론을 세워놓은 가능한가 되는 지식 → 정보 : 데이터 + 이론
  - DB : 구조적으로 정리된 데이터의 집합체  
→ DB 그 자체에서는 데이터를 조작하는 기능은 없다
  - DBMS : 데이터 관리, 저장, 검색 등 DB 조작 역할
  - SQL : DB에 명령 or 의사결정을 사람이 해야 함  
이 때 사용되는 명령어가 SQL  
SQL은 명령어(구문)와 명령방법(문법)이 정의되어 있다.
- 보통 단독사용 X 프로그램 사용



정확하게는 Oracle, MySQL, PostgreSQL  
등은 DBMS이다.

# DB 모델 유형

## 1) 계층형 DB 모델 (Hierarchical)

: tree구조 기반, 1:N 상위 종속관계  
부모 키를 가진 A, 자식키로 부모 키

## 2) 관계형

1:N, N:N, 1:1, 복잡도 ↑

## 3) 대상형

: object 형식 → Entity

자료구조  
object

## (\*) 대상형과 관계형

object view

c.f. Amz Aurora, SQL Server.

RDB : 주소록이나 액세스과 같은 표 형태로 되어 있음

NoRDB : 구조가 단순하고 시기상의 유동이 있다.  
(=NoSQL) 서비스는 빠르다

key-value, 문서형 구조, 그래프

SQL 가능

c.f. Amz DynamoDB, Amz ElastiCache

RDS: 관계형 DB 를 쉽게 쿠라우드에 최적화된 형태로 제공하는 서비스이다.

EC2처럼 쿠라우드에서 DB를 이용할 수 있다.

마이크로 서비스하기 때문에 멀티리스 / 관리 자동화  
기존 DB 이전 / 복제 가능

단일 AZ 구성: 한 곳에 가용 영역을 구축하는 것

다중 AZ 구성: 여러 가용 영역에 가용화의 이중화 구성을  
구현하는 것  
→ 가용한 서버를 통해 접속으로 부터  
데이터를 대체 사용하는 것이다.

RDS에서 사용할 수 있는 DB 엔진

: Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle Database  
SQL Server

Amazon은 RDS, MySQL 및 PostgreSQL과 호환 가능하다  
마이크로 서비스임

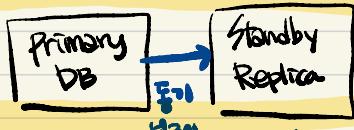
\* AWS에서 MySQL, PostgreSQL 사용법

- ① EC2에 설치
- ② RDS 사용
- ③ Aurora 사용

# RDS Cloud 복제 방식

1) Multi-AZ 복제 방식 → Cloud 정합성 유지 중요

: Active - Standby 복제



· 동기식 복제 사용

· 한 리전 내 고가용성 유지

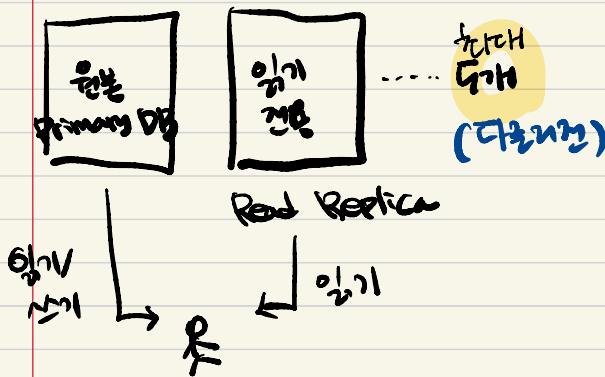
이동복제방식

Primary DB 능력

Active  
상태  
운영

2) Read Replica 복제 방식

: 읽기전용으로  
성능향상



## key-value Store (KVS)

: 어떤 데이터에 대해서 특정 Key 가 되는 값을  
절대화하여 저장하는 방식의 DB이다

. 관계형 DB 보다 RDB

### i) Dynamo DB (스토리지 저장)

: Key-value DB, 고급 기능 X 빠르고, 저렴함  
(f. RDB는 느리고 종복데이터 동일)

### ii) Elasti Cache (마이그레이션)

: 인메모리 DB

↳ DB 조작할 때마다 일부 기억 장치에 읽기/쓰기 수행

#### - Memcached

: 고급 기능 X 빠르고 저렴.

