TCP-ENO: Encryption Negotiation Option

draft-ietf-tcpinc-tcpeno

Andrea Bittau, Dan Boneh, Daniel Giffin, Mark Handley,
David Mazières, and Eric Smith

IETF95

Thursday, April 7, 2016

Goal

Abstract away details of TCPINC encryption protocols

Facilitate adoption of future TCP-level encryption specs

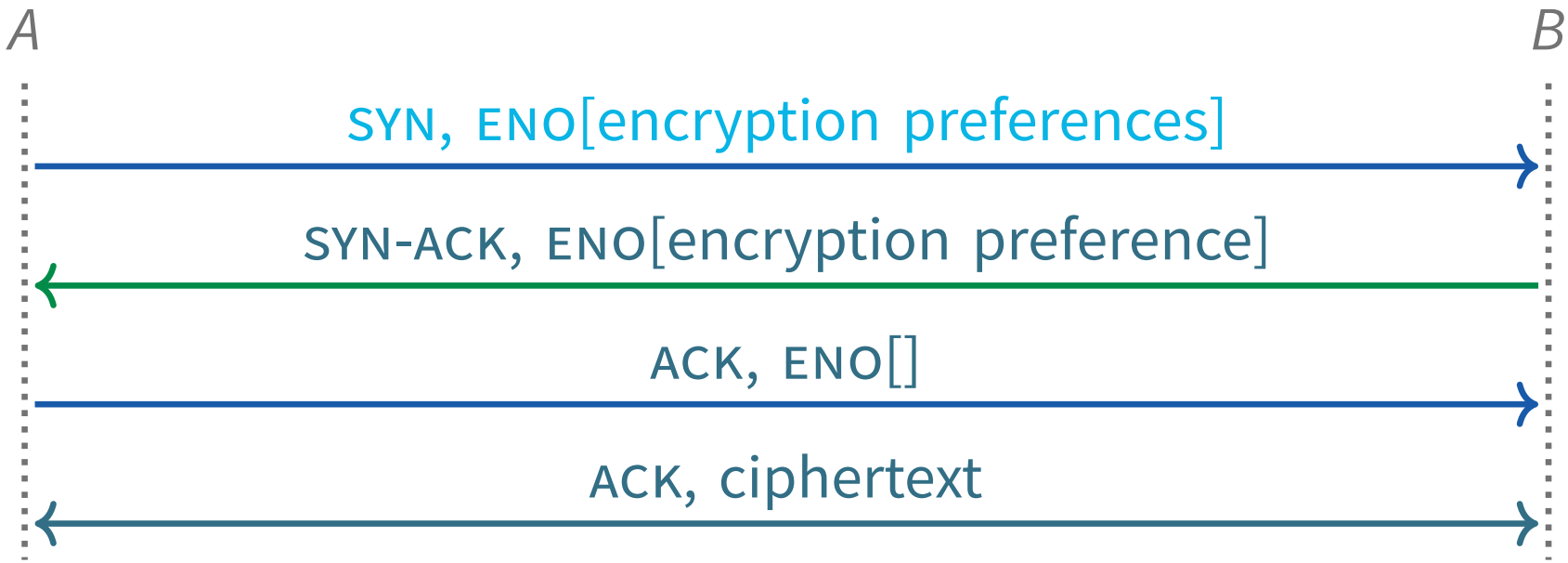
- New specs do not require additional TCP option kinds
- New specs incrementally deployable, fall back to older specs
- New specs compatibile with existing TCPINC-aware applications (recall charter requires authentication hooks)

