

# 제2장

# 개방형 시스템 상호 연결 사양

---

이 장에서는 다음과 같은 CompTIA Network+ 시험 목표를 다룹니다.

- 1.1 개방형 시스템 상호 연결(OSI) 모델의 계층과 캡슐화 개념을 비교 및 대조 하시오.
    - OSI 모델
      - 1단계 – 물리적
      - 레이어 2 – 데이터 링크
      - 레이어 3 – 네트워크
      - 4계층 – 전송
      - 레이어 5 – 세션
      - 6단계 – 프레젠테이션
      - 7단계 – 응용 프로그램
    - OSI 모델 맥락 내에서의 데이터 캡슐화 및 역캡슐화
  - 1.5 일반적인 포트와 프로토콜, 그 응용 분야 및 암호화된 대안에 대해 설명하십시오.
    - 비연결형 vs. 연결형
- 

이 장에서는 OSI(개방형 시스템 상호 연결) 모델을 분석해 보겠습니다. OSI 모델의 핵심 개념을 이해하는 것이 매우 중요하기 때문에 각 부분을 자세히 설명하겠습니다.

다. 이 필수적인 기초를 탄탄히 다진 후에는 네트워크 지식을 스스로 쌓아 나갈 수 있을 것입니다.

OSI 모델은 서로 다른 시스템 간의 안정적인 통신을 가능하게 하기 위해 개발된 7 개의 계층 구조로 이루어져 있습니다.

이 책은 Network+ 자격증에 관한 모든 것을 다루기 때문에 CompTIA에서 보는 OSI 모델을 이해하는 것이 매우 중요합니다. 따라서 각 계층을 CompTIA의 관점에서 설명하겠습니다.

또한 OSI 계층 구조를 따라 데이터가 전달될 때 데이터를 인코딩하는 과정인 캡슐화에 대한 소개도 해드리겠습니다.



Todd Lammle의 CompTIA 관련 영상과 질문을 보시려면  
다음 링크를 참조하세요 . [www.lammle.com](http://www.lammle.com)

---

## 인터넷 네트워킹 모델

초창기 네트워크에서는 컴퓨터들이 같은 제조사에서 만든 컴퓨터들과만 통신할 수 있었습니다. 예를 들어, 기업들은 DECnet 솔루션이나 IBM 솔루션 중 하나만 사용했고, 둘을 동시에 사용하는 경우는 없었습니다. 이러한 장벽을 허물기 위해 1970년대 후반 국제표준화기구(ISO)에서 *OSI(Open Systems Interconnection)* 참조 모델을 개발했습니다.

OSI 모델은 제조사들이 프로토콜 또는 표준 형태로 상호 운용 가능한 네트워크 장치와 소프트웨어를 개발하여 서로 다른 제조사의 네트워크가 호환되고 함께 작동할 수 있도록 돋기 위해 만들어졌습니다. 세계 평화처럼 완전히 실현되기는 어렵겠지만, 여전히 훌륭한 목표입니다.

OSI 모델은 네트워크의 주요 아키텍처 모델입니다. 이 모델은 한 컴퓨터의 애플리케이션에서 네트워크 매체를 통해 다른 컴퓨터의 애플리케이션으로 데이터와 네트워크 정보가 어떻게 전달되는지를 설명합니다. OSI 참조 모델은 이러한 접근 방식을 계층으로 나눕니다.

이제 이 계층적 접근 방식을 자세히 살펴보고, 핵심 개념을 활용하여 네트워크 문제를 해결하는 방법을 알아보겠습니다.

## 계층적 접근 방식

기본적으로 참조 모델은 통신이 어떻게 이루어져야 하는지에 대한 개념적 청사진입니다. 효과적인 통신에 필요한 모든 프로세스를 다루고 이러한 프로세스를 계층이라고 하는 논리적 그룹으로 나눕니다. 통신 시스템이 이러한 방식으로 설계되면 계층형 아키텍처라고 합니다.

이렇게 생각해 보세요. 당신과 친구들이 회사를 창업한다고 가정해 봅시다. 가장 먼저 할 일 중 하나는 모여 앉아 어떤 업무를 해야 하는지, 누가 그 업무를 맡을지, 어떤 순서로 진행해야 하는지, 그리고 업무들이 서로 어떻게 연관되는지를 고민하는 것입니다. 궁극적으로 이러한 업무들을 부서별로 묶을 수도 있습니다. 예를 들어 고객 서비스 부서, 재고 관리 부서, 배송 부서를 만들기로 했다고 가정해 봅시다. 각 부서는 고유한 업무를 가지고 있어 직원들이 바쁘게 일하고 자신의 업무에만 집중할 수 있도록 합니다.

이 시나리오에서 저는 부서를 커뮤니케이션 시스템의 계층 구조에 대한 은유로 사용하고 있습니다. 원활한 운영을 위해서는 각 부서 직원들이 서로를 신뢰하고 의지하여 각자의 업무를 능숙하게 수행하고 고유한 책임을 완수해야 합니다. 계획 회의를 진행하면서 전체 과정을 기록해 두면 나중에 운영 표준에 대한 논의를 용이하게 하고, 이를 비즈니스 청사진 또는 참고 모델로 활용할 수 있습니다.

사업을 시작하면 각 부서장은 사업 모델 청사진 중 해당 부서와 관련된 부분을 활용하여 할당된 업무를 수행할 실질적인 방법을 개발해야 합니다. 이러한 실질적인 방법, 즉 프로토콜은 표준 운영 절차 매뉴얼로 정리하여 철저히 준수해야 합니다. 매뉴얼에 포함된 절차들은 각기 다른 이유로 포함되며 중요도와 실행 수준도 다양합니다. 만약 파트너십을 맺거나 다른 회사를 인수하는 경우, 해당 회사의 사업 프로토콜이 자사의 프로토콜과 일치하거나 호환되는 것이 매우 중요합니다.

마찬가지로 소프트웨어 개발자는 참조 모델을 사용하여 컴퓨터 통신 프로세스를 이해하고 각 계층에서 정확히 무엇을 어떻게 수행해야 하는지 파악할 수 있습니다. 다시 말해, 특정 계층에 대한 프로토콜을 개발해야 하는 경우 해당 계층의 기능에만 집중하면 됩니다. 다른 계층의 기능은 신경 쓸 필요가 없는데, 각 계층의 요구 사항을 충족하는 다른 프로토콜이 이미 존재하기 때문입니다. 이러한 개념을 기술적으로 '바인딩'이라고 합니다. 서로 관련된 통신 프로세스는 특정 계층에서 바인딩되거나 그룹화됩니다.

## 참조 모델의 장점

OSI 모델은 계층적 구조를 가지고 있으며, TCP/IP 모델과 같은 다른 계층형 모델에도 동일한 이점이 적용될 수 있다는 점을 강조하고 싶습니다. OSI 모델을 비롯한 모든 네트워킹 모델의 핵심 목적은 서로 다른 제조사의 네트워크가 원활하게 상호 운용될 수 있도록 하는 것입니다.

다음은 OSI 계층 모델을 사용함으로써 얻을 수 있는 가장 중요한 이점 중 일부를 간략하게 정리한 목록입니다.

- OSI 모델은 네트워크 통신 프로세스를 더 작고 간단한 구성 요소로 나누어 구성 요소 개발, 설계 및 문제 해결을 지원합니다.
- 이는 네트워크 구성 요소의 표준화를 통해 여러 공급업체의 개발을 가능하게 합니다.
- 이는 모델의 각 계층에서 발생하는 특정 기능을 정의함으로써 산업 표준화를 촉진합니다.
- 이를 통해 다양한 유형의 네트워크 하드웨어와 소프트웨어가 통신할 수 있습니다.
- 이는 한 계층의 변경 사항이 다른 계층에 영향을 미치는 것을 방지하여 개발을 용이하게 하고 애플리케이션 프로그래밍을 훨씬 쉽게 만들어줍니다.

## OSI 참조 모델

OSI 규격의 가장 중요한 기능 중 하나는 유닉스/리눅스, 윈도우 또는 macOS 기반 인지 여부에 관계없이 서로 다른 호스트 간의 데이터 전송을 지원하는 것입니다.

하지만 OSI 모델은 물리적 모델이 아니라, 애플리케이션 개발자들이 네트워크에서 실행되는 애플리케이션을 만들고 구현하는 데 사용하는 개념적이고 포괄적이면서도 유동적인 지침 체계라는 점을 명심해야 합니다. 또한 네트워킹 표준, 장치 및 인터넷 네트워킹 체계를 만들고 구현하기 위한 프레임워크를 제공합니다. OSI 모델은 7개의 계층으로 구성됩니다.

- 응용 프로그램(7계층)
- 프레젠테이션(레이어 6)
- 세션(레이어 5)
- 전송(4계층)
- 네트워크(3계층)
- 데이터 링크(레이어 2)

- 물리적(1계층)

그림 2.1은 OSI 모델의 각 계층에서 발생하는 기능을 요약한 것입니다.

이 점을 염두에 두고, 이제 각 단계에서 어떤 일이 일어나는지 자세히 살펴보겠습니다.



어떤 사람들은 7층 피자를 기억하기 위해 "소시지 피자를 버리지 마세요(Please Do Not Throw Sausage Pizza Away)"라는 암기법을 사용하기도 합니다 (1층부터 시작해서 7층까지). 제가 지어낸 게 아니에요!

Application	<ul style="list-style-type: none"> <li>• File, print, message, database, and application services</li> </ul>
Presentation	<ul style="list-style-type: none"> <li>• Data encryption, compression, and translation services</li> </ul>
Session	<ul style="list-style-type: none"> <li>• Dialog control</li> </ul>
Transport	<ul style="list-style-type: none"> <li>• End-to-end connection</li> </ul>
Network	<ul style="list-style-type: none"> <li>• Routing</li> </ul>
Data Link	<ul style="list-style-type: none"> <li>• Framing</li> </ul>
Physical	<ul style="list-style-type: none"> <li>• Physical topology</li> </ul>

그림 2.1 레이어 기능

OSI의 7개 계층은 두 그룹으로 나뉩니다. 상위 3개 계층은 호스트 시스템 내에서 실행되는 애플리케이션들이 서로 그리고 최종 사용자와 통신하는 방식을 정의하는 규칙을 담고 있습니다. 하위 4개 계층은 실제 데이터가 종단 간 전송되는 방식을 정의합니다. 그림 2.2는 상위 3개 계층과 그 기능을 보여주고, 그림 2.3은 하위 4개 계층과 그 기능을 보여줍니다.

