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

Image by Moritz Kindler

- Limited computing power
- Bandwidth constraints
- Plagued by vulnerabilities[1]

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

Proliferation of low-powered devices

Image by Moritz Kindler

- Limited computing power
- Bandwidth constraints
- Plagued by vulnerabilities[1]

[1] Burgess, "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: 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[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".

TLS 1.3 is a secure authenticated key exchange protocol

TLS 1.3

- Q: Why not simply use TLS 1.3?
- A: It is not lightweight enough.

TLS 1.3 is a secure authenticated key exchange protocol

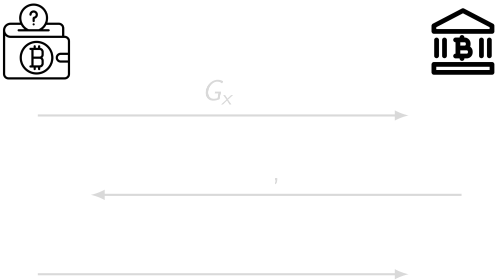
- Q: Why not simply use TLS 1.3?
- A: It is not lightweight enough.

(D)TLS 1.3 is not lightweight: up to 7x bandwidth usage

	Total protocol size (bytes)[1]
DTLS 1.3 (ECDHE)	880
TLS 1.3 (ECDHE)	789
EDHOC (STAT-STAT)	101

[1] **empty citation**.

EDHOC in SIG-SIG Mode: An AKE based on Diffie-Hellman



Initiator (1)

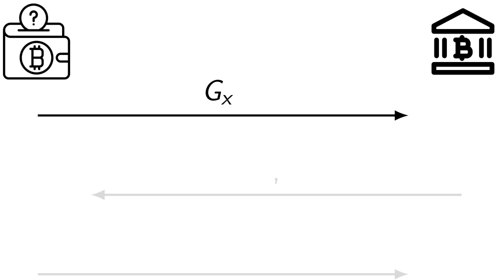
$x \xleftarrow{\$} \mathbb{Z}_q;\ G_x \leftarrow xG$

Responder(1)

$y \xleftarrow{\$} \mathbb{Z}_q;\ G_y \leftarrow yG$

Initiator(2)

EDHOC in SIG-SIG Mode: An AKE based on Diffie-Hellman



INITIATOR (1)

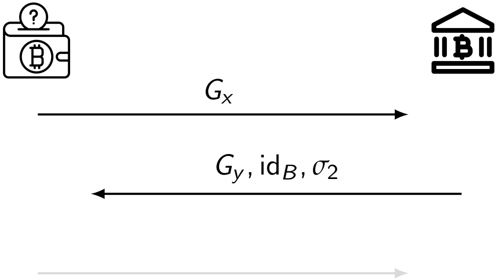
$$x \xleftarrow{\$} \mathbb{Z}_q;\ G_x \leftarrow xG$$

RESPONDER(1)

$$y \xleftarrow{\$} \mathbb{Z}_q;\ G_y \leftarrow yG$$

INITIATOR(2)

EDHOC in SIG-SIG Mode: An AKE based on Diffie-Hellman



<u>INITIATOR (1)</u>

$x \xleftarrow{\$} \mathbb{Z}_q; G_x \leftarrow xG$

<u>RESPONDER(1)</u>

$y \xleftarrow{\$} \mathbb{Z}_q; G_y \leftarrow yG$

$\tau_2 \leftarrow \mathrm{MAC}_{K_m}(\mathrm{id}_B)$

$\sigma_2 \leftarrow \mathrm{Sign}(sk_R, \tau_2 \dots)$

<u>INITIATOR(2)</u>

EDHOC in SIG-SIG Mode: An AKE based on Diffie-Hellman

<u>INITIATOR (1)</u>

$x \xleftarrow{\$} \mathbb{Z}_q;\ G_x \leftarrow xG$

<u>RESPONDER(1)</u>

$y \xleftarrow{\$} \mathbb{Z}_q;\ G_y \leftarrow yG$

$\tau_2 \leftarrow \mathsf{MAC}_{K_m}(\mathsf{id}_B)$

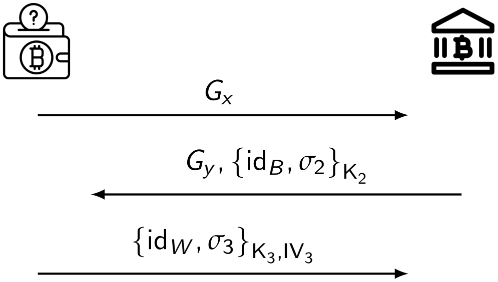
$\sigma_2 \leftarrow \mathsf{Sign}(sk_R, \tau_2 \ldots)$

