Disabling PAWS When Other Protections Are Available

draft-nishida-tcpm-disabling-paws-00

Yoshifumi Nishida

nishida@sfc.wide.ad.jp

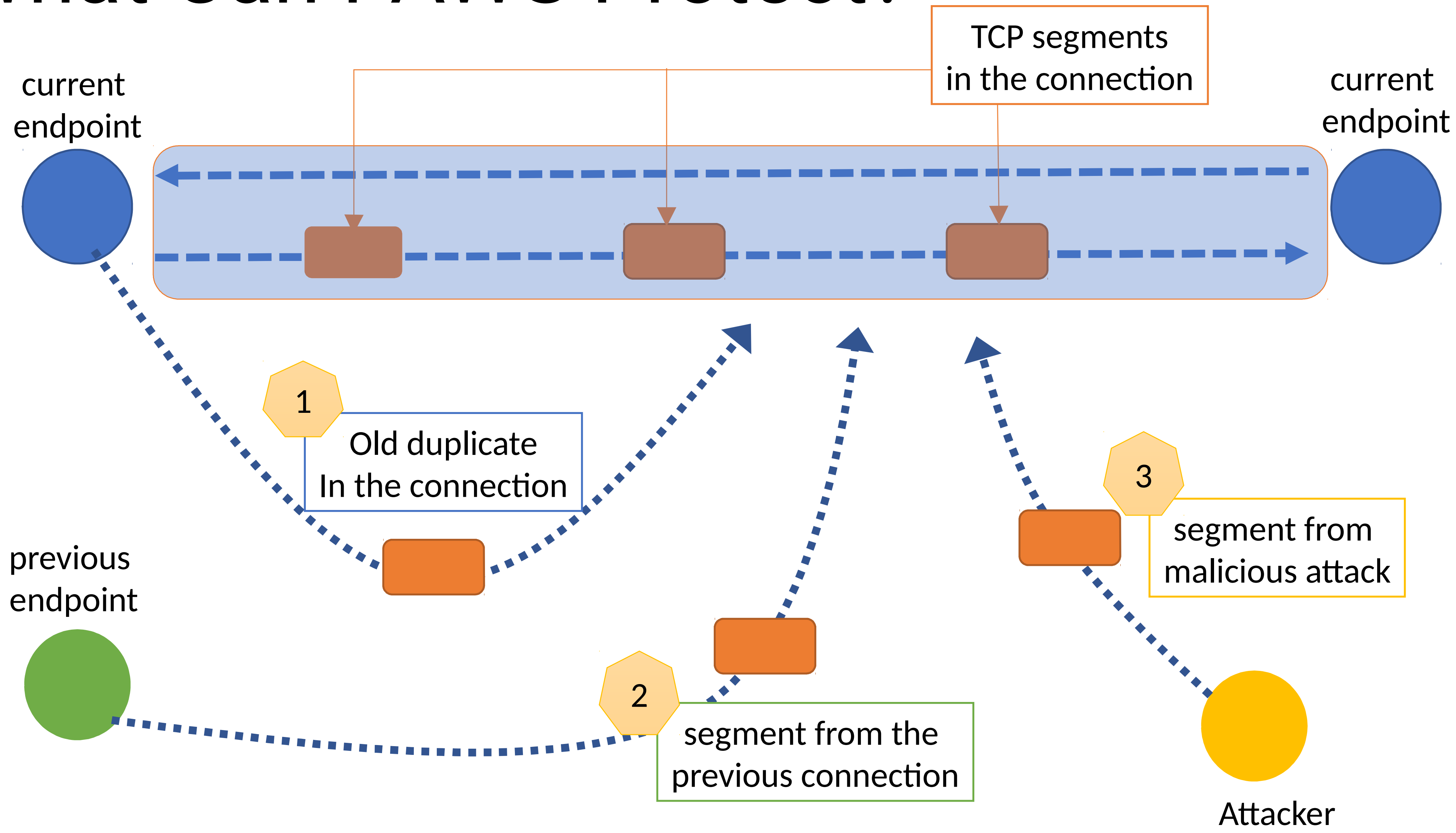
Background

- RFC7323 requires putting timestamp options in ALL segments
    - Timestamp consumes 25-30% of available option space

- Why do we need to put them in ALL segments?
    - RTTM ... Don’t need to measure RTT in every segment
    - PAWS ... Require TS options in all segments to provide protection

- If we have protections other than PAWS, we don’t need to put TS in all segments

What Can PAWS Protect?



How PAWS Works

- Compare TS value in the segment and most recent received TS value
    - If TS value in the segment is newer, PAWS thinks this segment is valid
        - if 0 < t1 – t2 < 2^31, then t1 is newer

- For old duplicate segments in the connection
    - Works! As TS value monotonically increases in a TCP connection
- For segments from previous connections
    - May work. If TS value monotonically increases across all TCP connections
- For segments from attackers
    - Will not work. By using random TS values, attackers' success rate will be 50%

Alternatives for PAWS

- Tcpinc
  - Encrypted segments can provide stronger protection

- MPTCP
  - Maintains 64 bits sequence number space in a session. Data Sequence Signal option can be used as a replacement of PAWS
    - Data Sequence Signal check is more strict than PAWS
- TLS
  - Same as tcpinc. Encrypted segments can provide stronger protection

If these technologies are available in a connection, we can disable PAWS

- They can provide stronger protections than PAWS

Another Possible Benefit

- TIME_WAIT is required to avoid seeing segments from previous connections with the same endpoints
  - 2MSL is required for safety purpose

- If we have new protections, we can recycle connections in TIME_WAIT
  - PAWS may be used for this purpose. But, it is sometime disabled
    - PAWS is not very reliable in some case (e.g when multiple clients behind a NAT)

What Will Be Needed?

- All we need is a signaling mechanism to disable PAWS and to use other protections
    - Check if both sides agreed to use new protections
        - We probably cannot disable PAWS without checking
            - RFC7323 requires to discard segments without TS option after it is negotiated

Possible Signaling Mechanisms

- New TCP options
  - Negotiate the feature during SYN exchange
- Extend TS option for feature negotiation
  - draft-scheffenegger-tcpm-timestamp-negotiation
- Extend protection mechanism
  - TCPINC … use 1 bit of gobal suboption in eno?
  - MPTCP … Extend MP_CAPABLE or use MP_EXPERIMENTAL option?

Discussions

- Does this look a good topic to proceed?

- If so, what should be done to be adopted?