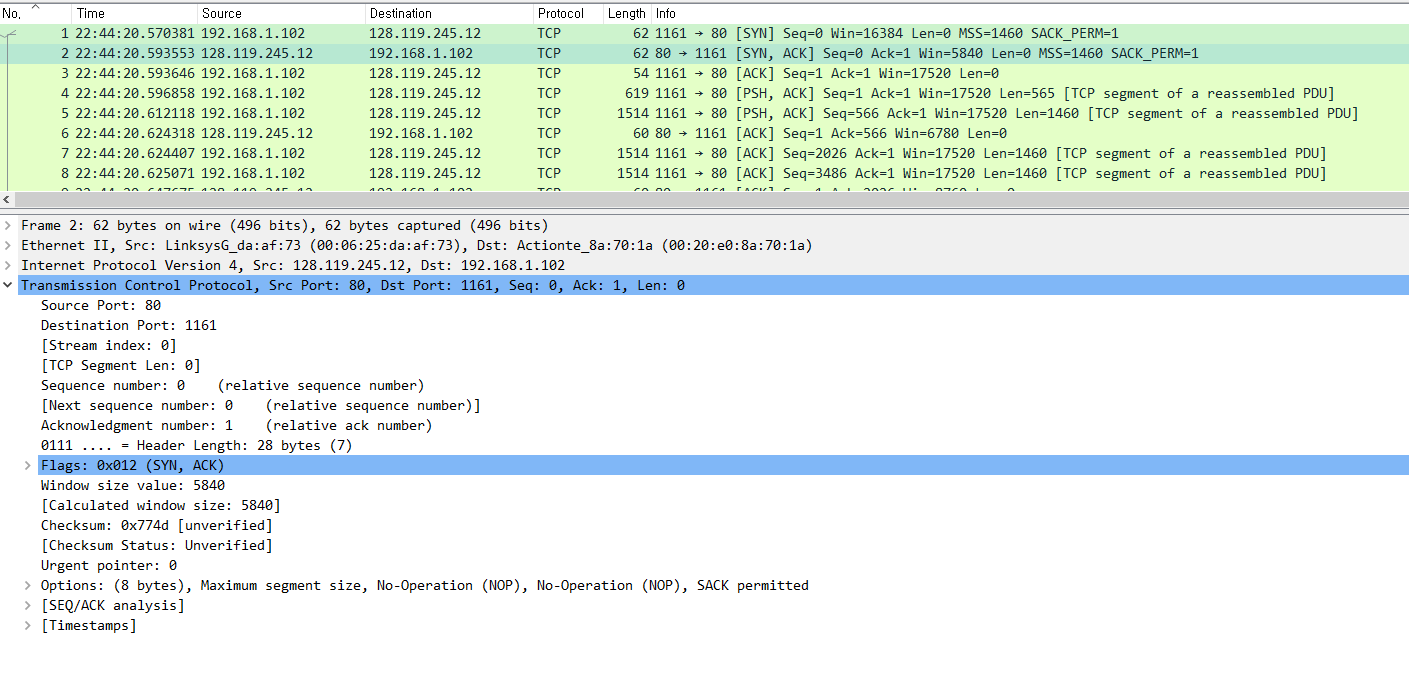
1. gaia.cs.umass.edu로 파일을 전송하는 클라이언트 컴퓨터(소스)에서 사용하는 IP주소와 TCP포트 번호는 무엇인가요?



위의 사진에서 보면 웹페이지 연결하기 위해 TCP 3-way Handshaking을 1~3번에서 실행하고 있다.

따라서 호스트는 1번의 Source이며 서버는 Destination임을 유추할 수 있다.

IP주소는 192.168.1.102 이며 TCP 포트번호는 1161이다.

2. gaia.cs.umass.edu의 IP주소는? 어떤 포트 번호에서 이 연결을 위해 TCP 세그먼트를

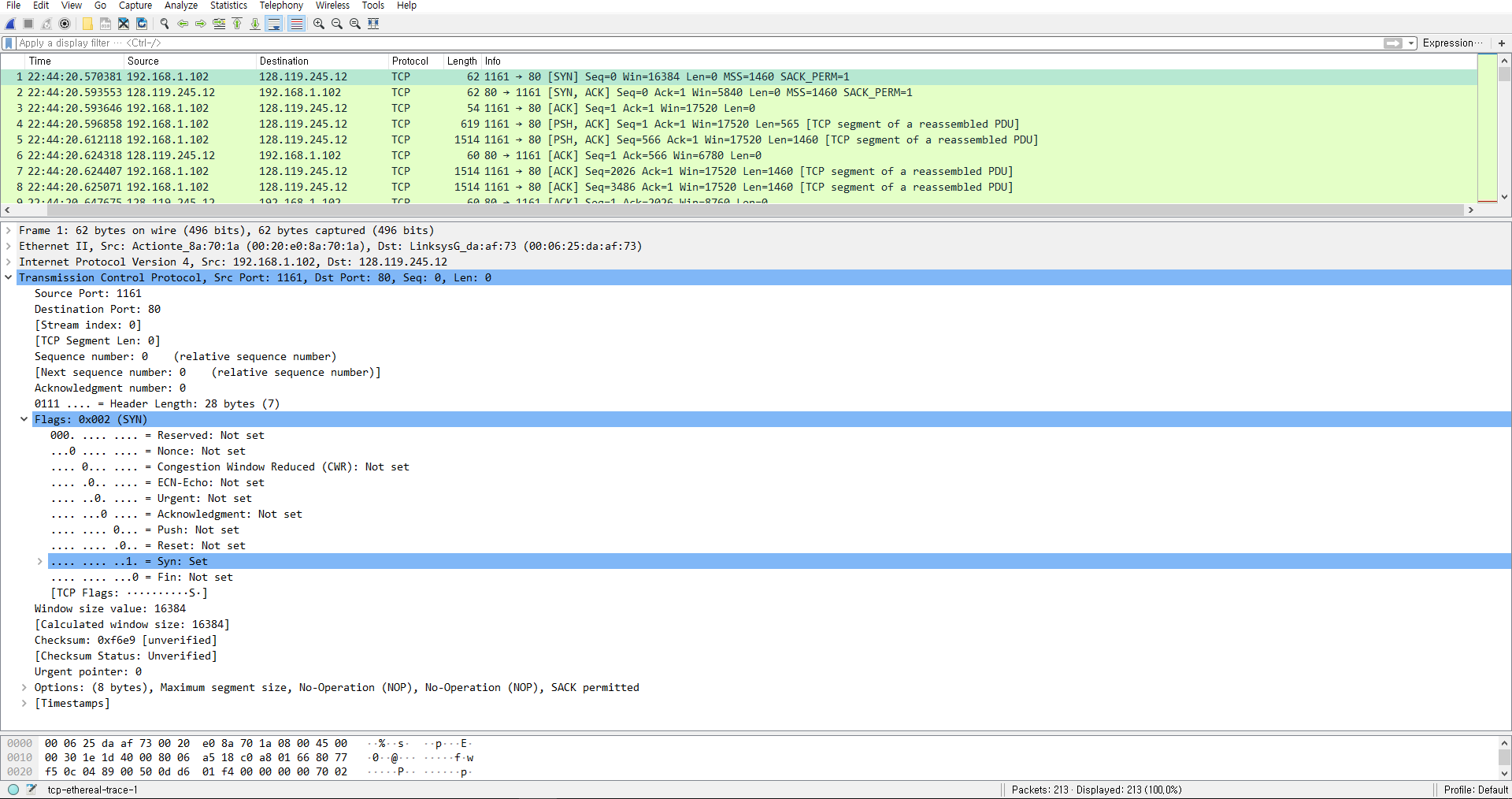
송수신 중인가요?

Gaia.cs.umass.edu의 IP 주소는 128.119.245.12이며 TCP 포트번호는 80번이다.

4. 클라이언트 컴퓨터와 gaia.cs.umass.edu사이의 TCP연결을 초기화하는데 사용되는

TCP SYN 세그먼트의 시퀀스 번호는 무엇인가요? 세그먼트를 SYN세그먼트로 식별하는

세그먼트의 내용은 무엇인가요?



위의 그림에서 보이다시피 시퀀스 넘버는 0이며 Flags에서 SYN 필드가 set 되어 1비트 처리되어

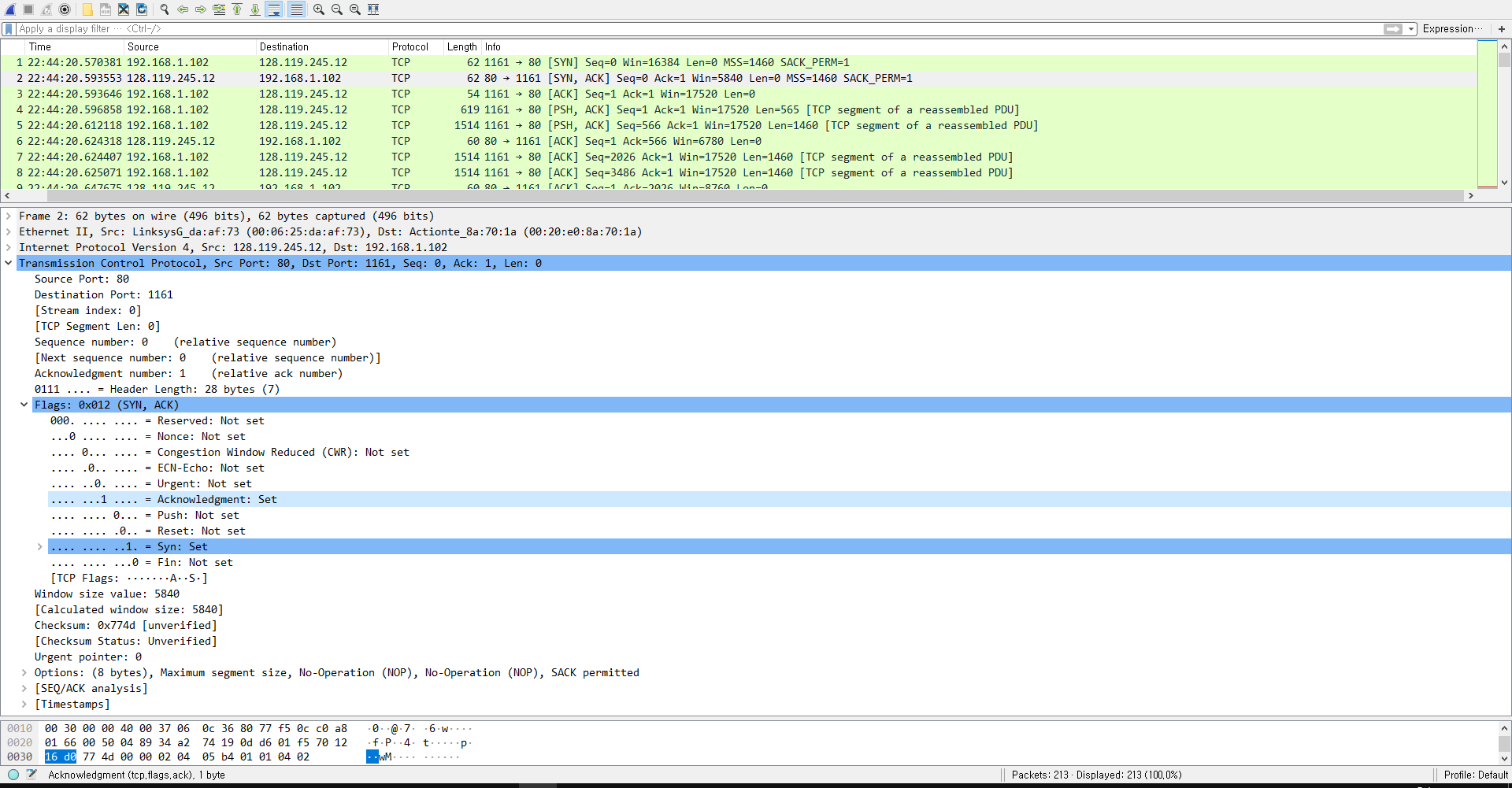
SYN세그먼트로 식별할 수 있다.

