

JOSE Working Group	M. Jones
Internet-Draft	Microsoft
Intended status: Standards Track	November 6, 2012
Expires: May 10, 2013	

JSON Private Key draft-jones-jose-json-private-key-01

Abstract

The JSON Private Key specification extends the JSON Web Key (JWK) and JSON Web Algorithms (JWA) specifications to define a JavaScript Object Notation (JSON) representation of private keys.

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 <http://datatracker.ietf.org/drafts/current/>.

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 May 10, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- 1. Introduction**
 - 1.1. Notational Conventions**
- 2. Terminology**
- 3. JWK Parameters for Private Keys**
 - 3.1. JWK Parameters for Elliptic Curve Private Keys**
 - 3.1.1. "d" (ECC Private Key) Parameter**
 - 3.2. JWK Parameters for RSA Private Keys**
 - 3.2.1. "d" (Private Exponent) Parameter**
 - 3.2.2. "p" (First Prime Factor) Parameter**
 - 3.2.3. "q" (Second Prime Factor) Parameter**
 - 3.2.4. "dp" (First Factor CRT Exponent) Parameter**
 - 3.2.5. "dq" (Second Factor CRT Exponent) Parameter**
 - 3.2.6. "qi" (First CRT Coefficient) Parameter**
 - 3.2.7. "oth" (Other Primes Info) Parameter**
 - 3.2.7.1. "r" (Prime Factor)**
 - 3.2.7.2. "d" (Factor CRT Exponent)**

- [3.2.7.3. "t" \(Factor CRT Coefficient\)](#)
- [4. Example Private Keys](#)
- [5. IANA Considerations](#)
 - [5.1. JSON Web Key Parameters Registration](#)
 - [5.1.1. Registry Contents](#)
- [6. Security Considerations](#)
- [7. Normative References](#)
- [Appendix A. Document History](#)
- [§ Author's Address](#)

1. Introduction

TOC

The JSON Private Key specification extends the JSON Web Key (JWK) [\[JWK\]](#) and JSON Web Algorithms (JWA) [\[JWA\]](#) specifications to define a JavaScript Object Notation (JSON) [\[RFC4627\]](#) representation of private keys.

1.1. Notational Conventions

TOC

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in Key words for use in RFCs to Indicate Requirement Levels [\[RFC2119\]](#).

2. Terminology

TOC

This specification uses the same terminology as the JSON Web Key (JWK) [\[JWK\]](#) and JSON Web Algorithms (JWA) [\[JWA\]](#) specifications.

3. JWK Parameters for Private Keys

TOC

This section defines additional JSON Web Key parameters that enable JWKs to represent private keys.

3.1. JWK Parameters for Elliptic Curve Private Keys

TOC

When the JWK `alg` member value is `EC`, the following member MAY be used to represent an Elliptic Curve private key:

3.1.1. "d" (ECC Private Key) Parameter

TOC

The `d` (ECC private key) member contains the Elliptic Curve private key value. It is represented as the base64url encoding of the value's unsigned big endian representation as a byte array. The array representation MUST not be shortened to omit any leading zero bytes. For instance, when representing 521 bit integers, the byte array to be base64url encoded MUST contain 66 bytes, including any leading zero bytes.

3.2. JWK Parameters for RSA Private Keys

TOC

When the `JWK alg` member value is `RSA`, the following member MAY be used to represent an RSA private key:

3.2.1. "d" (Private Exponent) Parameter

TOC

The `d` (private exponent) member contains the private exponent value for the RSA private key. It is represented as the base64url encoding of the value's unsigned big endian representation as a byte array. The array representation **MUST** not be shortened to omit any leading zero bytes. For instance, when representing 2048 bit integers, the byte array to be base64url encoded **MUST** contain 256 bytes, including any leading zero bytes.

3.2.2. "p" (First Prime Factor) Parameter

TOC

The `p` (first prime factor) member contains the first prime factor, a positive integer. It is represented as the base64url encoding of the value's unsigned big endian representation as a byte array.

3.2.3. "q" (Second Prime Factor) Parameter

TOC

The `q` (second prime factor) member contains the second prime factor, a positive integer. It is represented as the base64url encoding of the value's unsigned big endian representation as a byte array.

3.2.4. "dp" (First Factor CRT Exponent) Parameter

TOC

The `dp` (first factor CRT exponent) member contains the Chinese Remainder Theorem (CRT) exponent of the first factor, a positive integer. It is represented as the base64url encoding of the value's unsigned big endian representation as a byte array.

3.2.5. "dq" (Second Factor CRT Exponent) Parameter

TOC

The `dq` (second factor CRT exponent) member contains the Chinese Remainder Theorem (CRT) exponent of the second factor, a positive integer. It is represented as the base64url encoding of the value's unsigned big endian representation as a byte array.