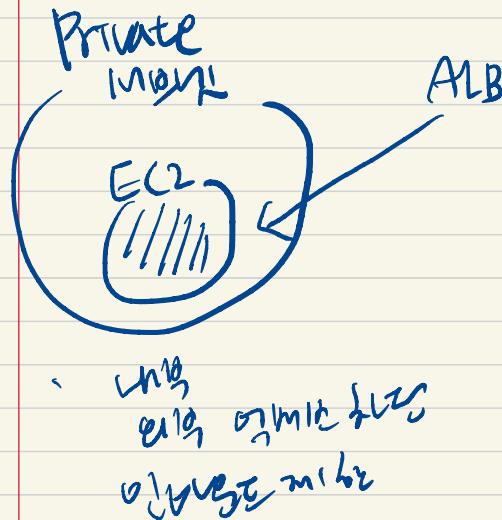
#### - Redis

: 데이터베이스, 큐, 브로커 및 캐시 역할을 하는 오픈소스  
NoSQL 및 미리생성된 데이터베이스

2. 한 회사가 VPC의 프라이빗 서브넷에 있는 Amazon EC2 인스턴스에 배포된 웹 애플리케이션을 실행합니다. 퍼블릭 서브넷을 가로지르는 애플리케이션 로드 밸런서(ALB)가 웹 트래픽을 EC2 인스턴스로 보냅니다. 이 회사는 EC2 인스턴스의 프라이빗 서브넷 내부 또는 외부의 다른 소스에서 액세스를 차단하는 동시에 ALB에서 EC2 인스턴스로의 인바운드 트래픽을 제한하는 새로운 보안 조치를 구현하려고 합니다.

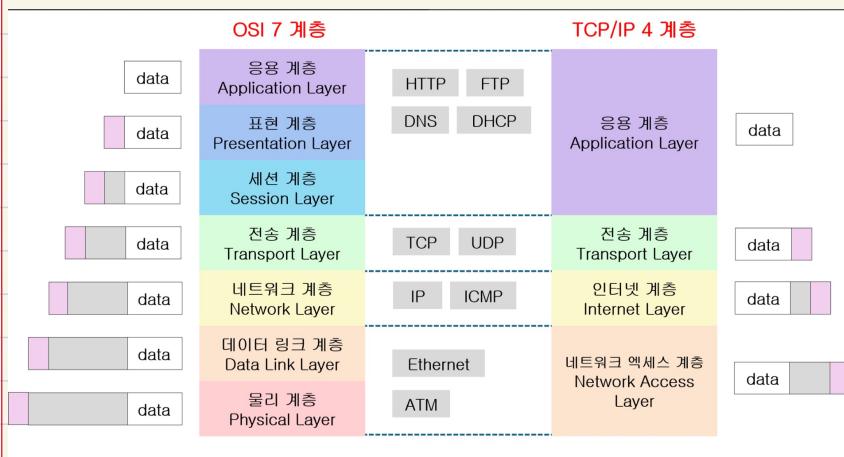
어떤 솔루션이 이러한 요구 사항을 충족할까요

- A. 인터넷에서 EC2 인스턴스의 개인 IP 주소로 트래픽을 전달하기 위해 경로 테이블에 경로를 구성합니다. ✗
- B. ALB의 보안 그룹에서 오는 트래픽만 허용하도록 EC2 인스턴스의 보안 그룹을 구성합니다. ↗
- C. EC2 인스턴스를 퍼블릭 서브넷으로 이동합니다. EC2 인스턴스에 일련의 Elastic IP 주소를 제공합니다. ✗
- D. ALB에 대한 보안 그룹을 구성하여 모든 포트에서 모든 TCP 트래픽을 허용합니다.



