Snake Mackerel – An Isogeny Based AKEM

Jonas Janneck[1], **Jonas Meers**[1], Massimo Ostuzzi[1], Doreen Riepel[2]

[1]Ruhr University Bochum
[2]CISPA Saarbrücken

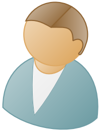
Swiss Isogeny Day 2025

AKEM

Authenticated **K**ey
Encapsulation **M**echanism

Definition

Alice

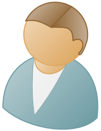


Bob

Definition

$(\mathsf{sk}_A, \mathsf{pk}_A) \xleftarrow{\$} \mathsf{Gen}$

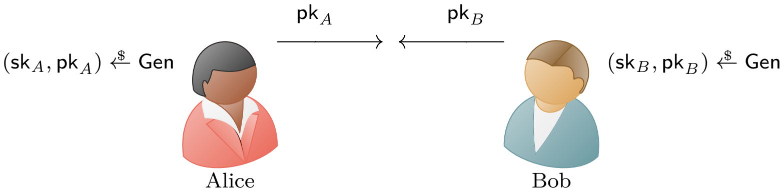
Alice



Bob

$(\mathsf{sk}_B, \mathsf{pk}_B) \xleftarrow{\$} \mathsf{Gen}$

Definition



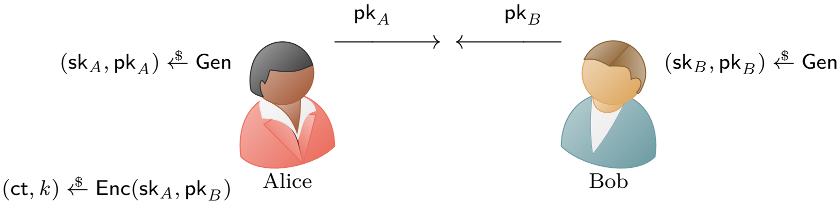
$(\mathsf{sk}_A, \mathsf{pk}_A) \xleftarrow{\$} \mathsf{Gen}$

$\xrightarrow{\mathsf{pk}_A}$

$\xleftarrow{\mathsf{pk}_B}$

$(\mathsf{sk}_B, \mathsf{pk}_B) \xleftarrow{\$} \mathsf{Gen}$

Definition



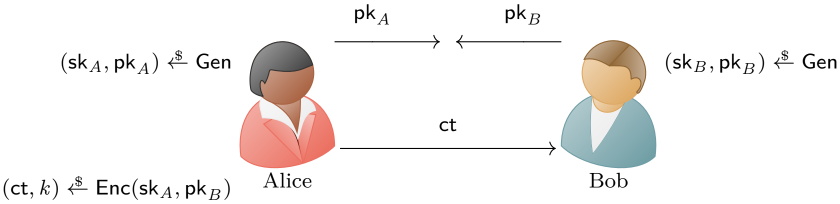
$(\mathsf{sk}_A, \mathsf{pk}_A) \xleftarrow{\$} \mathsf{Gen}$

