

1. 교재 1.4.1절을 참고하여 다음의 의미를 정리하시오.

(1) **processing delay**: 패킷의 헤더를 확인하고 패킷이 어디로 가야하는 지 결정할 때 지연되는 것을 말한다. 패킷의 에러를 체크할 때도 프로세싱 지연이 일어날 수 있다. 고속 라우터에서는 프로세싱 지연은 몇 마이크로초 이하로 걸린다. 이 과정 이후 라우터는 패킷을 링크된 라우터의 큐로 보낸다.

(2) **queuing delay**: 패킷이 큐에서 전송되기를 기다릴 때 발생하는 지연을 말한다. 지연의 길이는 몇 개의 패킷이 이미 큐에 있는지, 링크로 전송될 때까지 얼마나 대기해야 하는지에 따른다. 만약 큐가 비어 있고, 어떠한 패킷도 전송되고 있지 않다면 패킷의 지연은 0이다. 반면, 트래픽이 혼잡하고 많은 패킷이 전송되길 기다리고 있다면 지연은 길어질 것이다. 큐 지연은 마이크로초에서 밀리초 정도가 될 수 있다.

(3) **transmission delay**: 패킷 스위치에서 패킷은 FCFS(First-Come-First-Served)에 따라 전송되는데, 다음 노드에 먼저 도착한 패킷이 바로 그 다음 노드로 전송되는 것이 아니라, 뒤에 전송되는 다른 패킷들이 도착해야 전송될 수 있다. 이때 지연되는 시간을 트랜스미션 지연이라고 하며, 패킷의 길이와 링크 전송률의 영향을 받는다.

(4) **propagation delay**: 링크로 보내진 비트는 라우터 B로 전송되어야 하는데, 이때 걸리는 시간이 전파 지연이다. 비트는 링크의 전파 속도에 따라 전파되며, 전파 속도는 링크의 물리적 매체(구리선, 광섬유 등)에 따라 다르다. 전파 속도의 범위는 보통  $2 * 10^8$  meters/sec에서  $3 * 10^8$  meters/sec사이이며, 이는 빛의 속도보다 조금 느린 정도다. 전파 지연은 두 라우터의 거리  $d$ 를 전파 속도  $s$ 로 나눈 것으로,  $d/s$ 가 된다. WAN 전파 지연은 밀리세컨드 수준이다.

2. OSI 7 layer의 7개 layer가 수행하는 기능들을 설명하시오.

(1) **Application Layer**: 네트워크에서 가장 처음 접하게 되는 계층으로, 엔드 유저에게 가장 가깝다. 어플리케이션 프로세스가 네트워크 서비스에 접근할 수 있도록 하며, 엔드 유저가 해석할 수 있는 형태로 리소스를 보여주는 역할을 한다.<sup>1</sup>

(2) **Presentation Layer**: 어플리케이션 레이어에게 표준화된 형식으로 데이터를 제공하는 역할을 한다. 데이터의 암호화/복호화, 비교, 변환 등 어플리케이션 레이어 아래에서 적절한 형식으로 데이터를 가공한다.

---

<sup>1</sup> "The OSI Model – Application Layer", [InetDaemon](#), 2018.

(3) **Session Layer:** 서로 통신하는 엔드 시스템 사이의 세션을 구축/해제하고, 세션 메시지를 주고받으며 역할을 한다.

(4) **Transport Layer:** 발신 호스트와 수신 호스트가 네트워크를 통해 데이터를 주고받을 수 있도록 하며, 에러를 정정하는 일도 한다. 또한 어플리케이션 메시지와 트랜스포트 레이어 헤더 정보를 덧붙여 트랜스포트 레이어 세그먼트를 구성하고, 목적지 호스트의 주소를 포함해 네트워크 레이어에 전달한다.

(5) **Network Layer:** 네트워크 레이어는 데이터의 라우팅 경로를 제공하며, 차후 패킷의 네트워크 경로를 설정한다. 또한 논리적인 연결을 구축하고, 데이터를 포워딩/라우팅하기도 한다. 네트워크 레이어에는 대부분 IP가 쓰이기 때문에 통신을 용이하게 할 수 있다.