Application	<ul style="list-style-type: none"> <li>• Provides a user interface</li> </ul>
Presentation	<ul style="list-style-type: none"> <li>• Presents data</li> <li>• Handles processing such as encryption</li> </ul>
Session	<ul style="list-style-type: none"> <li>• Keeps different applications' data separate</li> </ul>

**그림 2.2** 상부층

Transport	<ul style="list-style-type: none"> <li>• Provides reliable or unreliable delivery</li> <li>• Performs error correction before retransmit</li> </ul>
Network	<ul style="list-style-type: none"> <li>• Provides logical addressing, which routers use for path determination</li> </ul>
Data Link	<ul style="list-style-type: none"> <li>• Combines packets into bytes and bytes into frames</li> <li>• Provides access to media using MAC address</li> <li>• Performs error detection, not correction</li> </ul>
Physical	<ul style="list-style-type: none"> <li>• Moves bits between devices</li> <li>• Specifies voltage, wire speed, and pin-out of cables</li> </ul>

**그림 2.3** 하부층

**그림 2.2를** 보면 실제 사용자가 응용 프로그램 계층에서 컴퓨터와 상호 작용한다는 것을 알 수 있습니다. 또한 상위 계층은 호스트 간 통신을 담당한다는 것도 분명합니다. 상위 계층은 네트워킹이나 네트워크 주소에 대해 아무것도 알지 못한다는 점을 기억하십시오. 이는 하위 네 계층의 책임입니다.

**그림 2.3은** 하위 4개 계층이 물리적 매체, 스위치 및 라우터를 통해 데이터가 전송되는 방식을 정의함을 보여줍니다. 또한 이러한 하위 계층은 전송 호스트에서 목적지 호스트의 애플리케이션으로 데이터 스트림을 재구성하는 방법을 결정합니다.

좋습니다. 그럼 애플리케이션 계층부터 시작해서 스택 아래쪽으로 내려가면서 살펴보겠습니다.

## 애플리케이션 계층

OSI 모델의 응용 계층은 사용자가 실제로 컴퓨터와 통신하거나 상호 작용하는 지점을 나타냅니다. 기술적으로 사용자는 응용 프로그램 프로세스, 인터페이스 또는 응용 프로그램 프로그래밍 인터페이스(API)를 통해 네트워크 스택과 통신합니다. 이러한 인터페이스는 사용 중인 응용 프로그램을 컴퓨터의 운영 체제에 연결합니다. 응용 계층은 통신 파트너의 가용성과 필요한 연결에 필요한 리소스를 선택하고 결정합니다. 또한 파트너 응용 프로그램을 조정하고 데이터 무결성 및 오류 복구 절차에 대한 합의를 형성합니다. 응용 계층은 네트워크 액세스가 곧 필요할 것으로 예상

될 때만 작동합니다. Chrome이나 Firefox를 예로 들어 보겠습니다. 시스템에서 TCP/IP, 네트워크 카드 등과 같은 네트워킹 구성 요소를 모두 제거하더라도 Chrome을 사용하여 로컬 HTML 문서를 문제없이 볼 수 있습니다. 하지만 HTTP를 사용하여 가져와야 하는 HTML 문서를 보거나 FTP 또는 TFTP를 사용하여 파일을 가져오려고 하면 문제가 발생합니다. Chrome이나 Firefox는 이러한 요청에 대해 응용 계층에 액세스하려고 시도하기 때문입니다. 그러니까 애플리케이션 계층은 계층 구조의 일부가 아닌 애플리케이션 프로그램과 그 바로 아래 계층 사이의 인터페이스 역할을 하며, 애플리케이션이 프로토콜 스택을 통해 정보를 전송할 수 있는 방법을 제공합니다. 다시 말해, 브라우저는 애플리케이션 계층에 속해 있지 않고, 원격 리소스를 처리해야 할 때 애플리케이션 계층 프로토콜과 상호 작용하는 것입니다.

애플리케이션 계층은 또한 의도된 통신 파트너의 가용성을 식별하고 확인하며, 요청된 통신에 필요한 충분한 리소스가 있는지 여부를 판단하는 책임도 있습니다.

이러한 작업은 컴퓨터 애플리케이션이 데스크톱 리소스 이상의 것을 필요로 하는 경우가 많기 때문에 중요합니다. 이러한 애플리케이션은 종종 여러 네트워크 애플리케이션의 통신 구성 요소를 통합합니다. 대표적인 예로는 파일 전송, 이메일, 원격 액세스, 네트워크 관리 활동, 인쇄 및 정보 검색과 같은 클라이언트-서버 프로세스가 있습니다. 많은 네트워크 애플리케이션이 기업 네트워크를 통한 통신 서비스를 제공하지만, 현재와 미래의 인터넷 네트워킹을 위해서는 현재의 물리적 네트워킹의 한계를 뛰어넘어야 할 필요성이 빠르게 발전하고 있습니다.



응용 프로그램 계층은 응용 프로그램들 간의 인터페이스 역할을 한다는 것을 기억하는 것이 중요합니다. 예를 들어, 마이크로소프트 워드는 응용 프로그램 계층에 직접 상주하는 것이 아니라, 응용 프로그램 계층 프로토콜과 상호 작용하는 것입니다. 나중에 [6장](#) "인터넷 프로토콜 소개"에서 FTP와 TFTP처럼 실제로 응용 프로그램 계층에 상주하는 주요 프로그램이나 프로세스에 대해 자세히 설명하겠습니다.

---

## 프레젠테이션 레이어

프레젠테이션 계층은 그 목적에서 이름을 따왔습니다. 애플리케이션 계층에 데이터를 표시하고 데이터 변환 및 코드 서식 지정을 담당합니다.

데이터 전송의 효율성을 높이는 한 가지 방법은 전송 전에 데이터를 표준 형식으로 변환하는 것입니다. 컴퓨터는 이러한 일반화된 형식의 데이터를 수신한 후, 읽기 쉽도록 원래 형식으로 다시 변환합니다. 예를 들어, EBCDIC를 ASCII로 변환하는 식입니다. 프레젠테이션 계층은 이러한 변환 서비스를 제공함으로써 한 시스템의 애플리케이션 계층에서 전송된 데이터를 다른 시스템의 애플리케이션 계층에서 읽고 이해할 수 있도록 보장합니다.

OSI 모델에는 표준 데이터의 형식을 정의하는 프로토콜 표준이 있습니다. 데이터 압축, 압축 해제, 암호화 및 복호화와 같은 작업은 모두 이 계층과 관련이 있습니다. 일부 표현 계층 표준은 멀티미디어 작업에도 사용됩니다.

## 세션 레이어

세션 계층은 프레젠테이션 계층 엔티티 간의 세션을 설정, 관리 및 해제하는 역할을 담당합니다. 또한 장치 또는 노드 간의 대화 제어를 제공합니다. 세션 계층은 시스템 간의 통신을 조정하고 세 가지 모드(단방향(*simplex*), 양방향(단일 방향만 가능, *half-duplex*), 양방향(*full-duplex*))를 제공하여 통신을 체계화 합니다. 요약하자면, 세션 계층은 기본적으로 애플리케이션의 데이터를 다른 애플리케이션의 데이터와 분리하여 관리합니다. 예를 들어, 세션 계층 덕분에 데스크톱에서 여러 웹 브라우저 세션을 동시에 실행할 수 있습니다.

## 전송 계층

전송 계층은 데이터를 분할하고 재조립하여 데이터 스트림을 생성합니다. 전송 계층에 위치한 서비스는 상위 계층 애플리케이션의 데이터를 처리하고 이를 동일한 데이터 스트림으로 통합합니다. 이러한 서비스는 종단 간 데이터 전송 서비스를 제공하며, 인터넷워크 상에서 송신 호스트와 수신 호스트 간의 논리적 연결을 설정할 수 있습니다.

전송 계층은 상위 계층 애플리케이션의 다중화, 가상 연결 설정 및 가상 회로 해제 메커니즘을 제공하는 역할을 합니다. 또한 네트워크에 의존하는 다양한 세부 정보를 상위 계층으로부터 숨겨 데이터 전송을 원활하게 합니다.

**6장**에서 전송 제어 프로토콜(TCP)과 사용자 데이터그램 프로토콜(UDP)에 대해 자세히 다루겠지만, 이미 이 두 프로토콜에 익숙하다면 둘 다 전송 계층에서 작동한다는 것을 알고 있을 것입니다. 또한 TCP는 신뢰할 수 있는 서비스이고 UDP는 그렇지 않다는 것도 알고 있을 것입니다. 이 두 프로토콜은 애플리케이션 개발자에게 TCP/IP 프로토콜을 사용할 때 선택할 수 있는 더 많은 옵션을 제공합니다.



신뢰할 수 있는 네트워킹이라는 용어는 전송 계층과 관련이 있으며, 승인, 순서 지정 및 흐름 제어가 사용됨을 의미합니다.