Minimize consumption of TCP option space

Avoid unnecessary round trips for connection setup

Revert to unencrypted TCP when encryption not possible

Overview of common case



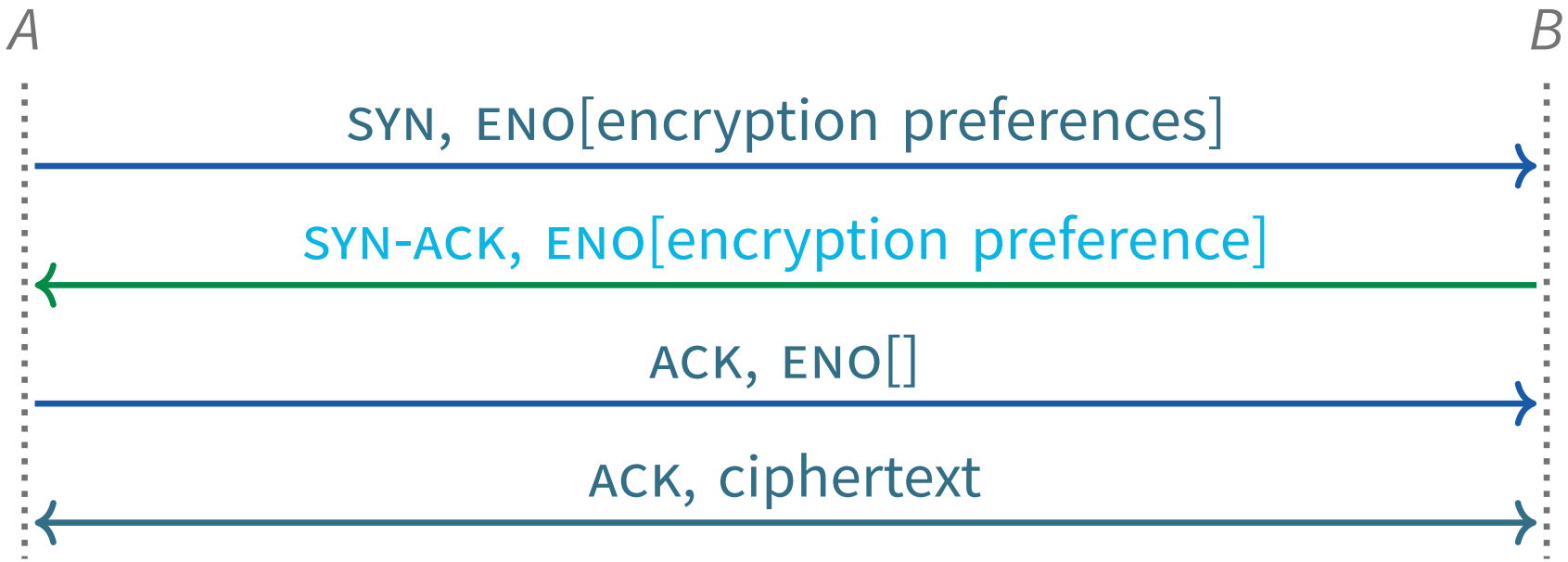
Active opener *A* lists spec preferences in ENO option

Passive opener *B* lists spec preferences in ENO option

***A* sends empty ENO option indicating encryption enabled**

If any of the above ENOs missing, revert to unencrypted TCP

Overview of common case



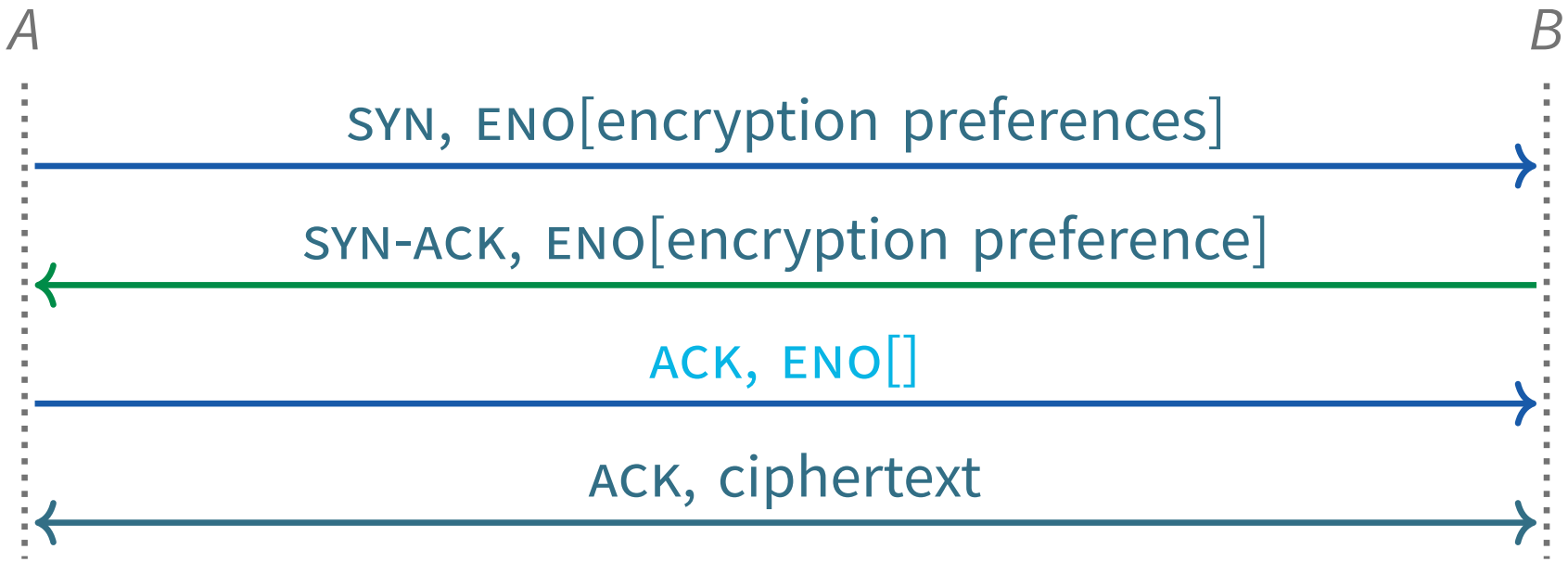
Active opener *A* lists spec preferences in ENO option

