네트워크 개론 Part 22 : TCP vs UDP, 사실과 오해 2

조인준
KBS 미디어기술연구소
차장

지난 편인 'TCP vs UDP, 사실과 오해 1'을 통해 일반적으로 많이 알려진 TCP와 UDP의 네 가지 차이점 및 그 원인에 관한 설명을 시작하며 [표 1]에 회색으로 표시한 연결지향 및 신뢰성에 관한 내용까지 다루었습니다. 이번 편에서는 노란색으로 표시한 흐름제어와 오버헤드에 관한 내용으로 이어가보겠습니다.

표 1. TCP vs UDP

	TCP	UDP
연결 지향적	○	X
신뢰성	UDP 대비 높음	TCP 대비 낮음
흐름제어	○	X
오버헤드	높음	낮음

• 흐름제어

[그림 1]은 인터넷을 간략화한 것입니다. 인터넷은 여러 네트워크를 연결하는 라우터의 복잡한 집합이라고 할 수 있습니다. 이 라우터마다 처리할 수 있는 데이터의 대역폭에 차이가 있어서 [그림 1] 하단에 표시된 바와 같이 송신 디바이스(Sender)가 수신 디바이스(Receiver)로 데이터를 전송하며 거치는 라우터 구간마다 대역폭의 고저가 있습니다.

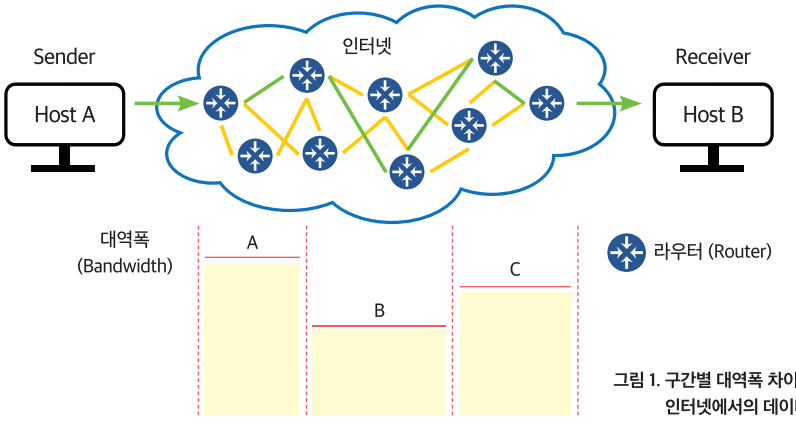

그림 1. 구간별 대역폭 차이가 있는 인터넷에서의 데이터 전송

TCP의 경우 수신 디바이스의 ACK 메시지를 이용한 흐름제어를 통해서 네트워크 사정에 따라 단위 시간당 전송 데이터양을 조절하는 메카니즘을 가지고 있습니다. 그러므로 처음엔 전송량을 늘리다가 [그림 1] B 구간의 대역폭을 넘어서며 패킷 유실이 발생하여 수신 디바이스의 ACK 메시지 수신에 실패하면 다시 단위 시간당 전송 데이터양을 줄이는 등의 흐름제어를 통해 B 구간의 대역폭 이하로 단위 시간당 전송 데이터양이 수렴하면서 안정적 데이터 전송을 이어가게 됩니다. 하지만 UDP의 경우는 수신 디바이스의 ACK 메시지를 이용하는 TCP와 달리 수신 디바이스의 데이터 수신 여부의 확인 없이 가능한 빨리 데이터를 전송하기만 하기 때문에 [그림 1] B 구간의 대역폭을 넘어서 전송하게 되는 경우 데이터 손실이 발생할 수밖에 없습니다. 때문에 상위(레이어 5)에서 수신 실패 관련 별도 처리를 하지 않는 한 전송 중 데이터 손실이 발생할 가능성이 높습니다.

• 오버헤드

TCP는 UDP에 비해 오버헤드가 높은 편입니다. 여기서 오버헤드는 TCP나 UDP를 사용하여 레이어 5(애플리케이션 레이어)의 데이터를 전송하기 위해 소비되는 추가의 데이터나 컴퓨팅 자원 정도의 의미로 이해하시면 됩니다. 이 맥락에서 보자면 TCP나 UDP의 오버헤드는 [그림 2]의 오렌지색으로 표시된 레이어 4 헤더(TCP/UDP 헤더)와 직접적으로 연관됩니다.