<u>INITIATOR(2)</u>

$\tau_3 \leftarrow \mathsf{MAC}_{K_m}(\mathsf{id}_W)$

$\sigma_3 \leftarrow \mathsf{Sign}(sk_I, \tau_3 \ldots)$

EDHOC in SIG-SIG Mode: An AKE with identity protection



<u>INITIATOR (1)</u>

$x \xleftarrow{\$} \mathbb{Z}_q;\ G_x \leftarrow xG$

<u>RESPONDER(1)</u>

$y \xleftarrow{\$} \mathbb{Z}_q;\ G_y \leftarrow yG$

$\tau_2 \leftarrow \mathsf{MAC}_{K_m}(\mathsf{id}_B)$

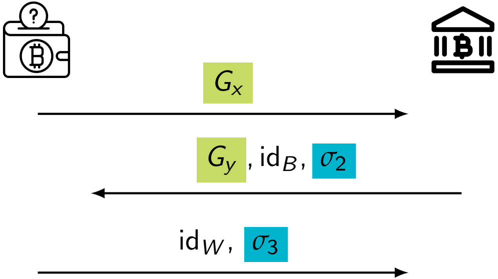
$\sigma_2 \leftarrow \mathsf{Sign}(sk_R, \tau_2 \ldots)$

<u>INITIATOR(2)</u>

$\tau_3 \leftarrow \mathsf{MAC}_{K_m}(\mathsf{id}_W)$

$\sigma_3 \leftarrow \mathsf{Sign}(sk_I, \tau_3 \ldots)$

EDHOC in SIG-SIG Mode: An AKE $\approx$ SIGMA



$\underline{\textsc{Initiator}\ (1)}$

$x \xleftarrow{\$} \mathbb{Z}_q;\ G_x \leftarrow xG$

$\underline{\textsc{Responder}(1)}$

$y \xleftarrow{\$} \mathbb{Z}_q;\ G_y \leftarrow yG$

$\tau_2 \leftarrow \mathsf{MAC}_{K_m}(\mathrm{id}_B)$

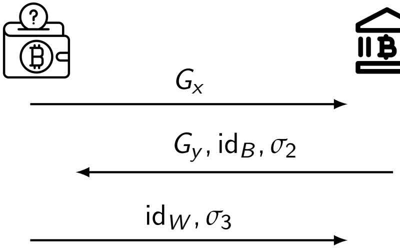
$\sigma_2 \leftarrow \mathsf{Sign}(sk_R, \tau_2 \ldots)$

$\underline{\textsc{Initiator}(2)}$

$\tau_3 \leftarrow \mathsf{MAC}_{K_m}(\mathrm{id}_W)$

$\sigma_3 \leftarrow \mathsf{Sign}(sk_I, \tau_3 \ldots)$

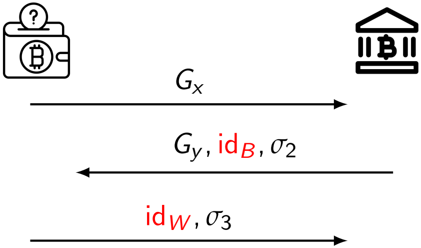
EDHOC SIG-SIG ≈ SIGMA: MAC "under" signature



$\tau_2 \leftarrow \mathrm{MAC}_{K_m}(\mathrm{id}_B)$

$\sigma_2 \leftarrow \mathrm{Sign}(sk_R, \tau_2 \ldots)$

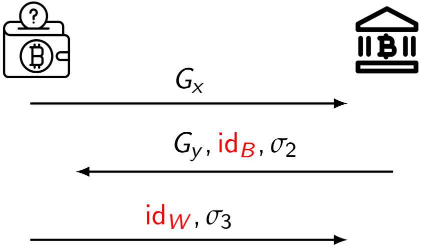
EDHOC SIG-SIG ≈ SIGMA: Abbreviated identities



