

ECTP 오류복구 성능평가

박주영* · 고석주* · 강신각*

*한국전자통신연구원

Performance Analysis of ECTP Error Control Mechanism

Juyoung Park* · Seok Joo Koh** · Shin Gak Kang*

*Electronics and Telecommunications Research Institute

E-mail : jypark@etri.re.kr

요 약

1:N 멀티캐스트 환경에서 신뢰적으로 데이터를 전송하기 위한 오류제어 메커니즘은 1:1 환경과는 달리 ACK 폭주나 중복된 재전송 데이터에 대한 고려가 반드시 필요하다. 이런 오류 제어 메커니즘들에 대한 많은 연구가 있었지만, 실제 서비스를 위해 구현된 경우는 그리 많지 않다. 본 고에서 다루는 ECTP는 현재 ITU-T와 ISO에서 국제 표준으로 승인된 신뢰적인 그룹 전송 프로토콜인데, 내부적으로는 신뢰적이면서 효율적으로 데이터를 전송하기 위해 다수의 타이머를 정의한다. 성능 평가는 RedHat7.2 환경에서 실제 구현한 ECTP의 트래픽을 다수의 파라미터들의 다양한 조합하였을 때를 인터넷 testbed상에서 이들 측정하였다. 끝으로 전송률의 개선이나 그룹 통신에 발생하는 제어 패킷들의 수는 오류 제어를 위한 각종 타이머 값이 크게 의존함을 보였다.

ABSTRACT

Reliable multicast data transmission in a 1:N environment needs more sophisticated error control mechanism than that of in 1:1 environment due to ACK implosion and duplicated retransmission. Although there have been many related research on error control in reliable multicast, real implemented protocols are rare. As one of the reliable multicast transport protocols, ECTP is selected as an international standard reliable multicast protocol by ITU-T and ISO and implemented on RedHat 7.2 machine by us. In this paper, we evaluate the performance of the error control mechanism in the respect of throughput and generated control packet numbers with a real implementation code. From the results, it is concluded that the suitable values of error control parameters can be obtained from the local group size and network environments.

키워드

reliable muticast transport protocol, error control performance

1. 서 론

인터넷 Reliable Multicast는 다수의 수신자들에게 신뢰적인 데이터 전송을 가능케 하는 메커니즘이다. 그런데 1:1 환경과는 달리 1:N환경에서는 고려되어야 할 많은사항들이 있는데, 우선 하나의 세션에 얼마나 많은 수신자들이 참가할 수 있는지에 관한 확장성 제공 방안, 흐름제어, 오류제어 등을 손꼽을 수 있다[1]. 본 고에서는 이들 고려 사항들 중에서 오류 제어메커니즘에 대한 내용을 다루었다. 인터넷에서의 오류는

전송 경로에서의 패킷 변형에 의한 손실보다는 손실에 의한 오류가 대부분을 차지하며, 이러한 오류들은 현재의 인터넷에서는 대부분 오류에 대한 재전송 요청과 재전송 방법을 사용하여 복구한다.

그런데 1:N 멀티캐스트 환경에서 오류를 복구하기 위해서는 제어 메시지에 의한 폭주(ACK implosion)방지 및 재전송 방법에 대한 세밀한 고려가 있어야 한다. ACK implosion은 다수의 수신자들이 하나의 송신자에게 제어 메시지를 동시에 보낼때 발생하며, 이때 송신자는 N개의 수신자들로부터 급작스런 제어 메시지

를 받게 된다. 이러한 ACK implosion 을 방지하기 위한 메커니즘이 각종 연구단체등에서 연구되어 왔으며, 가장 대표적인 방법들 중 하나는 제어메시지의 전달을 위한 계층적 제어 트리를 구성하여 오류 및 손실을 복구하는 방법을 들 수 있다. 또 다른 방법으로 계층적 제어 트리를 구성하는 대신 손실이 발생한 노드로부터 가장 근거리에 있는 노드로부터 복구 데이터를 수신하는 방법을 들 수 있다. 한편 손실된 데이터를 재전송하기 위한 메커니즘도 각기 다르다. 계층적 트리를 구성하는 경우 상위 부모 노드로부터 재전송을 하는 프로토콜로부터 이웃에게서 재전송 데이터를 받거나, 아예 손실이 발생하였을 경우 복구할 수 있도록 잉여 복구 데이터를 동시에 보내는 방법도 제안 되었다.

