MPTCP RobE
Robust session Establishment for MPTCP

draft-amend-mptcp-robe-00

IETF 105 Meeting, MPTCP, Montreal, July 2019
Markus Amend
RobE MOTIVATION SUMMARIZED

First introduced as idea at IETF99 relaxing MPTCP’s initial flow concept towards POTENTIAL initial flows, demanding:

IF THERE IS AT LEAST ONE FUNCTIONAL PATH, A CONNECTION MUST BE POSSIBLE

Detailed motivation can be found in IETF99 presentation: https://datatracker.ietf.org/meeting/99/materials/slides-99-mptcp-a-proposal-for-mptcp-robust-session-establishment-mptcp-robe-01
WORK SINCE IETF99

Paper published [https://dl.acm.org/citation.cfm?id=3232762](https://dl.acm.org/citation.cfm?id=3232762) elaborating the benefit of potential initial flows in terms of

- Robustness guaranteeing session establishment whenever one path is functional
- Handshake latency improvement compared to RFC6824 when initial flow has not the lowest latency
- Throughput gain by early availability of all subflows

Published draft document [https://tools.ietf.org/html/draft-amend-mptcp-robe-00](https://tools.ietf.org/html/draft-amend-mptcp-robe-00) defining two concepts:

- Simple RobE (RobE_SIM)
- Extended RobE (RobE_EXT)

Integrates and consider feedback from IETF 99

1. Separate robustness and latency/throughput improvement
2. No Key-A in RFC6824bis initial SYN/MP_CAPABLE for building relationship between several potential initial flows
3. Avoid computing overhead on sender and receiver side for increasing implementation probability (in Linux)
4. Ensure at least same authentication level of (sub)flows as MPTCP

LIFE IS FOR SHARING.
### MAIN CONCEPTS IN DRAFT DOC - RobE_SIM

1. Initiating MP setup sending MP_CAPABLE on all paths
2. Select initial flow based on fastest path return SYN/ACK
3. Reset other flow(s) and re-open using standard MP_JOIN

**Pro**
- Simple - Sender only implementation
- Full compliant to MPTCP v0/v1 standard
- Robust session setup for MPTCP v0
- Setup latency profits from fastest path

**Con**
- Wasting resources when resetting flow(s)
- Robustness for MPTCP v1 depends on initial flow final ACK

[*] Key-A in the first MP-capable is related to MPTCP v0 only and does not exist in MPTCP v1.
1. Initiating MP setup sending MP_CAPABLE on all paths
2. Select initial flow based on fastest path return SYN/ACK
3. Send new MP_JOIN_CAP option on other flows to immediately merge with initial flow

Pro
- Robust session setup for MPTCP v0 and v1
- Setup latency profits from fastest path
- Early availability of all subflows for highest throughput

Con
- Requires new MPTCP option MP_JOIN_CAP

[*] Key-A in the first MP-capable is related to MPTCP v0 only and does not exist in MPTCP v1.
### RobE_EXT – MP_JOIN_CAP option

<table>
<thead>
<tr>
<th>Kind</th>
<th>Length</th>
<th>Subtype</th>
<th>ADDR_ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sender's Key-A (64 bits)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HMAC (&gt;=96 bits)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Computational effort on receiver side

- **Key-A** ensures identification of related flows.
- **Key-B_fast_hash** enables MP session even when selected initial flow is not fully established yet (slight computational overhead).
- **HMAC** authenticates relationship of initial and potential initial flows.

Detailed explanation can be found on the mailinglist or in the draft document.
NEXT STEPS

Existing Linux prototype is outdated and does not comprise the changes since IETF99. Implementation from scratch is recommended. If someone e.g. from academia is interested, this could be a good thesis for a student.

RobE_EXT signalling is not yet fully specified:
- Might require 4-WHS like MP_JOIN
- Negotiation of RobE_EXT support and fallback mechanism needs to be discussed

Improve draft-00 content and integrate feedback from the mailinglist discussion and IETF105 meeting

Goal WG adoption; What is required therefore?
Thank you very much for your attention

If there are any questions, please feel free to ask.

Markus Amend
markus.amend@telekom.de

Further documents
Paper with detailed results: https://dl.acm.org/citation.cfm?id=3232762
Mailinglist: https://mailarchive.ietf.org/arch/browse/multipathtcp/?q=robe
Draft @Github: https://github.com/markusa/ietf-mptcp-robe

LIFE IS FOR SHARING.