Careful with MAc-then-SIGn: A Computational Analysis of the EDHOC Lightweight Authenticated Key Exchange Protocol

Felix Günther and Marc Ilunga

March 27, 2023
Proliferation of low-powered devices

- Limited computing power
- Bandwidth constraints
- Plagued by vulnerabilities

---

1Burgess, “Smart dildos and vibrators keep getting hacked – but Tor could be the answer to safer connected sex”. 

Image by Moritz Kindler
Proliferation of low-powered devices

- Limited computing power
- Bandwidth constraints
- Plagued by vulnerabilities

---

1Burgess, “Smart dildos and vibrators keep getting hacked – but Tor could be the answer to safer connected sex”. 
Authenticated Key Exchange (AKE) for constrained environments remains an issue

- Missing satisfactory solutions
- EDHOC: a proposal by the IETF LAKE WG.
- Use case: OSCORE¹ protocol (secure transport)
- 4 mutual authentication methods (static DH and/or Signature)
  - This talk: SIG-SIG
  - Design similar to TLS1.3 and based on SIGMA²

¹Selander et al., Object Security for Constrained RESTful Environments (OSCORE).
²Krawczyk, “SIGMA: The “SIGn-and-MAc” Approach to Authenticated Diffie-Hellman and Its Use in the IKE Protocols”
Authenticated Key Exchange (AKE) for constrained environments remains an issue

- Missing satisfactory solutions
- EDHOC: a proposal by the IETF LAKE WG.
  - Use case: OSCORE\(^1\) protocol (secure transport)
  - 4 mutual authentication methods (static DH and/or Signature)
    - This talk: SIG-SIG
    - Design similar to TLS1.3 and based on SIGMA\(^2\)

---

\(^1\)Selander et al., *Object Security for Constrained RESTful Environments (OSCORE)*.

\(^2\)Krawczyk, “SIGMA: The “SIGn-and-MAc” Approach to Authenticated Diffie-Hellman and Its Use in the IKE Protocols”
Authenticated Key Exchange (AKE) for constrained environments remains an issue

- Missing satisfactory solutions
- EDHOC: a proposal by the IETF LAKE WG.
- Use case: OSCORE\(^1\) protocol (secure transport)
  - 4 mutual authentication methods (static DH and/or Signature)
    - This talk: SIG-SIG
    - Design similar to TLS1.3 and based on SIGMA\(^2\)

---

\(^1\)Selander et al., *Object Security for Constrained RESTful Environments (OSCORE)*.

\(^2\)Krawczyk, “SIGMA: The “SIGn-and-MAc” Approach to Authenticated Diffie-Hellman and Its Use in the IKE Protocols”
Authenticated Key Exchange (AKE) for constrained environments remains an issue

- Missing satisfactory solutions
- EDHOC: a proposal by the IETF LAKE WG.
- Use case: OSCORE\(^1\) protocol (secure transport)
- 4 mutual authentication methods (static DH and/or Signature)
  - This talk: SIG-SIG
  - Design similar to TLS1.3 and based on SIGMA\(^2\)

---

\(^1\)Selander et al., *Object Security for Constrained RESTful Environments (OSCORE)*.
\(^2\)Krawczyk, “SIGMA: The “SIGn-and-MAC” Approach to Authenticated Diffie-Hellman and Its Use in the IKE Protocols”.
Authenticated Key Exchange (AKE) for constrained environments remains an issue

- Missing satisfactory solutions
- EDHOC: a proposal by the IETF LAKE WG.
- Use case: OSCORE\(^1\) protocol (secure transport)
- 4 mutual authentication methods (static DH and/or Signature)
  - This talk: SIG-SIG
  - Design similar to TLS1.3 and based on SIGMA\(^2\)

---

\(^1\) Selander et al., Object Security for Constrained RESTful Environments (OSCORE).

\(^2\) Krawczyk, “SIGMA: The “SIGn-and-MAC” Approach to Authenticated Diffie-Hellman and Its Use in the IKE Protocols”.
Authenticated Key Exchange (AKE) for constrained environments remains an issue

- Missing satisfactory solutions
- EDHOC: a proposal by the IETF LAKE WG.
- Use case: OSCORE\textsuperscript{1} protocol (secure transport)
- 4 mutual authentication methods (static DH and/or Signature)
  - This talk: SIG-SIG
  - Design similar to TLS1.3 and based on SIGMA\textsuperscript{2}

---

\textsuperscript{1}Selander et al., *Object Security for Constrained RESTful Environments (OSCORE)*.