Passive opener *B* lists spec preferences in ENO option

***A* sends empty ENO option indicating encryption enabled**

If any of the above ENOs missing, revert to unencrypted TCP

Overview of common case



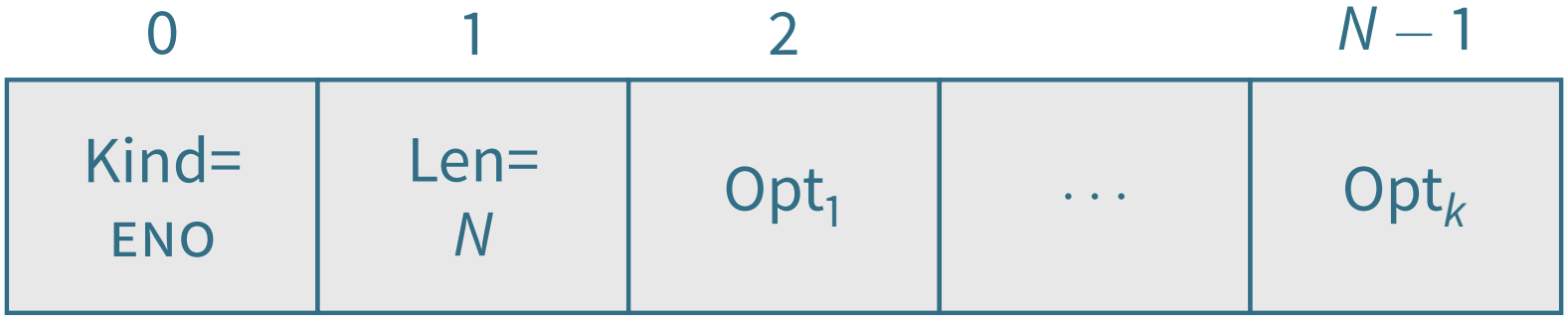
Active opener *A* lists spec preferences in ENO option