- id_X Short credential identifier for X
- size $\ll$ X.509 Cert
- need not be unique[1]

  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

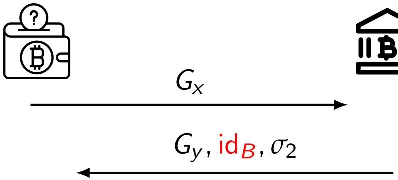


- id_X Short credential identifier for X
- size $\ll$ X.509 Cert
- need not be unique[1]

  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.

[1] Selander, Mattsson, and Palombini, *Ephemeral Diffie-Hellman Over COSE (EDHOC) – draft-ietf-lake-edhoc-17*, Section 3.5.3.

Abbreviated identifiers introduce new challenges



RunInit2

$\ldots$

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

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

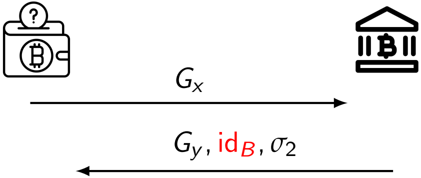
  if $\mathsf{Sig.Vf}(pk_U, \tau_2 \ldots, \sigma_2) = 1$:

    $\mathsf{pid} \leftarrow U$; **endforeach**

abort if $\mathsf{pid} = \perp$

$\ldots$

Abbreviated identifiers introduce new challenges



RUNINIT2

$\ldots$

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

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

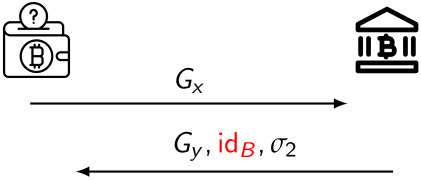
$\quad$ **if** $\mathrm{Sig.Vf}(pk_U, \tau_2 \ldots, \sigma_2) = 1$:

$\quad\quad \mathrm{pid} \leftarrow U;$ **endforeach**

abort if $\mathrm{pid} = \perp$

$\ldots$

Abbreviated identifiers introduce new challenges



What if an attacker also uses id_B?

Duplicate Signature Key Selection attacks.

RunInit2

$\ldots$

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

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

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

$\quad\quad \mathrm{pid} \leftarrow U;$ **endforeach**

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

$\ldots$

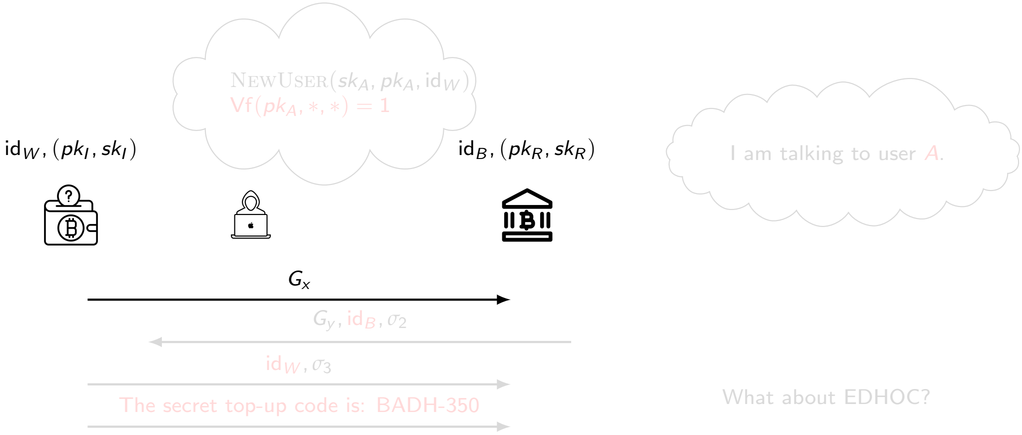
DSKS attacks: Signature unforgeability is not enough

