Protection Mechanisms for Well-behaved TCP Flows from Tampered-TCP at Edge Routers

Junichi Maruyama, Go Hasegawa, and Masayuki Murata
Osaka University, JAPAN

Congestion control in today’s Internet

• Transmission Control Protocol (TCP)
  – Instrumental in preventing congestion collapse
  – Limit transmission rate at the source
  – Window-based rate control -- Congestion window (CWND)
• Four algorithms:
  – Slow-start
  – Congestion-avoidance
  – Fast-retransmit
  – Fast-recovery
• Additive Increase
  Multiplicative Decrease (AIMD) algorithm

Definition of tampered-TCP

• TCP variants which modify their congestion control mechanisms for higher throughput
  – Larger increasing ratio (α) and smaller decreasing ratio (β)
  – Easy to be implemented to sender side TCP

Objectives of this work

• Propose new mechanisms to detect and regulate tampered-TCP connections at edge routers
  – Check the tampering property of TCP flows passing through an edge router
  – Discard packets from tampered TCP flows by considering the effect on TCP throughput
• Two approaches
  – Cwnd-based mechanism and Throughput-based mechanism

Cwnd-based Mechanism (1/2)

• Monitor TCP flow’s packets for estimating congestion window size
  – TCP sends packets in a window in bursty fashion
  – A long interval between successive two packets is considered as the boundary of two windows
• Estimate increase and decrease ratio of the window size (α, β) of each flow
  – From the changes in the estimated window size
  – Estimate the packet loss ratio at the router
    – By using number of transmitted/dropped packets at the router, obtained from the router’s MIB
Cwnd-based Mechanism (2/2)

- Assess the tampering property
  - The connection that doesn’t satisfy the fairness against TCP Reno connections is considered as tampered-TCP
    \[
    \frac{4}{3} \left( 1 - p' \right) < 1 - p_r \quad \left( 0 < p_r < 1 \right)
    \]
- Set the target packet discarding probability \( p' \)
  - \( p' \) is determined so as to equalize the throughput of the TCP Reno connection with that of the regulated tampered-TCP connection

Throughput-based Mechanism

- Observe TCP flow’s throughput
  - Taken from traffic monitoring tools such as sFlow and NetFlow
- Calculate the estimated Reno’s throughput
  - TCP Reno’s throughput under the same condition [18]
    \[
    \frac{RTT}{3} \left( 1 - p_r \right) \leq \tau_{\text{new}} \leq \min \left\{ 1, \frac{3}{16} \left( 1 - p_r \right) \right\} \left( 0 < p_r < 1 \right)
    \]
- Assess the tampering property
  - The TCP throughput is proportional to the inverse of the square root of the packet loss rate
    \[
    p' = \frac{1}{2} \left( 1 - p_r \right) \left( 0 < p_r \right)
    \]
- Set the target packet discarding probability \( p' \)
  - The TCP throughput is proportional to the inverse of the square root of the packet loss rate

Simulation setting

- 20 Reno flows and one tampered flow
- Bottleneck link: 50 Mbps, 10msec, 333 packets buffer
- Packet size: 1500 bytes
- Simulation time: 70 seconds
  - At the beginning, only TCP Reno flows transmit data
  - After 10 seconds, the tampered-TCP connection joins the network
- Metric
  - Throughput, detection time, detection ratio, and false positive ratio

Simulation results: Throughput ratio

- No control
  - The throughput ratio is larger than 1 in almost all region, meaning that the tampered-TCP connection occupies the bandwidth

Conclusion and future works

- We proposed new mechanisms at edge routers to detect and control tampered-TCP connections
  - Cwnd-based method
  - Throughput-based method
  ⇒ The proposed methods can keep the throughput ratio around 1, by intentionally discarding packets from tampered flows

Future works

- Investigation of the performance of the proposed mechanisms in the actual Internet environment
- Simulations with TCP variants for high-speed and long-distance networks
  - High-speed TCP, compound TCP, CUBIC, …
Simulation results: Detection ratio

- Both of proposed methods can detect tampered TCP flows regardless of the degree of tampering characteristics.

Simulation results: False positive ratio

Table: Throughput and throughput ratio of misassessed TCP Reno and Successfully-assessed TCP Reno connections.

<table>
<thead>
<tr>
<th>Method</th>
<th>(α, β)</th>
<th>Throughput (Mbps)</th>
<th>Throughput ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cwnd</td>
<td>(10, 0.7)</td>
<td>2.533</td>
<td>0.973</td>
</tr>
<tr>
<td>Througput</td>
<td>(10, 0.7)</td>
<td>2.648</td>
<td>1.081</td>
</tr>
</tbody>
</table>

The stronger the tampering property becomes, the larger false positive ratio becomes. The throughput or window size of TCP Reno unstably changes.