Passive opener *B* lists spec preferences in ENO option

***A* sends empty ENO option indicating encryption enabled**

If any of the above ENOs missing, revert to unencrypted TCP

ENO option in SYN segments



ENO is a container for a set of *suboptions*

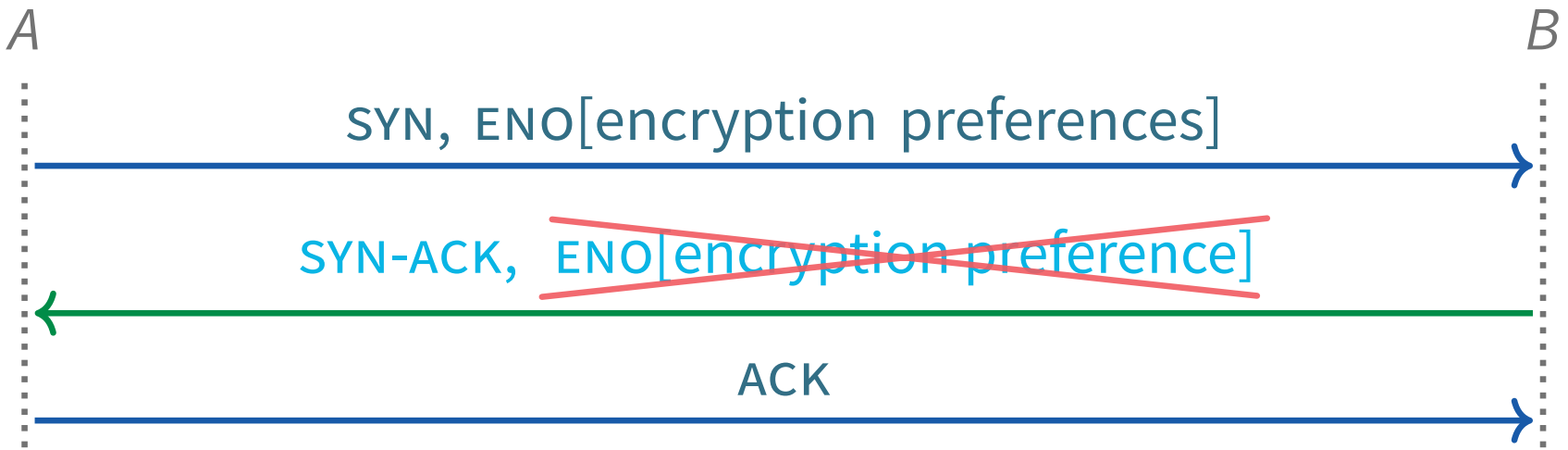
Zero or one *general* suboption

One or more *spec identifier* suboptions

- Lists supported encryption specs
- Host *B* (passive opener) SHOULD list only one spec if possible
- Otherwise, *B* lists in order of increasing preference

New: At most one ENO option allowed per SYN segment

ENO option in non-SYN segments



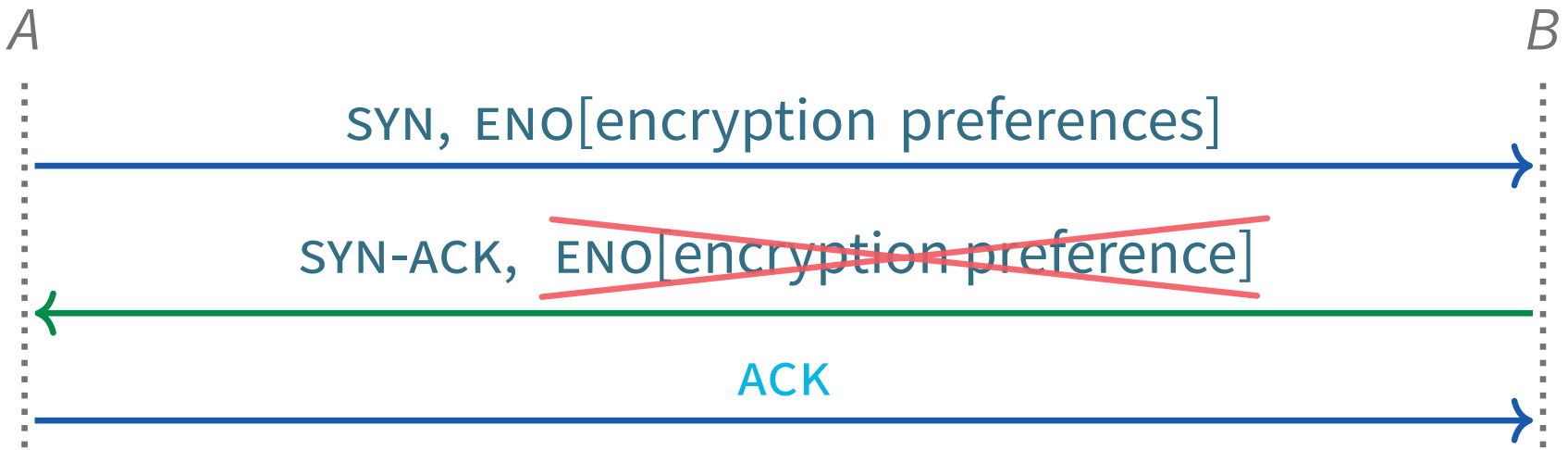
What if middlebox strips ENO from SYN-ACK?