3.2.6. "qi" (First CRT Coefficient) Parameter

TOC

The `qi` (first CRT coefficient) member contains the Chinese Remainder Theorem (CRT) coefficient of the second factor, a positive integer. It is represented as the base64url encoding of the value's unsigned big endian representation as a byte array.

3.2.7. "oth" (Other Primes Info) Parameter

TOC

The `oth` (other primes info) member contains an array of information about any third and subsequent primes, should they exist. When only two primes have been used (the normal case), this parameter **MUST** be omitted. When three or more primes have been used, the number of array elements **MUST** be the number of primes used minus two. Each array element **MUST** be an object with the following members:

3.2.7.1. "r" (Prime Factor)

The `r` (prime factor) parameter within an `oth` array member represents the value of a subsequent prime factor, a positive integer. It is represented as the base64url encoding of the value's unsigned big endian representation as a byte array.

3.2.7.2. "d" (Factor CRT Exponent)

The `d` (Factor CRT Exponent) parameter within an `oth` array member represents the CRT exponent of the corresponding prime factor, a positive integer. It is represented as the base64url encoding of the value's unsigned big endian representation as a byte array.

3.2.7.3. "t" (Factor CRT Coefficient)

The `t` (factor CRT coefficient) parameter within an `oth` array member represents the CRT coefficient of the corresponding prime factor, a positive integer. It is represented as the base64url encoding of the value's unsigned big endian representation as a byte array.

4. Example Private Keys

The following example JWK Set contains two keys represented as JWKs containing both public and private key values: one using an Elliptic Curve algorithm and a second one using an RSA algorithm. This example extends the example in Section 3 of [\[JWK\]](#), adding private key values. (Line breaks are for display purposes only.)

```
{ "keys":
  [
    { "alg": "EC",
      "crv": "P-256",
      "x": "MKBCTNIcKUSDii11ySs3526iDZ8AiTo7Tu6KPAqv7D4",
      "y": "4Et16SRW2YiLUrN5vfvVHuhp7x8PxltmWwLbbM4IFyM",
      "d": "870MB6gfuTJ4HtUnUvYMyJpr5eUZNP4Bk43bVdj3eAE",
      "use": "enc",
      "kid": "1"},

    { "alg": "RSA",
      "n": "0vx7agoebGcQSuuPiLJXZptN9nndrQmbXEps2aiAFbWhM78Lhwx4
cbbfAAAtVT86zWu1RK7aPFFxuhDR1L6tSoc_BJECPEbWKRXjBZCiFV4n3oknjhMst
n64tZ_2W-5JsGY4Hc5n9yBXArwl93lqt7_RN5w6Cf0h4QyQ5v-65YGjQR0_FDW2Q
vzqY368QQMicAtaSqzs8KJZgnYb9c7d0zgdAZHzu6QMqvRL5hajrn1n91Cb0pbIS
D08qNLYrckt-bFTwhAI4vMQFh6WeZu0fM4lFd2NcRwr3XPksINHaQ-G_xBniIqbw
0Ls1jF44-csFCur-kEgU8awapJzKnqDKgw",
      "e": "AQAB",
      "d": "X4cTteJY_gn4FYPsXB8rdXix5vwsG1FLN5E3EaG6RJoVH-HLLKD9
M7dx5oo7GURknchnrRweUkC7hT5fJLM0WbFAKNLWY2vv7B6NqXSzUvxT0_Ysfqij
wp3RTz1BaCxwp4doFk5N2o8Gy_nHNKroADiKJ46pRUoHsXywbReAdYaMwFs9tv8d
_cPVY3i07a3t8MN6TNwm0dSawm9v47UicL3Sk5ZiG7xojPLu4sbG1U2jx4IBTNBz
nbJSzFHK66jT8bgkuqsk0GjSkDJk19Z4qjwbsnn4j2WBii3RL-Us2lGVky8fkFz
me1z0HbIkfz0Y6mqn0Ytqc0X4jfcKoAC8Q",
      "p": "83i-7IvMGXoMXCskv73TKr8637Fi07Z27zv8oj6pbWUQyLPQBQxtPV
nwd20R-60eTDmD2ujnMt5PqMrm8RfmNhVWdtjjMmCMjOpSXicFHj7XOuVIYQyqV
WlWEh6dN36GVZYk93N8Bc9vY41xy8B9Rzz0GVQzXvNEvn700nVbfs",
      "q": "3dfOR9cuYq-0S-mkFLZgItgMEffzB2q3hWehMuG0oCuqnb3vobLyum
qjVZQ01dIrdwgTncdpYzBcOfW5r370AFXjiWft_NGEiovonizhKpo9VVS78TzFgx
kIdrecRezsZ-1kYd_s1qDbxtkDEgfAITAG9LUnADun4vIcb6yelxk",
```

```
"dp": "G4sPXkc6Ya9y8oJw9_ILj4xuppu0lzi_H7VTkS8xj5SdX3coE0oim
YwxIi2emTAue0U0a5dpgFGyBJ4c8tQ2VF402XRugKDTP8akYhFo5tAA77Qe_Nmtu
YZc3C3m3I24G2GvR5sSDxUyAN2zq8Lfn9EUms6rY30b8YeiKkTiBj0",
"dq": "s9lAH9fggBsoFR80ac2R_E2gw282rT2kG0AhvI1lETE1efrA6huUU
vMFbCmpn8lqew6vzznYY5SSQF7pMdC_agI3nG8Ibp1BUb0JUiraRNqUfLhcQb_d9
GF4Dh7e74WbRsobRonujTYN1xCaP6T061jvWrX-L18txXw494Q_cgk",
"qi": "GyM_p6JrXySiz1toFgKbWV-JdI3jQ4ypu9rbMwx3rQJBfmt0FoYzg
UIZEVFec0qwemRN81zoDAaa-Bk0KWNGDjJHZDdDmFhw3AN71I-puxk_mHZGJ11rx
yR8055XLSe3SPmRfKwZI6yU24ZxvQKFYItldldUKGz06Ia6zTKhAVRU",
"kid": "2011-04-29"}
]
}
```

