Re-ECN:
Adding Accountability for Causing Congestion to TCP/IP

draft-briscoe-tsvwg-re-ecn-tcp

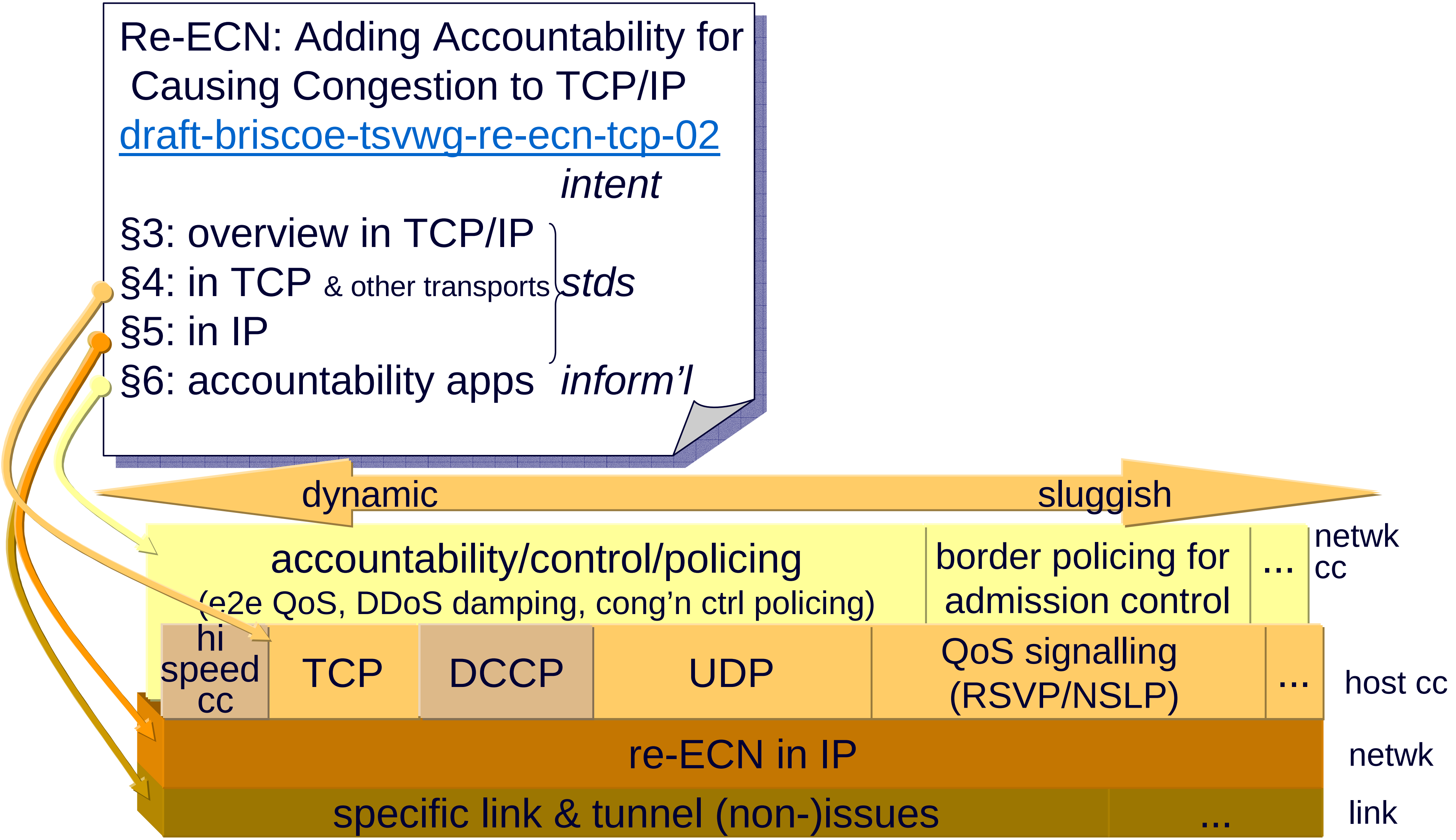
Bob Briscoe, BT & UCL
Arnaud Jacquet, Alessandro Salvatori & Martin Koyabe, BT

IETF-66 tsvwg Jul 2006

updated draft 02

- Re-ECN: Adding Accountability for Causing Congestion to TCP/IP
  - **updated draft:** draft-briscoe-tsvwg-re-ecn-tcp-02.txt
  - **ultimate intent:** standards track
  - **immediate intent:** re-ECN worth using last reserved bit in IP v4?
  - intended to split off apps section into draft-briscoe-tsvwg-re-ecn-apps, but didn’t
  - intent of previous draft 01 (IETF-66 Dallas Mar 06)**:**
    - – hold ECN nonce (RFC3540) at experimental
    - – get you excited enough to read it, and break it
- events since previous draft 01
  - since Mar 06, you’ve broken it (again)
    - – off-list: Salvatori (co-author), Bauer, Handley, Greenhalgh, Babiarz
    - – we’ve fixed it (changes to policing algorithms, not protocol)
  - you wanted to see IPv6 protocol encoding
    - – included in updated draft to assess necessity of IPv4 header change
  - revisions to draft (after recap slides)

recap doc roadmap



Re-ECN: Adding Accountability for Causing Congestion to TCP/IP

draft-briscoe-tsvwg-re-ecn-tcp-02

intent

§3: overview in TCP/IP

§4: in TCP & other transports

§5: in IP

§6: accountability apps

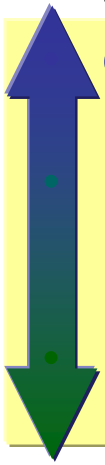
stds

inform'l

dynamic
sluggish
accountability/control/policing
(e2e QoS, DDoS damping, cong'n ctrl policing)
border policing for admission control
...
netwk cc
hi speed cc
TCP
DCCP
UDP
QoS signalling (RSVP/NSLP)
...
host cc
re-ECN in IP
netwk
specific link & tunnel (non-)issues
...
link

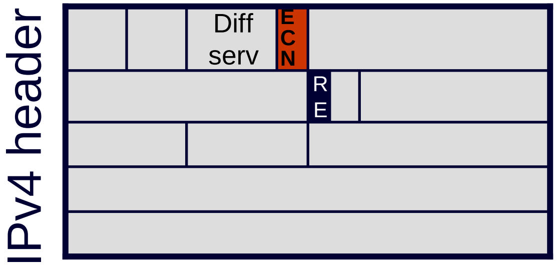
re-ECN recap: solution statement (§1)

- allows *some* networks to police congestion control at network layer



- conservative networks
    - might want to throttle if unresponsive to congestion (VoIP, video, DDoS)
- middle ground
    - might want to cap congestion caused per user (e.g. 24x7 heavy p2p sources, DDoS)
    - evolution of hi-speed/different congestion control
- liberal networks
    - open access, no restrictions
- many believe Internet is broken
    - not IETF role to pre-judge which is right answer to these socio-economic issues
    - Internet needs all these answers – balance to be determined by natural selection
    - ‘do-nothing’ doesn’t maintain liberal status quo, we just get more walls
- re-ECN goals
    - just enough support for conservative policies without breaking ‘net neutrality’
    - allow evolution of new congestion control, even for flows from liberal → conservative
    - nets that allow their users to cause congestion in other nets can be held accountable

