

1. Web 개요

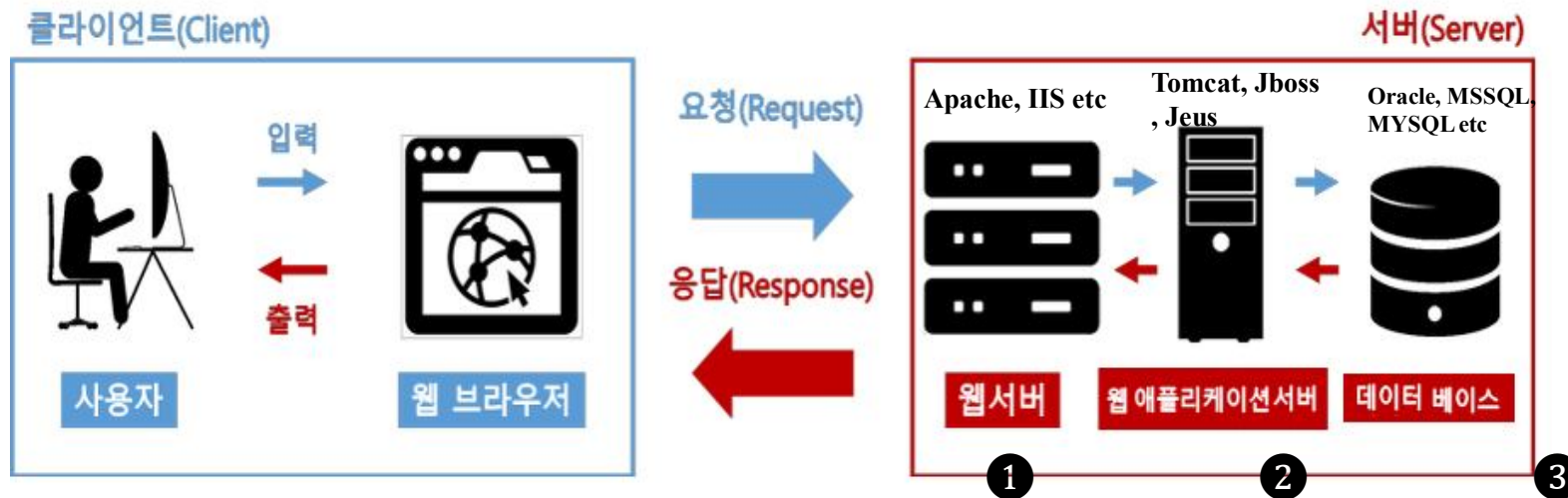
1.1 Web System Architecture

1.2 URI(Uniform Resource Identifier)

1.3 Web Proxy Server

1.4 Web Proxy를 이용한 HTTP 구조 분석

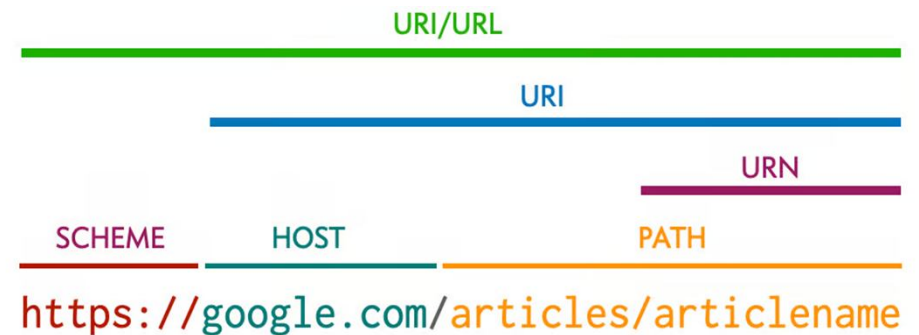
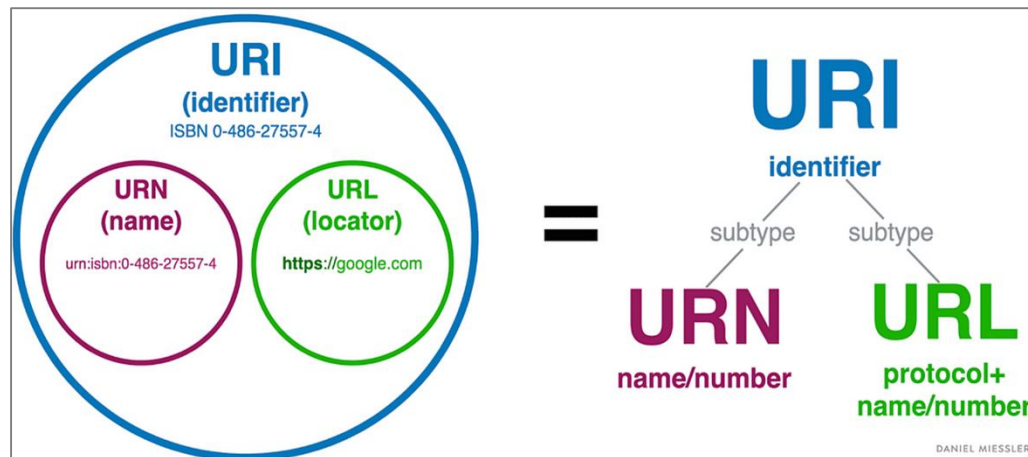
1.1 Web System Architecture

