네트워크 개론 Part 11 : 네트워크 통신 모델 II

글. 조인준 KBS 미디어기술연구소 차장

지난 연재를 통해 네트워크 통신 모델에 관한 소개를 시작하며 7개의 계층으로 구성된 OSI(Open Systems Interconnection) 모델을 우선 설명 드렸습니다. OSI 모델에 관한 이야기 중에 네트워크 통신을 이해하는데 유용한 측면이 많아서 현재의 인터넷을 통해 주로 사용되는 TCP/IP 모델이 있음에도 교육용으로 널리 쓰이고 있다고도 말씀드렸습니다.

이번 연재에서는 실제로 활용되고 있는 네트워크 통신 모델인 TCP/IP 모델에 대해 설명 드리겠습니다.

TCP/IP 모델

TCP/IP 네트워크 통신 모델이 널리 사용되게 된 주된 이유는 미국 국방부가 개발하여 군용 네트워크를 제외한 연구용 네트워크를 민간에 개방한 ARPANET(Advanced Research Projects Agency NETwork)에 실제로 적용된 모델이었다는 것과 관련 기술들이 ARPANET 참여 연구기관 등에서 지속해서 사용되며 개량되어 왔기 때문이라고 합니다. 또한, 새로운 기술을 만들기보다는 존재하는 유용한 기술들을 포용했기에 실용적 모델로 발전해올 수 있었다고 합니다.

TCP/IP 네트워크 통신 모델은 RFC-1122 문서에 정의되어 있으며 [그림 1]의 좌측과 같습니다. 참고로 RFC는 Request for Comments의 약자로 IETF(Internet Engineering Task Force : 국제 인터넷 표준화 기구)에서 제공 및 관리하는 문서들의 앞에 붙이는 명칭입니다. RFC-1122에는 TCP/IP 모델이 [그림 1]의 좌측과 같이 4계층 모델로 정의되어 있습니다. 하지만 링크 계층에 다양한 기술들이 등장하며 사람들이 이를 포괄하기 위해 링크 계층을 데이터 링크 계층과 물리 계층으로 분화하여 다루기 시작하였고, 인터넷 계층이 네트워크 계층으로 불리게 되면서 [그림 1]의 우측과 같은 5계층의 TCP/IP 모델이 현재 실질적으로 통용되는 TCP/IP 모델이라고 합니다. 이런 실질적 통용 모델에서 눈에 띄는 것은 [그림 2]와 같이 TCP/IP 모델이 OSI 모델과 잘 맞아떨어지게 된 점입니다.

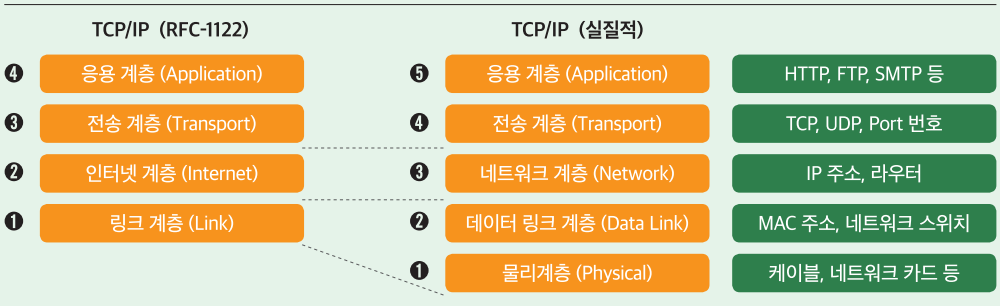

그림 1. TCP/IP 네트워크 통신 모델

[그림 2]와 같이 현재의 실질적 TCP/IP 모델은 ①~④ 계층까지 OSI 모델과 일치하며 ⑤ 계층에 OSI 모델의 ⑤~⑦ 계층을 포괄하는 대응 관계를 갖습니다. 그러면 [그림 3]과 같이 서버와 클라이언트 애플리

케이션 간의 네트워크 통신 과정을 통해서 TCP/IP 모델의 각 계층에서 어떤 처리가 이루어지는지 들여다보겠습니다.

