Web Bot Auth
Internet-Draft
Intended status: Standards Track
Expires: 21 December 2025

T. Meunier
Cloudflare

19 June 2025

HTTP Message Signatures Directory
draft-meunier-http-message-signatures-directory-01

Abstract

This document describes a method for clients using [HTTP-MESSAGE-SIGNATURES] to advertise their signing keys.

It defines a key directory format based on JWKS as defined in Section 5 of [JWK], as well as new HTTP Method Context to allow for in-band key discovery.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://thibmeu.github.io/http-message-signatures-directory/draft-meunier-http-message-signatures-directory.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-meunier-http-message-signatures-directory/.

Discussion of this document takes place on the Web Bot Auth Working Group mailing list (mailto:web-bot-auth@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/web-bot-auth/. Subscribe at https://www.ietf.org/mailman/listinfo/web-bot-auth/.

Source for this draft and an issue tracker can be found at https://github.com/thibmeu/http-message-signatures-directory.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Copyright Notice

Table of Contents

1. Introduction

[HTTP-MESSAGE-SIGNATURES] allow a signer to generate a signature over an HTTP message, and a verifier to validate it. The specification assumes verifiers have prior knowledge of signers' key material, requiring out-of-band key distribution mechanisms. This creates deployment friction and limits the ability to dynamically verify signatures from previously unknown signers.

This document defines: 1. A standardized key directory format based on JWKS for publishing HTTP Message Signatures keys 2. A well-known URI location for discovering these key directories 3. A new HTTP header field enabling in-band key directory location discovery

Together, these mechanisms enable key distribution and discovery for HTTP Message Signatures cryptographic material.

2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Configuration

The key directory is served as a JSON Web Key Set (JWKS) as defined in Section 5 of [JWK]. The "alg" parameter are restricted to algorithm registered against HTTP Signature Algorithms Section of [HTTP-MESSAGE-SIGNATURES-IANA]

The directory SHOULD be served over HTTPS. The directory MUST be served with media type application/http-message-signatures-directory+json.

Client application SHOULD validate the directory format and reject malformed entries.

4. HTTP Method Context Signature-Agent

A service sending signed requests as defined in [HTTP-MESSAGE-SIGNATURES] MAY include a Signature-Agent header field to communicate its signing key directory. This header field contains a URI allowing retrieval of an HTTP Message Signatures Directory as defined in Section 3.

4.1. Header Field Definition

The Signature-Agent header field is an Item Structured Header [STRUCTURED-HEADERS]. Its value MUST be a String containing a [URI]. The ABNF is:

```
Signature-Agent = sf-string   ; Section 3.3.3 of {{STRUCTURED-HEADERS}}
```

The URI scheme MUST be one of: - *https (RECOMMENDED)*: Points to an HTTPS resource serving the key directory - *http*: Points to an HTTP resource serving the key directory - *data*: Contains an inline key directory

When using the "data" URI scheme, the media type MUST be application/http-message-signatures-directory+json. The content MAY be base64 encoded as per [BASE64].

Multiple Signature-Agent header fields MAY be present in a request. Processors SHOULD use the first valid URI that provides a valid key directory.

5. Security Considerations

5.1. Key rotation

Clients SHOULD implement key rotation by including multiple keys in the directory with different validity period. When rotating keys, clients SHOULD:

1. Add the new key to the directory before its intended use date

2. Continue to include the old key until its expiration date

3. Remove expired keys from the directory

Servers SHOULD cache the directory contents and refresh upon expiration.

5.2. Binding keys to the directory authority

To ensure the authenticity and integrity of the key material provided by the directory, clients *SHOULD* validate the directory’s response.

When a directory server provides a key directory over HTTP or HTTPS, it is RECOMMENDED that it constructs and includes one HTTP Message Signatures per keys with the response, as defined in [HTTP-MESSAGE-SIGNATURES]. Each key SHOULD be used to provide one signature.

