

MPTCP Enhanced API

With simpler calls and smart connect

By Alexis Clarembeau



Content

We added a new layer over the current MPTCP API in order to make it:

- **Simpler:** Using and modifying paths can now be done without using complex structures (e.g: specifying hosts only by giving their name)
- **Smarter:** Based on previous connection information, we can automatically choose better subflow to connect first

Simpler API: Using string as host names

We now expose a new list of functions to use the MPTCP Api without the complex mptcp structures.

For instance, we defined the calls:

- `int mptcp_get_sub_list(sockfd, out);`
- `int mptcp_get_sub_tuple(sockfd, subid, ret_tuple);`
- `int mptcp_open_subflow(sockfd, f, host1, port1, host2, port2, prio);`

Which takes simpler arguments, such as a string for an host name, a int for a port, ...

Smarter API: Smart connect

But, when connecting a host, for instance, “multipath-tcp.org”, we can use:

- multiple source addresses:
 - 10.0.0.1 , 10.0.0.2 ,
- multiple destination addresses:
 - 213.251.128.146,213.251.188.146,2001:41d0:1:1992::1,2001:41d0:1:4a92::1

So, which one we need to try first?

Our idea: Decide this information based on previous connexion, using a cache

Current state, improvement

So now, the API is able to:

- Add, remove, change subflows **in a simpler way**
- Record information (RTT, bandwidth) in a cache on shutdown to **keep track of the best flows**
- When connecting to a host, **try first to connect on the best measured flow**

And in the future, we could:

- Extend the smart connection to **design a full smart path manager** which defines what subflow to open.