

TCP Four-Way Handshake

IETF 92 – Dallas, TX

David A. Borman

Quantum Corporation

March 24, 2015

Agenda

- TCP Four Way Handshake (TCP 4WAY HS)
- Data delivery with TCP 4WAY HS
- TCP options and TCP 4WAY HS
- TCP Unknown Option Option

draft-borman-tcpm-tcp4way-01.txt

david.borman@quantum.com

Background

- Cannot expand TCP option space in initial SYN
 - And maintain backwards compatibility
- Additional TCP options must go in another packet
 - Parallel SYN packets
 - Companion packet sent in parallel
 - Extended negotiation

Goal of Four Way Handshake

- Extend TCP option space during initial Handshake phase
 - Allow client to use EDO option in SYN/ACK
- Provide general mechanism

Three Way Handshake

Client

- Enter SYN-SENT
- SYN(seq=ISSA) ->
- Enter ESTABLISHED
- ACK(ISSB) ->

Server

- Enter SYN-RECEIVED
- <- SYN(seq=ISSB)/
ACK(ISSA)
- Enter ESTABLISHED

Four Way Handshake

Client

- Enter SYN SENT
- SYN (seq=ISSA) ->
- Enter **PRE-ESTAB**
- **SYN(seq=ISSA)/**
ACK(ISSB) ->
- Enter **ESTABLISHED**

Server

- Enter **SYN-ACK SENT**
- <- SYN (seq=ISSB)/
ACK(ISSA)
- Enter ESTABLISHED
- <- **ACK(ISSA)**

RED items indicate changes from Three Way Handshake

3Way ESTABLISHED State and Data

- Each side enters ESTABLISHED state after sending a SYN, and receiving an ACK of that SYN.
- Once in ESTABLISHED state, data may be delivered to the application.
- But, it is receiving the ACK of a SYN that makes it safe to deliver data.

4Way PRE-ESTABLISHED and Data

- Server enters ESTABLISHED on the same timeline as with the Three Way Handshake.
- Client enters PRE-ESTABLISHED with the 4Way handshake when it would have entered ESTABLISHED with the 3Way Handshake.
- Client data delivery can happen in PRE-ESTABLISHED while waiting for the final ACK.

TCP Options and SYN Packets

- Packets with the SYN bit are reliably delivered. Hence, any TCP options in them are also reliably delivered.
- With the 3Way Handshake, for options negotiated in SYN packets, the only reason to not put new options in the server's SYN/ACK is efficiency: they'll be ignored since they weren't in the client's SYN.
- With the 4Way Handshake, the client has a second opportunity to send options in a SYN packet, hence it is reasonable to put new options in the server's SYN/ACK

Changes to TCP State Machine

Passive

- LISTEN -> ALLOW-4WAY? (transitional)
- Yes: SYN-ACK-SENT
- No: SYN-RCVD

Active

- SYN-SENT -> CONTINUE-4WAY? (transitional)
- Yes: PRE-ESTABLISHED (like SYN-RCVD)
- No: ESTABLISHED

Issues with 4Way and TCP options

- One-way option negotiation
 - Server's response depends on client option
 - Can incur 1RTT delay
- Data that is dependent on TCP options
 - Late negotiation will delay data delivery
 - For example, encrypting data.

Implementing TCP 4WAY HS

- 4WAY Bit in TCP header
 - Current proposal in draft document
 - Not the only way
- A new TCP option
 - Uses up more option space in the initial SYN
- Implied support
 - Tied to another TCP option, e.g. EDO

Unknown Option Option

- Provides a mechanism for explicitly stating that a TCP option is unknown
 - Current mechanism only infers by lack of response
- A simple list of unknown option numbers
 - Only sent in response to receiving an unknown option
- Could be made a separate document

Summary

- 4Way Handshake allows for EDO from client
- Data delivery isn't delayed
- Extension to existing TCP state machine

- Questions?