

컨텐츠 고속 전달을 위한 네트워킹 기법

Ver. 1.1

2021.03.23

한국외국어대학교

정보통신공학과

이 재 성

김 유 림

지도교수 : 정 성 호

목 차

제 1 장 서론	4
제 1 절 연구 배경	4
제 2 절 연구 목표	5
제 3 절 논문 구성	5
 제 2 장 관련 연구	6
제 1 절 Content-Delivery Network	6
2.1.1 Content-Delivery Network 의 정의	6
2.1.2.1 CDN 필요 기술	9
2.1.2.2 CDN 캐싱 기법	9
2.1.2.3 CDN 으로 전송 가능한 콘텐츠 종류	10
2.1.2.4 CDN 의 장 단점	10
 제 2 절 Content-Centric Networking	11
2.2.1 Content-Centric Networking 의 정의	11
2.2.2 Content-Centric Networking 의 특징	13
2.2.2.1 CCN Message	13
2.2.2.2 CCN Router	14
2.2.2.3 CCN 의 장 단점	15
2.2.3 Content-Delivery Network 과의 차이점	15
 참고 문헌	16

그림 목차

그림 1. Content 집중으로 인한 서버의 병목 현상.....	4
그림 2. CDN 네트워크 구조.....	6
그림 3. CDN 동작 절차 (요청 콘텐츠가 CDN Cache Server에 존재할 때).....	7
그림 4. CDN 동작 절차 (요청 콘텐츠가 CDN Cache Server에 존재하지 않을 때)	8
그림 5. 기존 인터넷 프로토콜 스택과 CCN 프로토콜 스택	11
그림 6. CCN 네트워크 구조	12
그림 7. Interest Packet과 Data Packet	13
그림 8. CCN Router 구조.....	14

제 1 장 서론

제 1 절 연구 배경

4차 산업 혁명의 시대가 도래함에 따라 다양한 IT기술들이 등장하면서 기존 네트워크에서의 트래픽 양은 무궁무진하게 증가하고 있다. 데이터 트래픽 증가 추세에 비해 주파수나 무선통신망의 증가에는 한계가 있기 때문에 망 과부하와 서비스 품질 저하 현상이 발생한다. 따라서, 기존의 네트워크 환경에서 증가하는 트래픽을 처리하기 위한 다양한 네트워킹 아이디어들이 이미 고안 되었거나 현재 연구 중에 있다.

인터넷 설계 초창기에는 사용자들은 원격 접속이나 단말간의 연결을 주로 필요로 했다. 따라서 종단 간 통신에 적합한 형태로 설계되었고 현재까지 이어지고 있다. 하지만 최근 사용자들은 인터넷을 주로 콘텐츠를 찾는 데에 사용하고 실제로 인터넷 트래픽의 대부분은 콘텐츠 관련 트래픽이다[16]. 현재 네트워크 환경은 대용량의 정보를 전달하기에는 분명한 한계가 존재한다. 현재의 인터넷 구조는 IP 기반의 클라이언트-서버 모델을 기반으로 한다. 통신에 참여하는 양 끝단은 상호 연결 관계를 구축한 후, 패킷을 단일 경로를 통해 전송하는 일대일 전송을 하는데, 스마트폰과 같은 이동식 단말이 증가함에 따라 다양한 장소에서 동일한 정보를 필요로 하는 경우가 증가하고 있다. 결과적으로, 동일한 위치에 있는 정보에 대한 수요가 증가하게 되면 병목 현상과 같은 문제점이 발생할 수 있다. 그림 1은 여러 사용자의 콘텐츠 요청 시 발생할 수 있는 네트워크 상의 병목 현상을 보인다.

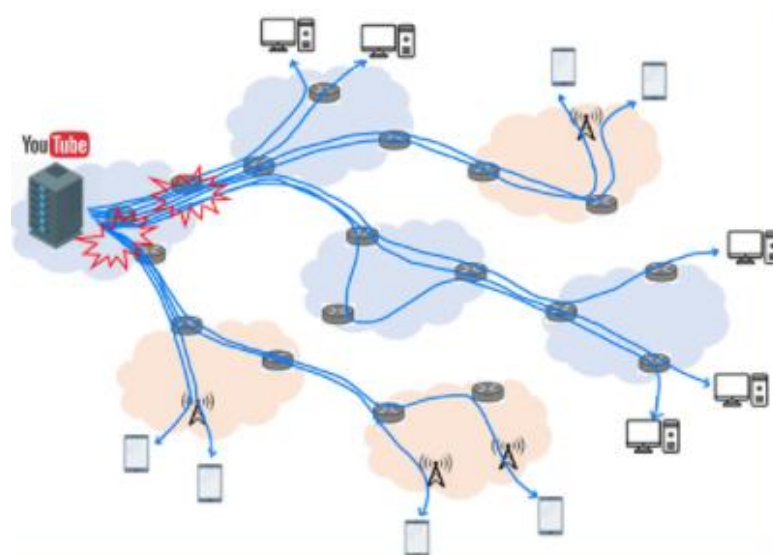


그림 1. Content 집중으로 인한 서버의 병목 현상

이와 같은 문제는 사용자들의 정보 사용 방식이 변화되었기 때문에 발생한다. 따라서, 현재의 사용자들은 원하는 정보의 위치가 아닌 정보 자체에 집중하는 방식으로 정보 사용 방식이 변화하고 있다 (Where to What).

이와 같이, 정보 자체에 집중하는 네트워킹 패킹 패러다임을 정보 중심 네트워킹 (ICN : Information-Centric Networking)이라 부른다. ICN의 프로젝트 중 하나인 콘텐츠 중심 네트워킹(CCN : Content-Centric Networking)은 사용자가 콘텐츠의 이름으로 요청하게 되고, CCN 기능이 탑재된 노드의 캐싱을 통해 원하는 콘텐츠를 보다 효율적이고 빠르게 수신할 수 있게 한다.