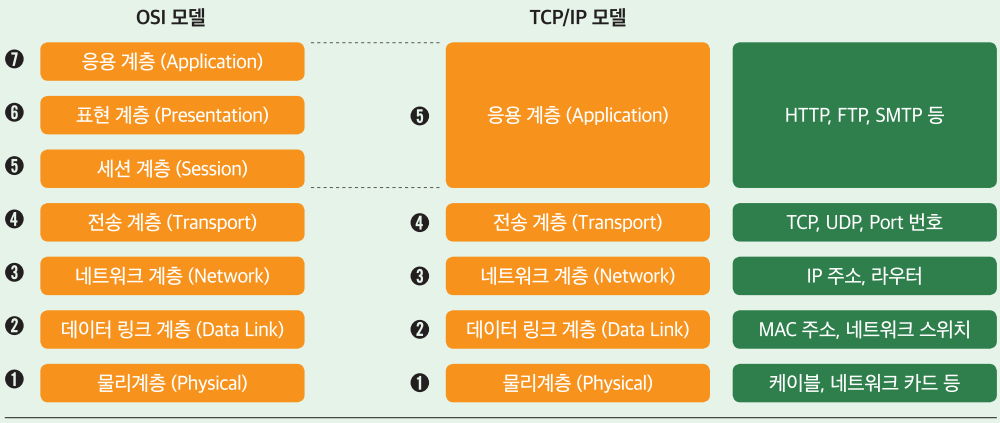

그림 2. OSI 모델 vs TCP/IP 모델

지금까지 네트워크 개론을 연재하면서 수차례 '포트(Port)'라는 것에 대해 언급하며 네트워크 통신을 하고 있는 애플리케이션을 식별하기 위해 사용한다는 간단한 설명만 드렸습니다. 웹서버와 같이 HTTP(HyperText Transfer Protocol)를 사용하는 서버의 경우 [그림 3]과 같이 80번 포트를 통해 클라이언트로부터 HTTP 요청(Request)을 받도록 RFC-2616에 정의되어 있으며, 클라이언트는 내부의 애플리케이션에 임의의 포트 번호를 할당하여 요청을 보낼 수 있습니다. 서버는 HTTP 요청을 받을 때 같이 포함되어온 클라이언트 애플리케이션의 포트 번호로 요청에 대한 응답을 보내어 해당 애플리케이션에 응답이 도착할 수 있게 합니다. 이제 그동안 모호했던 '포트 번호'와 애플리케이션의 관계가 조금 더 실체적으로 설명된 것 같습니다.

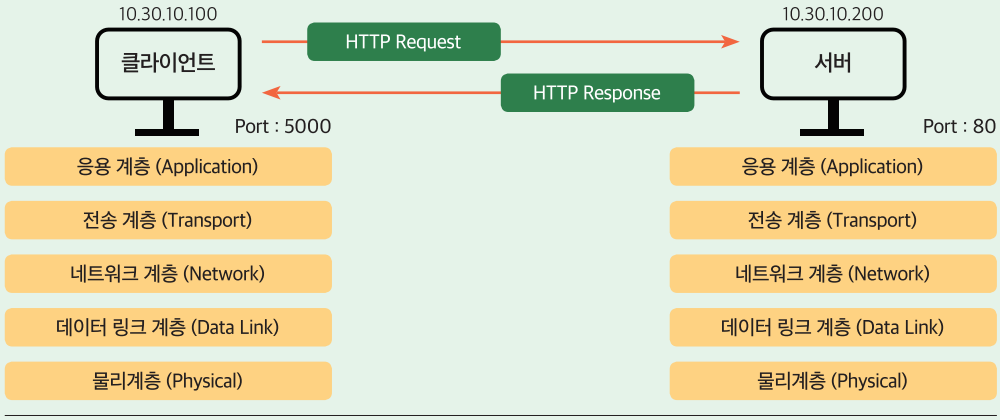

그림 3. 서버 ↔ 클라이언트 통신

[그림 3]의 HTTP 요청이 서버로 전달되는 과정을 각 계층에 따라 단계별로 풀어보겠습니다. 우선 클라이언트 내부의 애플리케이션은 [그림 4]와 같이 응용계층에서 서버에 보낼 요청을 HTTP 표준에 맞는 서식에 담아 HTTP Data를 생성합니다. HTTP Data가 생성되면 전송계층에서는 HTTP Data에 요청을 받아줄 서버의 포트 번호(Destination Port)와 응답을 받을 애플리케이션에 할당된 포트 번호(Source

Port)가 담긴 헤더를 [그림 5]와 같이 붙입니다. 이때 전송 계층에 TCP를 사용할지 UDP 등의 다른 방식을 사용할지에 따라서 헤더의 서식은 달라집니다.

