Multipath TCP Update

Philip Eardley, MPTCP WG Co-Chair

tsvarea

1st August, IETF-87, Berlin

Summary

- Brief introduction to Multipath TCP
- Status update on MPTCP implementations
  – draft-eardley-mptcp-implementations-survey
- Some examples of deployments and experiments

MPTCP experts, please feel free to chip in

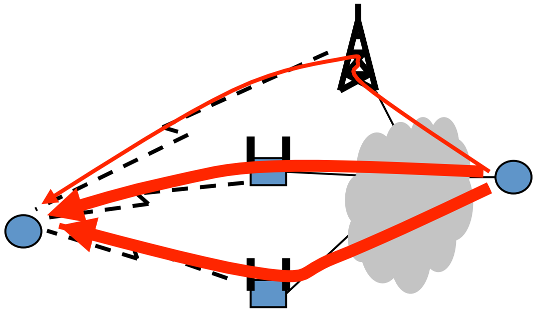
Multipath TCP – The basic idea

Enable a single TCP connection to use multiple paths simultaneously

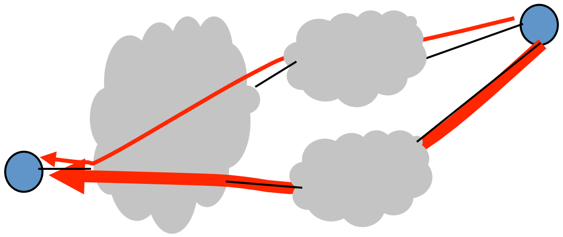
- Stop hiding multihoming
- Establish more than one path for the same connection (multiple addresses)
  - Use new TCP option for signalling
  - Paths may be used simultaneously (spread congestion in space)
  - Paths may be used sequentially ('handover')
- Looks like TCP…
  - to application (Support unmodified applications)
  - to network (Each TCP subflow is sent over a single path and appears like a regular TCP connection along this path)
  - fall back to TCP if necessary
- Olivier Bonaventure's MPTCP tutorial on Sunday (lots of refs)

Possible scenarios & benefits

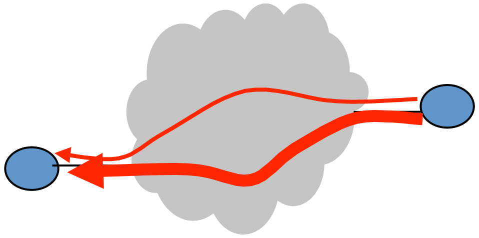
- A mobile node with 3G and WiFi
  - A form of mobility



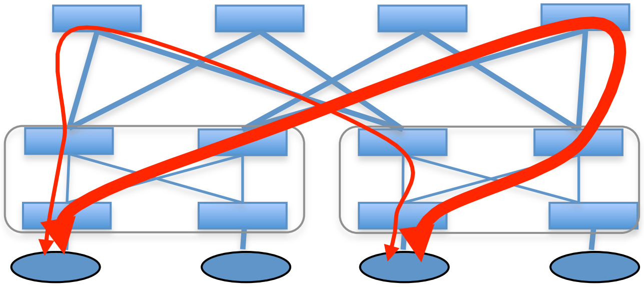
- A campus with 2 providers
  - Resilience



- Inside a network
  - Fast load balancing, TE on RTT
  - Increase utilisation, resource pooling



- Inside a data centre
  - Load balancing



- More info later

Status - Initial charter complete

- Signalling (RFC6824)
- Congestion (RFC6356)
- API (RFC6897)
- Architecture (RFC6182)
- Threats (RFC6181)
- Experimental or Informational
- Aim of current charter:
- Progress RFC6824 to Standards track