- EUF-CMA $\not\Rightarrow$ cannot find (pk^*, m^*):
  $\mathsf{Sig.Vf}(pk^*, m^*, \sigma) = 1$ (For *honestly generated* σ)
- Andrew Ayer, 2015: DSKS attack in the ACME protocol with RSA signatures impacts Let's Encrypt

DSKS attacks: Signature unforgeability is not enough

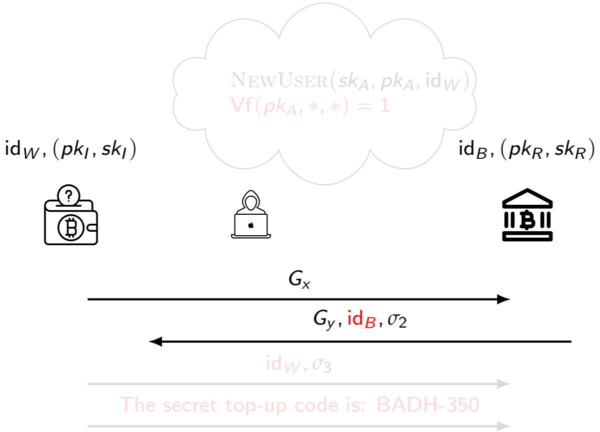
- EUF-CMA $\not\Rightarrow$ cannot find (pk^*, m^*):
  $\mathsf{Sig.Vf}(pk^*, m^*, \sigma) = 1$ (For *honestly generated* σ)
- Andrew Ayer, 2015: DSKS attack in the ACME protocol with RSA signatures impacts Let's Encrypt

DSKS vs SIGMA



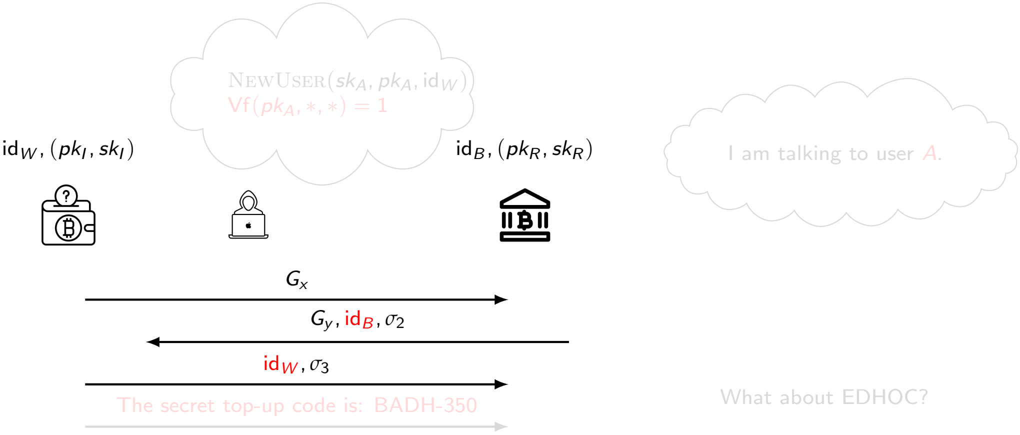
What about EDHOC?

DSKS vs SIGMA



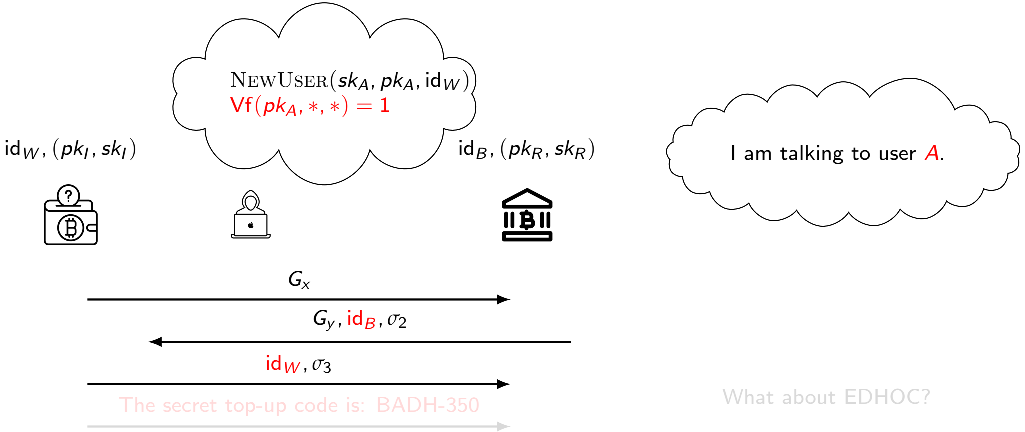
What about EDHOC?