제 2 절 연구 목표

CCN을 통해 콘텐츠를 보다 효율적으로 송수신 할 수 있게 되었지만, 분명한 한계는 존재한다. 대표적으로 경로의 부하 문제가 있다. 콘텐츠 요청을 위해 노드에서 Interest Packet을 보내게 되는데 여러 사용자가 동일한 콘텐츠 요청을 위한 Interest Packet을 보내게 되면 Packet Flooding 과정 속에서 링크에 부하가 굉장히 심해진다. 이러한 문제를 해결하기 위해 다양한 기법들이 존재하고 관련 연구도 진행 중에 있다.

본 논문에서는 또 다른 해결 방안 제시를 위하여 Resolution Server라는 새로운 서버를 도입한다. 사용자가 원하는 콘텐츠를 얻기 위해 Interest Packet을 보내기 앞서 Resolution Server에 콘텐츠의 위치를 요청하는 패킷을 보내 실제 콘텐츠가 어디에 저장되어 있는지 확인한다. 그 후, 사용자는 콘텐츠 요청을 위해 콘텐츠가 저장되어 있는 노드로 Interest Packet을 전송해 기존 CCN과 동일한 방법으로 콘텐츠를 얻게 된다.

제 3 절 논문 구성

본 논문에서는 Resolution Server를 도입하여 보다 효율적인 콘텐츠 전달 방안을 제시한다. 본 논문의 구성은 다음과 같다. 제 2 장에서는 콘텐츠 전달을 위한 또 다른 패러다임인 콘텐츠 전송 네트워크(CDN : Content-Delivery Network)의 정의와 특징, CCN의 정의와 특징을 살펴본다. 추가로 관련 연구들인 CDN과 CCN의 차이점을 살펴본다. 제 3 장에서는 본 논문에서 제안하는 Resolution Server를 도입한 CCN 네트워크의 특징을 자세히 살펴본다. 제 4 장에서는 기존 CCN 네트워크에서의 콘텐츠 전달 성능을 평가하고 본 논문에서 제시하는 새로운 CCN 네트워크의 성능을 평가한다. 또한, 각각의 성능을 비교하고 결과를 분석한다. 마지막으로, 제 5 장에서는 결론을 제시한다.

제 2 장 관련 연구

제 1 절 Content-Delivery Network

2.1.1 Content-Delivery Network의 정의

컨텐츠 전송 네트워크(CDN : Content-Delivery Network)는 컨텐츠의 복제 뿐만 아니라, 컨텐츠를 가능한 한 사용자에게 가깝게 가져오기 위해 대역폭을 최대화하고, 접근성을 높이며, 정확성을 유지하는 것을 목표로 1990년대 후반부터 본격적으로 상용화된 네트워크 패러다임이다. 사용자가 원하는 컨텐츠를 컨텐츠 제공자(Content Provider)에게 요청하면, 컨텐츠 제공자는 컨텐츠가 원래 저장되어있는 원 서버(Origin Server)에 요청하는 것이 아닌 사용자로부터 가장 가까이 위치한 서버에서 컨텐츠를 제공하게 한다. 즉, 컨텐츠를 요청할 때, 사용자로부터 가장 가까이 있는 CDN Cache Server가 서비스를 제공하게 한다. 이는 지리적으로 먼 곳에 위치한 컨텐츠가 장거리를 이동하며 생길 수 있는 문제를 최소화 해준다[1]. 또한, 사용자가 네트워크상에 가장 가까운 곳의 CDN Cache Server로부터 컨텐츠를 제공 받도록 하여, 트래픽이 특정 한 서버로 집중되지 않고 각 지역의 여러 서버로 분산되어 병목현상을 예방할 수 있다. 그림 2는 CDN이 적용된 네트워크 구조이다.

