

LAN 환경에서 Topology diameter 에 따른 네트워크 성능분석

김진희*, 한설흠*, 권경희*
*단국대학교 전자계산학과
e-mail : whitej2@dankook.ac.kr

Analysis of Network Performance according to Topology diameter of LAN

Jin-Hee Kim, Seol-Heum Han, Kyung-Hee Kwon
*Dept. of Computer Science, Dan-Kook University

요 약

인트라넷은 TCP 를 지원하는 기존의 LAN 환경에서 인터넷 기술을 그대로 채택하였다. 이는 인터넷 기반의 TCP Parameter 값을 그대로 인트라넷에 적용하였음을 의미하며 서로 다른 환경에서 같은 Parameter 값의 적용은 네트워크 성능 저하의 원인이 되기도 한다. 네트워크의 성능을 확인하는 방법으로 Rtt(Round Trip Time:응답시간)와 Thput(Throughput:처리률)이 있다. 이값은 전송 거리와 트래픽 양에 따라 다르게 나타난다. 따라서 본 논문에서는 LAN 기반으로 전송거리에 따른 즉, Topology diameter 와 TCP Parameter 와의 관계를 RTT 와 Thput 을 통해 분석해 보았다.

1. 서론

최근 몇 년 동안 인터넷 수요의 증가는 인터넷 서비스의 다양화를 가져왔고, 이러한 서비스는 기업내 인트라넷 환경으로 까지 연결 되게 되었다. 많은 기업들이 이미 인트라넷을 사용하고 있으며 인터넷을 사용하는 기업들중 80%이상이 인트라넷의 도입계획의사를 밝혔다고 한다[3]. 인터넷 서버 시장도 단순 회사 홈페이지 제작 수준이 아니라 인트라넷 기반하에서 전자메일, 자료실, 게시판, 원격회의 그리고 데이터베이스 이용과 온라인 교육에 이르기까지 효율적인 업무 수행을 가능하게 해주고 있다.

인트라넷은 TCP 를 지원하는 기존의 LAN 환경에서 인터넷 기술을 그대로 채택하였다. 이는 TCP 연구가 인터넷과 함께 하였으며 그것이 그대로 인트라넷에 적용되었다는 것을 의미한다. 그러나 일반적으로 네트워크의 성능을 평가하는데 이용되는 RTT(Round Trip Time: 응답시간)와 thput 은 인터넷 환경과 인트라넷 환경에 따라 많은 차이를 가져다 준다.

RTT 는 클라이언트가 서버에 요청을 하고 그 요청에 서버의 대답을 주기까지의 시간을 뜻하는 것으로 거리가 길면 당연히 RTT 값은 길어지고 거리가 짧으면 RTT 도 작아지는 것이다. RTT 값은 거리에 비례한다고 할 수 있는 것이다[4]. 또한 RTT 는 트래픽의 양과 무관하지가 않다. 트래픽은 클라이언트 서버간의 송수신되는 패킷 중 재전송으로 인해 만들어지는 패킷들과 세그먼트의 단편화(Fragmentation)로 오버헤드가 생기면서 늘어날 수 있다. 재전송과 패킷의 단편화는 인터넷상의 많은 라우터를 거치면서 생길 수 있다. 따라서 패킷 손실 및 전송지연등은 LAN 기반의 인트라넷보다는 많은 라우터와 전송거리가 긴 WAN 기반의 인터넷 환경에서 더 많이 나타나는 것이다. 그럼에도 불구하고 기업내 인트라넷 시스템의 TCP 설정이 WAN 을 대상으로 설정되어[1][2] 있다는 것은 성능 위주의 네트워크 설계가 요구되는 시점에서 반드시 수정되어야 할 사항이다. 앞서 RTT 는 거리와 트래픽의 양에 따라 달라질 수 있다고 하였다. 인트라넷상에서 인터넷 기반으로 설정된 TCP

Parameter 값 역시 패킷 전송거리와 트래픽의 양에 따라 다르게 적용되어야 하는 것이다. 따라서 본 논문에서는 전송 거리에 따른 적합한 TCP Parameter 값을 찾아 네트워크 성능을 향상시키는 것을 목적으로 한다.

2. 관련 연구

2.1 네트워크의 성능 향상을 위한 TCP Parameters

서론에서 이야기 하였듯이 TCP 네트워크 환경에서 Parameter 의 값은 인트라넷 기반의 LAN 보다는 인터넷 환경의 WAN 을 기본으로 설정되어 있다.