Directory server SHOULD include:

@authority as defined in Section 2.2.3 of [HTTP-MESSAGE-SIGNATURES]

Directory server SHOULD include the following @signature-params as defined in Section 2.3 of [HTTP-MESSAGE-SIGNATURES]

created as defined in Section 2.3 of [HTTP-MESSAGE-SIGNATURES]

expires as defined in Section 2.3 of [HTTP-MESSAGE-SIGNATURES]

keyid MUST be a base64url JWK SHA-256 Thumbprint as defined in Section 3.2 of [JWK-THUMBPRINT] for RSA and EC, and in Appendix A.3 of [JWK-OKP] for ed25519.

tag MUST be http-message-signatures-directory

Clients SHOULD validate these signatures using the keys provided by the directory. Clients SHOULD ignore keys from a directory response that do not have a corresponding valid signature. This validation ensures the integrity of the key set and its association with the intended directory.

6. Privacy Considerations

Key directories enable discovery of signing keys which may reveal information about the signing entity. Implementers should consider:

6.1. Directory Content

Key directories should only contain keys actively used for signing. Including additional keys or metadata may expose unnecessary information about the signing service.

6.2. Access Patterns

Verifiers accessing key directories may reveal information about signature verification patterns. Directory servers should avoid logging personally identifiable information from directory requests.

7. IANA Considerations

This section contains considerations for IANA.

7.1. Well-Known 'http-message-signatures-directory' URI

This document updates the "Well-Known URIs" Registry [WellKnownURIs] with the following values.

URI Suffix	Change Controller	Reference	Status	Related information
http-message-signatures-directory	IETF	[this document]	permanent	None

Table 1: 'http-message-signatures-directory' Well-Known URI

7.2. Media Types

The following entries should be added to the IANA "media types" registry:

* "application/http-message-signatures-directory+json"

The templates for these entries are listed below and the reference should be this RFC.

7.2.1. "application/http-message-signatures-directory+json" media type

Type name: application
Subtype name: http-message-signatures-directory
Required parameters: N/A
Optional parameters: N/A
Encoding considerations: "binary"
Security considerations: see Section 5
Interoperability considerations: N/A
Published specification: this specification
Applications that use this media type: Services that implement the

signer role for HTTP Message Signatures and verifiers that interact with the signer for the purpose of validating signatures.
Fragment identifier considerations: N/A
Additional information: Magic number(s): N/A
Deprecated alias names for this type: N/A
File extension(s): N/A
Macintosh file type code(s): N/A
Person and email address to contact for further information: see Authors' Addresses section
Intended usage: COMMON
Restrictions on usage: N/A
Author: see Authors' Addresses section
Change controller: IETF

8. References

8.1. Normative References

[HTTP] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "HTTP Semantics", STD 97, RFC 9110, DOI 10.17487/RFC9110, June 2022, <https://www.rfc-editor.org/rfc/rfc9110>.

[HTTP-MESSAGE-SIGNATURES] Backman, A., Ed., Richer, J., Ed., and M. Sporny, "HTTP Message Signatures", RFC 9421, DOI 10.17487/RFC9421, February 2024, <https://www.rfc-editor.org/rfc/rfc9421>.

[HTTP-MESSAGE-SIGNATURES-IANA] "HTTP Message Signatures", n.d., <https://www.iana.org/assignments/http-message-signature/http-message-signature.xhtml>.

[JWK] Jones, M., "JSON Web Key (JWK)", RFC 7517, DOI 10.17487/RFC7517, May 2015, <https://www.rfc-editor.org/rfc/rfc7517>.

[JWK-OKP] Liusvaara, I., "CFRG Elliptic Curve Diffie-Hellman (ECDH) and Signatures in JSON Object Signing and Encryption (JOSE)", RFC 8037, DOI 10.17487/RFC8037, January 2017, <https://www.rfc-editor.org/rfc/rfc8037>.

[JWK-THUMBPRINT] Jones, M. and N. Sakimura, "JSON Web Key (JWK) Thumbprint", RFC 7638, DOI 10.17487/RFC7638, September 2015, <https://www.rfc-editor.org/rfc/rfc7638>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[STRUCTURED-HEADERS] Nottingham, M. and P. Kamp, "Structured Field Values for HTTP", RFC 8941, DOI 10.17487/RFC8941, February 2021, <https://www.rfc-editor.org/rfc/rfc8941>.

[URI] Nottingham, M., "URI Design and Ownership", BCP 190, RFC 8820, DOI 10.17487/RFC8820, June 2020, <https://www.rfc-editor.org/rfc/rfc8820>.

[WellKnownURIs] "Well-Known URIs", n.d., <https://www.iana.org/assignments/well-known-uris/well-known-uris.xhtml>.