$\xrightarrow{\mathsf{pk}_A}$ $\xleftarrow{\mathsf{pk}_B}$

$(\mathsf{sk}_B, \mathsf{pk}_B) \xleftarrow{\$} \mathsf{Gen}$

$(\mathsf{ct}, k) \xleftarrow{\$} \mathsf{Enc}(\mathsf{sk}_A, \mathsf{pk}_B)$

Definition



$(\mathsf{sk}_A, \mathsf{pk}_A) \xleftarrow{\$} \mathsf{Gen}$

$(\mathsf{sk}_B, \mathsf{pk}_B) \xleftarrow{\$} \mathsf{Gen}$

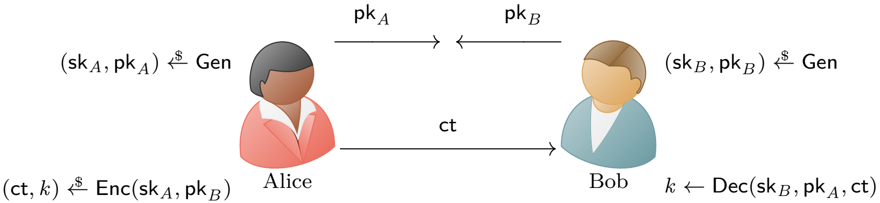
pk_A

pk_B

ct

$(\mathsf{ct}, k) \xleftarrow{\$} \mathsf{Enc}(\mathsf{sk}_A, \mathsf{pk}_B)$

Definition



pk_A

pk_B

$(\mathsf{sk}_A, \mathsf{pk}_A) \xleftarrow{\$} \mathsf{Gen}$

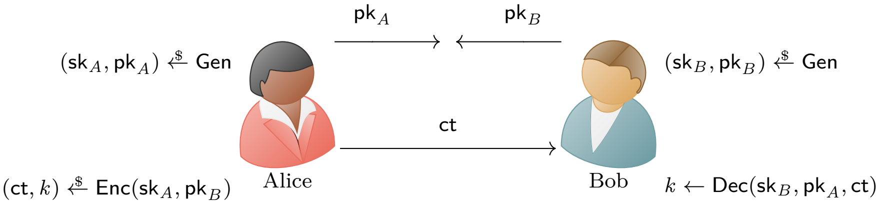
$(\mathsf{sk}_B, \mathsf{pk}_B) \xleftarrow{\$} \mathsf{Gen}$

ct

$(\mathsf{ct}, k) \xleftarrow{\$} \mathsf{Enc}(\mathsf{sk}_A, \mathsf{pk}_B)$

$k \leftarrow \mathsf{Dec}(\mathsf{sk}_B, \mathsf{pk}_A, \mathsf{ct})$

Definition



$(\mathsf{sk}_A, \mathsf{pk}_A) \xleftarrow{\$} \mathsf{Gen}$

pk_A

pk_B

$(\mathsf{sk}_B, \mathsf{pk}_B) \xleftarrow{\$} \mathsf{Gen}$

ct

$(\mathsf{ct}, k) \xleftarrow{\$} \mathsf{Enc}(\mathsf{sk}_A, \mathsf{pk}_B)$

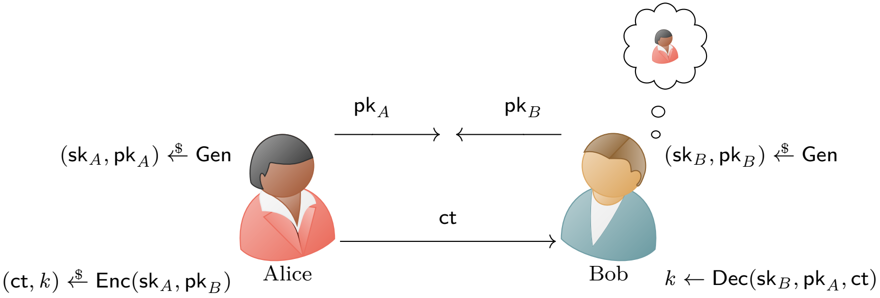
Alice

Bob

$k \leftarrow \mathsf{Dec}(\mathsf{sk}_B, \mathsf{pk}_A, \mathsf{ct})$

- **Confidentiality:** Only Alice and Bob know k

Definition



$(\mathsf{sk}_A, \mathsf{pk}_A) \xleftarrow{\$} \mathsf{Gen}$

pk_A

pk_B

$(\mathsf{sk}_B, \mathsf{pk}_B) \xleftarrow{\$} \mathsf{Gen}$

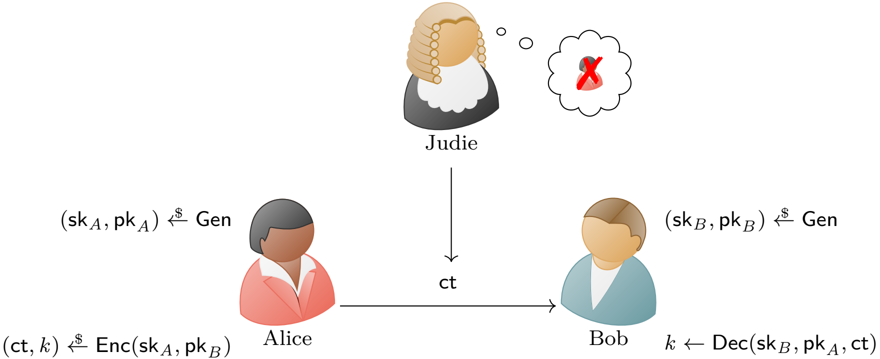
ct

$(\mathsf{ct}, k) \xleftarrow{\$} \mathsf{Enc}(\mathsf{sk}_A, \mathsf{pk}_B)$

$k \leftarrow \mathsf{Dec}(\mathsf{sk}_B, \mathsf{pk}_A, \mathsf{ct})$

- **Confidentiality:** Only Alice and Bob know k
- **Authenticity:** Bob knows that Alice sent ct