ECTP1[3]은 계층적 제어 트리를 구성하여 신뢰적 데이터를 전송하는 메커니즘을 사용하는데, 신뢰적 데이터 전송에 필요한 제어 메시지의 수를 줄이거나 보다 효율적인 데이터 전송을 위해서 사용자가 조절할 수 있는 파라미터 집합을 제공한다.

본 고에서는 전송률을 개선시키거나 그를 통신에 발생하는 패킷들의 수를 줄일 수 있는 방안을 제시하기 위하여, 실제 구현된 ECTP1 코드[4]를 인터넷 testbed 상에서 동작시키고 그 전송률과 발생된 패킷들을 측정하였다. 다양하게 조합된 파라미터들을 사용하는 ECTP1을 통해 측정된 데이터 값에 따라 파라미터간의 관계등을 분석하였다.

2절에서는 ECTP1에서 사용하는 오류제어 메커니즘에 대하여 간략한 소개를 하였으며, 시험에 사용될 시나리오 및 시험 환경을 3절에, 측정된 결과와 이를 통한 결론 및 향후 연구 방향을 제시하도록 한다.

II. ECTP1 오류 제어 메커니즘

신뢰적인 멀티캐스트 전송 프로토콜에서 오류를 복구하기 위해선 크게 송신자가 오류를 복구할 수 있도록 데이터를 전송할 때 잉여 제어 데이터를 in-band 혹은 out-band로 전송해줌으로써, 수신자가 재전송 요청을 하지 않아도 스스로 오류 복구를 할 수 있도록 해주는 메커니즘(FEC: Forward Error Correction)이나 오류가 발생하였을 경우 수신자가 송신자에게 재전송을 요청하는 일반적인 오류 복구 방식인 ARQ(Automatic Repeat Request) 방식을 많이 사용한다.

FEC방식을 사용할 경우 송신자는 수신자들을 관리할 필요나 손실을 복구하기 위한 재전송 버퍼를 관리할 필요가 없기 때문에 비교적 수신자들의 수가 큰 지역이나 Open-ended 멀티캐스트 그룹에서 사용되어진다. 그런데 이 방식은 수신자들이 오류를 복구할 수 있

도록 잉여 데이터를 전송해야 한다는 것과 복구를 위한 복잡한 알고리즘이 필요하다는 단점도 있다.

반면에 ARQ방식은 송신자가 데이터를 제대로 수신했는지에 관한 수신자 상태 정보를 관리해야하며, 아울러 재전송을 요청하였을 때, 데이터를 재전송해 주기 위한 재전송 버퍼를 관리해야 한다는 단점이 있지만, 자동 오류 복구를 위한 잉여 데이터를 보낼 필요가 없기 때문에, 단지 순수 데이터만을 전송하면 된다. 또한 불필요한 재전송을 막기 위한 정보관리를 통하여 수신자에 대한 관리가 용이하다.

ECTP1프로토콜은 신뢰적인 전송을 제공해주는 멀티캐스트 전송 프로토콜로써 오류를 복구하기 위해서 ARQ방식을 사용한다. 그런데 ARQ방식을 사용할 경우 다수의 수신자들로부터 동시에 재전송 요구등의 제어 메시지가 올수 있기 때문에 ACK implosion의 문제가 발생할 수 있다.

이러한 ACK implosion을 방지하기 위해서 ECTP1은 다음의 3가지 방법을 이용하여 동일 데이터 패킷에 대한 ACK를 최소화할 수 있는 방법을 제공한다.

■AGN (Ack Generation Number)

ECTP1의 모든 수신자들은 송신자로부터 고유한 CID(Child ID)를 할당받는데, ECTP1 송신자는 매핑된 CID와 수신자의 주소(IP주소와 Port) 정보를 관리함으로써 수신자들은 그 세션에서 고유한 CID를 할당해줄 수 있다.

