MPTCP extensions: DSS & Ext keys

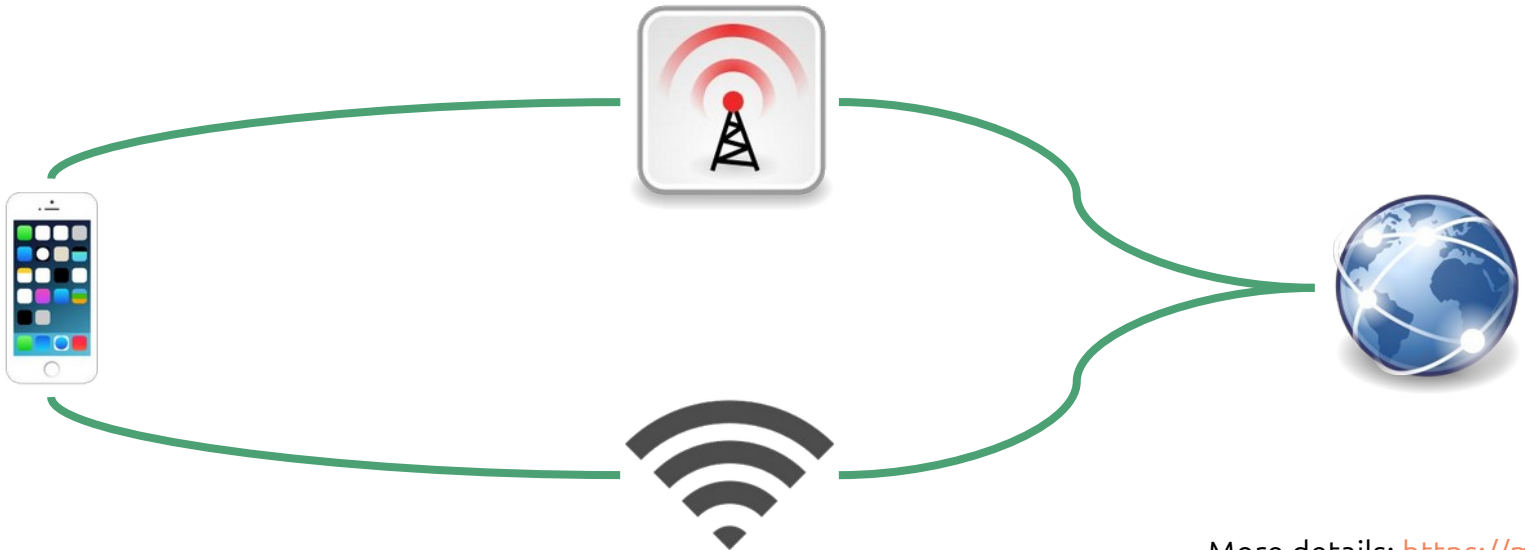
Matthieu Baerts (presenter) [1] [2]
Olivier Bonaventure [1]

[1]UCLouvain & WELRI, Belgium & [2]NGI0 Core

draft-baerts-tcpm-mptcpdss-00 – draft-baerts-tcpm-mptcpext-00

MPTCP (RFC 8684)

Exchange data for a single connection over different paths, simultaneously ⇒ Bring new capabilities



