HTTP Random access and live content
(status update)

Craig Pratt, Barbara Stark, Darshak Thakore
WG draft

• Review requested at IETF 98
  • No specific concerns/comments received

• Working on testing the “protocol”
  • Main focus on checking ”intermediary” behavior
Test framework

• Creating a “continuous” live video stream
• Hosted thru a
  • CDN
  • Caching proxy

• Custom client that issues the “live” byte range requests
• Test
  • that the implementation works as expected
  • Intermediaries (cdn, cache) work as expected
• Send out observations/results
Next steps

• Any feedback/questions?

• Issue LC?
  • Keep it open till we finish the testing
Background slides
How it works

• Client uses Range semantics to determine accessible bytes

<table>
<thead>
<tr>
<th>REQUEST</th>
<th>RESPONSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>HEAD /my_resource HTTP/1.1</td>
<td>HTTP/1.1 206 Partial Content</td>
</tr>
<tr>
<td>Range: bytes=0-</td>
<td>Content-Range: bytes 0-99408383/*</td>
</tr>
<tr>
<td></td>
<td>Content-Length: 99398384</td>
</tr>
</tbody>
</table>

• Client attempts to “discover” live random access support

<table>
<thead>
<tr>
<th>REQUEST</th>
<th>RESPONSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET /my_resource HTTP/1.1</td>
<td>HTTP/1.1 206 Partial Content</td>
</tr>
<tr>
<td>Range: bytes=99400000-9223372036854775</td>
<td>Content-Range: bytes 99400000-9223372036854775/*</td>
</tr>
<tr>
<td></td>
<td>Transfer-Encoding: chunked</td>
</tr>
</tbody>
</table>
“backward” compatibility

• "non supporting" server will respond as per RFC7233

REQUEST
GET /my_resource HTTP/1.1
Range: bytes=99400000-9223372036854775

RESPONSE
HTTP/1.1 206 Partial Content
Content-Range: bytes 99400000-99634867/*
Transfer-Encoding: chunked

Provides “large number” to indicate live random access

Non-supporting server sends back what it can support