5. gaia.cs.umass.edu가 SYN에 응답하여 클라이언트 컴퓨터에 보낸 SYNACK 세그먼트의

시퀀스 번호는 무엇인가요? SYNACK 세그먼트의 Acknowledgement필드 값은 무엇인가요?

gaia.cs.umass.edu가 그 값을 어떻게 지정했습니까? 세그먼트를 SYNACK세그먼트로

식별하는 세그먼트의 내용은 무엇인가요?



위의 그림에서 보이다시피 시퀀스 넘버는 0이며 Acknowledgement 필드 값은 1이다.

3-way Handshaking에서 ACK 번호는 전송된 Seq number에서 1을 더한 값이며

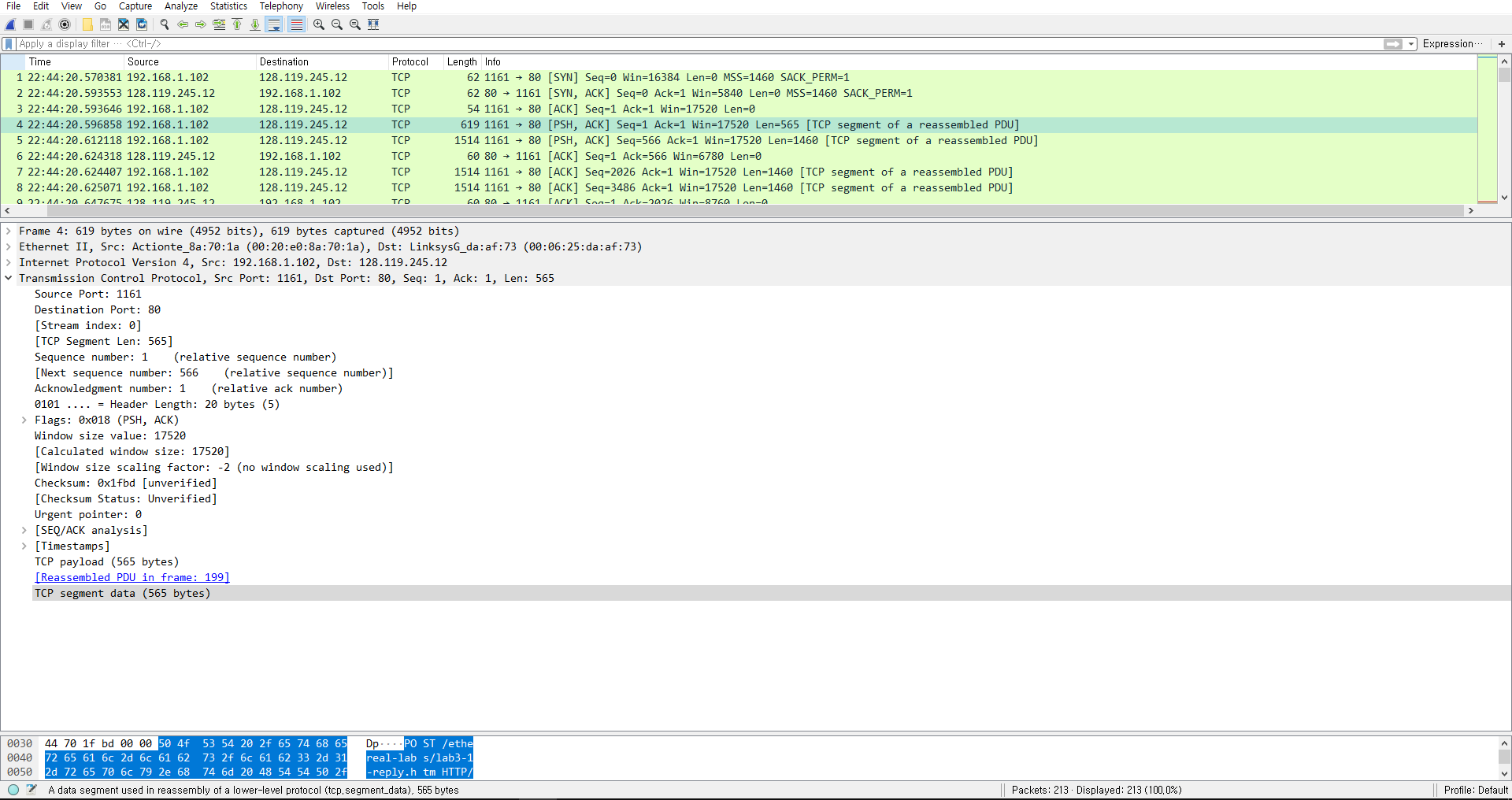
Flags에서 SYN 필드와 Acknowledgment 필드가 set 되어 1비트 처리되어

SYNACK세그먼트로 식별할 수 있다.

6. HTTP POST명령을 포함하는 TCP세그먼트의 시퀀스 번호는 무엇인가요?

(힌트: POST명령을 찾으려면 Wireshark하단의 패킷내용필드를 확인하여 DATA필드 내에 있는

“POST”가 있는 세그먼트를 찾는다.)

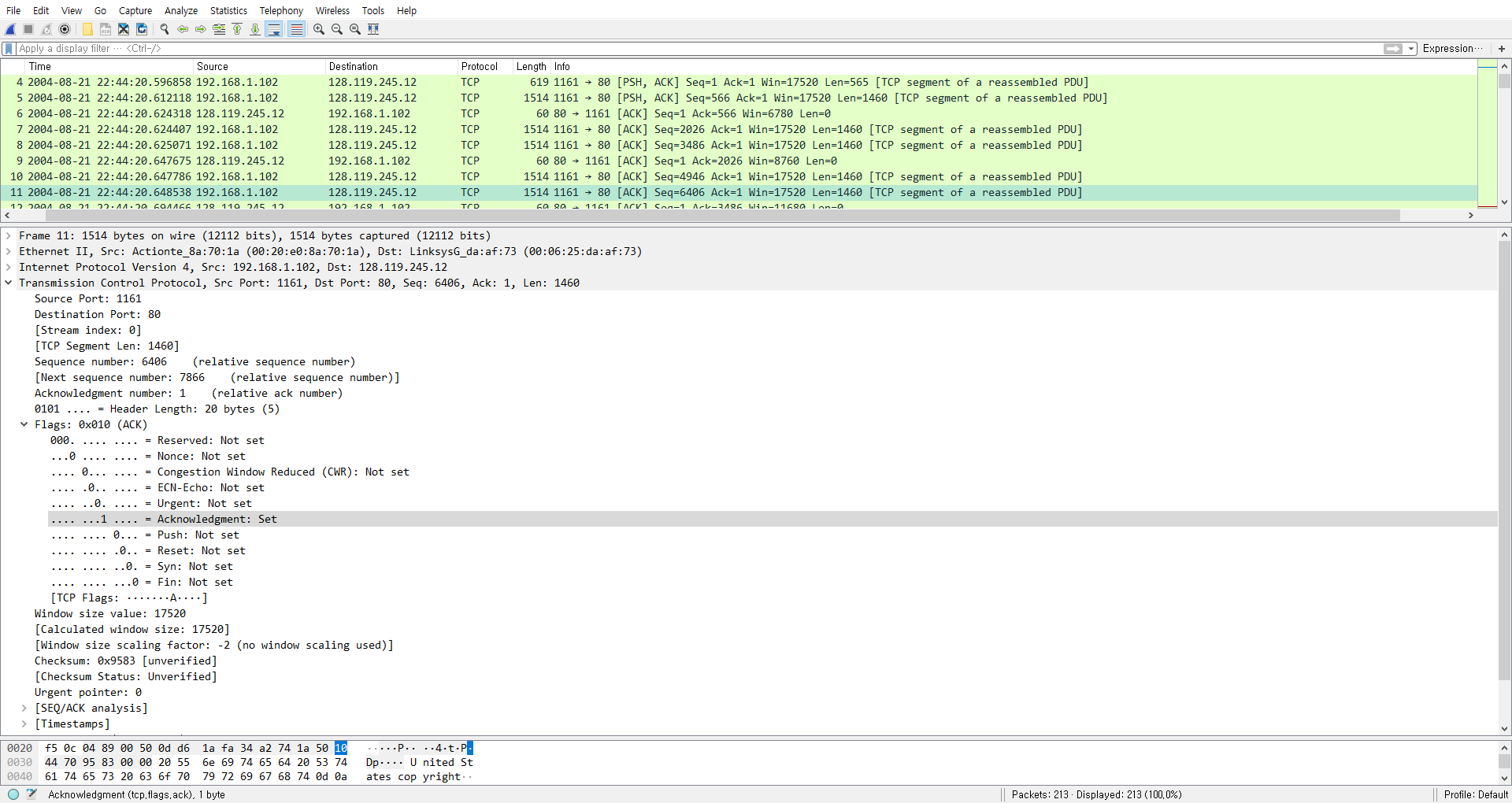


HTTP POST 명령을 포함하는 TCP 세그먼트의 시퀀스 번호는 1이다.

7. TCP 연결 상의 첫번째 세그먼트로 HTTP POST를 포함하는 TCP세그먼트로 간주하시고 답하세요.

1)TCP연결 (HTTP POST포함된 세그먼트 포함)의 처음 6개의 세그먼트의 시퀀스 번호는

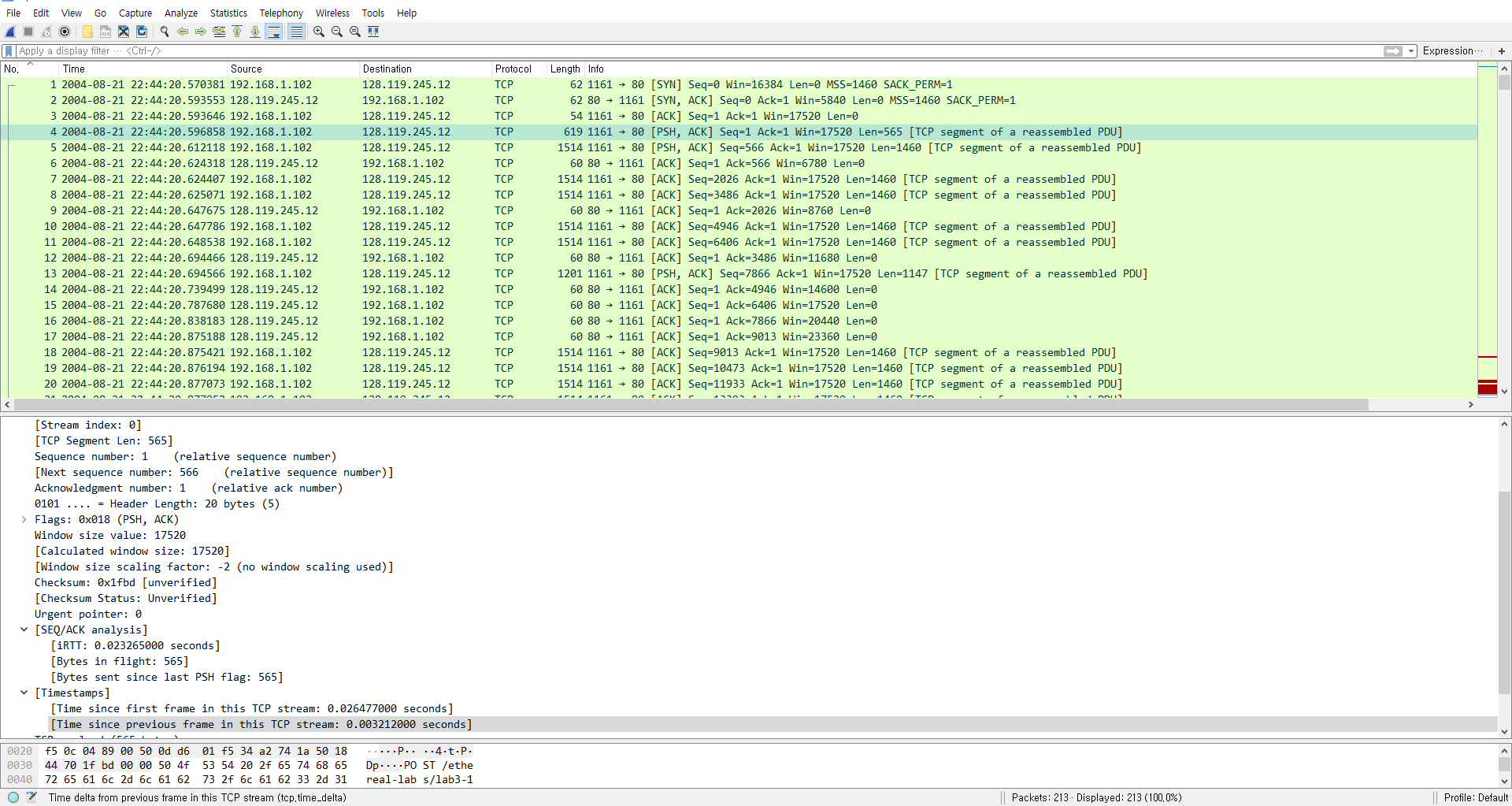
무엇인가요?