Definition



$(\mathsf{sk}_A, \mathsf{pk}_A) \overset{\$}{\leftarrow} \mathsf{Gen}$

$(\mathsf{sk}_B, \mathsf{pk}_B) \overset{\$}{\leftarrow} \mathsf{Gen}$

$(\mathsf{ct}, k) \overset{\$}{\leftarrow} \mathsf{Enc}(\mathsf{sk}_A, \mathsf{pk}_B)$

$k \leftarrow \mathsf{Dec}(\mathsf{sk}_B, \mathsf{pk}_A, \mathsf{ct})$

- **Confidentiality:** Only Alice and Bob know k
- **Authenticity:** Bob knows that Alice sent ct
- **Deniability:** Judie cannot be convinced that Alice sent ct

Generic Constructions

- **KEM + (Ring) Signature** (FrodoKEX+ [CHN$^+$24a], Gandalf-AKEM [GJK24])

Generic Constructions

- **KEM + (Ring) Signature** (FrodoKEX+ [CHN+24a], Gandalf-AKEM [GJK24])
- **Double NIKE** (DH, CSIDH) [AJKL23]

Generic Constructions

- **KEM + (Ring) Signature** (FrodoKEX+ [CHN$^+$24a], Gandalf-AKEM [GJK24])
- **Double NIKE** (DH, CSIDH) [AJKL23]
- **This Work: Split Ciphertext KEM + Identification Scheme** (DH-AKEM [ABF12], SnakeM)

Generic Constructions

- **KEM + (Ring) Signature** (FrodoKEX+ [CHN+24a], Gandalf-AKEM [GJK24])
- **Double NIKE** (DH, CSIDH) [AJKL23]
- **This Work: Split Ciphertext KEM + Identification Scheme** (DH-AKEM [ABF12], SnakeM)

  ⚠ Split Ciphertext KEM ≠ Split KEM

Generic Constructions

- **KEM + (Ring) Signature** (FrodoKEX+ [CHN$^+$24a], Gandalf-AKEM [GJK24])
- **Double NIKE** (DH, CSIDH) [AJKL23]
- **This Work: Split Ciphertext KEM + Identification Scheme** (DH-AKEM [ABF12], SnakeM)

⚠ Split Ciphertext KEM $\neq$ Split KEM

Split Ciphertext KEM

Given ID-Scheme ID, a Split Ciphertext KEM KEM requires

$$(\mathsf{ct}, k) \xleftarrow{\$} \mathsf{KEM.Enc}(\mathsf{pk}), \quad \mathsf{ct} = (\mathsf{ct}_0, \mathsf{ct}_1), \quad \mathsf{ct}_0 \in \mathrm{Im}(\mathsf{ID.Com}).$$

Generic Constructions

- **KEM + (Ring) Signature** (FrodoKEX+ [CHN$^+$24a], Gandalf-AKEM [GJK24])
- **Double NIKE** (DH, CSIDH) [AJKL23]
- **This Work: Split Ciphertext KEM + Identification Scheme** (DH-AKEM [ABF12], SnakeM)

  ⚠ Split Ciphertext KEM $\neq$ Split KEM

Split Ciphertext KEM

Given ID-Scheme ID, a Split Ciphertext KEM KEM requires

$$(\mathsf{ct}, k) \xleftarrow{\$} \mathsf{KEM.Enc}(\mathsf{pk}), \quad \mathsf{ct} = (\mathsf{ct}_0, \mathsf{ct}_1), \quad \mathsf{ct}_0 \in \mathrm{Im}(\mathsf{ID.Com}).$$

$\Rightarrow$ Reusing the commitment leads to a **more compact scheme** than plain KEM + Signature

Generic Constructions

- **KEM + (Ring) Signature** (FrodoKEX+ [CHN$^+$24a], Gandalf-AKEM [GJK24])
- **Double NIKE** (DH, CSIDH) [AJKL23]
- **This Work: Split Ciphertext KEM + Identification Scheme** (DH-AKEM [ABF12], SnakeM)

  ⚠ Split Ciphertext KEM $\neq$ Split KEM

Split Ciphertext KEM

Given ID-Scheme ID, a Split Ciphertext KEM KEM requires

$$(\mathsf{ct}, k) \overset{\$}{\leftarrow} \mathsf{KEM.Enc}(\mathsf{pk}), \quad \mathsf{ct} = (\mathsf{ct}_0, \mathsf{ct}_1), \quad \mathsf{ct}_0 \in \mathrm{Im}(\mathsf{ID.Com}).$$

$\Rightarrow$ Reusing the commitment leads to a **more compact scheme** than plain KEM + Signature