ECTP1 세션은 임의의 AGN값을 갖는다. AGN값은 수신자들에게 언제 ACK를 보낼 수 있는지를 결정할 수 있게 하는데, 이 값을 통하여 ACK implosion을 줄일 수 있다. AGN을 통한 ACK발생 규칙은 다음과 같다.

만일 패킷의 시퀀스 번호를 세션의 AGN값으로 나눈 나머지와 시퀀스 번호를 수신 노드 고유의 아이디인 CID로 나눌때, 동일한 값을 나타낼 경우 ACK를 보내도록 한다.

만일 수신자가 3이고, AGN값이 8일 경우, 전송되는 데이터 수에 대해 발생하는 ACK 패킷의 수는 3/8이 되며, 수신자가 3이고 AGN값이 1일경우 동일 데이터 패킷 수에 대하여 발생하는 ACK패킷의 수는 3배가 될 것이다.

■RXT(Retransmission Time)

ARQ에서 만일 어떤 데이터 패킷에 대한 일정한 시간 이내에 ACK가 도달하지 않을 경우 이를 ACK의 손실 혹은 전송한 데이터의 손실로 판단할 수 있어야 한다. ECTP1에서는 송신자가 RXT 시간 동안 수신자들로부터 ACK를 받지 못했을 경우 이를 손실로 판정하고 해당 패킷들을 재전송한다.

■RBT (Retransmission Backup Time)

ECTP1에서는 AGN을 이용하여 ACK의 수를 제한한다. 그런데 다수의 수신자들로 구성된 세션에서 데이터의 전달되는 중에 ACK를 발생함으로써 수신된 데이터를 제대로 반영하지 못하는 경우가 발생할 수 있다.

RBT 시간은 전송되는 도중에 ACK가 세션 전체에 올바르게 수신한 패킷에 대한 정보를 반영하지 못하는 경우를 완충하기 위한 파라미터로써, 비록 재전송 요청이 오더라도 해당 패킷을 전송한지 RBT시간 이내 일 경우 재전송하지 않는다.

이 메커니즘을 통하여 송신되고 있는 데이터에 대한 불필요한 중복 재전송을 막을 수 있다.

III. 성능 측정 시나리오

1) 측정 환경

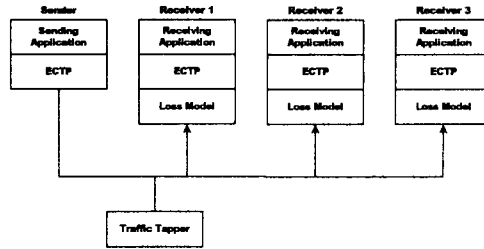
신뢰성을 보장하는 멀티캐스트 전송 프로토콜인 ECTP1의 성능을 측정하기 위해서 손실이 발생하는 환경의 구축이 필요한데, 기존의 방식들은 대부분 시뮬레이터를 사용하여 측정 및 테스트를 하고 있다. 그러나 현재 구현한ECTP1[4]의 경우 리눅스 사용자 레벨 계층의 프로토콜 모듈로써 동작하기 때문에 시뮬레이터로 포팅이 용이하지 않았다.

따라서 다음 그림 1에서 보이는 바와 같이, 패킷이 ECTP1에 도달하기 전에 각각의 수신자들의 "loss model" 모듈에 의해 임의의 손실을 발생할수 있도록 설계하였다.

만약 손실 모듈을 송신측에 추가할 경우, 모든 수신자들은 공통된 패킷들을 받을 수 없게 되는데, 이것은 송신자와 수신자들간의 라우터에서 발생할 수 있는 손실을 적절하게 시뮬레이트할 수 없다는 단점이 있다. 따라서 임의의 경로를 통해 도달한 패킷들에 대한 망 환경을 시뮬레이트하기 위해선 수신자단에서의 손실 모델을 적용하였다.

일정 손실률을 갖는 랜덤 손실을 발생하기 위해선 각 수신자측의 손실 발생 순차가 동일해서는 안되기 때문에, 각 손실모델의 확률 계산에 사용되는 랜덤 시드 값을 유일한 값으로 설정하였다.

송수신되는 모든 패킷들의 시간과 종류, 크기등을 ECTP1 패킷들을 위해서 별도로 구현한 Traffic Tapper를 통해 기록하였다.