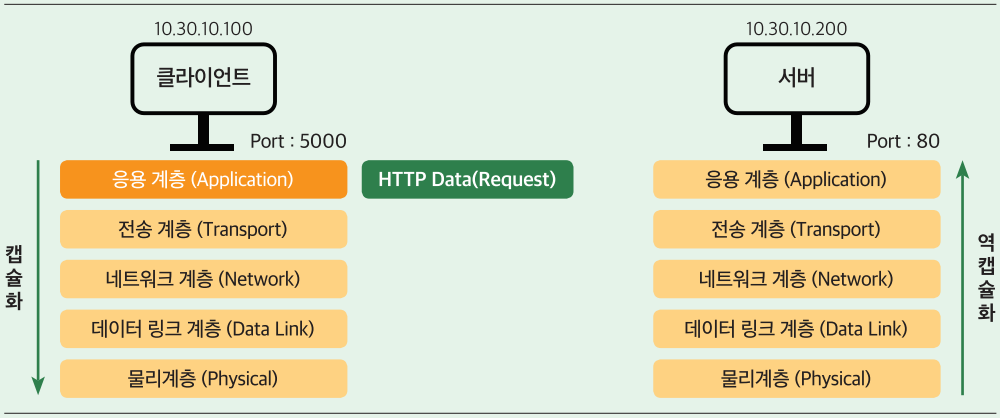

그림 4. TCP/IP 모델 응용 계층 처리

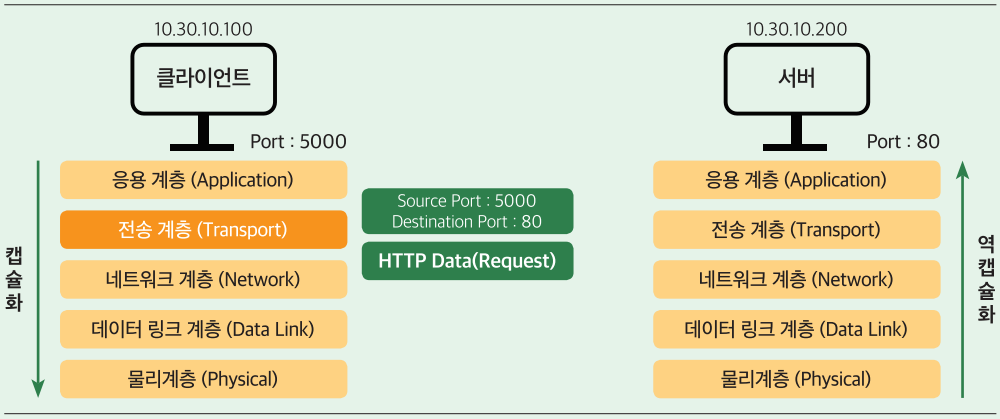

그림 5. TCP/IP 모델 전송계층 처리

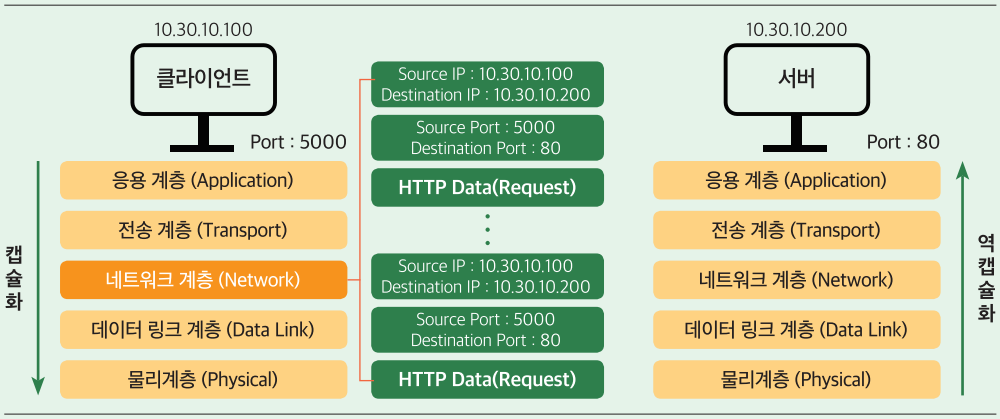

그림 6. TCP/IP 모델 네트워크 계층 처리

포트 번호가 추가된 HTTP Data가 네트워크 계층으로 내려가면 [그림 6]과 같이 전송에 적합한 사이즈로 HTTP Data를 나눈 후 포트 번호를 포함하는 전송계층 프로토콜 헤더 위에 서버의 IP 주소(Destination IP)와 응답을 받을 클라이언트의 IP 주소(Source IP) 정보가 담긴 헤더를 [그림 6]과 같이 추가합니다. 이

때 나누어진 각 조각은 지난 연재에서 소개드린 바와 같이 패킷(Packet)이라고 부릅니다.