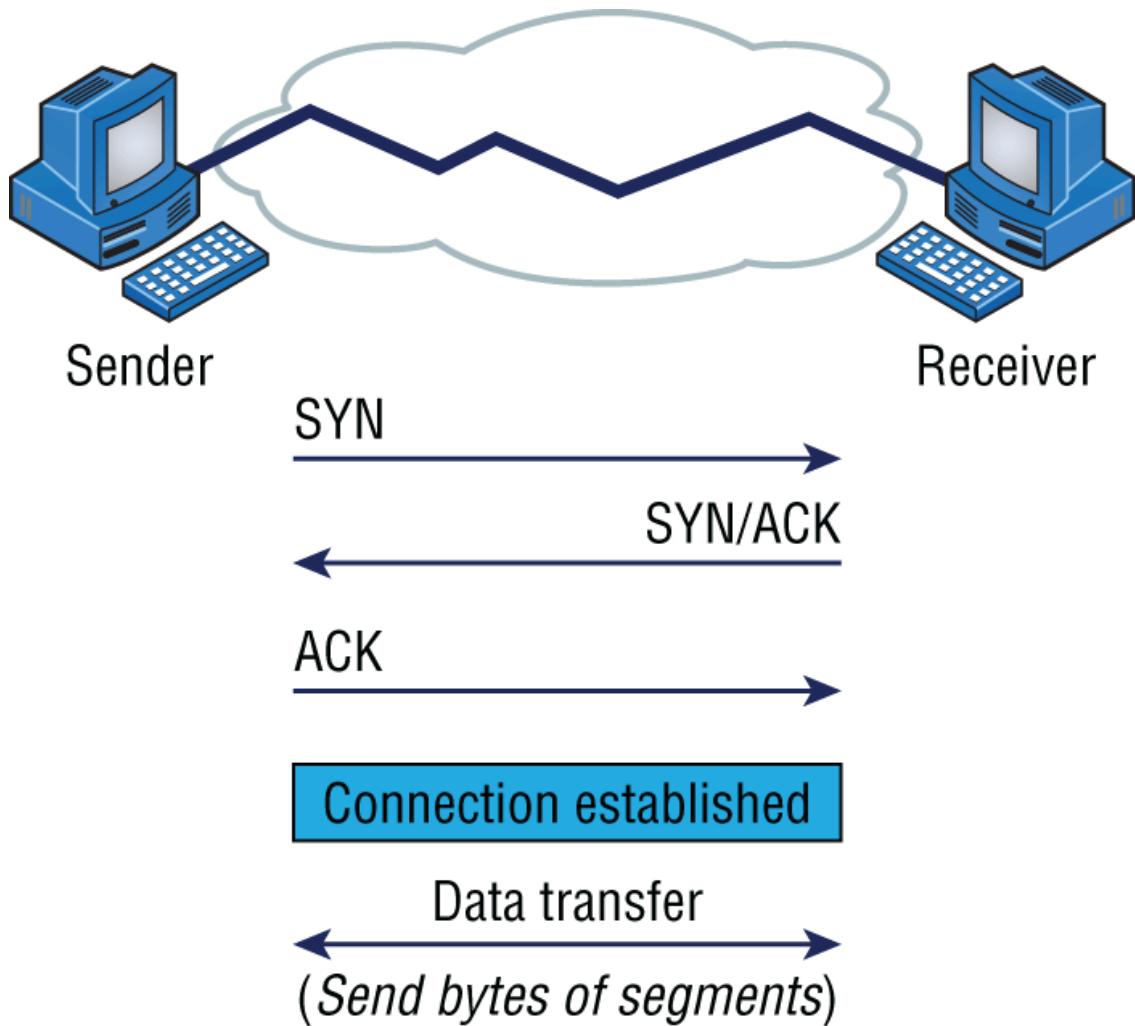
---

전송 계층은 비연결형 또는 연결형일 수 있지만, 특히 연결형 전송 계층 부분을 제대로 이해하는 것이 중요합니다. 이제 전송 계층의 연결형(신뢰할 수 있는) 프로토콜에 대해 자세히 살펴보겠습니다.

### 연결 지향적 커뮤니케이션

송신 호스트가 모델을 통해 세그먼트를 전송하기 전에, 송신측 TCP 프로세스는 수신측 TCP 프로세스와 연결을 설정하기 위해 통신합니다. 이렇게 생성된 연결을 가상 회로라고 합니다. 이러한 유형의 통신을 연결 지향형 통신이라고 합니다. 이 초기 핸드셰이크 과정에서 두 TCP 프로세스는 각 수신측 TCP가 확인 응답을 보내기 전에 양방향으로 전송할 정보의 양에 대해 합의합니다. 모든 것이 사전에 합의되면 신뢰할 수 있는 통신을 위한 경로가 마련됩니다.

**그림 2.4**는 송신 시스템과 수신 시스템 간에 발생하는 일반적인 안정적인 세션을 보여줍니다. 두 호스트의 애플리케이션 프로그램은 모두 연결이 시작될 것임을 각자의 운영 체제에 알리는 것으로 시작합니다.



**그림 2.4** 연결 지향 세션 설정

두 운영 체제는 네트워크를 통해 메시지를 주고받으며 전송이 승인되었고 양측 모두 전송 준비가 완료되었음을 확인합니다. 이러한 모든 필수 동기화가 완료되면 연결이 완전히 설정되고 데이터 전송이 시작됩니다. 이러한 가상 회로 설정 과정을 오버헤드라고 합니다.

호스트 간 정보 전송이 진행되는 동안 두 시스템은 주기적으로 서로 통신하며 프로토콜 소프트웨어를 통해 모든 것이 정상적으로 진행되고 데이터가 제대로 수신되고 있는지 확인합니다.

**그림 2.4**에 나타낸 연결 지향 세션, 즉 TCP 3방향 핸드셰이크의 단계를 요약해 보겠습니다.

1. 첫 번째 "연결 계약" 부분은 동기화 요청입니다.
2. 다음 단계에서는 요청을 확인하고 호스트 간 연결 매개변수(규칙)를 설정합니다. 이 단계에서는 수신자의 순서도 동기화하여 양방향 연결이 이루어지도록 요청합니다.

- 마지막 부분은 연결 승인 메시지입니다. 이 메시지는 수신 호스트에게 연결 계약이 수락되었고 연결이 설정되었음을 알립니다. 이제 데이터 전송을 시작할 수 있습니다.



제가 이 연결 설정에 대해 자세히 설명드린 이유는 여러분이 작동 방식을 명확하게 이해하실 수 있도록 하기 위함입니다. 이 전체 과정을 제가 이미 언급했던 "3방향 핸드셰이크"라고 부르며, SYN, SYN/ACK, ACK 또는 동기화, 동기화 확인, 확인이라고도 합니다.

간단해 보이지만, 실제로는 항상 순조롭게 진행되는 것은 아닙니다. 고속 컴퓨터가 네트워크 처리 속도보다 훨씬 빠른 속도로 데이터 트래픽을 생성하면 전송 중에 혼잡이 발생할 수 있습니다. 여러 대의 컴퓨터가 동시에 하나의 게이트웨이를 통해 또는 특정 목적지로 데이터그램을 전송하는 경우에도 혼잡이 발생할 수 있습니다. 후자의 경우, 특정 원인이 없더라도 게이트웨이나 목적지가 혼잡해질 수 있습니다. 어떤 경우든, 문제는 고속도로 병목 현상과 같습니다. 용량이 부족한데 비해 교통량이 너무 많은 것이죠. 대개 문제는 특정 차량 한 대가 아니라, 그 특정 경로에 너무 많은 차량이 있는 것입니다.

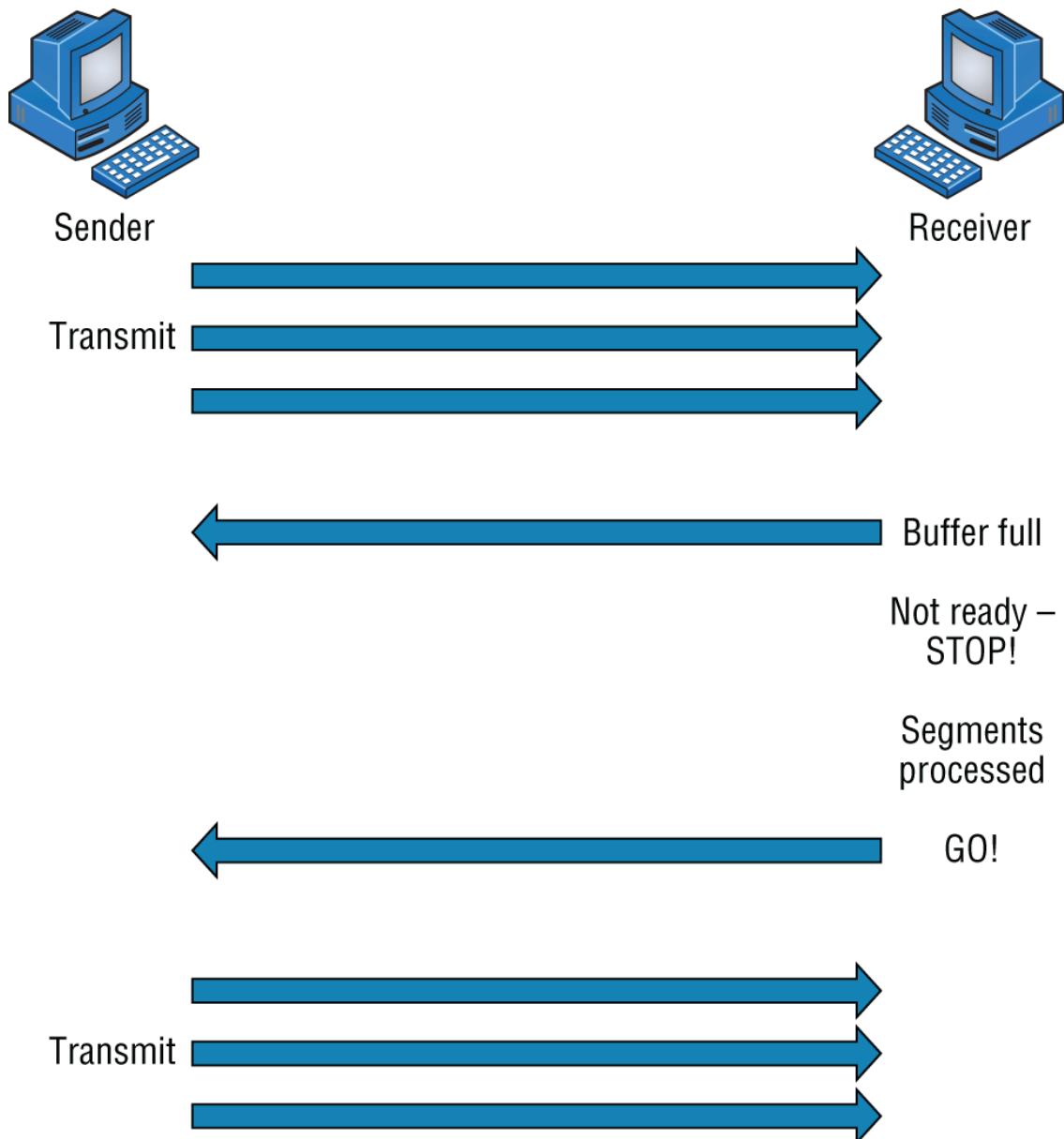
## 유량 제어

전송 계층에서는 흐름 제어를 유지하고 사용자가 시스템 간에 안정적인 데이터 전송을 요청할 수 있도록 함으로써 데이터 무결성을 보장합니다. 흐름 제어는 수신자가 송신자가 보내는 데이터 양을 제어할 수 있는 수단을 제공합니다. 이를 통해 연결의 한쪽에서 데이터를 보내는 호스트가 수신 호스트의 버퍼를 오버플로우시키는 것을 방지합니다. 버퍼 오버플로우는 데이터 손실을 초래할 수 있는 상황입니다. 안정적인 데이터 전송은 시스템 간에 연결 지향 통신 세션을 사용하며, 관련 프로토콜은 다음과 같은 사항을 보장합니다.

- 전송된 데이터는 수신 즉시 발신자에게 수신 확인 메시지가 전송됩니다.
- 확인 응답이 없는 부분은 모두 재전송됩니다.
- 세그먼트들은 목적지에 도착하면 원래 순서대로 다시 정렬됩니다.
- 혼잡, 과부하 및 데이터 손실을 방지하기 위해 관리 가능한 데이터 흐름이 유지 됩니다.

