Cache Digests for HTTP/2

Kazuho Oku
Pull Request #413

• proposes:
  – SENDING_CACHE_DIGEST SETTINGS Parameter
  – switch to Cuckoo Filters
  – thanks to Yoav Weiss for the proposal
• a SETTINGS parameter sent by client
  – “I’m going to send CACHE_DIGEST, so the server should decide what to push after seeing the digest”

• client strategy:
  – send CACHE_DIGEST frame for every request that goes to a new origin
  – send an empty CACHE_DIGEST frame with RESET flag set as a sign of not providing a digest for that particular origin
Cuckoo filters – the motivation

• on the client, maintain a persistent structure that can be sent as a digest
  – rather than iterating through the cache to build a digest using Golomb-coded Sets (GCS)
• events that modify the Cuckoo filter:
  – insert(url, etag)
  – evict(url, etag)
• cannot have fresh vs. stale distinction
  – since the browser cache does not fire an event when an entry becomes stale
Cuckoo filters – the size

- preliminary results (3,250 entries, P=1/256)

<table>
<thead>
<tr>
<th># of entries</th>
<th>full capacity</th>
<th>size (bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>uncompressed</td>
</tr>
<tr>
<td>1,021</td>
<td>4,084</td>
<td>5,637</td>
</tr>
<tr>
<td>1,109</td>
<td>4,436</td>
<td>11,269</td>
</tr>
<tr>
<td>2,019</td>
<td>8,076</td>
<td>11,269</td>
</tr>
<tr>
<td>4,027</td>
<td>16,108</td>
<td>22,533</td>
</tr>
</tbody>
</table>

from https://github.com/httpwg/http-extensions/pull/413#issuecomment-344949750
note: the 1,021 entries table has false-negative rate of ~1% due to collisions
GCS: 3,987 bytes
Cuckoo filters – the size

- the need to send digests of stale objects is an issue
  - we need to figure out how to push 304 in order to *use* stale digests
  - browsers could have 2x stale objects than fresh ones

<table>
<thead>
<tr>
<th>domain</th>
<th>fresh</th>
<th>stale</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>*.facebook.com</td>
<td>790</td>
<td>1,483</td>
<td>2,273</td>
</tr>
<tr>
<td>*.google.com</td>
<td>373</td>
<td>630</td>
<td>1,003</td>
</tr>
</tbody>
</table>
Cuckoo filters – server-side

• pros:
  – no need to decode before using the digest

• cons:
  – URL and ETag of the resource that you might push is required to lookup the filters even for fresh resources
    • only URL is needed in case of GCS of fresh digests
Cuckoo filters – the options

• a) replace GCS with Cuckoo filters?
  – we’d need to wait for a working implementation
• b) define both algorithms?
  – we could have a field that indicates the algorithm
• c) stick to using GCS
  – Cuckoo filters is essentially a per-origin metadata; browsers might be able to use such kind of metadata (e.g. list of [URL, Etag, becomes_stale_at]) to generate GCS