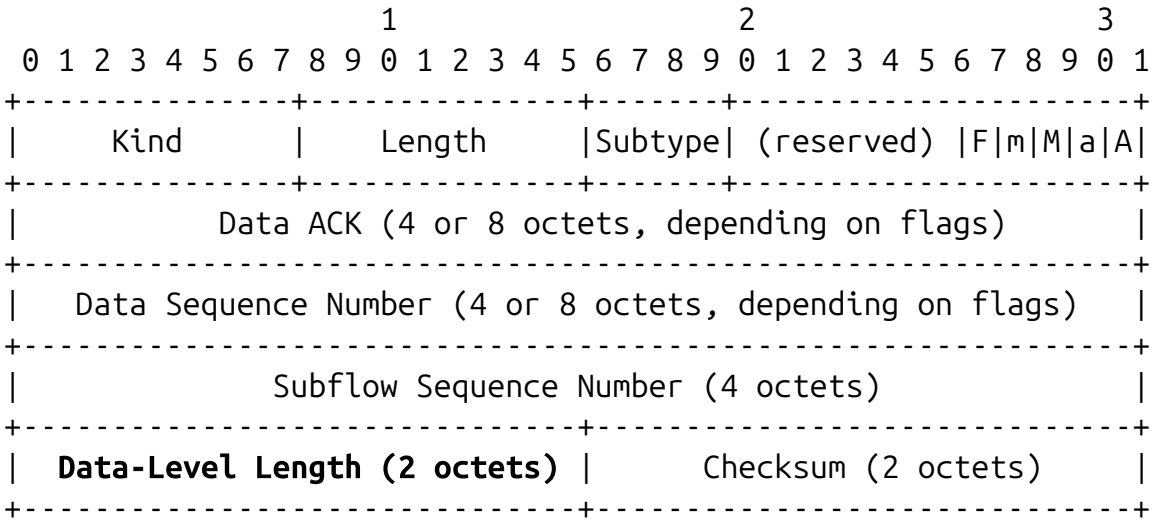
More details: https://mptcp.dev

MPTCP extensions

- Two: unrelated and independent
  - DSS extension
  - External keys
- Negotiated at the connection time
- Possible to fallback to “normal” MPTCP (RFC8684)

DSS extension

- **Goal**: allow packets of > 64 KB with MPTCP
- MPTCP with DATA: length limited to 64 KB



DSS extension

- **Why**: Linux can build egress packets > 64 KB since 2022.
- **Motivation**: Limitation with performance impact:

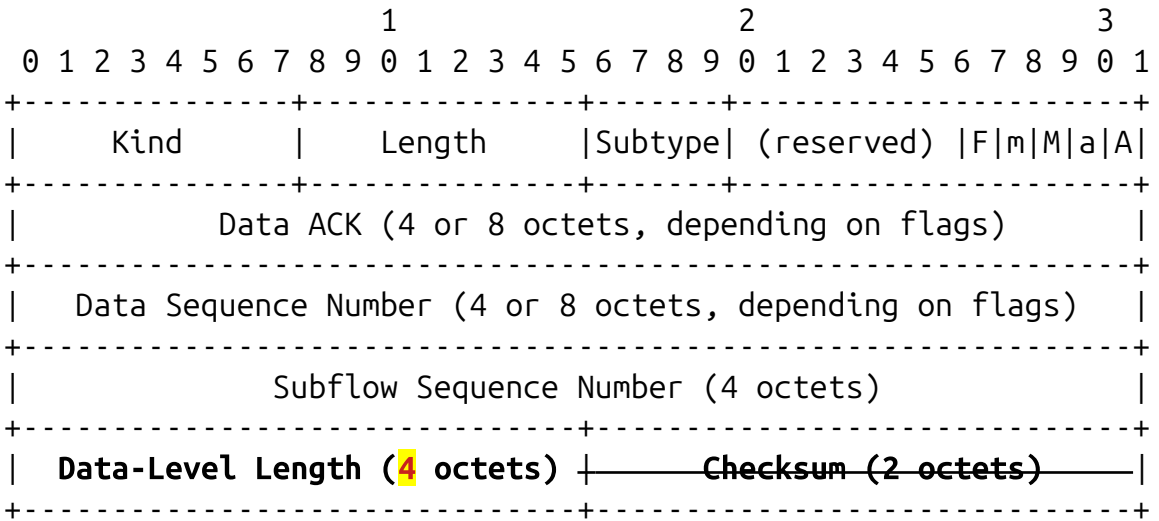
```
#define MPTCP_MAX_GSO_SIZE (…) /* ~64K */

if (ssk->sk_gso_max_size > MPTCP_MAX_GSO_SIZE)

        ssk->sk_gso_max_size = MPTCP_MAX_GSO_SIZE;
```