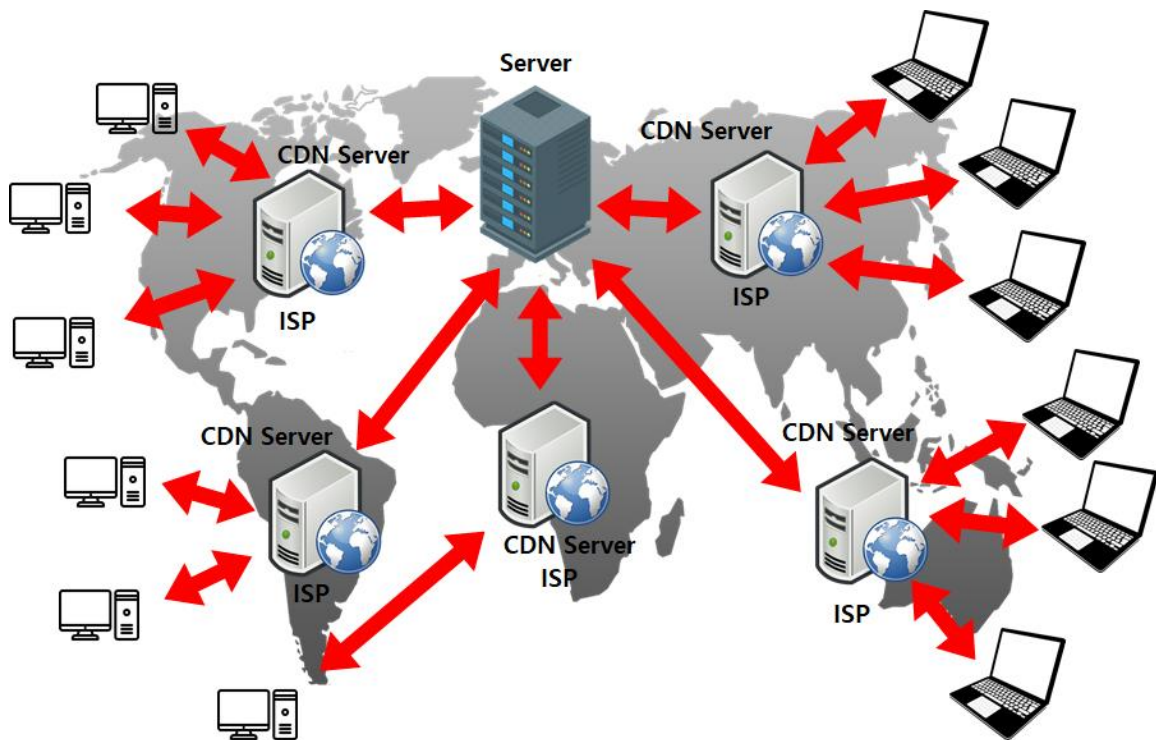


그림 2. CDN 네트워크 구조

CDN의 작동 원리는 다음과 같다. 먼저, 콘텐츠를 최초 요청할 때에는 원 서버에 요청하여 콘텐츠를 가져오게 되고 동시에, 사용자로부터 지리적으로 가장 가까운 CDN Cache Server에 저장한다. 이 후, 동일한 콘텐츠에 대한 모든 요청은 CDN Cache Server에서 처리하게 되고, 이는 CDN 업체에서 지정한 콘텐츠 만료 시점까지 지속된다. 만일, 요청한 콘텐츠가 존재하지 않는 경우 근처에 있는 CDN Cache Server로 부터 콘텐츠를 찾아 사용자에게 전송한다. 그림 3은 CDN Cache Server에 요청된 콘텐츠가 존재 할 때 사용자가 콘텐츠를 받아오는 절차를 나타내는 시퀀스 다이어그램이다 [4].

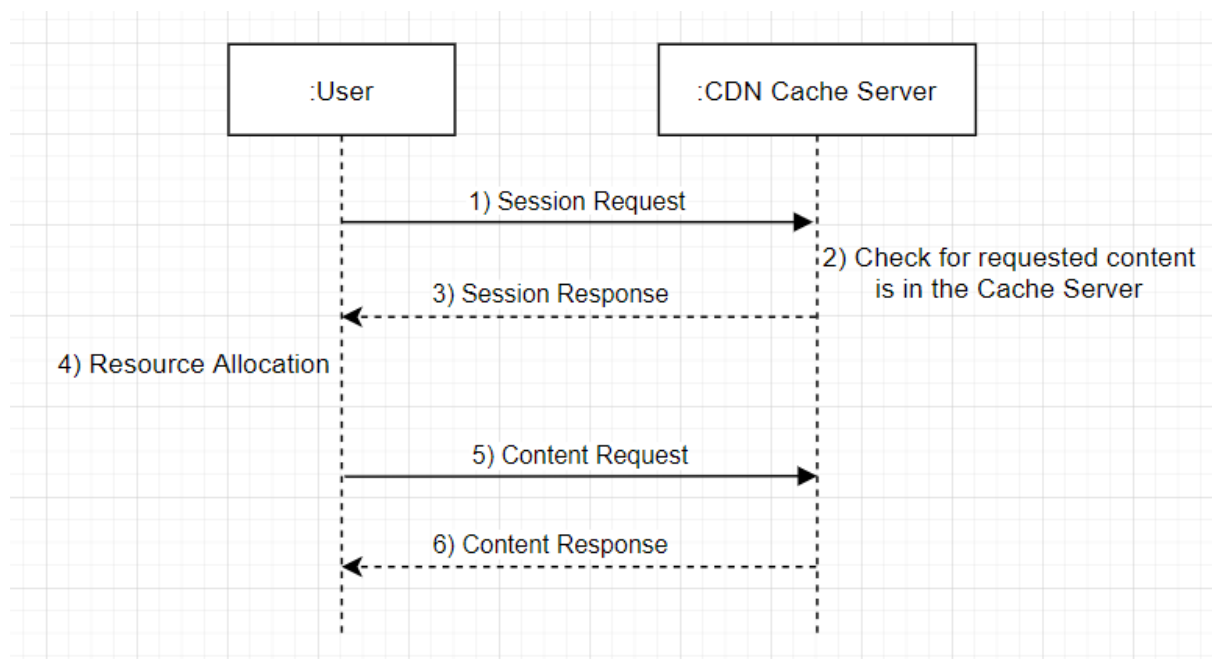


그림 3. CDN 동작 절차 (요청 콘텐츠가 CDN Cache Server에 존재할 때)

- 1) 사용자는 CDN Cache Server에 세션을 요청한다.
- 2, 3) CDN Cache Server에서는 요청된 콘텐츠가 있는지 확인한 후 세션 응답을 보낸다.
- 4, 5) 사용자가 콘텐츠가 존재한다는 응답을 받게 되면 콘텐츠를 저장할 메모리 자원 할당을 한 후, 실제 콘텐츠를 받았다는 요청을 보낸다.
- 6) CDN Cache Server는 요청한 콘텐츠를 응답으로 보낸다.

그림 4 는 CDN Cache Server 에 요청한 콘텐츠가 존재하지 않을 때의 절차를 나타낸 시퀀스 다이어그램이다 [4].

