Identifying Traffic Differentiation in Mobile Networks

Presented by Melanie Klerer
Motivation

- To understand differentiation as-is
  - Is there discrimination against high bandwidth applications (e.g., Netflix)?
- To measurement of mobile networks from end user perspective, as there has been limited ability to do such
- To investigate how different classes of applications are treated
  - Netflix (high bandwidth) vs Facebook (mixed bandwidth)
Overview of Contributions

- System for detecting differentiation on mobile networks
  - Record and replay that enables testing of arbitrary application (open or closed source)
- Android application that conducts differentiation tests without the need for special privileges
- Study of differentiation in mobile networks from the end user (Android application) perspective
Differentiation in Scope

- Better or worse performance based on application class
- Caused by traffic shaping middleboxes
- Focus on network performance perceived by applications
- Likely triggered by IP address, port number, number of connections, total bandwidth, time of day
System for detecting differentiation on the mobile network
Bytes transferred over time with and without preserving packet timing

WHAT'S THE DEAL WITH THESE GRAPHS?

THEY ARE SO COLORFUL AND MISLEADING, ARE THEY A 90'S SITCOM?
Validation

- Test system in testbed utilizing commercial shaping products
- Replay traffic with different features modified to reverse engineer classification
- Conclusion: replay successfully detects differentiation in operational networks, thus should extend beyond the test bed
Android Application in action
Challenges with identifying differentiation

- Per-client management
- NAT behavior
- “Translucent” HTTP proxies
- Content modifying proxies and transcoders
- Caching
### Differentiation Results

<table>
<thead>
<tr>
<th>ISP</th>
<th>YT</th>
<th>NF</th>
<th>SF</th>
<th>SK</th>
<th>VB</th>
<th>HO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Verizon</td>
<td>m</td>
<td>m</td>
<td>m</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T-Mobile</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AT&amp;T</td>
<td>f</td>
<td>f</td>
<td>f</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sprint</td>
<td>m/p</td>
<td>m/p</td>
<td>m/p</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Boost</td>
<td>m</td>
<td>m</td>
<td>m</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BlackWireless</td>
<td>60%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H2O</td>
<td>37%*</td>
<td>45%*</td>
<td>65%*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SimpleMobile</td>
<td>36%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NET10</td>
<td>p</td>
<td>p</td>
<td>p</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5: Shaping detection results per ISP in our dataset, for six popular apps: YouTube (YT), Netflix (NF), Spotify (SF), Skype (SK), Viber (VB), and Google Hangout (HO). When shaping occurs, the table shows the difference between average throughput (%) we detected. A dash (-) indicates no differentiation, (f) means IP addresses changed for each connection, (p) means a “translucent” proxy changed connection behavior from the original app behavior, (m) indicates that a middlebox modified content in flight between client and server. *For H2O network, replays with random payload have better performance than VPN and exposed replays, indicating a policy that favors non-video HTTP over VPN and streaming video.
Thoughts about the paper?