[그림 1] 측정 환경

2) 시나리오

손실 환경에서의 ECTP1 성능을 측정하기 위해서 패킷들이 전달되는 시간과 종류, 크기들을 기록한다. 본 손실 환경에서 ECTP1의 성능을 고려하기 위한 패킷 종류로써 DT, AK, RD 만을 관찰하기로 한다. 다른 패킷들 종류들 중에서 이들 3 종류의 패킷들만을 고려하는 이유는, 손실이 발생했을 경우 데이터의 복구 요청(AK)와 재전송 데이터(RD)를 측정함으로써 얼마나 빠르게 손실을 복구하는지를 판단하기 위해서이다.

실험은 각기 다른 손실률을 부여했을 때, 송신자로부터의 동일한 크기와 개수의 데이터를 다수의 수신자들에게 얼마만에 전달할수 있는가를 측정한다. 측정 결과의 해석은 데이터 전송이 빠른 시간에 종료될 경우, 데이터의 손실 복구가 잘 됨으로써 전송이 완료되었다고 해석할 수 있으며, 반대로 데이터 전송의 완료가 오래 걸릴 경우, 데이터 손실에 따른 복구 작업에 많은 시간이 소모되었기에 전송 완료가 늦어진 것으로 해석할 수 있다.

■손실률

손실환경에서의 동작을 측정하기 위해 다음과 같은 손실 조건을 부여하도록 한다.

1. 0% 손실
2. 2% 손실
3. 5% 손실
4. 10% 손실

각 손실 환경에서의 다음 파라미터값에 대한 전송 시간을 측정한다.

■파라미터 선정

다음과 같은 ECTP1 파라미터를 사용하여 ECTP1 파라미터를 조합 설정한다.

1. AGN= 1, 3,5,16
2. RBT = 600ms, 1000ms, 2000ms
3. RXT = 600ms, 1000ms, 2000ms

IV. 측정 결과

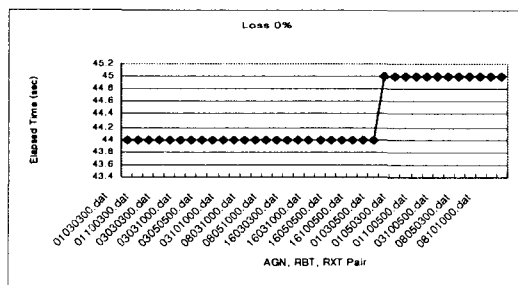
측정된 트래픽은 다음과 같은 형식으로 파일에 저장된다.

TIME	SrcAddr	DstAddr	(Sequence : PktType : Length in bytes)
04.794943	129.2.3.30:9090	-> 239.1.2.3:9090	(7352 : 1 :32)
04.795004	129.2.3.30:9090	-> 239.1.2.3:9090	(7352 :10:56)
04.799681	129.2.3.32:9090	-> 129.2.3.30:9090	(4850 : 3 :24)
04.799685	129.2.3.31:9090	->129.2.3.30:9090	(7239 : 3 :24)
04.799689	129.2.3.33:9090	->129.2.3.30:9090	(2624 : 3 :24)
04.803871	129.2.3.30:9090	->129.2.3.32:9090	(7352 : 4 :44)
. . . .			

측정은 0%, 2%, 5% 및 10% 손실 환경에서 데이터가 완전히 전달될 때까지의 전송되는 시간에 대하여 비교를 하도록 하였다. 실험을 위해서 송신자는 1000 bytes 의 2000개 데이터 패킷을 매 20msec마다 전송하도록 하고, 각 파라미터의 조합을 갖는 환경에서 전송완료 시간을 측정하였다.

아래의 그래프의 x축은 AGN, RXT, RBT, 및 Loss 를 표시한 것으로써, 만일 01100500.dat는 AGN=1, RXT= 10(*200ms), RBT = 5(*200ms), loss = 0%라는 의미이다. y축은 2000개의 패킷을 전달되는데 소요되는 시간을 sec단위로 표시하였다.