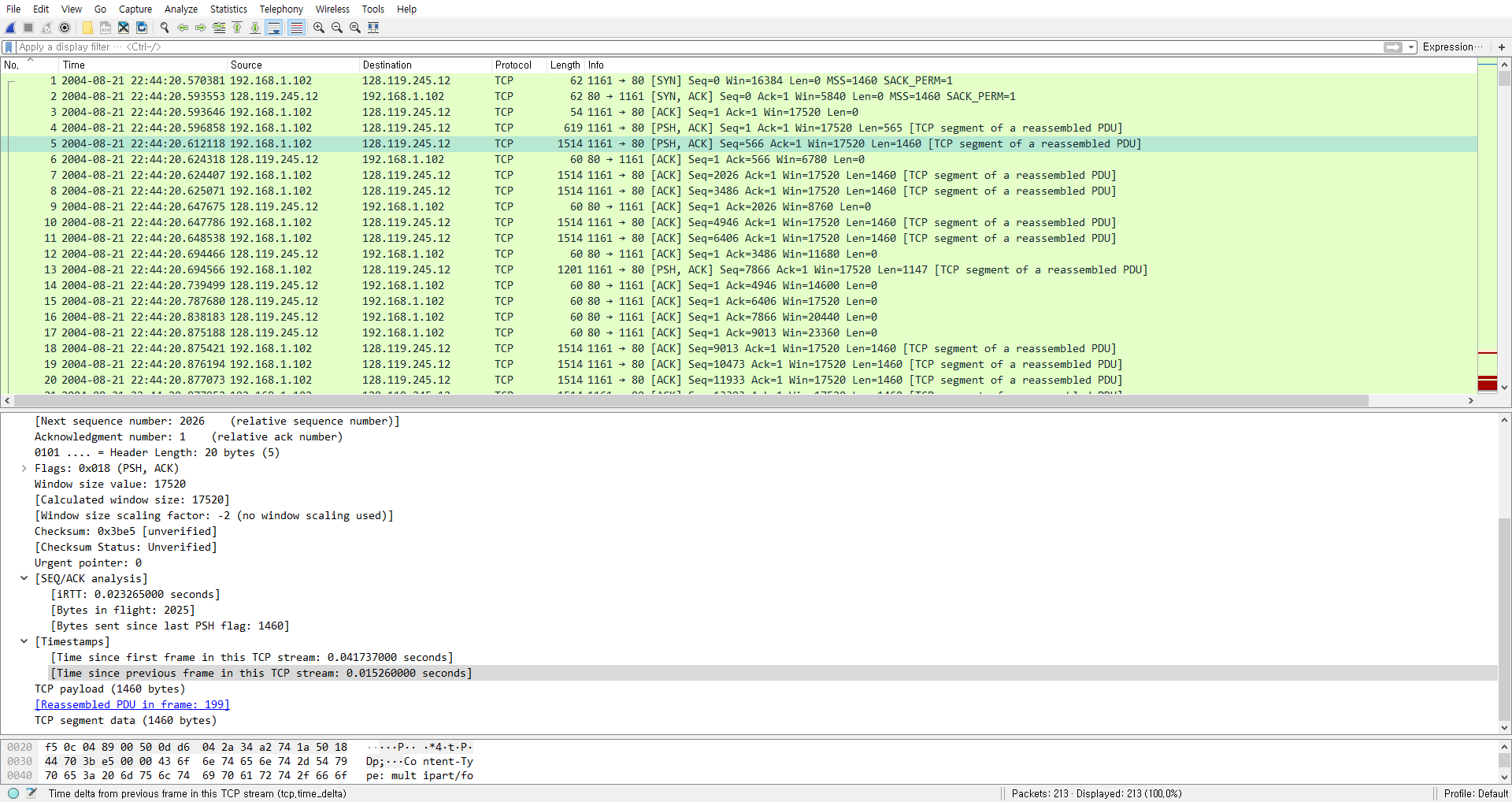
순서대로 1, 566, 2026, 3486, 4946, 6406이다.

아래의 문제에서는 각 세그먼트를 1번, 2번, 3번, 4번, 5번, 6번이라고 칭한다.

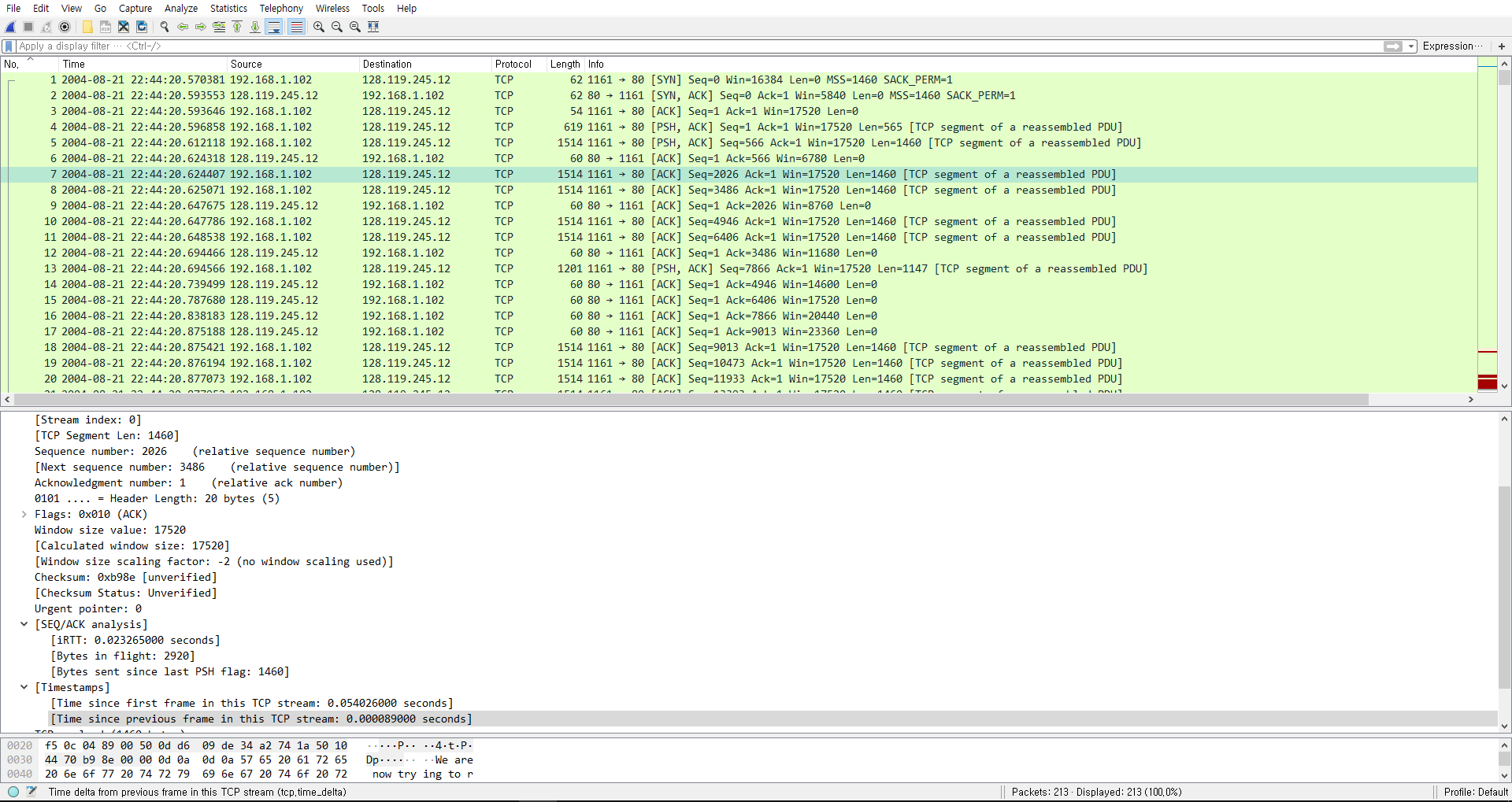
2)각 세그먼트는 언제 보내졌나요?



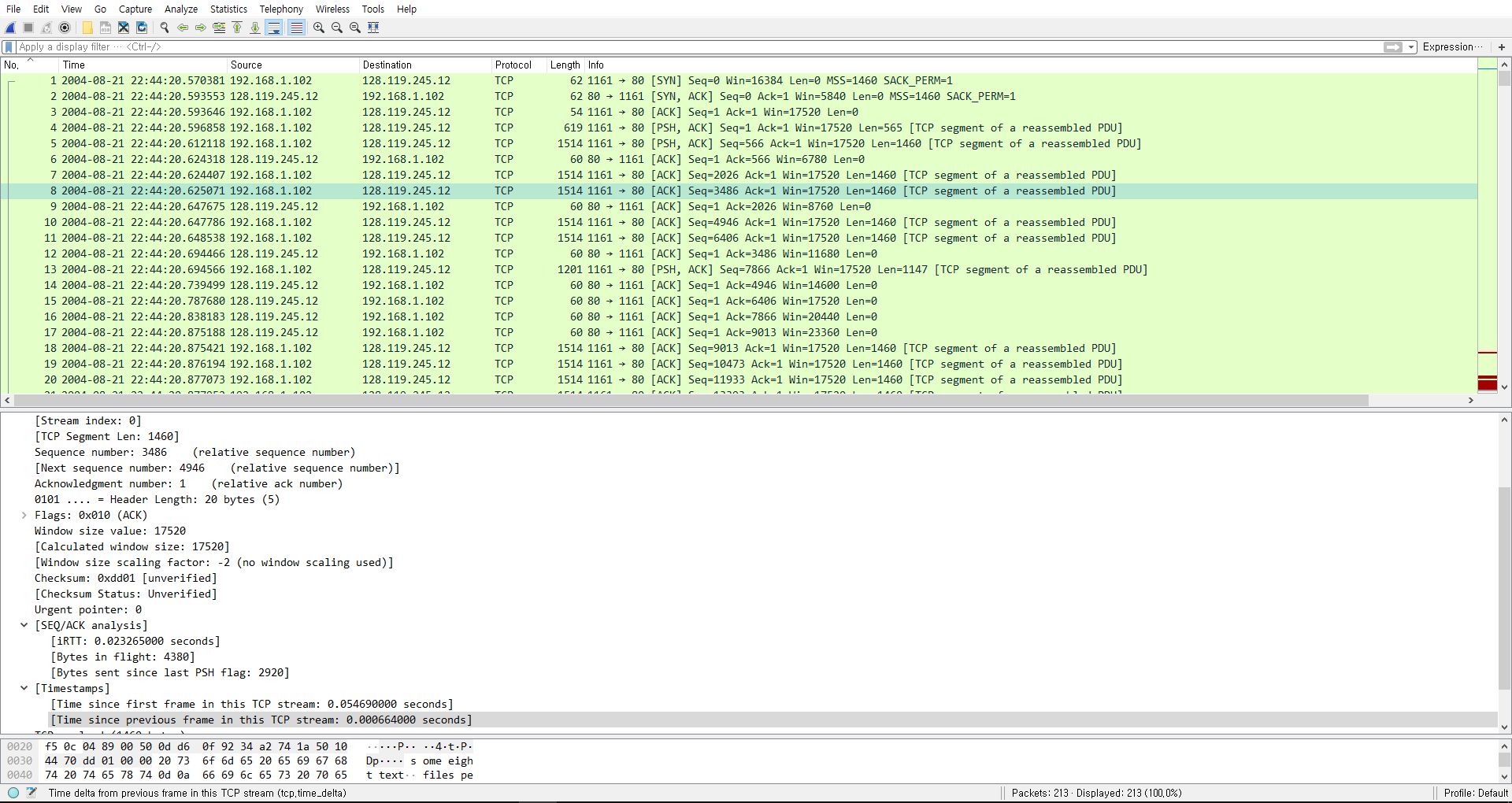
1번의 Timestamps이다. 첫 연결 후 0.026477000초에 전송되었다.