(6) **Data Link Layer:** 네트워크 레이어가 라우팅한 데이터그램은 이후 노드와 노드 사이를 움직이게 되는데, 데이터 링크 레이어는 피지컬 레이어를 통해 다음 노드로 데이터를 전송할 수 있도록 한다. 또한 신뢰성을 위한 에러 제어와 플로우 컨트롤 기능도 한다. 이를 위해 데이터 링크 레이어는 링크 레이어 프레임 구성해 피지컬 레이어로 보낸다.

(7) **Physical Layer:** 피지컬 레이어는 데이터 링크 레이어가 구성한 링크 프레임의 각 비트를 다음 노드로 보내는 역할을 한다. 피지컬 레이어는 링크의 물리적 매체(광케이블, 동축케이블 등)와 밀접하며, 실제 케이블의 배치나 전압, 연결 등 말 그대로 물리적 영역을 관할한다.

### 3. Application layer에서의 Client-Server 구조와 Peer-to-Peer 구조에 대하여 설명하시오.

클라이언트-서버 아키텍처에는 다른 호스트로부터 요청받은 데이터를 제공하는 호스트(서버)와 서버에게 서비스를 요청하는 호스트들(클라이언트)이 존재한다. 클라이언트는 서버에 접속해 리소스를 요청하고, 서버는 클라이언트가 요청한 리소스를 제공하는 방식으로 작동한다. 클라이언트-서버 구조는 중앙 집중 방식이며, 서버가 동작을 멈추면 클라이언트가 서비스를 제공받을 수 없게 된다.

반면 P2P 아키텍처에는 중앙 호스트가 없다. 대신 서로 연결되어 있는 각각의 호스트들(노드)이 직접 통신한다. 클라이언트-서버 구조를 빗대자면, P2P 구조에서는 각각의 호스트가 서버인 동시에 클라이언트가 되는 셈이다. 클라이언트-서버 구조와 다르게 한 노드에 문제가 발생해도 다른 노드들이 잘 동작한다면 서비스 제공이 완전히 중단되지는 않는다.

### 4. TCP에서의 connection establishment 과정을 설명하시오.

TCP에서는 three-way handshake 방식을 사용하며, 두 엔드 시스템 A, B가 TCP 통신을 한다고 하면,

SYN, SYN-ACK, ACK 세 단계를 거친다.<sup>2</sup>

(1) **SYN**: A가 B에게 접속을 요청하기 위해 TCP 패킷을 SYN 플래그와 함께 전송하고 B의 응답을 기다린다.

(2) **SYN-ACK**: B가 SYN과 ACK 플래그가 설정된 패킷을 A에게 전송해 연결을 수락한다.

(3) **ACK**: A가 B의 응답을 확인하고, ACK 플래그를 설정한 패킷을 B에게 전송해 TCP 통신의 시작을 알린다.

이렇게 세 단계의 연결 과정을 거침으로써 통신의 신뢰성을 확보할 수 있다.

5. 이전에는 멀티미디어 스트리밍을 위하여 UDP를 기반으로 한 RTSP 또는 RTP가 활용되었다.

최근에는 대역폭의 증가로 TCP를 기반으로 한 HTTP 스트리밍 방법이 많이 사용된다. 이러한 HTTP 스트리밍 방법에 대하여 조사하여 정리하시오. (Hint: 교재의 2.62절을 참고하여, 추가적인 내용을 조사하여 정리함)

HTTP 기반 스트리밍에서 영상은 HTTP 서버에 저장되어 있다. 사용자는 TCP 연결을 기반으로 URL을 통해 HTTP GET 요청을 보냄으로써 영상 파일을 요청할 수 있다. 서버는 HTTP 응답 메시지에 비디오 파일을 포함해 전송하며, 클라이언트는 전송받은 파일의 바이트(bytes)를 어플리케이션 버퍼에 담는다. 이렇게 영상을 받아오는 동시에 비디오 어플리케이션은 버퍼의 영상을 사용자에게 보여준다.

