JWS Clear Text JSON Signature Option

B. Jordan, Ed. - Broadcom
S. Erdtman - Spotify AB
A. Rundgren - Independent

Abstract

JWS Clear Text JSON Signature Option describes a method for extending JSON Web Signature (JWS) standard, called JWS/CT. By combining the detached mode of JWS with the JSON Canonicalization Scheme (JCS). Maintaining Signed JSON data in JSON format.

Why JSON Cleartext signatures

In many situations JSON data structures are already defined. To have the option to add a signature without packaging the data within the signature is important to improve backwards compatibility and adoption of signed data.

Large data structures in JSON format that are shared across and between organisations, and signed in later steps and reshared is not feasible with solutions that packages data within the signature, the nesting becomes unbearable.

Adding multiple signatures and nested data structures in different combinations is significantly easier to handle when the signature is packaged within the data and not the other way around.

If needing/wanting to transmit unsigned data it is easy to omit the signature instead of creating a JWS package and then use the “none” algorithm.

Maintaining JSON data as is while in transport is makes readability and debugging easier.

Constructs

JSON Canonicalization Schema and Detached JWS

RFC-8785 (JCS) Canonicalization

Human readable format

```
{
	"numbers": [333333333.33333329, 1E30, 4.50, 2e-3],
	"string": "\u20ac$\u000F\u000aA'\u0042\u0022\u005c\\\"\/",
	"literals": [null, true, false]
}
```

Canonical format (100% valid JSON):

{"numbers":[333333333.3333333,1e+30,4.5,0.002],"string":"
€$\u000f\nA'B\"\\\\\"/","literals":[null,true,false]}

Detach JWS Signature

Normal/Classic JWS Signature

eyJ0eXAiOiJKV1QiLA0KICJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJqb2UiLA0KICJleHAi
OjEzMDA4MTkzODAsDQogImh0dHA6Ly9leGFtcGxlLmNvbS9pc19yb290Ijp0cnVl
fQ.dBjftJeZ4CVP-mB92K27uhbUJU1p1r_wW1gFWFOEjXk

Detached JWS Signature

eyJ0eXAiOiJKV1QiLA0KICJhbGciOiJIUzI1NiJ9..dBjftJeZ4CVP-mB92K27uhbUJU
1p1r_wW1gFWFOEjXk

JWS/CT Signing Process

JWS/CT Signing Process

Create the JSON Object to be Signed

Canonicalize the JSON Object to be Signed

Generate a JWS String

Assemble the Signed JSON Object

```
{
  "statement": "Hello signed world!",
  "otherProperties": [2000, true]
}
```

JWS/CT Signing Process

Create the JSON Object to be Signed

Canonicalize the JSON Object to be Signed

Generate a JWS String

Assemble the Signed JSON Object

{"otherProperties":[2000,true],"statement":"Hello signed world!"}

JWS/CT Signing Process

Create the JSON Object to be Signed

Canonicalize the JSON Object to be Signed

Generate a JWS String

Assemble the Signed JSON Object

eyJhbGciOiJIUzI1NiJ9..VHVItCBCb8Q5CI-49imarDtJeSxH2uLU0DhqQP5Zjw4

JWS/CT Signing Process

Create the JSON Object to be Signed

Canonicalize the JSON Object to be Signed

Generate a JWS String

Assemble the Signed JSON Object


```
{
  "statement": "Hello signed world!",
  "otherProperties": [2000, true],
  "signature": "eyJhbGciOiJIU..."
}
```

JWS/CT Verification Process

JWS/CT Verification Process

Parse the Signed JSON Object

Fetch the Signature Property String

Remove the Signature Property String

Canonicalize the Remaining JSON Object

Validate the JWS String


```
{
  "statement": "Hello signed world!",
  "otherProperties": [2000, true],
  "signature": "eyJhbGciOiJIU..."
}
```

JWS/CT Verification Process

Parse the Signed JSON Object

Fetch the Signature Property String

Remove the Signature Property String

Canonicalize the Remaining JSON Object

Validate the JWS String

eyJhbGciOiJIUzI1NiJ9..VHVItCBCb8Q5CI-49imarDtJeSxH2uLU0DhqQP5Zjw4

JWS/CT Verification Process

Parse the Signed JSON Object

Fetch the Signature Property String

Remove the Signature Property String

Canonicalize the Remaining JSON Object

Validate the JWS String

```
{
  "statement": "Hello signed world!",
  "otherProperties": [2000, true]
}
```

JWS/CT Verification Process

Parse the Signed JSON Object

Fetch the Signature Property String

Remove the Signature Property String

Canonicalize the Remaining JSON Object

Validate the JWS String

{"otherProperties":[2000,true],"statement":"Hello signed world!"}

JWS/CT Verification Process

Parse the Signed JSON Object

Fetch the Signature Property String

Remove the Signature Property String

Canonicalize the Remaining JSON Object

Validate the JWS String

eyJhbGciOiJIUzI1NiJ9.eyJvdGhlclByb3BlcnRpZXMiOlsyMDAwLHRydWVdLCJzdGF0ZW1lbnQiOiJIZWxsbyBzaWduZWQgd29ybGQhIn0.VHVItCBCb8Q5CI-49imarDtJeSxH2uLU0DhqQP5Zjw4

JWS/CT Application Notes

The document does not dictate signature attribute name or location. It is up to the application to choose a suitable name and location for the signature attribute in its context.

The document does not define counter signatures, arrays of signatures or detached signatures, but it exemplifies how it could be done.

Path forward

We suggest that this work is moved forward under ISE, because:

- Canonicalization (RFC-8785) is published as an Independent Submission.
- JOSE WG is not active and the list has previously expressed limited interest in this work.

Path forward

We suggest that this work is moved forward under ISE, because:

- Canonicalization (RFC-8785) is published as an Independent Submission.
- JOSE WG is not active and the list has previously expressed limited interest in this work.

However

- **If you are interested we would love to get reviews**

Thank you!

Questions?
Comments!