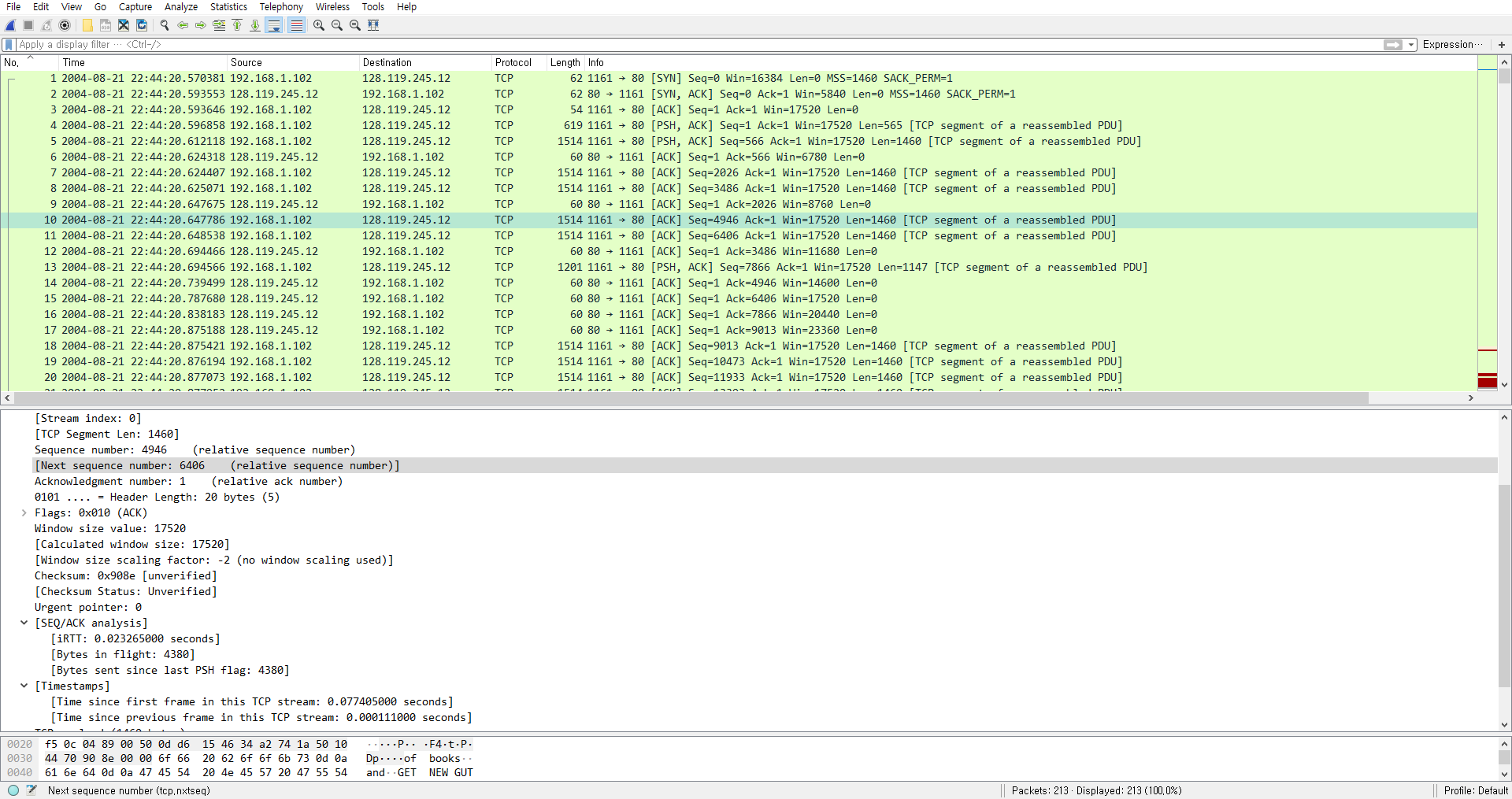
2번의 Timestamps이다. 첫 연결 후 0.041737000초에 전송되었다.



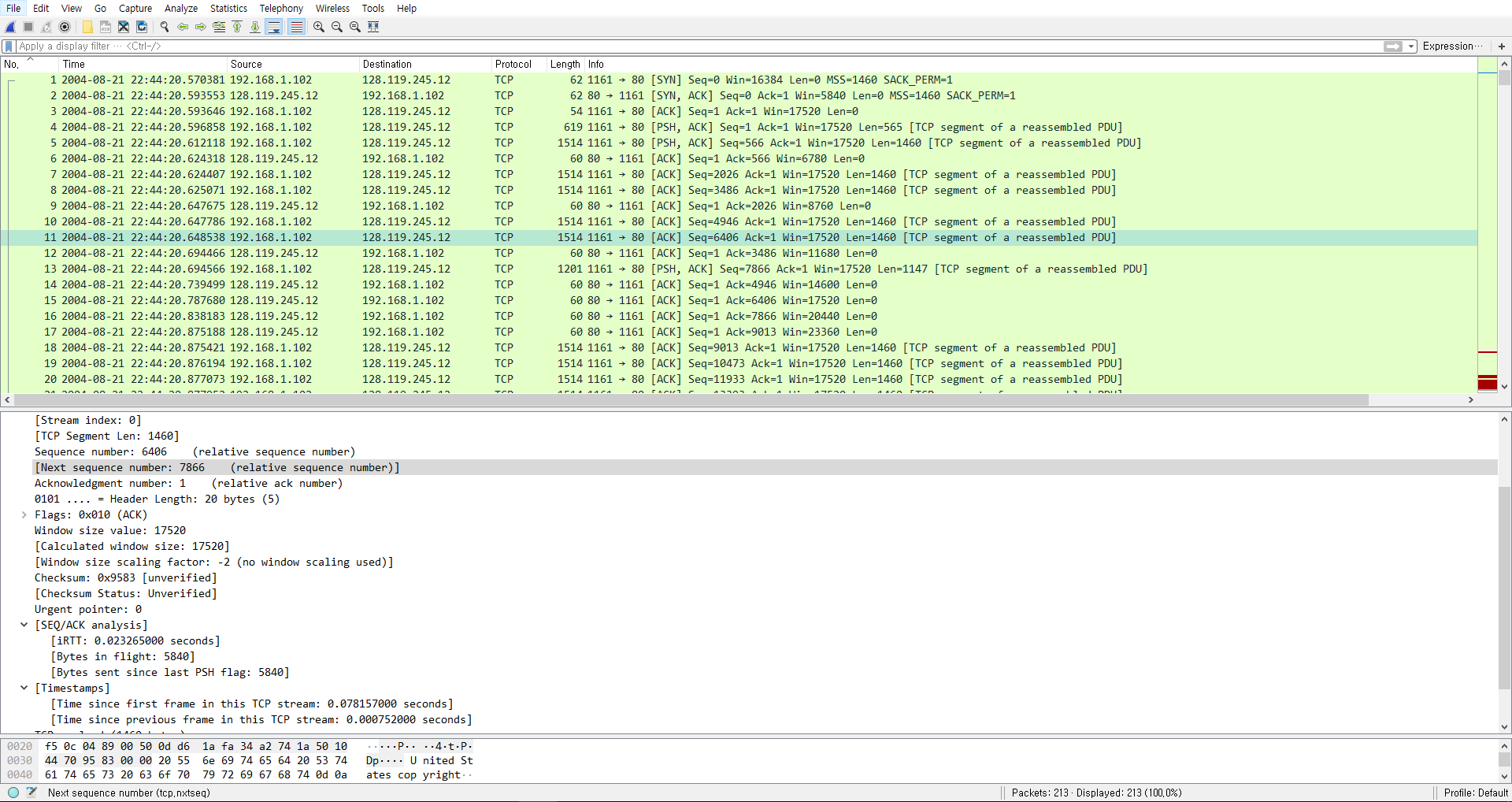
3번의 Timestamps이다. 첫 연결 후 0.054026000초에 전송되었다.



4번의 Timestamps이다. 첫 연결 후 0.054690000초에 전송되었다

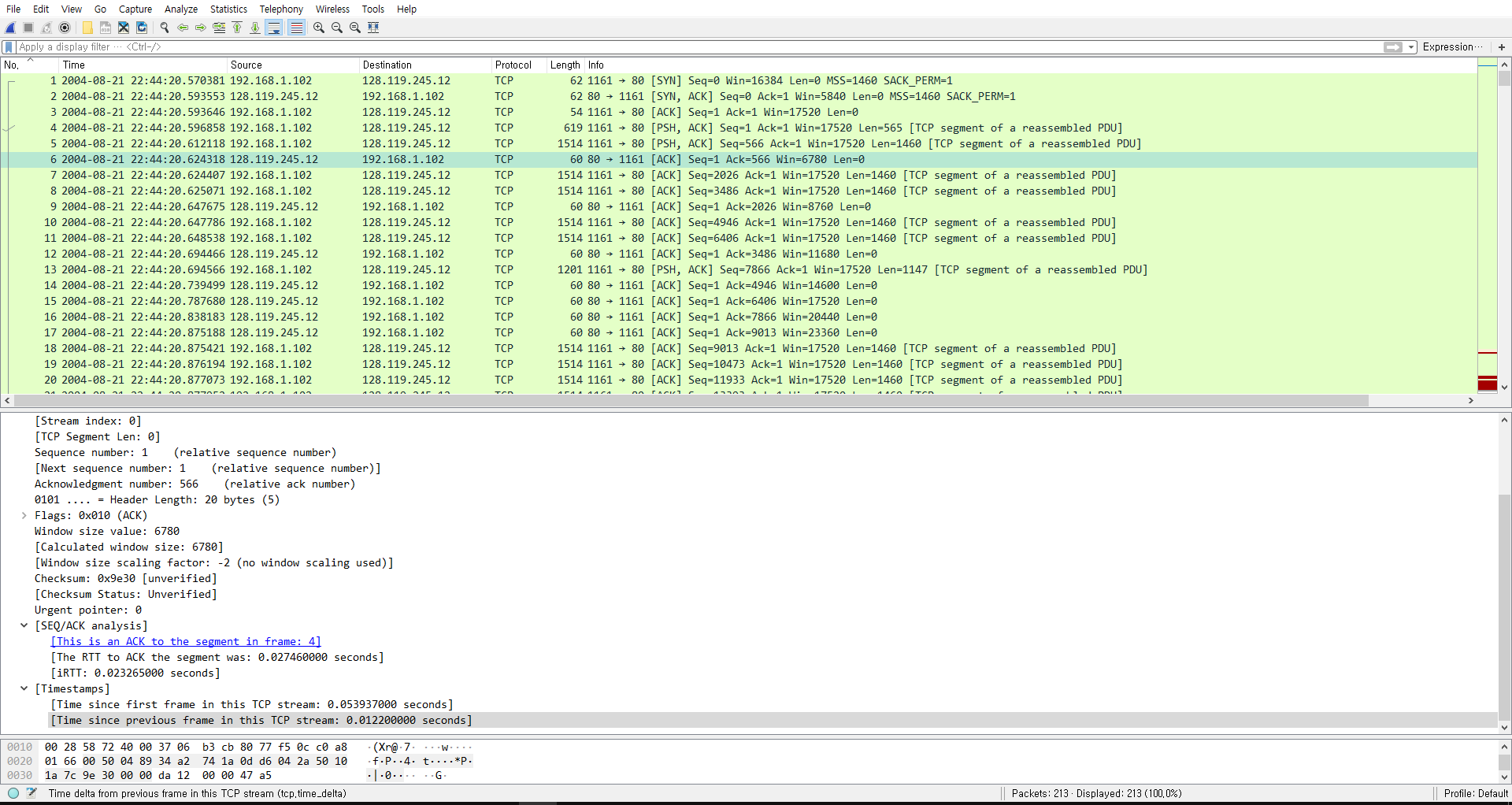
.