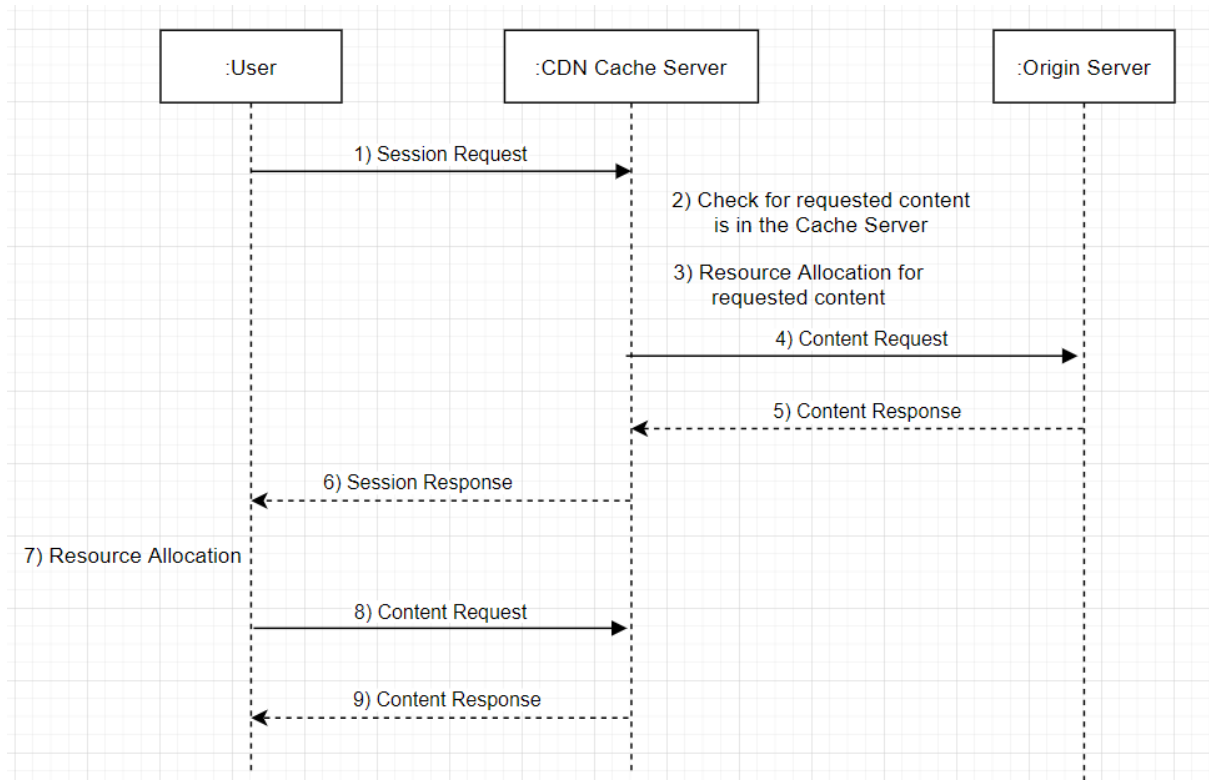


그림 4. CDN 동작 절차 (요청 콘텐츠가 CDN Cache Server에 존재하지 않을 때)

- 1) 사용자는 CDN Cache Server에 세션을 요청한다.
- 2) CDN Cache Server에서는 요청된 콘텐츠가 있는지 확인한다.
- 3) 요청 콘텐츠가 없다면, 콘텐츠 캐싱을 위한 메모리 자원을 할당한다.
- 4, 5) Origin Server로 콘텐츠 요청 후, 콘텐츠를 받아온다.
- 6) 사용자에게 콘텐츠가 존재한다는 세션 응답을 보낸다.
- 7, 8) 사용자가 콘텐츠가 존재한다는 응답을 받게 되면 콘텐츠를 저장할 메모리 자원 할당을 한 후, 실제 콘텐츠를 받겠다는 요청을 보낸다.
- 9) CDN Cache Server는 요청한 콘텐츠를 응답으로 보낸다.

2.1.2 Content-Delivery Network 의 특징

2.1.2.1 CDN 필요 기술

CDN의 효과적인 동작을 위해 다음과 같은 기술이 필요하다 [5].

1) Contents Synchronization

: Origin Server에 있는 콘텐츠 중 변경 사항이 있다면 CDN Cache Server와 동기화 한다.

2) Load Balancing

: 네트워크 트래픽을 여러 서버에 균등하게 분배하는 것으로 처리 속도를 높여주고 서버 용량을 효과적으로 활용할 수 있게 하는 기술이다. Origin Server에 들어오는 요청을 CDN Cache Server에 분산시켜 트래픽 스파이크를 효율적인 방식으로 처리할 수 있다 [15].

3) Streaming Mechanism

: 실시간으로 사용자가 원하는 콘텐츠를 효과적으로 전송하기 위한 기술이다. 이 중, Multicasting Streaming은 동시에 많은 사용자가 콘텐츠를 요청할 때 이를 처리할 수 있는 기술이다. On-Demand Streaming 은 CDN Cache Server로 콘텐츠 복제 후, 서비스를 진행하며 고품질 유니캐스트 스트리밍을 지원한다.

2.1.2.2 CDN 캐싱 기법

CDN의 캐싱 기법에는 다음과 같은 2가지 방식이 있다 [6].

1) Static Caching

: 사용자의 요청이 없더라도 원 서버에 있는 콘텐츠를 미리 CDN Cache Server에 복제해 놓는 캐싱 기법이다. 즉, 사용자가 원하는 콘텐츠는 항상 CDN Cache Server에 존재한다. 현재 국내 대부분의 CDN에서 이 캐싱 기법을 사용한다.