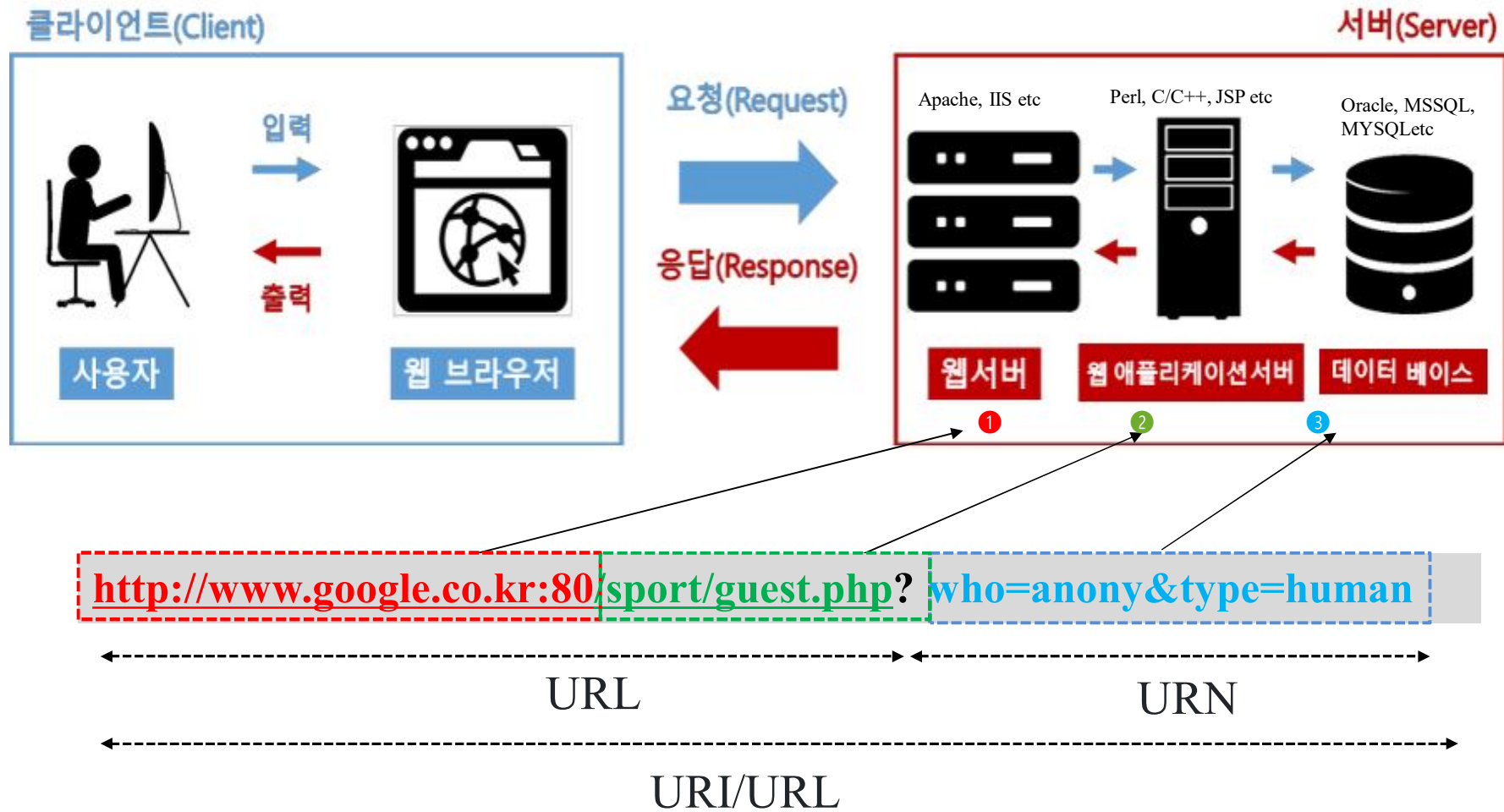


①	FrontServer (Web Server)	<ul style="list-style-type: none"> • HTTP request를 받아 HTTP response를 보내는 서버 • Apache, IIS, Nginx 등의 미들웨어가 동작
②	Application Server	<ul style="list-style-type: none"> • 업무처리 서버 (결제, 주문 등을 수행하는 프로그램 실행환경) • Apache Tomcat, WebSphere Application의 미들웨어로 구현 • PHP나 Perl 실행환경이 필요 시 mod_perl 또는 mod_php 등 설치
③	DB Server	<ul style="list-style-type: none"> • 애플리케이션 실행 시 생성되는 데이터 관리 기능을 가진 미들웨어 • MySQL, PostgreSQL, Oracle Database