5번의 Timestamps이다. 첫 연결 후 0.077405000초에 전송되었다.

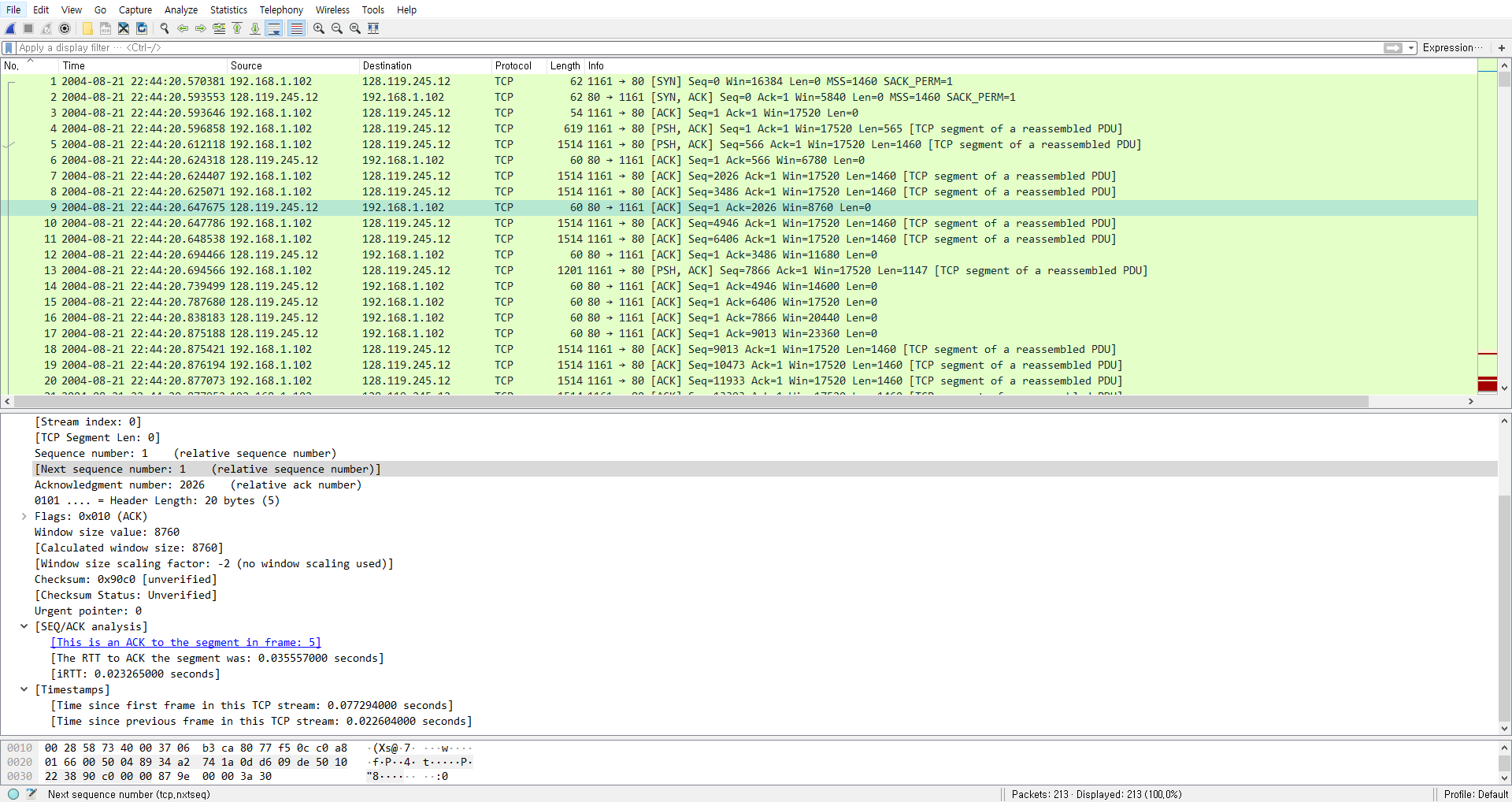


6번의 Timestamps이다. 첫 연결 후 0.078157000초에 전송되었다.

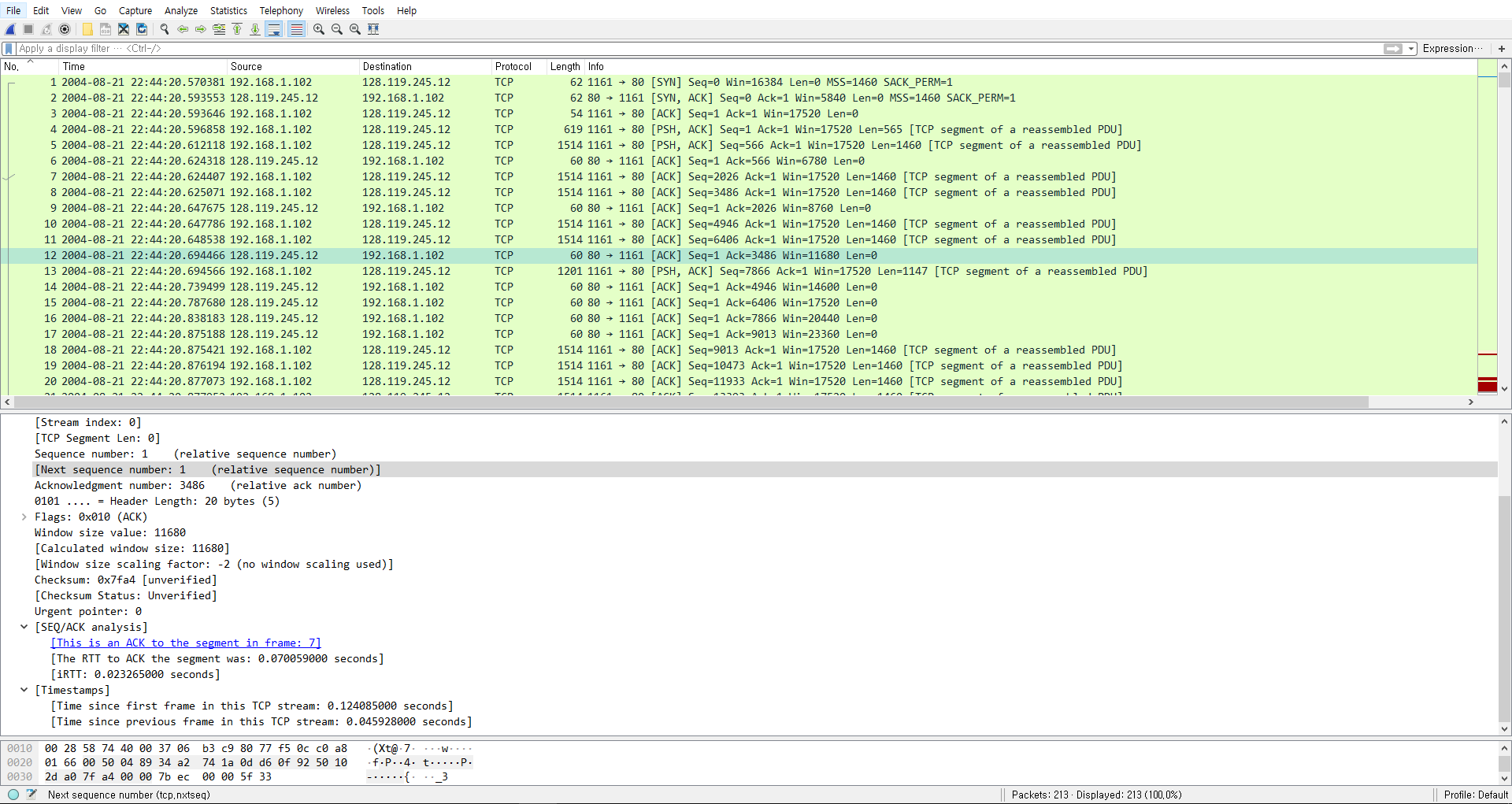
3)각 세그먼트에 대한 ACK는 언제 수신되었나요?



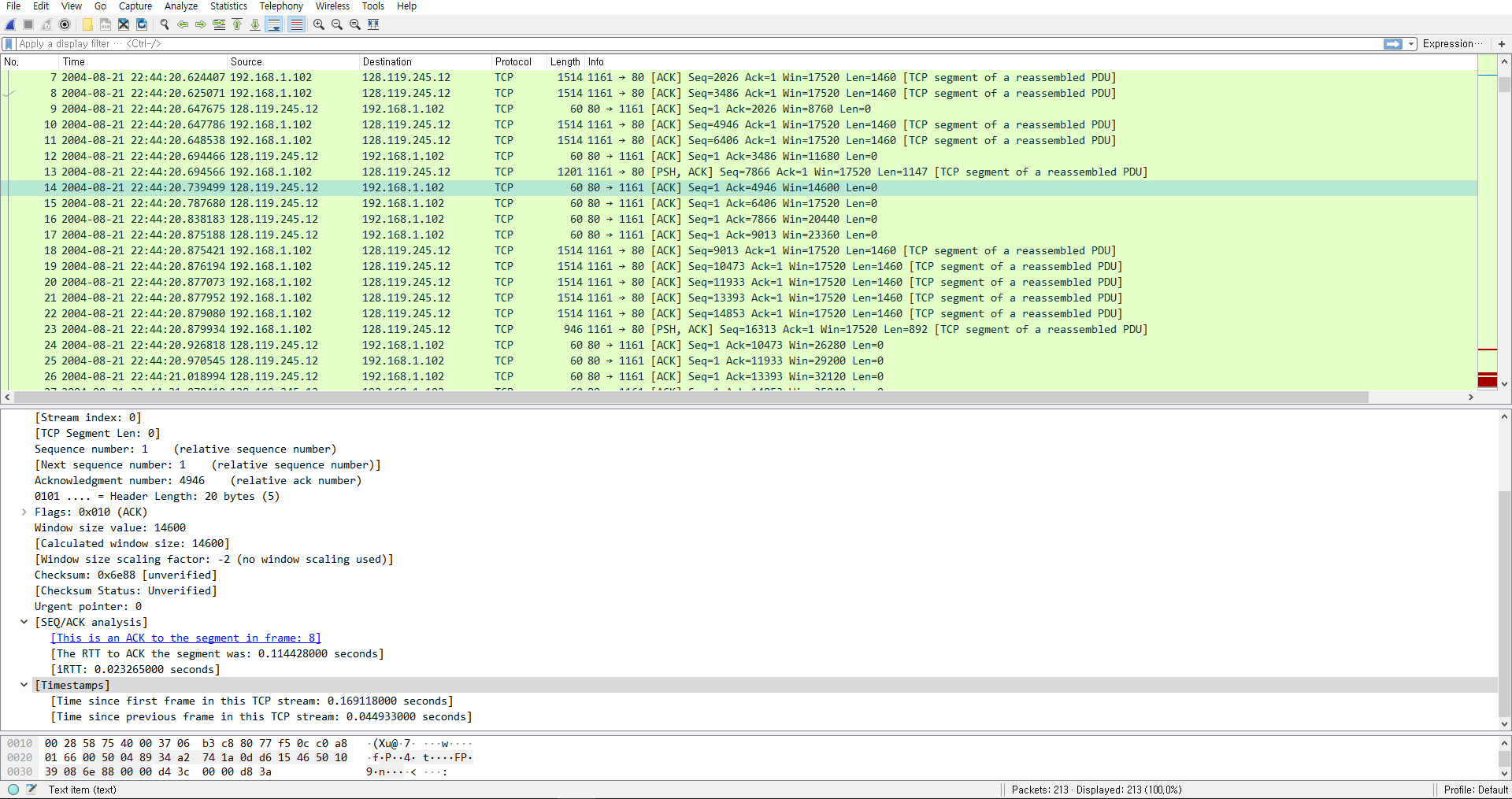
1번에 대한 ACK이다. 첫 연결 후 0.053937000초에 전송되었다.



