0-RTT TCP Converters

draft-ietf-tcpm-converters-06
IETF104, March 2019

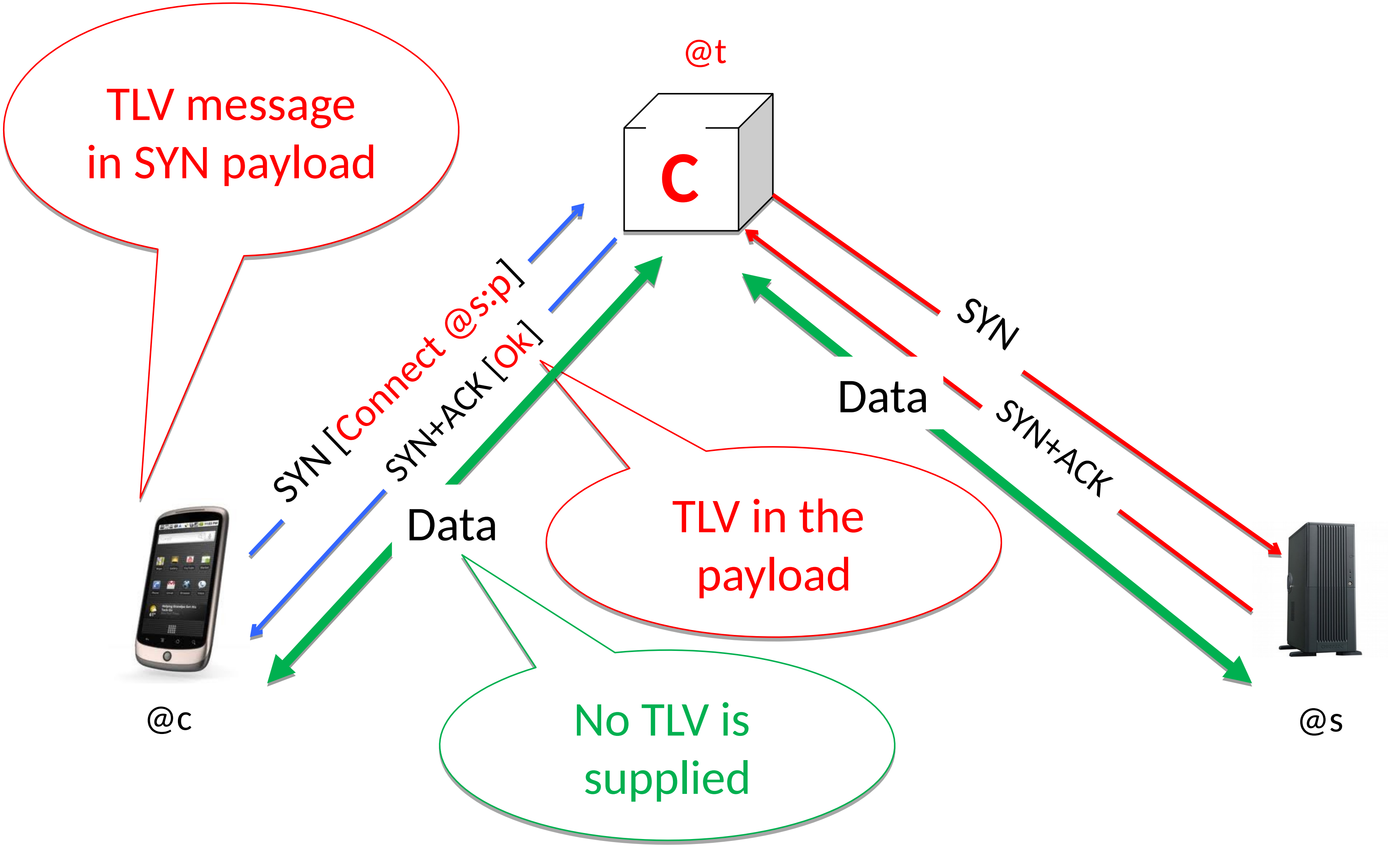
O. Bonaventure

M. Boucadair

The Basic Design

- Converter Protocol is an **application-level protocol** listening on a specific TCP port
  - Commands and responses are encoded as **TLVs**
    - Ensures extensibility
  - Commands are sent inside SYN
    - Provides **0-RTT** to minimize connection establishment delays
  - Responses are returned in SYN+ACK
  - A **plain transport mode** is used between Clients and Converters (no encapsulation)
- Clients can learn TCP options supported by Servers
  - Allows Clients to **bypass** the Converter