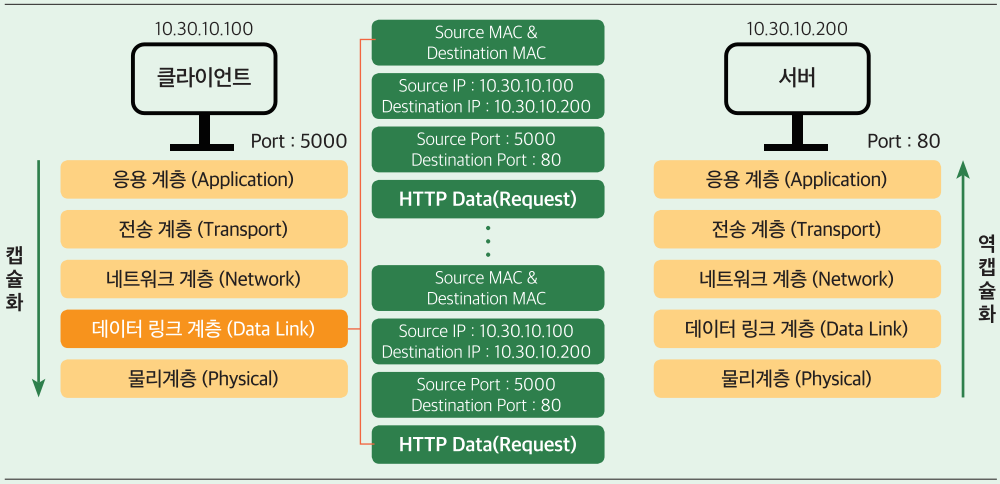

그림 7. TCP/IP 모델 데이터 링크 계층 처리

마지막으로 각 패킷을 보내는 클라이언트의 MAC 주소와 각 패킷이 향해야 할 MAC 주소(서버와 클라이언트가 같은 네트워크에 있을 경우 서버의 MAC 주소, 서버와 클라이언트가 서로 다른 네트워크에 있을 경우 클라이언트가 속한 네트워크의 게이트웨이 MAC 주소)를 포함하는 데이터 링크 계층 프로토콜 헤더를 [그림 7]과 같이 추가하여 물리계층을 통해 전달하면 서버로 패킷화 된 데이터가 전달됩니다. 패킷을 받은 서버는 캡슐화의 역순으로 헤더를 하나씩 떼어내며 클라이언트의 애플리케이션이 보낸 HTTP Data를 추출합니다.

한 가지 주목할 내용은 [그림 8]과 같이 역캡슐화 과정에서 네트워크 계층을 통할 때 전송을 위해 분할된 패킷들을 다시 합쳐 전송 계층 처리로 올려주게 됩니다. 역캡슐화 마지막 단계인 응용 계층을 통해 클라이언트의 요청을 받은 서버는 적절한 처리를 통해 응답을 보내며 응답을 보내는 과정에서는 보내는 쪽과 받는 쪽이 반대가 되므로 Source Port/IP/MAC이 서버 쪽의 값들로 채워지고, Destination Port/IP/MAC이 클라이언트 쪽의 값들로 채워집니다.

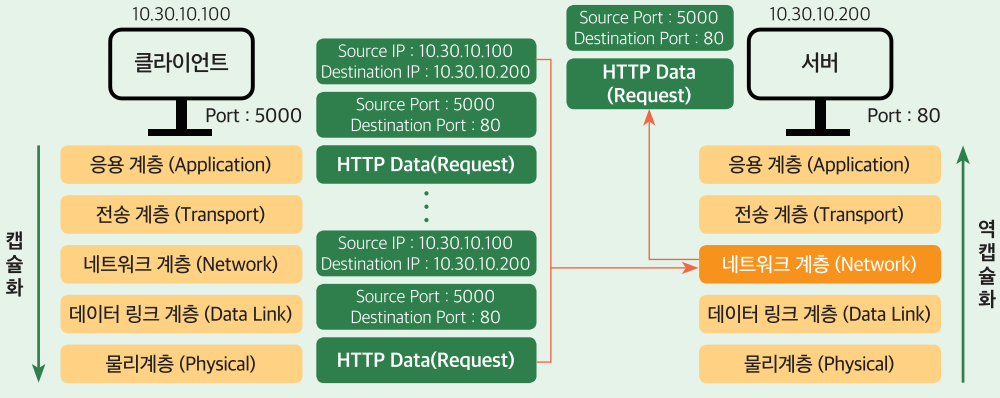

그림 8. TCP/IP 모델 서버 측 네트워크 계층 처리

지금까지 TCP/IP 네트워크 통신 모델과 이 모델이 실제로 어떻게 동작하고 있는지에 관한 개념적 설명을 드렸습니다. 다음 연재에서는 TCP와 UDP 프로토콜에 관한 내용들로 이어가 보겠습니다.