8.2. Informative References

[BASE64] Masinter, L., "The "data" URL scheme", RFC 2397, DOI 10.17487/RFC2397, August 1998, <https://www.rfc-editor.org/rfc/rfc2397>.

[CRYPTO-TEST-KEYS] Gutmann, P. and C. Bonnell, "Standard Public Key Cryptography (PKC) Test Keys", RFC 9500, DOI 10.17487/RFC9500, December 2023, <https://www.rfc-editor.org/rfc/rfc9500>.

[X509-PKI] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/rfc/rfc5280>.

Appendix A. Examples

A.1. Key Directory on example.com

```
GET /.well-known/http-message-signatures-directory HTTP/1.1
Host: example.com
Accept: application/http-message-signatures-directory+json

HTTP/1.1 200 OK
Content-Type: application/http-message-signatures-directory+json
Cache-Control: max-age=86400
{
  "keys": [{
    "kty": "OKP",
    "crv": "Ed25519",
    "kid": "NFcWBst6DXG-N35nHdzMrioWntdzNZghQSkjHNMMSjw",
    "x": "JrQLj5P_89iXES9-vFgrIy29clF9CC_oPPsw3c5D0bs",
    "use": "sig",
    "nbf": 1712793600,
    "exp": 1715385600
  }]
}
```

A.2. Delegation and chaining

There are multiple methods to perform delegation and chaining. There are no specific methods that have been favored by implementation so far, should they even support them. It is adviced to consider delegation as experimental for now, and provide input on the associated GitHub issue (https://github.com/thibmeu/http-message-signatures-directory/issues/27).

A.2.1. Key Directory on sub.example.com with a delegation from example.com via x5c full certificate chain

In this example, example.com key is testECCP256 provided in Section 2.3 of [CRYPTO-TEST-KEYS]. Certificate chain is passed via x5c key parameter defined in Section 4.7 of [JWK].

```
GET /.well-known/http-message-signatures-directory HTTP/1.1
Host: sub.example.com
Accept: application/http-message-signatures-directory

HTTP/1.1 200 OK
Content-Type: application/http-message-signatures-directory
Cache-Control: max-age=86400
{
  "keys": [{
    "kty": "OKP",
    "crv": "Ed25519",
    "kid": "NFcWBst6DXG-N35nHdzMrioWntdzNZghQSkjHNMMSjw",
    "x": "JrQLj5P_89iXES9-vFgrIy29clF9CC_oPPsw3c5D0bs",
    "use": "sig",
    "nbf": 1712793600,
    "exp": 1715385600,
    "x5c": [
      "MIIBYTCCAQagAwIBAgIUFDXRG3pgZ6txehQO2LT4aCqI3f0wCgYIKoZIzj0EAwIwFjEUMBIGA1UEAwwLZX
hhbXBsZS5jb20wHhcNMjUwNjEzMTA0MjQxWhcNMzUwNjExMTA0MjQxWjAaMRgwFgYDVQQDDA9zdWIuZXhhbXBsZS5
jb20wKjAFBgMrZXADIQAmtAuPk//z2JcRL368WCsjLb1yUX0IL+g8+zDdzkPRu6NdMFswCQYDVR0TBAIwADAOBgNV
HQ8BAf8EBAMCB4AwHQYDVR0OBBYEFKV3qaYNFbzQB1QmN4sa13+t4RmoMB8GA1UdIwQYMBaAFFtwp5gX95/2N9L34
9xEbCEJ17vUMAoGCCqGSM49BAMCA0kAMEYCIQC8r+GvvNnjI+zzOEDMOM/g9e8QLm00IZXP+tjDqah1UQIhAJHffL
ke9iEP1pUdm+oRLrq6bUqyLELi5TH2t+BaagKv",
      "MIIBcDCCARagAwIBAgIUS502rlCXxG2vviltGdfe3fmX4pIwCgYIKoZIzj0EAwIwFjEUMBIGA1UEAwwLZX
hhbXBsZS5jb20wHhcNMjUwNjEzMTA0MTQzWhcNMzUwNjExMTA0MTQzWjAWMRQwEgYDVQQDDAtleGFtcGxlLmNvbTB
ZMBMGByqGSM49AgEGCCqGSM49AwEHA0IABEIlSPiPt4L/teyjdERSxyoeVY+9b3O+XkjpMjLMRcWxbEzRDEy41bih
cTnpSILImSVymTQl9BQZq36QpCpJQnKjQjBAMA8GA1UdEwEB/wQFMAMBAf8wDgYDVR0PAQH/BAQDAgIEMB0GA1UdD
gQWBBRbcKeYF/ef9jfS9+PcRGwhCde71DAKBggqhkjOPQQDAgNIADBFAiEAwTOqm1zNAvZuQ8Zb5AftQIZotq4Xe6
GHz3+nJ04ybgoCIEEZtn1Pa+GCbmbWh12piHJBKh09TCA0feTedisbwzPV"
    ]
  }]
}
```

