On The Joint Use of TCP and Network Coding

Emmanuel Lochin

ISAE-SUPAERO

IETF’100 November 2017

Some parts presented in these slides have been patented
NC layer placement

Applications

user-level

kernel-level

Operating System

hardware
NC layer placement

Applications

user-level

kernel-level

Operating System

hardware

Applications

Transport

Network

Link

011010110100101
NC layer placement

Applications
  user-level
  kernel-level
Operating System
  hardware

Applications
  UDP
  IP
  Link
  011010110100101

Considering UDP...
NC layer placement

NC layer can be placed everywhere with UDP.
NC layer placement

Applications
user-level
kernel-level
Operating System

Hardware

Applications

UDP
Network Coding Layer
IP
Link
01101011010100101

NC layer can be placed everywhere with UDP
NC layer placement

NC layer can be placed everywhere with UDP
NC layer placement

However with TCP...
NC layer placement

Applications
- user-level
- kernel-level

Operating System

TCP

IP

Link

011010110100101

NC with Cross Layer TCP

Applications

Operating System

Hardware

NC layer must be below the transport layer and must interact with TCP
TCP uses losses as a congestion signal to compute its sending rate. Using a NC coding scheme below the TCP layer masks losses.
NC layer placement

Explained diagram:

- Applications
  - user-level
  - kernel-level

Operating System

Hardware

- Applications
  - NC TCP

IP

Link

011010110100101

Or you must to implement a new NC transport layer
NC layer placement

- Applications
  - user-level
  - kernel-level
- Operating System
- hardware

Obviously no point to be above

TCP
IP
Link

01101011010100101

Network/Coding/Layer
One solution is to implement a new TCP stack ⇒ a coded transport layer eg. CodedTCP

Problems

- Need to replace TCP and implement the transport layer stack, or need a proxy-like system
- TCP must remain sender based ⇒ need to change TCP negotiation semantic with CodedTCP
- You can’t take benefit of various TCP flavours and TCP evolution
The question is how to enable a cross-layer scheme between a NC and a TCP layer?

Idea is to use ECN as a cross-layering scheme between NC and TCP

⇒ BTW this is ECN job

Basically ECN is used to pilot/control TCP sending rate when NC is used

ECN is broadly supported: Linux, *BSD, OSX, Windows

All these OS follow RFC3168
Main principle below the IP layer

You wish to strictly behave as TCP
Don’t want to be as bad as TCP in presence of random losses? You can monitor/pilot the signal with for instance a Loss Discrimination Algorithm, ...
Main principle above the IP layer

Application Layer
Transport Layer / TCP
NC encoder
IP Layer
Link / Phy
Tunnel

TCP onward path
TCP forward path

Application Layer
Transport Layer / TCP
NC decoder
IP Layer
Link / Phy
Tunnel

Mark TCP CE bit
Transmit packet to upper layer

Decoded packet

Rebuilt lost packet?
YES
NO

Same principle but above IP layer
IPERF experiment with link capacity 10Mb/s, RTT 40ms and $\approx 2\%$ uniform random losses

Estimation based on Mathis et al. formula \( \frac{MSS}{RTT \sqrt{p}} \) gives 2.06 Mbit/s
NC layer (TETRYS) below TCP all losses are masked
TCP/TENTET
TCP

NC layer with ECN crosslayer (TENTET) below TCP all losses are masked but signaled to TCP with ECN
You might also choose when to signal
Conclusion

- Using ECN with a receiver NC layer allows to simply interact with TCP
- No modification of existing TCP stacks
- Compliant with all TCP version that follows RFC3168