A Simplified Example

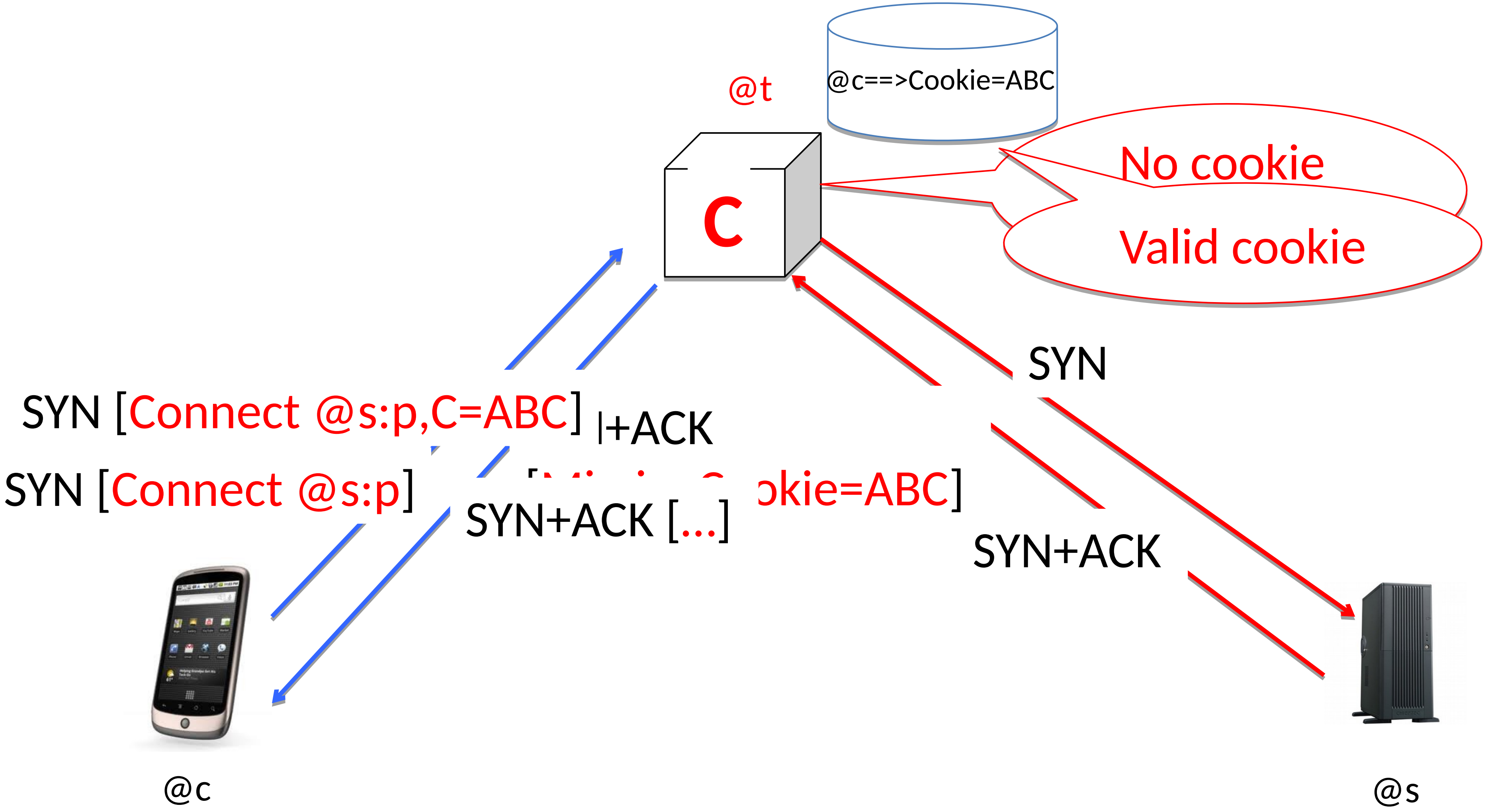


Main Changes Since IETF#103

- Integrate feedback from implementors
  - Various tweaks and clarifications
    - Removed error TLV from RSTs since some stacks cannot send/parse such packets easily
  - Open-source client library and wireshark dissectors released by Tessares (See https://www.tessares.net/technology/open-source-contributions/)

- Simplified the design by removing the requirement from using TFO
  - The protection provided by TFO in the previous design is now provided in the Convert protocol itself

The Converter Cookie



@t

@c==>Cookie=ABC

C

No cookie

Valid cookie

SYN [Connect @s:p,C=ABC]

]+ACK

SYN [Connect @s:p]

[Missing Cookie=ABC]

SYN+ACK [...]