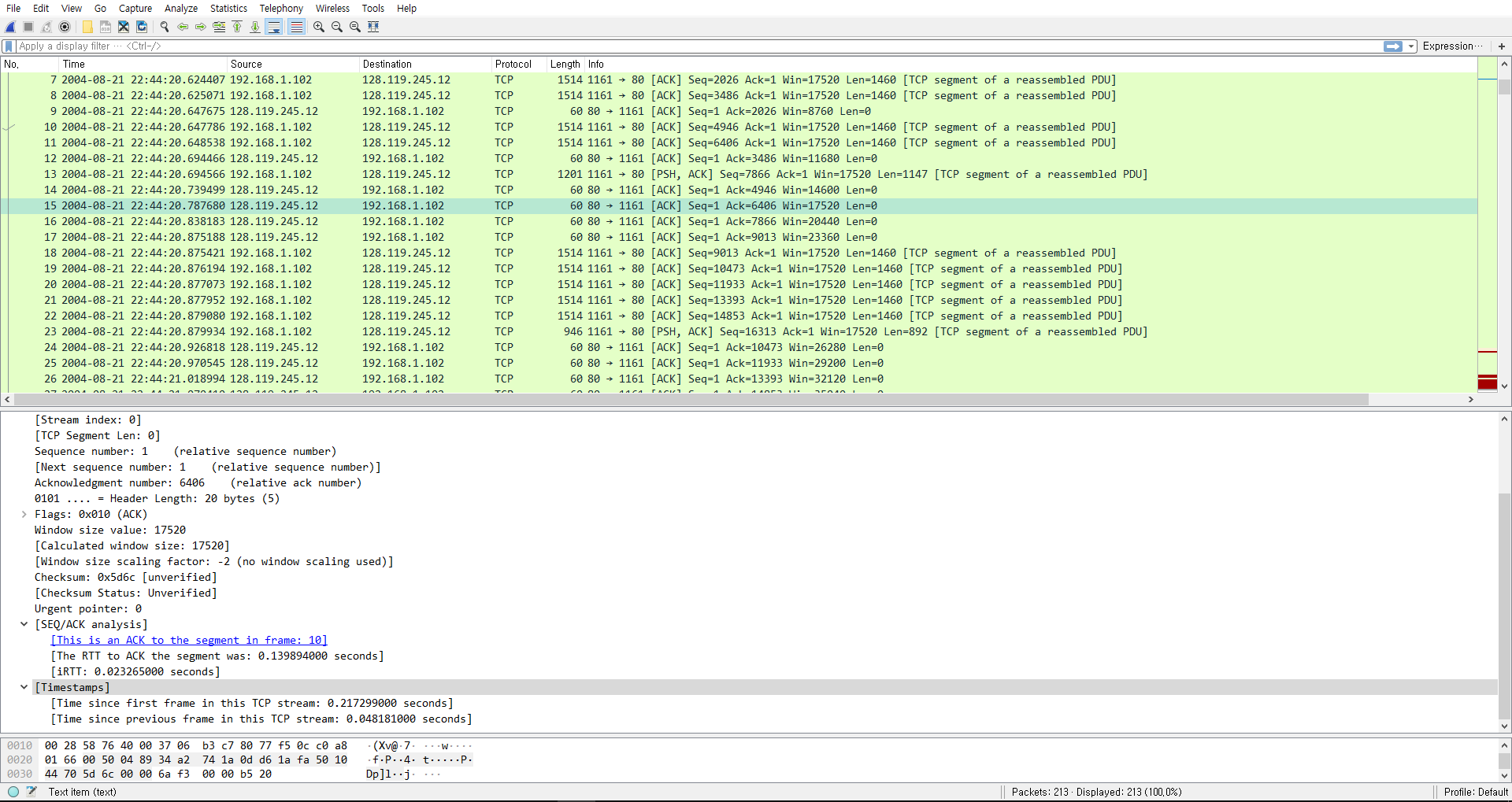
2번에 대한 ACK이다. 첫 연결 후 0.077294000초에 전송되었다.



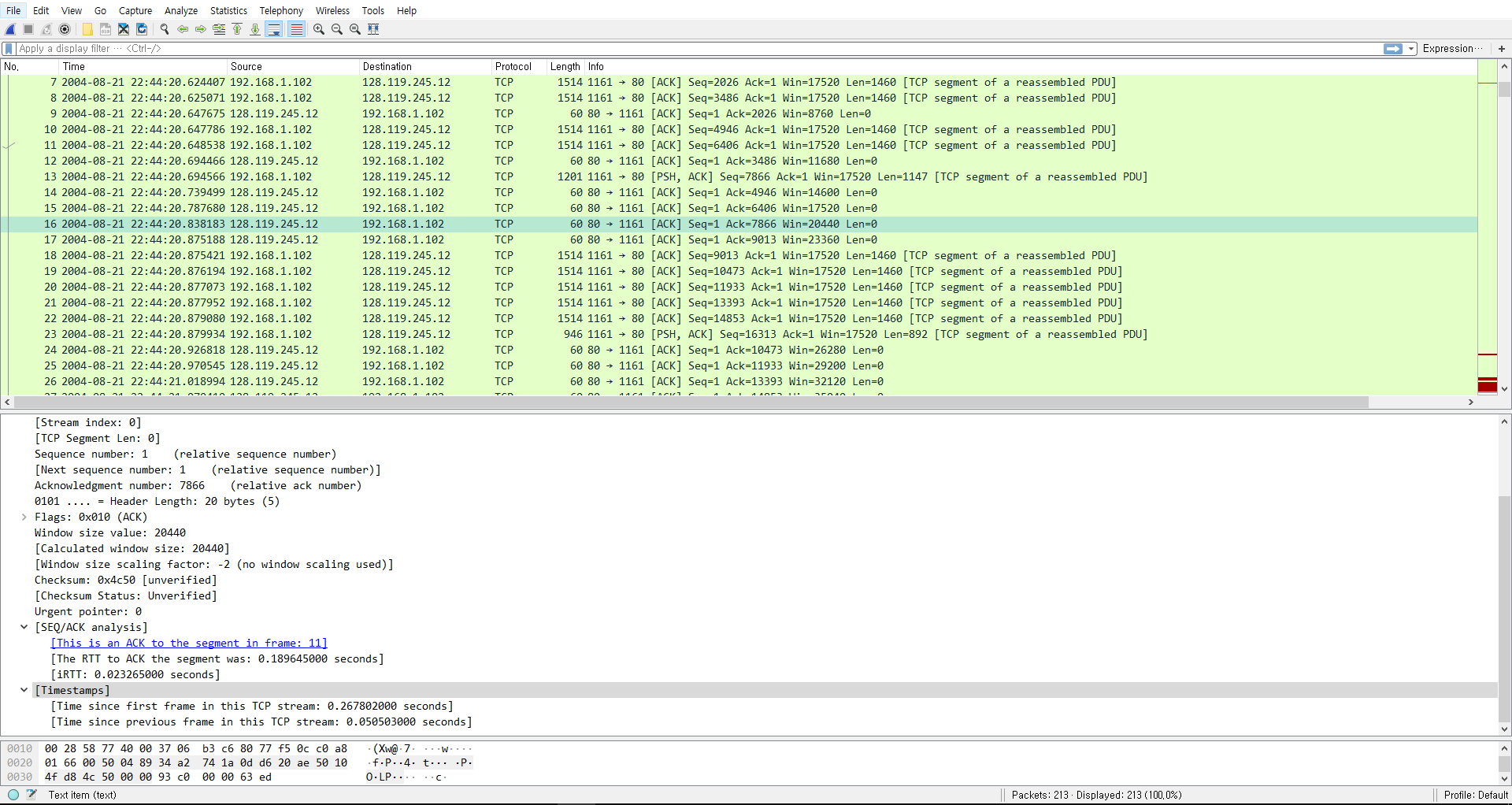
3번에 대한 ACK이다. 첫 연결 후 0.124085000초에 전송되었다.



4번에 대한 ACK이다. 첫 연결 후 0.169118000초에 전송되었다.



5번에 대한 ACK이다. 첫 연결 후 0.217299000초에 전송되었다.



6번에 대한 ACK이다. 첫 연결 후 0.267802000초에 전송되었다.

ACK의 번호는 전송받은 세그먼트의 다음 세그먼트의 시퀀스 번호를 가지고 있으므로

ACK번호와 동일한 시퀀스 번호를 가지는 세그먼트의 이전 세그먼트에 대한 응답이다.

4)각 TCP 세그먼트가 전송된 시점과 수신확인이 된 시점의 차이를 고려해 볼 때,

6개의 세그먼트 각각에 대한 RTT값은 얼마인가요?

2)번과 3)번에서 구한 시간을 뺀 값이 RTT이다.