자, 그렇다면 기계가 처리할 수 없을 정도로 빠른 속도로 데이터그램이 쏟아져 들어 올 때 무슨 일이 일어날까요? 기계는 버퍼라고 불리는 메모리 영역에 데이터그램을 저장합니다. 하지만 이 버퍼링 방식은 데이터그램이 짧은 시간 동안 집중적으로 들어올 때만 효과가 있습니다. 그렇지 않고 데이터그램 폭주가 계속되면, 기기의 메모리는 결국 한계에 도달하여 더 이상 처리할 수 없게 되고, 마치 댐이 무너지듯 추가로 들어오는 모든 데이터그램을 버려버리게 됩니다!

이 상황은 꽤 심각해 보이지만, 실제로 네트워크 플러드 제어의 전송 기능이 매우 효과적으로 작동하기 때문에 그렇지 않습니다. 어떻게 그럴 수 있을까요? 단순히 리소스를 낭비하고 데이터 손실을 허용하는 대신, 전송 계층은 [그림 2.5](#)에서 보는 것처럼 플러드의 발신자 또는 소스에게 "준비되지 않음" 표시를 보낼 수 있습니다.



**그림 2.5** 유량 제어 기능이 있는 전송 세그먼트

이 메커니즘은 신호등과 유사하게 작동하여, 송신 장치가 과부하된 상대방 장치로 세그먼트 트래픽 전송을 중단하도록 신호를 보냅니다. 상대방 장치의 수신 장치가 메모리 저장소(버퍼)에 쌓인 세그먼트를 처리한 후, "준비 완료" 전송 표시기를 보냅니다. 나머지 데이터그램 전송을 기다리는 장치가 이 "시작" 표시기를 수신하면 전송을 재개합니다.

기본적이고 신뢰할 수 있는 연결 지향형 데이터 전송 과정에서 데이터그램은 전송된 순서와 정확히 동일한 순서로 수신 호스트에 전달됩니다. 따라서 전송 과정에서 데이터 세그먼트가 손실, 중복 또는 손상된 경우 오류 알림이 전송됩니다. 이 오류는 수신 호스트가 모든 데이터 세그먼트를 올바른 순서로 수신했음을 확인하는 과정을 통해 수정됩니다.

요약하자면, 서비스는 다음과 같은 특징을 가질 경우 연결 지향적 서비스로 간주됩니다.

- 가상 회로가 설정됩니다(예: 3자 핸드셰이크).
- 시퀀싱 기법을 사용합니다.
- 감사의 글을 사용합니다.
- 흐름 제어를 사용합니다.

## 윈도우

이상적으로는 데이터 처리가 빠르고 효율적으로 이루어져야 합니다. 송신 기기가 각 데이터 세그먼트를 전송한 후 수신 기기로부터 확인 응답을 기다려야 한다면 당연히 속도가 느려질 것입니다. 하지만 송신 기기가 데이터 세그먼트를 전송한 후 수신 기기로부터 확인 응답을 처리하기 전까지 시간적 여유가 있기 때문에, 송신 기기는 이 시간을 활용하여 더 많은 데이터를 전송할 수 있습니다. 송신 기기가 확인 응답을 받지 않고 전송할 수 있는 데이터 세그먼트의 양(바이트 단위)을 윈도우라고 합니다.



윈도우는 처리되지 않은, 승인되지 않은 데이터 세그먼트의 양을 제어하는 데 사용됩니다.

---

윈도우 크기가 한쪽 끝에서 다른 쪽 끝으로 전송되는 정보의 양을 제어한다는 점을 이해하는 것이 중요합니다. 일부 프로토콜은 패킷 수를 관찰하여 정보량을 측정하지만, TCP/IP는 바이트 수를 계산하여 측정합니다.

그림 2.6은 두 가지 윈도우 크기(하나는 1, 다른 하나는 3)를 보여줍니다. 이 단순화된 예에서는 송신 기기와 수신 기기 모두 워크스테이션입니다.

윈도우 크기를 1로 설정하면 송신 기기는 각 데이터 세그먼트를 전송할 때마다 수신 확인을 기다린 후 다음 세그먼트를 전송합니다. 윈도우 크기를 3으로 설정하면 송신 기기는 수신 확인을 받기 전에 최대 세 개의 데이터 세그먼트를 전송할 수 있습니다. 실제로 윈도우 크기는 한 번에 전송할 수 있는 바이트 수를 제한하는 역할을 합니다.



수신 호스트가 승인해야 할 모든 세그먼트를 수신하지 못한 경우, 윈도우 크기를 줄임으로써 통신 세션을 개선할 수 있습니다.

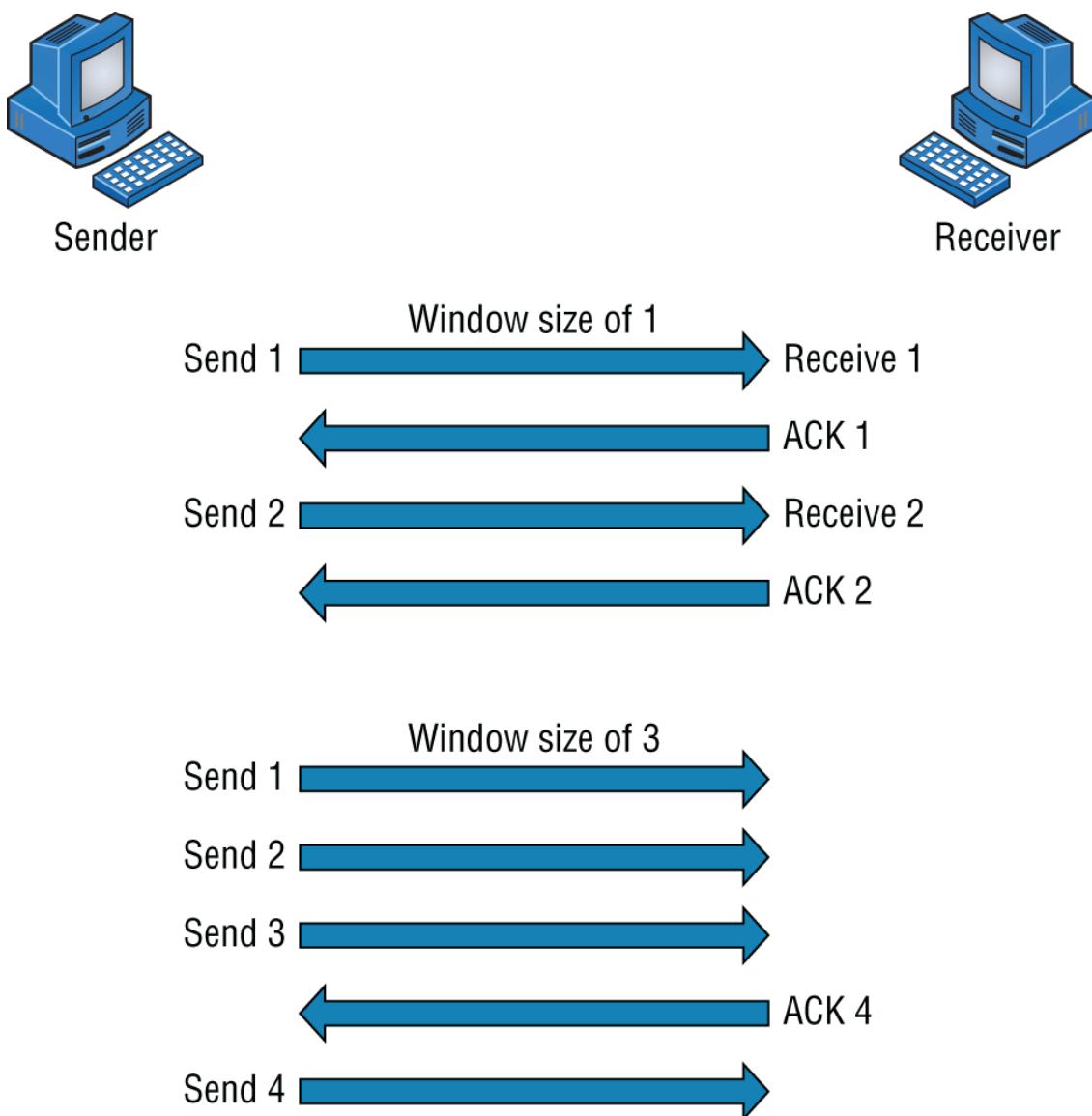


그림 2.6 윈도우링

### 감사의 말씀

신뢰할 수 있는 데이터 전송은 완벽하게 작동하는 데이터 링크를 통해 한 기기에서 다른 기기로 전송되는 데이터 스트림의 무결성을 보장합니다. 이는 데이터가 중복되거나 손실되지 않도록 보장합니다. 이러한 무결성은 '수신 확인 및 재전송'이라는 기술을 통해 달성됩니다. 이 기술은 수신 기기가 데이터를 수신하면 송신 기기로 확인 메시지를 전송하여 통신하는 것을 요구합니다. 송신 기기는 전송하는 각 세그먼트를 기록하고 다음 세그먼트를 전송하기 전에 수신 확인 메시지를 기다립니다.

세그먼트를 전송할 때 송신 기기는 타이머를 시작하고, 수신 측에서 확인 메시지가 반환되기 전에 타이머가 만료되면 재전송합니다.

그림 2.7에서 송신 장치는 세그먼트 1, 2, 3을 전송합니다. 수신 노드는 세그먼트 4를 요청하여 수신을 확인합니다. 수신 확인을 받으면 송신 장치는 세그먼트 4, 5, 6을 전송합니다. 세그먼트 5가 목적지에 도달하지 못하면 수신 노드는 해당 세그먼트의 재전송을 요청하여 이를 확인합니다. 송신 장치는 손실된 세그먼트를 재전송하고 수신 확인을 기다립니다. 수신 확인을 받아야만 세그먼트 7의 전송으로 넘어갈 수 있습니다.

전송 계층은 연결 지향 서비스를 반드시 사용해야 하는 것은 아닙니다. 그 선택은 애플리케이션 개발자에게 달려 있습니다. 연결 지향 서비스, 즉 가상 회로를 생성한 경우에는 TCP를 사용한다고 볼 수 있습니다. 가상 회로를 설정하지 않은 경우에는 UDP를 사용하며, 이는 비연결형 서비스로 간주됩니다.

