SD-JWT & SD-JWT VC 101

SD-JWT & SD-JWT VC

- Formats for
  - enabling **selective disclosure and key binding for JWS/JWT** (SD-JWT)
  - **credentials** based on that format (SD-JWT VC)
- Attributes are structured as JSON
- Specified in the OAuth Working Group at the IETF
  - SD-JWT: Working group last call
  - SD-JWT VC: Working group draft

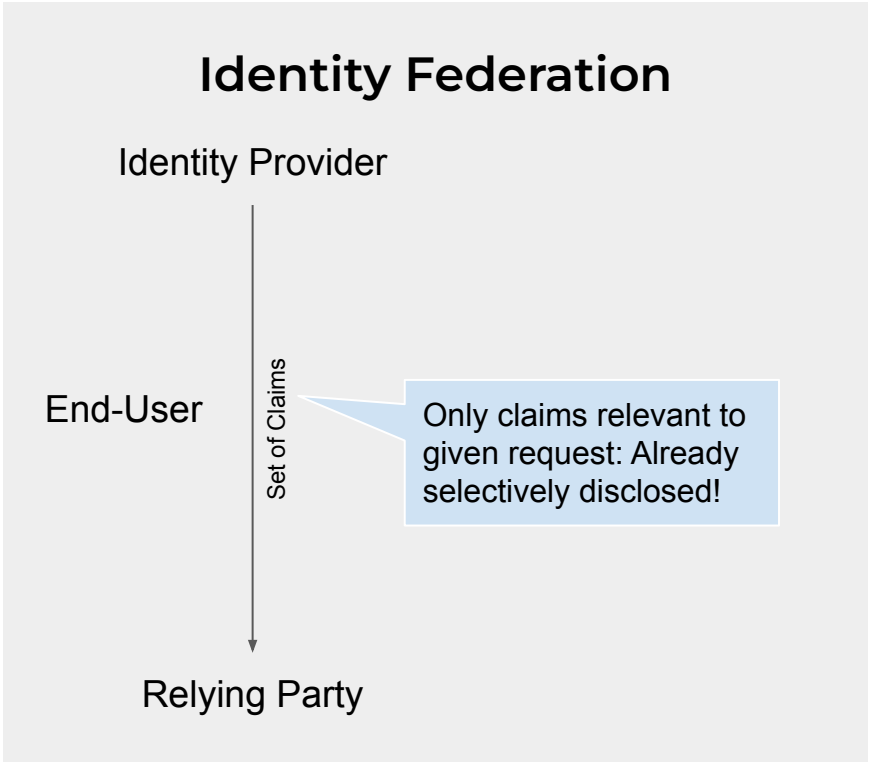
SD-JWT

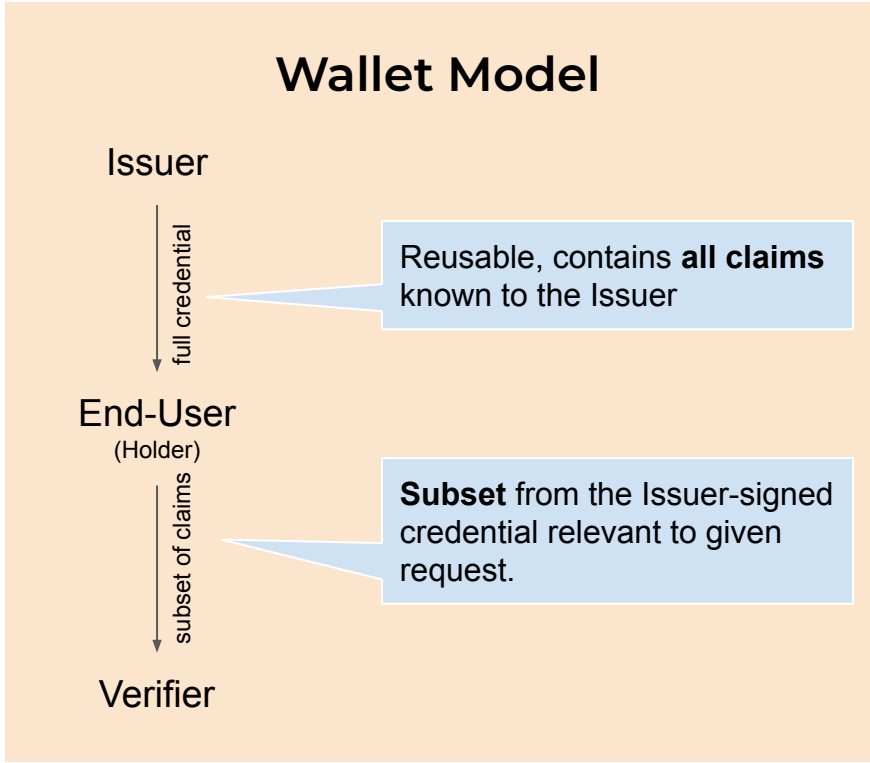
Selective Disclosure for JWTs
using a simple, salted-hash based format
— for verifiable credentials and more.

IETF Draft: https://datatracker.ietf.org/doc/draft-ietf-oauth-selective-disclosure-jwt/

Daniel Fett
Kristina Yasuda
Brian Campbell

Credential Issuance & Presentation Decoupled





Selective Disclosure

Issuer issued a whole set of claims:


```
{
   "iss": "https://server.example.com",
   "sub": "some-user-identifier",
   "aud": "s6BhdRkqt3",
   "given_name": "John",
   "family_name": "Doe",
   "email": "johndoe@example.com",
   "phone_number": "+1-202-555-0101",
   "address": {
       "street_address": "123 Main St",
       "locality": "Anytown",
       "region": "Anystate",
       "country": "US"
   },
   "birthdate": "1940-01-01"
}
```


✓ signed by Issuer

But **Verifier** only needs a subset in a given request:


```
{
   "iss": "https://server.example.com",
   "sub": "some-user-identifier",
   "aud": "s6BhdRkqt3",
   "given_name": "John",
   "family_name": "Doe",
   "email": "johndoe@example.com",
   "phone_number":
   "address": {
   },
   "birthdate":
}
```