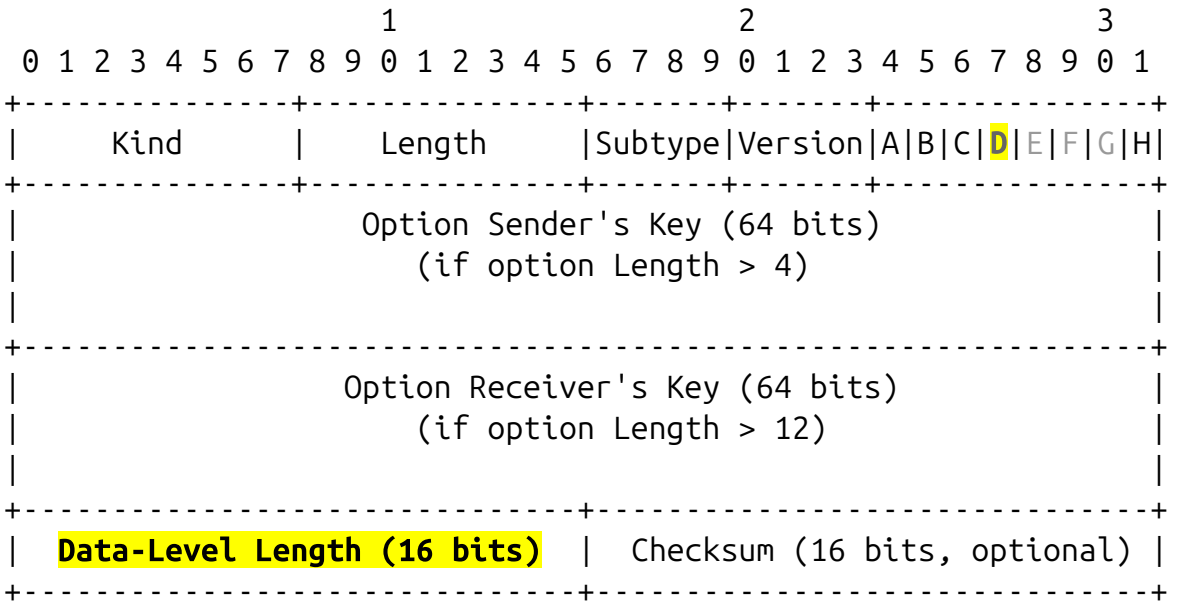
DSS extension

- **Solution**: extend DLL, drop Checksum (rarely used)



DSS extension

- **Negotiation**: using a reserved flag in SYNs (if OK)
- **Fallback**: keep 16 bits for the DLL in 3WHS



DSS extension

Questions?

External keys

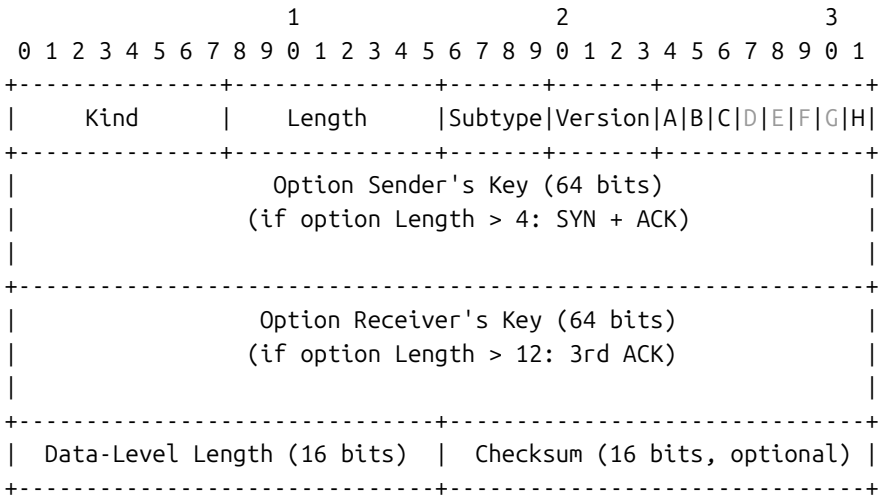
- **Motivation**: RFC 8684 sends the keys in clear
  - Attackers seeing initial handshakes can hijack MPTCP sessions
  - (Similar to TCP: such attackers could intercept / inject data)
  - Keys: used to establish new subflows, announce addresses, reset
- Linked to a previous draft: draft-paasch-mptcp-ssl