- *B* disables encryption when first non-SYN segment has no ENO

Contents of ENO options in non-SYN segments does not matter

- ENO option contents available to specs for all non-SYN segments
- Responsibility of spec to authenticate data (new: clarified in draft)

ENO option in non-SYN segments



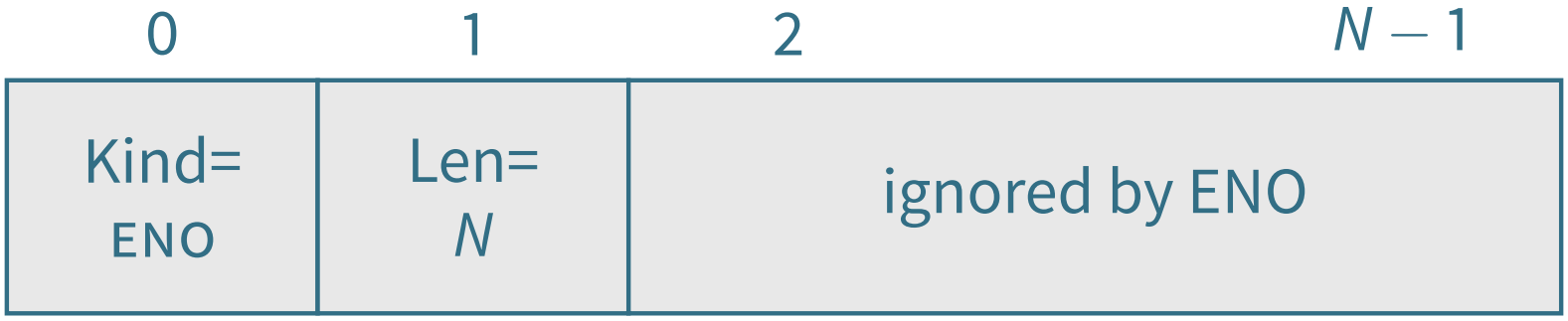
What if middlebox strips ENO from SYN-ACK?

- *B* disables encryption when first non-SYN segment has no ENO

Contents of ENO options in non-SYN segments does not matter

- ENO option contents available to specs for all non-SYN segments
- Responsibility of spec to authenticate data (new: clarified in draft)

ENO option in non-SYN segments



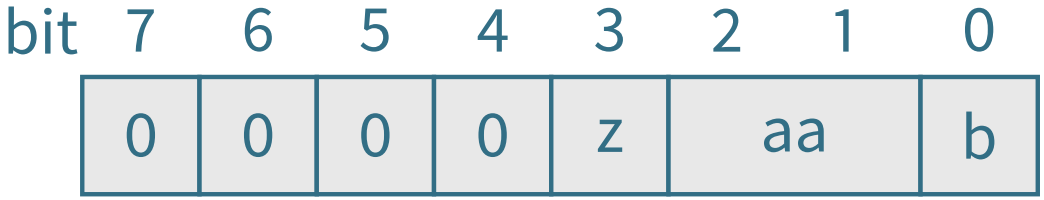
What if middlebox strips ENO from SYN-ACK?

- *B* disables encryption when first non-SYN segment has no ENO

Contents of ENO options in non-SYN segments does not matter

- ENO option contents available to specs for all non-SYN segments
- Responsibility of spec to authenticate data (new: clarified in draft)

The general suboption byte



b: "I want to play *B* role even if I'm active opener"

- Required to break symmetry of simultaneous open
- New: b bit more significant than active/passive opener "p" bit
- To fix: clarify that it is set by application using extended API

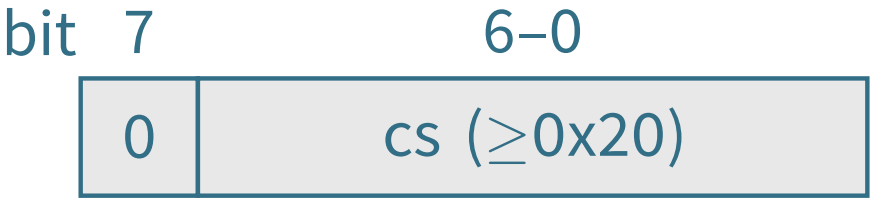
aa: Out of band signaling that application is aware of ENO