✓ signed by Issuer

SD-JWT in 5 Simple Steps

Step 1: Prepare User Data

```
{
 "iss": "https://example.com",
 "type": "IdentityCredential",
 "cnf": {"jwk": {"kty": "RSA","n": "0vx....Kgw","e": "AQAB" } },

 "given_name": "Max",
 "family_name": "Mustermann",
 "email": "mustermann@example.com",
 "address": {
   "street_address": "Musterstr. 23",
   "locality": "Berlin",
   "country": "DE"
  }
}
```

SD-JWT in 5 Simple Steps

Step 2: Create Disclosures

```
{
 "iss": "https://example.com",
 "type": "IdentityCredential",
 "cnf": {"jwk": {"kty": "RSA","n": "0vx....Kgw","e": "AQAB" } },

 "given_name": "Max",·····  ["GO0r26nO-iW50ZcAoOilFw", "given_name", "Max"]
 "family_name": "Mustermann",·····  ["cSlbR135i0NjhsouMxrjjg", "family_name", "Mustermann"]
 "email": "mustermann@example.com",·····  ["oHDt43Vwuhpo8mzaprgCcw", "email", "mustermann@example.com"]
 "address": {
   "street_address": "Musterstr. 23",·····  ["rGc0KtY6WmflywTTKEWIEQ", "street_address", "Musterstr. 23"]
   "locality": "Berlin",·····  ["pGQMQx-2tH2XwC_eQCFn4g", "locality", "Berlin"]
   "country": "DE"  ·····  ["TI15M8G5UIxPiWNZ-VLYBA", "country", "DE"]
  }
}
```

salt claim name claim value

SD-JWT in 5 Simple Steps

Step 3: Hash Disclosures & Replace Original Claims

```
{
 "iss": "https://example.com",
 "type": "IdentityCredential",
 "cnf": {"jwk": {"kty": "RSA","n": "0vx....Kgw","e": "AQAB" } },

  "_sd": [ "EW1oOegqa5mGcbytT5S-kAubcEjYEUwRkXlu2vC5l2O",        ← ["GOOr26nO-iW5OZcAoOilFw", "given_name", "Max"]
           "FEx-ITHt41I8_cnOSS-hvoLneX_RGlJo_8o2xRNhfdk",        ← ["cSlbR135iONjhsouMxrjjg", "family_name", "Mustermann"]
           "igg7H5fn2eBEMIEkE5Ckbm23QuwDJlTYoKRip08dYIc" ],      ← ["oHDt43Vwuhpo8mzaprgCcw", "email", "mustermann@example.com"]
 "address": {
   "_sd": [ "gqB5kmAwyry88aHjaAeO-USX6JOMaojukKsheo38OOc",      ← ["rGcOKtY6WmflywTTKEWIEQ", "street_address", "Musterstr. 23"]
            "w8InvxsPXdKoowuVpyBMgl1b9_R2b6Xpa3OYOIjgQro",      ← ["pGQMQx-2tH2XwC_eQCFn4g", "locality", "Berlin"]
            "vOnlYtcjr872fP3Wa75Ozl7c-6_MOVdIUNtwLKKxZw0" ]     ← ["TI15M8G5UIxPiWNZ-VLYBA", "country", "DE"]
  }
}
```

SD-JWT in 5 Simple Steps

Step 4: Sign SD-JWT & Encode for Transport

```
{
  "iss": "https://example.com",
eyJhbGciOiAiUlMyNTYiLCAia2lkIjogImNBRUlVcUowY21MekQxa3pHemhlaUJhZzBZ
UkF6VmRsZnhOMjgwTmdIYUEifQ.eyJpc3MiOiAiaHR0cHM6Ly9leGFtcGxlLmNvbS9pc
3N1ZXIiLCAiY25mIjogeyJqd2siOiB7Imt0eSI6ICJSU0EiLCAibiI6ICIwdng3YWdvZ
WJHY1FTdS4uLi4tY3NGQ3VyLWtFZ1U4YXdhcEp6S25xREtndyIsICJlIjogIkFRQUIif
X0sICJ0eXBlIjogIklkZW50aXR5Q3JlZGVudGlhbCIsICJjcmVkZW50aWFsU3ViamVjd
CI6IHsiX3NkIjogWyJFVzFvMGVncWE1bUdjYnl0VDVTLWtBdWJjRWpZRVV3UmtYbHUyd
kM1bDIwIiwgIkZFeC1JVEhONDFJOF9jbjBTUy1odm9MbmVYX1JHbEpvXzhvMnhSTmhmZ
GsiLCAiUXhKVi0yVjFIOG1jbHRSNnZWQzRtM3JlVTVhTkg5d2RKejJVZG1SbOkxRSIsI
CJhdFVuMVRZd1JBbDRHUTdQZUV0WGFNdzJmNHVJVGlKclgOODV3TTh2NjdFIiwgImZUT
XczdmtrRUx3TDFYTnVZSzhIN3pCS0NIdV91aWY2MFNsRzFweVhJVVEiLCAiaWdnNOg1Z
m4yZUJFTUlFa0U1Q2tibTIzUXV3REpsVFlvS1JpcDA4ZFlJYyIsICJ0cFV0bDcwaHBVX
3hucnZaaTBHaEdvUlIxam1OMXpZZ3Z2NUlZMEF4N0tjIl0sICJhZGRyZXNzIjogeyJfc
2QiOiBbImdxQjVrbUF3eXJ5ODhhSGphQWVPLVVTWDZKT01hb2p1a0tzaGVvMzhPMGMiL
CAidk9ubFlOY2pyODcyZlAzV2E3NU96bDdjLTZfTU9WZElVTnR3TEtLeFp3MCIsICJ3O
EludnhzUFhkS29vd3VWcHlCTWdsMWI5X1IyYjZYcGEzT1lPSWpnUXJvIl19fSwgImlhd
CI6IDE1MTYyMzkwMjIsICJleHAiOiAxNTE2MjQ3MDIyLCAic2RfZGlnZXN0X2Rlcml2Y
XRpb25fYWxnIjogInNoYS0yNTYifQ.1UHEPtLLUXOT51jH3gg-3C-ZidWzsB9Un-VxmM
VdQtTbLLhwDTB6HJtt15p43yCXTzdpiZxtDI6fr07TpODy_Umg3Q5_FxFj4WHnsVuVzu
ASU8cFlGPi6xgH9D3w1G2hqepBS8DyQ5bA_p5kN_tKJVoP1xWhcQujRJ8kkEKQsRia4F
hrBldl8f41wgu_ipPqh1Ix4BVI7GJClZNx94nWPT7JUFkI6Y6JkahLf3S6gBOMxtmLAe
YOqkuz8VeOZNfl_CDog55kVTkArorfoL6D6TEjI__-w6YyUOPnIRJXJOwrYfoyhNl8LK
AP38QYMpdR7z_rsvHpQHzFAPTmevnHDg
  }
}
```