A.2.2. Key Directory on sub.example.com with a delegation from example.com via a leaf certificate and AIA field

In this example, example.com key is testECCP256 provided in Section 2.3 of [CRYPTO-TEST-KEYS]. Certificate chain is passed via x5c key parameter defined in Section 4.7 of [JWK], and the root certificate is signaled by the presence of an Authority Information Access extension as defined in Section 5.2.7 of [X509-PKI].

```
GET /.well-known/http-message-signatures-directory HTTP/1.1
Host: sub.example.com
Accept: application/http-message-signatures-directory

HTTP/1.1 200 OK
Content-Type: application/http-message-signatures-directory
Cache-Control: max-age=86400
{
  "keys": [{
    "kty": "OKP",
    "crv": "Ed25519",
    "kid": "NFcWBst6DXG-N35nHdzMrioWntdzNZghQSkjHNMMSjw",
    "x": "JrQLj5P_89iXES9-vFgrIy29clF9CC_oPPsw3c5D0bs",
    "use": "sig",
    "nbf": 1712793600,
    "exp": 1715385600,
    "x5c": [
      "MIIBYTCCAQagAwIBAgIUFDXRG3pgZ6txehQO2LT4aCqI3f0wCgYIKoZIzj0EAwIwFjEUMBIGA1UEAwwLZX
hhbXBsZS5jb20wHhcNMjUwNjEzMTA0MjQxWhcNMzUwNjExMTA0MjQxWjAaMRgwFgYDVQQDDA9zdWIuZXhhbXBsZS5
jb20wKjAFBgMrZXADIQAmtAuPk//z2JcRL368WCsjLb1yUX0IL+g8+zDdzkPRu6NdMFswCQYDVR0TBAIwADAOBgNV
HQ8BAf8EBAMCB4AwHQYDVR0OBBYEFKV3qaYNFbzQB1QmN4sa13+t4RmoMB8GA1UdIwQYMBaAFFtwp5gX95/2N9L34
9xEbCEJ17vUMAoGCCqGSM49BAMCA0kAMEYCIQC8r+GvvNnjI+zzOEDMOM/g9e8QLm00IZXP+tjDqah1UQIhAJHffL
ke9iEP1pUdm+oRLrq6bUqyLELi5TH2t+BaagKv"
    ]
  }]
}
```

The AIA extension is as follow

```
X509v3 extensions:
  Authority Information Access:
    CA Issuers - URI:https://example.com/.well-known/http-message-signatures-directory.cr
t
```

The verifier should validate the signature with the public key in the Signature-Agent, match the public key with the leaf cert, then fetch the root cert from the AIA URI and verify the leaf cert with it.

A.2.3. Key Directory on sub.example.com with a delegation from example.com via x5u field

Leveraging x5c imposes that a PEM encoded certificate is present in the returned JWKS. If size is a constraint, or deployment imposes a more dynamic certificate management, directory server may use x5u key parameter defined in Section 4.6 of [JWK].

```
GET /.well-known/http-message-signatures-directory HTTP/1.1
Host: sub.example.com
Accept: application/http-message-signatures-directory

HTTP/1.1 200 OK
Content-Type: application/http-message-signatures-directory
Cache-Control: max-age=86400

{
  "keys": [{
    "kty": "OKP",
    "crv": "Ed25519",
    "kid": "NFcWBst6DXG-N35nHdzMrioWntdzNZghQSkjHNMMSjw",
    "x": "JrQLj5P_89iXES9-vFgrIy29clF9CC_oPPsw3c5D0bs",
    "use": "sig",
    "nbf": 1712793600,
    "exp": 1715385600,
    "x5u": "https://example.com/.well-known/http-message-signature-chain/sub.example.com.
crt"
}
```

A.3. Request with HTTP Signature-Agent

This extend the examples from Appendix B of [HTTP-MESSAGE-SIGNATURES].

```
POST /foo?param=Value&Pet=dog HTTP/1.1
Host: example.com
Signature-Agent: https://directory.test
{"hello": "world"}

HTTP/1.1 200 OK
{"message": "good dog"}
```

A.4. Request with data URI Signature-Agent

A Signature-Agent using data URI can be used to communicate an ephemeral keys, as long as there is a chain to a certificate trusted by the origin.