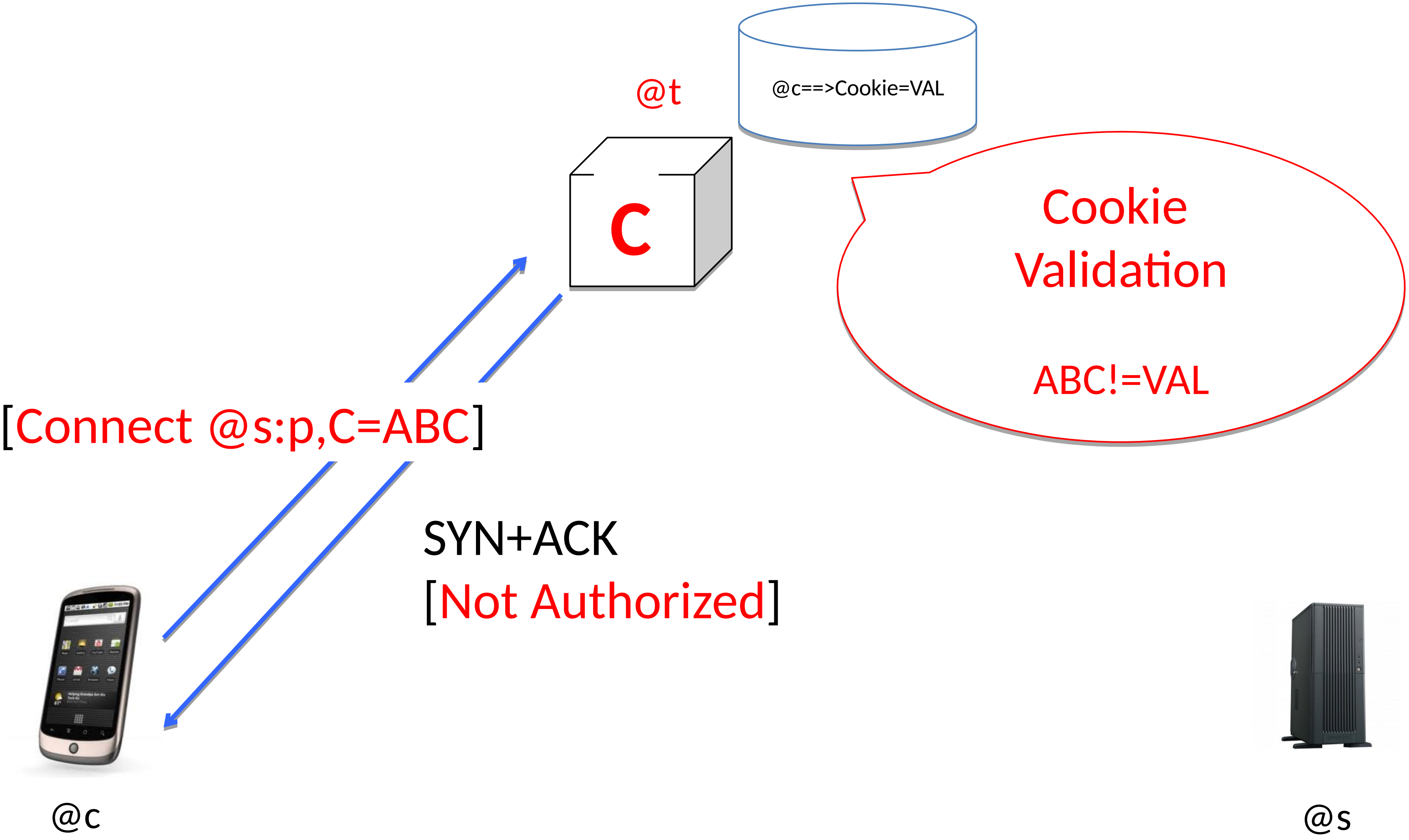
SYN

SYN+ACK

@c

@s

The Converter Cookie



@t

@c==>Cookie=VAL

C

Cookie Validation

ABC!=VAL

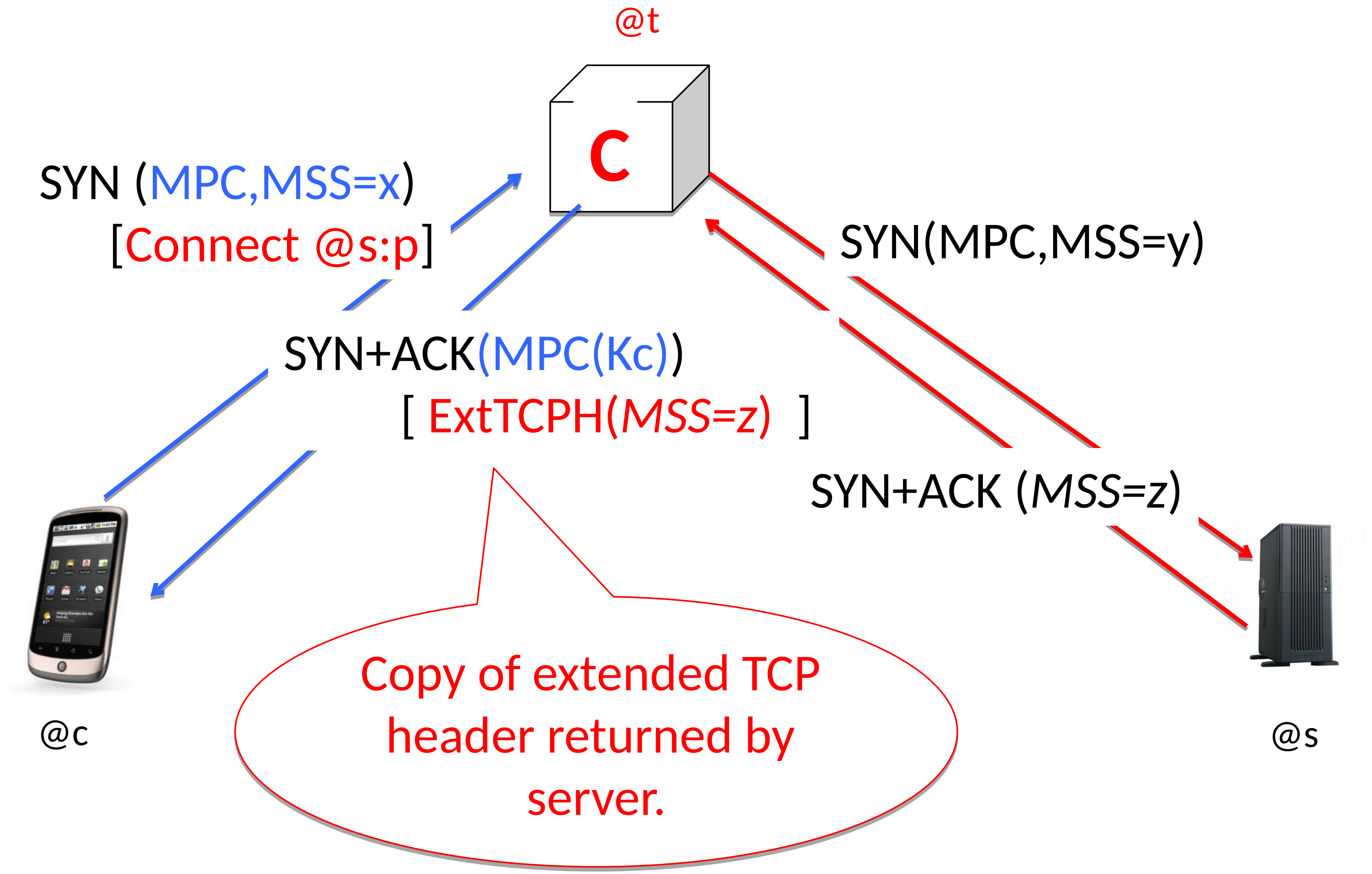
SYN [Connect @s:p,C=ABC]

SYN+ACK [Not Authorized]

@c

@s

Converted-Assisted MPTCP



@t

C

SYN (MPC,MSS=x)
[Connect @s:p]

SYN(MPC,MSS=y)

SYN+ACK(MPC(Kc))
[ExtTCPH(*MSS=z*)]