HTTP 스트리밍이 주로 사용되자 영상을 끊김없이 스트리밍하기 위한 방법이 모색되었고, HTTP를 기반으로 하는 adaptive HTTP streaming 방식이 등장했다. 기존의 스트리밍 방식은 서버에서 한 영상 파일을 전송받는 방식이었기 때문에 사용자의 네트워크 상황에 따라 영상이 끊길 수밖에 없었다. 한편 adaptive streaming은 같은 영상을 여러 해상도로 인코딩해 다양한 버전을 저장해두고, 네트워크 상황에 따라 적절한 해상도의 영상 데이터를 전송하도록 하여 끊김 없는 스트리밍을 제공한다.<sup>3</sup> 가령 3G 환경에서는 낮은 해상도의 영상 청크(chunk)를 전송하다가 LTE 환경이 되면 높은 해상도의 영상을 스트리밍할 수 있다.<sup>4</sup>

adaptive streaming 방식을 쓸 때는 HTTP 서버에 있는 각각의 비디오가 서로 다른 URL을 가진다. 서버는 매니페스트 파일(Manifest file)을 가지고 있으며, 클라이언트는 이 파일을 보고 네트워크에 따른 영상 청크를 요청할 수 있다. 매니페스트 형식에는 애플이 만든 HLS(HTTP Live Streaming)와 DASH(Dynamic Adaptive Streaming over HTTP)가 있다.

---

<sup>2</sup> "What is the TCP connection establishment and termination?", [Quora](#), 2018.

<sup>3</sup> 김성호, "웹 기술로 구현하는 Adaptive HTTP Streaming", [TOAST Meetup](#), 2017.

<sup>4</sup> "Live streaming web audio and video", [MDN web docs](#), 2018.

6. POP3와 IMAP의 특징에 대하여 설명하시오. 교재와 강의자료 외에 다른 자료도 참고하여 정리하시오.

POP3는 인증, 트랜잭션, 업데이트 세 단계를 거친다. 인증은 유저 에이전트가 사용자 이름과 비밀번호를 보내는 것이다. 얼마 후 트랜잭션이 시작되면 유저 에이전트가 메시지를 탐색한다. 이때 유저 에이전트는 메시지에 삭제 마크를 붙이거나, 메일에 관한 통계를 얻을 수 있다. 세 번째로 업데이트는 사용자가 quit 명령을 내렸을 때 시작되며, POP3 세션이 끝난 뒤 메일 서버는 삭제 마크가 붙은 메일을 지운다.

POP3의 업데이트 단계에는 다운로드 삭제 (Download and delete) 모드와 다운로드 유지(Download and keep) 모드가 있다. 기본적으로 POP3는 메일을 다운로드 받으면 메일 서버에서 해당 메시지를 삭제한다. 따라서 다른 컴퓨터에서는 해당 메시지에 접근할 수 없다. 이것이 다운로드 삭제 모드다. 일부는 다운로드 유지 모드로 작동하는데, 이 경우 메일 서버에 메시지의 복사본을 남겨둔다. 한편, POP3 메일 서버는 유저 에이전트와 동기화되지 않으며, 공용 폴더 액세스도 지원하지 않는다.<sup>5</sup>

IMAP은 POP3보다 다양한 기능을 제공한다. POP3와 달리 IMAP 서버는 유저 에이전트와 동기화되며, 폴더 생성 등의 작업을 하면 유저 에이전트와 메일 서버가 똑같이 변경된다. 또한 서버에 메일이 보관되기 때문에 여러 호스트에서 같은 메시지에 접근할 수 있고, 다른 메일 클라이언트에서도 접근할 수 있다.

7. 다음과 같은 환경을 고려하기로 할때, 물음에 답하시오.