\textsuperscript{2}Krawczyk, “SIGMA: The “SIGn-and-MAc” Approach to Authenticated Diffie-Hellman and Its Use in the IKE Protocols”.
TLS 1.3 is a secure authenticated key exchange protocol

Q: Why not simply use TLS 1.3?
A: It is not lightweight enough.
Context and Motivation

TLS 1.3 is a secure authenticated key exchange protocol

- Q: Why not simply use TLS 1.3?
- A: It is not lightweight enough.
(D)TLS 1.3 is not lightweight: up to 7x bandwidth usage

Total protocol size (bytes)\(^1\)

\begin{itemize}
  \item DTLS 1.3 (ECDHE) \hspace{1cm} 880
  \item TLS 1.3 (ECDHE) \hspace{1cm} 789
  \item EDHOC (STAT-STAT) \hspace{1cm} 101
\end{itemize}

\footnote{empty citation.}
EDHOC in SIG-SIG Mode: An AKE based on Diffie-Hellman

\[ \text{INITIATOR (1)} \]
\[ x \leftarrow \mathbb{Z}_q; G_x \leftarrow xG \]

\[ \text{RESPONDER (1)} \]
\[ y \leftarrow \mathbb{Z}_q; G_y \leftarrow yG \]

\[ \text{INITIATOR (2)} \]
EDHOC in SIG-SIG Mode: An AKE based on Diffie-Hellman

**INITIATOR (1)**

\[ x \leftarrow \mathbb{Z}_q; \ G_x \leftarrow xG \]

**RESPONDER (1)**

\[ y \leftarrow \mathbb{Z}_q; \ G_y \leftarrow yG \]

**INITIATOR (2)**
EDHOC in SIG-SIG Mode: An AKE based on Diffie-Hellman

**Initiator (1)**

\[ x \leftarrow \mathbb{Z}_q; \; G_x \leftarrow xG \]

**Responder (1)**

\[ y \leftarrow \mathbb{Z}_q; \; G_y \leftarrow yG \]
\[ \tau_2 \leftarrow \text{MAC}_{K_m}(id_B) \]
\[ \sigma_2 \leftarrow \text{Sign}(sk_R, \tau_2 \ldots) \]

**Initiator (2)**

\[ G_x \]
\[ G_y, id_B, \sigma_2 \]
**EDHOC in SIG-SIG Mode: An AKE based on Diffie-Hellman**

**Initiator (1)**
\[ x \overset{\$}{\leftarrow} \mathbb{Z}_q; \ G_x \leftarrow xG \]

**Responder (1)**
\[ y \overset{\$}{\leftarrow} \mathbb{Z}_q; \ G_y \leftarrow yG \]
\[ \tau_2 \leftarrow \text{MAC}_{K_m}(id_B) \]
\[ \sigma_2 \leftarrow \text{Sign}(sk_R, \tau_2 \ldots) \]

**Initiator (2)**
\[ \tau_3 \leftarrow \text{MAC}_{K_m}(id_W) \]
\[ \sigma_3 \leftarrow \text{Sign}(sk_I, \tau_3 \ldots) \]
EDHOC in SIG-SIG Mode: An AKE with identity protection

$G_x$  

$G_y, \{id_B, \sigma_2\}_{K_2}$  

$\{id_W, \sigma_3\}_{K_3, IV_3}$  

**INITIATOR (1)**  
$x \xleftarrow{\$} Z_q; G_x \leftarrow xG$

**RESPONDER (1)**  
$y \xleftarrow{\$} Z_q; G_y \leftarrow yG$
$\tau_2 \leftarrow MAC_{K_m}(id_B)$
$\sigma_2 \leftarrow \text{Sign}(sk_R, \tau_2 \ldots)$

**INITIATOR (2)**  
$\tau_3 \leftarrow MAC_{K_m}(id_W)$
$\sigma_3 \leftarrow \text{Sign}(sk_I, \tau_3 \ldots)$
EDHOC in SIG-SIG Mode: An AKE \approx \text{SIGMA}

**Initiator (1)**

- $x \leftarrow Z_q; \ G_x \leftarrow xG$
- $\tau_2 \leftarrow \text{MAC}_{K_m}(\text{id}_B)$
- $\sigma_2 \leftarrow \text{Sign}(sk_R, \tau_2 \ldots)$