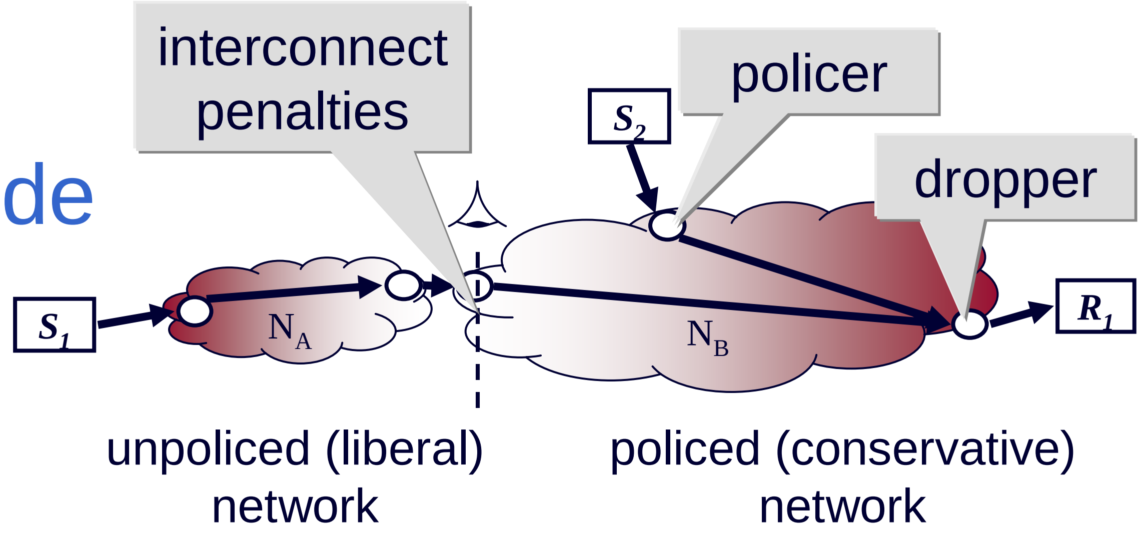
re-ECN in 1 slide

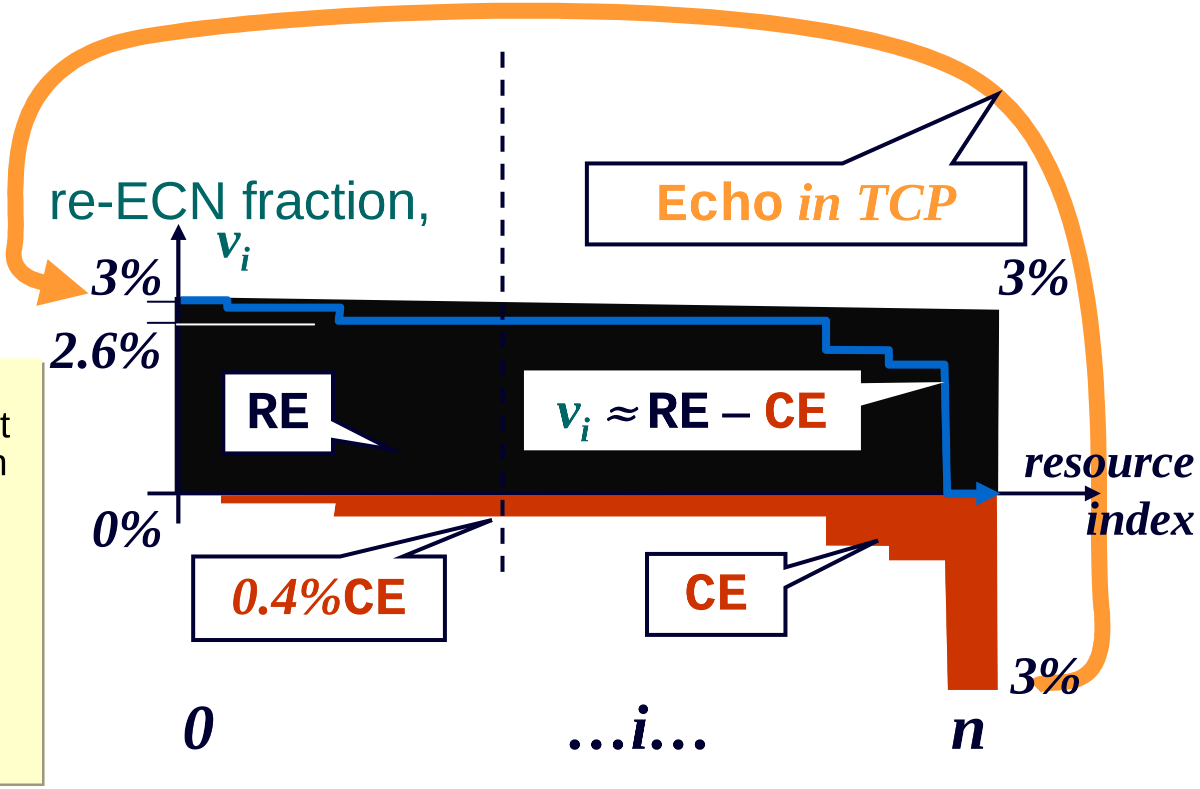


RE \ ECN	ECT(1) 01	CE 11
0	+1	0
1	0	-1

worth

- sender aims to balance every congestion experienced (**CE**) event by blanking new re-ECN extension (**RE**) flag in IP hdr
- at any point on path, diff betw fractions of **RE** & **CE** is downstream congestion
- drop persistently negative flows
- ECN routers unchanged





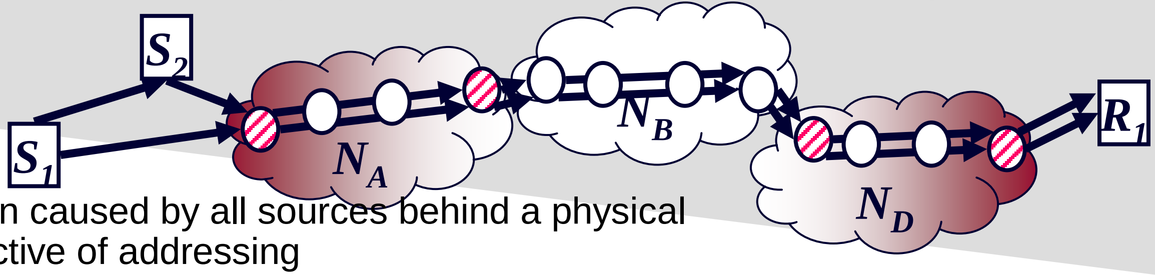
changes from draft 01 to 02

- listed (temporarily) at start of draft
  - added evolvability arguments against bottleneck policing (§6.1.2)
  - added (non-)issues with tunnels (§5.6),
    IPSec encryption and layered congestion notification (§5.7)
  - added IPv6 re-ECN protocol encoding (§5.2)
  - added reasoning for earlier change from 3 to 4 codepoints (§B)
  - new attacks and modified algorithm defences (§6.1.6 & §6.1.7)
  - minor editorial changes throughout
- HTML coloured diffs via
  - <www.cs.ucl.ac.uk/staff/B.Briscoe/pubs.html#retcp>

