draft-duchene-mptcp-load-balancing-01

- Added the *Application Layer Authentication* section

  ... decouples the token signalled in the TCP options from the key used in authentication allowing the token to carry arbitrary information.

- Next step: security considerations
Address Advertisement

- Several additions proposed in draft-duchene-mptcp-add-addr-00 discussed during IETF #96 and #97

- 2 integrated in RFC6824bis:
  - "NO JOIN" flag in MP_CAPABLE : “do not connect back to this address”
  - Echo” flag in ADD_ADDR : making ADD_ADDR reliable
Example

C->L : SYN+MP_CAPABLE

L->C SYN+ACK+MP_CAPABLE(nojoin)

L->C ADD_ADDR(5.6.7.9)

C->S2 SYN+MP_JOIN

S2->C SYN+ACK+MP_JOIN
Address advertisement reliability

C->L : SYN+MP_CAPABLE

L->C SYN+ACK+MP_CAPABLE(nojoin)  
L->C ADDR_ADDR(E=0, 5.6.7.9)  
L->C ADD_ADDR_ADDR(E=0, 5.6.7.9)  
L->C ADD_ADDR_ADDR(E=1, 5.6.7.9)
Making Multipath TCP friendlier to load balancers and anycast

Goal: **slightly** change Multipath TCP to be compatible with existing load balancers.

How:
- implementing the “NO JOIN” flag
- implementing the ADD_ADDR reliability
- designing a load balancing-specific path manager
Load Balancing Path Manager

General idea:

- Adding a public IP address to each server
- Advertise this IP address **reliably** to the client
- Restricting the initial subflow: putting the load balancer “off path”

the load balancer is only used to match a server and a client
Load Balancing Path Manager

WHEN A NEW CONNECTION IS ESTABLISHED:

    ip_addr = GENERATE_NEW_IP() /* Generate a specific IP address */
    ADVERTISE_TO_CLIENT(ip_addr) /* Advertise that IP to the client */
    SET_BACKUP_MODE(get_first_subflow()) /* Change the first subflow to backup mode */
Application: Layer-4 load balancer

- **NAT**: Client sends requests to the load balancer, which then sends them to the servers.
- **Direct Server Return**: Client sends requests directly to the servers.
- **Multipath TCP**: Client sends requests to multiple servers simultaneously for load balancing.
Application: Layer-4 load balancer

- Clients
- LVS
- Server 1
- Server 2
- Server 3

100Mbps link
1Gbps link
Layer-4 load balancer: results

- MPTCP isn’t significantly affected by the loss
- The latency affects only the connection establishment
- Transfer rates (10MB):
  - TCP : 16Mbits/sec
  - MPTCP : 803Mbits/sec
Application: Anycast
Application: Anycast

Clients

Router

Server 1
- Anycast addr.
- Public prefix.

Server 2
- Anycast addr.
- Public prefix.

Server 3
- Anycast addr.
- Public prefix.
Anycast: results

- ECMP pool of 3 servers
- Every 10 sec.: remove a server for 5 sec.
- 3 servers : 2800Mbit/s
- 2 servers : 1900Mbit/s
- Spikes in RST when a server is removed and when it’s re-added
Anycast: results

- No drop in BW
- No RST sent

Multipath TCP can be deployed to support anycast services.
Conclusions

Some simple changes allows MultiPath TCP to:

- work with **unmodified** layer-4 load balancers while:
  - improving performances
  - improving reliability
  - solving the bottleneck problem

- be deployed to support Anycast services

*new use case for MultiPath TCP!*

The complete results will be presented at ICNP 2017.