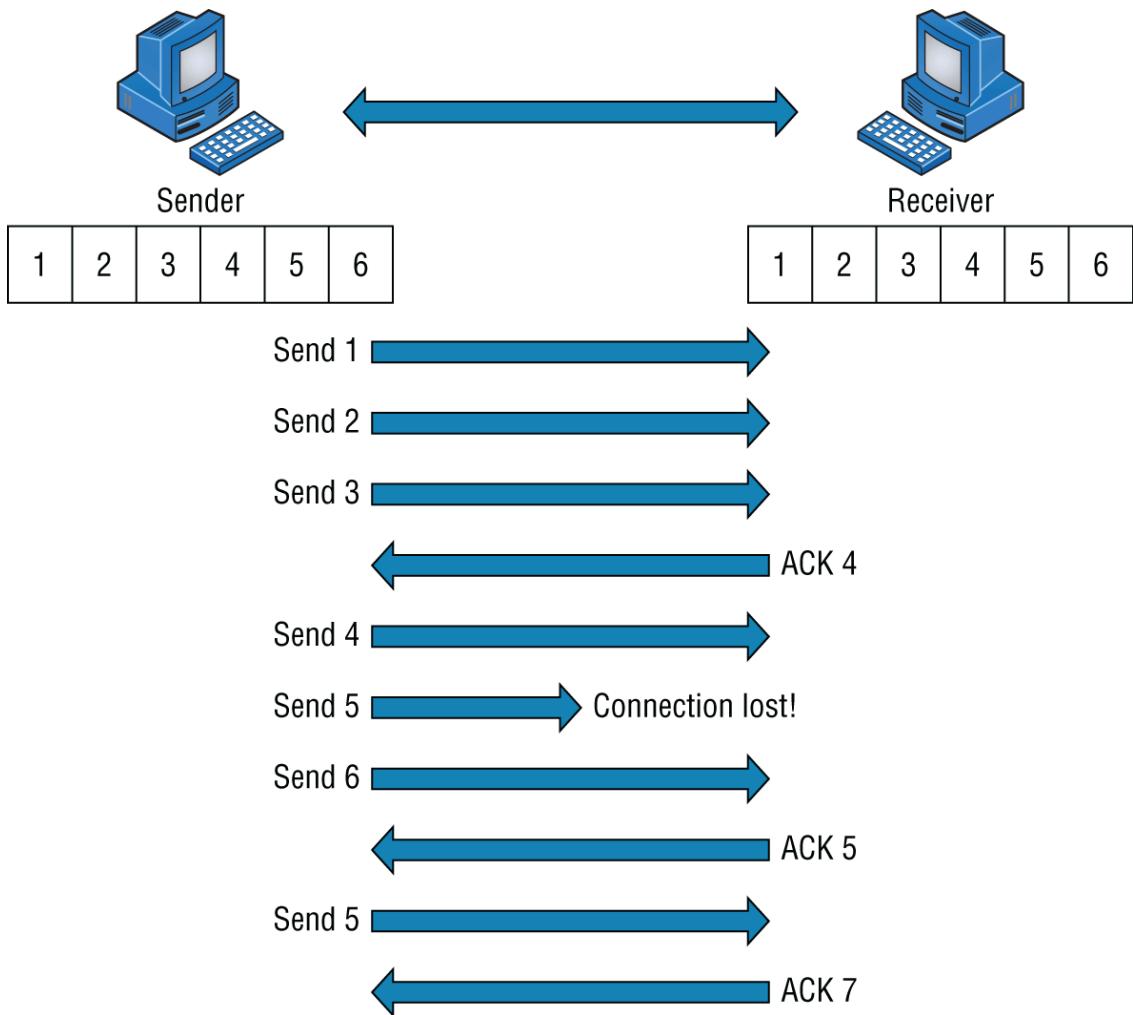


그림 2.7 전송 계층의 안정적인 전달



전송 제어 프로토콜(TCP)과 사용자 데이터그램 프로토콜(UDP)은 전송 계층에서 작동하는 프로토콜이며, 6장에서 자세히 다룰 것입니다.

## 네트워크 계층

네트워크 계층은 논리적 장치 주소 지정을 관리하고, 네트워크 상의 장치 위치를 추적하며, 데이터를 이동하는 최적의 경로를 결정합니다. 즉, 네트워크 계층은 로컬로 연결되지 않은 장치 간에도 트래픽을 전송해야 합니다. 라우터는 네트워크 계층에서 정의되는 3계층 장치로, 인터넷워크 내에서 라우팅 서비스를 제공합니다.

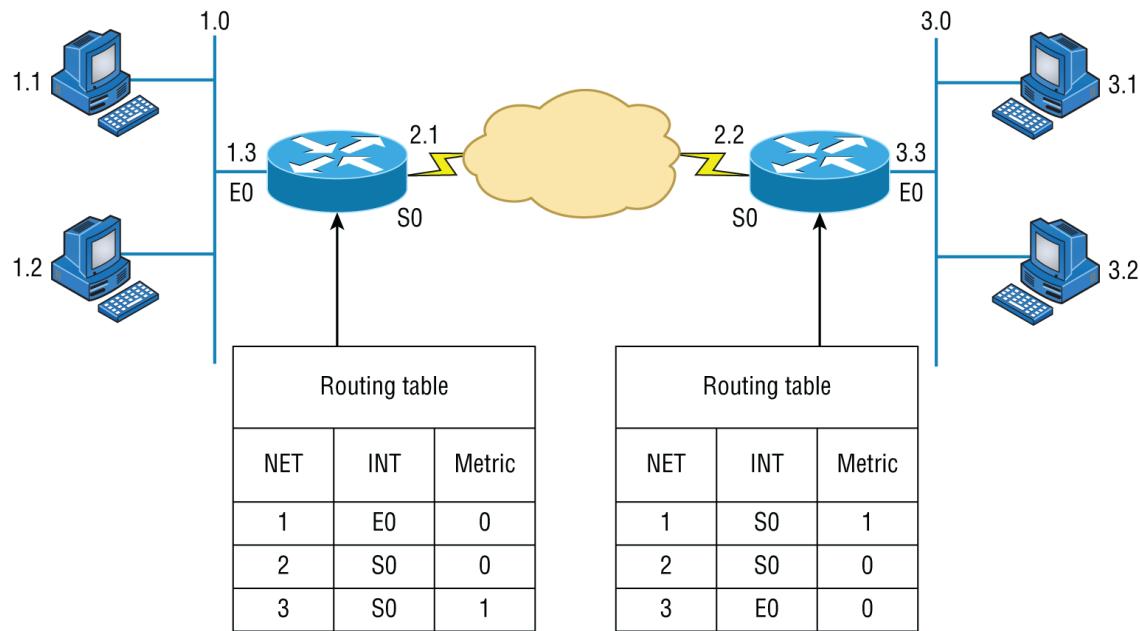
작동 방식은 다음과 같습니다. 먼저 라우터 인터페이스에서 패킷을 수신하면 목적지 IP 주소를 확인합니다. 패킷이 해당 라우터로 향하는 것이 아니면 라우터는 라우

팅 테이블에서 목적지 네트워크 주소를 조회합니다. 라우터가 적절한 종료 인터페이스를 선택하면 패킷을 해당 인터페이스로 보내 프레임을 생성한 후 로컬 네트워크로 전송합니다. 라우터가 라우팅 테이블에서 패킷의 목적지 네트워크에 대한 항목을 찾을 수 없으면 패킷을 폐기합니다.

네트워크 계층에서는 두 가지 유형의 패킷이 사용됩니다.

- **데이터 패킷** 은 인터넷 네트워크를 통해 사용자 데이터를 전송하는 데 사용됩니다. 데이터 트래픽을 지원하는 데 사용되는 프로토콜을 라우팅 프로토콜이라고 합니다. 라우팅 프로토콜의 두 가지 예로는 인터넷 프로토콜(IP)과 인터넷 프로토콜 버전 6(IPv6)이 있으며, 이에 대해서는 [7장 "IP 주소 지정"](#) 에서 자세히 알아보겠습니다.
- **경로 업데이트** 패킷은 인터넷워크 내 모든 라우터에 연결된 네트워크에 대한 정보를 인접 라우터에 업데이트하는 데 사용됩니다. 경로 업데이트 패킷을 전송하는 프로토콜을 라우팅 프로토콜이라고 하며, 일반적인 프로토콜로는 RIP(Routing Information Protocol), RIPv2, EIGRP(Enhanced Interior Gateway Routing Protocol), OSPF(Open Shortest Path First) 등이 있습니다. 경로 업데이트 패킷은 각 라우터의 라우팅 테이블을 구축하고 유지 관리하는 데 사용됩니다.

[그림 2.8](#)은 라우팅 테이블을 보여줍니다. 라우터에서 사용하는 라우팅 테이블에는 다음과 같은 정보가 포함됩니다.



**그림 2.8** 라우터에서 사용되는 라우팅 테이블

- **네트워크 주소** 는 프로토콜별로 지정된 네트워크 주소입니다. 라우터는 각 라우팅 프로토콜에 대한 라우팅 테이블을 유지해야 하는데, 각 라우팅 프로토콜은 IP 및 IPv6와 같은 다양한 주소 체계를 포함하는 네트워크를 관리해야 하기 때문입니다. 마치 특정 거리에 사는 주민들이 사용하는 각기 다른 언어로 된 거리 표지판과 같다고 생각하면 됩니다. 만약 '캣(Cat)'이라는 거리에 미국인, 스페인인, 프랑스인이 살고 있다면, 표지판에는 '캣/가토/챗(Cat/Gato/Chat)'이라고 쓰여 있을 것입니다.
- **인터페이스란** 패킷이 특정 네트워크로 향할 때 거치게 될 출구 인터페이스를 의미합니다.
- **메트릭** 값은 원격 네트워크까지의 거리를 나타냅니다. 다양한 라우팅 프로토콜은 이 거리를 계산하는 방식이 서로 다릅니다. 라우팅 프로토콜에 대해서는 [9장 "IP 라우팅 소개"](#)에서 자세히 다루겠습니다. 지금은 RIP와 같은 일부 라우팅 프로토콜이 흡 카운트(패킷이 원격 네트워크로 이동하는 동안 거치는 라우터의 수)라는 것을 사용한다는 점만 알아두시면 됩니다. 다른 라우팅 프로토콜은 대역폭, 회선 지연, 심지어 틱 카운트(1/18초)를 사용하여 라우팅 결정을 내립니다.

라우터는 브로드캐스트 도메인을 분할합니다. 즉, 기본적으로 브로드캐스트는 라우터를 통과하지 않습니다. 이는 네트워크 트래픽을 줄여주기 때문에 바람직한 기능입니다. 라우터는 또한 충돌 도메인을 분할하지만, 이는 레이어 2(데이터 링크 계층) 스위치를 사용해서도 구현할 수 있습니다.



### 브로드캐스트 도메인과 충돌 도메인은 5장에서 자세히 다

룰 것입니다. 지금은 라우터가 브로드캐스트 도메인을 분할하고 스위치가 충돌 도메인을 분할한다는 점만 기억해 두세요.

