OIDC federation 101

Roland Hedberg @ TNC18

OIDC Services

1. Issuer Discovery
2. Provider Configuration Information
3. Client Registration
4. Authentication/Authorization
5. Access Token
6. User Info

OIDC Services

1. Issuer Discovery
2. Provider Configuration Information
3. Client Registration
4. Authentication/Authorization
5. Access Token
6. User Info

OIDC Identity federation
- governing principals

- Allow dynamic discovery and registration without losing trust
- Enforcement of federation and organisation policies
- Allow delegation of entity registration
- Metadata transport and origin independent
- Self-contained metadata

OIDC Identity federation
- building blocks

- Trusted 3rd party
- Chain of verifiable claims
- Compounded metadata

HOTEL D'ATHÈNES

TOURNARIE, PROPRIÉTAIRE

21, RUE D'ATHÈNES, 21

(GARE SAINT-LAZARE)

TÉLÉPHONE : GUTENBERG 00-28

R. C. SEINE 804.781

PARIS (IXe) LE 15/12/28

Paris, entre Genève
et Bordeaux

Mon ami,

Reçu votre mot en
quittant Genève.

Serai de retour vers le Noël

Téléphone au Lakhond, puis à
Montfort-l'Amaury 89.

A bientôt, en hâte
de vous embrasser

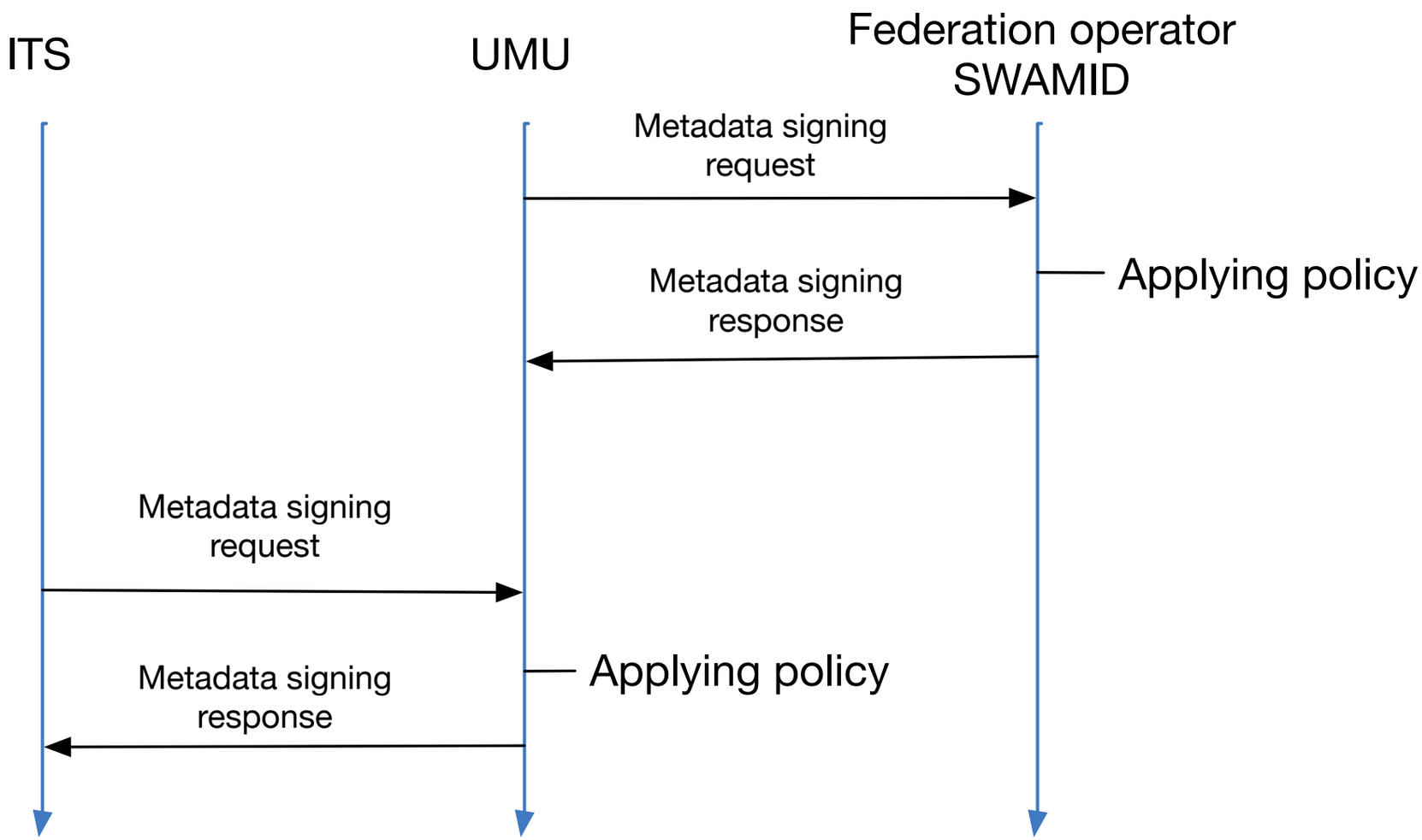
Maurice Merd

Verifiable claims

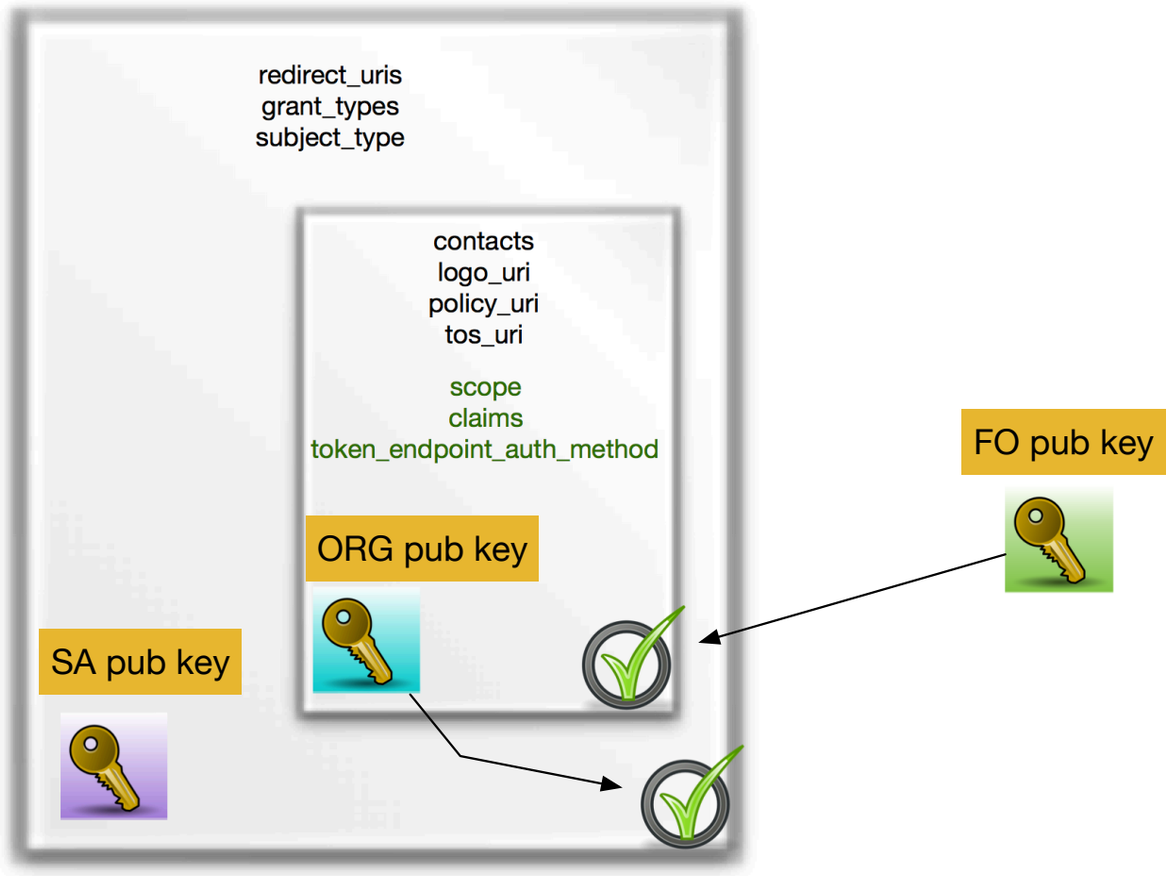
- Well defined set of attributes

- Signed by known entity

OIDC Identity federation - metadata construction



OIDC Identity federation
- compounded metadata statement



Basic components

```
ms_X
    Metadata Statement signing request by X without signing keys
    and signed metadata statements.

SK[X]
    Signing keys that belong to X

X(MS)
    Metadata Statement signed by X
```

Using these basic components, we can now describe a simple signed Metadata Statement as:

```
SWAMID(ms_UMU + SK[UMU])
```

And a slightly more complex as:

```
UMU(ms_ITS + SK[ITS) + SWAMID(ms_UMU + SK[UMU]))
```

Flattening

- What is specified high up can only be made more restrictive further down.

Example:

```
SWAMID: scope=[‘openid’, ‘eduperson’, ‘mail’]

UMU: scope=[‘openid’, ‘mail’]
     scope=[‘openid’, ‘eduperson’, ‘noreduperson’]
```