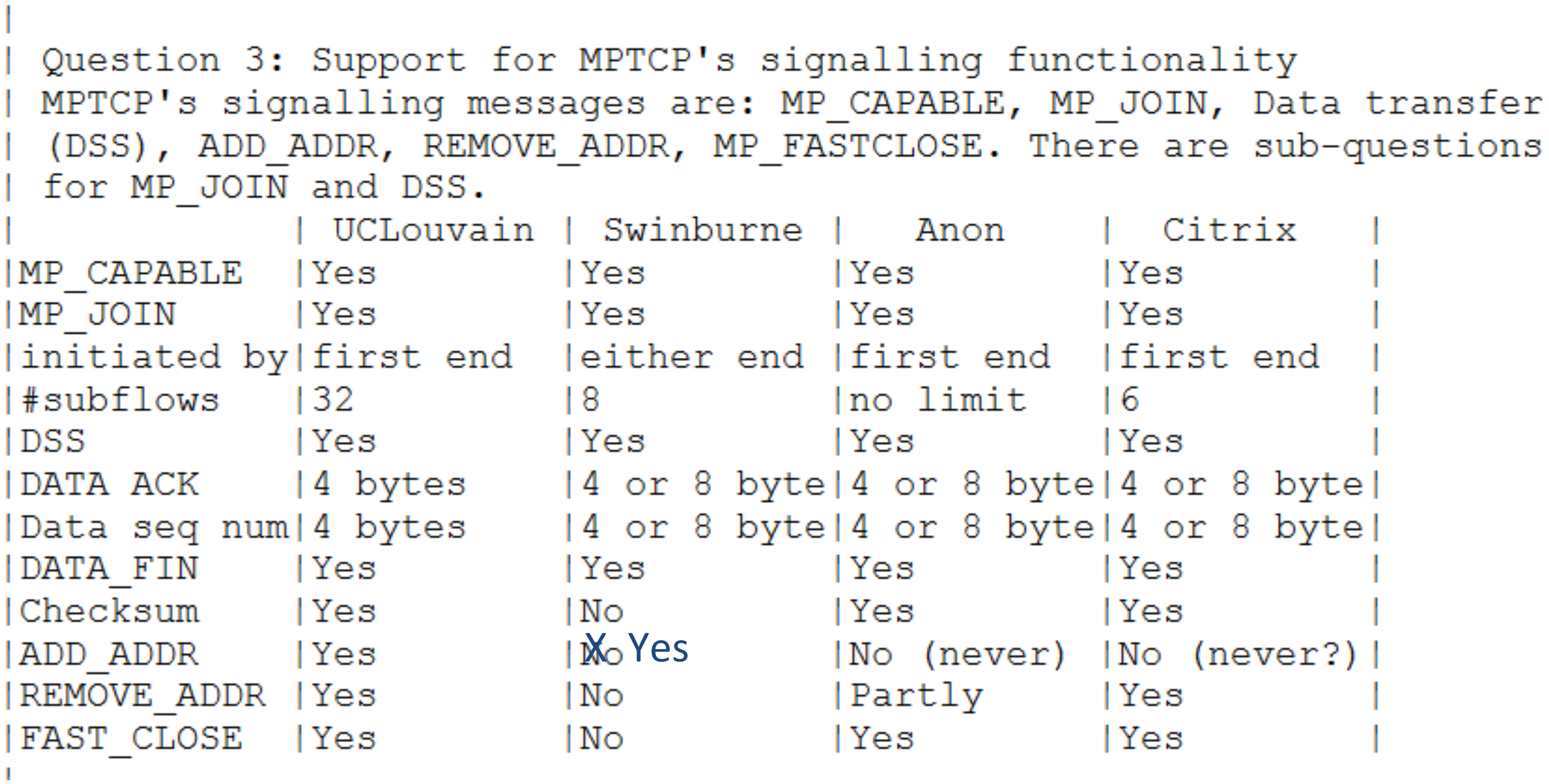
Status - Implementations

- We have 5 independent implementations!
  - – Linux, UCLouvain
  - – FreeBSD, Swinburne
  - – Commercial OS, Anon *
  - – NetScaler, Citrix
  - – User-space **
- RFC6824 is well implemented and understood
- Interoperate with Linux ‘reference’