라우터의 각 인터페이스는 별도의 네트워크를 나타내므로 고유한 네트워크 식별 번호를 할당해야 하며, 해당 라우터에 연결된 네트워크의 모든 호스트는 동일한 네트워크 번호를 사용해야 합니다.

그림 2.9는 라우터가 인터넷 네트워크 내에서 어떻게 작동하는지 보여줍니다.

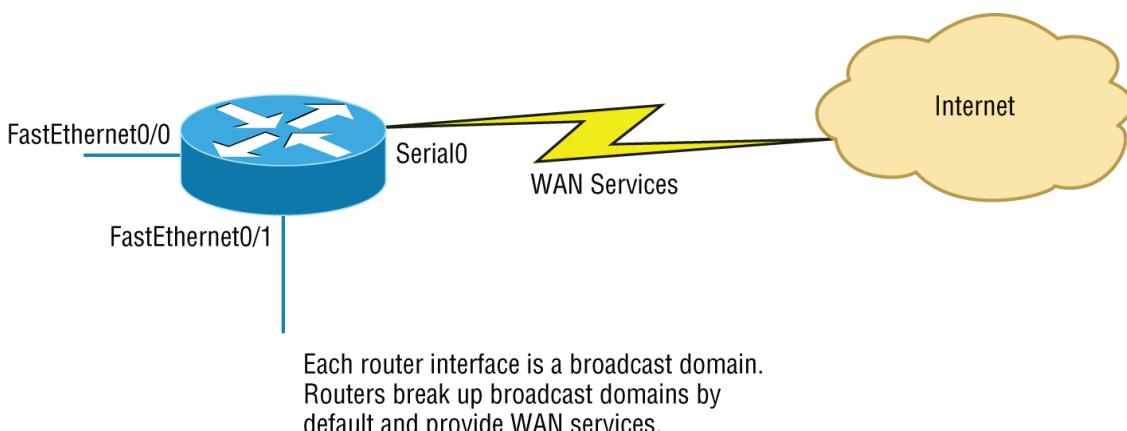


그림 2.9 인터넷 네트워크의 라우터

라우터에 대해 꼭 기억해 두어야 할 몇 가지 핵심 사항은 다음과 같습니다.

- 라우터는 기본적으로 브로드캐스트 또는 멀티캐스트 패킷을 전달하지 않습니다.
- 라우터는 네트워크 계층 헤더의 논리적 주소를 사용하여 패킷을 전달할 다음 흡 라우터를 결정합니다.
- 라우터는 관리자가 생성한 액세스 목록을 사용하여 인터페이스로 들어오거나 나가는 패킷 유형에 대한 보안을 제어할 수 있습니다.
- 라우터는 필요한 경우 레이어 2 브리징 기능을 제공할 수 있으며 동일한 인터페이스를 통해 동시에 라우팅할 수 있습니다.
- 레이어 3 장치(이 경우 라우터)는 가상 LAN(VLAN) 간의 연결을 제공합니다.
- 라우터는 특정 유형의 네트워크 트래픽에 대해 서비스 품질(QoS)을 제공할 수 있습니다.



라우터는 레이어 3 스위치라고도 불립니다. 이 두 용어는 서로 바꿔 쓸 수 있습니다.

## 데이터 링크 계층

데이터 링크 계층은 데이터의 물리적 전송을 제공하고 오류 알림, 네트워크 토폴로지 및 흐름 제어를 처리합니다. 즉, 데이터 링크 계층은 하드웨어(MAC) 주소를 사용하여 LAN에서 메시지가 올바른 장치로 전달되도록 보장하고 네트워크 계층의 메시지를 물리 계층에서 전송할 수 있는 비트로 변환합니다.

데이터 링크 계층은 메시지를 데이터 프레임이라는 조각으로 나누고, 목적지 및 출발지 하드웨어 주소를 포함하는 사용자 지정 헤더를 추가합니다. 이 추가 정보는 마치 아폴로 프로젝트의 달 착륙선에 엔진, 항법 장치 및 기타 도구들이 부착되었던 방식과 유사하게 원래 메시지를 감싸는 캡슐 역할을 합니다. 이러한 다양한 장비들은 비행의 특정 단계에서만 유용했으며, 해당 단계가 완료되면 착륙선에서 분리되어 폐기되었습니다. 이는 네트워크를 통해 데이터가 전송되는 과정을 설명하는 훌륭한 비유입니다.

라우터는 네트워크 계층에서 작동하며 특정 호스트의 위치에는 관심이 없다는 점을 이해하는 것이 중요합니다. 라우터는 네트워크의 위치와 원격 네트워크를 포함한 모든 네트워크에 도달하는 최적의 경로에만 관심을 가집니다. 라우터는 네트워크에 대해 매우 집착적인데, 이 경우에는 집착이 오히려 긍정적인 요소입니다! 데이터 링크 계층은 로컬 네트워크에 있는 각 장치를 고유하게 식별하는 역할을 담당합니다.

호스트가 로컬 네트워크 내의 개별 호스트로 패킷을 전송하거나 라우터 간에 패킷을 전송하기 위해 데이터 링크 계층은 하드웨어 주소 지정을 사용합니다. 라우터 간에 패킷이 전송될 때마다 데이터 링크 계층에서 제어 정보로 프레임화됩니다. 그러나 수신 라우터에서는 해당 정보가 제거되고 원래 패킷만 온전히 남게 됩니다. 이러한 패킷 프레임화 과정은 패킷이 최종적으로 올바른 수신 호스트에 도달할 때까지 각 흡마다 반복됩니다. 중요한 것은 패킷 자체는 전송 경로를 따라 절대 변경되지 않으며, 단지 다양한 미디어 유형으로 제대로 전달되는 데 필요한 제어 정보로 캡슐화될 뿐이라는 점입니다.

그림 2.10은 이더넷과 IEEE(전기전자공학회) 규격을 적용한 데이터 링크 계층을 보여줍니다.

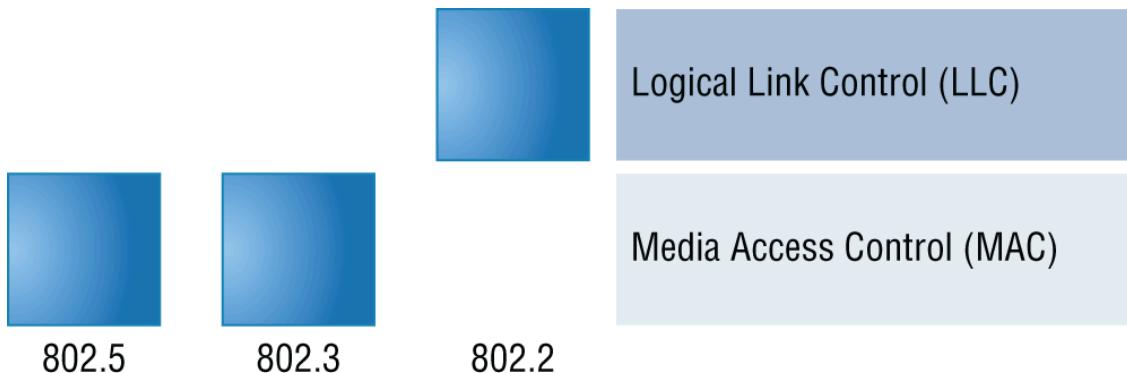


그림 2.10 데이터 링크 계층

자세히 살펴보면 IEEE 802.2 표준이 다른 IEEE 표준과 함께 사용될 뿐만 아니라, 해당 표준에 기능을 추가한다는 것을 알 수 있습니다.

IEEE 이더넷 데이터 링크 계층은 두 개의 하위 계층으로 구성됩니다.

- **MAC(Media Access Control)** 은 패킷이 미디어에 어떻게 배치되는지를 정의합니다. 경쟁형 미디어 액세스는 "선착순" 액세스 방식으로, 모든 사용자가 동일한 대역폭을 공유합니다. 이름에서 알 수 있듯이, 물리적 주소 지정과 논리적 토폴로지가 이 계층에서 정의됩니다. 논리적 토폴로지란 무엇일까요? 물리적 토폴로지를 통과하는 신호 경로입니다. 회선 관리, 오류 알림(수정은 아님), 프레임의 순차적 전달, 그리고 선택적으로 흐름 제어 또한 이 하위 계층에서 사용될 수 있습니다.
- **논리 링크 제어(LLC)** 헤더는 네트워크 계층 프로토콜을 식별하고 이를 캡슐화하는 역할을 하며, 데이터 링크 계층에 프레임 수신 후 패킷을 어떻게 처리해야 하는지 알려줍니다. 작동 방식은 다음과 같습니다. 호스트는 프레임을 수신하고 LLC 헤더를 확인하여 패킷의 목적지(예: 네트워크 계층의 IP 프로토콜)를 파악합니다. LLC는 또한 흐름 제어 및 제어 비트의 순서 지정을 제공할 수 있습니다.

---

## 프로젝트 802

데이터 링크 계층의 주요 구성 요소 중 하나는 IEEE 802 소위원회가 근거리 통신망(LAN) 및 광역 통신망(MAN) 표준에 대해 수행한 연구의 결과물입니다. 이 위원회는 1980년 2월에 회의를 개최했으며, 1980년의 숫자 80과 두 번째 달의 숫자 2를 따서 프로젝트 802라는 이름을 만들었습니다. 802 표준의 명칭은 항상 마침표(.) 뒤에 한 자리 또는 두 자리 숫자가 붙습니다. 이 숫자들은 802 표준 내의 특정 범주를 지정합니다. 이러한 표준은 다음 표에 나열되어 있습니다.

기준	주제
802.1	LAN/MAN 관리(및 미디어 액세스 제어 브리지)
802.2	논리 링크 제어
802.3	CSMA/CD(이더넷)
802.4	토큰 패스 버스
802.5	토큰 패싱 링
802.6	분산 큐 이중 버스(DQDB) 광역 네트워크(MAN)
802.7	광대역 근거리 통신망
802.8	광섬유 LAN 및 MAN
802.9	등시성 LAN
802.10	LAN/MAN 보안
802.11	무선 LAN
802.12	수요 우선 접근 방식

## 기준      주제

802.15 무선 개인 영역 네트워크

802.16 무선 광역 네트워크(WiMAX라고도 함)

802.17 탄력적인 패킷 링

참고로 현재 활성화된 802 표준은 802.1, 802.3, 802.11, 802.15뿐입니다. 나머지 표준들은 폐기되었거나 사용이 중단된 상태입니다.

이 사이드바에서 기억해야 할 것은 802.3은 이더넷과 관련된 모든 것을 나타내고, 802.11은 무선과 관련된 모든 것을 나타낸다는 점입니다.

