Cachable OSCORE

draft-amsuess-core-cachable-oscore

Christian Amsüss, Marco Tiloca

2020-07-31

Background

`multicast-notifications`

Comparison with ICNs

OSCON

Caching and OSCORE

$$\left.\begin{array}{l}\text{POST / 2.01}\\ \text{KID and PIV in request}\end{array}\right\}\quad\text{uncachable}$$

... and it's only one client anyway

For every complex problem, there is a solution...

that is simple, neat and ~~wrong~~ insufficient

$$\left.\begin{array}{l}\text{Group OSCORE}\\ \text{FETCH / 2.05}\\ \text{magically hit cache}\end{array}\right\}\quad \text{verification fails}$$

Consensus request

- Pick request sender KID and PIV
- Trust in the request[1]

The ideal candidate to generate a Consensus Request is the server: "Ticket Requests"

[1] It'd be a pity if someone requested `/whom-i-know`, and gave you the response claiming they requested `/whom-to-trust`

Ticket Request example

```
Client                    Proxy              Server

 enc(GET /a, C:1) ------------------------->
 <--------- enc(Try enc(GET /a, S:1), S/C:1)

 enc(GET /a, S:1) ------>
                      (cache hit)
 <--- enc(2.05 data, S:2)
```

Assuming pre-existing multicast setup

`multicast-notifications`'s Phantom Requests are Ticket Requests

1. Great for observations
2. Great for large representations[2]
3. Not so great for everything else

[2] Unless outer-block mode is used. Which you want. In which case see 3.

Magically hitting the cache key

```
Client                                          Proxy

 enc(GET /a, C:1), H(/a) ------------------->
 <- enc(2.05 data, S:2) Resp-For enc(GET /a, S:1)
```

... provided `H(/a)` is derived the same for every request

(actually it's rather hashing the complete plaintext|AAD)

Now that we all agree...

```
Client                                          Proxy

 enc(GET /a, C:H(/a)) -------------------->
 <--------------------- enc(2.05 data, S:1)
```

[3] Also very nice for B.2 mode

Now that we all agree...

```
Client                                          Proxy

 enc(GET /a, C:H(/a)) -------------------->
 <--------------------- enc(2.05 data, S:1)
```

- Hash over all input to encryption (incl. AAD)
- PartIV too short for sufficient hash – ID-Detail[3]
- In group it's encrypt-and-sign – deterministic client with private key known to group members

[3] Also very nice for B.2 mode

Questions

- Practicality
- Cryptography
- Interest in CoRE