네트워크 성능에 영향을 미치는 TCP Parameters 이다.

- MSS(Maximum Segment Size)
- RTO(Retransmission Time Out: 재전송 타임아웃)
- Window Size
- Receiver Buffer Size
- TCP Time Wait
- Delay Acknowledgement

위 parameter 들 중 전송 거리에 영향을 받을 수 있는 것으로 RTO 와 RcvBuffer(Receiver Buffer) 를 들 수 있다.

RTO 란 패킷이 분실되었거나 전송 지연으로 인하여 주어진 시간 내에 패킷이 도착하지 않을 경우 재전송 타임 아웃을 의미한다. 패킷을 실제로 잃어버렸을 경우 이 시간이 지나치게 길면 재전송 또한 늦게 이루어지고 데이터 수신 또한 지연되며, 지나치게 짧게 설정되면 잦은 재전송으로 인하여 오히려 트래픽을 유발 네트워크의 과부하를 조장하게 된다. 따라서 적합한 RTO 값은 전송 거리에 따라 다르게 설정되어야 하는 것이다.

RcvBuffer 와 전송 거리는 $delay * Bandwidth$ (대역폭)의 관계로 설명할 수 있다

TCP 연결에서 데이터의 량은 수신측의 RcvBuffer size 에 의해서 제한을 받는다. 이때의 RcvBuffer 란 수신측에서 수신 가능한 Buffer size 를 뜻하는 것으로 수신자의 RcvBuffer Size 가 얼마나 커야 하는가?는 네트워크의 대역폭이 아니고 $delay * 대역폭$ 으로 설명한다. RcvBuffer 크기에 따라 전송 가능한 송신 데이터의 양이 달라지고 네트워크의 Thput 도 달라지게 되는 것이다. delay 는 보통 전송 지연 또는 소요시간으로 설명된다.

$$소요시간 = 파시간(거리/광속) + 전송시간 + 큐잉시간$$

이때 단일 링크에서 작동하는 환경이라면 전송 시간과 큐잉 시간은 문제가 되지 않고, 소요 시간은 단지 전파 시간에만 영향을 받는다. 즉 delay 는 전송 거리로 설명할 수 있는 것이다.

따라서 RcvBuffer 가 전송 거리와 상관없이 크게만 설정되면 패킷 드랍(drop)은 줄일 수 있으나 버퍼 안에서의 대기 시간이 증가, 곧 네트워크 상에서 패킷의 지연 시간이 더 증가됨을 의미한다.

네트워크 상에서 전송 거리가 다르고 같은 대역폭을 갖는 경우라면 수신자의 RcvBuffer Size 와 RTO 값은 전송 거리에 영향을 받는다.

3. 네트워크 구성

3.1 네트워크 구성

본 논문에서는 네트워크 시뮬레이터인 ns-2 를 이용하였으며, 시뮬레이션 환경에서 다음과 같은 가정 하에 연구를 진행하였다.

첫째, TCP 타입은 TCP Reno 이다. 이는 TCP 의 여러 가지 구현 중 대표적이며 가장 많이 이용되는 것이 Reno 이기 때문이다.

둘째, ACK 에 대한 손실률을 0%로 하였으며, 수신측에서부터 ACK 가 전송되는 동안에 error 가 없음을 가정한 것이다.

셋째, 데이터 송수신 방법은 반이중(half-duplex)으로 가정하였다.

시뮬레이션 환경은 LAN 으로 하였다. LAN 환경에 적합하고 전송 거리에 따라 적합한 TCP parameter 값을 찾아보기 위함이며, LAN 또는 확장된 학교 크기의 LAN 에서 delay 는 20ms 의 값이 적당하다고 한다[4]. 시뮬레이션 상에서의 거리는 관련 연구를 통해 설명하였듯이 delay 로 설명되며, delay 에 따른 적합한 RTO 값과 RcvBuffer 의 값을 구할 수 있다.