5. IANA Considerations

TOC

5.1. JSON Web Key Parameters Registration

TOC

This specification registers the parameter names defined in [Section 3.1](#) and [Section 3.2](#) in the IANA JSON Web Key Parameters registry [\[JWK\]](#).

5.1.1. Registry Contents

TOC

- Parameter Name: [d](#)
- Change Controller: IETF
- Specification Document(s): [Section 3.1.1](#) of [\[\[this document \]\]](#)
- Parameter Name: [d](#)
- Change Controller: IETF
- Specification Document(s): [Section 3.2.1](#) of [\[\[this document \]\]](#)
- Parameter Name: [p](#)
- Change Controller: IETF
- Specification Document(s): [Section 3.2.2](#) of [\[\[this document \]\]](#)
- Parameter Name: [q](#)
- Change Controller: IETF
- Specification Document(s): [Section 3.2.3](#) of [\[\[this document \]\]](#)
- Parameter Name: [dp](#)
- Change Controller: IETF
- Specification Document(s): [Section 3.2.4](#) of [\[\[this document \]\]](#)
- Parameter Name: [dq](#)
- Change Controller: IETF
- Specification Document(s): [Section 3.2.5](#) of [\[\[this document \]\]](#)
- Parameter Name: [qi](#)
- Change Controller: IETF
- Specification Document(s): [Section 3.2.6](#) of [\[\[this document \]\]](#)
- Parameter Name: [oth](#)
- Change Controller: IETF
- Specification Document(s): [Section 3.2.7](#) of [\[\[this document \]\]](#)

6. Security Considerations

TOC

The security considerations for this specification are the same as those for the JSON Web Key

(JWK) **[JWK]** specification and the portion of the JSON Web Algorithms (JWA) **[JWA]** specification that pertains to key representations.

7. Normative References

TOC

- [JWA]** **Jones, M.**, "**JSON Web Algorithms (JWA)**," November 2012.
- [JWK]** **Jones, M.**, "**JSON Web Key (JWK)**," November 2012.
- [RFC2119]** **Bradner, S.**, "**Key words for use in RFCs to Indicate Requirement Levels**," BCP 14, RFC 2119, March 1997 (**TXT**, **HTML**, **XML**).
- [RFC4627]** Crockford, D., "**The application/json Media Type for JavaScript Object Notation (JSON)**," RFC 4627, July 2006 (**TXT**).
-

Appendix A. Document History

TOC

[[to be removed by the RFC editor before publication as an RFC]]

-01

- Changed the names of the RSA key parameters so that the identifiers are the same as those used in RFC 3447.
- Added the RSA private key fields enabling Chinese Remainder Theorem (CRT) calculations, based upon their use in RFC 3447.

-00

- Created draft-jones-jose-json-private-key to facilitate discussion of the question from the W3C WebCrypto WG to the IETF JOSE WG of whether JOSE plans to support a format for representing private keys.
-

Author's Address

TOC

Michael B. Jones
Microsoft
Email: mbj@microsoft.com
URI: <http://self-issued.info/>