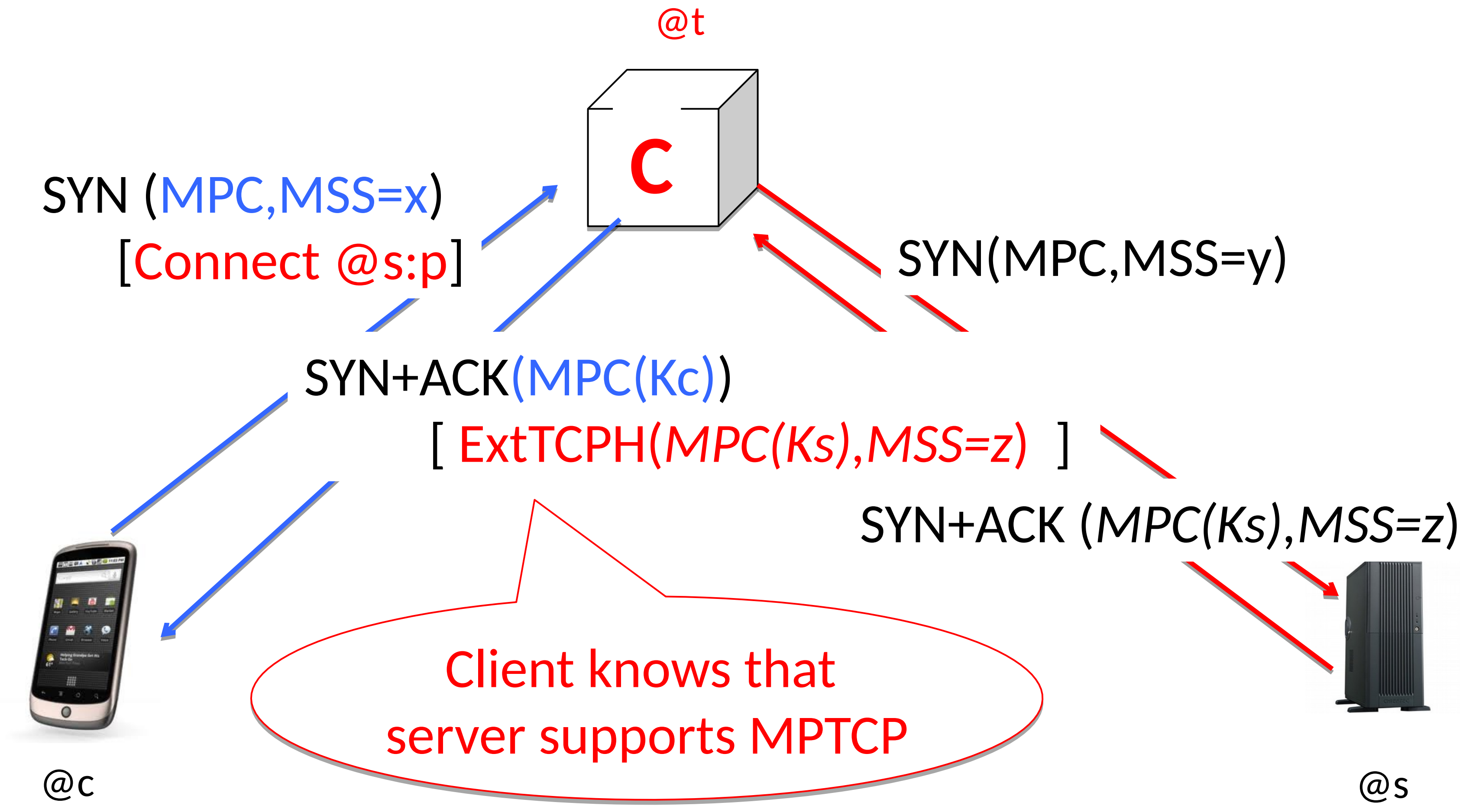
SYN+ACK (*MSS*=z)

Copy of extended TCP header returned by server.

@c

@s

Converted-Assisted MPTCP: Bypass



Client knows that server supports MPTCP

Status & Next Step

- A simplified design which takes into account feedback from implementors and comments raised during email discussions

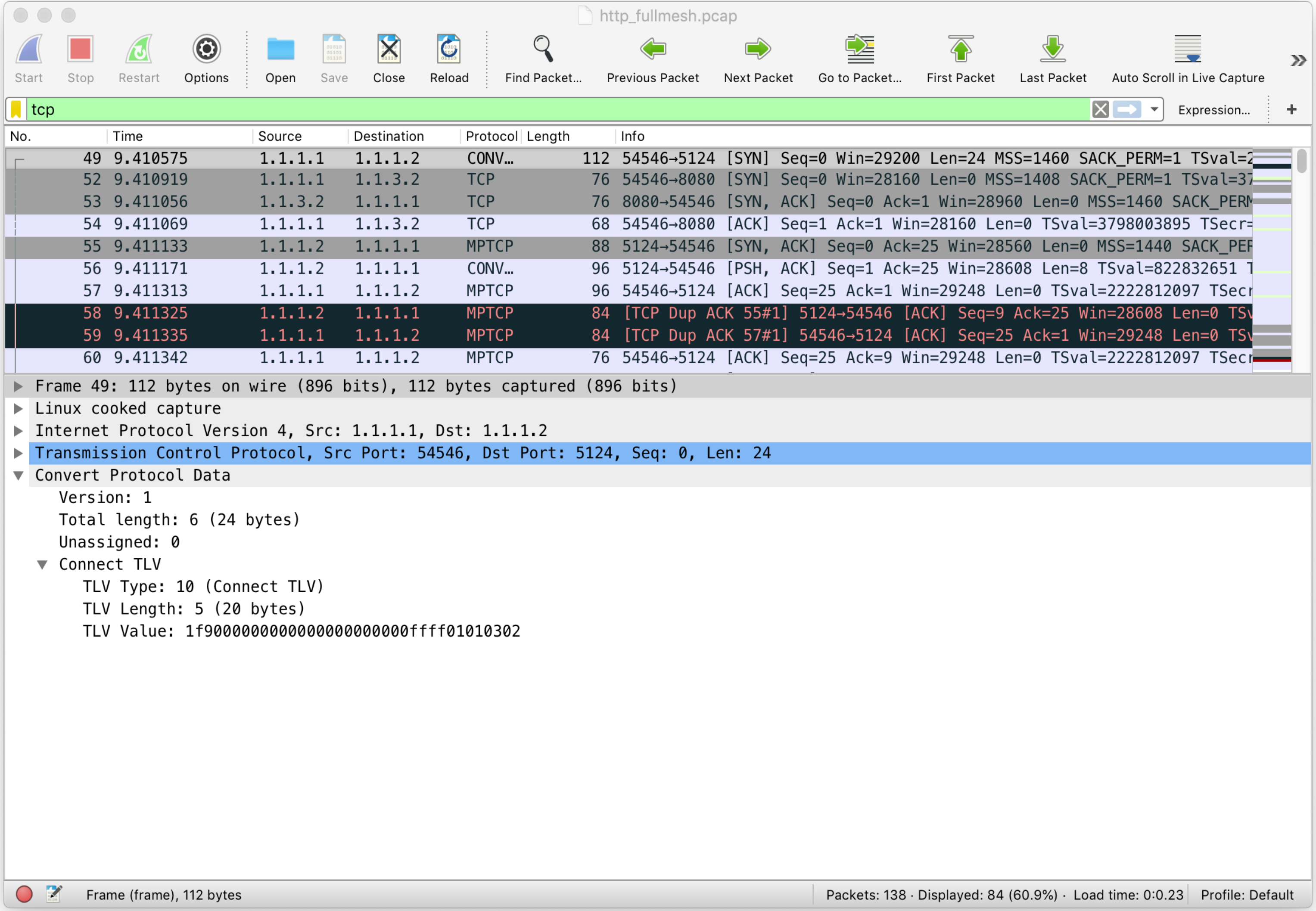
- Adoption from other standardisation bodies
  - Broadband Forum for WT-378
  - 3GPP for the ATSSS service in 5G networks (TS 23.501)