← ["GOOr26nO-iW5OZcAoOilFw", "given_name", "Max"]
← ["cSlbR135iONjhsouMxrjjg", "family_name", "Mustermann"]
← ["oHDt43Vwuhpo8mzaprgCcw", "email", "mustermann@example.com"]

← ["rGcOKtY6WmflywTTKEWIEQ", "street_address", "Musterstr. 23"]
← ["pGQMQx-2tH2XwC_eQCFn4g", "locality", "Berlin"]
← ["TI15M8G5UIxPiWNZ-VLYBA", "country", "DE"]

SD-JWT in 5 Simple Steps

Step 5: Base64url-encode Disclosures for Transport

```
{
  "iss": "https://example.com",
eyJhbGciOiAiUlMyNTYiLCAia2lkIjogImNBRUlVcUowY21MekQxa3pHemhlaUJhZzBZ
UkF6VmRsZnhOMjgwTmdIYUEifQ.eyJpc3MiOiAiaHR0cHM6Ly9leGFtcGxlLmNvbS9pc
3N1ZXIiLCAiY25mIjogeyJqd2siOiB7Imt0eSI6ICJSU0EiLCAibiI6ICIwdng3YWdvZ
WJHY1FTdS4uLi4tY3NGQ3VyLWtFZ1U4YXdhcEp6S25xREtndyIsICJlIjogIkFRQUIif
X0sICJ0eXBlIjogIklkZW50aXR5Q3JlZGVudGlhbCIsICJjcmVkZW50aWFsU3ViamVjd
CI6IHsiX3NkIjogWyJFVzFvMGVncWE1bUdjYnl0VDVTLWtBdWJjRWpZRVV3UmtYbHUyd
kM1bDIwIiwgIkZFeC1JVEh0NDFJOF9jbjBTUy1odm9MbmVYX1JHbEpvXzhvMnhSTmhmZ
GsiLCAiUXhKViOyVjFIOG1jbHRSNnZWQzRtM3JlVTVhTkg5d2RKejJVZG1SbOkxRSIsI
CJhdFVuMVRZd1JBbDRHUTdQZUVOWGFNdzJmNHVJVGlKclg0ODV3TTh2NjdFIiwgImZUT
XczdmtrRUx3TDFYTnVZSzhIN3pCS0NIdV91aWY2MFNsRzFweVhJVVEiLCAiaWdnNOg1Z
m4yZUJFTUlFaOU1Q2tibTIzUXV3REpsVFlvS1JpcDA4ZFlJYyIsICJOcFVObDcwaHBVX
3hucnZaaTBHaEdvUlIxam1OMXpZZ3Z2NUlZMEF4N0tjIlOsICJhZGRyZXNzIjogeyJfc
2QiOiBbImdxQjVrbUF3eXJ5ODhhSGphQWVPLVVTWDZKT01hb2p1aOtzaGVvMzhPMGMiL
CAidk9ubFlOY2pyODcyZlAzV2E3NU96bDdjLTZfTU9WZElVTnR3TEtLeFp3MCIsICJ3O
EludnhzUFhkS29vd3VWcHlCTWdsMWI5X1IyYjZYcGEzT1lPSWpnUXJvIl19fSwgImlhd
CI6IDE1MTYyMzkwMjIsICJleHAiOiAxNTE2MjQ3MDIyLCAic2RfZGlnZXNOX2Rlcml2Y
XRpb25fYWxnIjogInNoYS0yNTYifQ.1UHEPtLLUXOT51jH3gg-3C-ZidWzsB9Un-VxmM
VdQtTbLLhwDTB6HJtt15p43yCXTzdpiZxtDI6fr07TpODy_Umg3Q5_FxFj4WHnsVuVzu
ASU8cFlGPi6xgH9D3w1G2hqepBS8DyQ5bA_p5kN_tKJVoP1xWhcQujRJ8kkEKQsRia4F
hrBldl8f41wgu_ipPqh1Ix4BVI7GJClZNx94nWPT7JUFkI6Y6JkahLf3S6gBOMxtmLAe
YOqkuz8VeOZNfl_CDog55kVTkArorfoL6D6TEjI__-w6YyUOPnIRJXJOwrYfoyhNl8LK
AP38QYMpdR7z_rsvHpQHzFAPTmevnHDg
  }
}
```

```
← ["GO0r26nO-iW50ZcAoOilFw", "given_name", "Max"]
← ["cSlbR135jONjbsouMxrjig", "family_name", "Mustermann"]
~WyJHTzByMjZuTy1pVzUwWmNBb09pbEZ3IiwgImdpdmVuX25hbWUiLCAiTWF4Il0
~WyJjU2xiUjEzNWkwTmpoc291TXhyampnIiwgImZhbWlseV9uYW1lIiwgIk11c3Rlcm1
hbm4iXQ
~WyJvSER0NDNWd3VocG84bXphcHJnQ2N3IiwgImVtYWlsIiwgIm11c3Rlcm1hbm5AZXh
hbXBsZS5jb20iXQ
~WyJyR2MwS3RZNldtZmx5d1RUS0VXSUVRIiwgInN0cmVldF9hZGRyZXNzIiwgIk11c3R
lcnN0ci4gMjMiXQ
~WyJwR1FNUXgtMnRIMlh3Q19lUUNGbjRnIiwgImxvY2FsaXR5IiwgIkJlcmxpbiJd
~WyJUSTE1TThHNVVJeFBpV05aLVZMWUJBIiwgImNvdW50cnkiLCAiREUiXQ
```

→ Done!

