Wasfi Momen

CSC 6220: Assignment 3

1. The 8bit checksum is 00100110. At the receiver's side, all the 8bit chunks of the sender's message and the checksum value will be added together. The resulting sum should be 11111111. If the receiver computes the sum and gets a 0 for any value, it means there was an error in the transmission, an in-memory error, or even a hardware error.
2.

Rdt 1.0 – Assumes network layer is reliable and there are no errors. Creates two state machines for the sender and receiver.
Rdt 2.0 – Adds error detection and feedback by including ACKs and NAKs. Now, sender waits for ACK or NAK from receiver and the receiver checks the packet for corruption. Cannot handle garbled or duplicate ACKs or NAKs. Still assumes no packet loss.
Rdt 2.1 – Adds a 1bit sequence number. Can now detect garbled NAKs or ACKs and the sequence number indicates whether the packet being sent is a new packet or an old packet retransmission, solving the duplicate problem. Still assumes no packet loss.
Rdt 2.2 – Same as 2.1, but instead of NAKs we ACK the last noncorrupted packet sequence number. Sender now must check sequence number to decide whether to retransmit. Duplicates can now issue retransmission. Still assumes no packet loss.
Rdt 3.0 – Now assumes packet loss. Adds an additional function to each state as a timer. If the timer runs out, the sender will retransmit. ACKs, sequence number can also issue retransmission.

The problem of Go-Back-N was of performance considerations. If one packet is dropped en route, then all packets before must be retransmitted. This can be a significant problem in large amounts of packets where the probability of packet loss is high. If packet 999 out of 1,000 packets is lost, then packets 1-999 will have to be retransmitted (assuming the window size is 1,000).

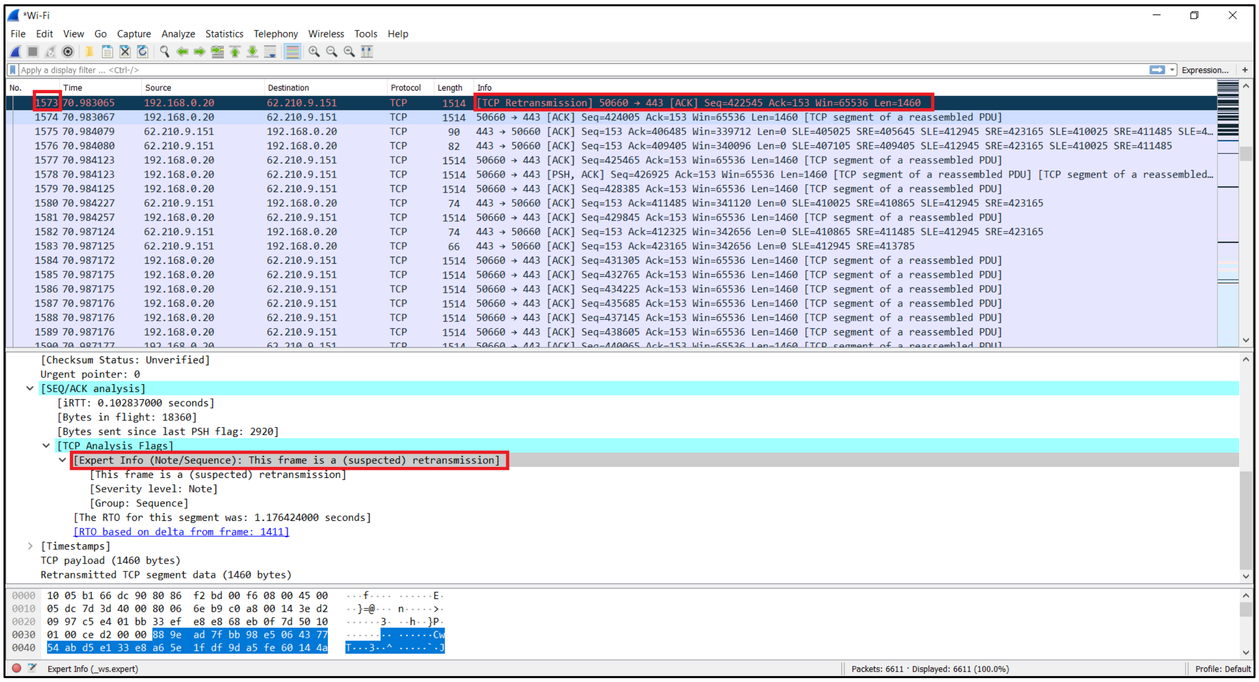
3.

