Signing HTTP Messages

draft-ietf-httpbis-message-signatures

HTTP Working Group Virtual Interim Meeting
May 26, 2020
GET / HTTP/1.1
Host: httpwg.org
Accept: text/html
Date: Tue, 20 May 2020 20:51:35 GMT
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Authorization: Bearer 1234abcd5678efab
Identify Elements to Sign

GET / HTTP/1.1
Host: httpwg.org
Accept: text/html
Date: Tue, 20 May 2020 20:51:35 GMT
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US, en; q=0.9
Authorization: Bearer 1234abcd5678efab
Create Signature Input

GET / HTTP/1.1
Server: httpwg.org
Accept: text/html
Date: Tue, 20 May 2020 20:51:35 GMT
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US, en;q=0.9
Authorization: Bearer 1234abcd5678efab

(created): 1590007895
(request-target): GET /
host: httpwg.org
authorization: Bearer 1234abcd5678efab
Attach Signature to Message

Signature: keyId="test-key-a",
created=1590007895,
headers="(created) (request-target) host authorization"
signature="1234567890abcdef..."
Status

• Adopted as working group item
• Published -00 draft
• Converted to Markdown
• Open issues, big and small
Incoming Changes: Structured Headers

Signature-Input: sig1=(::request-target, host, authorization);
   keyId="key-a";
   created=159000789

Signature: sig1=::AbCd1234...==:
Structured Headers: Two Header Fields

• Signature-Input
  • Replaces "headers" parameter
  • Dictionary of lists
  • Other parameters as...parameters

• Signature
  • Dictionary of byte sequences

Signature-Input: sig1=(
  ::request-target, host, authorization);
keyId="test-key-a";
created=159000789

Signature: sig1=:AbCd1234...==:
Signature-Input: sig2=(cookie; name=my_cookie, signature; id=sig1, list-header; count=4, dict-header; key=foo);
keyId="key-b";
created=159000789
Multiple Signatures

Signature-Input: sig2=(
    signature;key=sig1,
    x-forwarded-for);
keyId="key-b";
created=159000789

Signature: sig2=:AbCd1234...==:
Problems Solved

• Confusing "headers" parameter name
• Bespoke header field value format
• Signature parameters are not signed
• Support multiple signatures
  • over different content
  • with different keys
• Signing parts of unstructured headers (e.g., specific cookies)
• Signing parts of structured headers

• ...Signature input construction?
Signature-Input in Signature Input

(::request-target, host, authorization);
  keyId="key-a"; created=159000789
::request-target: GET /
host: httpwg.org
authorization: Bearer 1234abcd5678efab
Next Task: \texttt{keyId} and \texttt{algorithm}

- JWA: use it, reference it, or duplicate it?

- Key and algorithm coupling
  - Was: separate \texttt{keyId} and \texttt{algorithm} parameters
  - Now: vestigal \texttt{algorithm} parameter; key definition determines all
Effects of the Coupling \texttt{keyId} and \texttt{algorithm}

- Prevents signer from using arbitrary algorithms

- Completely externalizes algorithm definition, identification

- Requires multiple key IDs to use the same key with multiple algorithms
  - Maybe this is good, actually?

- ...will implementers invent structured \texttt{keyId} formats?
Other Open Items/Next Steps

• Alignment with Web Packaging's Signed Exchanges

• Signature input format (bespoke or not?)

• Improve serialization rules (e.g., % encoding, collapsing whitespace)

• More content identifiers (::method, ::path, ::query, ...)