- We believe that the document is ready for WG Last Call

Backup

- Examples from a first packet trace

SYN from client to converter



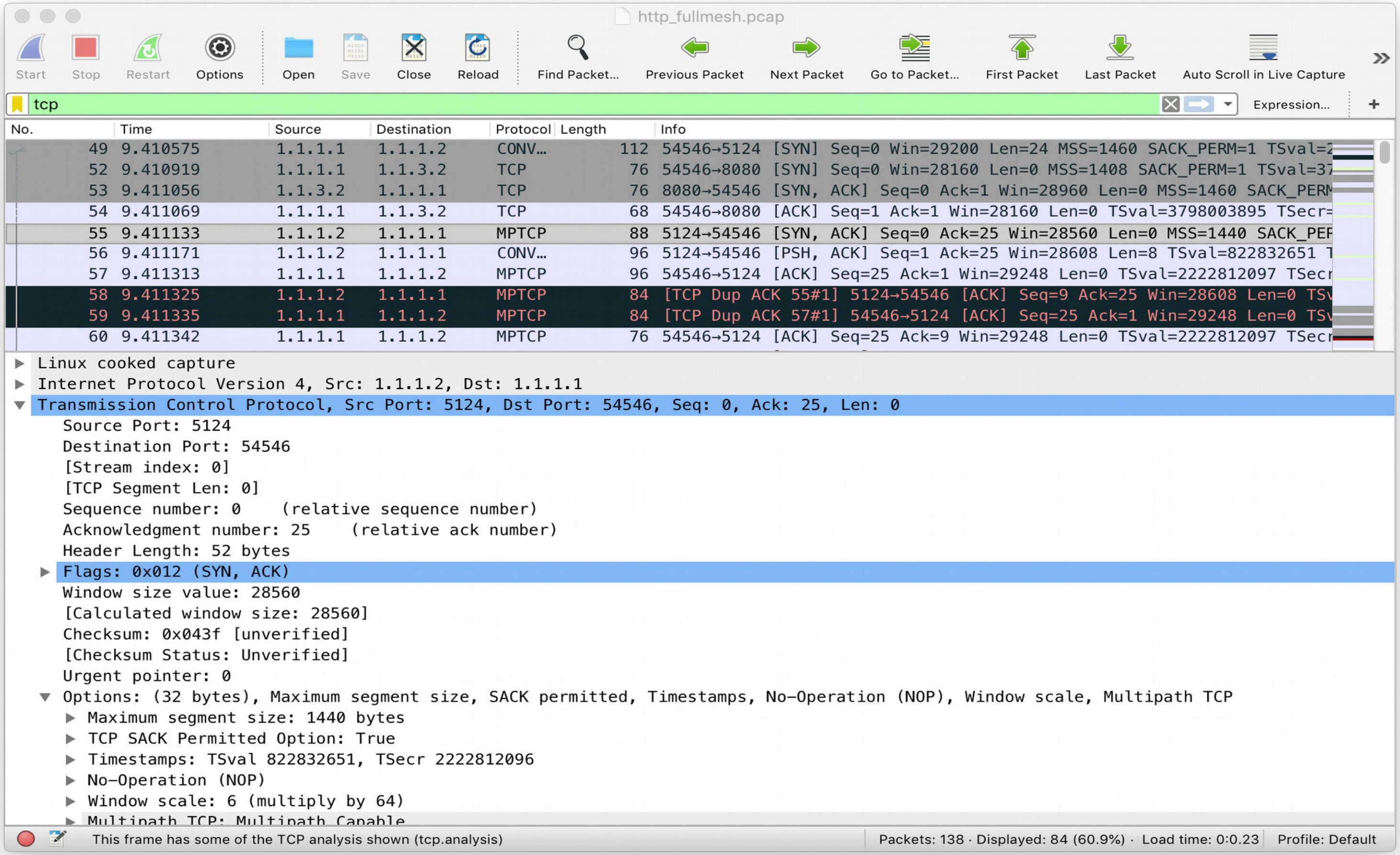
tcp

```
No.     Time       Source    Destination  Protocol Length Info
    49  9.410575   1.1.1.1   1.1.1.2      CONV…    112 54546→5124 [SYN] Seq=0 Win=29200 Len=24 MSS=1460 SACK_PERM=1 TSval=2
    52  9.410919   1.1.1.1   1.1.3.2      TCP       76 54546→8080 [SYN] Seq=0 Win=28160 Len=0 MSS=1408 SACK_PERM=1 TSval=37
    53  9.411056   1.1.3.2   1.1.1.1      TCP       76 8080→54546 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM
    54  9.411069   1.1.1.1   1.1.3.2      TCP       68 54546→8080 [ACK] Seq=1 Ack=1 Win=28160 Len=0 TSval=3798003895 TSecr=
    55  9.411133   1.1.1.2   1.1.1.1      MPTCP     88 5124→54546 [SYN, ACK] Seq=0 Ack=25 Win=28560 Len=0 MSS=1440 SACK_PER
    56  9.411171   1.1.1.2   1.1.1.1      CONV…     96 5124→54546 [PSH, ACK] Seq=1 Ack=25 Win=28608 Len=8 TSval=822832651 T
    57  9.411313   1.1.1.1   1.1.1.2      MPTCP     96 54546→5124 [ACK] Seq=25 Ack=1 Win=29248 Len=0 TSval=2222812097 TSecr
    58  9.411325   1.1.1.2   1.1.1.1      MPTCP     84 [TCP Dup ACK 55#1] 5124→54546 [ACK] Seq=9 Ack=25 Win=28608 Len=0 TSv
    59  9.411335   1.1.1.1   1.1.1.2      MPTCP     84 [TCP Dup ACK 57#1] 54546→5124 [ACK] Seq=25 Ack=1 Win=29248 Len=0 TSv
    60  9.411342   1.1.1.1   1.1.1.2      MPTCP     76 54546→5124 [ACK] Seq=25 Ack=9 Win=29248 Len=0 TSval=2222812097 TSecr
```

```
▶ Frame 49: 112 bytes on wire (896 bits), 112 bytes captured (896 bits)
▶ Linux cooked capture
▶ Internet Protocol Version 4, Src: 1.1.1.1, Dst: 1.1.1.2
▶ Transmission Control Protocol, Src Port: 54546, Dst Port: 5124, Seq: 0, Len: 24
▼ Convert Protocol Data
    Version: 1
    Total length: 6 (24 bytes)
    Unassigned: 0
  ▼ Connect TLV
      TLV Type: 10 (Connect TLV)
      TLV Length: 5 (20 bytes)
      TLV Value: 1f90000000000000000000000ffff01010302
```