* Not publicly available

** RFC compliant, but no longer maintained

Implementations survey
(1) signalling

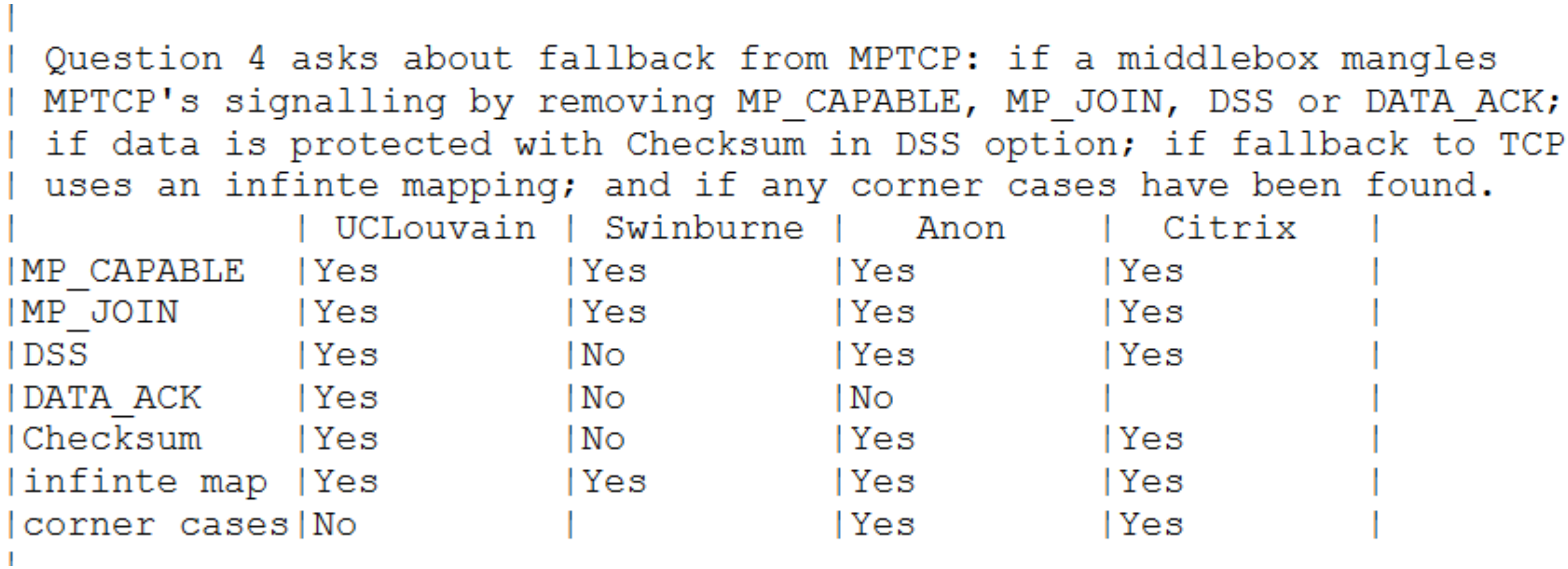


Question 3: Support for MPTCP's signalling functionality
MPTCP's signalling messages are: MP_CAPABLE, MP_JOIN, Data transfer (DSS), ADD_ADDR, REMOVE_ADDR, MP_FASTCLOSE. There are sub-questions for MP_JOIN and DSS.

