

# 프로세스 관리

## 6 클라이언트 서버 환경에서 통신

## 프로세스 관리 - 학습목표

### 3.

#### 프로세스 관리

- 프로세스의 개별 구성요소를 이해한다.
- 스케줄링의 목적 및 종류와 흐름을 확인한다.
- 컨텍스트 스위치 개념을 이해한다.
- 운영체제에서의 프로세스 생성/종료를 이해한다.
- 공유 메모리와 메시지 전달을 통한 프로세스간 통신에 대해 이해하고 비교한다.
- 소켓 통신과 원격 프로시저 콜을 이용한 client-server 통신을 이해한다.

## 클라이언트 서버 환경에서 소켓 통신 통신

3.

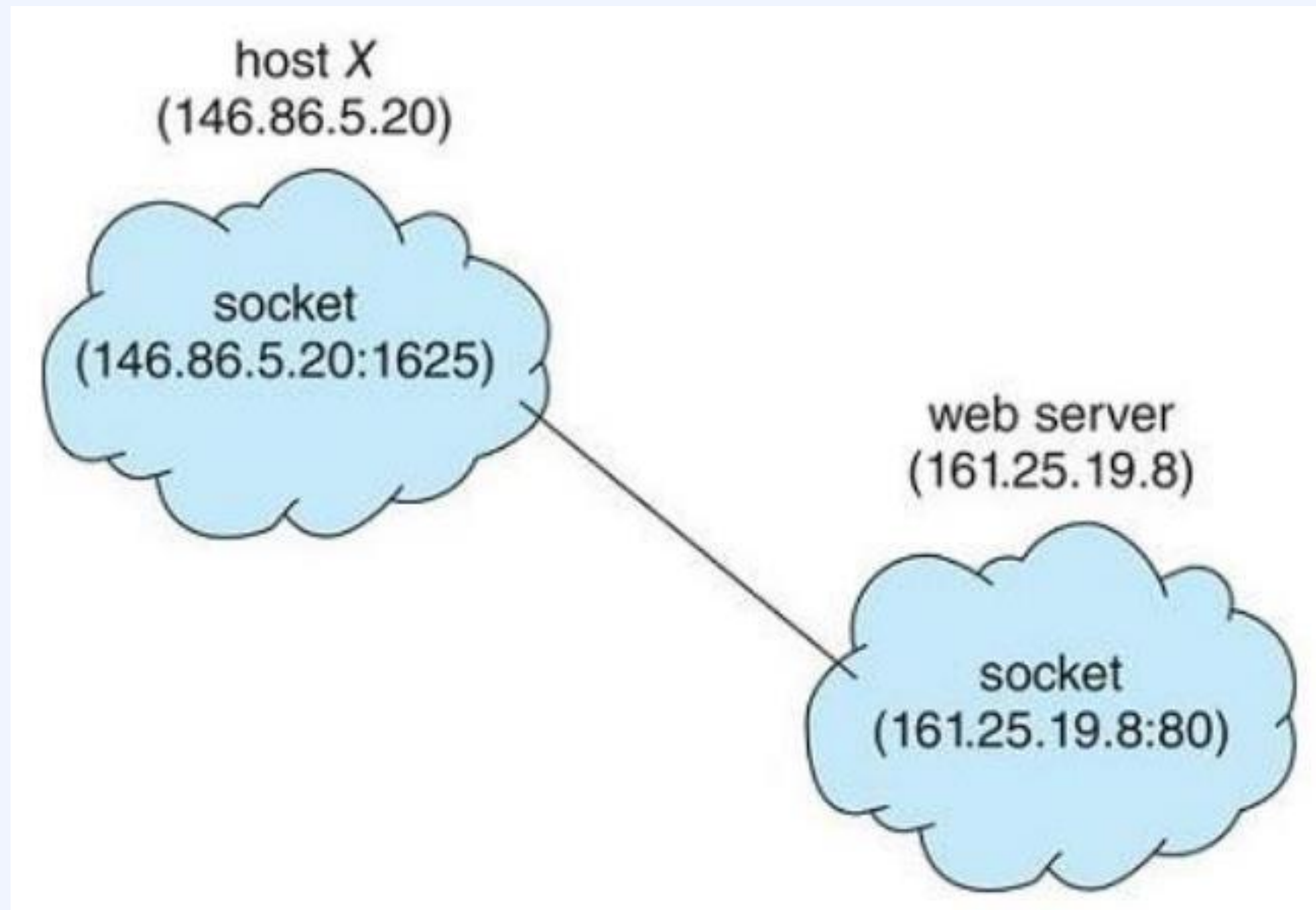
프로세스 관리

- 소켓(Socket)
  - ➔ 커뮤니케이션을 하기 위한 통신의 극점(Endpoint)
  - ➔ IP주소 및 Port 번호를 조합하여 식별
- 클라이언트-서버 아키텍처를 사용
  - ➔ 서버는 특정 포트를 listen하면서 클라이언트 요청에 대기
  - ➔ 요청을 받으면, 서버는 클라이언트 소켓으로부터의 연결 요청을 수락함으로써 연결이 완성