2) Dynamic Caching

: CDN Cache Server에 콘텐츠가 없는 상태에서 사용자가 요청 시 콘텐츠를 원 서버로부터 다운로드 받아와 사용자에게 전달한다. 이 후, 동일한 콘텐츠를 요청 받게되면 이미 캐싱된 콘텐츠를 전달한다. CDN Cache Server에 저장되어있는 콘텐츠들은 일정 TTL이 지나게 되면 삭제된다. 현재 Global CDN업체, 통신사업자향 CDN 장비 솔루션에서 이 캐싱 기법을 사용한다.

2.1.2.3 CDN으로 전송 가능한 콘텐츠 종류

CDN에서 웹사이트 상에서의 콘텐츠 전송은 가장 흔하게 사용되는 사례이다. 이 밖에도 4K 및 HD 품질의 비디오, 오디오 스트리밍과 다양한 소프트웨어 다운로드, 다양한 정보들이 포함된 데이터 레코드 등 다양한 콘텐츠를 전송한다. 즉, 디지털화 할 수 있는 모든 데이터는 CDN으로 전송이 가능하다[6].

2.1.2.4 CDN의 장 단점

CDN은 성능, 가용성, 보안등의 측면에서 다양한 장점을 지닌다. 사용자가 자신이 원하는 콘텐츠에 대한 요청이 원 서버에 도달할 때까지 기다리지 않고 CDN Cache Server에 바로 접속하여 콘텐츠를 제공 받게 된다. 이는 사용자가 경험하게 되는 성능이 대폭 향상한다. 또한, 여러 사용자의 동시 다발 접속으로 인한 트래픽 급증, 네트워크 결함 문제, 잠재적 서버 중단 현상과 같은 상황에서도 CDN Cache Server가 사용자에게 콘텐츠를 제공할 수 있는 가용성의 장점이 있다. CDN은 트래픽 분산으로 인한 DDos와 같은 공격에 대응이 가능하다. 즉, 공격자들의 악성 공격 행위를 막아내 콘텐츠 제공업체와 사용자들을 동시에 보호할 수 있는 보안적인 측면의 장점이 있다 [6].

반면, CDN에서도 분명한 단점은 존재한다. 사용자가 CDN Cache Server에 요청한 콘텐츠가 만약 존재하지 않는다면, 원 서버까지 요청하여 콘텐츠를 받아와 사용자에게 응답해야 한다. 즉, CDN은 초기에 항상 원 서버에까지 연결 과정 후 콘텐츠 캐싱을 해야한다. 또한, CDN Cache Server가 설치되어 있지 않은 국가에서의 요청일 경우, 다른 지역의 CDN Cache Server로 요청할 수 도 있고 혹은 CDN을 사용하지 않을 때 보다 더 먼 곳으로 요청할 수도 있다. 이는 지연 시간을 증가시킬 수 있다. 즉, CDN은 서버 중심적이기 때문에 실질적으로 사용자에서 가장 가까운 노드에서의 캐싱이 제한적이다. CDN은 사용자가 요청하는 콘텐츠를 원 서버 뿐만 아니라 CDN Cache Server 혹은 이미 콘텐츠를 전달받은 사용자들과 같이 다양한 경로를 통해 콘텐츠를 제공할 수 있게 설계 되어있다. 그러나, CDN은 오버레이 네트워크 형태로 구성되어 있어 CDN을 활용하기 위해 많은 비용이 요구되어 일반적인 사용자들은 쉽게 사용하지 못하고 보통 기업체에서 사용해 CDN을 제공한다 [13]. 대표적인 예로 Netflix가 있다. 제 2 절에서는 사용자로부터 물리적으로 가장 가까운 위치에 있는 라우터 단에서 콘텐츠 전송이 가능하여 CDN보다 더욱 효율적인 네트워킹 방식인 콘텐츠 중심 네트워킹 (CCN : Content-Centric Networking)을 살펴본다.

제 2 절 Content-Centric Networking

2.2.1 Content-Centric Networking의 정의

기존의 인터넷 구조는 콘텐츠의 위치를 기반으로 통신하는 구조이다. 이러한 이유로 인해 콘텐츠를 보유하고 있는 호스트의 위치를 식별하기 위한 식별자인 IP 주소는 필수적이었다. 또한, 다른 사용자가 동일한 콘텐츠를 요청하는 경우 반복되는 절차 수행으로 인한 비효율성과 콘텐츠 전달 시간이 증가하는 단점이 존재하였다.

이러한 문제점을 개선한 네트워킹 방식이 정보 중심 네트워킹(ICN : Information-Centric Networking)이다. ICN은 호스트의 위치를 식별하는 IP 주소 대신 해당 콘텐츠의 Name을 이용하여 정보 전달과 요청 및 라우팅을 수행한다. 또한, 이러한 콘텐츠의 정보를 이용하여 원하는 콘텐츠를 고속으로 수신할 수 있게 되었다.

ICN의 여러 관련 기술 중 가장 대표적인 기술은 콘텐츠 중심 네트워킹(CCN : Content-Centric Networking)이다. CCN은 콘텐츠에 Name를 부여하여 콘텐츠의 위치가 아닌 Name을 통해 정보 자체를 획득하는 구조이다. 그림 1은 기존 인터넷 프로토콜 스택과 CCN 프로토콜 스택 구조를 보여준다 [9].