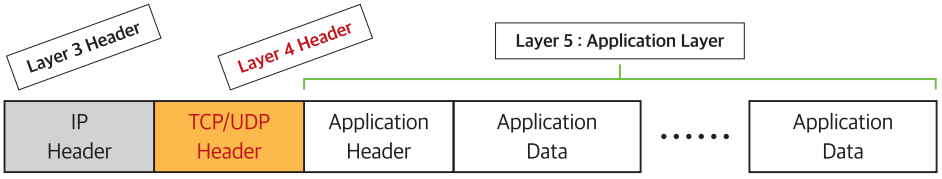

그림 2. IP 패킷의 구조

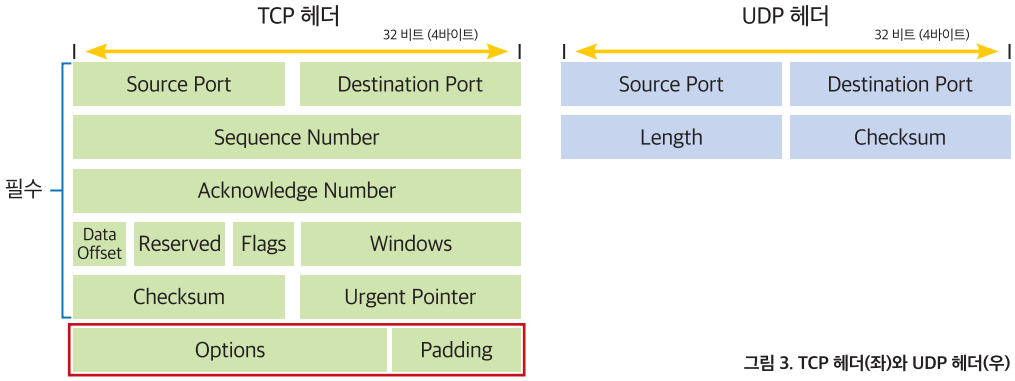

그림 3. TCP 헤더(좌)와 UDP 헤더(우)

[그림 3]은 TCP와 UDP의 헤더를 보여주고 있습니다. TCP 헤더의 크기는 기본적으로 [그림 3]에 표시한 필수 필드에 해당하는 20바이트가 최소 크기가 되며, 밑에 붉은 사각형으로 표시한 옵션 부분이 경우에 따라 0~40바이트가 될 수 있습니다. 그러므로 TCP 헤더는 최소 20바이트에서 최대 60바이트의 사이즈를 갖습니다. 이에 반해 UDP 헤더는 8바이트로 매우 간단한 구조를 갖고 있습니다. 이렇게 헤더 구조의 복잡도나 사이즈에 차이가 있는 것은 UDP의 경우 목적지 관련 정보와 전송하는 데이터 사이즈 등에 관한 최소 필요 정보만 포함하고 TCP와 같이 목적지에 데이터가 제대로 도달했는지 확인 및 도달 후 데이터의 순서를 정렬하기 위한 정보 등 안정적 전송을 보장하기 위한 데이터를 포함하지 않기 때문입니다.

• TCP vs UDP에 관한 오해

TCP와 UDP에 관해 알려진 속설들이 몇 가지가 있고 그중에 어떤 것은 표현에 대한 오해로 인해 사실과는 다르게 해석되는 것도 있습니다. 대표적인 오해 사례 몇 가지를 다음 페이지에 나열해보았습니다.

① UDP가 TCP보다 더 빠르다?

흔히 알려진 속설 중에 UDP가 TCP보다 빠르다는 속설이 있습니다. UDP가 TCP보다 빠르다고 할 때는 같은 조건에서 UDP를 사용했을 때 TCP보다 수신 디바이스에서 수신하는데 걸리는 시간이 짧아져야 합니다. 하지만 무슨 방식을 사용하던 다른 조건이 동일하다면 UDP와 TCP로 동시에 데이터를 송신했을 때 수신 디바이스에서 수신하는 시간에 차이가 발생할 수 없습니다. 아마도 이런 속설은 보내야 하는 응용계층(Application Layer)의 데이터가 동일할 때, TCP 헤더가 UDP 헤더 대비 오버헤드가 높기에 전체적으로 더 많은 데이터를 전송해야 하므로 전송완료까지 시간이 더 걸릴 수 있어서 생긴 오해 같습니다. 따라서 'UDP가 TCP보다 더 빠르다'라는 표현은 사실과 다른 해석의 여지가 있으며, 'UDP가 TCP보다 오버헤드가 적다'가 정확한 표현 같습니다.