# 네트워킹

## OSI 7 계층 모델 : 네트워크 통신을 위한



## IP 주소와 주소ing

↪ 인터넷상에서 IT자원을 식별하는 고유한 주소

- [IPv4 : 가용범위 부족 · 효율↓ · 많이 사용]
- [IPv6 : 가용범위 ↑]

## 피복적 주소 / 프라이빗 주소

IP 주소는 통신유도체 따라 페복적 / 프라이빗으로 나뉨

실제 인터넷에서  
사용하지 않고 인터넷 밖에 공유하지  
제공하는 유일한 공인 IP 주소

인터넷 외부  
통신 네트워크 내부에 있는  
사용되는 관리자와 제공하는  
사내 IP 주소

## 공인 IP 주소 · 유통 IP주소

인터넷에 공개되는

주소

유통주소

IP 주소 네트워크에 대처

등록주소

네트워크  
0000

서브넷과

부분네트워크

서브넷 마스크

서브넷을 구분하는 것

▼ 그림 3-3 서브넷 마스크를 이용한 서브넷 구분

서브넷 마스크	서브넷 구분기준			호스트 ID 0000000	동일한 서브넷 구분기준
	네트워크 ID	255 11111111	255 11111111		
IP 주소	192 11000000	168 10101000	100 01100100	0 00000001	동일한 서브넷
	192 11000000	168 10101000	100 01100100	2 00000010	동일한 서브넷
	192 11000000	168 10101000	200 11001000	1 00000001	다른 서브넷