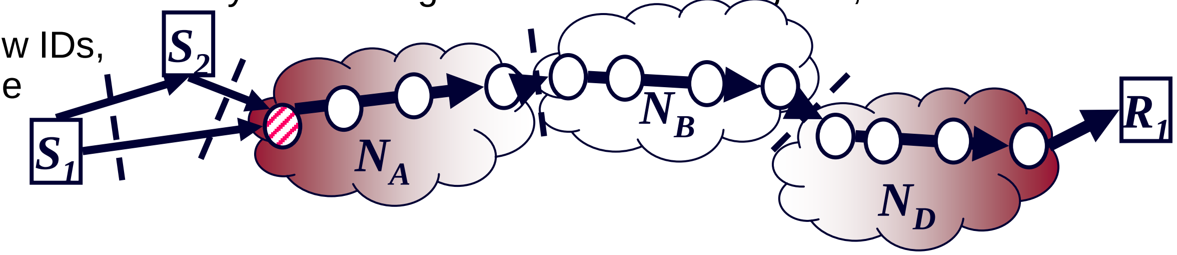
bottleneck policing harmful to evolvability

…and bypass-able anyway

- bottleneck policers: active research area since 1999
  - detect misbehaving flows causing ‘unfair’ share of congestion
  - located at each potentially congested routers
  - what right have these policers to assume a specific congestion response for a flow?
    - if they could police accurately, new congestion control evolution would require per-flow authorisation from all policers on the path (cf. IntServ)
  - malicious sources can bypass them by splitting flow IDs
    - even splitting flow across multiple intermediate hosts (or src address spoofing)



- re-ECN policing
  - polices congestion caused by all sources behind a physical interface, irrespective of addressing
  - within that, can also choose to police per-flow, per flow setup, per-destination etc.
  - evolution of new behaviours by bilateral agreement with first ingress, if at all
  - dropper uses flow IDs, but no advantage to split IDs



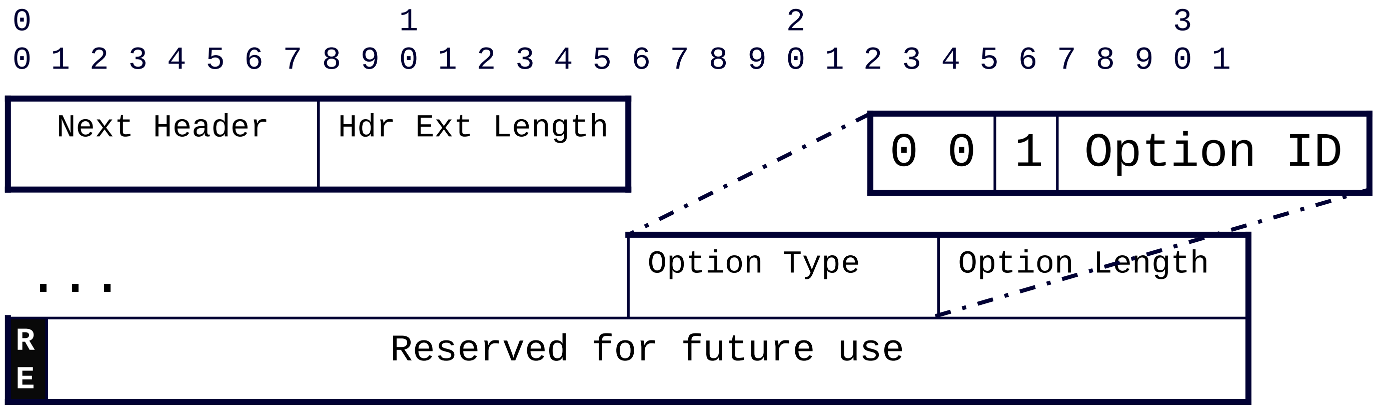
(non-)issues with layering & tunnels

- general non-issue
  - **RE** flag shouldn't change once set by sender (or proxy)
  - policers merely read **RE** to compare with **CE** introduced so far
  - OK as long as **CE** represents congestion since same origin that set **RE**
- IP in IP tunnels
  - OK if tunnel entry copies **RE** and **CE** to outer header
  - but full functionality RFC3168 ECN tunnel resets **CE** in outer header
    - no reason given in RFC3168 – arbitrary decision?
- IP payload encryption (e.g. IPSec ESP)
  - non-issue – re-ECN designed to work only in network layer header
  - flow-ID obfuscation also non-issue – re-ECN only uses flow ID uniqueness, if at all
- layer 2 congestion notification (ATM, Frame, ... MPLS, 802.3ar)
  - non-issue given IP layer should accumulate **CE** from each 'L2 network' into ECN

- considering guideline I-D on layered congestion notification

IPv6 re-ECN protocol encoding

- IPv6 hop-by-hop options header extension
  - new Congestion hop-by-hop option type



```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
```

Next Header | Hdr Ext Length

0 0 | 1 | Option ID

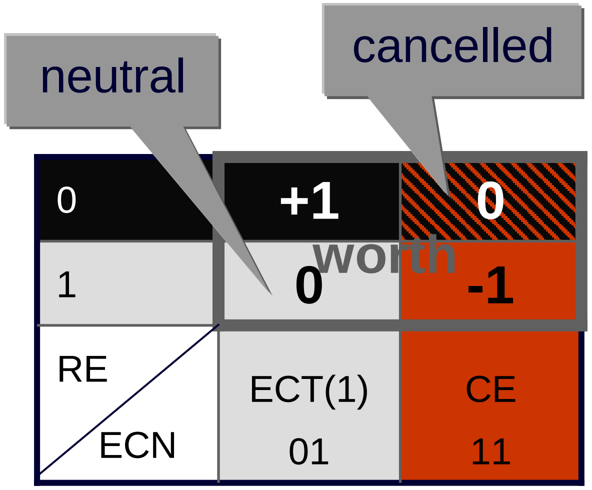
...

Option Type | Option Length

RE | Reserved for future use

  - action if unrecognized (AIU) = 00 ‘skip and continue’
  - changeable (C) flag = 1 ‘may change en route’
    - even tho RE flag shouldn’t change en route (AH would just tell attackers which packets not to attack)
- seems wasteful for 1 bit, but we plan:
  - future hi-speed congestion control I-D using multi-bit congestion field
  - other congestion-related fields possible
    - e.g. to distinguish wireless loss and per-packet vs per-bit congestion

attacks on re-ECN & fixes

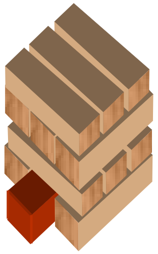


RE ECN	ECT(1) 01	CE 11
0	+1	0
1	0	-1

- recap: why two codepoints worth 0?
  - when no congestion send neutral (0)
  - packet marked ‘cancelled’ if network happens to mark a packet (-1) which the sender used to re-echo congestion (+1); +1 – 1 = 0
  - in draft 00, congestion marking of +1 packet turned it to -1 not 0, but networks could cheat by focusing marking on +1 (see §B)
- but now can’t attacker just send cancelled packets?
  - immune from congestion marking
  - simple fix: policer counts cancelled with +1 towards *path* congestion
    - should have specified this anyway, as both represent path congestion
    - also check proportion of cancelled to +1 packets same as -1 to neutral
- set of attacks using persistently negative dummy traffic flows
  - see next presentation for border policing fix
- one remaining known vulnerability if attacker can spoof another flow ID
  - known since early on – plan to focus effort on fixing this next

summary

- optional ‘net neutral’ policing of causes of congestion
  - liberal networks can choose not to police, but still accountable
- simple architectural fix
  - generic accountability hook per datagram
  - requires one bit in IPv4 header
  - or IPv6 hop-by-hop option – more wasteful but plan to use space



