httpbis
Internet-Draft
Intended status: Informational
Expires: January 3, 2016

M. Thomson
Mozilla

July 2, 2015

Content-Signature Header Field for HTTP
draft-thomson-http-content-signature-00

Abstract

A Content-Signature header field is defined for use in HTTP. This header field carries a signature of the payload body of a message.

Status of This Memo

This Internet-Draft will expire on January 3, 2016.

Copyright Notice

Table of Contents

1. Introduction

The Content-Signature header field carries a signature of the payload body of an HTTP message [RFC7230]. This allows for content to be protected from modification.

The exchange of high-value messages via intermediaries is often necessary in HTTP for operational reasons. While those intermediaries might be trusted with the information that they forward, some clients or servers might desire greater assurances about the integrity of the information they receive.

No protection is provided for header fields. If integrity is important, only the information in the message payload can be relied upon.

No key management mechanism is defined. Other specifications are expected to describe how recipients determine what credibility is attributed to any given signature key.

1.1. Terminology

RFC 2119 [RFC2119] defines the terms MUST, SHOULD, and MAY.

1.2. Example

The following HTTP/1.1 response is signed. Line wrapping is added to fit formatting constraints.

```
HTTP/1.1 200 OK
Date: Wed, 17 Jun 2015 17:14:17 GMT
Content-Length: 15
Encryption-Key: keyid=a;
    p256ecdsa=BDUJCg0PKtFrgI_lc5ar9qBm83cH_QJomSjXYUkIlswX
              KTdYLlJjFEWlIThQ0Y-TFZyBbUinNp-rou13Wve_Y_A
Content-Signature: keyid=a;
    p256ecdsa=Hil-_2xU6BjQcU6a8nhMCChLr-fkrek5tE6pokWlJb0
              HkQiryW045vVpljN_xBbF8sTrsWb9MiQLCdYlP1jZtA

Hello, World!
```

2. The Content-Signature Header Field

The Content-Signature header field uses the extended ABNF syntax defined in Section 1.2 of [RFC7230] and the "parameter" rule from [RFC7231].

```
Content-Signature = 1#csig_params
csig_params = [ parameter *( ";" parameter ) ]
```

Each content signature is separated by a comma (,) and is compromised of zero or more colon-separated parameters.

The message payload is prefixed with the string "Content-Encryption:" and a single zero-valued octet before being passed to the signature algorithm. This discriminator string reduces the chances that a signature is viable for reuse in other contexts.

The following parameters are defined:

keyid: This parameter identifies the key that was used to produce the signature. This could identify a key that is carried in the Encryption-Key header field. This parameter can always be provided together with other parameters.

p256ecdsa: This parameter contains an ECDSA [X.692] signature on the P-256 curve [FIPS186]. The signature is produced using the SHA-256 hash [FIPS180-2]. The resulting signature is encoded using URL-safe variant of base-64 [RFC4648]. No parameters other than "keyid" can be specified along with the "p256ecdsa" parameter.

Additional header field values can be defined and registered. The parameter MUST describe how the signature is produced and encoded.

Though the parameter defined in this document do not contain any optional or parameterized features, new signature algorithms MAY use additional parameters for conveying information about optional features. The definition of new parameters SHOULD describe what parameters can be combined with that parameter and the resulting semantics.

The Content-Signature header field might be most efficiently produced as a trailer field. This allows for the production of the message body and the signature in a single pass.

3. Describing Signature Keys

A message MAY include a signing key. This can be used to provision trusted keys.

Providing an encryption key is typically only useful where the provision of the key can be attributed a higher level of trust than the signature. A message sent using out-of-band content-encoding [I-D.reschke-http-oob-encoding] is one situation that benefits from the use of this header field.

Alternatively, explicitly including a public key can allow a verifier to correctly identify the key that was used if the "keyid" parameter is not sufficient.

This document defines a new parameter for use with the "Encryption-Key" header field. The "p256ecdsa" parameter conveys an uncompressed P-256 public key [X.692] that is encoded using URL-safe variant of base-64 [RFC4648].

4. Security Considerations

Determining whether a signature is valid is only a small part of authenticating a message. This document doesn't describe a complete solution for identifying which signing keys are accepted.

This scheme does not authenticate header fields, or other request or response metadata. A recipient of a signed payload needs to be especially careful that decisions that rely on authenticating the payload do not take any unauthenticated material as input. In particular, the request URI and the "Content-Type" header field are not authenticated by this scheme.

No replay protection is offered for signatures. This means that valid messages can be captured and replayed. Since there is no binding between the identity of a resource and the signature, the content of a message can be replayed for a request or response to a different resource; requests can be replayed as responses; and messages can be replayed at different times.

Replay protection can be provided by including information in the message payload itself that binds the content to a specific resource, time or any other contextual information.

5. IANA Considerations

5.1. Content-Signature Header Field

This memo registers the "Content-Signature" HTTP header field in the Permanent Message Header Registry, as detailed in Section 2.

- Field name: Content-Signature

- Protocol: HTTP

- Status: Standard

- Reference: Section 2 of this specification

- Notes:

5.2. Content-Signature Parameter Registry

A registry is established for parameters used by the "Content-Signature" header field under the "Hypertext Transfer Protocol (HTTP) Parameters" grouping. The "Hypertext Transfer Protocol (HTTP) Encryption Parameters" operates under an "Specification Required" policy [RFC5226]. The designated expert is advised to consider the guidance in Section 2 when reviewing new registrations.

- Parameter Name: The name of the parameter.

- Purpose: A brief description of the purpose of the parameter.

- Reference: A reference to a specification that defines the semantics of the parameter.

The initial contents of this registry are:

5.2.1. keyid

- Parameter Name: keyid

- Purpose: Identify the key that is in use.

- Reference: Section 2 of this document

5.2.2. p256ecdsa

- Parameter Name: p256ecdsa

- Purpose: Conveys a signature using P-256, ECDSA and SHA-256 as described in Section 2 of this document.

- Reference: Section 2 of this document

5.3. The p256ecdsa Parameter for the Encryption-Key Header Field

The "p256ecdsa" parameter is registered in the "Hypertext Transfer Protocol (HTTP) Encryption Parameters" registry established in [I-D.thomson-http-encryption], with the following values:

- Parameter Name: p256ecdsa

- Purpose: Conveys a signing key for use with the parameter of the same name on the "Content-Signature" header field.

- Reference: Section 3 of this document

6. References

6.1. Normative References

[FIPS180-2]
Department of Commerce, National., "NIST FIPS 180-2, Secure Hash Standard", August 2002.

[FIPS186] National Institute of Standards and Technology (NIST), "Digital Signature Standard (DSS)", NIST PUB 186-4 , July 2013.

[I-D.thomson-http-encryption]
Thomson, M., "Encrypted Content-Encoding for HTTP", draft-thomson-http-encryption-01 (work in progress), July 2015.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, October 2006.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

[RFC7230] Fielding, R. and J. Reschke, "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", RFC 7230, June 2014.

[RFC7231] Fielding, R. and J. Reschke, "Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content", RFC 7231, June 2014.

[X.692] ANSI, "Public Key Cryptography For The Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)", ANSI X9.62 , 1998.

6.2. Informative References

[I-D.reschke-http-oob-encoding]
Reschke, J. and S. Loreto, "'Out-Of-Band' Content Coding for HTTP", draft-reschke-http-oob-encoding-00 (work in progress), June 2015.

Author's Address

Martin Thomson
Mozilla

Email: martin.thomson@gmail.com