Design Principles

	SD-JWT
Complexity	Selective disclosure, as simple as possible
Algorithms	Standard cryptography: JWS Signature + Hash function
Format	JWT & JSON
Security	Security-by-design Easy to understand & verify Hardware binding possible Cryptographic agility
Availability	Widely-available JWT libraries can be leveraged Already five independent implementations
Use Cases	Universal (beyond identity use cases)

Issuer

Issuance

JWT
plain-text claims
+ hashed Disclosures

✓ signed by Issuer

Disclosures
salt + claim name + claim value

```
WyJrSEhwOTEtdEFadDhtOUUOSmwOWGJRIiwgImdpdmVuX25hbWUiLCAiSm9obiJd
WyJQaklxcEdXbDRlQjRRcm9EaHFRdzB3IiwgImZhbWlseV9uYW1lIiwgIkRvZSJd
WyJ4bmlQNEpadE5XSUgtTGtfRHQtby1BIiwgInN0cmVldF9hZGRyZXNzIiwgIkRvZXNOcmVldCAxIlO
WyJLdGZzeHhUbTJtdzBZTFVjS1pVOHRBIiwgImxvY2FsaXR5IiwgIkFueXRvd24iXQ
```

End-User
(Holder)

Presentation

Verifier

Issuer

Issuance

JWT
plain-text claims
+ hashed Disclosures

✓ signed by Issuer

Disclosures
salt + claim name + claim value

WyJrSEhwOTEtdEFadDhtOUUOSmwOWGJRIiwgImdpdmVuX25hbWUiLCAiSm9obiJd
WyJQaklxcEdXbDRlQjRRcm9EaHFRdzB3IiwgImZhbWlseV9uYW1lIiwgIkRvZSJd
WyJ4bmlQNEpadE5XSUgtTGtfRHQtby1BIiwgInN0cmVldF9hZGRyZXNzIiwgIkRvZXN0cmVldCAxIl0
WyJLdGZzeHhUbTJtdzBZTFVjS1pVOHRBIiwgImxvY2FsaXR5IiwgIkFueXRvd24iXQ

End-User
(Holder)

Presentation

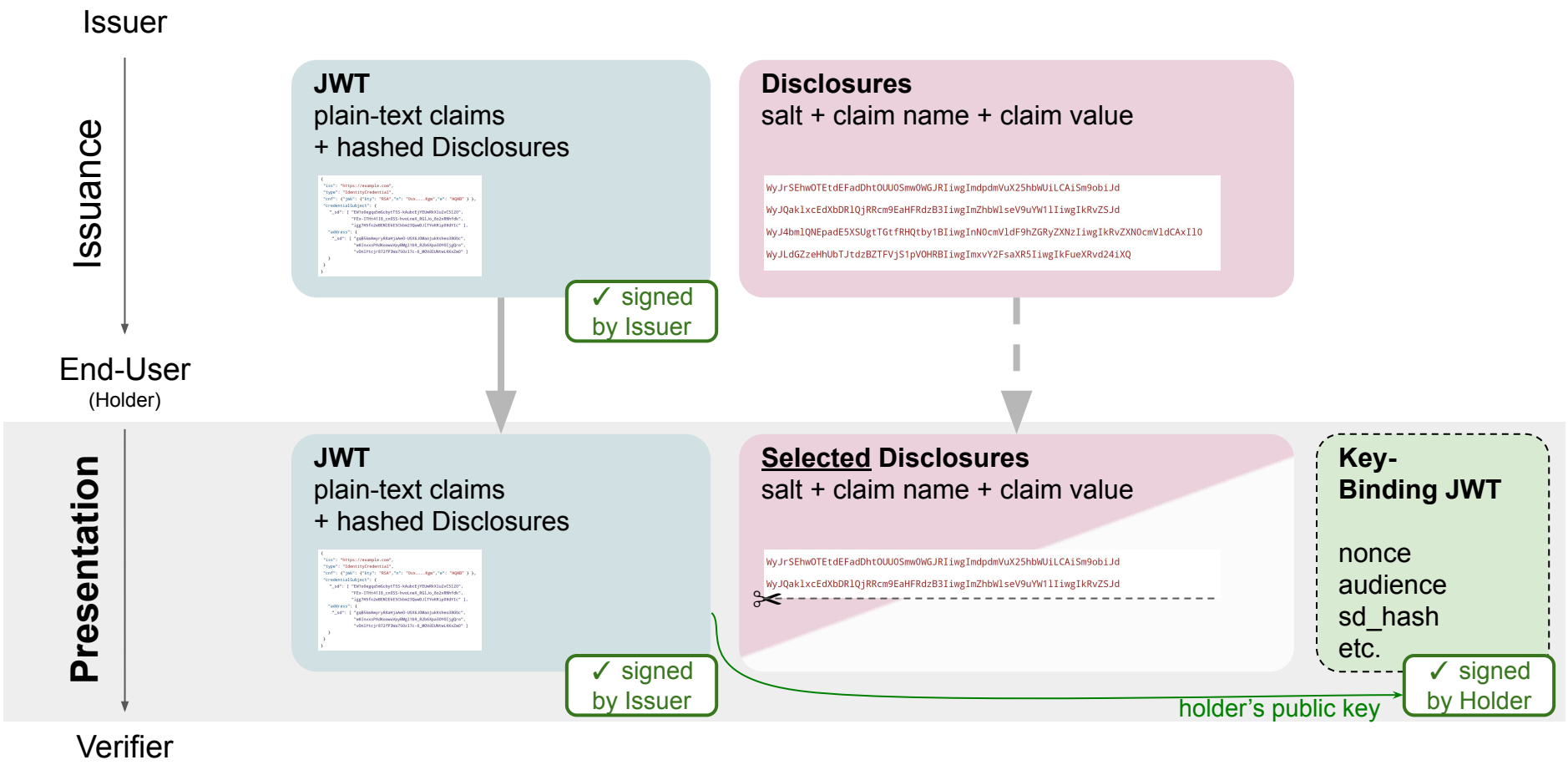
JWT
plain-text claims
+ hashed Disclosures