External keys

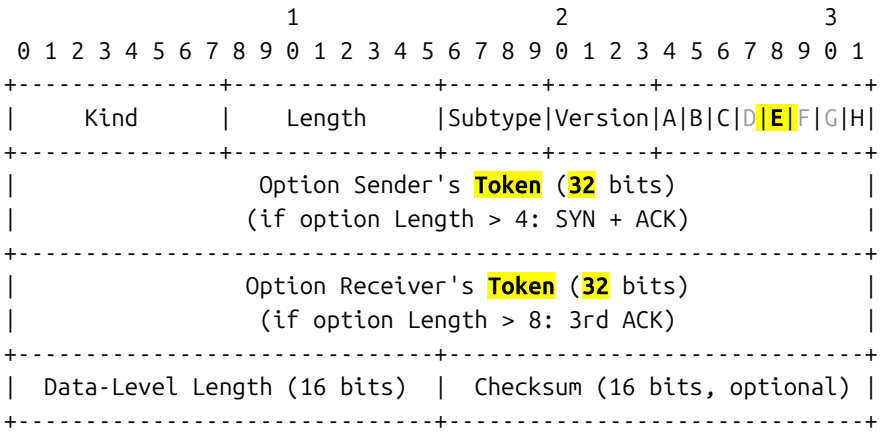
- Application-level protocols already negotiate keys: TLS, SSH
- **Solution**: Use these keys to secure MPTCP!
- Apps will only have to share the keys with the kernel.

External keys

- **Negotiation**: using a reserved flag in SYNs (if OK)
- **Fallback**: OK, the client sends a token in the 3rd ACK



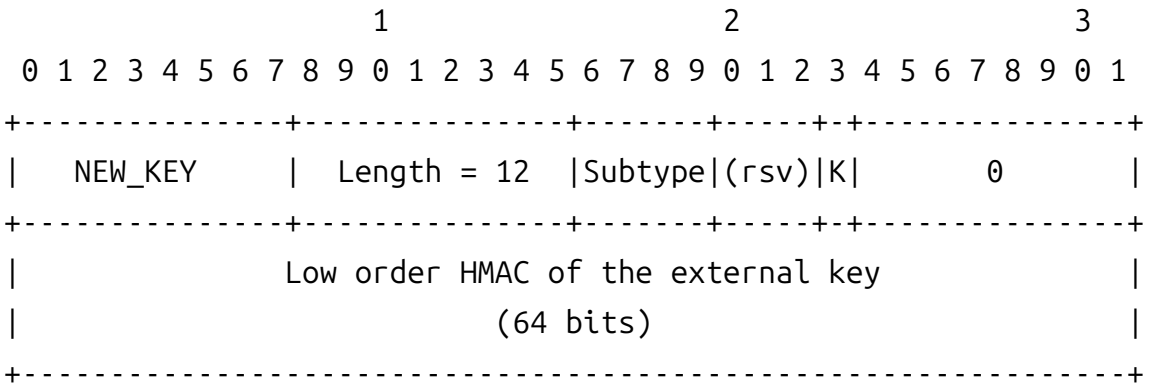
MP_CAPABLE - RFC 8684



MP_CAPABLE - draft-baerts-tcpm-mptcpext-00

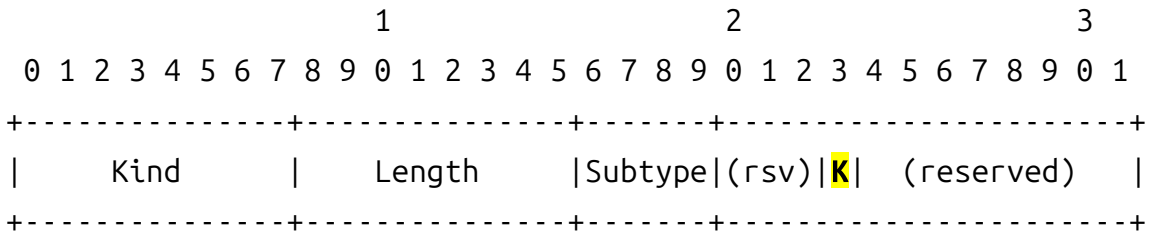
External keys

- Both sides need to send NEW_KEY to change key:
  - K is the key ID: max 2 keys: current and old/new one
  - Can be sent in an ACK, retransmitted if needed



External keys

- Signalling options need to use one of the reserved bits (K) for the key ID: `MP_JOIN`, `ADD_ADDR`, `MP_FASTCLOSE`



- `MP_FASTCLOSE` will no longer include the Receiver's Key, but a Truncated HMAC.

External keys

Questions?