하지만 와이어샤크에서는 친절하게도 RTT값을 알려주었다. 3)번의 그림 참조

따라서

1번의 RTT : 0.027460000 초

2번의 RTT : 0.035557000 초

3번의 RTT : 0.070059000 초

4번의 RTT : 0.114428000 초

5번의 RTT : 0.139894000 초

6번의 RTT : 0.189645000 초

5)각 ACK를 받은 후 EstimatedRTT값은 얼마인가요? 이때, EstimatedRTT값이 첫번째 세그먼트에

대해 측정된 RTT와 같다고 가정하고 모든 후속 세그먼트들은 EstimatedRTT 계산 식을 통해

산출된다. (힌트: wireshark의 listring of captured packets에서 TCP세그먼트를 선택하고

Statistics-> TCP Stream Graph-> Round Trip Time Graph를 통해 왕복시간그래프 확인이가능 )

\*EstimatedRTT = (1-α)\*EstimatedRTT + α\*SampleRTT, α=0.125

따라서

1번 : **0.02746**

**2번 :** (0.02746 \* 0.875) + (0.035557 + 0.125) = **0.028472125**

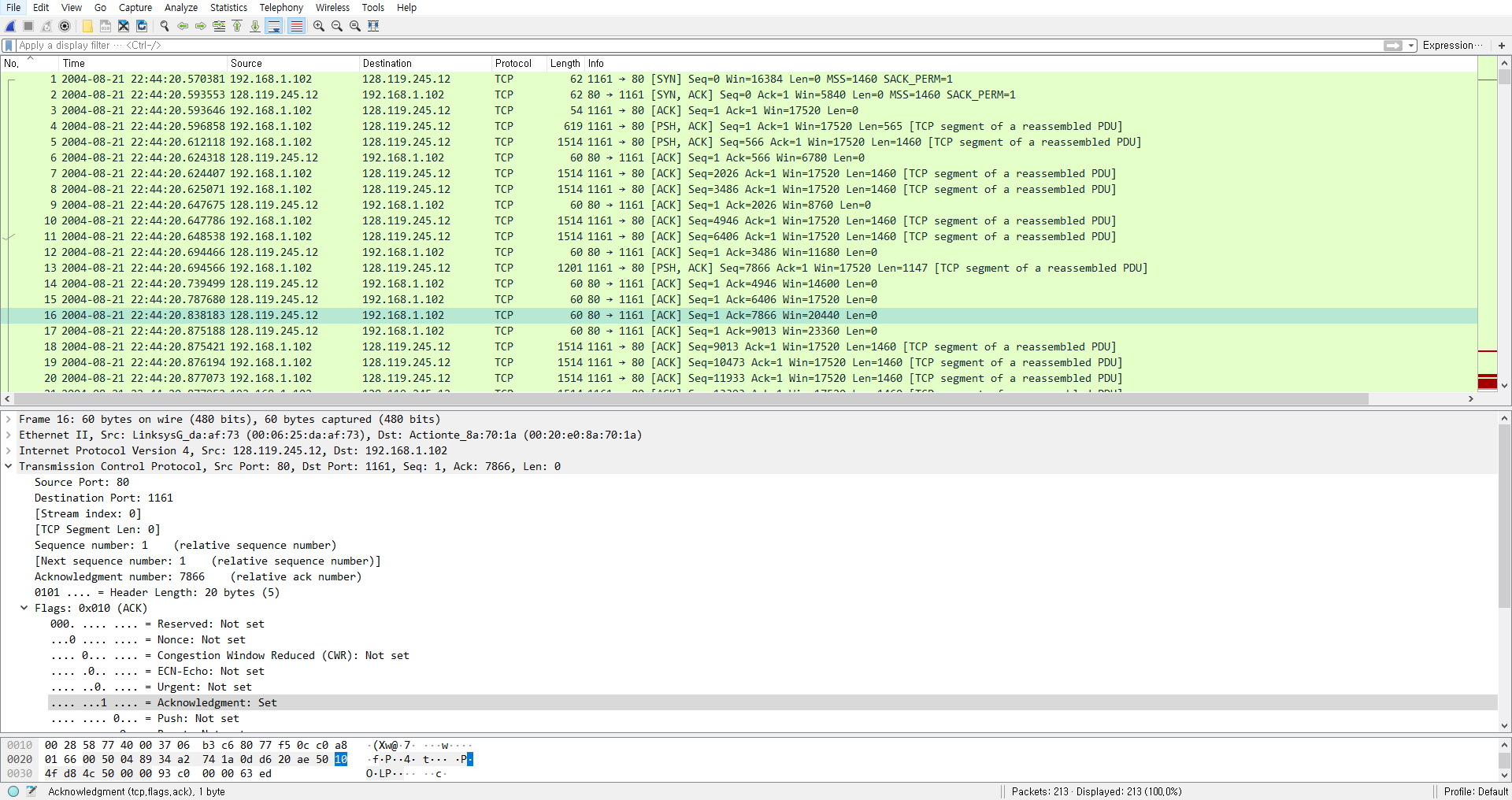
**3번 :** (0.028472125 \* 0.875) + (0.070059 \* 0.125) = **0.033670484**

4번 : (0.033670484 \* 0.875) + (0.114428 \* 0.125) = **0.43765174**

5번 : (0.043765174 \* 0.875) + (0.139894 \* 0.125) = **0.055781277**

6번 : (0.055781277 \* 0.875) + (0.189645 \* 0.125) = **0.072514242**

8. 처음 6개의 TCP세그먼트 각각의 길이는 얼마인가요?

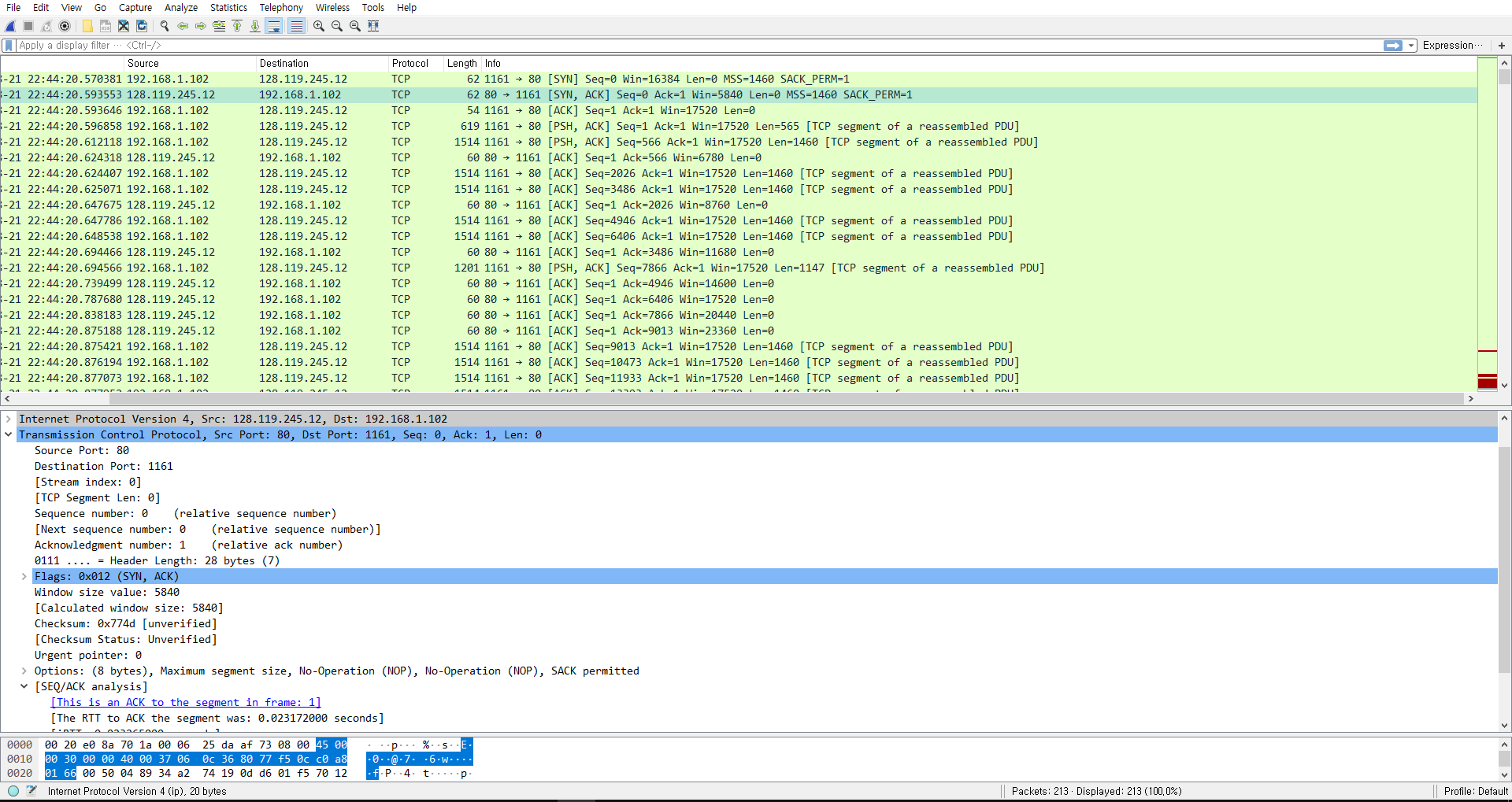


1번 : 565Bytes

2~6번 : 1460Bytes

9. 전체 추적에 있어서 수신시 사용가능한 버퍼 공간의 최소량은 얼마인가요?

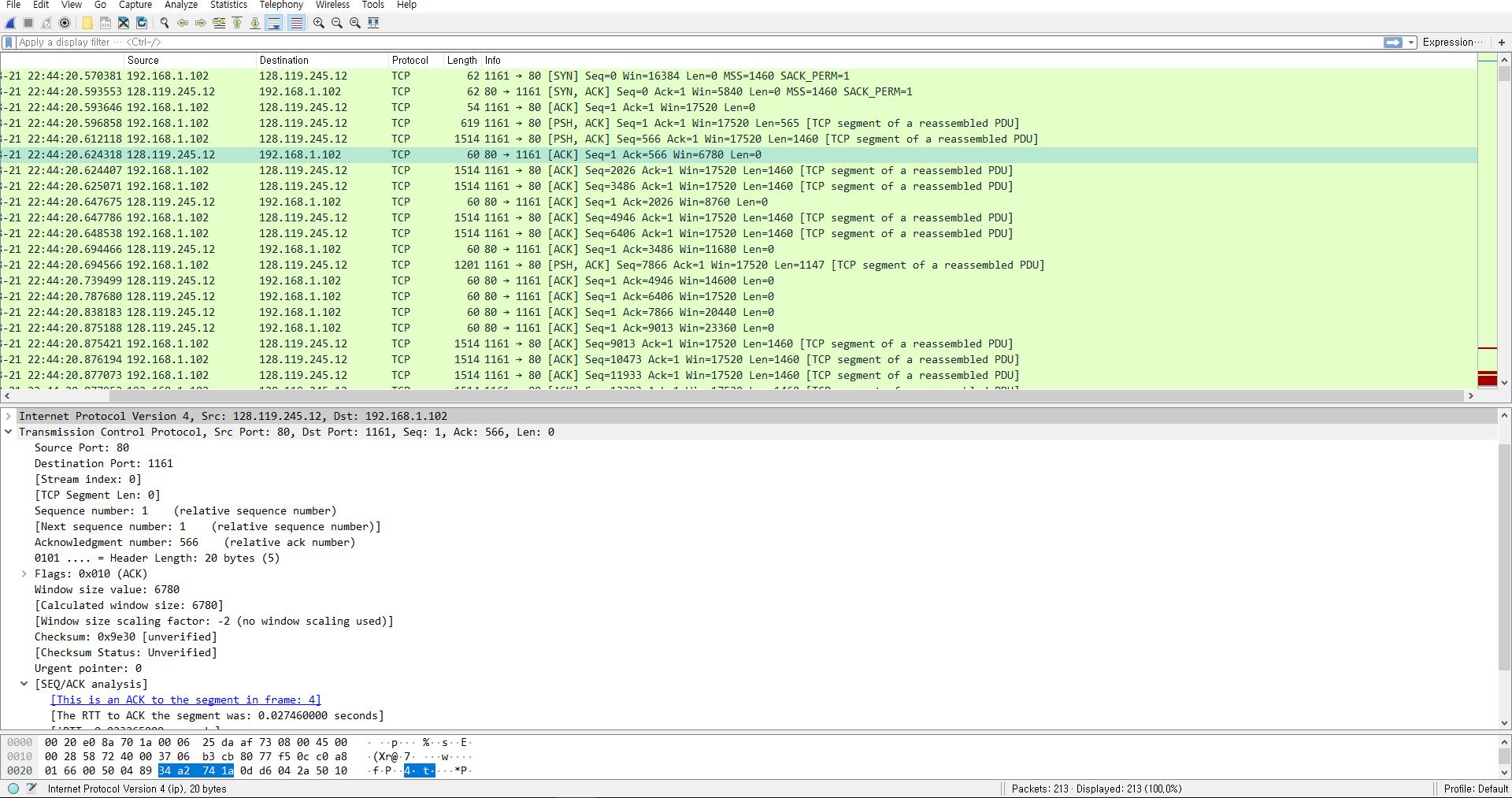
수신기 버퍼 공간이 부족하여 송신자는 감속되나요?