1.2 URI(Uniform Resource Identifier)

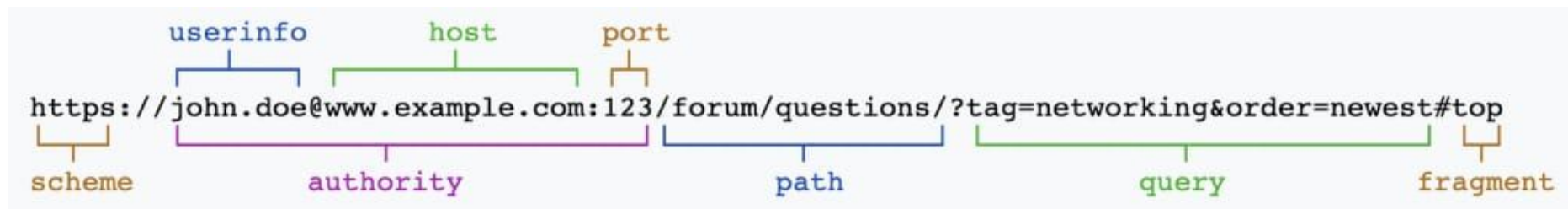
- 자원을 식별하는 통합된 방법
- URL(Resource Locator, 자원의 위치), URN(Resource Name, 자원의 이름)을 포함





◎ URL (Uniform Resource Locator)

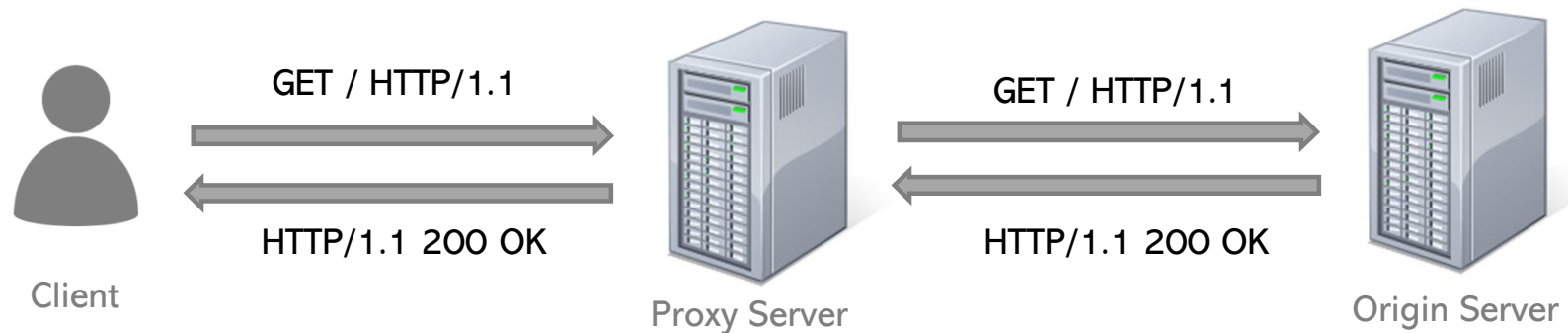
scheme:[//[user[:password]@]host[:port]][/path][?query][#fragment]



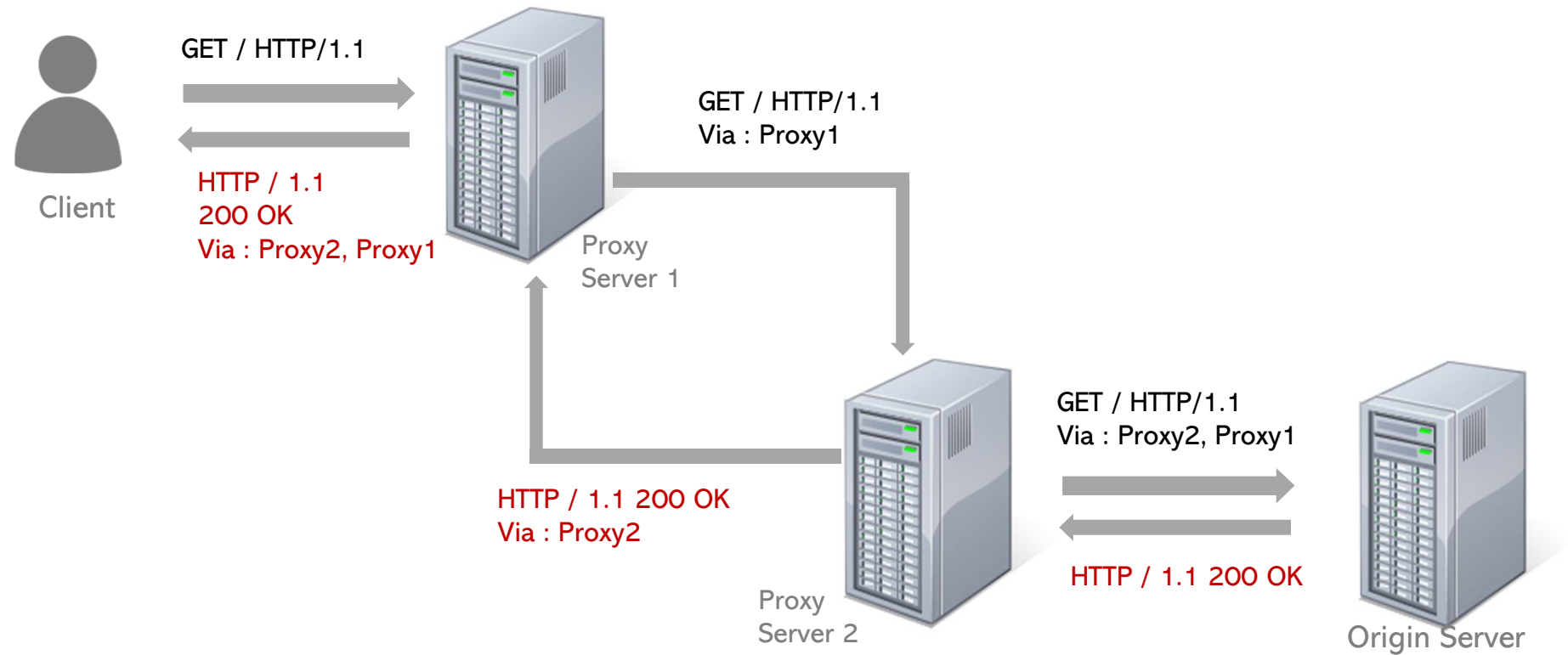
- scheme : 사용할 프로토콜을 뜻하며 웹에서는 http 또는 https를 사용
- userinfo : (서버에 있는) 데이터에 접근하기 위한 사용자의 이름과 비밀번호
- Host와 port : 접근할 대상(서버)의 호스트명과 포트번호
- path : 접근할 대상(서버)의 경로에 대한 상세 정보
- query : 접근할 대상에 전달하는 추가적인 정보 (파라미터)
- fragment : 메인 리소스 내에 존재하는 서브 리소스에 접근할 때 이를 식별하기 위한 정보