Maximum packet size		1000 bytes
TCP type		FTP
Client	TCP Type	TCP Reno
	Node	10 개
Server	TCP Type	TCP Reno2 (RTO 값 고정)
	Node	1 개
Bandwidth		10 Mbps
Delay		0.1~20 ms
Queue		DropTail

[표 1] 시뮬레이션 모델

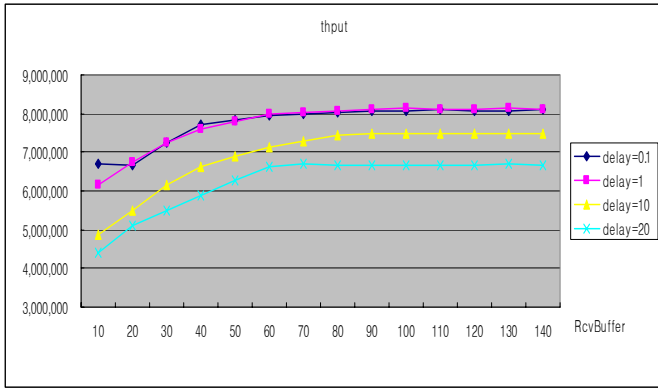
[표 1]에서 TCP 프로토콜 Type 을 클라이언트와 서버에 각각 다르게 설정하였다. Server 측의 프로토콜 type 은 기존 재전송 타임아웃 알고리즘을 적용하지 않고 수정된 알고리즘(TCP Reno2 : RTO 값 고정)을 적용하였다.

3.2 실험 결과

3.2.1 delay 와 RcvBuffer

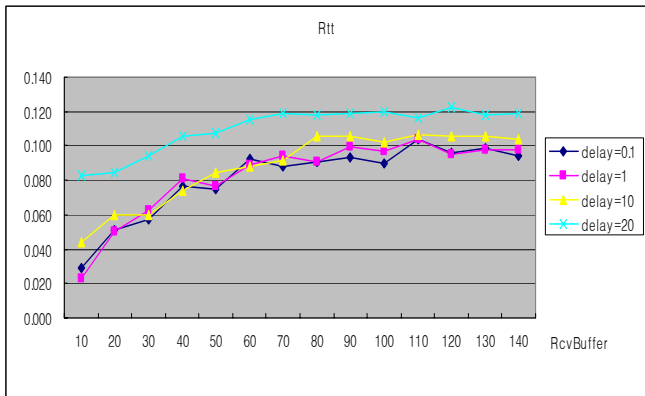
LAN 의 최대 크기를 20ms(delay) 로 하고 0.1ms 에서 20ms 까지 거리에 따른 RcvBuffer(패킷수)의 관계를 시뮬레이션을 통해 얻은 결과이다.

RcvBuffer Size 60 부터 140 까지 처리율의 변화가 거의 없고 10 에서 60 까지 급격하게 처리율이 증가하는 것을 확인 할 수 있다. 이는 LAN 에서 적합한 RcvBuffer 크기가 40~60 임을 의미한다.



[그림 1] Throughput

[그림 2]는 RcvBuffer Size 가 작으면 네트워크상에서 지연 시간이 짧아지고 이는 곧 Rtt 값이 작아짐을 의미하며 RcvBuffer Size 가 커지면 Buffer 안에서의 대기시간이 많아지므로 Rtt 값은 커지게 되는 것이다.



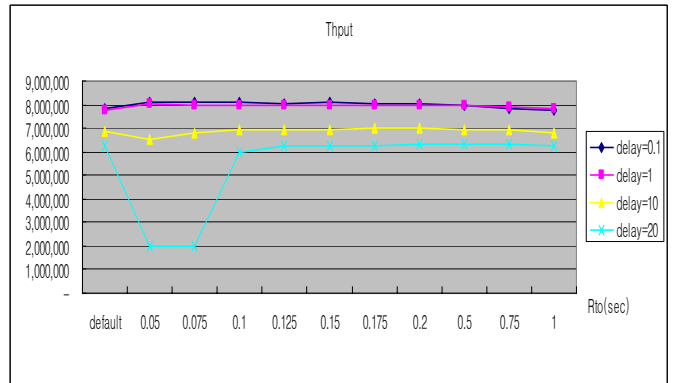
[그림 2] Rtt

[그림 1]과 마찬가지로 60~70 정도부터 Rtt 값 역시 거의 변화가 없음을 알 수 있다. 앞서 RcvBuffer=delay * Bandwidth 라고 하였다. RcvBuffer Size 를 delay 와 Bandwidth 의 관계를 고려하지 않고 크게 설정하면 Rtt 나 Thput 에 영향을 미치지 보다는 자원낭비로 결부되는 것이다. 위 두가지 Thput 과 Rtt 를 delay 에 따른 RcvBuffer 의 관계로 분석해보면 delay 가 0.1ms 일때는 RcvBuffer 값이 40, 1ms 일때는 50, delay 10~20ms 일 경우 60 이 가장 적합한 것으로 확인 할 수 있다.