퍼블릭 서브넷

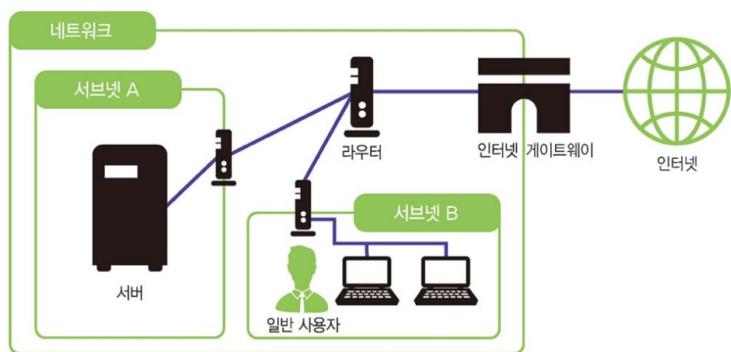
외부에서 접근 가능한 리소스 핸드リング / 공인IP

프라이빗 서브넷

: 외부 접근 불가능 / 내부망에 연결 / 사설IP 주소

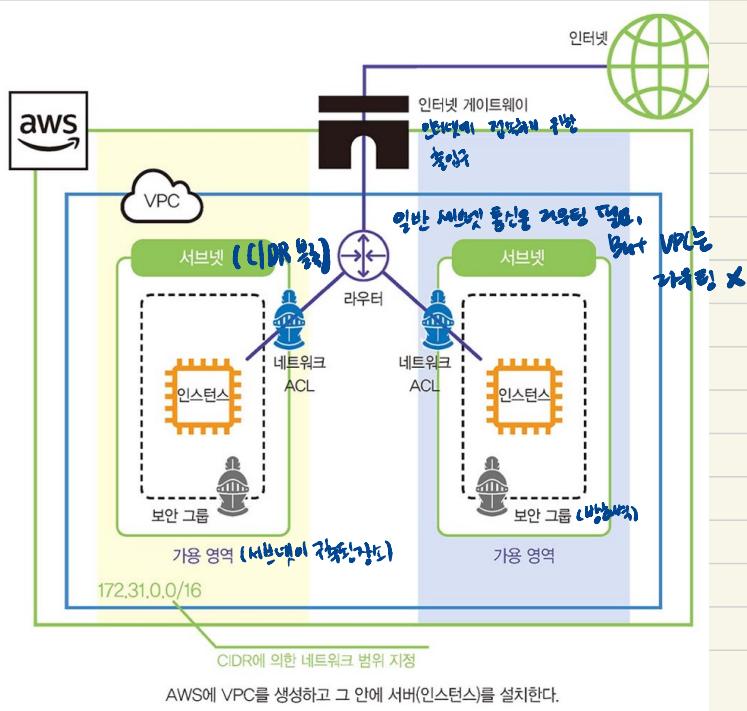
# VPC란? Virtual Private Cloud

- AWS가 제공하는 AWS 전용 가상 네트워크이다.
- 네트워크, 서브넷 범위, 라우팅 테이블 등 가상 네트워킹 환경 설정

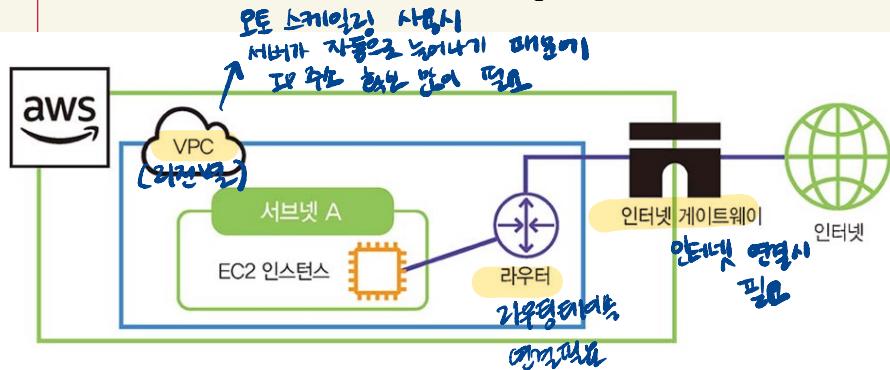


▲ 그림 6-1 일반적인 네트워크의 예

인터넷 연결 설정  
필요함



# VPC 설정시 필요한 사항

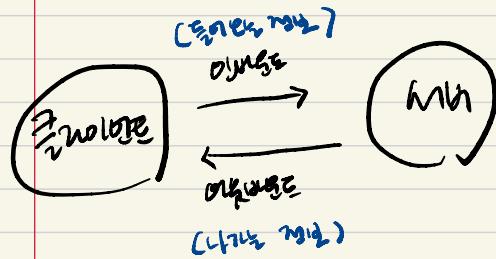


## Elastic IP

- : AWS가 제공하는 정적인 공인(public) IP주소
- EC2는 정지 후 재시작하면 IP가 바뀌는데, 이 때 사용하는 것  
제정당 제공됨

# 부정부수

\* 대체로 대체로는 전문가의 전문적인 입장



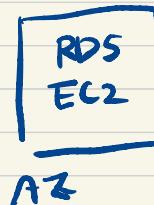
3. 한 회사가 Auto Scaling 그룹의 Amazon EC2 인스턴스에서 웹 애플리케이션을 실행합니다. 이 애플리케이션은 Amazon RDS for PostgreSQL DB 인스턴스에서 실행되는 데이터베이스를 사용합니다. 트래픽이 증가하면 애플리케이션이 느리게 실행됩니다. 이 데이터베이스는 트래픽이 많은 기간 동안 많은 읽기 부하를 경험합니다.

솔루션 아키텍트는 이러한 성능 문제를 해결하기 위해 어떤 조치를 취해야 합니까? (두 가지를 선택하세요.)