---

## 물리 계층

드디어 바닥에 도달했습니다. 물론 나쁜 의미는 아닙니다. 이제 물리 계층에 이르렀는데, 이 계층은 두 가지 중요한 역할을 합니다. 바로 비트를 송수신하는 것입니다. 비트는 1 또는 0의 값만 가지는데, 이는 숫자로 표현되는 모스 부호와 같습니다. 물리 계층은 다양한 종류의 실제 통신 매체와 직접 통신합니다. 매체 종류에 따라 비트 값을 표현하는 방식이 다릅니다. 어떤 매체는 오디오 톤을 사용하고, 어떤 매체는 상태 전환, 즉 전압이 높음에서 낮음으로, 낮음에서 높음으로 변하는 것을 이용합니다. 각 매체 유형에는 사용해야 할 적절한 비트 패턴, 데이터가 매체 신호로 인코딩되는 방식, 그리고 물리적 매체의 연결 인터페이스의 다양한 특성을 설명하는 특정 프로토콜이 필요합니다.

물리 계층은 최종 시스템 간의 물리적 연결을 활성화, 유지 및 비활성화하기 위한 전기적, 기계적, 절차적 및 기능적 요구 사항을 명시합니다. 또한 이 계층은 데이터 단말 장비(DTE) 간의 인터페이스를 식별하는 곳이기도 합니다. 데이터 통신 장비(DCE)가 있습니다. (일부 구직 전화 회사 직원들은 여전히 DCE를 데이터 회로 종단 장비라고 부릅니다.) DCE는 일반적으로 고객에게 설치되며, DTE는 DCE에 연결된 장치입니다. DTE에서 이용 가능한 서비스는 대부분 모뎀 또는 채널 서비스 장치//데이터 서비스 장치(CSU/DSU)인 DCE 장치를 통해 접속됩니다.

물리 계층의 커넥터와 다양한 물리적 토폴로지는 표준에 의해 정의되어 서로 다른 시스템 간의 통신을 가능하게 합니다.

마지막으로, 물리 계층은 전송 매체의 레이아웃, 즉 토폴로지를 지정합니다. 물리적 토폴로지는 케이블이 물리적으로 배치되는 방식을 설명하며, 앞서 "데이터 링크 계층" 섹션에서 다른 논리적 토폴로지와는 대조적입니다. 다양한 물리적 토폴로지에는 버스, 스타, 링, 메시 등이 있으며, [1장 "네트워크 소개"](#)에서 설명했습니다.

## 캡슐화 소개

호스트가 네트워크를 통해 다른 장치로 데이터를 전송할 때, 데이터는 캡슐화 과정을 거칩니다. 즉, OSI 모델의 각 계층에서 프로토콜 정보로 감싸집니다. 각 계층은 수신 장치의 해당 계층과만 통신합니다.

통신 및 정보 교환을 위해 각 계층은 프로토콜 데이터 단위(PDU)를 사용합니다. PDU는 모델의 각 계층에서 데이터에 첨부된 제어 정보를 담고 있습니다. 일반적으로 데이터 필드 앞의 헤더에 첨부되지만, 데이터 필드의 끝부분(트레일러)에 위치할 수도 있습니다.

송신 장치에서 데이터 캡슐화 방식은 다음과 같이 작동합니다.

1. 사용자 정보는 네트워크 전송을 위한 데이터로 변환됩니다.
2. 데이터는 세그먼트로 변환되고, 송신 호스트와 수신 호스트 간에 안정적인 연결이 설정됩니다.
3. 데이터 세그먼트는 패킷 또는 데이터그램으로 변환되고, 각 패킷이 인터넷 네트워크를 통해 라우팅될 수 있도록 헤더에 논리적 주소가 추가됩니다. 패킷은 하나의 데이터 세그먼트를 전달합니다.
4. 패킷 또는 데이터그램은 로컬 네트워크에서 전송하기 위해 프레임으로 변환됩니다. 하드웨어(이더넷) 주소는 로컬 네트워크 세그먼트에서 호스트를 고유하게 식별하는 데 사용됩니다. 프레임은 패킷을 전달합니다.
5. 프레임은 비트로 변환되고, 디지털 인코딩 및 클럭킹 방식이 사용됩니다.

[그림 2.11은](#) 전송 호스트에서 사용자 데이터가 어떻게 캡슐화되는지를 보여줍니다.

다음 장으로 넘어가기 전에, 프레임이 비트로 변환되고 인코딩 및 클럭킹 방식이 사용될 때 5단계는 어떻게 작동하는지 살펴보겠습니다. 이것을 변조 기법이라고 하며, 이 장을 이 논의로 마무리하겠습니다.

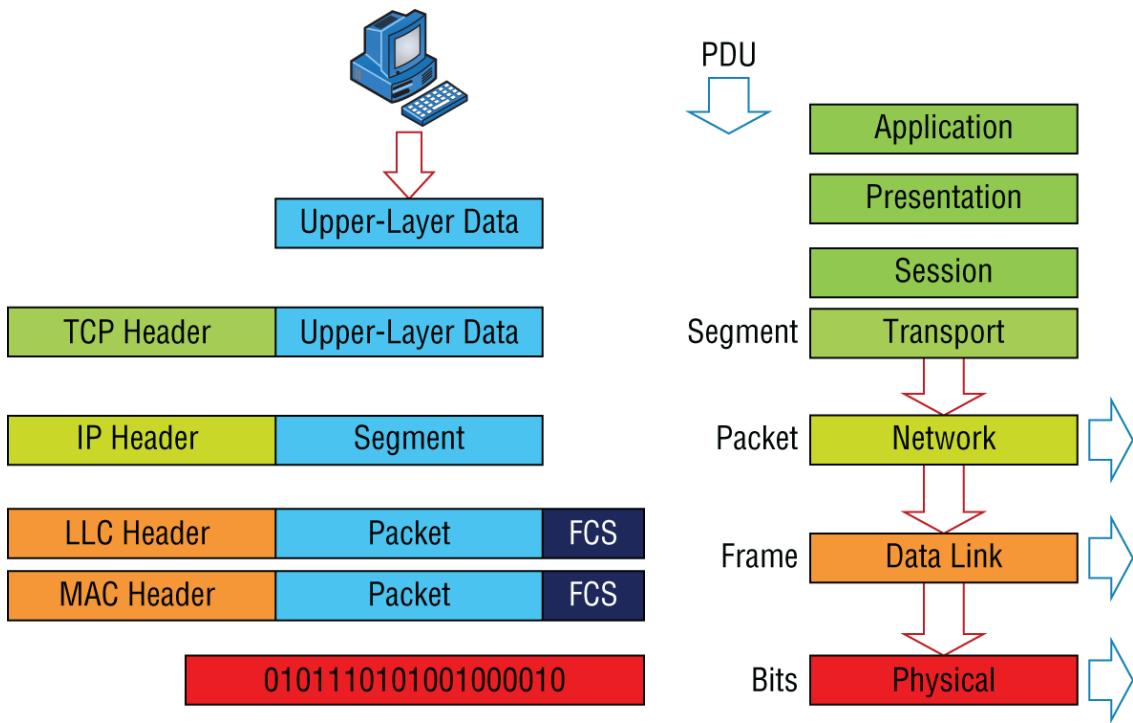


그림 2.11 데이터 캡슐화

## 변조 기법

네트워크에서 변조는 전송될 정보를 담고 있는 신호를 이용하여 반송파 신호라고 불리는 파형의 하나 이상의 속성을 변화시키는 과정입니다.

파형 변조는 기저대역(이더넷 또는 무선) 메시지 신호를 통과대역 신호로 변환합니다. (통과대역은 대역통과 필터링된 신호라고도 하며, 필터를 통과할 때 감쇠되지 않는 주파수 또는 파장 범위를 말합니다.) 현재 네트워크에서 변조는 디지털 또는 아날로그 신호를 물리적으로 전송 가능한 다른 신호로 변환하는 과정입니다.

변조기는 신호를 변조하는 장치이고, 복조기는 변조의 역과정인 복조를 수행하는 장치입니다. 일반적으로 변조기와 복조기 두 가지 동작을 모두 수행할 수 있는 이러한 장치를 모뎀(변조기-복조기)이라고 부릅니다.

디지털 변조의 목적은 아날로그 대역 통과 채널을 통해 디지털 비트 스트림을 전송하는 것입니다. (공중 교환 전화망을 통한 데이터 전송이 좋은 예인데, 이 경우 대역 통과 필터가 주파수 범위를 300~3400Hz로 제한하거나 제한된 무선 주파수 대역을 사용합니다.) 아날로그 변조의 목적은 아날로그 기저대역(또는 저역 통과) 신호(예: 오디오 신호, 무선 네트워크 또는 TV 신호)를 다른 주파수의 아날로그 대역 통과 채널을 통해 전송하는 것입니다.

아날로그 및 디지털 변조는 주파수 분할 다중화(FDM)라는 방식을 사용하는데, 이는 여러 개의 저역 통과 정보 신호를 서로 다른 통과 대역 채널(여러 개의 서로 다른

주파수)을 사용하여 동일한 공유 물리적 네트워크를 통해 동시에 전송하는 방식입니다.

이더넷 네트워크에서 사용되는 디지털 기저대역 변조 방식(라인 코딩이라고도 함)은 기저대역 채널을 통해 디지털 비트 스트림을 전송하는 데 사용됩니다. 기저대역이란 변조되는 신호가 사용 가능한 전체 대역폭을 활용한다는 것을 의미합니다.

시분할 다중화(TDM)는 전송선로 양 끝에 동기화된 네트워크 장치를 사용하여 공통 신호 경로를 통해 여러 개의 독립적인 신호를 송수신하는 방식입니다. 이 방식에서는 각 신호가 교대로 특정 시간 동안만 전송선로에 나타납니다. 수신단에서는 신호를 다시 원래 형태로 역다중화합니다.

다음 몇 장에 걸쳐 네트워킹에 대한 기초적인 내용을 더 배우고 나면, 6장에서 캡슐화 방법으로 돌아와 더 자세히 논의하겠습니다.

## 요약

이제 여러분은 탄탄한 기본 지식을 갖추게 되었습니다. 이를 바탕으로 더욱 발전해 나가면 자격증 취득에 한 걸음 더 다가설 수 있습니다.

이 장에서 배운 내용을 잠시 복습해 보겠습니다. 먼저 인터넷 네트워킹 모델과 그 장점에 대해 논의했습니다. 그다음 OSI 모델, 즉 애플리케이션 개발자가 모든 유형의 시스템이나 네트워크에서 실행될 수 있는 애플리케이션을 설계하는 데 도움이 되는 7계층 모델에 대해 설명했습니다. 각 계층은 모델 내에서 고유한 역할과 책임을 가지고 있으며, 이를 통해 견고하고 효율적인 통신이 이루어지도록 합니다. 각 계층에 대한 자세한 설명과 OSI 모델의 명세를 이해하는 방법에 대해 이야기했습니다.

