OAuth Working Group
Internet Draft
Intended status: Proposed Standard
Expires: August 9, 2022

D.W.Chadwick
Crossword Cybersecurity

February 9, 2022

JWT URI

draft-chadwick-oauth-jwk-uri-00.txt

Status of This Memo
This is an Internet Standards Track document.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups.
Note that other groups may also distribute
working documents as Internet-Drafts.

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.
The list of current Internet-Drafts can be accessed at
https://www.ietf.org/1id-abstracts.html

The list of Internet-Draft Shadow Directories can be accessed at
https://www.ietf.org/shadow.html

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 9, 2022.

Copyright Notice

Abstract
This specification registers a kind of URI that represents a JSON

Web Key (JWK) value. This enables JWKs to be used, for instance, as key identifiers in contexts requiring URIs.

Table of Contents

1. Introduction

A JSON Web Key (JWK) [RFC7517] is a JavaScript Object Notation (JSON) data structure that represents a cryptographic key.
This specification defines a URI prefix indicating that the portion of the URI following the prefix is a JWK. This enables JWKs to be communicated in contexts requiring URIs, including in specific JSON Web Token (JWT) [RFC7519] claims.
JWK URIs are proposed to be used in the [SIOPv2] specification as one kind of subject identifier in a context requiring that the identifier be a URI. In this case, the subject identifier is derived from a public key represented as a JWK. Expressing the identifier as a JWK URI enables this kind of identifier to be differentiated from other kinds of identifiers that are also URIs, such as Decentralized Identifiers (DIDs) [DID-Core].

---

2. Requirements Notation and Conventions

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. JWK URI

The following URI prefix is defined to indicate that the portion of

the URI following the prefix is a JWK:

    o "urn:ietf:params:oauth:jwk"

The prefix MUST be followed by a colon and a JWK value that is formed by performing a base64 encoding of the JWK to form a URI representing a JWK.

4. Comparison of JWK URIs with JWK Thumbprint URIs

To produce or validate a JWK Thumbprint, both the sender and the receiver have to have the JWK available to them. Then they have to canonicalise the JWK as described in [RFC7638], and finally hash the octets of the UTF-8 representation of this JSON object with a pre-agreed algorithm in order to both obtain the same hash value. The way that the JWK Thumbprint URI is used in SIOPv2 [SIOPv2] is as follows:

1. the SIOP creates an asymmetric key pair and encodes the public key as a JWK
2. the SIOP creates the JWK Thumbprint as described in [RFC7638] and converts it to a URI as described in [JONES]
3. the SIOP passes both the JWK and JWK Thumbprint URI to the RP in the JWT
4. the RP extracts the JWK and JWK Thumbprint from the JWT
5. the RP re-computes the JWK Thumbprint from the JWK
6. the RP compares the computed JWK Thumbprint with the received JWK Thumbprint to confirm that they are equal.

One can see that the use of JWK Thumbprint URIs is both inefficient (in all cases) and a significant disadvantage (in some cases). If the JWK URI (as described in this document) is transferred instead of the JWK and JWK Thumbprint URI then:
a) The SIOP will never need to create the JWK Thumbprint URI. The RP may only need to create the JWK Thumbprint if it needs this, for example, as a unique subject identifier. Even in this case, there is still an advantage to the RP in receiving the JWK URI instead of the JWK Thumbprint URI, in that the RP no longer needs to pre-agree a hashing algorithm with the SIOP. Thus the RP can independently determine which hashing algorithm to use when creating its own JWK Thumbprint.
(Note. If the SIOP were able to canonicalise the same public key in a JWK in different ways and produce different thumbprints from the same public key, then the canonicalisation

algorithm is broken, and the RP would never to able to deterministically produce the same thumbprints each time.)

b) In those cases where the SIOP uses ephemeral key pairs and a different public key each time it communicates with an RP, then neither party needs to produce the JWK Thumbprint as it will never be seen again. It is a significant disadvantage to have to use JWK Thumbprints in this case.
One possible disadvantage of using JWK URIs instead of JWK Thumbprint URIs is the resulting increase in size of the JWT. Base 64 encoding a JWK string increases its size by 33%. However this increase in JWT size is offset by the decrease in size by not needing to include the JWT thumbprint URI as well as the JWK. The trade off is the processing of JWKs to produce thumbprints by the sender and receiver versus the overhead of transferring larger JWTs.

5. Security Considerations

The security considerations of [RFC7638] will apply when the RP is using [RFC7638] to produce thumbprints.

6. IANA Considerations

6.1. OAuth URI Registration

This specification proposes to register the following value in the IANA "OAuth URI" registry [IANA.OAuth.Parameters] established by [RFC6755].

6.1.1. Registry Contents

- URN: urn:ietf:params:oauth:jwk
- Common Name: JWK URI
- Change controller: IESG
- Specification Document: [[this specification]]

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
[IANA.OAuth.Parameters] IANA, "OAuth Parameters", <http://www.iana.org/assignments/oauth-parameters>.
[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.rfceditor.org/info/rfc2119>.
[RFC7638] Jones, M. and N. Sakimura, "JSON Web Key (JWK)

Thumbprint", RFC 7638, DOI 10.17487/RFC7638, September 2015, <https://www.rfc-editor.org/info/rfc7638>.
[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/info/rfc8174>.

7.2. Informative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
[DID-Core] Sporny, M., Guy, A., Sabadello, M., and D. Reed, "Decentralized Identifiers (DIDs) v1.0", Aug 2021, <https://www.w3.org/TR/2021/PR-did-core-20210803/>.
[RFC6755] Campbell, B. and H. Tschofenig, "An IETF URN Sub-Namespace for OAuth", RFC 6755, DOI 10.17487/RFC6755, October 2012, <https://www.rfc-editor.org/info/rfc6755>.
[RFC7517] Jones, M., "JSON Web Key (JWK)", RFC 7517, DOI 10.17487/RFC7517, May 2015, <https://www.rfceditor.org/info/rfc7517>.
[RFC7519] Jones, M., Bradley, J., and N. Sakimura, "JSON Web Token (JWT)", RFC 7519, DOI 10.17487/RFC7519, May 2015, <https://www.rfc-editor.org/info/rfc7519>.
[SIOPv2] Yasuda, K. and M. B. Jones, "Self-Issued OpenID Provider v2", December 2021, <https://openid.net/specs/openidconnect-self-issued-v2-1_0.html>.

[JONES] Yasuda, K., Jones, M., "JWK Thumbprint URK", Internet Draft draft-ietf-oauth-jwk-thumbprint-uri-00

8. Acknowledgments

to be done.

Appendix A. Document History

[[to be removed by the RFC Editor before publication as an RFC]]
-00
o Created initial draft.

Authors' Addresses
David W Chadwick
Crossword Cybersecurity
Email: david.chadwick@crosswordcybersecurity.com
URI: https://www.linkedin.com/in/davidwchadwick/