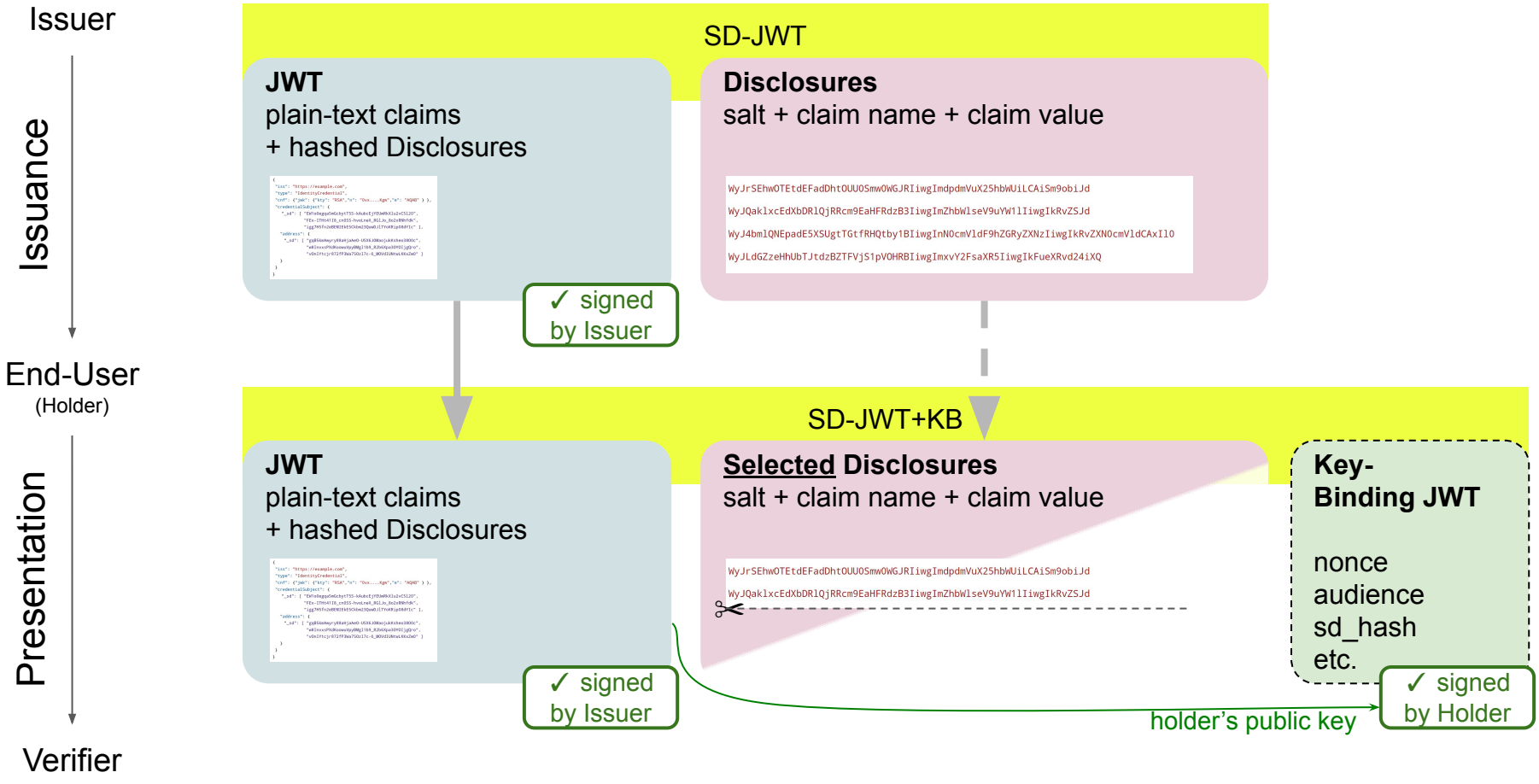
✓ signed by Issuer

<u>Selected</u> Disclosures
salt + claim name + claim value

WyJrSEhwOTEtdEFadDhtOUUOSmwOWGJRIiwgImdpdmVuX25hbWUiLCAiSm9obiJd
WyJQaklxcEdXbDRlQjRRcm9EaHFRdzB3IiwgImZhbWlseV9uYW1lIiwgIkRvZSJd

Verifier


Issuer
Issuance
End-User
(Holder)
Presentation
Verifier
JWT
plain-text claims
+ hashed Disclosures
✓ signed
by Issuer
Disclosures
salt + claim name + claim value
WyJrSEhwOTEtdEFadDhtOUUOSmwOWGJRIiwgImdpdmVuX25hbWUiLCAiSm9obiJd
WyJQaklxcEdXbDRlQjRRcm9EaHFRdzB3IiwgImZhbWlseV9uYW1lIiwgIkRvZSJd
WyJ4bmlQNEpadE5XSUgtTGtfRHQtby1BIiwgInN0cmVldF9hZGRyZXNzIiwgIkRvZXN0cmVldCAxIl0
WyJLdGZzeHhUbTJtdzBZTFVjS1pVOHRBIiwgImxvY2FsaXR5IiwgIkFueXRvd24iXQ
JWT
plain-text claims
+ hashed Disclosures
✓ signed
by Issuer
Selected Disclosures
salt + claim name + claim value
WyJrSEhwOTEtdEFadDhtOUUOSmwOWGJRIiwgImdpdmVuX25hbWUiLCAiSm9obiJd
WyJQaklxcEdXbDRlQjRRcm9EaHFRdzB3IiwgImZhbWlseV9uYW1lIiwgIkRvZSJd
Key-
Binding JWT
nonce
audience
sd_hash
etc.
holder's public key
✓ signed
by Holder


Issuer
Issuance
End-User
(Holder)
Presentation
Verifier
SD-JWT
JWT
plain-text claims
+ hashed Disclosures
Disclosures
salt + claim name + claim value
WyJrSEhwOTEtdEFadDhtOUUOSmwOWGJRIiwgImdpdmVuX25hbWUiLCAiSm9obiJd
WyJQaklxcEdXbDRlQjRRcm9EaHFRdzB3IiwgImZhbWlseV9uYW1lIiwgIkRvZSJd
WyJ4bmlQNEpadE5XSUgtTGtfRHQtby1BIiwgInN0cmVldF9hZGRyZXNzIiwgIkRvZXN0cmVldCAxIl0
WyJLdGZzeHhUbTJtdzBZTFVjS1pVOHRBIiwgImxvY2FsaXR5IiwgIkFueXRvd24iXQ
✓ signed
by Issuer
SD-JWT+KB
JWT
plain-text claims
+ hashed Disclosures
Selected Disclosures
salt + claim name + claim value
WyJrSEhwOTEtdEFadDhtOUUOSmwOWGJRIiwgImdpdmVuX25hbWUiLCAiSm9obiJd
WyJQaklxcEdXbDRlQjRRcm9EaHFRdzB3IiwgImZhbWlseV9uYW1lIiwgIkRvZSJd
✓ signed
by Issuer
Key-
Binding JWT
nonce
audience
sd_hash
etc.
holder's public key
✓ signed
by Holder

