ConEx Concepts and Uses

draft-moncaster-conex-concepts-uses-01

Toby Moncaster
John Leslie (JLC)
Bob Briscoe (BT)
Rich Woundy (Comcast)
draft status

draft-moncaster-conex-concepts-uses-01

- Individual draft
- Intended charter milestone: use-cases
- Intended status: Informational
- Intended next step: WG item
Overview

- The problem
- Congestion marking (ECN)
- Congestion exposure
- Where do we stand?
- ConEx use cases
  - ConEx components
  - Traffic management
    - Managing the right traffic
    - Encouraging better congestion control
    - Targeted capacity provisioning
  - Other use cases
- Questions
- Next Steps
- Summary
The Problem

- The problem can be characterised in at least two ways:
  - Capacity Sharing – sharing limited resources between concurrent flows
  - Congestion Management – improving performance and delay for all

- Understanding congestion is definitely key
  - Too much traffic arriving too quickly = congestion

- Capacity sharing currently myopic:
  - In time (queues have no idea of past history of traffic)
  - In space (traffic may be causing problems elsewhere)

- Queues can only apply pressure by indicating congestion
  - Best signalled in forward direction (unlike Source Quench)
  - Requires honesty from receiver who wants the data as fast as possible
  - Needs sender to reduce rate, but it would rather send fast too

- Whole path congestion not visible at forwarding layer
  - Can't tell whether traffic is responsive to congestion
The Problem continued

- Capacity sharing suffers from a key problem – how to measure it
- Current approaches (rate and volume) are bad as they don’t reflect actual network conditions
- Congestion is a good measure of impact on other users
- Congestion-volume is a better metric to measure this
  - Congestion-volume = volume x congestion (units of bytes)
  - Congestion-Rate = rate x congestion (units of bps)
  - For a 1Mbps flow, 0.1% congestion = 125 bytes congestion-volume in 1 second
- Congestion-volume is measure of how much excess traffic was in network over any sampling interval (millisec, minute, month, ...)
- Congestion-volume can be measured per-packet, per-flow, per-user, per-network, ...
- ConEx means congestion-volume can be measured as easily as volume
Congestion Marking (ECN)

➢ Traditionally queues indicate congestion by dropping packets
  • Relies on stateful transport to spot gaps in data
  • Can lead to unwanted synchronisation effects

➢ RED improves this by dropping packets before queue overflows
  • Packets dropped probabilistically
  • Drop probability increases as the queue grows

➢ ECN builds on RED
  • ECN marks packets instead of dropping them
  • Sender still responds as if there were a drop
  • But no data is lost so less re-transmission

➢ ECN shows how much congestion traffic has already experienced

➢ But can’t see how much congestion traffic is going to encounter
Congestion Exposure

- Whole path congestion is hidden from network
  - Congestion is known to the end-systems (ECN marks or loss)
  - At any point, ECN reveals congestion so far
- What is needed is knowledge of congestion on rest of path
- ECN gives congestion experienced on every packet
- ConEx sender adds congestion expected for every packet
- ConEx enables packets to carry
  a) *Congestion experienced* (e.g. ECN markings)
  b) *Congestion expected* (total congestion sender expects the packet to see)
- subtracting a from b gives congestion on rest of path
- ConEx mechanism to be defined in later document
ConEx Design Requirements

- **Accuracy** – ConEx info should be as accurate as possible.
  - Congestion is measured in fractions of a percent
  - Source must be trusted to correctly declare the expected congestion
  - Destination must feed back accurate whole-path congestion

- **Timeliness** – ConEx info needs to be as recent as possible
  - Design of network imposes min 1RTT delay
  - Transport protocol should seek to minimise delays
  - Feedback needs to be fast enough to prevent info going “stale”

- **Visibility** – ConEx should be visible at every node on the path
  - ConEx must be visible in IP layer
  - ConEx markings need to survive tunneling, middleboxes, firewalls, etc
Where Do We Stand?

- Long process leading up to chartering
- ConEx chartered in June 2010 with limited scope
- Concentrates on one usage scenario:
  - end hosts and receiving network are ConEx enabled (other networks might not be enabled)
  - note difference between Use Case and Usage Scenario
- Can consider other use cases:
  - "Experiments on use cases are encouraged and the WG will solicit feedback from such deployments."
- This draft covers Milestone 1 “Use Cases Description” (info)
- Several use cases explored. Some go beyond charter, but demonstrate how powerful ConEx can be
Lots of use cases for ConEx

Charter focuses on use cases for following scenario:

Green elements ConEx-Enabled. Grey elements not Enabled

NB: the symmetry of most networks implies that ISP Z can be a ConEx-Enabled source network for any traffic that Dest sends into the network
Two new network components defined:

- **ConEx Monitor** – uses ConEx to measure/report congestion-volume
- **ConEx Policer** – uses ConEx to actively control traffic (delay, expedite or drop)

Policers and Monitors can be at Ingress, Egress or Border:

- Border can do policer or monitor functions
  - policing can mitigate serious congestion
  - Monitoring can see (and deter) congestion
Traffic Management

- ISPs often perform traffic management:
  - Aim is to give majority of users an adequate service at peak times
  - Users targeted based on application, traffic rate, volume transferred, etc

- ConEx policers offer an alternative:
  - Each sender is declaring the congestion they expect to cause
  - This can be used to control the impact they have on others