- o client와 local DNS server간의 RTT는  $R_1$  임
- o DNS server들간의 RTT를  $R$ 임
- o server들은 caching한 정보가 없음

(1) Iterative Query 방식 사용시, client가 request한 후 부터 response를 받을때까지의 총 시간을 구하시오.

client가 local DNS server에게 호스트네임을 쿼리할 때  $R_1$  시간이 걸린다. local DNS server가 root DNS server와 통신할 때  $R$  시간이 걸리고, local DNS server가  $N$ 개의 TLD DNS server와 통신할 때는  $NR$  시간이 걸리게 된다. 이후 local DNS server가 authoritative DNS server와 통신할 때  $R$  시간이 걸린다.

따라서 총  $R_1 + NR + 2R$  시간이 걸린다.

---

<sup>5</sup> "POP3 and IMAP4", [Microsoft Docs](#), 2018.

(2) Recursive Query 방식 사용시, client가 request한 후 부터 response를 받을때까지의 총 시간을 구하시오.

client와 local DNS server가 통신할 때  $R_1$  시간이 걸린다. local DNS server가 root DNS server와 통신할 때  $R$ , root DNS server가  $N$ 개의 TLD DNS server와 통신할 때  $NR$  시간이 걸리게 된다. 이후 TLD DNS server가 authoritative DNS server와 통신할 때  $R$  시간이 걸린다.

따라서 총  $R_1 + NR + 2R$  시간이 걸린다.

(3) local DNS server가 이전 record를 cache하고 있을때, (1)과 (2)의 시간은 어떻게 되는지를 구하시오.

client가 local DNS server와 통신하면 되므로,  $R_1$  시간이 소요된다.

8. 교재의 BitTorrent 파트 내용을 설명하여 정리하시오. (Hint: neighboring peers, rarest first, the highest rate, unchoked, optimistically unchoked, tit-for-tat등의 용어를 찾아서 의미를 읽고 내용을 정리하시오)

BitTorrent는 파일 공유를 위한 P2P 프로토콜이다. 피어들은 콘텐츠 파일의 메타 데이터를 가진 ‘토렌트’ 파일을 다운로드, 배포한다. 어떤 피어(클라이언트)가 특정 파일을 요청하면 요청 메시지가 트래커에게 전송된다. 트래커는 자신의 swarm에서 피어들을 선택해 피어리스트를 만들어 클라이언트에게 전달하고, 클라이언트는 피어리스트에서 피어를 선택해 chunks(파일 조각)을 받는다.<sup>6</sup> 여기서 선택, 연결된 피어들을 neighboring peers라고 하며, 이들은 언제든지 변경될 수 있다. 클라이언트가 neighbor로부터 받아올 chunk를 결정할 때는 rarest-first 방식을 사용하는데, 이는 클라이언트의 neighbor들이 적게 가지고 있는 chunk를 먼저 받는 것을 말한다.

클라이언트는 10초마다 가장 높은 비율로 데이터를 제공하는 neighbor를 찾아 우선 순위를 부여하고 받은 chunk를 다시 보낸다. 이때 대상이 되는 피어들을 unchoked되었다고 말한다. 또한 30초마다 랜덤으로 neighbor를 하나 찾아 chunk를 보내는데, 이때 선택된 파트너 피어는 optimistically unchoked되었다고 한다. 이때 파트너 피어가 클라이언트의 상위 업로더가 된다면 클라이언트나 파트너 피어는 더 나은 파트너를 찾을 때까지 서로 chunk를 주고받는다. 한편, 이들을 제외한 다른 neighbor들은 choked되었다고 한다.

이러한 BitTorrent의 매커니즘은 처음에 협력하고, 이후에는 상대방의 행동에 따라 반응하는 tit-for-tat에

---

<sup>6</sup> 박용현 외 2명, 「비트 토렌트에서 건강한 피어 선택 기반의 파일 다운로드 속도 향상」, 『한국정보기술학회논문지』 제16권 제1호, 134쪽, 2018.

기초한다. 이를 통해 다양한 피어들이 다양한 chunk를 가질 수 있게 된다.<sup>7</sup>

---

<sup>7</sup> Lei Jin, Xin Jin, 「Understanding the Evolution of Cooperation in Distributed Systems」, 『Smart Computing Review』 제4권 제3호, 172쪽, 2014.