	UCLouvain	Swinburne	Anon	Citrix
MP_CAPABLE	Yes	Yes	Yes	Yes
MP_JOIN	Yes	Yes	Yes	Yes
initiated by	first end	either end	first end	first end
#subflows	32	8	no limit	6
DSS	Yes	Yes	Yes	Yes
DATA ACK	4 bytes	4 or 8 byte	4 or 8 byte	4 or 8 byte
Data seq num	4 bytes	4 or 8 byte	4 or 8 byte	4 or 8 byte
DATA_FIN	Yes	Yes	Yes	Yes
Checksum	Yes	No	Yes	Yes
ADD_ADDR	Yes	~~No~~ Yes	No (never)	No (never?)
REMOVE_ADDR	Yes	No	Partly	Yes
FAST_CLOSE	Yes	No	Yes	Yes

- Signalling works well
- ADD_ADDR needs more discussion
- Details in draft-eardley-mptcp-implementations-survey

Implementations survey (2) fallback

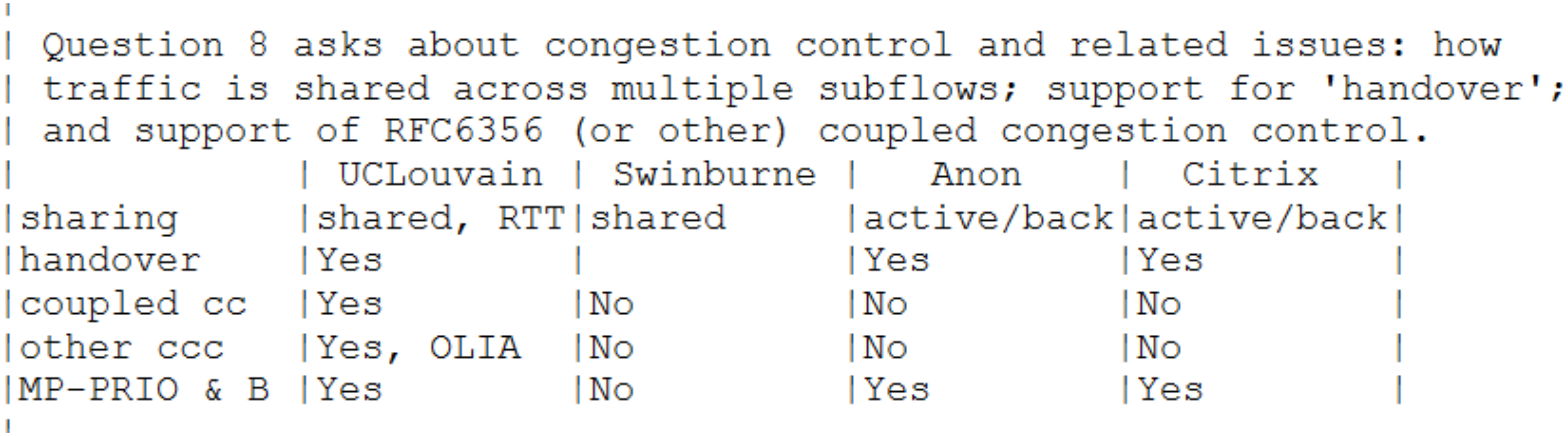


Question 4 asks about fallback from MPTCP: if a middlebox mangles MPTCP's signalling by removing MP_CAPABLE, MP_JOIN, DSS or DATA_ACK; if data is protected with Checksum in DSS option; if fallback to TCP uses an infinte mapping; and if any corner cases have been found.

	UCLouvain	Swinburne	Anon	Citrix
MP_CAPABLE	Yes	Yes	Yes	Yes
MP_JOIN	Yes	Yes	Yes	Yes
DSS	Yes	No	Yes	Yes
DATA_ACK	Yes	No	No	
Checksum	Yes	No	Yes	Yes
infinte map	Yes	Yes	Yes	Yes
corner cases	No		Yes	Yes

- Fall-back to TCP works well
- A few clarifications are needed
- Details in draft-eardley-mptcp-implementations-survey