**Responder (1)**

- $y \leftarrow Z_q; \ G_y \leftarrow yG$
- $\tau_2 \leftarrow \text{MAC}_{K_m}(\text{id}_B)$
- $\sigma_2 \leftarrow \text{Sign}(sk_R, \tau_2 \ldots)$

**Initiator (2)**

- $\tau_3 \leftarrow \text{MAC}_{K_m}(\text{id}_W)$
- $\sigma_3 \leftarrow \text{Sign}(sk_I, \tau_3 \ldots)$
EDHOC SIG-SIG $\approx$ SIGMA: MAC "under" signature

$G_x \rightarrow G_y, \text{id}_B, \sigma_2$

$\text{id}_W, \sigma_3 \leftarrow \tau_2 \leftarrow \text{MAC}_{K_m}(\text{id}_B)$

$\sigma_2 \leftarrow \text{Sign} (sk_R, \tau_2 \ldots)$
EDHOC SIG-SIG $\approx$ SIGMA: Abbreviated identities

- $\text{id}_X$ Short credential identifier for $X$
- $\text{size} \ll \text{X.509 Cert}$
- need not be unique

*applications MUST NOT assume that 'kid' values are unique and several keys associated with a 'kid' may need to be checked [by the recipient] before the correct one is found.*
EDHOC SIG-SIG ≈ SIGMA: Abbreviated identities

- $id_X$: Short credential identifier for $X$
- Size $\ll$ X.509 Cert
- Need not be unique

Applications MUST NOT assume that 'kid' values are unique and several keys associated with a 'kid' may need to be checked [by the recipient] before the correct one is found.

---

1Selander, Mattsson, and Palombini, *Ephemeral Diffie-Hellman Over COSE (EDHOC)* – draft-ietf-lake-edhoc-17, Section 3.5.3.
Abbreviated identifiers introduce new challenges

**RunInit2**

\[
\text{...}
\text{foreach } (U, pk_U) \text{ with id}_U = \text{id}_B :\]

\[
\tau_2 \leftarrow \text{MAC}(\text{id}_U, \ldots)
\]

\[
\text{if } \text{Sig.Vf}(pk_U, \tau_2, \ldots, \sigma_2) = 1 :\]

\[
\text{pid} \leftarrow U; \text{ endforeach}
\]

\[
\text{abort } \text{if } \text{pid} = \bot
\]

\[
\text{...}
\]
Abbreviated identifiers introduce new challenges

```
RunInit2

... 

foreach \((U, pk_U)\) with \(id_U = id_B\):
    \(\tau_2 \leftarrow \text{MAC}(id_U, \ldots)\)
    if \(\text{Sig.Vf}(pk_U, \tau_2 \ldots, \sigma_2) = 1\):
        pid \leftarrow U; \text{ endforeach}
    abort if pid = ⊥
... 
```
The EDHOC SIG-SIG protocol

Abbreviated identifiers introduce new challenges

$G_x, G_y, \text{id}_B, \sigma_2$

What if an attacker also uses $\text{id}_B$?
Duplicate Signature Key Selection attacks.

**RunInit2**

...  

foreach $(U, pk_U)$ with $\text{id}_U = \text{id}_B$:

$\tau_2 \leftarrow \text{MAC}(\text{id}_U, \ldots)$

if $\text{Sig.Vf}(pk_U, \tau_2 \ldots, \sigma_2) = 1$:

$\text{pid} \leftarrow U$;  
endforeach

abort  if $\text{pid} = \bot$

...
DSKS attacks: Signature unforgeability is not enough

- EUF-CMA $\not\Rightarrow$ cannot find $(pk^*, m^*)$:
  $$\text{Sig.Vf}(pk^*, m^*, \sigma) = 1 \text{ (For honestly generated } \sigma)$$

- Andrew Ayer, 2015: DSKS attack in the ACME protocol with RSA signatures impacts Let’s Encrypt
DSKS attacks: Signature unforgeability is not enough

- EUF-CMA $\Rightarrow$ cannot find $(pk^*, m^*)$: 
  $$\text{Sig.Vf}(pk^*, m^*, \sigma) = 1 \text{ (For honestly generated } \sigma)$$

- Andrew Ayer, 2015: DSKS attack in the ACME protocol with RSA signatures impacts Let’s Encrypt
The EDHOC SIG-SIG protocol

**DSKS vs SIGMA**

\[ \text{NewUser}(sk_A, pk_A, id_W) \]
\[ \forall f(pk_A, *, *) = 1 \]

I am talking to user A.