1.3 Web Proxy Server

- 서버와 클라이언트 간의 요청과 응답을 중계하는 역할을 수행하는 서버(프로그램)
 - 트래픽 부하, 캐시, 보안 등 여러가지 목적으로 사용



1) Multi-Layer Web Proxy Server



2) 기능

- 보안

- 실제 서버 또는 클라이언트의 IP를 숨김
- 프록시 서버의 IP만 공개함으로써 해킹 대비 가능

- 성능

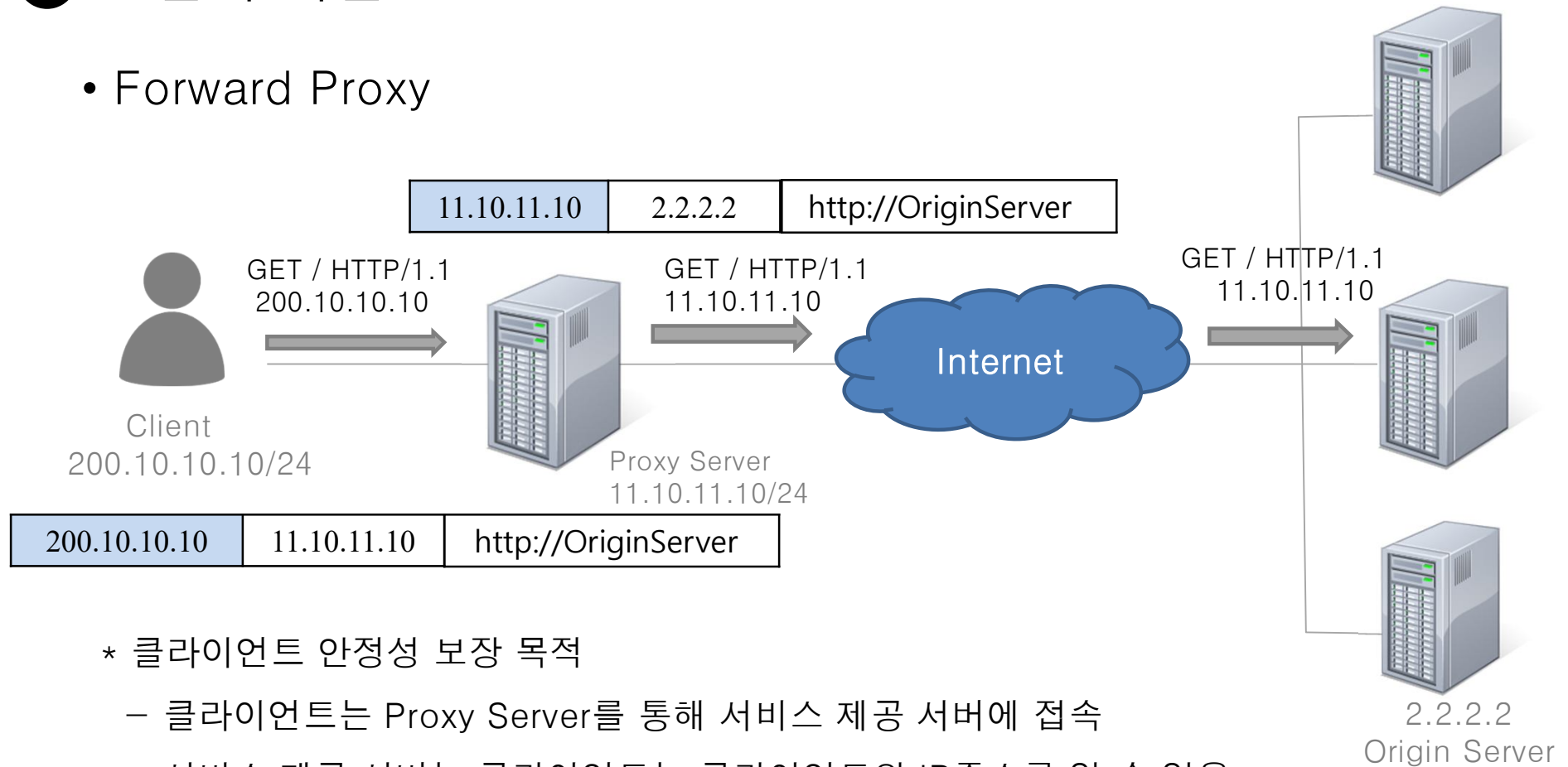
- 캐싱 기능과 트래픽 분산으로 성능 향상을 가져올 수 있음
- 캐싱 기능 : 자주 사용되는 동일한 요청을 캐싱하여 재활용하는 방식
- 실제 서버를 호출하지 않고 프록시 서버가 대신 응답을 주어 서버의 자원 사용을 줄임