→ RDS 퍼포먼스 트래픽 \*

- A. DB 인스턴스에 대한 자동 크기 조정을 켭니다.
- B. DB 인스턴스에 대한 읽기 복제본을 만듭니다. 읽기 복제본에 읽기 트래픽을 보내도록 애플리케이션을 구성합니다.
- C. DB 인스턴스를 Multi-AZ DB 인스턴스 배포로 변환합니다. 대기 DB 인스턴스로 읽기 트래픽을 보내도록 애플리케이션을 구성합니다.
- D. Amazon ElastiCache 클러스터를 만듭니다. ElastiCache 클러스터에서 쿼리 결과를 캐시하도록 애플리케이션을 구성합니다.
- E. EC2 인스턴스가 DB 인스턴스와 동일한 가용성 영역에 프로비저닝되도록 자동 크기 조정 그룹 서브넷을 구성합니다.

캐시링으로  
속도향상



: 자연 강소 O  
But DB  
연결 수준에는  
동일

4. 한 회사가 여러 Amazon EC2 인스턴스에서 웹 애플리케이션을 호스팅합니다. EC2 인스턴스는 사용자 수요에 따라 확장되는 자동 확장 그룹에 있습니다. 이 회사는 장기적 약속 없이 비용 절감을 최적화하고자 합니다.

☞ X

☞ X

이러한 요구 사항을 충족하기 위해 솔루션 아키텍트는 어떤 EC2 인스턴스 구매 옵션을 권장해야 합니까?

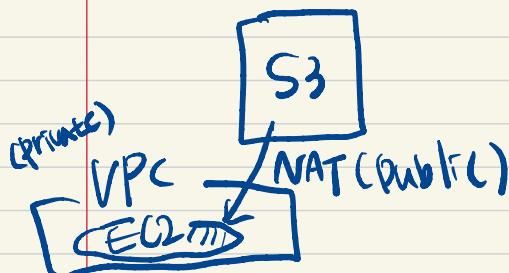
- A. 전용 인스턴스만
- B. 주문형 인스턴스만 → 가장 비쌈
- C. 온디맨드 인스턴스와 스팟 인스턴스의 혼합
- D. 온디맨드 인스턴스와 예약 인스턴스의 혼합

5. 한 회사에는 동일한 AWS 리전의 Amazon S3 버킷에서 대량의 데이터를 읽고 쓰는 서비스가 있습니다. 이 서비스는 VPC의 프라이빗 서브넷 내의 Amazon EC2 인스턴스에 배포됩니다. 이 서비스는 퍼블릭 서브넷의 NAT 게이트웨이를 통해 Amazon S3와 통신합니다. 그러나 이 회사는 데이터 출력 비용을 줄이는 솔루션을 원합니다.

어떤 솔루션이 이러한 요구 사항을 가장 비용 효율적으로 충족할까요?

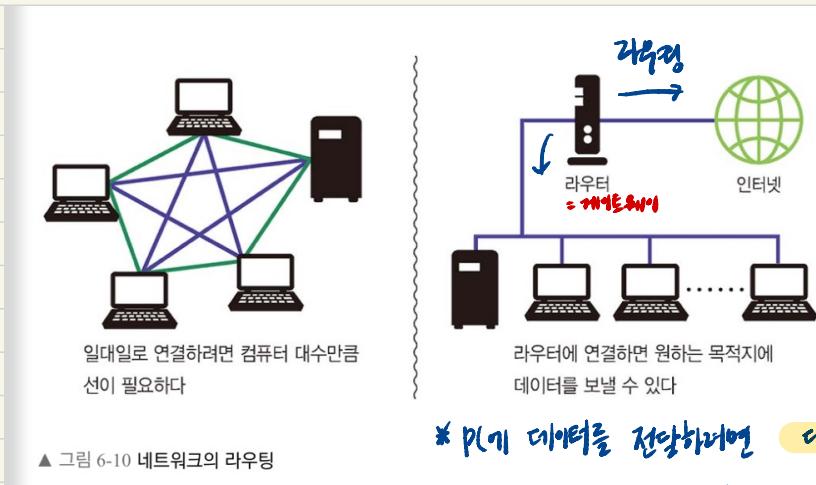
- A. 퍼블릭 서브넷에 전용 EC2 NAT 인스턴스를 프로비저닝합니다. 이 인스턴스의 탄력적 네트워크 인터페이스를 모든 S3 트래픽의 대상으로 사용하도록 프라이빗 서브넷의 경로 테이블을 구성합니다.
- B. 프라이빗 서브넷에 전용 EC2 NAT 인스턴스를 프로비저닝합니다. 퍼블릭 서브넷의 경로 테이블을 구성하여 이 인스턴스의 탄력적 네트워크 인터페이스를 모든 S3 트래픽의 대상으로 사용합니다.
- C. VPC 게이트웨이 엔드포인트를 프로비저닝합니다. 모든 S3 트래픽의 경로로 게이트웨이 엔드포인트를 사용하도록 프라이빗 서브넷의 경로 테이블을 구성합니다.
- D. 두 번째 NAT 게이트웨이를 프로비저닝합니다. 이 NAT 게이트웨이를 모든 S3 트래픽의 대상으로 사용하도록 프라이빗 서브넷의 경로 테이블을 구성합니다.