The secret top-up code is: BADH-350

What about EDHOC?
The EDHOC SIG-SIG protocol

DSKS vs SIGMA

\[ \text{NEWUSER}(sk_A, pk_A, id_W) \]
\[ \forall f(pk_A, *, *) = 1 \]

\[ G_x \]

\[ G_y, id_B, \sigma_2 \]

\[ id_W, \sigma_3 \]

The secret top-up code is: BADH-350

I am talking to user A.

What about EDHOC?

I am talking to user A.

The secret top-up code is: BADH-350

What about EDHOC?
The EDHOC SIG-SIG protocol

**DSKS vs SIGMA**

\[ \text{NEWUSER}(sk_A, pk_A, id_W) \]
\[ Vf(pk_A, *, *) = 1 \]

I am talking to user \( A \).

What about EDHOC?

The secret top-up code is: BADH-350

Felix Günther and Marc Ilunga
**DSKS vs SIGMA: identity misbinding (w/ strong attackers)**

NEWUSER($sk_A, pk_A, id_W$)

$Vf(pk_A, *, *) = 1$

I am talking to user $A$.

The secret top-up code is: BADH-350

What about EDHOC?
DSKS vs SIGMA: identity misbinding (w/ strong attackers)

\[ \text{NEWUSER}(sk_A, pk_A, id_W) \]
\[ Vf(pk_A, *, *) = 1 \]
\[ id_W, (pk_I, sk_I) \]
\[ id_B, (pk_R, sk_R) \]

I am talking to user A. BADH-350 checks out! I’ll send 350 cryptos to user A...

What about EDHOC?

The secret top-up code is: BADH-350
EDHOC provides strong authentication guarantees even under colliding identifiers

- Assuming universal exclusive ownership$^1$ of the signature schemes
- S-UEO for signature scheme $\Sigma$ (informal):
  - Key pair: $(pk, sk) \xleftarrow{\$} \Sigma.KGen()$
  - Adversary $A$ obtains $set(m_i, \sigma_i)$ (produced by $sk$)
  - Goal of $A$: Produce $(pk^*, m^*)$ s.t $Vf(pk^*, m^*, \sigma_j) = 1$ and $pk \neq pk^*$
  - S-UEO $\implies A$ cannot succeed.

---

$^1$Pornin and Stern, "Digital Signatures Do Not Guarantee Exclusive Ownership".
Security Model: Multi-Stage Key Exchange Model

\[ \text{Oracles} = \begin{cases} \text{Send} \\ \text{NewSession} \\ \text{RevLongTermKey} \\ \text{RevSessionKey} \\ \text{NewUser}^1 (sk, pk, id) \end{cases} \]

\[ G_{MSKE} \]

---

1Boyd et al., “ASICS: Authenticated Key Exchange Security Incorporating Certification Systems”. 
MSKE: Security goals

- Key indistinguishability
- Forward security
- Explicit authentication: When a session accepts with an authenticated peer, there is indeed a corresponding session of that peer.
MSKE: Security goals

- Key indistinguishability
- Forward security
  - Explicit authentication: When a session accepts with an authenticated peer, there is indeed a corresponding session of that peer.
MSKE: Security goals

- Key indistinguishability
- Forward security
- Explicit authentication: When a session accepts with an authenticated peer, there is indeed a corresponding session of that peer.
MSKE Security of EDHOC SIG-SIG

Let $A$ be an MSKE adversary. For at most $n_U$ users and $n_S$ sessions, there exists adversaries $B_j$ such that:

$$\text{Adv}_{MSKE}^{\mathcal{A}}(\text{EDHOC-Sig-Sig}) \leq \frac{n_S^2}{q} + \text{Adv}_{CR}^{\mathcal{B}_4}(H) + 4n_S \left( n_U \cdot \text{Adv}_{B_{I,2}}^{\text{SUF-CMA}}(\text{Sig}) + \text{Adv}_{B_{I,4}}^{\text{S-UEO}}(\text{Sig}) \right) + 4n_S \left( n_U \cdot \text{Adv}_{B_{II,A2}}^{\text{EUF-CMA}}(\text{Sig}) + \text{Adv}_{B_{II,B2}}^{\text{snPRF-ODH}}(\text{Extract}) + \text{Adv}_{B_{II,B3}}^{\text{PRF}}(\text{Expand}) \right)$$