For opening a connection, the sequence numbers will be 0 until the last ACK in the 3-way handshake is received, then it will be 1. The FIN bit will be set to 0 always. The SYN bit will be 1 until the last ACK, then the SYN bit will be 0.

For closing a connection, the sequence number of both the sender and receiver will be incremented by how many bytes they have transferred. Once the decision is made to terminate the connection, one of the sides will set the FIN bit to 1 with the SYN bit at 0. The other side will ACK with a FIN bit of 1 and SYN bit of 0. The final ACK will have both SYN and FIN set to 0.

4.

An example of duplicate ACK when uploading and downloading a file on https://pdftoimage.com/. The duplicate ACK has the same sequence, acknowledgement, and source, destination ports.



```
No.   Time      Source         Destination    Protocol Length Info
 1573 70.983065 192.168.0.20   62.210.9.151   TCP      1514   [TCP Retransmission] 50660 → 443 [ACK] Seq=422545 Ack=153 Win=65536 Len=1460
 1574 70.983067 192.168.0.20   62.210.9.151   TCP      1514   50660 → 443 [ACK] Seq=424005 Ack=153 Win=65536 Len=1460 [TCP segment of a reassembled PDU]
 1575 70.984079 62.210.9.151   192.168.0.20   TCP        90   443 → 50660 [ACK] Seq=153 Ack=406485 Win=339712 Len=0 SLE=405025 SRE=405645 SLE=412945 SRE=423165 SLE=410025 SRE=411485 SLE=4…
 1576 70.984080 62.210.9.151   192.168.0.20   TCP        82   443 → 50660 [ACK] Seq=153 Ack=409405 Win=340096 Len=0 SLE=407105 SRE=409405 SLE=412945 SRE=423165 SLE=410025 SRE=411485
 1577 70.984123 192.168.0.20   62.210.9.151   TCP      1514   50660 → 443 [ACK] Seq=425465 Ack=153 Win=65536 Len=1460 [TCP segment of a reassembled PDU]
 1578 70.984123 192.168.0.20   62.210.9.151   TCP      1514   50660 → 443 [PSH, ACK] Seq=426925 Ack=153 Win=65536 Len=1460 [TCP segment of a reassembled PDU] [TCP segment of a reassembled…
 1579 70.984125 192.168.0.20   62.210.9.151   TCP      1514   50660 → 443 [ACK] Seq=428385 Ack=153 Win=65536 Len=1460 [TCP segment of a reassembled PDU]
 1580 70.984227 62.210.9.151   192.168.0.20   TCP        74   443 → 50660 [ACK] Seq=153 Ack=411485 Win=341120 Len=0 SLE=410025 SRE=410865 SLE=412945 SRE=423165
 1581 70.984257 192.168.0.20   62.210.9.151   TCP      1514   50660 → 443 [ACK] Seq=429845 Ack=153 Win=65536 Len=1460 [TCP segment of a reassembled PDU]
 1582 70.987124 62.210.9.151   192.168.0.20   TCP        74   443 → 50660 [ACK] Seq=153 Ack=412325 Win=342656 Len=0 SLE=410865 SRE=411485 SLE=412945 SRE=423165
 1583 70.987125 62.210.9.151   192.168.0.20   TCP        66   443 → 50660 [ACK] Seq=153 Ack=423165 Win=342656 Len=0 SLE=412945 SRE=413785
 1584 70.987172 192.168.0.20   62.210.9.151   TCP      1514   50660 → 443 [ACK] Seq=431305 Ack=153 Win=65536 Len=1460 [TCP segment of a reassembled PDU]
 1585 70.987175 192.168.0.20   62.210.9.151   TCP      1514   50660 → 443 [ACK] Seq=432765 Ack=153 Win=65536 Len=1460 [TCP segment of a reassembled PDU]
 1586 70.987175 192.168.0.20   62.210.9.151   TCP      1514   50660 → 443 [ACK] Seq=434225 Ack=153 Win=65536 Len=1460 [TCP segment of a reassembled PDU]
 1587 70.987176 192.168.0.20   62.210.9.151   TCP      1514   50660 → 443 [ACK] Seq=435685 Ack=153 Win=65536 Len=1460 [TCP segment of a reassembled PDU]
 1588 70.987176 192.168.0.20   62.210.9.151   TCP      1514   50660 → 443 [ACK] Seq=437145 Ack=153 Win=65536 Len=1460 [TCP segment of a reassembled PDU]
 1589 70.987176 192.168.0.20   62.210.9.151   TCP      1514   50660 → 443 [ACK] Seq=438605 Ack=153 Win=65536 Len=1460 [TCP segment of a reassembled PDU]
 1590 70.987177 192.168.0.20   62.210.9.151   TCP      1514   50660 → 443 [ACK] Seq=440065 Ack=153 Win=65536 Len=1460 [TCP segment of a reassembled PDU]
```

```
    [Checksum Status: Unverified]
    Urgent pointer: 0
  ∨ [SEQ/ACK analysis]
        [iRTT: 0.102837000 seconds]
        [Bytes in flight: 18360]
        [Bytes sent since last PSH flag: 2920]
      ∨ [TCP Analysis Flags]
          ∨ [Expert Info (Note/Sequence): This frame is a (suspected) retransmission]
                [This frame is a (suspected) retransmission]
                [Severity level: Note]
                [Group: Sequence]
            [The RTO for this segment was: 1.176424000 seconds]
            [RTO based on delta from frame: 1411]
  > [Timestamps]
    TCP payload (1460 bytes)
    Retransmitted TCP segment data (1460 bytes)
```