② TCP가 UDP보다 안전하다?

TCP가 UDP보다 안전하다는 이야기도 많이 듣습니다. 이 안전하다는 표현 때문에 기술적 내용을 모르는 사람들은 TCP에 일종의 보안 기술이 적용되어 있다고 오해하는 경우도 있는 것 같습니다. 하지만 레이어 4 프로토콜인 TCP나 UDP가 제공하는 보안 기술은 없습니다. 우리가 흔히 아는 보안 기술인 SSL(Secure Sockets Layer), TLS(Transport Layer Security), SSH(Secure Shell)은 레이어 5에서 제공되는 기술이고 레이어 3에서는 IPSec(Internet Protocol Security)이라는 보안 기술이 있습니다. 따라서 TCP가 UDP보다 안전하다는 말의 의미는 ACK 메시지를 활용한 수신 디바이스의 데이터 수신 확인 및 유실된 데이터의 재전송 등을 통해서 안전하게 데이터를 전송해준다는 의미이며 보안과는 상관이 없습니다.

③ UDP는 신뢰성이 낮다?

이 말은 UDP를 사용하면 모든 경우 신뢰성이 낮다고 이해되는 경향이 있습니다. 하지만 UDP 프로토콜 자체가 TCP와 비교했을 때 신뢰성이 낮다고 볼 수 있는 것이지 UDP를 사용한다고 무조건 신뢰성이 낮은 것은 아닙니다. UDP를 사용하던 TCP를 사용하던 다른 조건이 같다면 수신 디바이스에 데이터가 도착하거나 유실될 확률은 같습니다. 다만, TCP는 데이터 유실 여부를 확인하고 후속 조치를 취하는 방법을 프로토콜 안에 내장하고 있는 것이고, UDP의 경우는 이를 레이어 5에서 각기 다른 애플리케이션의 목적에 따라 그 목적에 맞는 방법으로 데이터 유실 문제를 해결하도록 하고 있습니다. 따라서 UDP 방식 자체의 신뢰성이 아닌 애플리케이션 전체의 신뢰성을 평가하는 것이 적절하며 UDP를 사용한다고 무조건 전체 애플리케이션의 신뢰성이 낮다고 생각하는 것은 오해입니다.

③ TCP는 완벽한 전송을 보장한다?

TCP는 수신 측에 온전히 데이터가 도착했는지 등의 확인과 데이터 유실시 재전송을 통한 최선의 전송 수단을 제공합니다. 하지만 이는 네트워크를 구성하는 기기들에 물리적 문제나 오작동이 없을 때에 해당하는 이야기일 뿐 여러 고장 상황을 고려했을 때에도 TCP가 전송을 보장하는 마법 같은 기술을 탑재하고 있는 것은 아닙니다. TCP는 데이터 전송 여부의 확인과 유실시 재전송이 가능할 뿐이지 네트워크의 고장이나 장애를 극복할 수 있는 것은 아니므로 완벽한 전송을 보장 한다기보다 특별한 고장이 없을 시 최선의 전송을 보장한다는 의미로 이해하는 것이 맞는 것 같습니다.

• UDP 응용 분야

TCP와 비교하다 보면 왠지 UDP는 좋은 방식이 아닌듯한 부정적 인상을 받게 됩니다. 하지만 실제로 UDP는 여러 응용분야에 사용되고 있는 괜찮은 방식 중의 하나입니다. 다음은 레이어 5의 프로토콜 중 레이어 4에 UDP를 채용하고 있는 것들에 대한 예입니다.

① 간단한 요청 및 응답으로 동작하는 애플리케이션

UDP 기반의 간단한 요청 및 응답 구조로 동작하는 애플리케이션 중 DNS(Domain Name System) 서버의 경우를 예로 들

어보겠습니다. DNS 서버는 UDP를 사용하여 "방송과기술 웹 사이트(tech.kobeta.com)의 IP 주소를 알려줘"와 같은 요청을 받으면 110.10.xxx.xxx 형식의 IP 주소를 응답으로 보내줍니다. 이렇게 UDP를 사용하면 한 번의 클라이언트 요청에 대해 한 번의 서버 대답으로 양쪽 총 2번의 패킷 전송이면 마무리됩니다.