- ConEx Egress policer identifies users with most congestion-volume.
  - Prioritise traffic depending on congestion it has declared
  - Penalise traffic that has caused excessive congestion
Managing the Right Traffic

- Lots of debate about traffic management
  - Current approaches tend to be relatively unfocused
  - Assumptions made about when "peak time" happens
  - Often targets specific applications - big problem for Net Neutrality camp

- ConEx approach is better
  - Only targets traffic that contributes most to congestion
  - Because it monitors actual congestion it always knows when peak time is
  - Wholly application-agnostic – only cares about impact of traffic on the network

- Overall this is better for ISP and its users
  - Less damaging to customer relationships
  - Allows some bandwidth differentiation without QoS in the net
  - No need for expensive flow-aware kit in backhaul or access
Encouraging Better CC

- Lots of current work looking at better congestion control
- LEDBAT introduced idea of highly reactive congestion control
  - Designed for bulk data transfers which don’t care about instantaneous rate
  - Backs off as soon as it detects queue building - reacts to congestion before other transports need to
- MulTCP and related work introduced weighted congestion control
  - Application chooses how much to react to congestion by assigning a weight
  - High priority apps don’t back off much, low priority back off more
  - Logical extension is fully weighted congestion control
Encouraging Better CC continued

- Current traffic management disincentivises use of LEDBAT
  - LEDBAT still transfers high volumes, so is still targeted
  - LEDBAT used for applications like P2P, so is still targeted
  - LEDBAT can still reach high data rates, so is still targeted

- ConEx encourages LEDBAT-like transports
  - ConEx based traffic management brings correct incentives
  - Traffic is controlled based on congestion it causes
  - LEDBAT causes less congestion so gets less control

- ConEx encourages use of more adaptable congestion controls
  - Applications choose how reactive they want to be
  - Interactive applications can react less to maintain their quality
  - Background applications can back off more and recover at quieter times
  - All that matters is overall Congestion-volume...
Targeted Capacity Provisioning

- Better traffic management means:
  - Users stop causing unnecessary congestion
  - Protocol designers avoid unnecessary congestion
- So any congestion remaining reflects real demand
- Congestion-volume can be used to measure this demand
  - Can measure at each physical interface
  - Can measure over investment timescales
  - Can identify precise capacity demand
- Without ConEx you can’t tell if demand is real
  - Investments may be “wasted”
  - Users may not see real benefit
- More on this in next revision...
Other Use Cases

- Charter focused on ConEx-enabled destination network
  - CDN distributing e.g. Movies; User watching VoD;
- Can add ingress policing for traffic heading in other direction
  - End user transferring P2P; Live video chat with remote user via relay server;
- 3 other use cases already discussed in draft:
  - ConEx for DDoS mitigation – network can identify and track excess congestion and block it before it causes problems. This could be a big incentive to deploy
  - ConEx “QoS” (builds on weighted CC) – user can prioritise traffic with no network involvement. Makes sense with ingress policing.
  - Congestion accounting: works best with full deployment. But even simple deployment at sender allows operators to monitor congestion-causing traffic
- Other use cases discussed on mailing list. Intend to add more use cases to draft
Questions

- Did we pick a reasonable set of use cases?
- Should we add a non-commercial use case like campus, corporate, etc?
Next Steps

➢ Believe this is ready for adoption as first WG draft
➢ Lots of work already done
➢ Discussions on and off list
  • Need to tweak layout
  • Might add more use cases from those suggested on mailing list
  • Expand “Other Issues” section
➢ Big question: How can we summarise ConEx?
  • A way to reduce overall congestion?
  • A metric to improve capacity sharing?
  • A metric to allow better traffic management?
  • All the above and more?
Conclusions

- This draft describes some of the use cases for ConEx
- By no means exhaustive – this is a radical idea that will generate some truly innovative uses
- Included a brief description of a possible mechanism as readers need that to understand the use cases
- Congestion-volume is the key metric for controlling capacity sharing
- Introduced the ConEx Monitor and the ConEx Policer
- Highlighted several use cases, concentrated on one in particular
ConEx Concepts and Uses

spare slides
ConEx verifier

- So far have presented ConEx in “naive” manner
  - Assumes sender is reasonably honest
  - Assumes no-one wants to subvert ConEx info

- ConEx verifier can check this
  - Uses moving average to ensure Congestion-experienced ≈ Congestion-expected for given flow
  - Can penalise flows that have marked imbalance over time
mediating between modern cc’s

- The world used to be a simpler place:
  - Traffic was TCP or UDP
  - End-systems followed same basic rules
  - Most traffic simple bulk data

- Things are much more complicated now:
  - Lots of different congestion controllers (CUBIC, Compound, etc)
  - Traffic mix much more complex now (streaming video, interactive chat, etc)

- ConEx allows for any congestion controller imaginable
  - Only thing that matters is overall contribution to congestion-volume
  - User (or their apps) free to make their own choices
Raising the DDoS Bar

- DDoS is a serious problem – currently no robust solution
  - ConEx Border Policers can help raise the bar
    - ConEx Policers limit traffic rate towards congestion hot-spots
    - Policers can rate-limit non-ConEx traffic routing towards same hot-spot
  - ConEx Border Monitors can help raise the bar too
    - DDoS traffic shows ultra-high congestion, so shows up at border

- DDoS protection grows as ConEx deployment increases
- Details are important but way beyond scope of use cases document