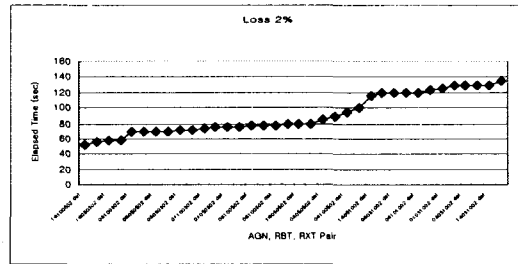
[그림 2]는 0%의 손실 환경에서의 측정된 값이다. 세션 설정단계에 약 3초를 사용한 후로부터, 2000개 패킷이 모두 전달되기 위한 시간은, 약 41초 ~44초 범위에 속하였다.



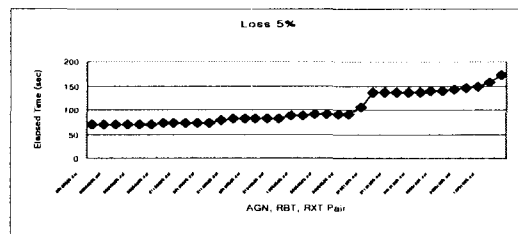
[그림 2] 0% 손실 환경에서의 전송 시간

[그림 3]은 3개의 수신자들이 각각 독립된 2%의 랜덤 손실이 발생했을 경우의 데이터 전송 완료 시간을 보인 그래프이다.

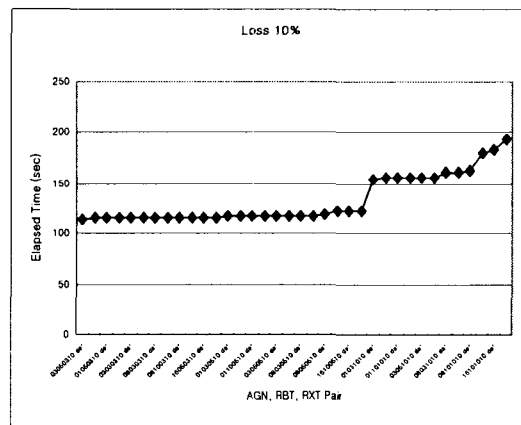
그림에서 보느냐와 같이 동일한 손실률임에도 불구하고, AGN, RBT, RXT시간의 설정에 따라서 각기 다른 성능을 갖는 것을 볼 수 있다. 마찬가지로 [그림4]와 [그림5]에서도 동일한 양상이 나타남을 확인할 수 있다.



[그림 3] 2% 손실 환경에서의 전송 시간



[그림 4] 5% 손실 환경에서의 전송 시간



[그림 5] 10% 손실 환경에서의 전송 시간

V. 결론 및 향후 연구

지금까지 4개의 손실확률에서 각기 다른 파라미터 값을 갖는 ECTP1을 사용하여 2000개의 1000바이트 데이터를 20msec마다 전송할 때 소요되는 시간을 측정함으로써, 각 파라미터의 상관성과 ECTP1의 성능을 평가하였다.

손실이 없을 경우, 데이터의 전송률에 거의 상응하는 성능을 보인 것과, 손실이 비교적 작을 경우일지라도 재전송 타임(RXT)이 클 때 전송률이 나빠질 수 있음을 보였다. 반면에 [그림 5]에서와 같이 재전송 타임이 너무 작을 경우 어느 적절한 경우의 것보다 전송률

이 나빠짐을 볼 수 있다.

앞에서의 고찰을 통하여 전송률은 손실률 등과 같이 단지 어느 한 파라미터에만 의존하는 것이 아니라 다른 파라미터들(AGN, RXT, RBT)과도 밀접한 관계가 있음을 보였으며, 손실률에 따른 적절한 다른 파라미터들의 선별과 손실률에 따른 적절한 파라미터 변경 등의 작업이 추가적으로 필요할 것으로 전망된다.

참고문헌

- [1] IETF RMT WG,
"http://www.ietf.org/charter/rmt-wg.html"
- [2] ISO605|ITU 13252, ECTS(Enhanced
Communications Transport Services)
- [3] ITU-T 14476-1, ECTP1 Part1
- [4] ECTP Official Homepage,
"http://ectp.etri.re.kr"