- bottleneck policing considered harmful (& ineffective)
- fixed re-ECN vulnerabilities while keeping simplicity
- changing IPv4 header isn’t a task taken on lightly
  - now it’s matured, we plan to discuss in network area too

Re-ECN:
Adding Accountability for Causing Congestion to TCP/IP

draft-briscoe-tsvwg-re-ecn-tcp-02

Q&A

Emulating Border Flow Policing using Re-ECN on Bulk Data

draft-briscoe-tsvwg-re-ecn-border-cheat

Bob Briscoe, BT & UCL

IETF-66 tsvwg Jul 2006

simple solution to a hard problem?

- Emulating Border Flow Policing using Re-ECN on Bulk Data
  - **updated draft:** draft-briscoe-tsvwg-re-ecn-border-cheat-01
  - **ultimate intent:** informational
  - **exec summary:** claim we can now scale flow reservations to any size internetwork *and* prevent cheating

recap doc roadmap

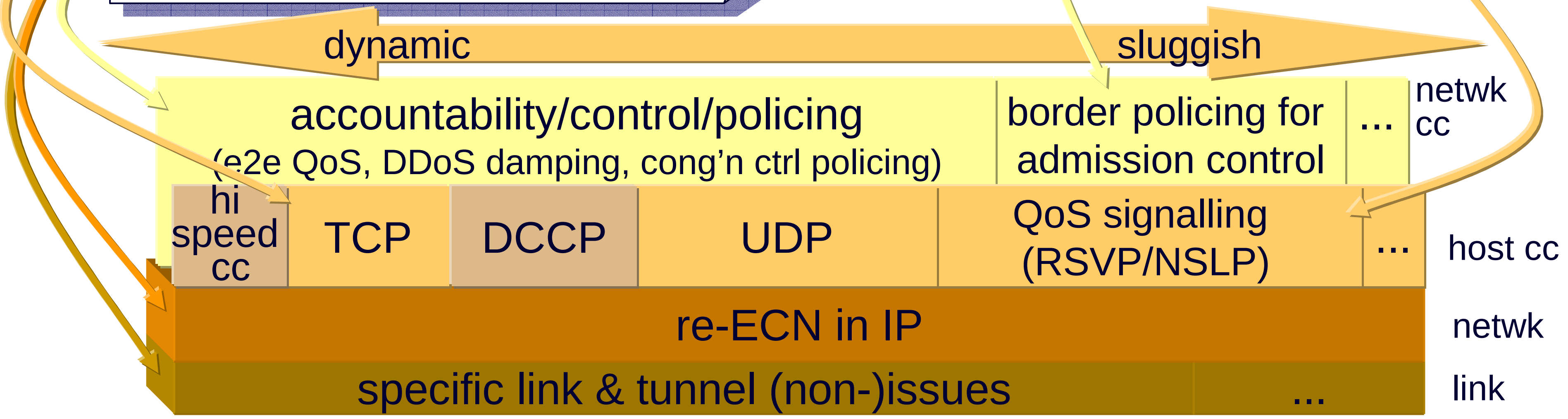
Re-ECN: Adding Accountability for Causing Congestion to TCP/IP
draft-briscoe-tsvwg-re-ecn-tcp-02

	intent
§3: overview in TCP/IP §4: in TCP & others §5: in IP	*stds*
§6: accountability apps	*inform'l*

Emulating Border Flow Policing using Re-ECN on Bulk Data
draft-briscoe-tsvwg-re-ecn-border-cheat-01
intent: informational

RSVP Extensions
for Admission Control over Diffserv
using Pre-congestion Notification
draft-lefaucheur-rsvp-ecn-00

	intent
adds congestion f/b to RSVP	*stds*

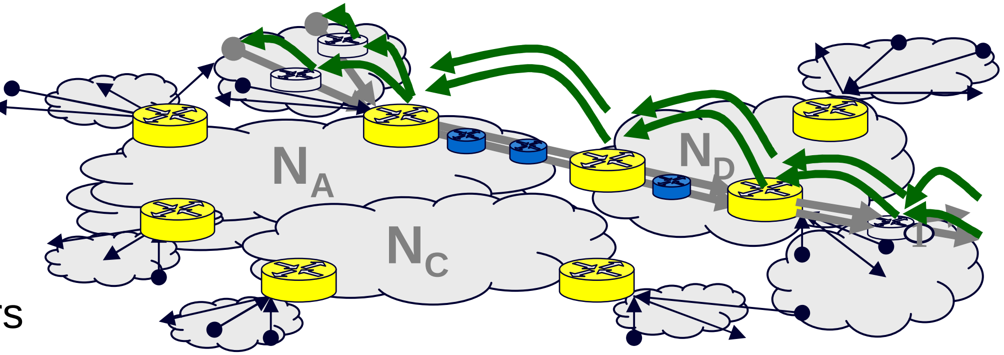


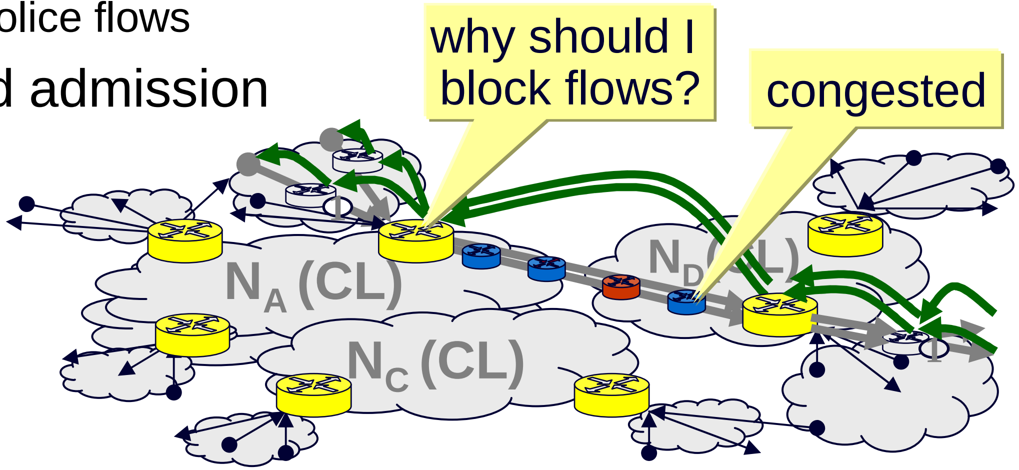
dynamic ⟷ sluggish

accountability/control/policing (e2e QoS, DDoS damping, cong'n ctrl policing)			border policing for admission control	...	netwk cc
hi speed cc	TCP	DCCP	UDP	QoS signalling (RSVP/NSLP) ...	host cc
re-ECN in IP					netwk
specific link & tunnel (non-)issues ...					link

problem statement

- policing flow admission control
  - a network cannot trust its neighbours not to act selfishly
  - if it asks them to deny admission to a flow
    - it has to check the neighbour actually has blocked the data
  - if it accepts a reservation
    - it has to check for itself that the data rate fits within the reservation
- traditional solution
  - flow rate policing at borders
  - session border controllers too complex if they also have to rate police flows
- can pre-congestion-based admission control span the Internet?
  - without per-flow processing at borders?




why should I block flows?