- 트래픽 분산

- 일부 프록시 서버는 로드 밸런싱도 제공
- 여러 대의 분산된 서버가 있다면 서버의 트래픽을 분산시킴

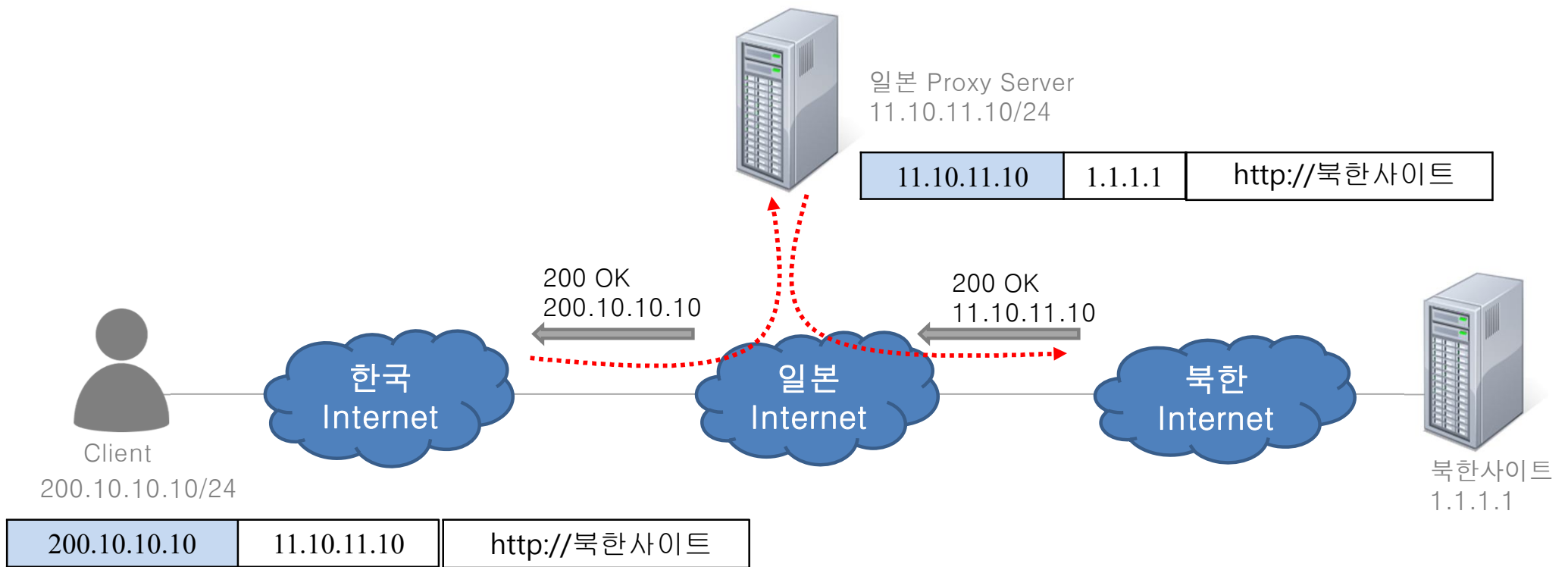
1 보안적 측면

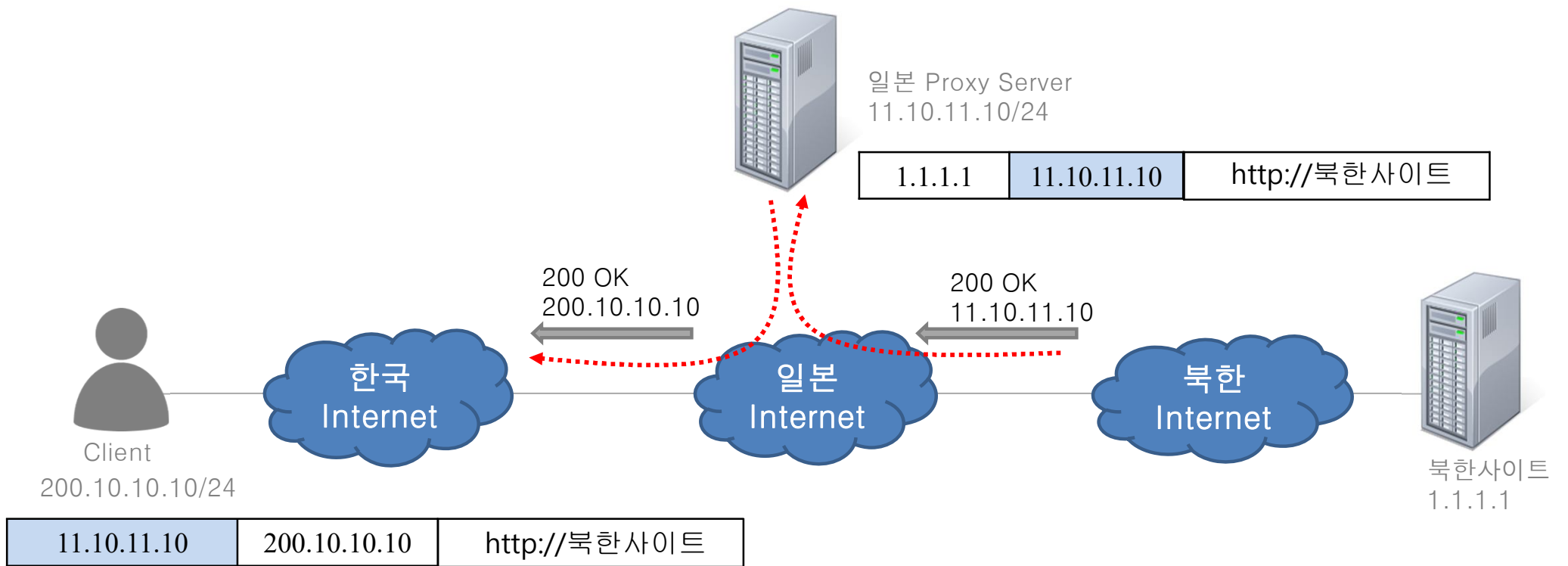
