Digest ~~Headers~~ Fields

(was: Resource Digests, was: RFC 3230)

HTTPWG Interim 2022-02

draft-ietf-httpbis-digest-headers

[last interim slides] [latest editor copy]

Since September 2021 Interim

WGLC

- Deep feedback from a few people, thank you
- Addressed most of this, some editorial change are still to do
- The question of Structured Fields (SF) came up again
  - One way or the other, let’s agree an answer and not revisit the question

Digest fields in editors copy

-07 plus WGLC edits

```
Digest = 1#representation-data-digest
   representation-data-digest = digest-algorithm "="
                                <encoded checksum output>

Content-Digest = 1#content-digest
   content-digest = digest-algorithm "="
                    <encoded checksum output>

Want-Digest = 1#want-digest-value
Want-Content-Digest = 1#want-digest-value
want-digest-value = digest-algorithm [ ";" "q" "=" qvalue]

digest-algorithm = token
```

These headers use the #rule for a list syntax. Compatible with RFC 3230.

These lists contain token.

Encoded checksum output format varies. Not all base64. Some allow different character sets.

Summary: Incompatible with Structured Fields

So what about Structured Fields?

Option 1: Status Quo.
Achieves goal of updating Digest and Want-Digest. Adds Content-Digest and Want-Content-Digest by popular demand. Keep legacy list format for all.

Option 2: “Three headers”
Achieves goal of updating Digest and Want-Digest. Both remain as legacy list.
New: Representation-Digest and Content-Digest are SF.
New: Want-Representation-Digest and Want-Content-Digest are SF.

Option 3: “Two headers”
Digest and Want-Digest **do not** get updated. RFC 3230 stays alive but inconsistent if people want it.
New: Representation-Digest and Content-Digest are SF.
New: Want-Representation-Digest and Want-Content-Digest are SF.

Option 2: Three headers

PR #1393. Text diff

Clear definition of “Representation Digest” concept that is used in Digest and Representation-Digest. Updates to digest algorithms to support 3 headers.

sf-dictionary - Keys are digest algorithms, values are sf-binary. Dupe keys handled.

```
Representation-Digest = sf-dictionary
Content-Digest = sf-dictionary
```

sf-list - items are digest algorithms. ‘q’ parameter is defined.

```
Want-Representation-Digest = sf-list
Want-Content-Digest = sf-list
```

Option 3: Two headers

PR #1394. Text diff

Basically like Option 2 except less consideration for Digest

sf-dictionary - Keys are digest algorithms, values are sf-binary.

```
Representation-Digest = sf-dictionary
Content-Digest = sf-dictionary
```

sf-list - items are digest algorithms. ‘q’ parameter is defined.

```
Want-Representation-Digest = sf-list
Want-Content-Digest = sf-list
```

Comparison of formats

Current:

```
Digest:                  sha-512=WZDPaVn/7XgHaAy8pmojAkGWoRx2UFChF41A2svX+TaPm
                         AbwAgBWnrIiYllu7BNNyealdVLvRwE\nmTHWXvJwew==
Content-Digest:          sha-512=WZDPaVn/7XgHaAy8pmojAkGWoRx2UFChF41A2svX+TaPm
                         AbwAgBWnrIiYllu7BNNyealdVLvRwE\nmTHWXvJwew==
Want-Digest:             sha-512;q=1, sha-256;q=0.2
Want-Content-Digest:     sha-512;q=1, sha-256;q=0.2
```

New:

```
Digest:                  sha-512=WZDPaVn/7XgHaAy8pmojAkGWoRx2UFChF41A2svX+TaPm
                         AbwAgBWnrIiYllu7BNNyealdVLvRwE\nmTHWXvJwew==
Representation-Digest:   sha-512=:WZDPaVn/7XgHaAy8pmojAkGWoRx2UFChF41A2svX+TaPm
                         AbwAgBWnrIiYllu7BNNyealdVLvRwE\nmTHWXvJwew==:
Content-Digest:          sha-512=:WZDPaVn/7XgHaAy8pmojAkGWoRx2UFChF41A2svX+TaPm
                         AbwAgBWnrIiYllu7BNNyealdVLvRwE\nmTHWXvJwew==:
Want-Digest:             sha-512;q=1, sha-256;q=0.2
Want-Representation-Digest: sha-512;q=1, sha-256;q=0.2
Want-Content-Digest:     sha-512;q=1, sha-256;q=0.2
```

Comparison of formats (easy diff)

Current:

```
Digest:                 sha-512=WZDPaVn/7XgHaAy8pmojAkGWoRx2UFChF41A2svX+TaPm
                        AbwAgBWnrIiYllu7BNNyealdVLvRwE\nmTHWXvJwew==
Content-Digest:         sha-512=WZDPaVn/7XgHaAy8pmojAkGWoRx2UFChF41A2svX+TaPm
                        AbwAgBWnrIiYllu7BNNyealdVLvRwE\nmTHWXvJwew==
Want-Digest:            sha-512;q=1, sha-256;q=0.2
Want-Content-Digest:    sha-512;q=1, sha-256;q=0.2
```

New:

```
Digest:                 sha-512=WZDPaVn/7XgHaAy8pmojAkGWoRx2UFChF41A2svX+TaPm
                        AbwAgBWnrIiYllu7BNNyealdVLvRwE\nmTHWXvJwew==
Representation-Digest:  sha-512=:WZDPaVn/7XgHaAy8pmojAkGWoRx2UFChF41A2svX+TaPm
                        AbwAgBWnrIiYllu7BNNyealdVLvRwE\nmTHWXvJwew==:
Content-Digest:         sha-512=:WZDPaVn/7XgHaAy8pmojAkGWoRx2UFChF41A2svX+TaPm
                        AbwAgBWnrIiYllu7BNNyealdVLvRwE\nmTHWXvJwew==:
Want-Digest:            sha-512;q=1, sha-256;q=0.2
Want-Representation-Digest: sha-512;q=1, sha-256;q=0.2
Want-Content-Digest:    sha-512;q=1, sha-256;q=0.2
```

Pick one and move on

<table>
<tr><th></th><th>Option 1:
Update 3230,
add Content-Digest</th><th>Option 2:
Update 3230,
introduce new Digest SF</th><th>Option 3:
leave RFC3230 behind,
introduce new Digest SF</th></tr>
<tr><td>Digest</td><td rowspan="4">Becomes consistent with HTTP

Syntax backward compatible with RFC3230 to support current implementers (OpenBankingEurope, EU cross-border transactions)

Signature guidance

No SF</td><td>Becomes consistent with HTTP

Syntax backward compatible with RFC3230 to support current implementers

Current implementers can plan a transition to representation-digest</td><td>Remains Inconsistent with HTTP

Current implementers will remain inconsistent with HTTP

No signature guidance</td></tr>
<tr><td>Want-Digest</td></tr>
<tr><td>Content-Digest</td><td rowspan="3">Use SF (List or Dictionary)

New implementers will adopt Representation-Digest</td><td rowspan="3">Use SF (List or Dictionary)

New implementers will adopt Representation-Digest</td></tr>
<tr><td>Want-Content-Digest</td></tr>
<tr><td>Representation-Digest</td><td>X</td></tr>
<tr><td>Want-Representation-Digest</td><td>X</td><td></td></tr>
</table>

If we pick any SF option, there's more work

Need to choose the syntax of SF. Suggestions below

```
Representation-Digest, Content-Digest: sf-dictionary
```

Keys are algorithms. Digest's digest-algorithm is token. Incompatible, need IANA massaging.

```
Want-Representation-Digest, Want-Content-Digest: sf-list
```

List items are sf-token, a little different to key. Needs IANA messaging.

'q' parameter is reinvention of HTTP qvalue. Should we standardize a common SF qvalue rather than reinvent it everywhere?

Thanks!

Roberto Polli - robipolli@gmail.com

Lucas Pardue - lucaspardue.24.7@gmail.com