Example Presentation

Issuer-signed SD-JWT~Disclosures~KB-JWT

Key-Binding JWT Body:


```
{
  "nonce": "1234567890",
  "aud": "https://verifier.example.org",
  "iat": 1698077790,
  "sd_hash": "34t8vCC_c_vVLnOadBtwh4dFvBErSl9zKOqwm6ihT_U"
}
```

Reconstructing the Original Data


```
{
   "_sd": [
      "fOBUSQvo46yQO-wRwXBcGqvnbKIueISEL961_Sjd4do"
   ],
   "iss": "https://issuer.example.com",
   "iat": 1683000000,
   "exp": 1883000000,
   "sub": "6c5c0a49-b589-431d-bae7-219122a9ec2c",
   "_sd_alg": "sha-256",
   "cnf": {
      "jwk": {
         "kty": "EC",
         "crv": "P-256",
         "x": "TCAER19Zvu3OHF4j4W4vfSVoHIP1ILilDls7vCeGemc",
         "y": "ZxjiWWbZMQGHVWKVQ4hbSIirsVfuecCE6t4jT9F2HZQ"
      }
   }
}
```

```
["2GLC42sKQveCfGfryNRN9w", "address", {"street_address":
"Schulstr. 12", "locality": "Schulpforta", "region":
"Sachsen-Anhalt", "country": "DE"}]
```

```
fOBUSQvo46yQO-wRwXBcGqvnbKIueISEL961_Sjd4do
```

```
{
   "nonce": "1234567890",
   "aud": "https://verifier.example.org",
   "iat": 1721206049,
   "sd_hash": "GpLqgSVPUPqeetF7H4jxV3GkPn3BkT1pf3yY2I4G4ew"
}
```

Reconstructing the Original Data

Reconstructing the Original Data

Reconstructing the Original Data


```
{
   "address": {
      "street_address": "Schulstr. 12",
      "locality": "Schulpforta",
      "region": "Sachsen-Anhalt",
      "country": "DE"
   }
   "iss": "https://issuer.example.com",
   "iat": 1683000000,
   "exp": 1883000000,
   "sub": "6c5c0a49-b589-431d-bae7-219122a9ec2c",
   "_sd_alg": "sha-256",
   "cnf": {
       "jwk": {
           "kty": "EC",
           "crv": "P-256",
           "x": "TCAER19Zvu3OHF4j4W4vfSVoHIP1ILilDls7vCeGemc",
           "y": "ZxjiWWbZMQGHVWKVQ4hbSIirsVfuecCE6t4jT9F2HZQ"
       }
   }
}
```


… repeat for all disclosures …

Done!

Any Element may be Selectively Disclosable

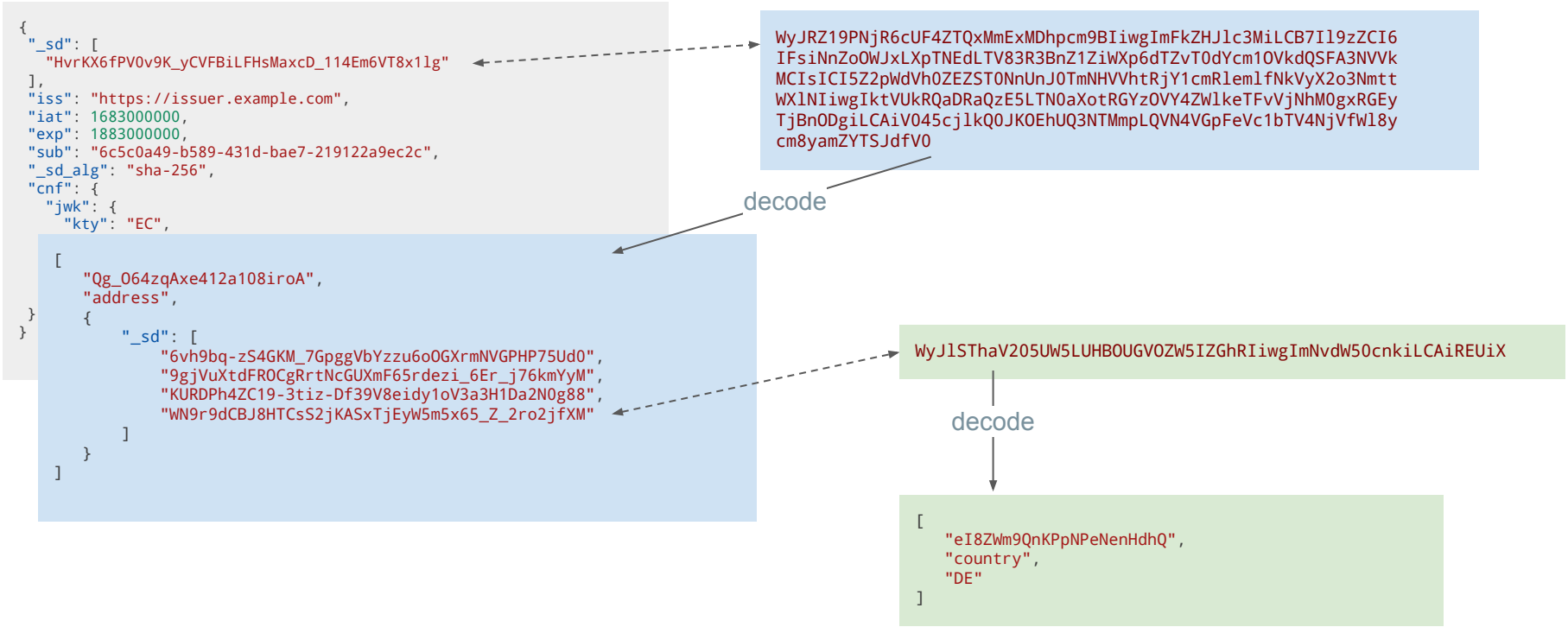
in sub-structures


```
{
 "iss": "https://issuer.example.com",
 "iat": 1683000000,
 "exp": 1883000000,
 "sub": "6c5c0a49-b589-431d-bae7-219122a9ec2c",
 "address": {
   "_sd": [
     "6vh9bq-zS4GKM_7GpggVbYzzu6oOGXrmNVGPHP75Ud0",
     "9gjVuXtdFROCgRrtNcGUXmF65rdezi_6Er_j76kmYyM",
     "KURDPh4ZC19-3tiz-Df39V8eidy1oV3a3H1Da2N0g88"
   ],
   "country": "DE"
 },
 "_sd_alg": "sha-256"
}
```


array elements


```
{
 "iss": "https://issuer.example.com",
 "iat": 1683000000,
 "exp": 1883000000,
 "sub": "user_42",
 "nationalities": [
   {
     "...": "pFndjkZ_VCzmyTa6UjlZo3dh-ko8aIKQc9DlGzhaVYo"
   },
   {
     "...": "7Cf6JkPudry3lcbwHgeZ8khAv1U1OSlerP0VkBJrWZ0"
   }
 ],
 "_sd_alg": "sha-256",
 "cnf": {...}
}
```