A Z



# NAT / 라우팅

네트워크 간 데이터 주고받는 1 방식



\* PC가 데이터를 전달하려면 대상 설정 후고 필요

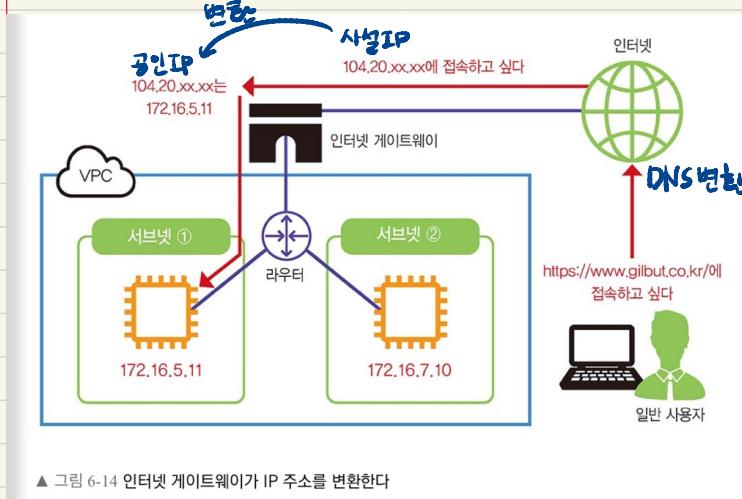
\* IP 주소를 통하여 연결, 모니터, 거우터 등 네트워크 장치를 관리하기 다 있다.

IP주소

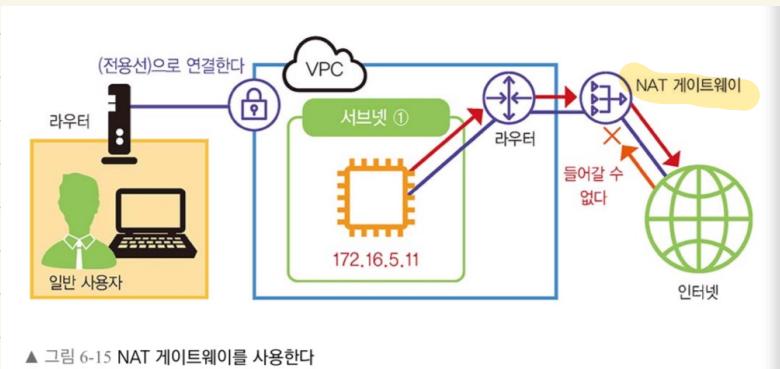
# 라우터

: 데이터를 주고받을 기기

I) 인터넷 게이트웨이 → 인터넷 연결 담당



▲ 그림 6-14 인터넷 게이트웨이가 IP 주소를 변환한다



▲ 그림 6-15 NAT 게이트웨이를 사용한다