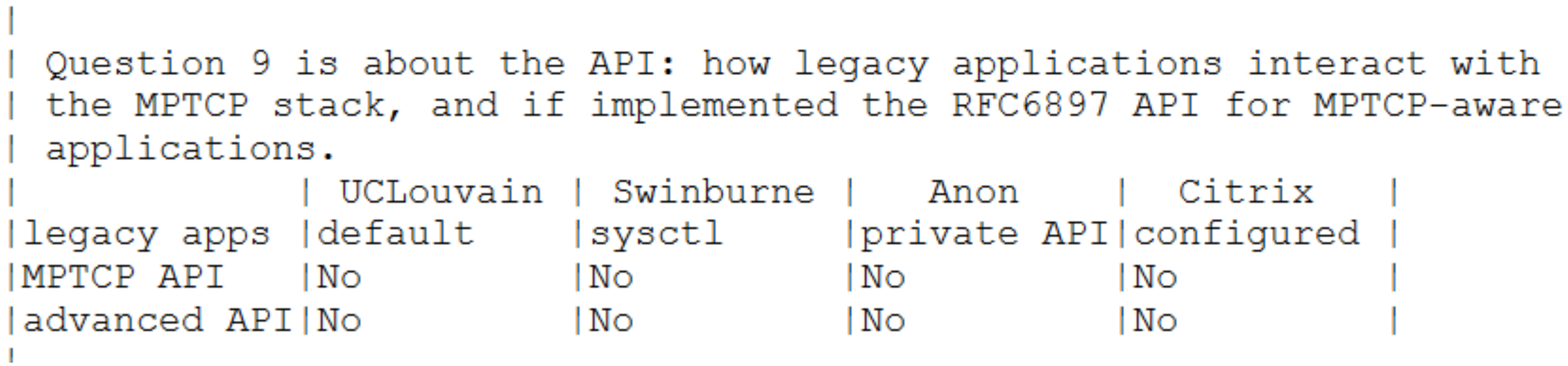
Implementations survey (3) congestion control



```
|
| Question 8 asks about congestion control and related issues: how
| traffic is shared across multiple subflows; support for 'handover';
| and support of RFC6356 (or other) coupled congestion control.
|            | UCLouvain | Swinburne |   Anon    |  Citrix   |
|sharing     |shared, RTT|shared     |active/back|active/back|
|handover    |Yes        |           |Yes        |Yes        |
|coupled cc  |Yes        |No         |No         |No         |
|other ccc   |Yes, OLIA  |No         |No         |No         |
|MP-PRIO & B |Yes        |No         |Yes        |Yes        |
|
```

- Use of mptcp for ‘active standby’
- OLIA is proposed improvement to RFC6356, draft-khalili-mptcp-congestion-control
- Several other multipath CC algorithms in the literature
- Details in draft-eardley-mptcp-implementations-survey

Implementations survey
(4) API



```
|
| Question 9 is about the API: how legacy applications interact with
| the MPTCP stack, and if implemented the RFC6897 API for MPTCP-aware
| applications.
|             | UCLouvain | Swinburne |    Anon     |  Citrix    |
|legacy apps |default    |sysctl      |private API|configured |
|MPTCP API   |No         |No          |No          |No          |
|advanced API|No         |No          |No          |No          |
|
```

- API not really been explored yet
- Details in draft-eardley-mptcp-implementations-survey

Next steps

- Moving RFC6824 to Standards track
  1. ADD_ADDR needs more discussion
  2. Fall-back needs a bit more clarification
  3. 'Better' security may be needed
     - Now: during initial handshake exchange keys in clear, then use keyed HMAC – do we try & do something 'better'
  4. More operational experience of different use cases, scenarios…
     - "particularly looking for cases where MPTCP could be detrimental in some way"
- Implementation advice (heuristics)
- MPTCP-aware middlebox (where at least one end host is MPTCP-enabled)
- Your help would be very welcome!

