Q1 *I (T)C(P) You (su20-final-q7)* **(26 points)**

EvanBot builds a new course feature that sends announcements to students over TCP. To receive announcements, a student initiates a TCP connection with the server. The server sends the announcements and terminates the connection.

Q1.1 (3 points) Assuming that no adversaries are present, which of the following does communication over a TCP connection guarantee? Select all that apply.

- ■ That both the server and the client can detect if a particular announcement needs to be resent
- ■ That different announcements are delivered in the same order they were sent in
- ☐ That announcements are delivered using the most efficient path through the internet
- ○ None of the above

Solution: TCP guarantees that messages will be retransmitted until they are successfully delivered, and that messages will be delivered in the correct order. TCP makes no guarantees about what path a packet takes through the Internet.

Q1.2 (3 points) When only an on-path adversary is present, which of the following does communication over a TCP connection guarantee? Select all that apply.

- ☐ That both the server and the client can detect if a particular announcement needs to be resent
- ☐ That different announcements are delivered in the same order they were sent in
- ☐ That announcements are delivered using the most efficient path through the internet
- ● None of the above

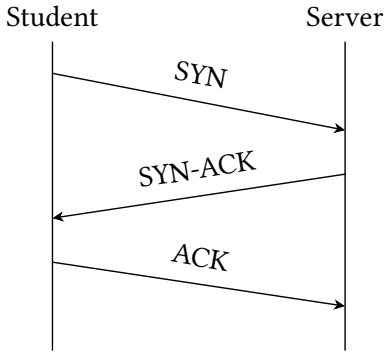
Solution: An on-path attacker has access to the TCP sequence numbers, so they can inject arbitrary messages. Since the attacker can interfere with all messages, TCP no longer has any guarantees about message delivery. TCP still makes no guarantees about what path a packet takes through the Internet

Q1.3 (3 points) Suppose that EvanBot instead sends announcements over UDP. Assuming that no adversaries are present, which of the following might happen? Select all that apply.

- ■ That both the server and the client can detect if a particular announcement needs to be resent
- ■ That different announcements are delivered in the same order they were sent in
- ■ That announcements are delivered using the most efficient path through the internet
- ○ None of the above

Solution: UDP no longer guarantees delivery, so some announcements might not be delivered. However, UDP does not require a handshake at the beginning, so announcements can be delivered more quickly. UDP has no guarantees about what order announcements arrive in, so the server will no longer detect if packets arrive out of order.

EvanBot realizes that the server is sending messages to the student, but the student only responds with ACKs and never sends any messages after the initial handshake. They design a *Half TCP* protocol which provides TCP's properties for communications from the server to the student, but not for communications from the student to the server. This is accomplished using a modified version of the standard three step handshake pictured below.



Q1.4 (5 points) Some sequence numbers are no longer necessary in *Half TCP*. Which fields **do not** need to be transmitted? Select all that apply.

- ■ The sequence number in the SYN packet
- ☐ The sequence number in the SYN-ACK packet
- ■ The ACK number in the SYN-ACK packet
- ■ The sequence number in the ACK packet
- ☐ The ACK number in the ACK packet
- ○ None of the above

Solution: The key insight here is that because the student isn't sending messages to the server, the student's sequence numbers are no longer necessary. The SYN and ACK packets are sent from the student to the server, so their sequence numbers are no longer necessary. The SYN-ACK packet is sent from the server to the student, so its ACK number is no longer necessary.

An earlier version of the solutions incorrectly marked H, K as the set of correct answers. When revising the exam, we changed the question to be "which fields **do not** need to be transmitted," which caused the set of correct answers to be inverted.

Q1.5 (3 points) Which of these are consequences of moving from TCP to *Half TCP* for this application? Select all that apply.

- ☐ The student will no longer receive announcements in the correct order.
- ■ The server will not have to keep track of as much state
- ■ The student will not have to keep track of as much state
- ○ None of the above

Solution: Announcements are sent from the server to the student. We are still using sequence numbers in this direction, so the announcements are still received in the correct order. Because the server and student each only need to keep track of one sequence number instead of two, they both do not need to keep track of as much state.

The 161 staff likes security and decides to use TLS over *Half TCP*. Assume that the staff server has a valid certificate for their public key.

For each different adversary below, select all attacks which become ***easier*** when running TLS over *Half TCP* compared to normal TCP.

Q1.6 (3 points) Off-path adversary

- ■ RST Injection Attack
- ☐ Interfere with a TLS handshake to learn the master key
- ☐ Replay an encrypted command from a previous TLS connection
- ○ None of the above

Q1.7 (3 points) On-path adversary

- ☐ RST Injection Attack
- ☐ Interfere with a TLS handshake to learn the master key
- ☐ Replay an encrypted command from a previous TLS connection
- ● None of the above

Q1.8 (3 points) Man-in-the-middle adversary

- ☐ RST Injection Attack
- ☐ Interfere with a TLS handshake to learn the master key
- ☐ Replay an encrypted command from a previous TLS connection
- ● None of the above

Solution: The key insight here is that attacks on the TLS protocol are not made any easier by using half-TCP, because the cryptographic messages sent between the student and the server are unchanged. The only attack that becomes easier is the RST injection attack for an off-path attacker, since the attacker doesn't need to guess sequence numbers when injecting a RST packet from the student to the server. On-path and MITM attackers can see all sequence numbers, so RST injection is not any easier for them.

Q2 *Mutuality (SP21 Final Q9)* **(0 points)**

(Question 2 continued…)