Recursive Selective Disclosure for Fine-Grained Release



Security Considerations (I)

Signature verification: Verifiers could verify the signature inadequately/partially and accept tampered credentials

Mitigating measures:

- Simple processing model, specified in detail in the standard
- Established algorithms enable the use of existing implementations

Manipulation of disclosures: If the hashes of the disclosures are not checked by the verifier, manipulated plaintext values could be accepted.

Mitigating measures:

- Design: Generally no assignment to the document possible without hash calculation
- Processing model specified in detail

Security Considerations (II)

Missing check of key binding: Verifiers could accept credentials without key binding

Mitigating measures:

- Different formats with/without key binding
- Differentiation in terminology
- Detailed discussion in the standard

Privacy Considerations

Unlinkability (“unlinkability”): Several presentations of the same credential can be traced back to the same person (due to the same hash values).

Mitigating measures:

- Single use: Credentials are always issued in groups - same data, different salt values. Each individual credential is then only used once.

SD-JWT VC

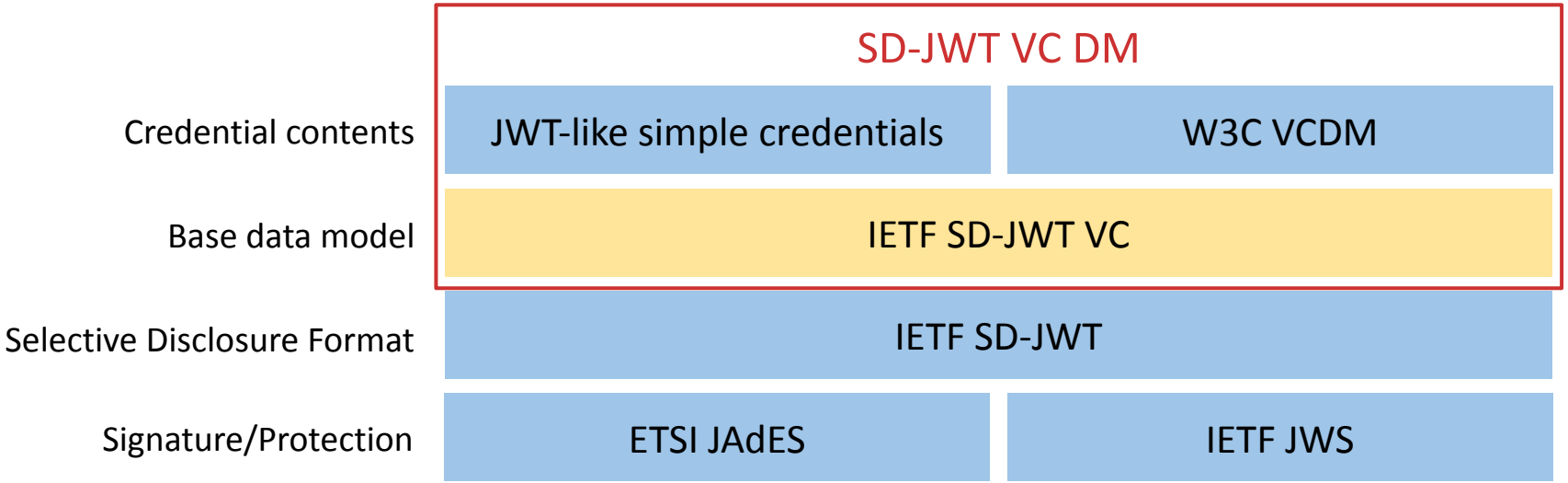
SD-JWT VC

Credentials based on SD-JWT VC
using an extensible data model

IETF Draft: https://datatracker.ietf.org/doc/draft-ietf-oauth-sd-jwt-vc/

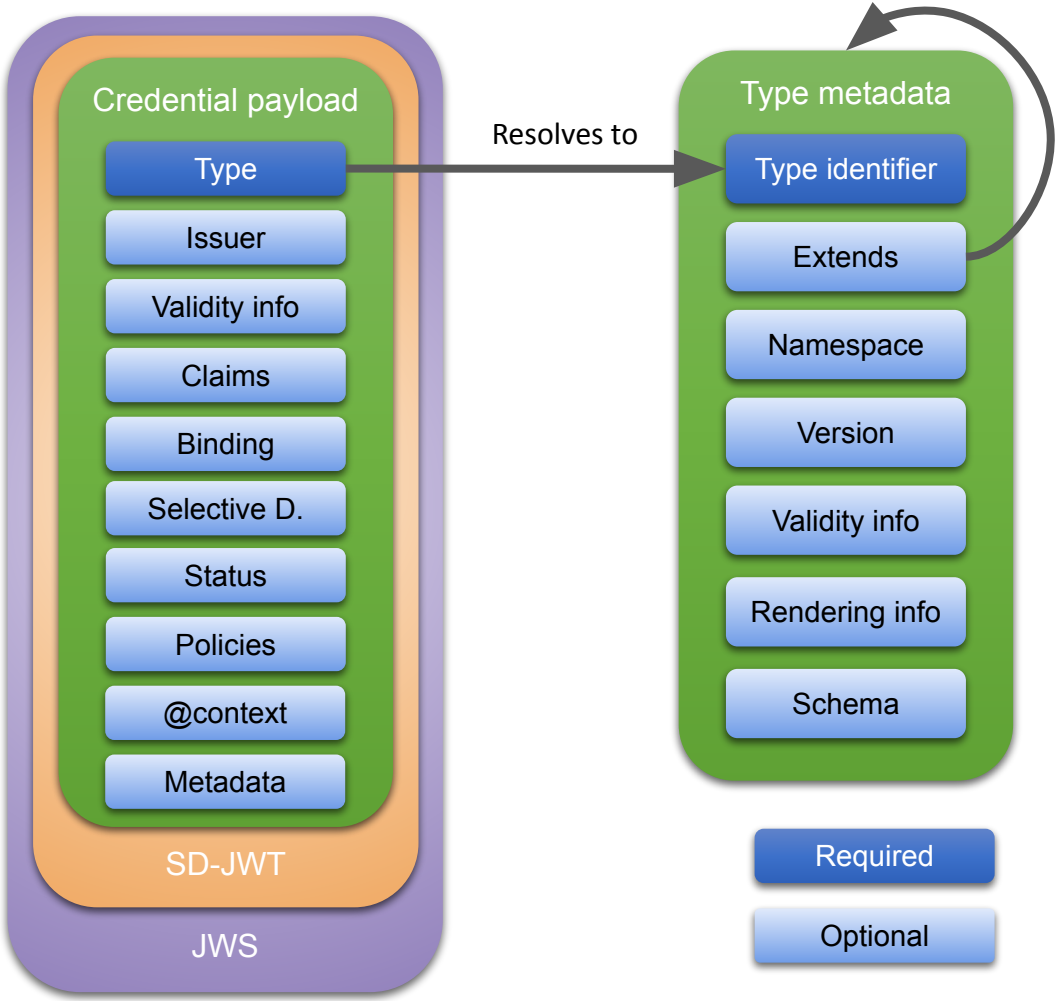
Daniel Fett
Oliver Terbu
Brian Campbell

SD-JWT VC DM


SD-JWT VC DM
Credential contents
JWT-like simple credentials
W3C VCDM
Base data model
IETF SD-JWT VC
Selective Disclosure Format
IETF SD-JWT
Signature/Protection
ETSI JAdES
IETF JWS

Overview

- The **core data model** consists of a set of required and optional claims
- The **type identifier** resolves to **type metadata** that contains additional information about the credential
- The data model allows to express simple and complex information sets



Defined Claims

- **iss** — The Issuer of the Verifiable Credential. The value of iss MUST be a URI.
- **nbf** — The time before which the Verifiable Credential MUST NOT be accepted before validating.
- **exp** — The expiry time of the Verifiable Credential after which the Verifiable Credential is no longer valid.
- **cnf** — Contains the confirmation method identifying the proof of possession key. For proof of cryptographic Key Binding, the Key Binding JWT in the presentation of the SD-JWT MUST be signed by the key identified in this claim.
- **vct** — The type of the Verifiable Credential, e.g., https://credentials.example.com/identity_credential.
- **status** — The information on how to read the status of the Verifiable Credential.
