Protecting EST Payloads with OSCORE
draft-selander-ace-coap-est-oscore

Göran Selander, Ericsson
Shahid Raza, RISE
Martin Furuhed, Nexus
Mališa Vučinić, Inria
Timothy Claeys

ACE Working Group meeting @ IETF 116
Context

• To make full potential of LAKE/EDHOC requires matching certificate enrollment
  • In particular, enrolment of certificates for static DH public keys
  • Target devices typically have EDHOC-OSCORE implementation

• RFC 9148
  • Published in April 2022, output of ACE
  • EST-coaps: Specifies Enrollment over Secure Transport (EST) with coaps
  • Follows closely the EST design, security with DTLS
  • Profiles EST for constrained environments

• ACE charter
  • “The Working Group will examine how to use Constrained Application Protocol (CoAP) as a transport medium for certificate enrollment protocols, such as EST and CMPv2, ...”
EST-oscore (this draft)

• Old draft, first version published in March 2017
• Protects EST payloads with OSCORE
• Follows the structure of RFC 9148, EDHOC-oscore instead of DTLS
• Agreement in a previous ACE WG interim meeting to work on this draft, but to complete EST-coaps first
• Revived for IETF 116
• Latest update includes support for enrollment of static DH keys
## Operational differences with EST-coaps

<table>
<thead>
<tr>
<th></th>
<th>EST-coaps</th>
<th>EST-oscore</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message protection</td>
<td>DTLS Record</td>
<td>OSCORE</td>
</tr>
<tr>
<td>Mutual authentication</td>
<td>DTLS handshake</td>
<td>EDHOC</td>
</tr>
<tr>
<td>EST-server ↔ Registrar Trust Relation</td>
<td>Required</td>
<td>Not required</td>
</tr>
</tbody>
</table>
## Protocol Layering

<table>
<thead>
<tr>
<th>EST messages</th>
</tr>
</thead>
<tbody>
<tr>
<td>EDHOC</td>
</tr>
<tr>
<td>OSCORE</td>
</tr>
<tr>
<td>CoAP or HTTP</td>
</tr>
<tr>
<td>UDP or TCP</td>
</tr>
</tbody>
</table>
Authentication

• Mutual authentication required between EST-oscore client and server
• Uses EDHOC (draft-ietf-lake-edhoc)
• Authentication based on certificates
• Channel binding using “edhoc-unique”
  • edhoc-unique = EDHOC-Exporter(TBD1, "EDHOC Unique", length)
  • Byte string added as *challengePassword* of PKCS#10 Request
• Optimizations
  • Combined EDHOC message_3 and OSCORE request (draft-ietf-core-oscore-edhoc)
  • Certificates may be CBOR-encoded (draft-ietf-cose-cbor-encoded-cert)
  • Certificates may be referenced (draft-ietf-cose-x509)
  • PKCS#10 response may be a reference to the enrolled certificate
## EST Functions

<table>
<thead>
<tr>
<th></th>
<th>EST-coaps</th>
<th>EST-oscore</th>
</tr>
</thead>
<tbody>
<tr>
<td>/crt</td>
<td>MUST</td>
<td>MUST</td>
</tr>
<tr>
<td>/sen</td>
<td>MUST</td>
<td>MUST</td>
</tr>
<tr>
<td>/sren</td>
<td>MUST</td>
<td>MUST</td>
</tr>
<tr>
<td>/skg</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
</tr>
<tr>
<td>/skc</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
</tr>
<tr>
<td>/att</td>
<td>OPTIONAL</td>
<td>OPTIONAL</td>
</tr>
</tbody>
</table>
Enrollment of Static DH Keys

- EDHOC supports authentication using static DH keys
  - The most efficient EDHOC authentication method in terms of message size
- This draft adds the support for the enrollment of static DH keys

Procedure
- Client obtains CA’s DH key using /crt
- Client generates the DH keypair following the DH group parameters of the CA
- Client follows the steps in Section 4 of RFC 6955 to sign PKCS#10 object
- Uses OSCORE KDF and MAC algorithms
Next Steps

• Complete the Security and Privacy Considerations section
• Reviews?
Thank you!