Use cases

- Some examples of how people are using MPTCP today

Commercial deployment of MPTCP

- First commercial deployment in 2012
- Initial target markets :- emergency services incident command units, mobile offices. Deployed in multiple EU countries
- MPTCP's benefits are speed and reliability
- Implementation:
  - Specialised hardware developed, 4 x UMTS radios combined with specialised amplifier.
  - MPTCP Louvain implementation
  - Dynamic use of tunnels and proxies used to aggregate traffic
- MPTCP core functionality works well. Efficiency ~85%
- Issues include
  - Middleboxes
  - Big queues (slow feedback) on mobile networks

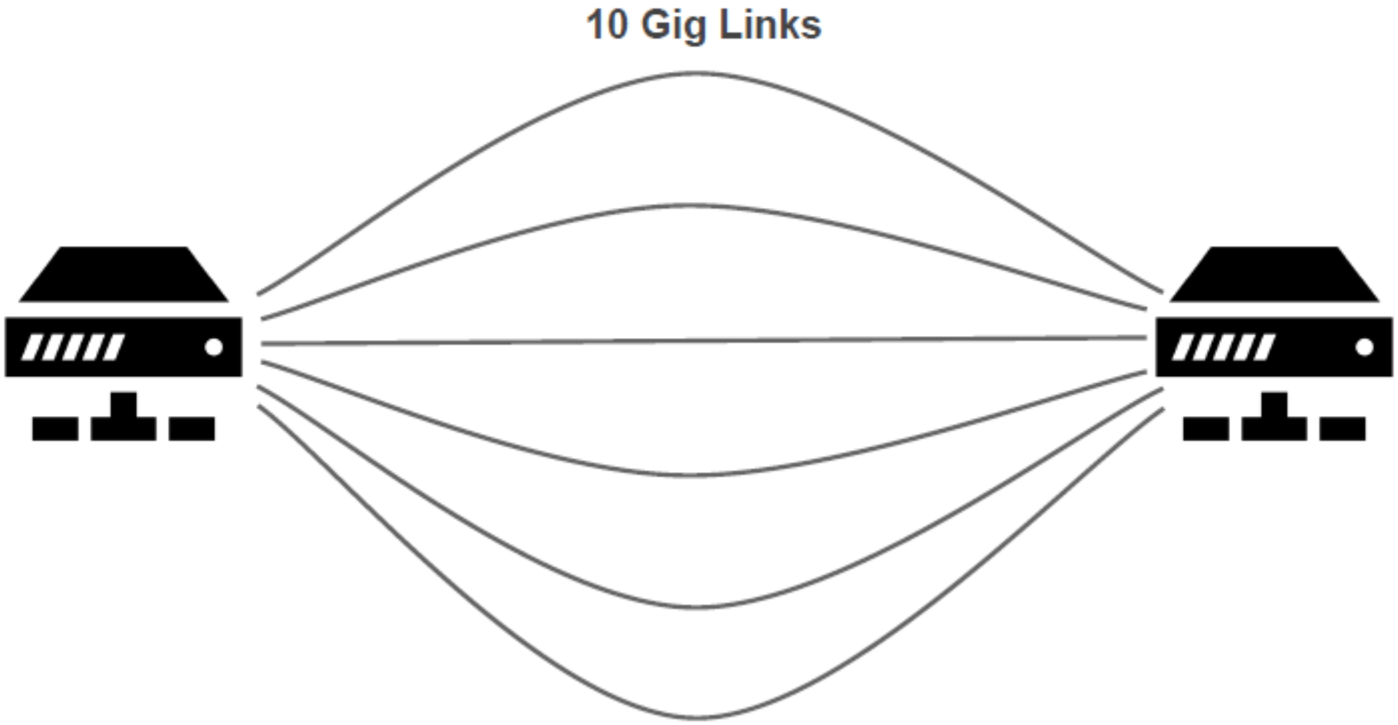
Multipath Networks

justin.collery@multipathnetworks.com

Mptcp interop in Berlin on Wed

- Christoph Paasch & Nigel Williams

52Gbit/s with MPTCP

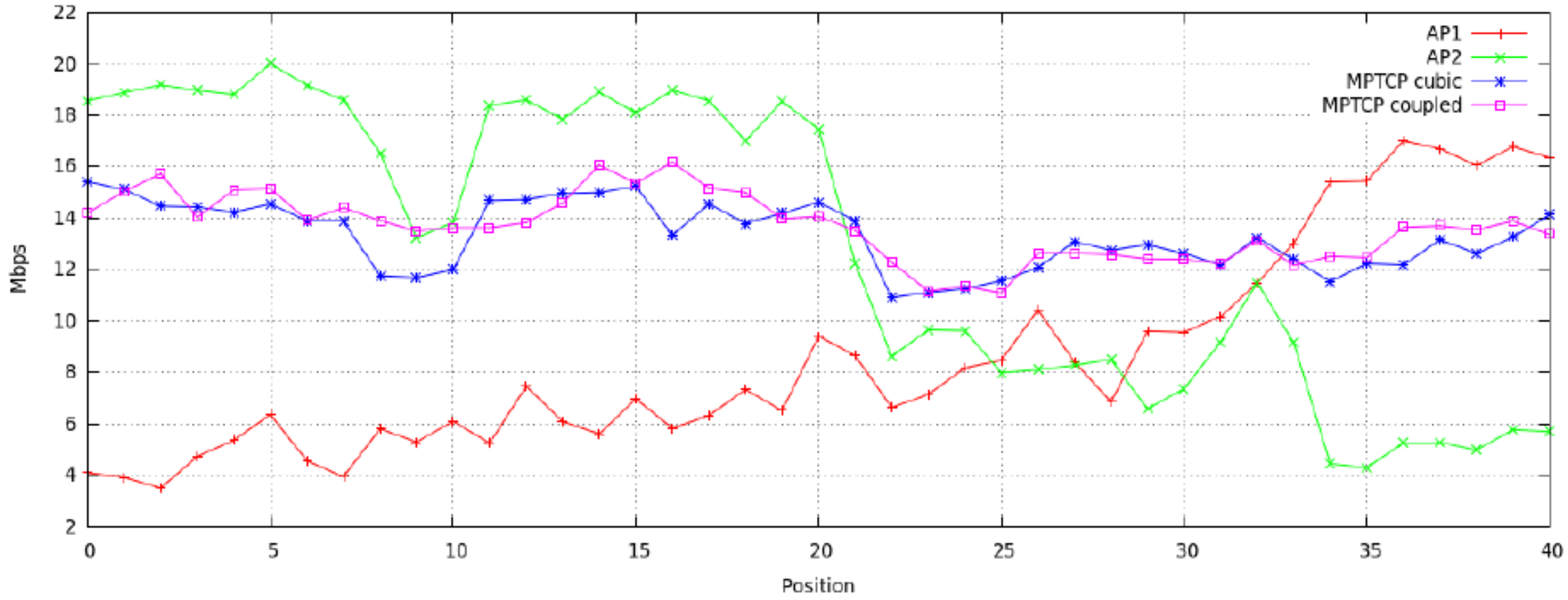


- See Christoph's talk in mptcp wg meeting

A Simple Mobile Scenario

Moving from one AP to another

AP1 AP2



- Note, mobile client has only single NIC
- See Costin’s slides in mptcp wg meeting

Use-case 4 Wide-area VM Migration

- Moving VMs across datacenters is useful
  - Unsolved problem: making sure TCP connections survive the migration
- MPTCP's connection identifier enables us to move the endpoint of a connection by just adding a new sublow with the new IP address.
- Xen + VM running Linux with MPTCP
  - Minimal changes needed to hypervisor
  - Works like a charm
- Live connection migration is also possible
  - Except moving processes is tougher...

- Ask Costin!

Summary

- We have 4 independent, maintained implementations of MPTCP
  - RFC6824 is well implemented and understood
  - Aim to capture implementation advice (heuristics) (but just ask)
- Our main aim is to move RFC6824 to Standards track - we'd like your help!
  - To agree what needs to be improved
  - To advise on how to do any improvements
- More operational experience of different use cases and deployments
  - Already MPTCP being used in a commercial deployment
- MPTCP & middleboxes
  - Getting middleboxes to be more MPTCP-friendly
  - Proxy scenario where at least one end host is MPTCP-enabled