• Forward Proxy



* 클라이언트 안정성 보장 목적

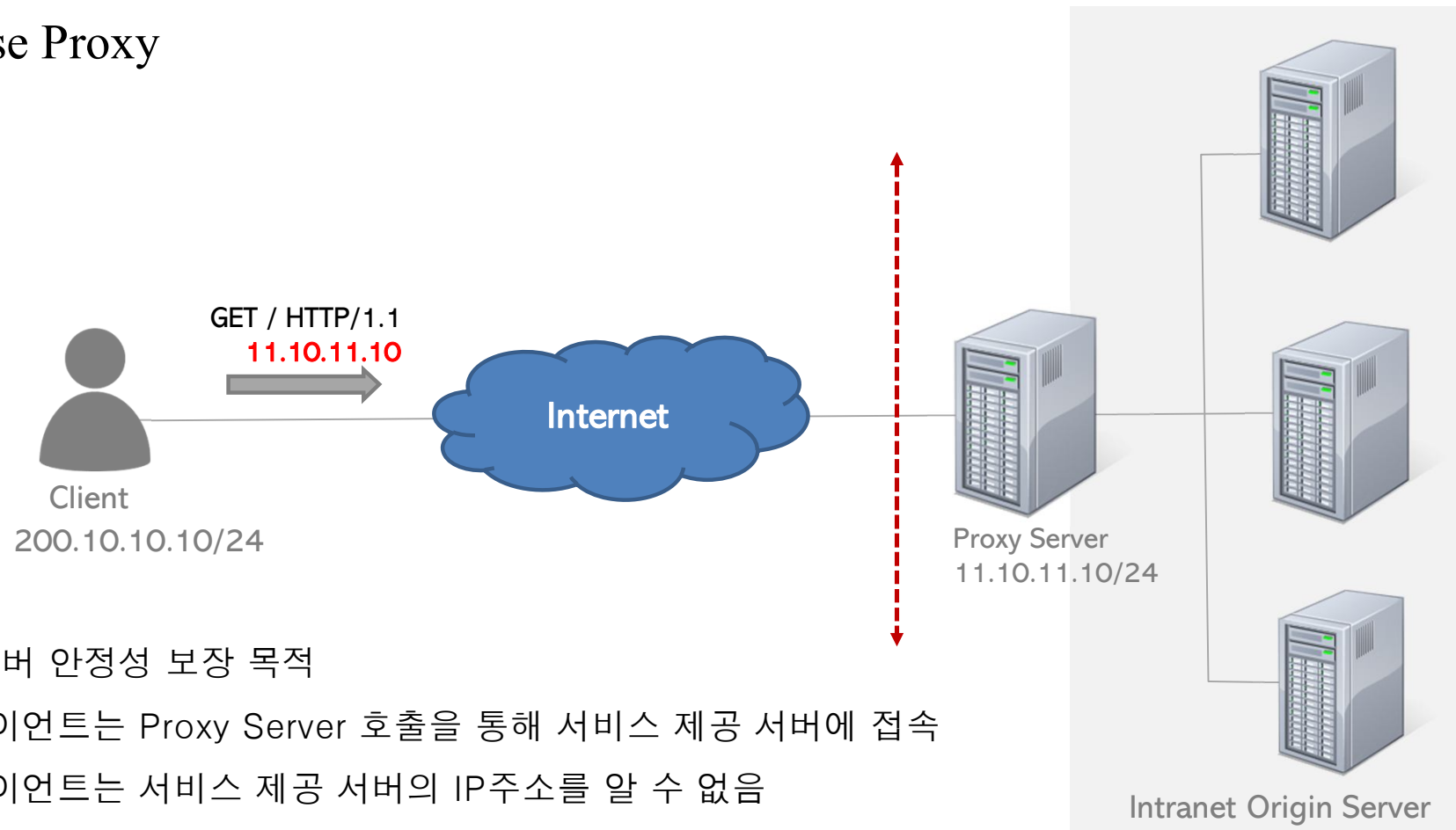
- 클라이언트는 Proxy Server를 통해 서비스 제공 서버에 접속
- 서비스 제공 서버는 클라이언트는 클라이언트의 IP주소를 알 수 없음