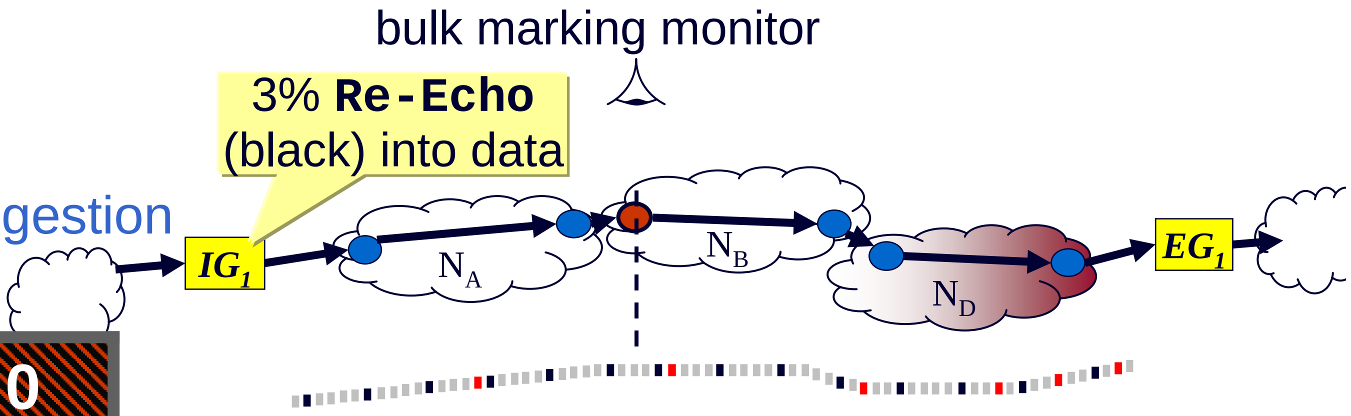
congested

re-ECN for downstream congestion marking

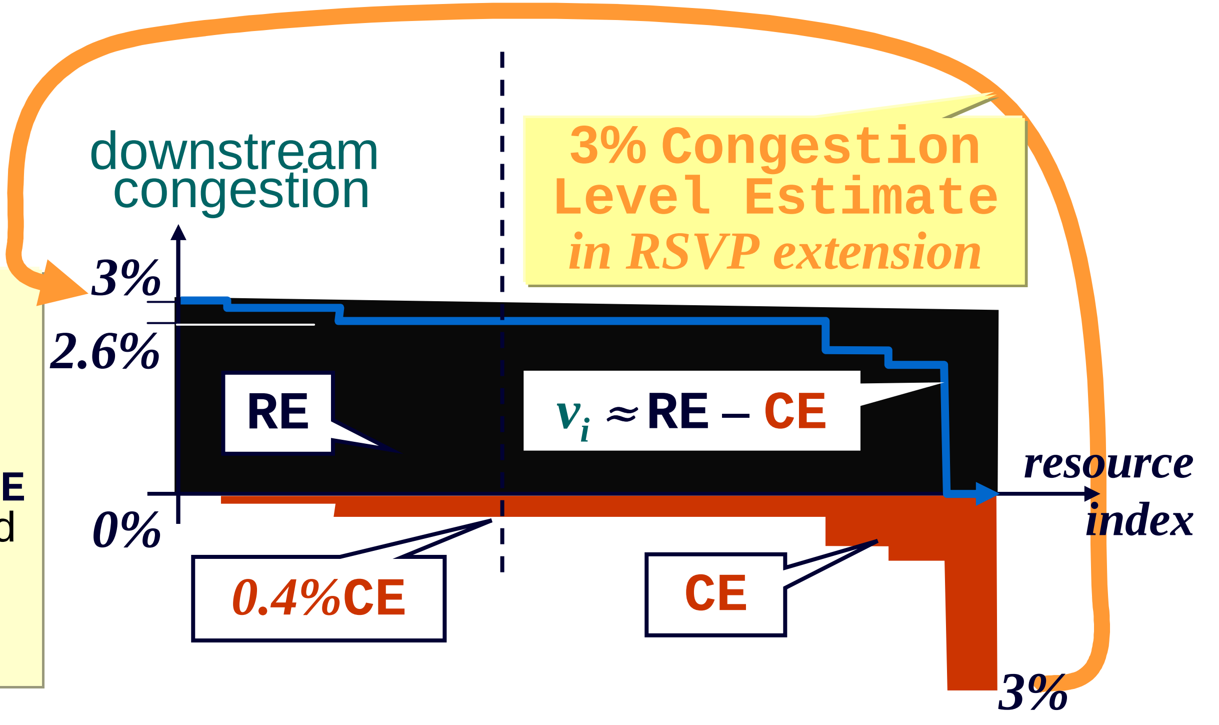
worth

0	+1	0
1	0	-1
RE ECN	ECT(1) 01	CE 11

- ingress gateway blanks **RE**, in same proportion as fraction of **CE** arriving at egress
- N_B applies penalty to N_A in proportion to bulk volume of **RE** less bulk volume of **CE** marked packets over, say, a month
- PCN marking unchanged