Backup slides

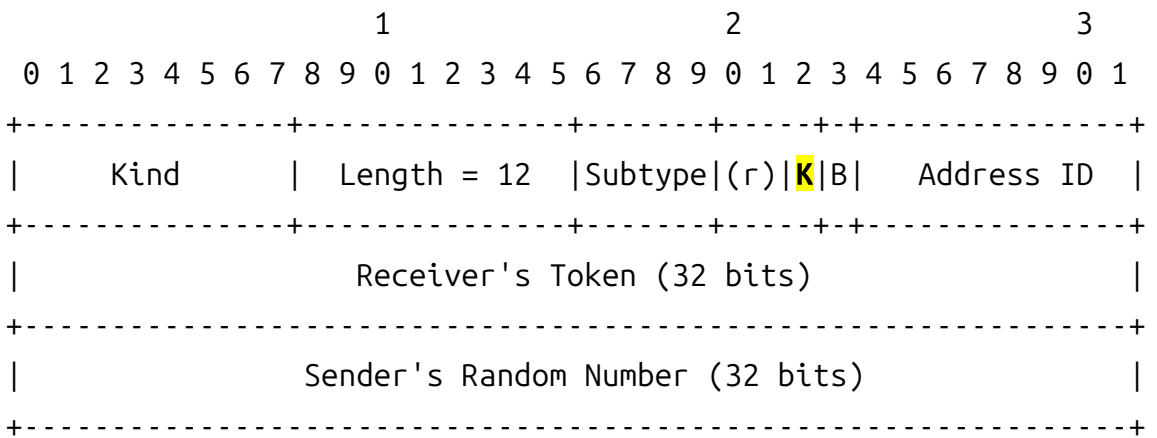
Flags in SYNs + MP_CAPABLE:

- RFC 8684 says:

  D through H: The remaining bits, labeled "D" through "H", are used for ***crypto algorithm negotiation****. In this specification,* ***only the rightmost bit, labeled "H", is assigned****. Bit "H" indicates the use of HMAC-SHA256 (as defined in Section 3.2). An implementation that only supports this method MUST set bit "H" to 1 and bits "D" through "G" to 0.*

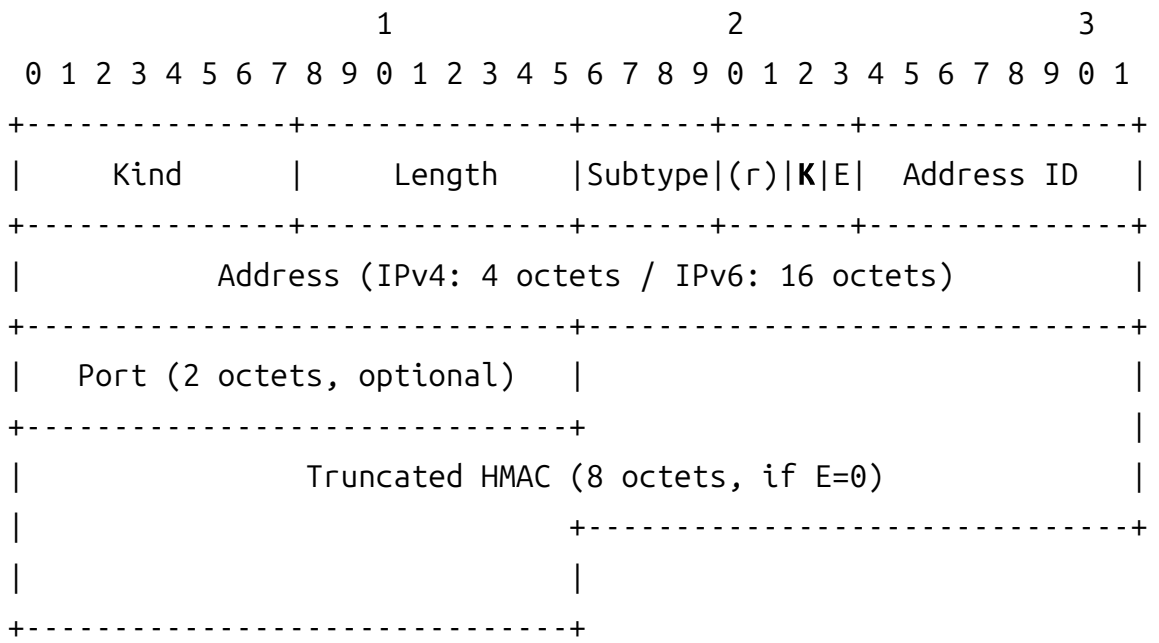
External keys

- New `MP_JOIN`



External keys

- New ADD_ADDR



External keys

- New MP_FASTCLOSE