TCP uses sequences to keep track of how much data has been sent between the two endpoints.

Sequence numbers vs Acknowledgement numbers

## TCP

Session-oriented communication

There’s a setup process that must occur before information can be transferred

There’s a teardown

Three-way handshake

SYN

SYN-ACK

ACK

4-way Teardown

FIN-ACK

ACK

FIN-ACK

ACK

Often the teardown isn’t clean

Just like a phone-call, sometimes one side simply hangs up w/o following the ideal mechanism.

Ending a conversation is much looser than starting one.

Full duplex connection

## FIN vs RST

### FIN

FIN is greatly preferred way of closing a connection.

FIN is like saying, I’m ready to end the conversation, but want to hear everything you have to say first.

RST is like, there’s no conversation, stop talking.

### Reset

An immediate close of a connection.

Allows resources previously allocated to a connection to be reclaimed.

Sent from a device in response to detecting an erroneous condition

Receipt of data from a device it doesn’t have a connection with.

Receipt of data having an invalid sequence or acknowledgement number

Devices receiving a RST flag can do one of several things depending in what state they are in.

If in LISTEN state, ignore the RST

If in SYN-RECEIVED state, but previously in LISTEN state, returns to LISTEN

Otherwise, causes connection to return to CLOSED state and connection to be aborted

RST

Sent in response to a SYN if the server is unable to talk.

Sent during normal data transfer when one side of the connection receives data it isn’t expecting.

Tells the other side that an error has occurred.

Provide a signal that a problematic connection should be closed.

## Idle Connections

A connection which has had no data transported over it for an extended period of time

Keep-alives are special null-segments containing no data intended simply to elicit an acknowledgement from the other side of an open connection, or a RST if the other side is closed.

## Wireshark

Filter by contents of a frame

frame contains “[www.google.com](http://www.google.com)”

Filter by source or destination

Ip.addr == 192.168.1.160

Show data as test

Right click data in protocol => Protocol preferences => Show data as text

Show only tcp frames

tcp

Tcp && ip.addr == 192.168.1.160

Terms

TCP

Transmission Control Protocol

Defined by RFC-793

RFC

Request for Comment

The specification for how things are expected to work

TCP Segment

A part of a TCP stream