- **sub** — The identifier of the Subject of the Verifiable Credential. The Issuer MAY use it to provide the Subject identifier known by the Issuer. There is no requirement for a binding to exist between sub and cnf claims.
- **iat** — The time of issuance of the Verifiable Credential. See [RFC7519] for more information.

Example: Simplified PID

The data model represents a simplified PID without selective disclosure


```
{
 "vct": "eudi:example:pid",

 "given_name": "Jack",
 "family_name": "Dougherty",
 "birthdate": "1980-05-23",

 "cnf": {
   "jwk": {
     "kty": "EC",
     "crv": "P-256",
     "x": "52aDI_ur05n1f_p3jiYGUU82oKZr3m4LsAErM536crQ",
     "y": "ckhZ-KQ5aXNL91R8Eufg1aOf8Z5pZJnIvuCzNGfdnzo"
   }
 }
}
```


(All examples shortened for presentation.)

Example: Simplified PID

Same as before, with selective disclosure.

After processing, data structure as shown on previous slide is restored.


```
{
 "vct": "eudi:example:pid",
 "_sd_alg" : "sha-256" ,

 "_sd": [
   "09vKrJMOlyTWMOsjpu_pdOBVBQ2M1y3KhpH515nXkpY"   ,
   "2rsjGbaC0ky8mTOpJrPioWTq0_daw1sX76poUlgCwbI"   ,
   "EkO8dhW0dHEJbvUHlE_VCeuC9uRELOieLZhh7XbUTtA"
 ],

 "cnf": {
   "jwk": {
     "kty": "EC",
     "crv": "P-256",
     "x": "52aDI_ur05n1f_p3jiYGUU82oKZr3m4LsAErM536crQ",
     "y": "ckhZ-KQ5aXNL91R8Eufg1aOf8Z5pZJnIvuCzNGfdnzo"
   }
 }
}
```


(All examples shortened for presentation.)

German EUDI Wallet PID Proposal (1/3)

```
{
   // Base data (SD-JWT VC DM)
   "vct": "https://example.bmi.bund.de/credential/pid/1.0",
      // metadata would define this as an extension of the base type, e.g.,
      https://example.eudi.eu/credential/pid/1.0

   "vct#integrity": "sha256-jo8433ot48....utul8ura33",

   // Base dataset that always needs to be present

   "given_name": "Erika",
   "family_name": "Mustermann",
   "birthdate": "1963-08-12",

   // Additional data

   "source_document_type": "id_card",

   "address": {
       "street_address": "Heidestraße 17",
       "locality": "Köln",
       "postal_code": "51147",
       "country": "DE"
   },
```

German EUDI Wallet PID Proposal (2/3)

```
"nationalities": [
    "DE"
],

"gender": "female",
"birth_family_name": "Gabler",

"place_of_birth": {
    "locality": "Berlin",
    "country": "DE"
},

"also_known_as": "Schwester Agnes",

// Derived claims
"age_equal_or_over": {
    "12": true,
    "14": true,
    "16": true,
    "18": true,
    "21": true,
    "65": false
},
```

German EUDI Wallet PID Proposal (3/3)

```
    // key binding (SD-JWT VC DM)
    "cnf": {
        "jwk": {
            "kty": "EC",
            "crv": "P-256",
            "x": "52aDI_ur05n1f_p3jiYGUU82oKZr3m4LsAErM536crQ",
            "y": "ckhZ-KQ5aXNL91R8Eufg1aOf8Z5pZJnIvuCzNGfdnzo"
        }
    },



    "iat": 1712231700,
    "exp": 1806839700,
    "issuing_authority": "DE",
    "issuing_country": "DE"

}
```