Assumption scheme

<table>
<thead>
<tr>
<th>Assumption</th>
<th>scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collision resistance</td>
<td>SHA2, Shake128</td>
</tr>
<tr>
<td>SUF-CMA</td>
<td>Ed25519</td>
</tr>
<tr>
<td></td>
<td>ECDSA</td>
</tr>
<tr>
<td>S-UEO</td>
<td>Ed25519</td>
</tr>
<tr>
<td></td>
<td>ECDSA</td>
</tr>
<tr>
<td>EUF-CMA</td>
<td>Ed25519</td>
</tr>
<tr>
<td></td>
<td>ECDSA</td>
</tr>
<tr>
<td>PRF-ODH</td>
<td>HKDF.Extract</td>
</tr>
<tr>
<td></td>
<td>KMAC</td>
</tr>
<tr>
<td>PRF</td>
<td>HKDF.Expand</td>
</tr>
<tr>
<td></td>
<td>KMAC</td>
</tr>
</tbody>
</table>
MSKE Security of EDHOC SIG-SIG

Let $\mathcal{A}$ be an MSKE adversary. For at most $n_U$ users and $n_S$ sessions, there exists adversaries $B_j$ such that:

$$\text{Adv}_{\mathcal{A}}^{\text{MSKE}}(\text{EDHOC-Sig-Sig}) \leq \frac{n_S^2}{q} + \text{Adv}_{B_4}^{\text{CR}}(H) + 4n_S \left( n_U \cdot \text{Adv}_{B_{1.2}}^{\text{SUF-CMA}}(\text{Sig}) + \text{Adv}_{B_{1.4}}^{\text{S-UEO}}(\text{Sig}) \right) + 4n_S \left( n_U \cdot \text{Adv}_{B_{1.2}}^{\text{EUF-CMA}}(\text{Sig}) + \text{Adv}_{B_{1.2}}^{\text{PRF-ODH}}(\text{Extract}) + \text{Adv}_{B_{1.2}}^{\text{PRF}}(\text{Expand}) \right)$$
ECDSA might be fine for EDHOCC

- S-UEO ✗: EDHOCC includes the pub key alongside messages to be signed (√)
- SUF-CMA ✗: Implementations could use “canonical” signatures (√ ?).
Positive collaboration with the LAKE working group

- Numerous contributions to EDHOC by several parties
  - Jacomme et al.: Full symbolic analysis of latest draft\(^1\)
  - Cottier & Pointcheval: Computation analysis of STAT-STAT\(^2\)
  - Norman et al.: Early symbolic analysis\(^3\)
- Reminiscent of development of TLS 1.3

---

\(^1\) Jacomme et al., “A comprehensive, formal and automated analysis of the EDHOC protocol”.

\(^2\) Cottier and Pointcheval, *Security Analysis of the EDHOC protocol*.

\(^3\) Norrman, Sundararajan, and Bruni, “Formal Analysis of EDHOC Key Establishment for Constrained IoT Devices”.

Chasing a moving target

- Worked through drafts (12-17)
- In an ideal world: tooling for automated proofs
Chasing a moving target

- Worked through drafts (12-17)
- In an ideal world: tooling for automated proofs
Contributions overview: Insights from our computational analysis

- Dedicated session key ($\text{PRK}_{out}$) added in draft 14 (with Jacomme et al.\textsuperscript{1})
- Full credentials in transcript hashes in key derivation.
- Transcript hashes from plaintext instead of ciphertexts
- Key separation in key derivation

\textsuperscript{1} Jacomme et al., “A comprehensive, formal and automated analysis of the EDHOC protocol”. 
Conclusion

EDHOC is a LAKE for constrained environments with new security challenges

Our contributions:
- Strong security model for the LAKE setting
- Security analysis and proof that EDHOC(SIG-SIG) is a secure LAKE in a strong adversarial model
- Design contributions to EDHOC

LAKE WG highly welcoming of security analysis and inputs

See EuroS&P 2023 Paper (eprint ia.cr/2022/1705)
marc.ilunga@trailofbits.com
Conclusion

- EDHOC is a LAKE for constrained environments with new security challenges
- Our contributions:
  - Strong security model for the LAKE setting
  - Security analysis and proof that EDHOC(SIG-SIG) is a secure LAKE in a strong adversarial model
  - Design contributions to EDHOC
- LAKE WG highly welcoming of security analysis and inputs

See EuroS&P 2023 Paper
(eprint ia.cr/2022/1705)
marc.ilunga@trailofbits.com