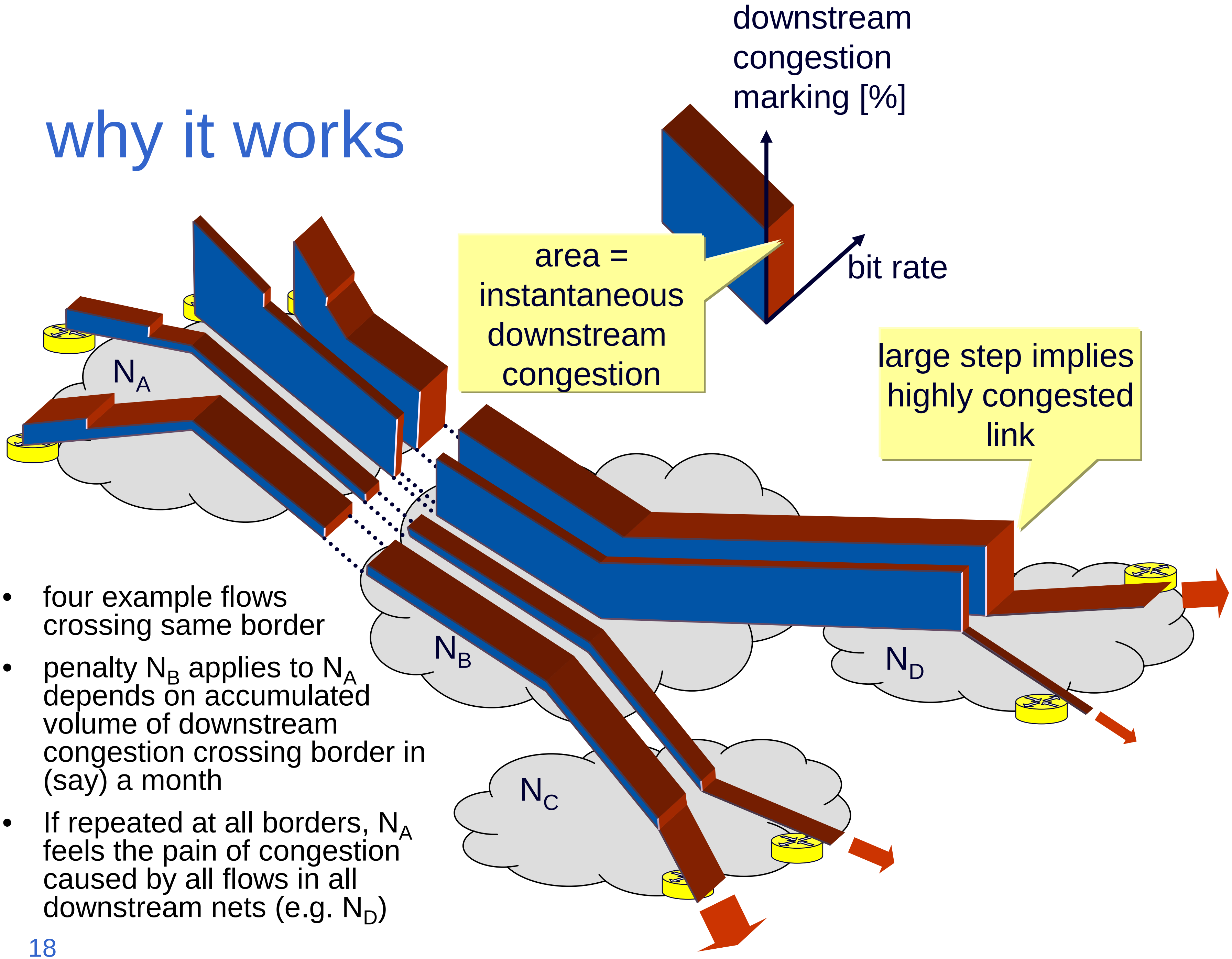


3% **Re-Echo** (black) into data



3% Congestion Level Estimate *in RSVP extension*

why it works

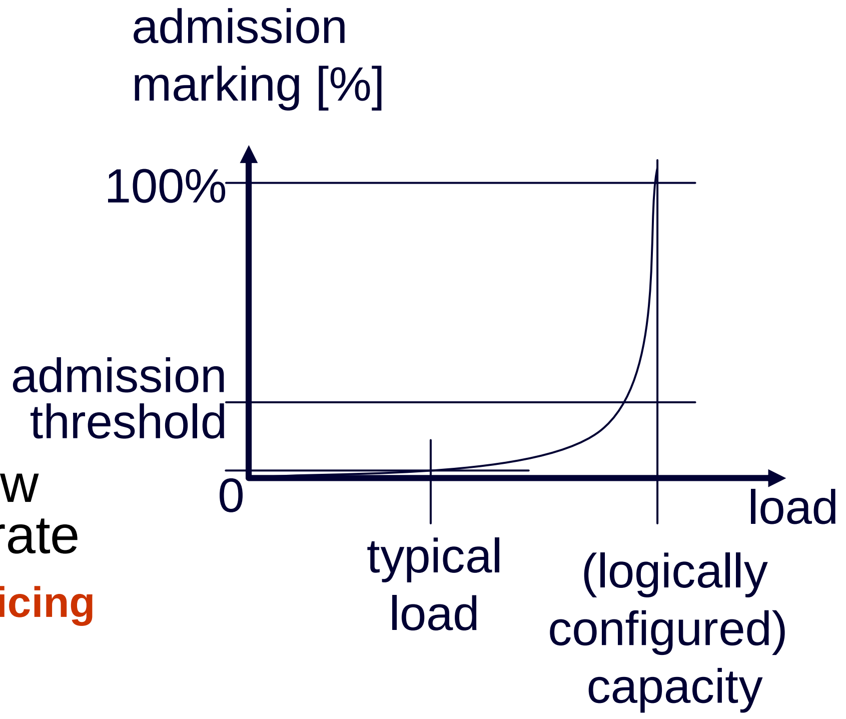


area = instantaneous downstream congestion

large step implies highly congested link

- four example flows crossing same border
- penalty N_B applies to N_A depends on accumulated volume of downstream congestion crossing border in (say) a month
- If repeated at all borders, N_A feels the pain of congestion caused by all flows in all downstream nets (e.g. N_D)

solution rationale



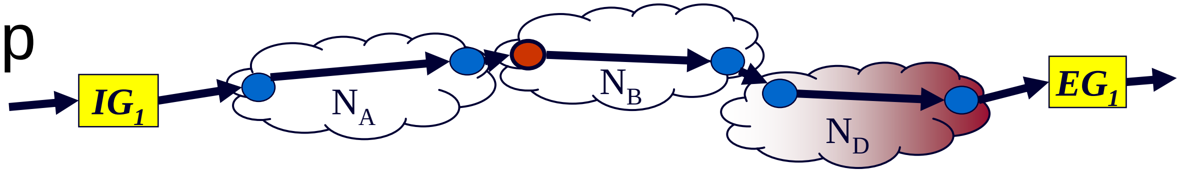
- <0.01% packet marking at typical load
  - addition of any flow makes little difference to marking
- penalties to ingress of each flow appear proportionate to its bit rate
  - **emulates border flow rate policing**
- as load approaches capacity
  - penalties become unbearably high (~1000x typical)
  - insensitive to exact configuration of admission threshold
  - **emulates border admission control**
- neither is a perfect emulation
  - but should lead to the desired behaviour
  - fail-safes if networks behave irrationally (e.g. config errors) – see draft