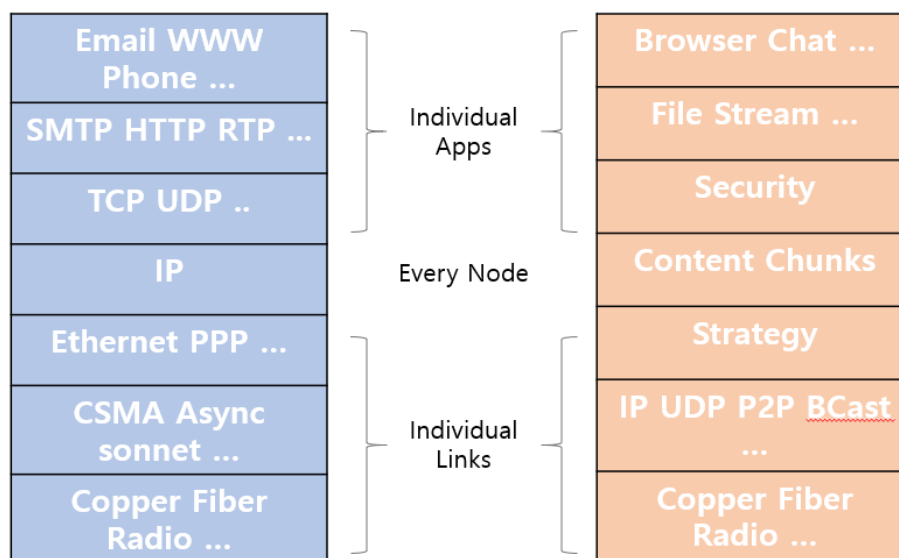


그림 5. 기존 인터넷 프로토콜 스택과 CCN 프로토콜 스택

기존 인터넷 프로토콜 스택과는 달리 CCN 프로토콜 스택은 위치 정보를 나타내는 IP 주소가 아닌 Name을 사용한다. 또한 이름으로 식별 가능한 Content Chunk, Strategy, Security 계층을 사용한다.

CCN 에서 사용자는 요청 메시지에 콘텐츠 name 을 넣어 보낸다. 모든 노드에 해당 콘텐츠가 없다면 요청 패킷이 중앙 서버까지 전달된다. 중앙 서버에서 name 과 상응하는 콘텐츠를 응답 메시지에 담아 거쳐왔던 노드를 지나서 사용자에게 전달된다. 이때 콘텐츠 분배를 위해 거쳐간 모든 Node 들은 해당 콘텐츠를 일시적으로 보관한다. 그 후, 다른 단말이 동일 콘텐츠를 요구하는 경우 중앙 서버까지 요청할 필요 없이 해당 콘텐츠를 보관하고 있는 노드에 요청 후 응답을 받는다. 이러한 패러다임은 중앙 서버에 요청 후 응답 받는 시간을 대폭 줄여 고속으로 콘텐츠 분배가 가능하게 해주며 기존 네트워크 환경의 병목현상을 방지한다. 그림 2 는 CCN 네트워크 구조를 보여준다.

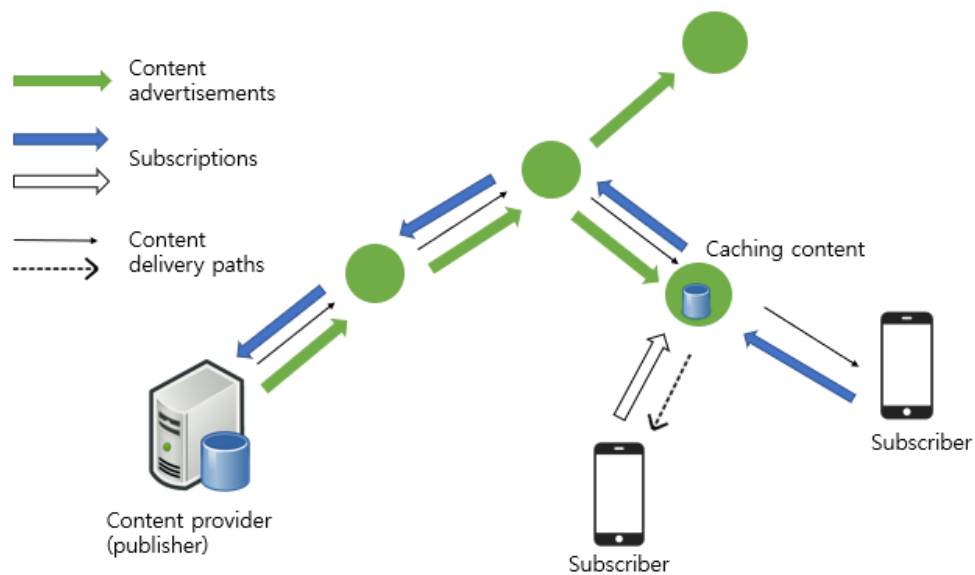


그림 6. CCN 네트워크 구조

2.2.2 Content-Centric Networking의 특징

2.2.2.1 CCN Message

CCN에서 사용하는 메시지 패킷은 Interest Packet과 Data Packet이 있다. Interest Packet에 사용자가 요청 받고자 하는 콘텐츠의 이름을 넣어 전송한다. 여러 라우터에 걸쳐 해당 콘텐츠를 가지고 있는 노드까지 전송된다. 해당 콘텐츠를 가지고 있는 노드는 응답으로 Data Packet에 해당하는 콘텐츠를 담아 사용자에게 콘텐츠를 공급한다. Interest Packet과 Data Packet은 기존 네트워크에서 사용하는 IP 패킷과 달리 콘텐츠 요청자 및 제공자의 IP주소가 없는 것을 볼 수 있다. 그림 3는 Interest Packet과 Data Packet의 구조를 보여준다 [9].

