Performance of multipath HIP vs MPTCP

Dmitriy Kuptsov, Ilya Nikolaevskiy, Tatiana Polishchuk, Andrei Gurtov
HIIT, Aalto University
CWC, University of Oulu
Finland
Background and Motivation

- Multiple number of network interfaces per device
- Bandwidth aggregation and more reliable communication for multihomed hosts
- Several transport layer solutions: MP-TCP, MP-SCTP, MRTP
- HIP provides abstraction between transport and network layers:
  - “Optimal” place to implement generic multipath routing
Advantages of multipath HIP

- Multipath functionality for all transport layer protocols
- Support for legacy applications
- Middle box traversal
- Mobility support
- Security is available out-of-the-box
Multipath HIP implementation

- Per path SAs are established using HIP multihoming extension
- Forwarding
  - Fastest path first forwarding rule
  - Buffer support to minimize reordering
  - Periodic path probing and statistics aggregation
- Prototype implementation in HIPL
Evaluation. Setup 1

- Small testbed:
  - Host with 2 wireless interfaces
  - Host with 2 wired interfaces
  - Both connected to a server with a single interface
- TCP bulk transfer
- MPTCP Linux implementation as comparison point (MPTCP version 0.6, kernel version 2.6.36)
Wired experiment (mHIP w/o buffer)

- We achieved almost ~80% of the aggregated bandwidth: single wired path - 9.3 Mb/s, 2 paths - 16.5 Mb/s
- Almost no variance in throughput
Wireless experiment

- **mHIP (w/o buffer):** Aggregated bandwidth increases steadily by > 20% with tweaked TCP dupthresh parameter
- **mHIP (w buffer):** Median throughput increased by > 35%
- **MPTCP:** Merely shows a marginal improvement (~10%)

Both MPTCP and mHIP use dupthresh that gives best results
MHIP w/o buffer. TCP dupthresh tweaks

- The highest median throughput (>24Mb/s) is achieved with TCP reordering factor set to 7
- Tweaking dupthresh is useful depending on network conditions
Evaluation. Setup 2. Synthetic tests

- Goal: Controlled experiments
  - Emulate specific loss, delay and jitter
- Emulate wide spectrum of network conditions: from mild to harsh
- Observe the trends for mHIP and MPTCP
Results

- **Light gray**: both mHIP and MPTCP perform well in almost ideal networks.

- **Gray**: mHIP performs relatively well in networks with heterogeneous links with high delays and jitter. MPTCP is not even close to single path TCP.

- **Dark gray**: under sever loss neither mHIP nor MPTCP can perform well.

<table>
<thead>
<tr>
<th>delay(ms)/jitter(ms)/loss(%)</th>
<th>Goodput (Mbits/s) median/std.dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TCP (1st)</td>
</tr>
<tr>
<td>0/0/0.1</td>
<td>8.7</td>
</tr>
<tr>
<td>5/3/0</td>
<td>8.1</td>
</tr>
<tr>
<td>5/3/0</td>
<td>8.1</td>
</tr>
<tr>
<td>5/2/0.1</td>
<td>7.45</td>
</tr>
<tr>
<td>20/10/0</td>
<td>7</td>
</tr>
<tr>
<td>100/50/0</td>
<td>2.2</td>
</tr>
<tr>
<td>100/50/0</td>
<td>2.3</td>
</tr>
<tr>
<td>100/30/0</td>
<td>4.9</td>
</tr>
<tr>
<td>50/30/5</td>
<td>0.55</td>
</tr>
<tr>
<td>100/0/0</td>
<td>5.2</td>
</tr>
<tr>
<td>100/5/0</td>
<td>4.45</td>
</tr>
<tr>
<td>100/5/0.1</td>
<td>3.1</td>
</tr>
<tr>
<td>0/30/5</td>
<td>0.55</td>
</tr>
<tr>
<td>100/5/0.1</td>
<td>3</td>
</tr>
</tbody>
</table>
Conclusions

- Multipath HIP:
  - Works perfectly well in networks that have low jitter
  - In networks with high jitter, multipath HIP achieves > 20% gain when TCP reordering factor is tuned properly and no additional buffer is used
  - In networks with high jitter multipath HIP achieves >35% gain when buffer is used and no additional tweaks to TCP parameters is needed
  - Poorly performs under sever loss
Conclusions

- **MPTCP (or rather its particular implementation):**
  - Works perfectly well in almost ideal networks (small delay and almost no jitter)
  - Performs awfully when links are heterogeneous, e.g., have different delays
  - Does not show good results when paths have high delays and jitter
  - Poorly performs under severe loss
- TCP reordering factor can be tuned in flight to adapt TCP to channel conditions
Future work

- We still have to experiment with a network which has high jitter and high loss characteristics
- Optimal parameters for buffer (timeout, size, etc), and TCP reordering factor (depending on channel jitter and loss) are still under question
- Some minor bug fixes in HIPL code