또한 네트워킹에서 사용되는 캡슐화 방식에 대해서도 논의했습니다. 캡슐화는 이해해야 할 매우 중요한 개념이며, 이 책 전체에 걸쳐 계속해서 다룰 것입니다.

이 장은 디지털 및 아날로그 신호 변조에 대한 간략한 소개로 마무리되었습니다.

## 시험 필수 사항

**OSI 계층 구조를 기억하세요.** OSI 모델의 7개 계층과 각 계층의 기능을 반드시 기억하고 이해해야 합니다. 응용 계층, 표현 계층, 세션 계층은 상위 계층으로, 사용

자 인터페이스에서 응용 프로그램으로의 통신을 담당합니다. 전송 계층은 네트워크 분할, 순서 지정, 가상 회로를 제공합니다. 네트워크 계층은 논리적 네트워크 주소 지정과 인터넷워크를 통한 라우팅을 제공합니다. 데이터 링크 계층은 네트워크 매체 상에서 데이터의 프레임 구성 및 배치를 담당합니다. 물리 계층은 1과 0을 디지털 신호로 인코딩하여 네트워크 세그먼트를 통해 전송하는 역할을 합니다.

**데이터 링크 계층의 하위 계층들을 알아야 합니다.** OSI 계층 외에도, 데이터 링크 계층만이 유일하게 하위 계층을 가지고 있으며, 각 하위 계층의 기능이 무엇인지 아는 것은 매우 중요합니다. 데이터 링크 계층은 LLC와 MAC이라는 두 개의 하위 계층으로 구성됩니다. LLC 하위 계층은 다음과 같습니다. MAC 서브레이어는 주로 네트워크 계층 프로토콜의 다중화를 담당합니다. 물리적 주소 지정과 네트워크에 데이터를 배치할 적절한 시점을 결정하는 역할을 합니다.

**OSI 모델의 각 계층에서 작동하는 장치를 알아두세요.** 허브와 리피터는 비트만 처리하므로 1계층 장치입니다. 모든 네트워킹 장치는 네트워크에 물리적으로 연결되어 있기 때문에 모두 1계층에서 작동하지만, 허브와 리피터는 이 계층에서만 작동하는 반면 다른 장치는 더 높은 계층에서도 작동합니다. 하지만 일반적으로 장치는 지원하는 가장 높은 계층에서 작동한다고 간주합니다. 해당 계층의 기능이 네트워크에 장치를 구현하는 주된 이유이기 때문입니다. 예를 들어 스위치와 브리지는 2계층 주소를 이해하고 이를 기반으로 결정을 내리기 때문에 2계층 장치로 간주됩니다. 라우터는 비슷한 이유로 3계층 장치입니다. 라우터는 3계층 주소를 처리하기 때문입니다. 애플리케이션을 실행하는 워크스테이션과 같은 네트워킹 장치는 네트워크에 연결된 애플리케이션에 서비스를 제공하는 애플리케이션 계층 프로토콜을 포함해야 하므로 애플리케이션 계층에서 작동한다고 합니다(또는 모든 계층에서 작동한다고도 합니다).

## 필기 실험

필기 실험 문제의 답은 [부록 A](#)에서 확인할 수 있습니다.

- 통신 파트너의 가용성과 연결에 필요한 리소스를 선택하고 결정하며, 파트너 애플리케이션을 조정하고, 데이터 무결성 및 오류 복구 절차에 대한 합의를 형성하는 계층은 어느 계층입니까?
- 데이터 링크 계층의 프레임을 전기 신호로 변환하는 역할을 하는 계층은 어느 계층입니까?
- 라우팅은 어느 계층에서 구현되어 두 종단 시스템 간의 연결 및 경로 선택을 가능하게 합니까?

4. 데이터의 형식, 표현 방식, 인코딩 방식 및 변환 방식을 정의하는 계층은 무엇입니까?
5. 애플리케이션 간 세션 생성, 관리 및 종료는 어느 계층에서 담당합니까?
6. 물리적 링크를 통한 데이터 전송을 관리하고 주로 물리적 주소 지정 및 프레임의 순서 있는 전달과 관련된 계층은 무엇입니까?
7. 네트워크를 통해 종단 노드 간의 안정적인 통신을 제공하고 가상 회로를 설정, 유지 및 종료하는 메커니즘과 정보 흐름을 제어하는 메커니즘을 제공하는 계층은 무엇입니까?
8. 라우터가 경로를 결정하는 데 사용하는 논리적 주소 지정은 어느 계층에서 제공됩니까?
9. 전압, 배선 속도, 커넥터 핀 배열을 지정하고 장치 간에 비트를 전송하는 계층은 무엇입니까?
10. 비트를 바이트로, 바이트를 프레임으로 결합하고 MAC 주소 지정을 사용하는 계층은 무엇입니까?

## 복습 문제

복습 문제에 대한 답은 [부록 B](#)에서 찾을 수 있습니다.

1. 호스트 1이 호스트 2로 SYN 패킷을 보냈습니다. 호스트 2는 이에 대한 응답으로 무엇을 보낼까요?
  - A. ACK
  - B. 낙크
  - C. SYN/ACK
  - D. SYN/NAK
  - E. SYN
2. TCP와 UDP는 OSI 모델의 어느 계층에 속합니까?
  - A. 1
  - B. 2
  - C. 3
  - D. 4
3. OSI 모델의 어느 계층이 프로그램이 네트워크 인프라에 접근할 수 있는 진입점을 제공합니까?
  - A. 애플리케이션
  - B. 수송
  - C. 회로망
  - D. 물리적

4. 인터넷상의 서버에 접속하여 서버의 링크를 클릭했는데 시간 초과 메시지가 나타났습니다. 이 메시지의 원인은 어떤 계층일까요?
- A. 애플리케이션
  - B. 수송
  - C. 회로망
  - D. 물리적
5. OSI 모델의 어떤 계층이 코드 및 문자 집합 변환과 데이터 형식 인식을 담당합니다?
- A. 애플리케이션
  - B. 프레젠테이션
  - C. 세션
  - D. 회로망
6. 브리지, 허브, 라우터는 OSI 모델의 어느 계층에서 각각 주로 작동합니까?
- A. 물리적, 물리적, 데이터 링크
  - B. 데이터 링크, 데이터 링크, 네트워크
  - C. 데이터 링크, 물리적, 네트워크
  - D. 물리적, 데이터 링크, 네트워크
7. OSI 모델의 어느 계층이 데이터를 전송 매체에 적합한 신호로 변환하는 역할을 담당합니까?
- A. 애플리케이션
  - B. 회로망
  - C. 데이터 링크
  - D. 물리적
8. 수신 호스트가 승인해야 할 모든 세그먼트를 수신하지 못했습니다. 호스트가 이 통신 세션의 신뢰성을 높이려면 어떻게 해야 할까요?
- A. 다른 소스 포트 번호를 보내십시오.
  - B. 가상 회로를 재시작합니다.
  - C. 순번을 줄이세요.
  - D. 창 크기를 줄이세요.
9. 단일 LAN 세그먼트의 커버리지 영역을 확장하는 데 사용할 수 있는 레이어 1 장치는 무엇입니까? (두 가지를 선택하십시오.)
- A. 방화벽
  - B. NIC
  - C. 바퀴통
  - D. 연발총
  - E. RJ-45 트랜시버
10. 데이터 스트림의 분할은 OSI 모델의 어느 계층에서 발생합니까?

A. 물리적

B. 데이터 링크

C. 회로망

D. 수송

11. 데이터를 캡슐화할 때, 올바른 순서는 무엇입니까?

A. 데이터, 프레임, 패킷, 세그먼트, 비트

B. 세그먼트, 데이터, 패킷, 프레임, 비트

C. 데이터, 세그먼트, 패킷, 프레임, 비트

D. 데이터, 세그먼트, 프레임, 패킷, 비트

12. 다리를 이용한 분할의 두 가지 목적은 무엇입니까? (두 가지를 선택하십시오.)

A. 방송 도메인을 추가하려면

B. 더 많은 충돌 영역을 생성하기 위해

C. 사용자에게 더 많은 대역폭을 제공하기 위해

D. 사용자들이 더 많은 방송을 시청할 수 있도록

13. 응답, 순서 지정 및 흐름 제어는 OSI 계층 중 어느 계층의 특징입니까?

A. 레이어 2

B. 레이어 3

C. 레이어 4

D. 레이어 7

14. 다음 중 순서 및 감사의 글에 관해 맞는 것은 무엇입니까? (해당하는 모든 항목을 선택하십시오.)

A. 전송된 데이터는 수신 즉시 발신자에게 수신 확인 메시지가 전송됩니다.

B. 세그먼트가 수신되지 않으면 가상 회로를 더 느린 전송 간격으로 처음부터 다시 시작해야 합니다.

C. 확인 응답이 없는 부분은 모두 재전송됩니다.

D. 세그먼트들은 목적지에 도착하면 원래 순서대로 다시 정렬됩니다.

E. 모든 세그먼트는 시간 슬롯 간격으로 재전송됩니다.

15. 유량 제어의 목적은 무엇입니까?

A. 확인 응답을 받지 못한 경우 데이터가 재전송되도록 하기 위함입니다.

B. 목적지 장치에서 세그먼트를 올바른 순서로 재조립하려면

C. 수신자가 발신자가 전송하는 데이터 양을 제어할 수 있는 수단을 제공하기 위해

D. 각 세그먼트의 크기를 조절하기 위해

16. OSI 모델의 어느 계층에 IP가 위치합니까?

A. 수송

B. 회로망

C. 데이터 링크

D. 물리적

17. 다음 중 OSI 모델에서 가장 상위 계층은 무엇입니까?

- A. 수송
- B. 세션
- C. 회로망
- D. 프레젠테이션

18. 라우터는 OSI 계층 중 어느 계층에서 라우팅을 수행합니까?

- A. 물리적
- B. 데이터 링크
- C. 회로망
- D. 수송
- E. 애플리케이션

19. OSI 모델의 각 계층 이름의 첫 글자를 올바른 순서대로 기억하는 데 사용할 수 있는 연상 기법은 다음 중 무엇입니까?

- A. 모든 사람들이 가공된 데이터를 필요로 하는 것 같다.
- B. 절대로 누구도 경찰견을 위협해서는 안 됩니다.
- C. 소시지 피자를 버리지 마세요.
- D. 사람들은 하루 종일 새로운 직업에 도전해 봐야 합니다.

20. CSMA/CD 프로토콜을 규정하는 IEEE 표준은 무엇입니까?

- A. 802.2
- B. 802.3
- C. 802.5
- D. 802.11