野村総合研究所

JOSE WG @ IETF 86

JSON Serialization Specifications

- JWS JSON Serialization
- JWE JSON Serialization

March 13, 2013

Nomura Research Institute
Nat Sakimura

Documents:

JSON Web Signature JSON Serialization

- draft-jones-jose-jws-json-serialization
- http://tools.ietf.org/html/draft-jones-jose-jws-json-serialization-04

JSON Web Encryption JSON Serialization

- draft-jones-jose-jwe-json-serialization
- http://tools.ietf.org/html/draft-jones-jose-jwe-json-serialization-04

JSON Serialization Goals

JSON representation for JWS, JWE values

Support multiple signatures/recipients

Use identical crypto operations as compact (dot-separated) serializations

Design Methodology

Use JSON members for each JWS/JWE element

- (instead of separating them with '.' characters)

Use single JSON values for elements common to multiple signatures/recipients

- JWS Payload
- JWE Initialization Vector
- JWE Ciphertext

Use JSON arrays for elements specific to each signature/ recipient

- JWS Header, JWS Signature
- JWE Header, JWE Encrypted Key, JWE Integrity Value

Specs Very Stable

- Last change was to use an array of structures for per-recipient values (Oct. 20, 2012)
  - At the request of the working group

It's Time for WG Draft Status

Once rechartering is complete, it's time to add these as working group documents

- *To meet needs of use cases requiring multiple signatures/recipients*

Whether to combine these with JWS, JWE?

- Rechartering will allow us either option
- Simpler to first publish WG versions of existing docs
- Keeping them separate makes life easier for developers who only need the Compact Serialization

Backup Slides

Headers from Example JWS-JS

```
{"alg":"RS256"}


{"alg":"ES256"}
```

Example JWS-JS


```
{"recipients":[
  {"header":"eyJhbGciOiJSUzI1NiJ9",
   "signature":
    "cC4hiUPoj9Eetdgtv3hF80EGrhuB__dzERat0XF9g2VtQgr9PJbu3XOiZj5RZ
     mh7AAuHIm4Bh-0Qc_lF5YKt_O8W2Fp5jujGbds9uJdbF9CUAr7t1dnZcAcQjb
     KBYNX4BAynRFdiuB--f_nZLgrnbyTyWzO75vRK5h6xBArLIARNPvkSjtQBMHl
     b1L07Qe7K0GarZRmB_eSN9383LcOLn6_dO--xi12jzDwusC-eOkHWEsqtFZES
     c6BfI7noOPqvhJ1phCnvWh6IeYI2w9QOYEUipUTI8np6LbgGY9Fs98rqVt5AX
     LIhWkWywlVmtVrBp0igcN_IoypGlUPQGe77Rw"},
  {"header":"eyJhbGciOiJFUzI1NiJ9",
   "signature":
    "DtEhU3ljbEg8L38VWAfUAqOyKAM6-Xx-F4GawxaepmXFCgfTjDxw5djxLa8IS
     lSApmWQxfKTUJqPP3-Kg6NU1Q"}],
 "payload":
  "eyJpc3MiOiJqb2UiLA0KICJleHAiOjEzMDA4MTkzODAsDQogImh0dHA6Ly9leGF
   tcGxlLmNvbS9pc19yb290Ijp0cnVlfQ"
}
```

Headers from Example JWE-JS

```
{"alg":"RSA1_5","enc":"A128CBC+HS256"}

{"alg":"RSA-OAEP","enc":"A128CBC+HS256"}
```

Example JWE-JS


```
{"recipients":[
  {"header":"eyJhbGciOiJSU0ExXzUiLCJlbmMiOiJBMTI4Q0JDK0hTMjU2In0",
   "encrypted_key":
    "O6AqXqgVlJJ4c4lp5sXZd7bpGHAw6ARkHUeXQxD1cAW4-X1x0qtj_AN0mukqE
     Ol4Y6UOwJXIJY9-G1ELK-RQWrKH_StR-AM9H7GpKmSEji8QYOcMOjr-u9H1Lt
     _pBEieG802SxWz0rbFTXRcj4BWLxcpCtjUZ31AP-sc-L_eCZ5UNl0aSRNqFsk
     uPkzRsFZRDJqSSJeVOyJ7pZCQ83fli19Vgi_3R7XMUqluQuuc7ZHOWixi47jX
     lBTlWRZ5iFxaS8G6J8wUrd4BKggAw3qX5XoIfXQVlQZE0Vmkq_zQSIo5LnFKy
     owooRcdsEuNh9B9Mkyt0ZQElG-jGdtHWjZSOA",
   "integrity_value":"RBGhYzE8_cZLHjJqqHuLhzbgWgL_wV3LDSUrcbkOiIA"},
  {"header":"eyJhbGciOiJSU0EtT0FFUCIsImVuYyI6IkExMjhDQkMrSFMyNTYifQ",
   "encrypted_key":
    "myoFYZHErXG4gMVWl9UrFOCFIwvOUudYrxTsRsOt6maTc3W8G1FqGVOIBSZve
     BdZz2LqS42xta5OXEwLYaocObUxtfH9H8vMsjO-mBo7U9mp_PkS9PqVJMkeEe
     PLhzNLH0ecq7nYT6AFr5sSt4WMOPjSwHVQWtx43fZt4HvYaE_vgeSrxdi8KAb
     xbLzK_-qcYT6H7cwOMZrT6SFcXgLXESuKpF0azSGQtUmo0MLICP0YPBecGLTo
     PiveOH2awKZx0FkzPwi4JmOIvnAJ_wVQQJDVELwO9SIoF8olCQRHGyZ9rzDrr
     GRkoYgm2jVz-x0BuFVQFa4ZNufudtiT8pQxKg",
   "integrity_value":"i45dXWFjRKk805VtjIw_8iqGq1r9qPV7ULDLbnNAC_Q"}],
 "initialization_vector":"AxY8DCtDaGlsbGljb3RoZQ",
 "ciphertext":
  "1eBWFgcrz40wC88cgv8rPgu3EfmC1p4zT0kIxxfSF2zDJcQ-iEHk1jQM95xAdr5Z"
}
```