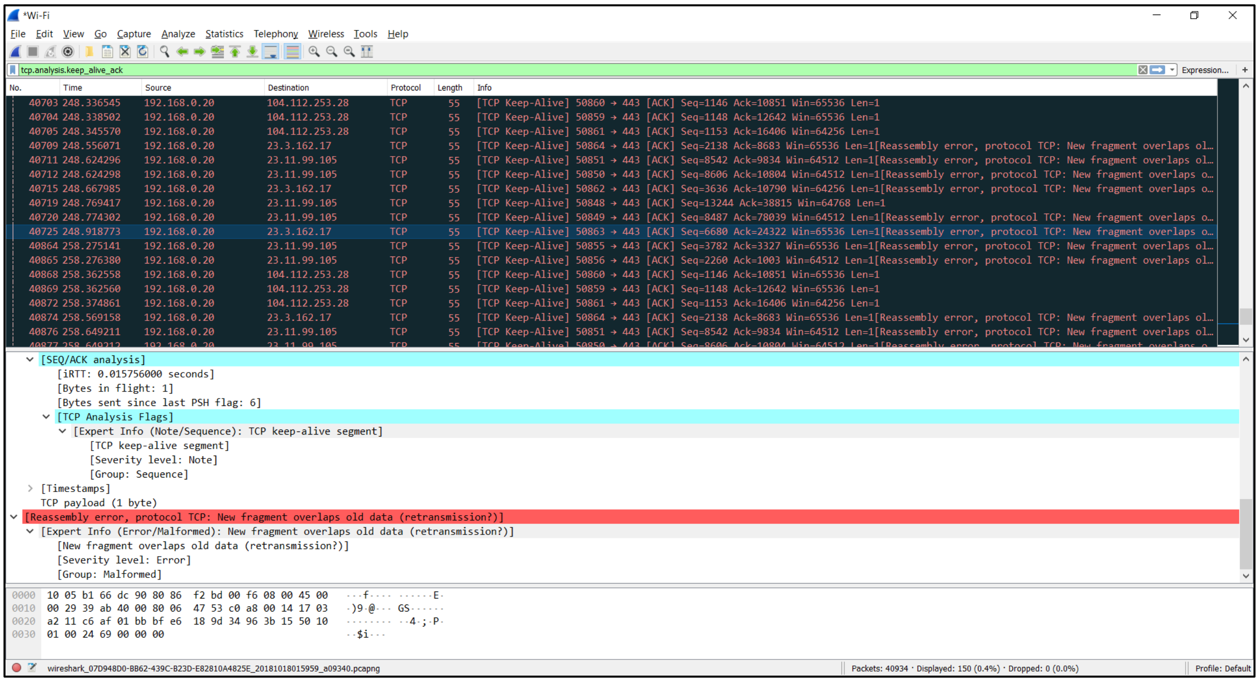
```
0000  10 05 b1 66 dc 90 80 86  f2 bd 00 f6 08 00 45 00   ···f···· ······E·
0010  05 dc 7d 3d 40 00 80 06  6e b9 c0 a8 00 14 3e d2   ··}=@··· n·····>·
0020  09 97 c5 e4 01 bb 33 ef  e8 e8 68 eb 0f 7d 50 10   ······3· ··h··}P·
0030  01 00 ce d2 00 00 88 9e  ad 7f bb 98 e5 06 43 77   ········ ······Cw
0040  54 ab d5 e1 33 e8 a6 5e  1f df 9d a5 fe 60 14 4a   T···3··^ ·····`·J
```

