Advanced Networking

TCP
SACK, RED/ECN and Throughput Modeling

Renato Lo Cigno
Renato.LoCigno@dit.unitn.it

TCP Throughput

- What is the throughput achievable by TCP?
- Integral of the window size in time
- Can we predict TCP throughput?
- What are the free parameters
  - Loss probability (is it independent from TCP itself?)
  - RTT, Number of connections, ...
- Can we decouple flow from congestion control?
- Can we avoid dropping packets due to congestion?

SACK Option (RFC 2081)

- Negotiation at startup to verify if both ends are enabled
- "Holes" in the receiver buffer sent back to the sender as couple of pointers in ACK optional fields
- Can improve performance (not much!) with highly correlated losses
- Can sometimes lead to blocks and timeouts (implemntation bugs?)

Timestamp Option

- RFC 1323 (as windowscale)
- "Normal" TCP can only compute one RTT sample per window since the only timer is overwritten every new transmission
- With timestamp each segment is stamped with time, ACKs are stamped too
- There can be 1 RTT sample per segment
- RTT can be computed more precisely and RTO can be set more accurately
- Additionally can solve window wrap around problems

Explicit Congestion Notification (ECN)

- RFC 3168
- Routers alert end systems to growing congestion
  - End systems reduce offered load
  - With implicit congestion notification, TCP deduces congestion by noting increasing delays or dropped segments
- Benefits of ECN
  - Prevents unnecessary lost segments
    - Alert end systems before congestion causes dropped packets
    - Retransmissions which add to load avoided
  - Sources informed of congestion quickly and unambiguously
    - No need to wait for retransmit timeout or three duplicate ACKs
- Disadvantages
  - Changes to TCP and IP header
  - New information between TCP and IP
    - New parameters in IP service primitives

Changes Required for ECN

- Two new bits added to TCP header
  - TCP entity on hosts must recognize and set these bits
- TCP entities exchange ECN information with IP
- TCP entities enable ECN by negotiation at connection establishment time
- TCP entities respond to receipt of ECN information
- Two new bits added to IP header
  - IP entity on hosts and **routers** must recognize and set these
- IP entities in hosts exchange ECN information with TCP
- IP entities in routers must set ECN bits based on congestion

IP Header

- Prior to introduction of differentiated services IPv4 header included 8-bit Type of Service field
- IPv6 header included 8-bit traffic class field
- With DS, these fields reallocated
  - Leftmost 6 bits dedicated to DS field,
  - Rightmost 2 bits designated currently unused (CU)
- RFC 3260 renames CU bits as ECN field

The ECN field has the following interpretations:

Value	Label	Meaning
00	Not-ECT	Packet is not using ECN
01	ECT (1)	ECN-capable transport
10	ECT (0)	ECN-capable transport
11	CE	Congestion experienced

TCP Header

- To support ECN, two new flag bits added
- ECN-Echo (ECE) flag
  - Used by receiver to inform sender when CE packet has been received
- Congestion Window Reduced (CWR) flag
  - Used by sender to inform receiver that sender's congestion window has been reduced

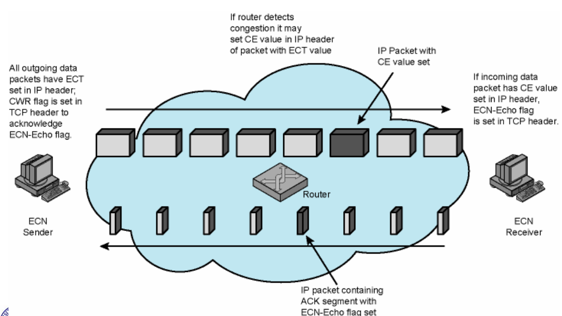
TCP Initialization

- TCP header bits used in connection establishment to enable end points to agree to use ECN
- A sends SYN segment to B with ECE and CWR set
  - A ECN-capable and prepared to use ECN as both sender and receiver
- If B prepared to use ECN, returns SYN-ACK segment with ECE set CWR not set
- If B not prepared to use ECN, returns SYN-ACK segment with ECE and CWR not set

Basic Operation

- TCP host sending data sets ECT code (10 or 01) in IP header of every data segment sent
- If sender receives TCP segment with ECE set, sender adjusts congestion window as for fast recovery from single lost segment
- Next data segment sent has CWR flag set
  - Tells receiver that it has reacted to congestion
- If router begins to experience congestion, may set CE code (11) in any packet with ECT code set
- When receiver receives packet with ECT set it begins to set ECE flag on all acknowledgments (with or without data)
  - Continues to set ECE flag until it receives segment with CWR set

Basic ECN Operation



Open problems

- How to properly react to multiple ECN indications so that window is not reduced too much
  - make a single reduction per RTT
  - some “complicated” heuristics to achieve that
- When buffers are full, packets are lost in bursts
  - problem in general with TCP, also with ECN
- Can we manage buffers to avoid bursty losses?

Random Early Detection (RED)

- Or Random Early Discard (again RED ☺)
- Routers can ser ECN congestion bits (or discard packets) before actual losses occur for overflow
- Distributes losses (markings) more evenly across connections
- Avoid (hopefully!), noxious "sutained overload" buildup
- Unfortunately does not work well
  - Why?
  - Modeling/Understanding/Insight

TCP related projects

- Understand TCP empiric models (2-3 possible projects)
  - Possible to associate with a presentation/seminar to the class
- TCP Control Tecniques (1-2 projects)
  - RED
  - RIO
  - Variations thereof
- TCP as a closed loop system (2-3 projects)
  - Analysis
  - Design
- Simulation with ns-2 of simple scenarios (∞ projects)