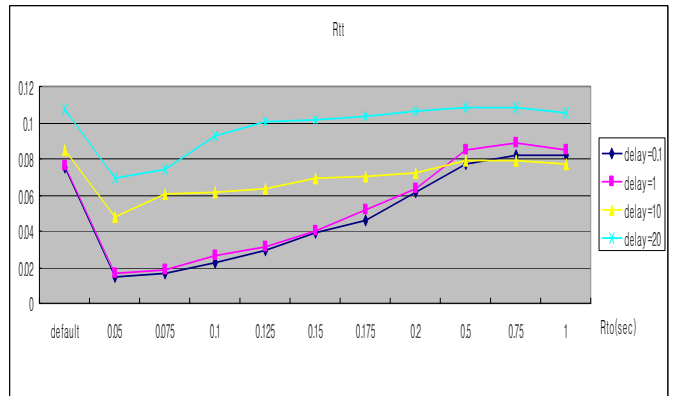
3.2.2 delay 와 Rto

그래프에서 default 란 기존 알고리즘을 적용했을 때를 의미하며 Rto 값을 0.025(sec) 로 고정시켜 테스트하고 0.025 간격으로 해서 1 (sec) 일때까지 결과이다. LAN 에서 Thput 이 가장 높을 경우와 Rtt 값이 가장 작을 경우를 [그림 3]와 [그림 4]를 통해 확인해 보면 Rto 값이 0.05 에서 0.125 인 경우로 확인 할 수 있다. 실제 테스트는 0.025 도 해 보았으나 지나치게 잦은 재전송으로 패킷 손실이 큰 것을 확인하였다.

delay 에 따른 적합한 Rto 값은 0.1ms 일 경우 Rto 는 0.05(sec), 1ms 는 0.075(sec), 10ms 는 0.1(sec) 그리고 delay 20ms 일 경우 Rto 는 0.1~0.125(sec) 이 적합한 값으로 확인 되었다.



[그림 4] Throughput



[그림 5] Rtt

4. 결론

네트워크 Topology diameter, 즉 delay 와의 관계로 설명될 수 있는 것으로 TCP Parameter 중에 RcvBuffer 와 Rto 가 있다.

시뮬레이션 결과 네트워크 Topology diameter 에 따라 적합한 Parameter 값을 그리고 성능개선을 확인할 수 있었다. Rto 의 경우에는 기존 알고리즘을 그대로 적용하였을 경우보다 많은 성능 개선을 가져왔으며, LAN 환경내에서도 전송거리에 따라 다른 Rto 값과 RcvBuffer Size 가 적용되어야 함을 확인하였다.

따라서 TCP Parameter 값이 네트워크 diameter 와 관계없이 획일화 되어 적용되고, 많은 기업이 인트라넷 도입 등을 고려하는 시점에서 Topology diameter 에 따라 적합한 Parameter 값의 적용을 고려해 보아야 할 것이다.

참고문헌

[1] 김진희, 권경희 “재전송 타임아웃 간격의 범위 조절에 의한 Web 서버의 성능향상”, 정보처리학회 추계학술대회, 제 9권 제 2호, 2002

- [2] 전철완, 권경희 “윈도우 사이즈 조절을 통한 내부 네트워크의 성능향상”, 한국정보처리학회 2003 춘계학술대회”, 제 10권 제1호, pp 1189 -1192
- [3] <http://www.cs.columbia.edu/~hgs/internet/traffic.html>
- [4] Larry L.Peterson, Bruce S. Davie, “Computer Networks”, Morgan Kaufmann, 1999, pp. 22-28, 455,
- [5] Gary R. Wright W. Reichard Stevens, “TCP/IP Illustrated Volume2(1)”, Addison Wesley, 2000.
- [6] Chadi Barakat, Eitan Altman, and Walid Dabbous, “On TCP Performance in a Heterogeneous Network”, IEEE Communications Magazine, pp 40-46, January. 2000.
- [7] David M. Nicol, “Comparison of Network Simulators Revisited”, Dartmouth College , May 20, 2002.
- [8] Hung-Ying Tyan and Jennifer Hou. Design, realization, and evaluation of a component-based compositional network simulation environment. *In Proceedings of the 2002 Communication Networks and Distributed Systems Modeling and Simulation Conference*, San Antonio, Texas, January 2002. Society for Computer Simulation.