Expert Info (_ws.expert) | Packets: 6611 · Displayed: 6611 (100.0%) | Profile: Default

*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...

5.

Here, packet 40725 is an ACK retransmission due to keep-alive timeout.



tcp.analysis.keep_alive_ack

```
No.    Time       Source         Destination      Protocol Length Info
 40703 248.336545 192.168.0.20   104.112.253.28   TCP        55   [TCP Keep-Alive] 50860 → 443 [ACK] Seq=1146 Ack=10851 Win=65536 Len=1
 40704 248.338502 192.168.0.20   104.112.253.28   TCP        55   [TCP Keep-Alive] 50859 → 443 [ACK] Seq=1148 Ack=12642 Win=65536 Len=1
 40705 248.345570 192.168.0.20   104.112.253.28   TCP        55   [TCP Keep-Alive] 50861 → 443 [ACK] Seq=1153 Ack=16406 Win=64256 Len=1
 40709 248.556071 192.168.0.20   23.3.162.17      TCP        55   [TCP Keep-Alive] 50864 → 443 [ACK] Seq=2138 Ack=8683 Win=65536 Len=1[Reassembly error, protocol TCP: New fragment overlaps ol…
 40711 248.624296 192.168.0.20   23.11.99.105     TCP        55   [TCP Keep-Alive] 50851 → 443 [ACK] Seq=8542 Ack=9834 Win=64512 Len=1[Reassembly error, protocol TCP: New fragment overlaps ol…
 40712 248.624298 192.168.0.20   23.11.99.105     TCP        55   [TCP Keep-Alive] 50850 → 443 [ACK] Seq=8606 Ack=10804 Win=64512 Len=1[Reassembly error, protocol TCP: New fragment overlaps o…
 40715 248.667985 192.168.0.20   23.3.162.17      TCP        55   [TCP Keep-Alive] 50862 → 443 [ACK] Seq=3636 Ack=10790 Win=64256 Len=1[Reassembly error, protocol TCP: New fragment overlaps o…
 40719 248.769417 192.168.0.20   23.11.99.105     TCP        55   [TCP Keep-Alive] 50848 → 443 [ACK] Seq=13244 Ack=38815 Win=64768 Len=1
 40720 248.774302 192.168.0.20   23.11.99.105     TCP        55   [TCP Keep-Alive] 50849 → 443 [ACK] Seq=8487 Ack=78039 Win=64512 Len=1[Reassembly error, protocol TCP: New fragment overlaps o…
 40725 248.918773 192.168.0.20   23.3.162.17      TCP        55   [TCP Keep-Alive] 50863 → 443 [ACK] Seq=6680 Ack=24322 Win=65536 Len=1[Reassembly error, protocol TCP: New fragment overlaps o…
 40864 258.275141 192.168.0.20   23.11.99.105     TCP        55   [TCP Keep-Alive] 50855 → 443 [ACK] Seq=3782 Ack=3327 Win=65536 Len=1[Reassembly error, protocol TCP: New fragment overlaps ol…