버퍼 공간의 최소량은 첫 세그먼트를 전송할 때 이므로 시퀀스 넘버가 0인 세그먼트의

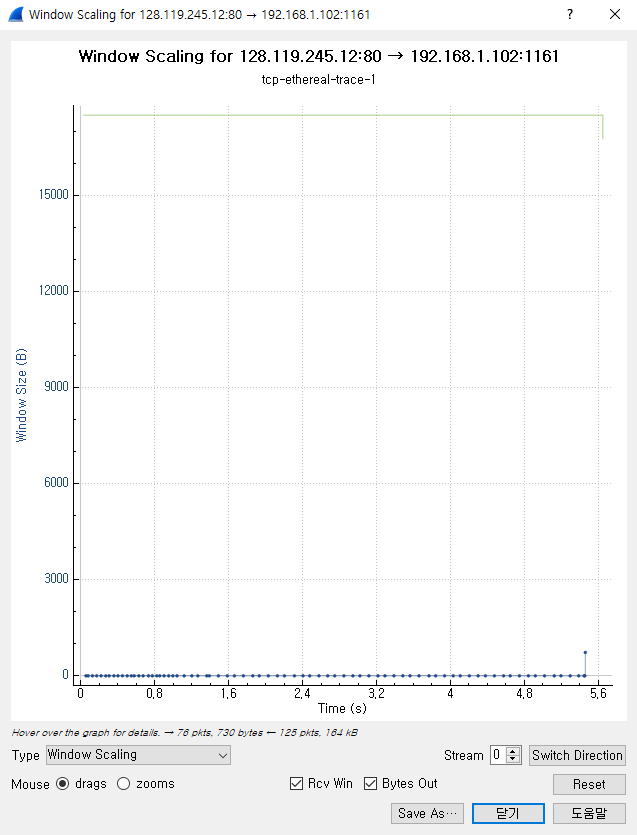
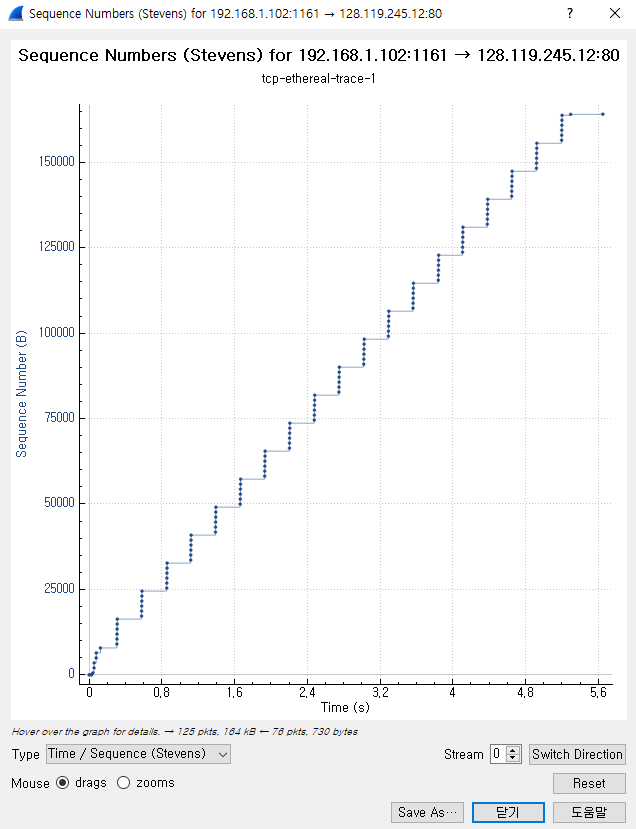
Window Size Value : 5840Bytes이다.

또한



다음 세그먼트에서 버퍼의 크기가 증가하므로 버퍼 공간이 부족하여 감소하지 않는다.

10. 추적파일에서 재전송된 세그먼트가 있나요? 이 질문에 답하기 위해 무엇을 확인해야하나요?



재전송 된 세그먼트는 없다.

재전송이 발생하면 세그먼트 시퀀스 번호가 동일한 패킷이 발송되며

Window Size가 감소한다.

하지만 왼쪽의 그림에서 시퀀스 번호가 동일한 패킷은 발생하지 않으며

오른쪽 그림에서 Window Size는 일정하다.

따라서 재전송 된 세그먼트가 없다고 유추할 수 있다.

11. 수신자는 ACK에서 보통 얼마나 많은 데이터를 인식하나요? 수신자는 수신된

모든 세그먼트에 대해 ACK를 하는 경우에는 식별이 가능한가요?

앞서 보인 TCP 세그먼트의 데이터 크기는 565에서 증가하다가

1460Bytes의 데이터 크기를 가진다. 따라서, 보통 1460Bytes의 데이터를 인식한다고 볼 수 있다.

또한 수신자는 시퀀스 넘버와 ACK 넘버를 통해 식별이 가능하다.

12. TCP 연결에 있어서 처리량 (시간당 전송된 바이트수)은 얼마인가요?

이 값을 어떻게 계산했는지 설명하세요.

마지막 ACK 번호가 164091이므로 전송된 총 데이터는 164090Byte이며

첫 데이터 전송 마지막 ACK 도착 시간(5.455830000초)까지 계산하면 총 처리량이 나온다.

wireshark에서 Timestamp는 TCP 연결(3-way handshaking)이 완료 된 이후 시간을 알 수 있다.

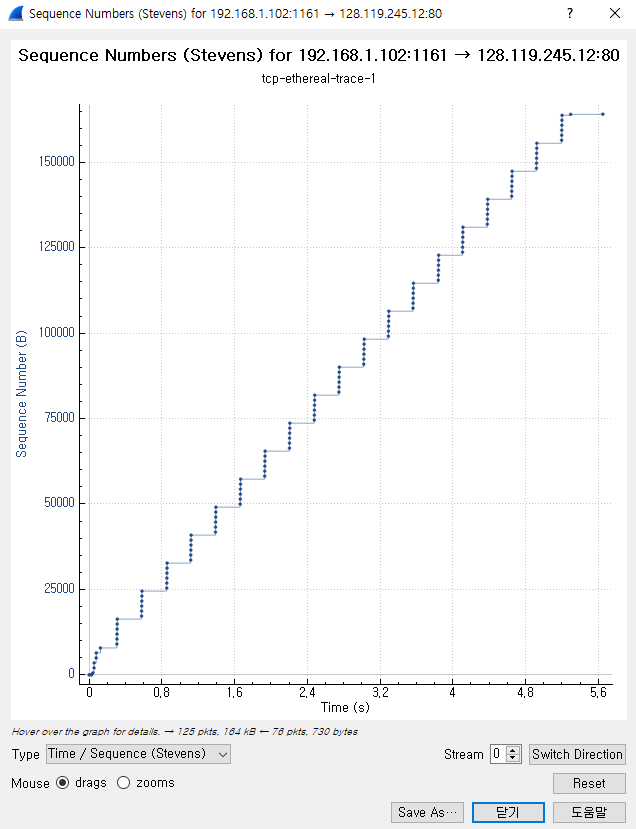
따라서, 164090/5.455830=30,076.083748943790404026518421578‬…

총 처리량은 약 30076.08 이다.

13. 플롯을 통해 TCP의 슬로스타트(slowstart)단계가 시작되고 끝나는 점과

혼잡회피(congestion avoidance)가 발생한 곳을 식별할 수 있나요? 측정한 데이터가

우리가 배운 TCP의 이상적인 동작과 다른 점을 말하세요.

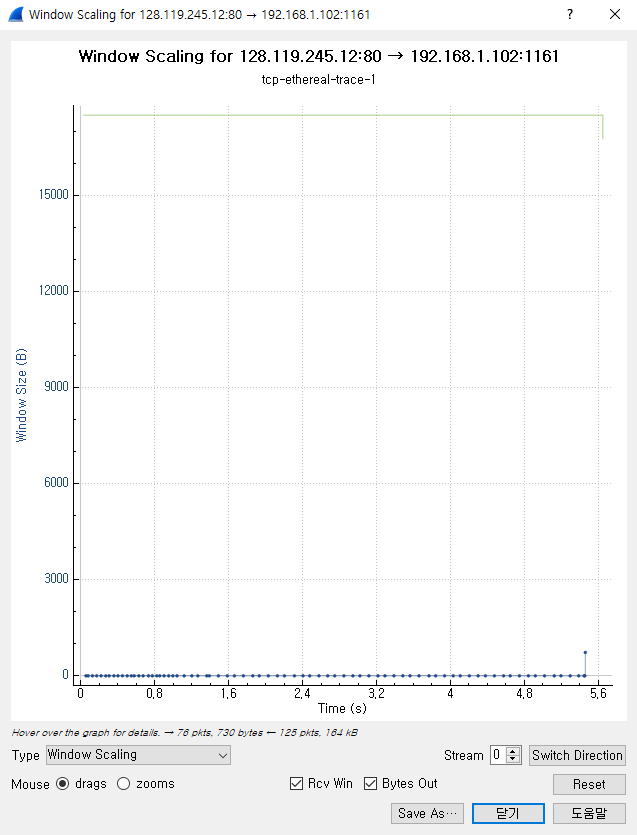


TCP의 슬로스타트 단계가 시작되는 지점은 패킷의 전송이 시작되는 부분으로,

HTTP POST 명령 이후에 시작된다. 따라서 4번째 세그먼트 전송 이후 슬로스타트가 시작된다.

시간당 전송 세그먼트의 수가 일정하므로 슬로스타트가 끝나는 점과

혼잡회피가 발생하는 곳을 알기 어렵다.



윈도우 사이즈를 확인해보아도 크기가 일정하기 때문에,

슬로스타트가 끝나는 점과 혼잡회피가 발생한 곳을 식별할 수 없다.

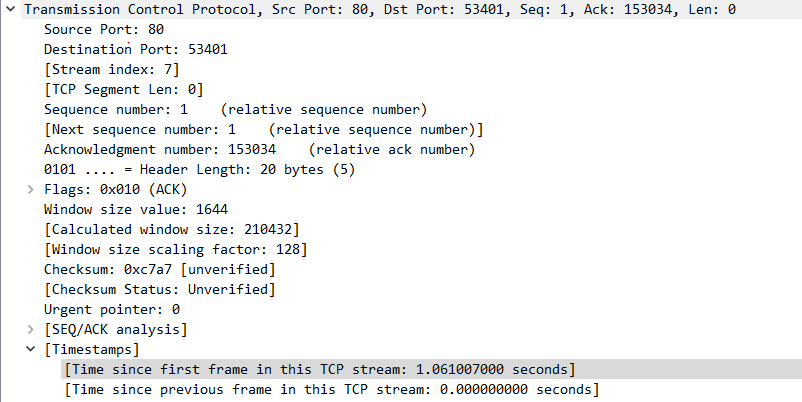
우리가 배운 TCP의 이상적인 동작과 다른 점은,

어플리케이션 계층에서 전송할 데이터 크기를 한정해 주어, 재전송이 발생하지 않다면

임계값과는 별개로 일정한 수의 세그먼트로 일정 크기의 데이터를 전송한다는 것이다.

14. 앞서 실습했던 여러분의 클라이언트 컴퓨터에서 gaia.cs.umass.edu로 파일을 전송할 때

(upload alice.txt) 수집한 추적(captured trace) 을 보고 12번의 문제를 다시 풀어보세요.



따라서 처리량은 153034/1.061007000=144,234.67517179434254439414631572…

약 144234.68 이다.