DSKS vs SIGMA

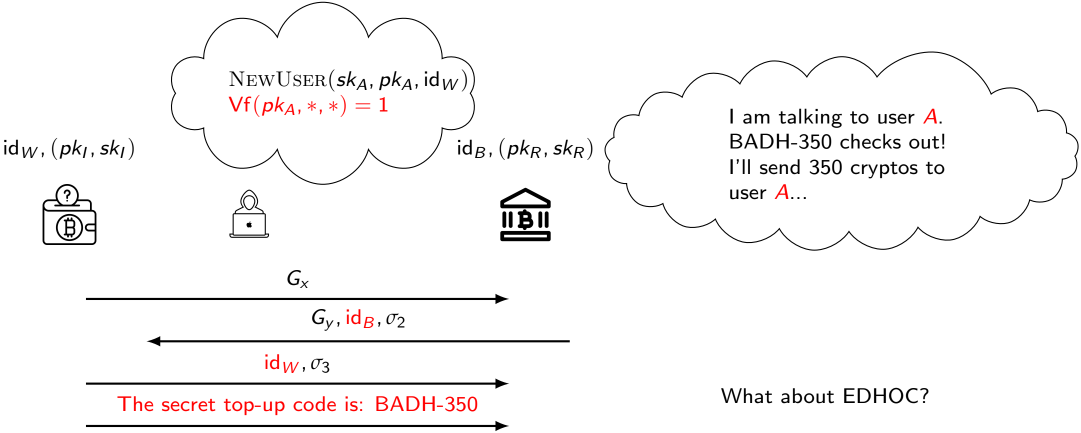


What about EDHOC?

DSKS vs SIGMA: identity misbinding (w/ strong attackers)



DSKS vs SIGMA: identity misbinding (w/ strong attackers)



What about EDHOC?

EDHOC provides strong authentication guarantees even under colliding identifiers

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

[1] Pornin and Stern, "Digital Signatures Do Not Guarantee Exclusive Ownership".

Security Model: Multi-Stage Key Exchange Model

$\mathbf{G}_{MSKE}$

$$\mathcal{O}\text{racles} = \left\{ \begin{array}{l} \textsc{Send} \\ \textsc{NewSession} \\ \textsc{RevLongTermKey} \\ \textsc{RevSessionKey} \\ \textsc{NewUser}^{1}(sk, pk, \mathrm{id}) \end{array} \right\}$$

$\mathcal{A}$

[1] Boyd 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

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 $\mathcal{B}_j$ such that:

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

Assumption	scheme	
Collision resistance	SHA2, Shake128	✓
SUF-CMA	Ed25519	✓
	ECDSA	✗
S-UEO	Ed25519	✓
	ECDSA	✗
EUF-CMA	Ed25519	✓
	ECDSA	✓
PRF-ODH	HKDF.Extract	✓
	KMAC	(?)
PRF	HKDF.Expand	✓
	KMAC	✓

MSKE Security of EDHOC SIG-SIG

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 $\mathcal{B}_j$ such that:

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

Assumption	scheme	
Collision resistance	SHA2, Shake128	✓
SUF-CMA	Ed25519	✓
	ECDSA	✗
S-UEO	Ed25519	✓
	ECDSA	✗
EUF-CMA	Ed25519	✓
	ECDSA	✓
PRF-ODH	HKDF.Extract	✓
	KMAC	(?)
PRF	HKDF.Expand	✓
	KMAC	✓

ECDSA might be fine for EDHOC

- S-UEO ✗: EDHOC 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 (PRK_{out}) added in draft 14 (with Jacomme et al.[1])
- Full credentials in transcript hashes in key derivation.
- Transcript hashes from plaintext instead of ciphertexts
- Key separation in key derivation

[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