 40865 258.276380 192.168.0.20   23.11.99.105     TCP        55   [TCP Keep-Alive] 50856 → 443 [ACK] Seq=2260 Ack=1003 Win=64512 Len=1[Reassembly error, protocol TCP: New fragment overlaps ol…
 40868 258.362558 192.168.0.20   104.112.253.28   TCP        55   [TCP Keep-Alive] 50860 → 443 [ACK] Seq=1146 Ack=10851 Win=65536 Len=1
 40869 258.362560 192.168.0.20   104.112.253.28   TCP        55   [TCP Keep-Alive] 50859 → 443 [ACK] Seq=1148 Ack=12642 Win=65536 Len=1
 40872 258.374861 192.168.0.20   104.112.253.28   TCP        55   [TCP Keep-Alive] 50861 → 443 [ACK] Seq=1153 Ack=16406 Win=64256 Len=1
 40874 258.569158 192.168.0.20   23.3.162.17      TCP        55   [TCP Keep-Alive] 50864 → 443 [ACK] Seq=2138 Ack=8683 Win=65536 Len=1[Reassembly error, protocol TCP: New fragment overlaps ol…
 40876 258.649211 192.168.0.20   23.11.99.105     TCP        55   [TCP Keep-Alive] 50851 → 443 [ACK] Seq=8542 Ack=9834 Win=64512 Len=1[Reassembly error, protocol TCP: New fragment overlaps ol…
 40877 258.649212 192.168.0.20   23.11.99.105     TCP        55   [TCP Keep-Alive] 50850 → 443 [ACK] Seq=8606 Ack=10804 Win=64512 Len=1[Reassembly error, protocol TCP: New fragment overlaps o…
```

```
  ∨ [SEQ/ACK analysis]
        [iRTT: 0.015756000 seconds]
        [Bytes in flight: 1]
        [Bytes sent since last PSH flag: 6]
      ∨ [TCP Analysis Flags]
          ∨ [Expert Info (Note/Sequence): TCP keep-alive segment]
                [TCP keep-alive segment]
                [Severity level: Note]
                [Group: Sequence]
  > [Timestamps]
    TCP payload (1 byte)
∨ [Reassembly error, protocol TCP: New fragment overlaps old data (retransmission?)]
  ∨ [Expert Info (Error/Malformed): New fragment overlaps old data (retransmission?)]
        [New fragment overlaps old data (retransmission?)]
        [Severity level: Error]
        [Group: Malformed]
```

```
0000  10 05 b1 66 dc 90 80 86  f2 bd 00 f6 08 00 45 00   ···f···· ······E·
0010  00 29 39 ab 40 00 80 06  47 53 c0 a8 00 14 17 03   ·)9·@··· GS······
0020  a2 11 c6 af 01 bb bf e6  18 9d 34 96 3b 15 50 10   ········ ··4·;·P·
0030  01 00 24 69 00 00 00                               ··$i···
```

wireshark_07D948D0-BB62-439C-B23D-E82810A4825E_20181018015959_a09340.pcapng | Packets: 40934 · Displayed: 150 (0.4%) · Dropped: 0 (0.0%) | Profile: Default

*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression...