Unencoded Digest (fka Identity-Digest fka id- prefix)

draft-pardue-httpbis-identity-digest

IETF 122 HTTPbis 2025-03-21

Digestif

- RFC 9530 defines **Content-Digest** and **Repr-Digest**
- It didn't start this way
  - draft-ietf-httpbis-digest-headers-00 redefined how the existing **Digest** header was used
  - We **added** two special digest algorithm identifiers id-sha-256 id-sha-512 to cover the case of *"digest of the representation-data of the resource when no content coding is applied (eg. "Content-Encoding: identity")"*
- In other words, the WG already adopted this work once before in 2019

Antics with semantics

```
GET /boringstring HTTP/1.1
Host: example.org
Accept-Encoding: gzip
```

Antics with semantics

```
GET /boringstring HTTP/1.1
Host: example.org
Accept-Encoding: gzip
```


```
HTTP/1.1 200 OK
Content-Encoding: gzip
Repr-Digest: sha-256=:XyjvEuFb1P5rqc2le3vQm7M96DwZhvmOwqHLu2xVpY4=:
Unencoded-Digest: sha-256=:5Bv3NIx05BPnh0jMph6v1RJ5Q7kl9LKMtQxmvc9+Z7Y=:

<content>
```

Antics with semantics

```
GET /boringstring HTTP/1.1
Host: example.org
Accept-Encoding: gzip
```

Digest of the gzipped response message content

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Repr-Digest: sha-256=:XyjvEuFb1P5rqc2le3vQm7M96DwZhvmOwqHLu2xVpY4=:
Unencoded-Digest: sha-256=:5Bv3NIx05BPnh0jMph6v1RJ5Q7kl9LKMtQxmvc9+Z7Y=:

<content>
```

Digest of the non-gzipped response message content

Antics with semantics

```
GET /boringstring HTTP/1.1
Host: example.org
Accept-Encoding: gzip
```

Digest of the representation

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Repr-Digest: sha-256=:XyjvEuFb1P5rqc2le3vQm7M96DwZhvmOwqHLu2xVpY4=:
Unencoded-Digest: sha-256=:5Bv3NIx05BPnh0jMph6v1RJ5Q7kl9LKMtQxmvc9+Z7Y=:

<content>
```

Digest of the unencoded representation

Antics with semantics


```
HEAD /boringstring HTTP/1.1
Host: example.org
Accept-Encoding: gzip
```

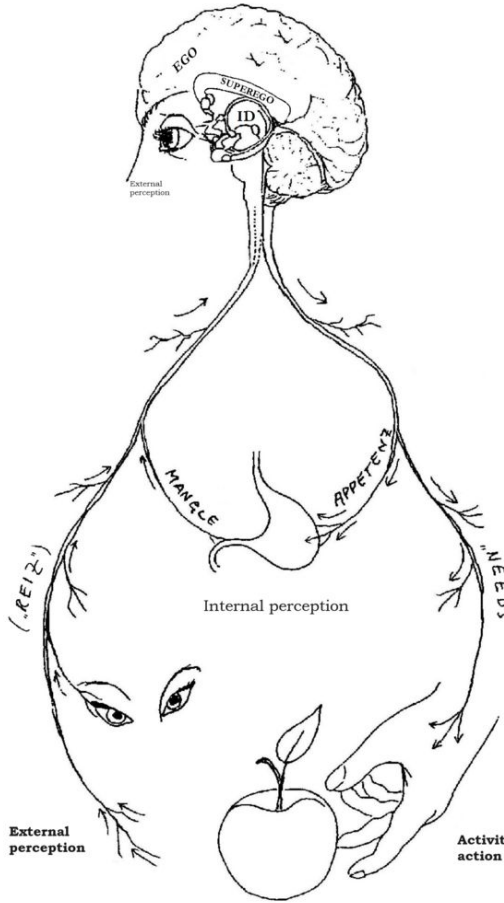
Digest of the representation

```
HTTP/1.1 200 OK
Content-Encoding: gzip
Repr-Digest: sha-256=:XyjvEuFb1P5rqc2le3vQm7M96DwZhvmOwqHLu2xVpY4=:
Unencoded-Digest: sha-256=:5Bv3NIx05BPnh0jMph6v1RJ5Q7kl9LKMtQxmvc9+Z7Y=:
```

Digest of the unencoded representation

id, ego, and superego

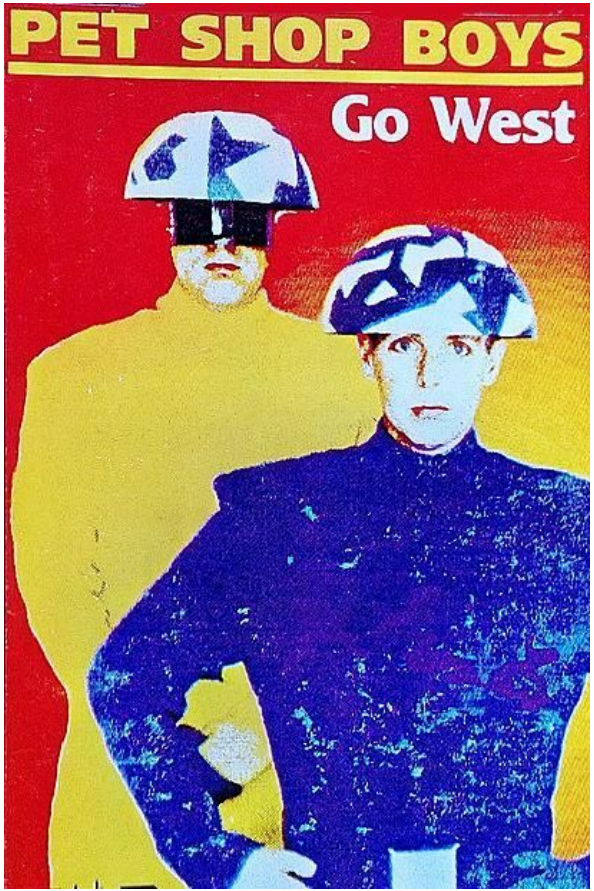
- https://github.com/httpwg/http-extensions/issues/885
  - 2019: "how about we make id- a generic prefix for any algorithm"
  - 2021: "remove id- prefix entirely, punt to some other work"
- Much time elapsed and RFC 9530 published in 2024
  - That process gave us the two headers we know and love
- Anticipating RFC publication, new Identity-Digest I-D published in 2023
  - Short mailing list discussion
  - Naming things is hard
  - No active interest or use cases



Go, Mike West

Signature-based integrity - https://github.com/WICG/signature-based-sri

"TL;DR: It would be nice if web developers could verify the provenance of resources they depend upon, establishing the technical foundations upon which they can increase confidence in the integrity of their dependencies. We offer brittle, content-based integrity mechanisms today which can (in theory) but do not (in practice) enable this capability. This proposal explores an alternative."

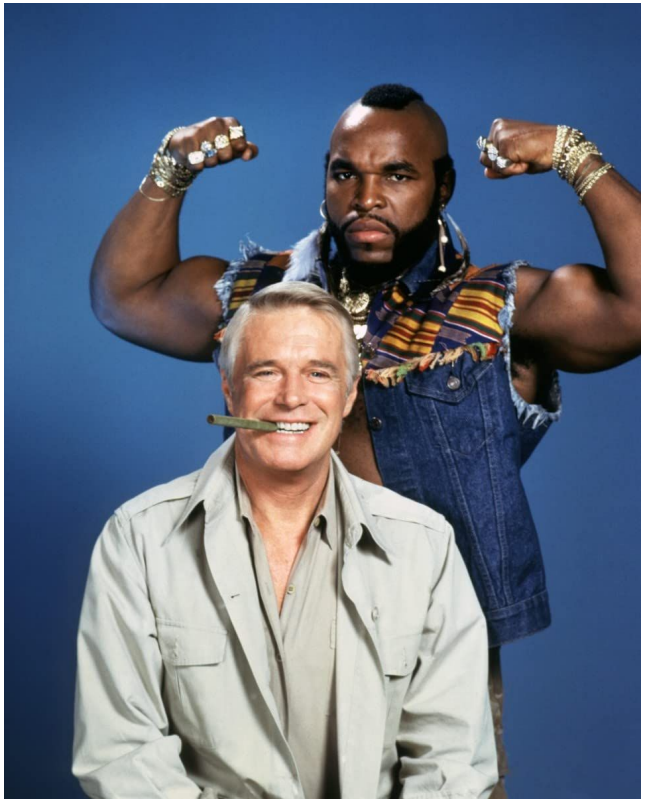


When the plan comes together

Turns out, we built (nearly all) the pieces needed for this: HTTP Signatures and HTTP Digests

What's missing from the IETF is Unencoded-Digest.

```
HTTP/1.1 200 OK
Accept-Ranges: none
Vary: Accept-Encoding
Content-Type: text/javascript; charset=UTF-8
Access-Control-Allow-Origin: *
Identity-Digest: sha-512=:[base64-encoded digest of
`console.log("Hello, world!");`]:
Signature-Input: sig1=("identity-digest";sf); alg="Ed25519";
keyid="[base64-encoded public key]"; tag="sri"
Signature: sig1=:[base64-encoded result of Ed25519([response
metadata], [private key])]:
```



Apéritif ?

Ignore the name. We adopted then punted on the concept of digests of unencoded things.

There's now a use case with implementation intent.

Can we adopt, brush up the minor details and finally publish?