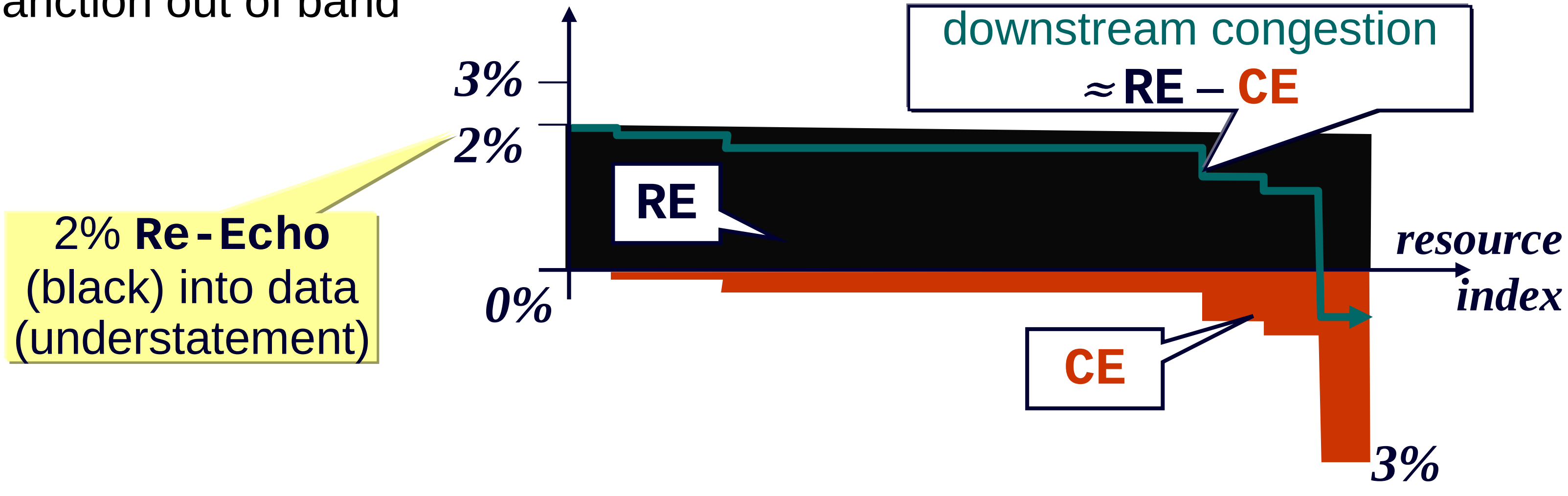
note well: not standardising contracts

- want to avoid protocols that depend on particular business models
  - only standardise the re-ECN protocol
  - then networks can choose to use the metric in various ways
- border penalties could be tiered thresholds, directly proportionate usage charge, etc.
  - networks can choose other, broadly similar arrangements
  - or choose not to use metric, and to do per-flow processing instead
- outside Diffserv region, networks can use whatever flow-based business model they choose, as now

why should ingress re-echo honestly?

- if N_D detects persistent negative balance between **RE** and **CE**, triggers sanctions
- probably not drop
  - raise mgmt alarm
  - sanction out of band





downstream congestion ≈ **RE** – **CE**

2% **Re-Echo** (black) into data (understatement)

dummy traffic attacks on re-ECN

- sanctions against persistently negative flows may not discourage dummy traffic
- various attacks ([Salvatori, Bauer] see draft), eg.
    - a network sends negative dummy traffic with just enough TTL to cross border [Salvatori]
        - offsets penalties from other positive traffic
- fix is to estimate contribution from negative flows crossing border by sampling
    - inflate penalties accordingly – removes attack motivations
    - see draft for details and example algorithm in appendix

summary

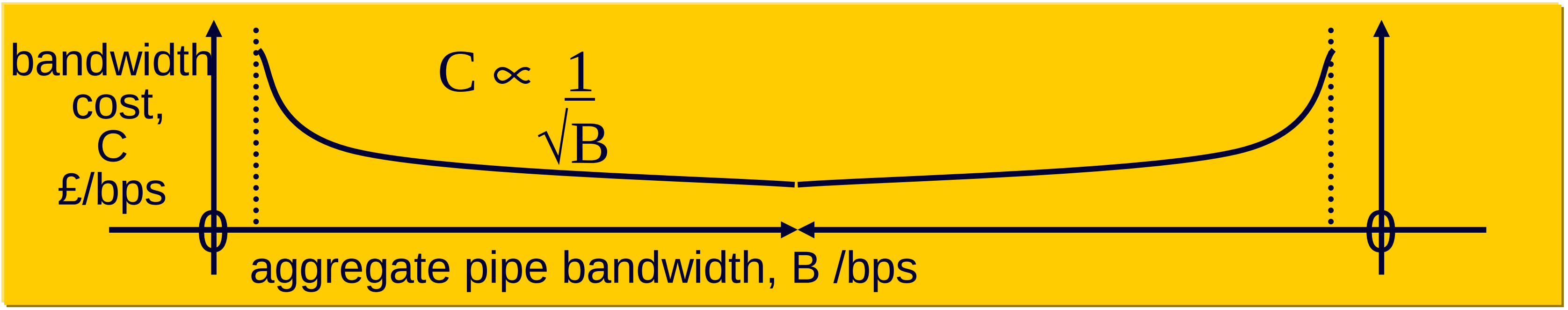
- claim we can now scale flow reservations
  to any size internetwork *and* prevent cheating
  - without per-flow processing in Internet-wide Diffserv region
  - just bulk passive counting of packet marking over, say, a month
  - sufficient emulation of per-flow policing
- see draft for
  - results of security analysis, considering collusions etc.
  - incremental deployment story
  - protocol details (aggregate & flow bootstrap, etc)
  - border metering algorithms, etc
- comments solicited, now or on list

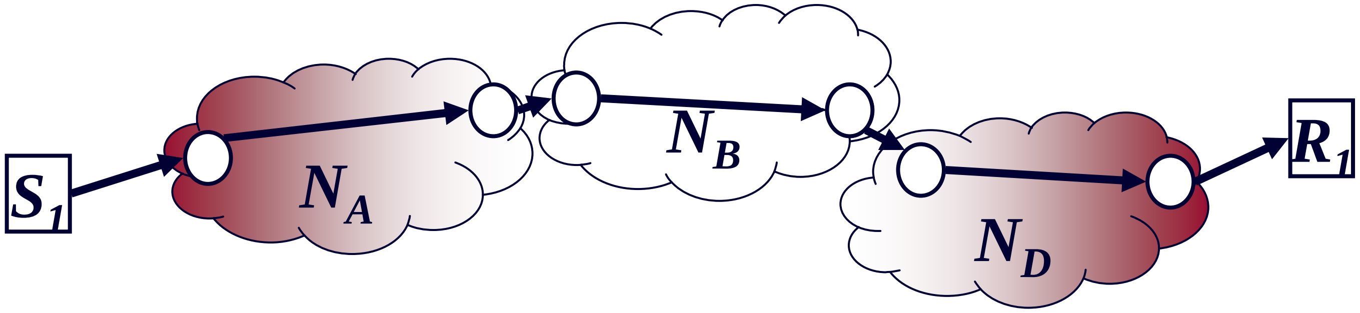
Emulating Border Flow Policing using Re-ECN on Bulk Data

draft-briscoe-tsvwg-re-ecn-border-cheating-01

Q&A

path congestion typically at both edges





- congestion risk highest in access nets
    - cost economics of fan-out
- but small risk in cores/backbones
    - failures, anomalous demand

you MUST do this
you may not do this

- logically consistent statements
- build-time compliance
  - usual standards compliance language (§2)
- run-time compliance
  - incentives, penalties (§6 throttling, dropping, charging)
- hook in datagram service for incentive mechanisms
  - they can make run-time compliance advantageous to all

extended ECN codepoints: summary

- extra semantics backward compatible with previous ECN codepoint semantics

ECN code-point	ECN [RFC3168] codepoint	RE flag	Extended ECN codepoint	re-ECN meaning	`worth'
00	not-ECT	0	Not-RECT	Not re-ECN capable transport	
		1	FNE	Feedback not established	**+1**
01	ECT(1)	0	Re-Echo	Re-echo congestion event	**+1**
		1	RECT	Re-ECN capable transport	**0**
10	ECT(0)	0	---	'Legacy' ECN use	
		1	--CU--	Currently unused	
11	CE	0	CE(0)	Congestion experienced with Re-Echo	**0**
		1	CE(-1)	Congestion experienced	**-1**

flow bootstrap

- feedback not established (**`FNE`**) codepoint; RE=1, ECN=00
  - sent when don’t know which way to set RE flag, due to lack of feedback
  - ‘worth’ **+1**, so builds up credit when sent at flow start
- after idle >1sec
  next packet MUST be **`green`**
  - enables deterministic flow state mgmt (policers, droppers, firewalls, servers)
- **`green`** packets are ECN-capable
  - routers MAY ECN mark, rather than drop
  - strong condition on deployment (see draft)