그런데 TCP를 쓴다면 3-Way 핸드셰이크로 양쪽이 3번 패킷을 주고받아 연결을 설정한 다음 클라이언트가 "방송과기술 웹 사이트(tech.kobeta.com)의 IP 주소를 알려줘"라는 요청을 송신하고 이에 대한 서버의 ACK을 수신하며 2번의 패킷 전송을 하고, 서버가 110.10.xxx.xxx 형식의 IP 주소를 응답으로 보낸 후에 클라이언트의 ACK 메시지를 수신하며 다시 2번의 패킷 전송을 한 다음 마지막으로 연결을 종료하며 4-Way 핸드셰이크로 4번 패킷을 주고받습니다. 웹 사이트 IP 주소 하나를 알아내는데 총 11개의 패킷이 클라이언트와 서버를 오갑니다.

UDP를 사용하면 패킷 전송 2번이면 끝나는 일을 TCP를 사용하면 11번 패킷 전송을 해야 하므로, 이런 경우 UDP를 쓰는 것이 효율적입니다. 그런데 "전송 성공 여부를 확인 안 하는 UDP의 특성상 IP 주소를 못 받으면 문제가 생기지 않나요?"라는 의문이 독자여러분께 생길 것 같습니다. 이 경우 DNS 서버와 클라이언트 사이에 같은 요청을 두 번이나 세 번 보내서 전송이 실패하는 경우를 대비할 수 있고, 이렇게 두세 번 보내더라도 총 패킷의 개수는 4개나 6개 정도이기에 TCP를 사용하는 것보다 훨씬 간단하게 처리할 수 있습니다. 비슷한 예로 NTP(Network Time Protocol), SNMP(Simple Network Management Protocol), DHCP(Dynamic Host Configuration Protocol) 등의 프로토콜이 있습니다.

② 레이어 5에 전송 확인 메커니즘이 내장된 애플리케이션

TFTP(Trivial File Transfer Protocol)라는 파일 전송 프로토콜은 UDP 기반의 프로토콜이며 자체 내에 전송 확인을 하는 메커니즘을 내장하고 있습니다. 쉽게 말하면 파일을 여러 개의 블록으로 쪼개어 UDP로 전송하고, 각 블록에 대한 수신 확인 메시지를 UDP로 받아서 해당 블록의 전송 성공을 확인하면 다음 블록의 전송으로 넘어갑니다. 만약 TFTP에 TCP를 사용한다면 전송 확인을 위한 절차가 레이어 4와 레이어 5에서 중복으로 발생해서 불필요하게 복잡한 처리를 하게 됩니다.

③ 실시간 콘텐츠 전송이 필요한 애플리케이션

VoIP(Voice over IP)를 이용한 인터넷 전화나 RTP(Real-time Transfer Protocol)를 이용한 실시간 영상 전송의 경우 UDP를 많이 채용하고 있습니다. 이유는 TCP를 사용하면 일부 패킷의 유실이 발생했을 때 재전송을 기다려야 하고 이때 음성이나 영상이 멈추게 되는 것이 애플리케이션 목적에 맞지 않기 때문입니다. 상대방과 인터넷을 통해 대화를 할 때 일부 패킷의 유실로 한 두 개의 단어가 빠져도 문맥상 이해에 문제가 없는 경우가 많고, 문맥으로 이해가 안 되면 다시 물어보는 것이 유실된 데이터를 받을 때까지 음성이 멈추는 것보다 실제 사용에 낫기 때문입니다. RTP를 이용한 라이브 콘텐츠 전송도 같은 경우입니다. 일부 영상에 훼손되더라도 실시간으로 전달되는 것이 중요한 콘텐츠라면 유실된 패킷을 다시 받는 동안 화면이 멈추어 있는 것보다 그냥 일부 화면이 훼손된 채로 실시간으로 보여줄 수 있는 최대의 영상 정보를 보여주는 것이 나은 경우도 있기 때문입니다.

이상으로 TCP와 UDP에 관한 모든 이야기를 마치겠습니다. 다음 편에서는 우리가 일상에서 많이 접하는 레이어 5 프로토콜들을 소개해드리겠습니다.

P.S.

C군이 여러분께 전하는 내용 중 전문적 성격이 짙은 것은 엄밀한 언어를 사용하여 설명하기에는 한계가 있습니다.
본 내용은 설명하는 대상에 대한 전체적 맥락의 이해에만 이용하시고, 그 이상은 권위 있는 전문자료를 참고하시기 바랍니다.