In this example, the directory is signed by example.com. The CA is self-signed, even though it MAY be part of an existing PKI.

```
POST /foo?param=Value&Pet=dog HTTP/1.1
Host: example.com
Signature-Agent: data:application/http-message-signatures-directory;utf8,{"keys":[{"kty":
"OKP","crv":"Ed25519","kid":"NFcWBst6DXG-N35nHdzMrioWntdzNZghQSkjHNMMSjw","x":"JrQLj5P_89
iXES9-vFgrIy29clF9CC_oPPsw3c5D0bs","use":"sig","nbf":1712793600,"exp":1715385600,"x5c":["
MIIBYTCCAQagAwIBAgIUFDXRG3pgZ6txehQO2LT4aCqI3f0wCgYIKoZIzj0EAwIwFjEUMBIGA1UEAwwLZXhhbXBsZ
S5jb20wHhcNMjUwNjEzMTA0MjQxWhcNMzUwNjExMTA0MjQxWjAaMRgwFgYDVQQDDA9zdWIuZXhhbXBsZS5jb20wKj
AFBgMrZXADIQAmtAuPk//z2JcRL368WCsjLb1yUX0IL+g8+zDdzkPRu6NdMFswCQYDVR0TBAIwADAOBgNVHQ8BAf8
EBAMCB4AwHQYDVR0OBBYEFKV3qaYNFbzQB1QmN4sa13+t4RmoMB8GA1UdIwQYMBaAFFtwp5gX95/2N9L349xEbCEJ
17vUMAoGCCqGSM49BAMCA0kAMEYCIQC8r+GvvNnjI+zzOEDMOM/g9e8QLm00IZXP+tjDqah1UQIhAJHffLke9iEP1
pUdm+oRLrq6bUqyLELi5TH2t+BaagKv","MIIBcDCCARagAwIBAgIUS502rlCXxG2vviltGdfe3fmX4pIwCgYIKoZ
Izj0EAwIwFjEUMBIGA1UEAwwLZXhhbXBsZS5jb20wHhcNMjUwNjEzMTA0MTQzWhcNMzUwNjExMTA0MTQzWjAWMRQw
EgYDVQQDDAtleGFtcGxlLmNvbTBZMBMGByqGSM49AgEGCCqGSM49AwEHA0IABEIlSPiPt4L/teyjdERSxyoeVY+9b
3O+XkjpMjLMRcWxbEzRDEy41bihcTnpSILImSVymTQl9BQZq36QpCpJQnKjQjBAMA8GA1UdEwEB/wQFMAMBAf8wDg
YDVR0PAQH/BAQDAgIEMB0GA1UdDgQWBBRbcKeYF/ef9jfS9+PcRGwhCde71DAKBggqhkjOPQQDAgNIADBFAiEAwTO
qm1zNAvZuQ8Zb5AftQIZotq4Xe6GHz3+nJ04ybgoCIEEZtn1Pa+GCbmbWh12piHJBKh09TCA0feTedisbwzPV"]}]
}
{"hello": "world"}

HTTP/1.1 200 OK
{"message": "good dog"}
```

Acknowledgments

TODO acknowledge.

Changelog

v01

- Update content-type from application/http-message-signatures-directory to application/http-message-signatures-directory+json
- Add delegation and chaining examples: full x5c chain, AIA extension, and x5u
- Add inline directory example with data URI
- Fix well-known path in examples

v00

- Initial draft
- Definition of Signature-Agent and its three supported URI https, http, and data.
- Leverages JWKS as a directory fo HTTP Message Signatures
- Well-known and content-type

Author’s Address

Thibault Meunier
Cloudflare
Email: ot-ietf@thibault.uk