② 보안적 측면 & 트래픽분산

• Reverse Proxy

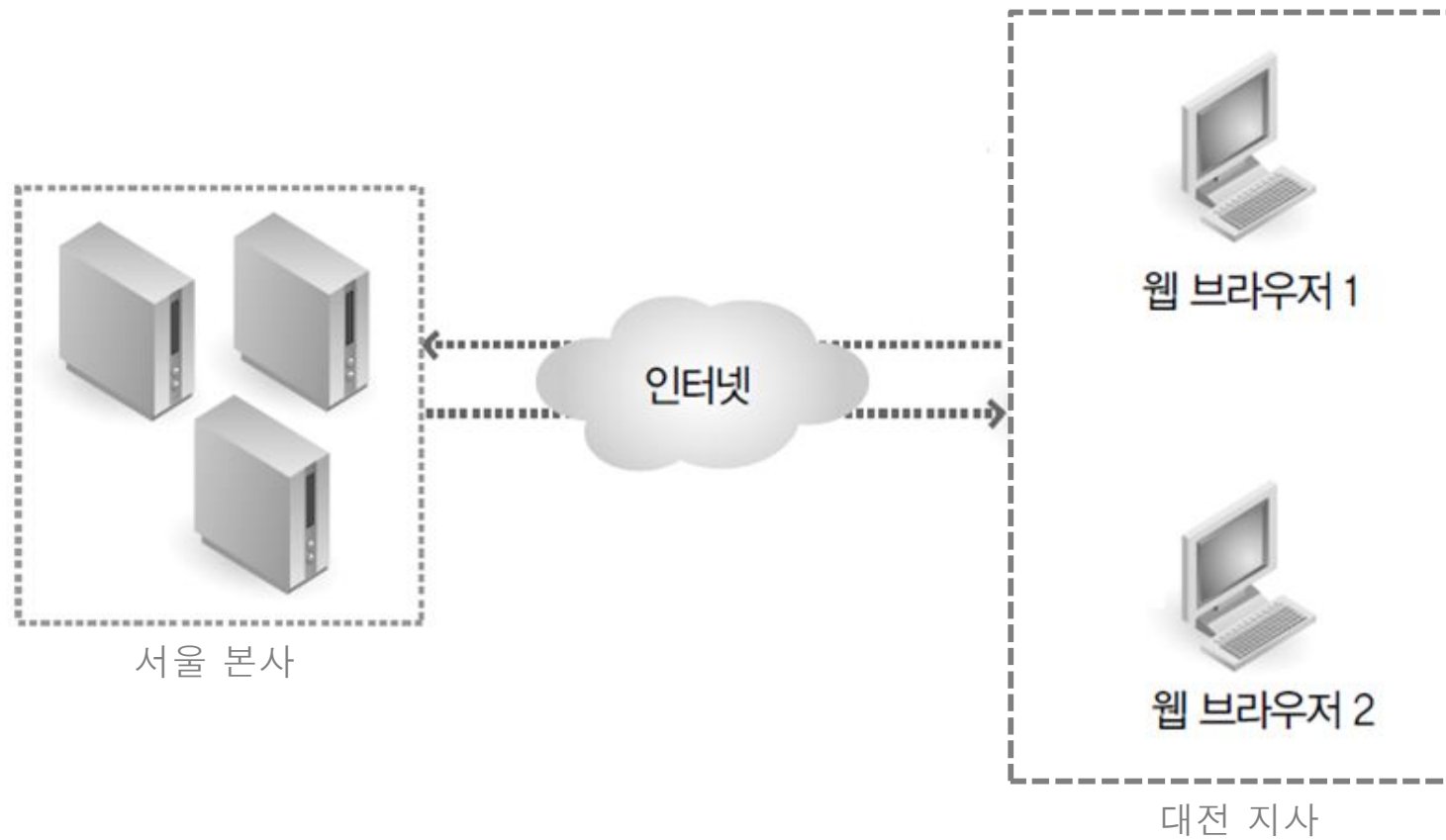


* 내부 서버 안정성 보장 목적

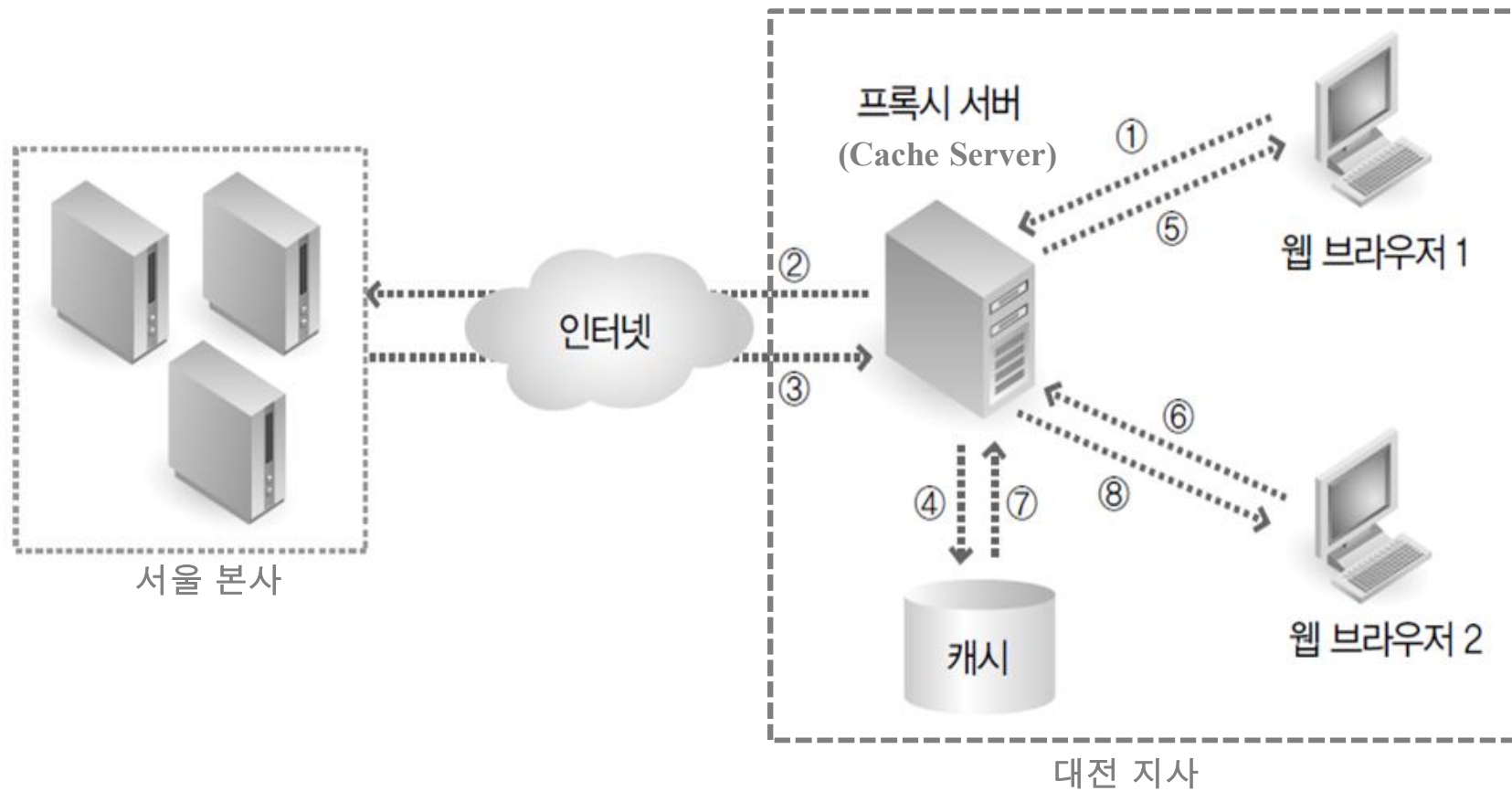
- 클라이언트는 Proxy Server 호출을 통해 서비스 제공 서버에 접속
- 클라이언트는 서비스 제공 서버의 IP주소를 알 수 없음

③ 성능적 측면

- HTTP Proxy가 없는 경우 작업 지연 발생



- HTTP Proxy가 있는 경우 작업 지연이 낮음



2.4 Web Proxy를 이용한 HTTP 구조 분석

- 웹 클라이언트와 웹 서버 간에 전달되는 모든 HTTP 패킷을 확인 및 수정 가능
- 웹의 구조 파악, 취약점 점검, 웹 해킹에는 웹 프록시 툴을 사용

Web Proxy 환경구성