Frame (frame), 112 bytes | Packets: 138 · Displayed: 84 (60.9%) · Load time: 0:0.23 | Profile: Default

SYN+ACK returned by converter



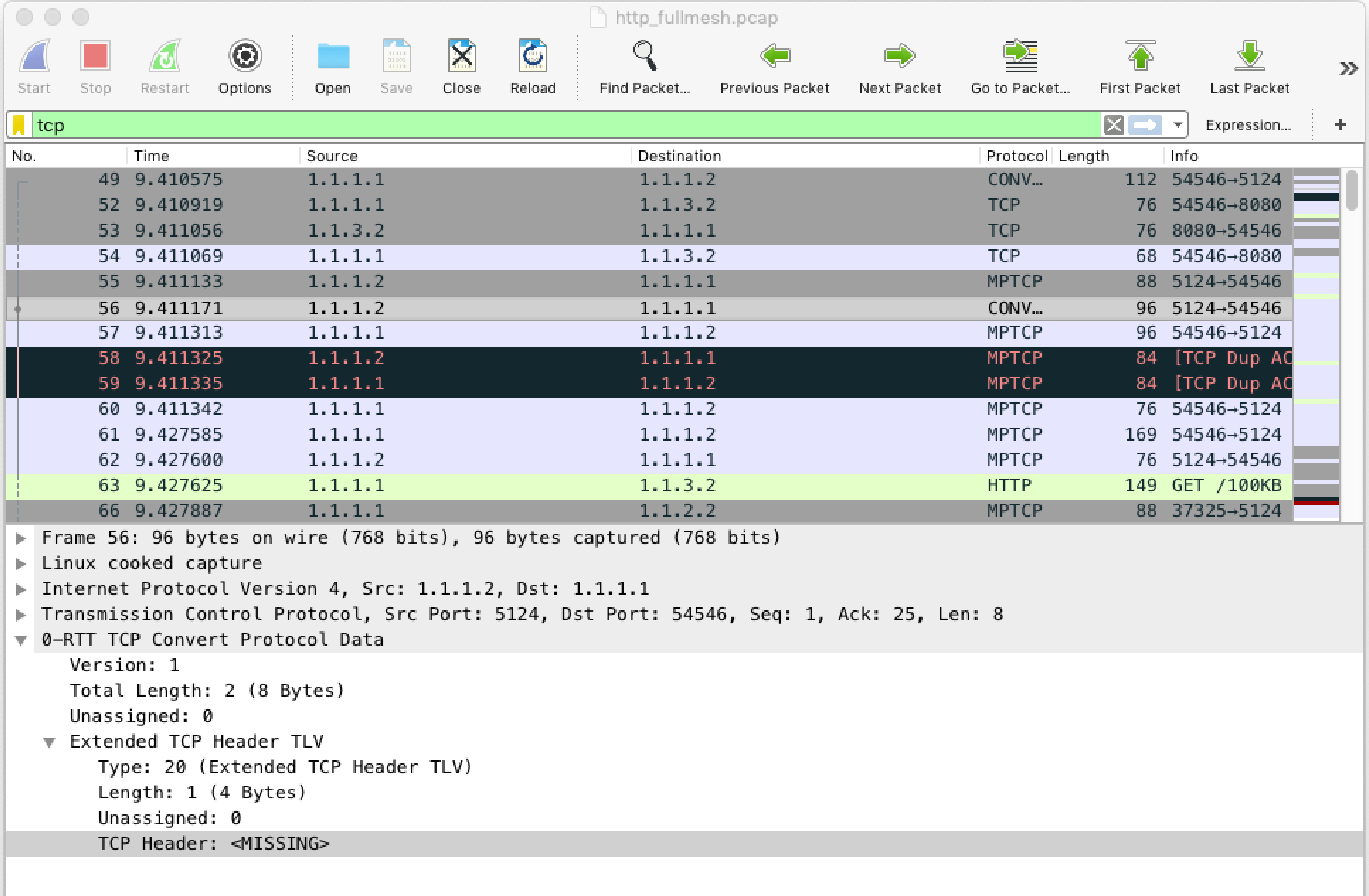
```
tcp
```

```
No.        Time        Source     Destination  Protocol Length  Info
   49 9.410575    1.1.1.1    1.1.1.2      CONV…      112  54546→5124 [SYN] Seq=0 Win=29200 Len=24 MSS=1460 SACK_PERM=1 TSval=2
   52 9.410919    1.1.1.1    1.1.3.2      TCP         76  54546→8080 [SYN] Seq=0 Win=28160 Len=0 MSS=1408 SACK_PERM=1 TSval=37
   53 9.411056    1.1.3.2    1.1.1.1      TCP         76  8080→54546 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM
   54 9.411069    1.1.1.1    1.1.3.2      TCP         68  54546→8080 [ACK] Seq=1 Ack=1 Win=28160 Len=0 TSval=3798003895 TSecr=
   55 9.411133    1.1.1.2    1.1.1.1      MPTCP       88  5124→54546 [SYN, ACK] Seq=0 Ack=25 Win=28560 Len=0 MSS=1440 SACK_PER
   56 9.411171    1.1.1.2    1.1.1.1      CONV…       96  5124→54546 [PSH, ACK] Seq=1 Ack=25 Win=28608 Len=8 TSval=822832651 T
   57 9.411313    1.1.1.1    1.1.1.2      MPTCP       96  54546→5124 [ACK] Seq=25 Ack=1 Win=29248 Len=0 TSval=2222812097 TSecr
   58 9.411325    1.1.1.2    1.1.1.1      MPTCP       84  [TCP Dup ACK 55#1] 5124→54546 [ACK] Seq=9 Ack=25 Win=28608 Len=0 TSv
   59 9.411335    1.1.1.1    1.1.1.2      MPTCP       84  [TCP Dup ACK 57#1] 54546→5124 [ACK] Seq=25 Ack=1 Win=29248 Len=0 TSv
   60 9.411342    1.1.1.1    1.1.1.2      MPTCP       76  54546→5124 [ACK] Seq=25 Ack=9 Win=29248 Len=0 TSval=2222812097 TSecr
```

```
▶ Linux cooked capture
▶ Internet Protocol Version 4, Src: 1.1.1.2, Dst: 1.1.1.1
▼ Transmission Control Protocol, Src Port: 5124, Dst Port: 54546, Seq: 0, Ack: 25, Len: 0
    Source Port: 5124
    Destination Port: 54546
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 0    (relative sequence number)
    Acknowledgment number: 25    (relative ack number)
    Header Length: 52 bytes
  ▶ Flags: 0x012 (SYN, ACK)
    Window size value: 28560
    [Calculated window size: 28560]
    Checksum: 0x043f [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
  ▼ Options: (32 bytes), Maximum segment size, SACK permitted, Timestamps, No-Operation (NOP), Window scale, Multipath TCP
    ▶ Maximum segment size: 1440 bytes
    ▶ TCP SACK Permitted Option: True
    ▶ Timestamps: TSval 822832651, TSecr 2222812096
    ▶ No-Operation (NOP)
    ▶ Window scale: 6 (multiply by 64)
    ▶ Multipath TCP: Multipath Capable
```

This frame has some of the TCP analysis shown (tcp.analysis) | Packets: 138 · Displayed: 84 (60.9%) · Load time: 0:0.23 | Profile: Default