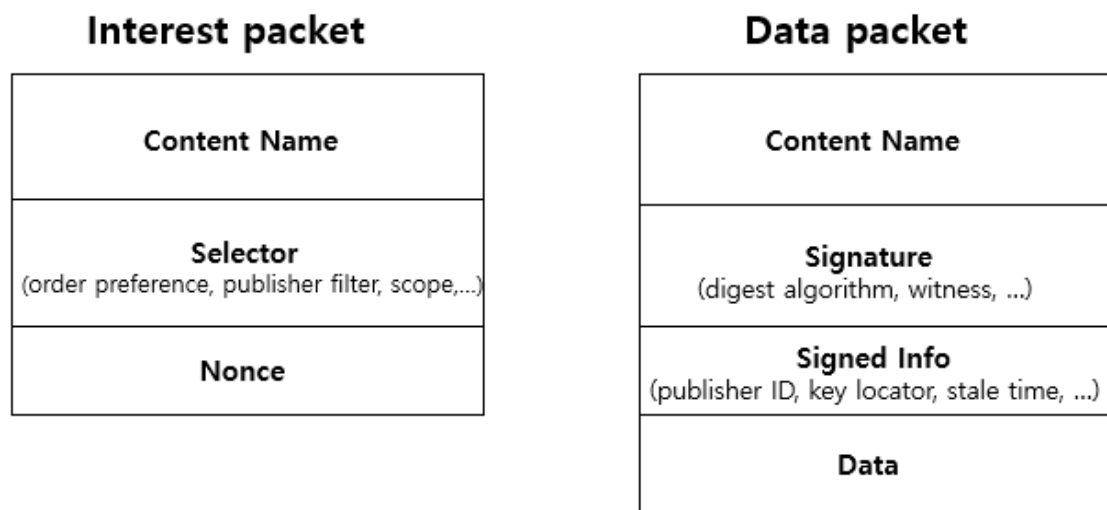


그림 7. Interest Packet과 Data Packet

2.2.2.2 CCN Router

CCN 노트에서 패킷 포워딩을 위해 Content Store(CS), Forwarding Information Base(FIB), 그리고 Pending Interest Table(PIT)라는 세가지 구성요소를 갖는다. CS는 교환되는 콘텐츠를 저장하는 기능을 수행하며, FIB는 콘텐츠 이름 기반의 라우팅 정보를 구성하고 있다. 마지막으로 PIT는 interest packet의 정보와 In/Out 경로를 기록하기 위해 사용된다. CCN 네트워크는 요청 패킷이 이동하는 경로와 해당 요청 패킷에 대한 응답 패킷이 이동하는 경로가 서로 대칭적이므로 역방향 정보를 PIT에 기록한다[10]. Interest Packet이 수신되면 Interest Packet이 지정하는 콘텐츠 이름과 CS에서 연관된 캐싱 정보의 유무를 확인한다. CS에 연관된 캐싱 정보가 있다면 콘텐츠를 사용자에게 보내준다. 만약 CS에 연관된 캐싱 정보가 없다면, 유입된 Interest Packet의 정보가 PIT 테이블에 기록된다. 그 다음 Interest Packet의 전송 경로 결정을 위해 FIB를 참조하여 전송 경로를 결정한다. 추 후에 Data Packet이 수신되면, PIT 테이블을 참조하여 이전에 Interest Packet의 수신여부를 확인한다. 관련된 정보가 이전에 없었다면 수신된 Data Packet을 CS에 저장 후 폐기한다. 만약 관련된 정보가 이전에 수신된 것 이라면 CS에 해당 콘텐츠를 저장하고 Data Packet을 PIT 테이블을 참조하여 다음 노드로 전달한다[11]. 그림 4은 CCN 라우터 구조를 보여 준다.