- Enables applications to negotiate use authentication hooks
  - 00: Application is not aware of TCP-ENO (default)
  - 01: Application is aware of TCP-ENO
  - 10: Reserved (do not send, interpret same as 01)
  - 11: Application is aware, disable TCP-ENO if peer application is not

z: Reserved (send as 0 and ignore on receipt)

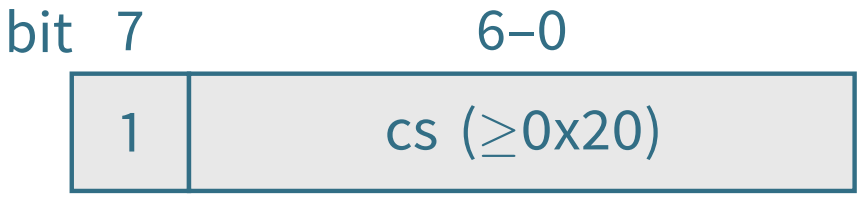
Spec identifier suboptions

One-byte suboptions indicate support for spec cs:

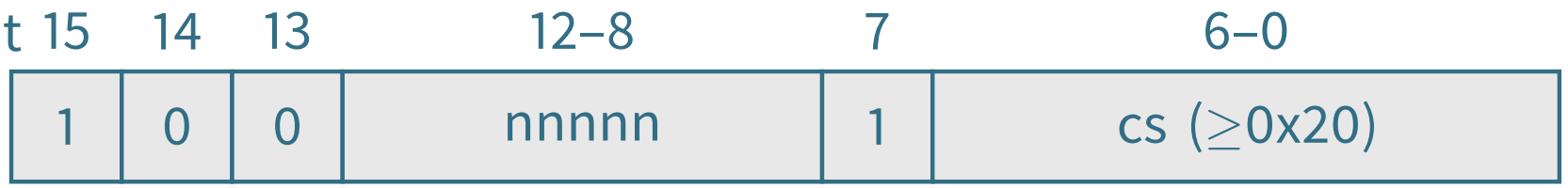


Multi-byte options followed by spec-specific suboption data

- Simple multibyte option extends to end of TCP ENO option, introduced by setting high bit:

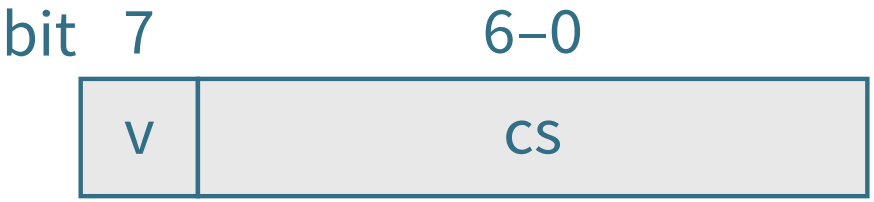


- New: delimited suboption followed by (1 + nnnnn)-bytes of data



- Also 12-bit length option, probably not useful before long options

Summary of suboption bytes



v	cs	meaning
0	0x00-0x0f	general suboption
0	0x10-0x1f	reserved
1	0x00-0x1f	Length byte (length given by **cs**)
0	0x20-0x7f	Supoption without data
1	0x20-0x7f	Supoption followed by data

Credit: Yoav Nir

Summary of changes

At most one TCP-ENO option allowed per SYN segment

- Avoids problems with middleboxes reordering options

Delimited suboptions allow multiple ENO suboptions with data

- One suboption with data still requires no extra bytes
- Need extra marker byte for each additional suboption with data
- Previously reserved cs values 16–31 now used by markers

Role priority bit now the most significant bit in role selection

- Works even when one side does not know of simultaneous open

Still to do

Some oblique points in draft should be clarified or removed

- E.g., application to DANE, what happens with future changes to size of crypto keys, TCP segments, or options

Adjust delimited option terminology

- Maybe “length byte” instead of “marker”

Client/server vs. active/passive opener

- We find active/passive less ambiguous (in keeping with RFC793)
- Alternative: explicitly redefine “Client” and “Server” like RFC7413 (TFO)

Fine-tune forward secrecy requirements