- **`green`** also serves as state setup bit
  [Clark, Handley & Greenhalgh]
  - protocol-independent identification of flow state set-up
  - for servers, firewalls, tag switching, etc
  - don’t create state if not set
  - may drop packet if not set but matching state not found
  - firewalls can permit protocol evolution without knowing semantics
  - some validation of encrypted traffic, independent of transport
  - can limit outgoing rate of state setup
- considering I-D [Handley & Greenhalgh]
  - state-setup codepoint independent of, but compatible with, re-ECN
- **`green`** is ‘soft-state set-up codepoint’ (idempotent), to be precise

previous re-ECN protocol (IP layer)

ECN code-point	standard designation
00	not-ECT
10	ECT(0)
01	ECT(1)
11	CE

- sender re-inserts congestion feedback into forward data: “re-feedback”

on every **Echo-CE** from transport (e.g. TCP)

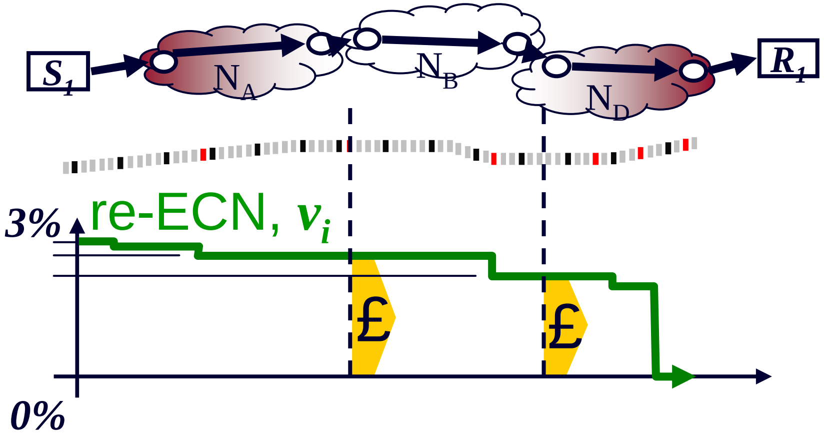
sender sets **ECT(0)**
else sets **ECT(1)**

- Feedback-Established (FE) flag

IPv4 control flags		
FE	DF	MF

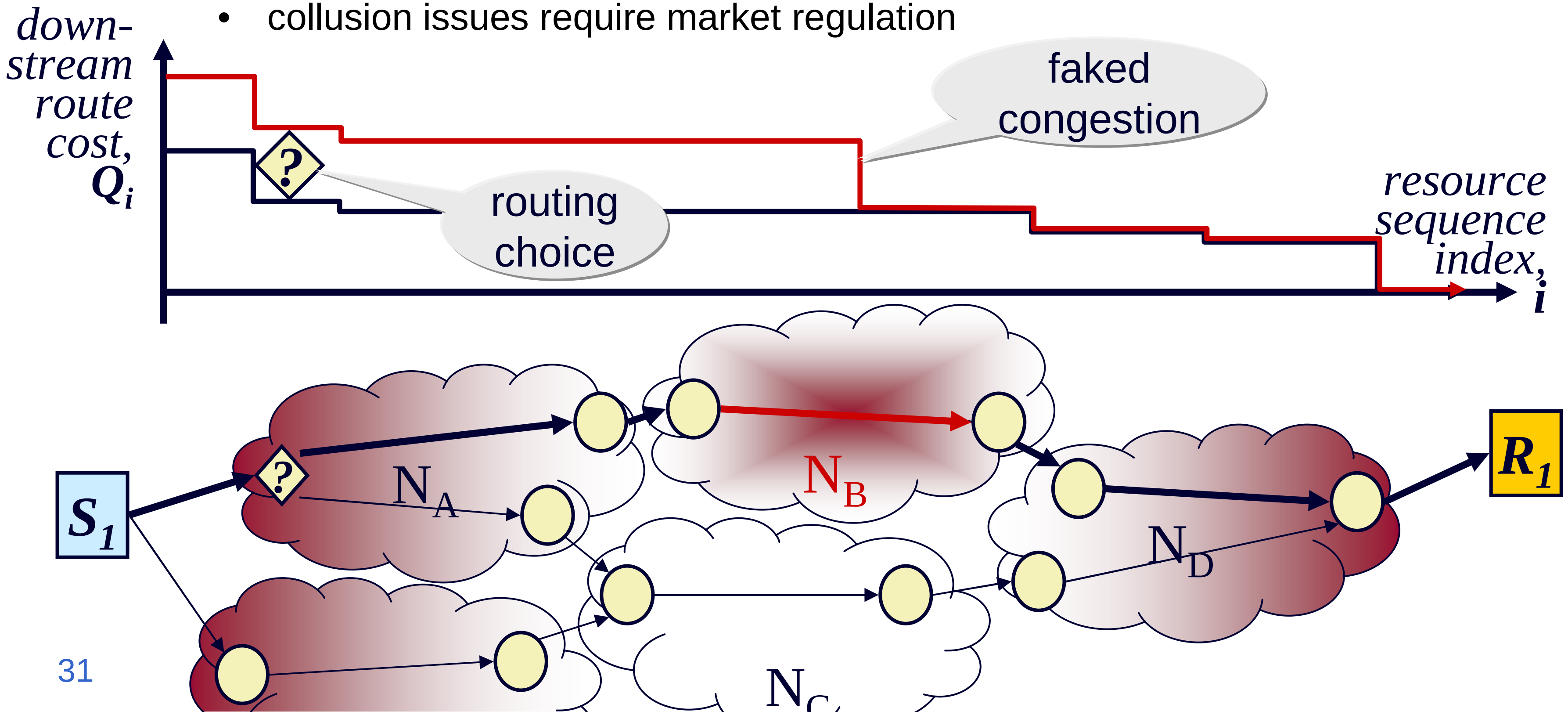
other applications

- congestion-history-based policer (congestion cap)
  - throttles causes of past heavy congestion (zombies, 24x7 p2p)
- DDoS mitigation
- QoS & DCCP profile flexibility
  - ingress can unilaterally allow different rate responses to congestion
- load sharing, traffic engineering
  - multipath routers can compare downstream congestion
- bulk metric for inter-domain SLAs or charges
  - bulk volume of **`ECT(0)`**less bulk volume of **`CE`**
  - upstream networks that do nothing about policing, DoS, zombies etc will break SLA or get charged more



congestion competition – inter-domain routing

- if congestion → profit for a network, why not fake it?
  - upstream networks will route round more highly congested paths
  - N_A can see relative costs of paths to R_1 thru N_B & N_C
- the issue of monopoly paths
  - incentivise new provision
  - collusion issues require market regulation


down-stream route cost, Q_i
?
routing choice
faked congestion
resource sequence index, i
S_1
?
N_A
N_B
N_C
N_D
R_1