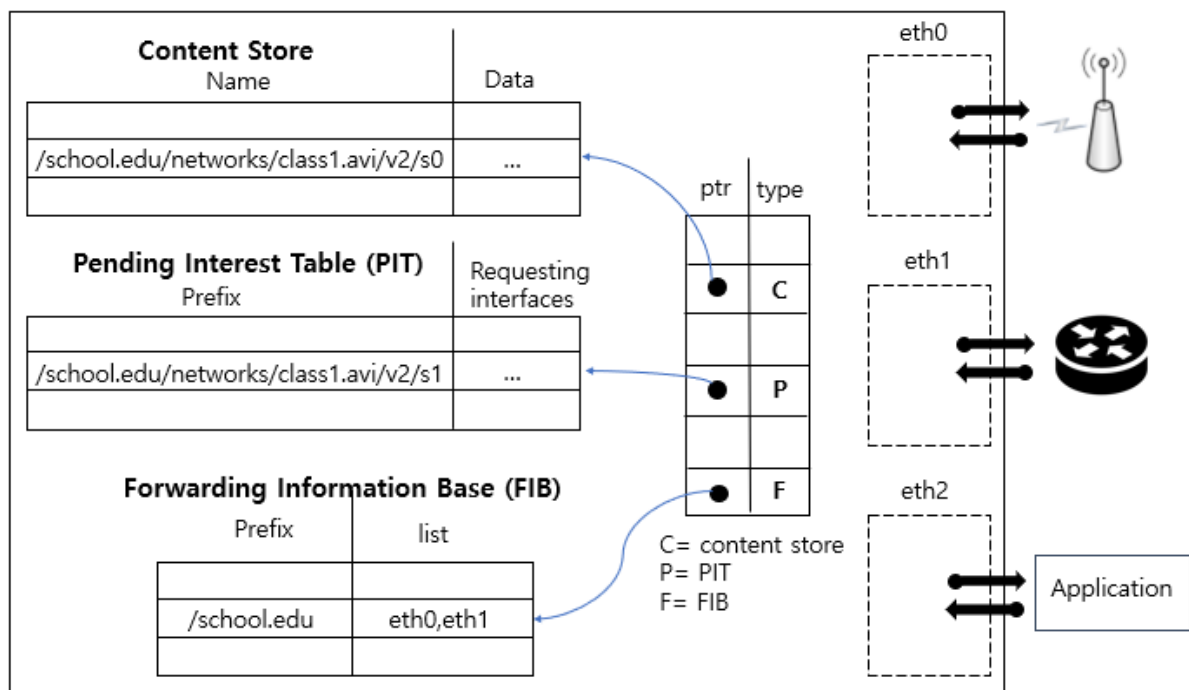


그림 8. CCN Router 구조

2.2.2.3 CCN의 장 단점

CCN은 중간 노드들이 콘텐츠를 보관할 수 있어 공급자까지 요청 메시지를 보낼 필요 없이 해당 콘텐츠를 가지고 있는 노드가 응답메시지를 보낼 수 있다. 따라서 병목현상을 방지 할 수 있으며 고속으로 콘텐츠 분배가 가능하다. 또한 CCN은 공급자와의 연결을 유지할 필요가 없어 이동 단말에 적합하며 Name에 인증 정보가 포함되어 있어 콘텐츠의 빠른 검증이 가능하다. 더불어 공급자의 콘텐츠가 분산되어 있으므로 공격자가 Interest Packet을 통한 직접적인 공격이 어려워 DDos 공격에 방어할 수 있다[12].

반면, CCN은 분명한 한계와 문제점이 존재한다. CCN 라우터가 CCN 프로토콜을 지원하기 위해 보관 해야 하는 정보의 양이 매우 크다. 응답 패킷을 역방향으로 전송하기 위한 정보를 보관해야 할 뿐 아니라, 서비스 대상 정보 각각에 대한 FIB 엔트리를 만들어 보관하여야 하므로 서비스 대상 정보가 폭증할 경우 라우터의 메모리에 가해지는 부담이 증가한다[10]. 또한 한 라우터에 여러 노드가 같은 콘텐츠에 대한 Interest Packet을 보낸다면 연결되어 있는 라우터에 과부하가 생길 뿐 아니라 해당 라우터가 동일한 Interest Packet 여러 개를 전달한다는 비효율성도 존재한다. 더하여 노드 내 캐시 데이터를 사용하여 콘텐츠를 전달한다고 하여도 최소 1번 이상은 해당 노드가 다른 노드로부터 데이터를 전송 받아야한다. 이때 데이터 전송은 WAN 회선을 통하여 전송되는데 다양한 타입의 많은 데이터가 전송 되므로 WAN 회선이 부족해지는 현상이 발생한다 [14]. 따라서 본 논문에서는 이러한 문제점을 해결하는 Resolution Server를 두어 효율적인 콘텐츠 분배에 대해 연구한다.

2.2.3 Content-Delivery Network과의 차이점

CDN은 사용자 주변에 있는 서버에 콘텐츠를 캐싱하고 사용자가 해당 콘텐츠를 전달 받고자 할 때, 가장 근접한 CDN Cache Server를 통해서 콘텐츠를 전달받는다. 이와 달리, CCN에서는 콘텐츠 캐싱 장소가 사용자 주변의 어떤 노드이건 관계없다. 노드가 라우터이건 기지국이건 상관없이 CCN 캐싱 메모리만 존재한다면 해당 콘텐츠를 캐싱할 수 있고, CDN보다 더 효율적이고 빠르게 원하는 콘텐츠를 전달 받을 수 있다.

참고 문헌

- [1] 장수아, 김상택. 자체 개발 콘텐츠 전송 네트워크 구축 유인 분석 (2018)
- [2] Mustafa Aljumaily. Content Delivery Networks Architecture, Features, and Benefits (2016)
- [3] Vakali, A. and Pallis, G. Content delivery networks: Status and trends. IEEE Internet Computing 7, 6 (2003), 68–74.
- [4] Ankit Desai, Jekishan K. Parmar, Sanjay Chaudhary, Content Delivery Networks : Technology Survey and Research Challenges(2015)
- [5] <https://ijbgo.tistory.com/32>
- [6] <https://dtaxi.tistory.com/1060>
- [7] Abdullaziz Yasin, A.K.M. Fazlul Haque, Sirajum Munira, Zahirul Islam. Content Delivery Network Architecture and Load Balancing
- [8] Video Content Streaming with CDN
http://www.krnet.or.kr/board/data/dprogram/846/F3-2_%B6%F3%BC%BA%C1%D6_%C3%D6%C1%BE.pdf
- [9] V. Jacobson, et al, “Networking Name Content”, in Proceedings of the 5th international conference on Emerging networking experiments and technologies, ser. CoNEXT '09. New York, NY, USA: ACM, (2009)
- [10] 이병준, 전홍석, 송호영. 정보 중심 네트워킹 연구동향(2012)
https://ettrends.etri.re.kr/ettrends/134/0905001707/27-2_080-088.pdf
- [11] 이지훈, 이주용. 이동 콘텐츠 중심 네트워크에서의 효율적인 계층적 이동성 관리 방안 (2018)
<http://jkais99.org/journal/Vol19No02/vol19no02p06.pdf>
- [12] 최상일. Contents Centric Networking 개요(2013)
<https://protocol.knu.ac.kr/tech/CPL-TR-13-01-CCN.pdf>
- [13] 김대엽, 콘텐츠 중심 네트워킹의 데이터 캐시 정책 비교 연구 (2017)
- [14] 이만용, 김영한. CCN(Content-Centric Networking) 환경에서 회선 상태를 고려한 데이터 중복 제거 전송 기술 적용(2015)
- [15] CDN 장점 <https://limelightkr.co.kr/cdn-%EC%9E%A5%EC%A0%90/>

[16] 송정환, 이문영, 권태경. 콘텐츠 중심 네트워크에서 무선 데이터 전송의 성능 분석
(2015) https://mmlab.snu.ac.kr/html/publications/docs/2015kics_jhsong.pdf