

Protecting EST Payloads with OSCORE

draft-ietf-ace-coap-est-oscore-02

Göran Selander, Ericsson

Shahid Raza, RISE

Martin Furuhed, Nexus

Mališa Vučinić, Inria

Timothy Claeys

Status

- Published -02 on 9 July 2023
- Resolves all comments from ACE interim on 06 June 2023
- Goal of the presentation
 - Present the resolutions to the issue from the interim
 - Ask for reviews

Server-generated private keys (1/2)

Agreed action: *Keep the option of server-generated private keys in the draft with added security considerations*

- **Context:** See minutes of ACE interim on 05 June 2023
 - <https://notes.ietf.org/notes-ietf-interim-2023-ace-05-ace>
- **Action taken:**
 - Commit: <https://github.com/ace-wg/est-oscore/commit/f43decc8b9e30ae2c3830c504c2a8907d6b87c5c>
 - See next slide

Server-generated private keys (2/2)

```
391 + ## Server-generated Private Keys
392 +
393 + This document enables the EST client to request generation of private keys and the enrollment of the corresponding public key
    through /skg and /skc functions.
394 + As discussed in {{Section 9 of RFC9148}}, the transport of private keys generated at EST-server is inherently risky.
395 + The use of server-generated private keys may lead to the increased probability of digital identity theft.
396 + Therefore, implementations SHOULD NOT use server-generated private key EST functions.
397 +
398 + A cryptographically secure pseudo-random number generator is required to be available to generate good quality private keys
    on EST-clients.
399 + A cryptographically secure pseudo-random number generator is also a dependency of many security protocols.
400 + This includes the EDHOC protocol, which EST-oscore uses for the mutual authentication of EST-client and EST-server.
401 + If EDHOC is used and a secure pseudo-random number generator is available, the EST-client MUST NOT use server-generated
    private key EST functions.
402 + However, EST-oscore also allows pre-shared OSCORE contexts to be used for authentication, meaning that EDHOC may not
    necessarily be required in the protocol stack of an EST-client.
403 + If EDHOC is not used for authentication, and the EST-client device does not have a cryptographically secure pseudo-random
    number generator, then the EST-client MAY use the server-generated private key functions.
404 +
405 + Although hardware random number generators are becoming dominantly present in modern IoT devices, it has been shown that many
    available hardware modules contain vulnerabilities and do not produce cryptographically secure random numbers.
406 + It is therefore important to use multiple randomness sources to seed the cryptographically secure pseudo-random number
    generator.
```

Security considerations on channel binding (1/2)

- **Context**

- Channel binding is optional, both in EST-oscore and in RFC9148
- Achieved by including tls-unique value in CSR
- This is a mitigation against the Triple SHAKE attack on TLS 1.2
- Attack not viable on (D)TLS 1.3 but channel binding is still optional
- EST-oscore achieves channel binding by including edhoc-unique byte string in the CSR
- edhoc-unique is generated through the EDHOC_Exporter interface
- EDHOC_Exporter relies on the full handshake transcript, not susceptible to Triple SHAKE kind of attack

- **Action taken**

- Channel binding kept optional even though it might not be necessary to protect against Triple SHAKE kind of attack
- Added a security consideration
- See next slide

Security considerations on channel binding (2/2)

```
407 + ## Considerations on Channel Binding
408 +
409 + {{Section 3 of RFC9148}} specifies that the use of channel binding is optional, and achieves it by including the tls-unique
    value in the CSR.
410 + As a rationale, {{Section 9 of RFC9148}} discusses the Triple SHAKE attack: the attack relies on the absence of the server
    certificate as a dependency in the tls-unique value in case of TLS 1.2.
411 + This was mitigated in TLS 1.2 with {{RFC7627}}, and in TLS 1.3 through the tls-exporter API, which computes the value by
    taking into account the full handshake transcript.
412 + Similarly, this specification when used with EDHOC achieves channel binding through the EDHOC-Exporter interface, which also
    relies on the full handshake transcript.
413 + Therefore, authentication based on EDHOC is not susceptible to the same attack as the one considered in {{RFC9148}}.
414 + At the time of the writing, it seems to be safe not to require channel binding and the inclusion of EDHOC-Exporter value in
    CSR.
415 + However, this specification makes channel binding OPTIONAL, as a mitigation against any other attacks that might be
    discovered in future.
```

Next Steps

- More reviews kindly requested
- Aiming at WGLC before IETF 118

Thank you!