## 클라이언트 서버 환경에서 소켓 통신 예시 통신

3.

프로세스 관리

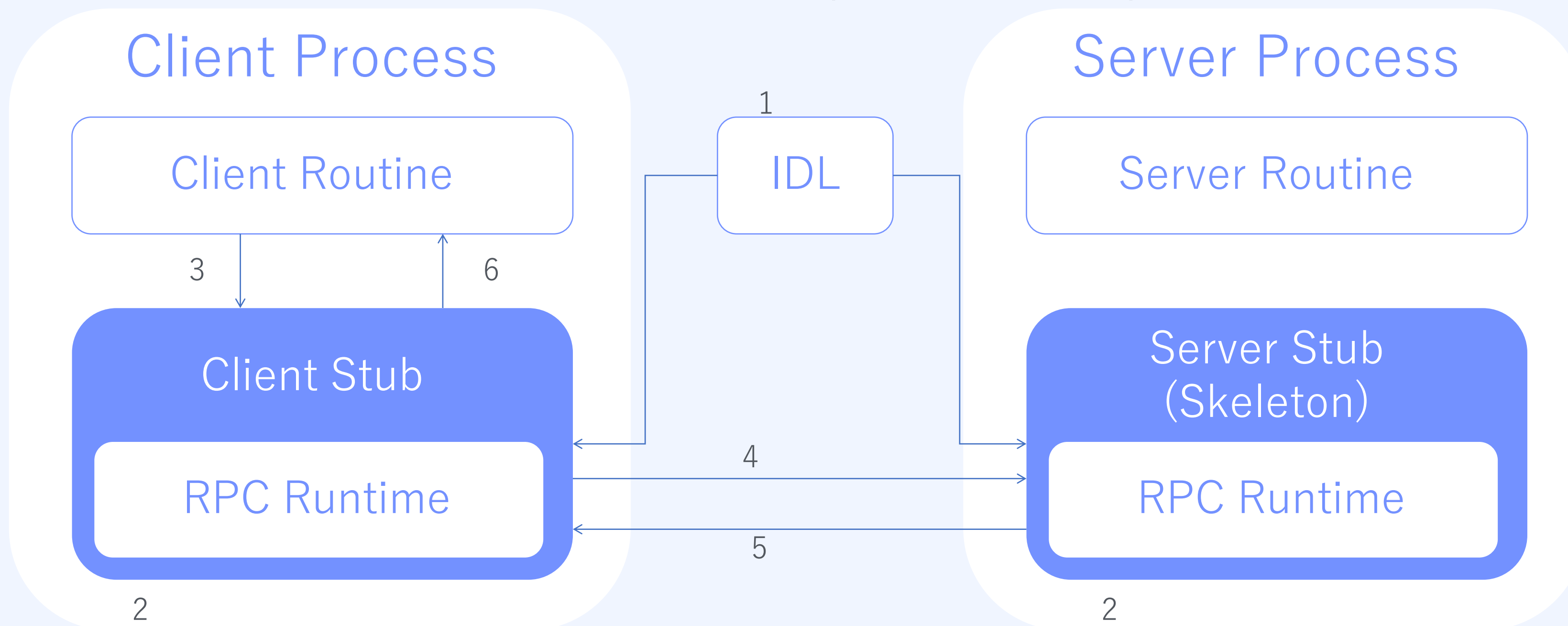


## 클라이언트 서버 환경에서 원격 프로시저 호출 통신

### 3. 프로세스 관리

### 원격 프로시저 호출(Remote Procedure Call, RPC)

- 네트워크로 연결된 서버 상의 프로시저(함수, 메소드 등)를 원격으로 호출할 수 있는



1 RPC 생성 2 생성한 stub과 client 프로그램 빌드 3 함수 호출 4 함수 호출 5,6 결과 반환