Response returned by converter

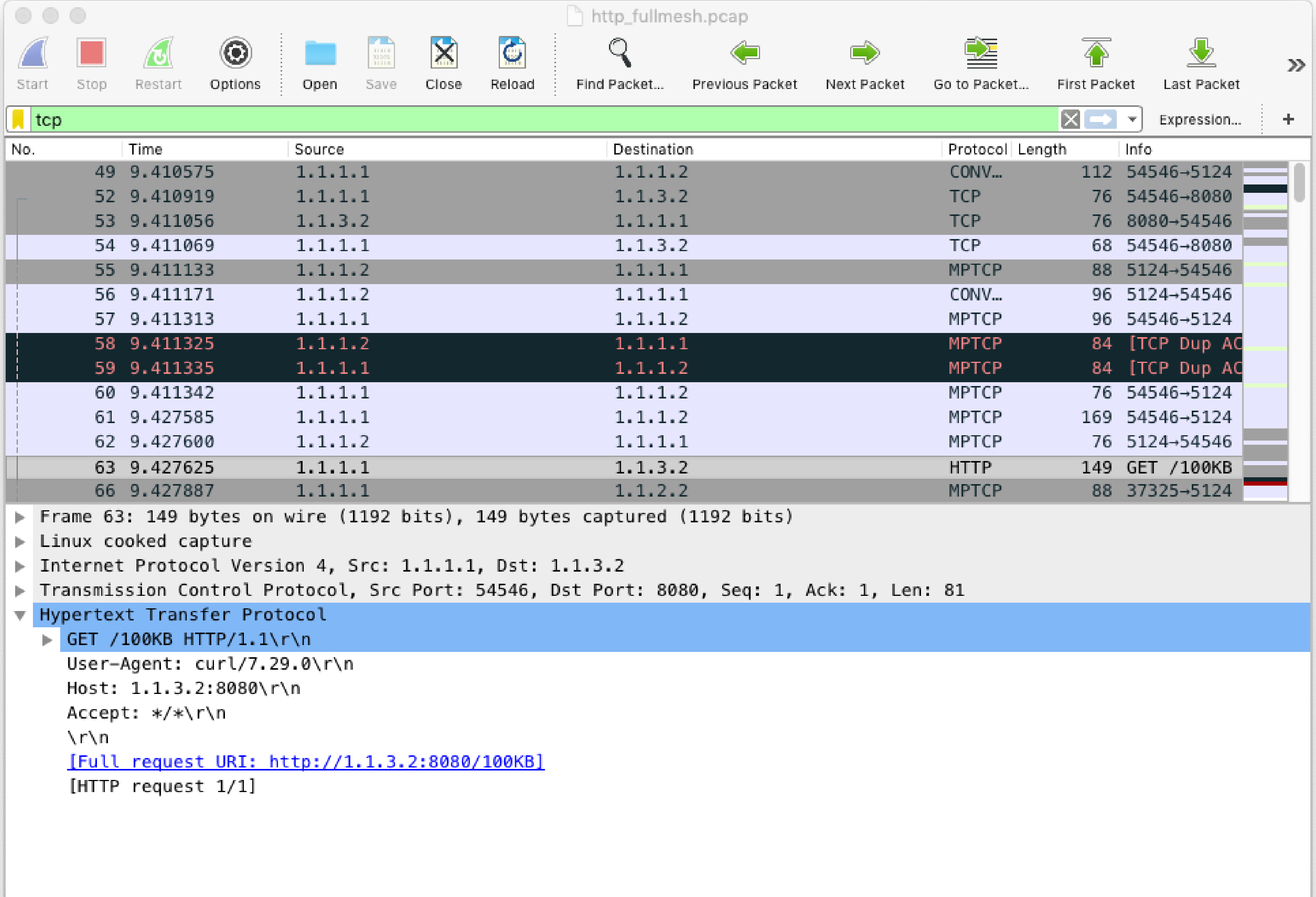


tcp

No.	Time	Source	Destination	Protocol	Length	Info
49	9.410575	1.1.1.1	1.1.1.2	CONV…	112	54546→5124
52	9.410919	1.1.1.1	1.1.3.2	TCP	76	54546→8080
53	9.411056	1.1.3.2	1.1.1.1	TCP	76	8080→54546
54	9.411069	1.1.1.1	1.1.3.2	TCP	68	54546→8080
55	9.411133	1.1.1.2	1.1.1.1	MPTCP	88	5124→54546
56	9.411171	1.1.1.2	1.1.1.1	CONV…	96	5124→54546
57	9.411313	1.1.1.1	1.1.1.2	MPTCP	96	54546→5124
58	9.411325	1.1.1.2	1.1.1.1	MPTCP	84	[TCP Dup AC
59	9.411335	1.1.1.1	1.1.1.2	MPTCP	84	[TCP Dup AC
60	9.411342	1.1.1.1	1.1.1.2	MPTCP	76	54546→5124
61	9.427585	1.1.1.1	1.1.1.2	MPTCP	169	54546→5124
62	9.427600	1.1.1.2	1.1.1.1	MPTCP	76	5124→54546
63	9.427625	1.1.1.1	1.1.3.2	HTTP	149	GET /100KB
66	9.427887	1.1.1.1	1.1.2.2	MPTCP	88	37325→5124

```
▶ Frame 56: 96 bytes on wire (768 bits), 96 bytes captured (768 bits)
▶ Linux cooked capture
▶ Internet Protocol Version 4, Src: 1.1.1.2, Dst: 1.1.1.1
▶ Transmission Control Protocol, Src Port: 5124, Dst Port: 54546, Seq: 1, Ack: 25, Len: 8
▼ 0-RTT TCP Convert Protocol Data
    Version: 1
    Total Length: 2 (8 Bytes)
    Unassigned: 0
  ▼ Extended TCP Header TLV
      Type: 20 (Extended TCP Header TLV)
      Length: 1 (4 Bytes)
      Unassigned: 0
      TCP Header: <MISSING>
```

HTTP GET from client



tcp

No.	Time	Source	Destination	Protocol	Length	Info
49	9.410575	1.1.1.1	1.1.1.2	CONV...	112	54546→5124
52	9.410919	1.1.1.1	1.1.3.2	TCP	76	54546→8080
53	9.411056	1.1.3.2	1.1.1.1	TCP	76	8080→54546
54	9.411069	1.1.1.1	1.1.3.2	TCP	68	54546→8080
55	9.411133	1.1.1.2	1.1.1.1	MPTCP	88	5124→54546
56	9.411171	1.1.1.2	1.1.1.1	CONV...	96	5124→54546
57	9.411313	1.1.1.1	1.1.1.2	MPTCP	96	54546→5124
58	9.411325	1.1.1.2	1.1.1.1	MPTCP	84	[TCP Dup AC
59	9.411335	1.1.1.1	1.1.1.2	MPTCP	84	[TCP Dup AC
60	9.411342	1.1.1.1	1.1.1.2	MPTCP	76	54546→5124
61	9.427585	1.1.1.1	1.1.1.2	MPTCP	169	54546→5124
62	9.427600	1.1.1.2	1.1.1.1	MPTCP	76	5124→54546
63	9.427625	1.1.1.1	1.1.3.2	HTTP	149	GET /100KB
66	9.427887	1.1.1.1	1.1.2.2	MPTCP	88	37325→5124

```
▶ Frame 63: 149 bytes on wire (1192 bits), 149 bytes captured (1192 bits)
▶ Linux cooked capture
▶ Internet Protocol Version 4, Src: 1.1.1.1, Dst: 1.1.3.2
▶ Transmission Control Protocol, Src Port: 54546, Dst Port: 8080, Seq: 1, Ack: 1, Len: 81
▼ Hypertext Transfer Protocol
  ▶ GET /100KB HTTP/1.1\r\n
    User-Agent: curl/7.29.0\r\n
    Host: 1.1.3.2:8080\r\n
    Accept: */*\r\n
    \r\n
    [Full request URI: http://1.1.3.2:8080/100KB]
    [HTTP request 1/1]
```