$\Rightarrow$ Our **generic** construction SnakeM can be instantiated from **isogenies**

Generic Constructions

- **KEM + (Ring) Signature** (FrodoKEX+ [CHN+24a], Gandalf-AKEM [GJK24])
- **Double NIKE** (DH, CSIDH) [AJKL23]
- **This Work: Split Ciphertext KEM + Identification Scheme** (DH-AKEM [ABF12], SnakeM)

⚠ Split Ciphertext KEM ≠ Split KEM

Split Ciphertext KEM

Given ID-Scheme ID, a Split Ciphertext KEM KEM requires

$$(\mathsf{ct}, k) \xleftarrow{\$} \mathsf{KEM.Enc}(\mathsf{pk}), \quad \mathsf{ct} = (\mathsf{ct}_0, \mathsf{ct}_1), \quad \mathsf{ct}_0 \in \mathrm{Im}(\mathsf{ID.Com}).$$

⇒ Reusing the commitment leads to a **more compact scheme** than plain KEM + Signature

⇒ Our **generic** construction SnakeM can be instantiated from **isogenies**

⇒ SnakeM is only **5×** larger than DH-AKEM (64 vs. 296 Bytes) – naive approach 370 Bytes

Snake Mackerel = POKÉ + SQIsignHD

⇢ non-rational

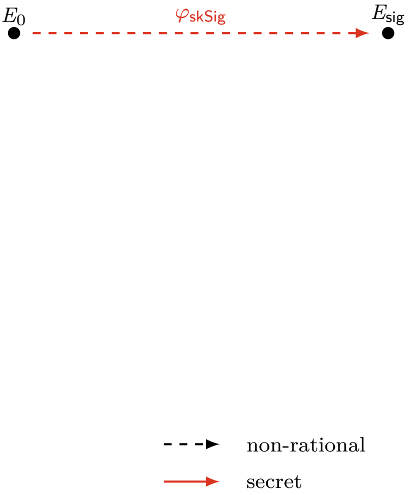
→ secret

Snake Mackerel = POKÉ + SQIsignHD

- - -► non-rational

—► secret

Snake Mackerel = POKÉ + SQIsignHD



Snake Mackerel = POKÉ + SQIsignHD

$$\begin{tikzcd}
E_0 \arrow[rr, dashed, "\varphi_{\mathsf{skSig}}"] \arrow[dd, dashed, "\varphi_{\mathsf{com}}"] & & E_{\mathsf{sig}} \\
 & & \\
E_{\mathsf{com}} & &
\end{tikzcd}$$

⇢ non-rational

→ secret

Snake Mackerel = POKÉ + SQIsignHD

$$\begin{tikzcd}
E_0 \arrow[r, dashed, "\varphi_{\mathsf{skSig}}"] \arrow[d, dashed, "\varphi_{\mathsf{com}}"] & E_{\mathsf{sig}} \arrow[d, "H(E_{\mathsf{com}}, m)"'] \\
E_{\mathsf{com}} & E_{\mathsf{chal}}
\end{tikzcd}$$

⇢ non-rational

→ secret

Snake Mackerel = POKÉ + SQIsignHD

$$\begin{tikzcd}
E_0 \arrow[r, dashed, "\varphi_{\mathsf{skSig}}"] \arrow[d, dashed, "\varphi_{\mathsf{com}}"] & E_{\mathsf{sig}} \arrow[d] \\
E_{\mathsf{com}} \arrow[r, dashed, "{\sigma = (q, \varphi_{\mathsf{rsp}}(U), \varphi_{\mathsf{rsp}}(V))}"] & E_{\mathsf{chal}}
\end{tikzcd}$$

⇢ non-rational

→ secret

Snake Mackerel = POKÉ + SQIsignHD

$X_0 \in E_0[D]$

E_0

$$\begin{tikzcd}
E_0 \arrow[r, dashed, "\varphi_{\mathsf{skSig}}"] \arrow[d, dashed, "\varphi_{\mathsf{com}}"] & E_{\mathsf{sig}} \arrow[d] \\
E_{\mathsf{com}} \arrow[r, dashed, "\sigma = (q, \varphi_{\mathsf{rsp}}(U), \varphi_{\mathsf{rsp}}(V))"] & E_{\mathsf{chal}}
\end{tikzcd}$$

- - -▶ non-rational

——▶ secret

Snake Mackerel = POKÉ + SQIsignHD

$X_1 = [\alpha]\varphi_{\mathsf{skEnc}}(X_0)$ $X_0 \in E_0[D]$

$$\begin{tikzcd}
E_{\mathsf{enc}} & & E_0 \arrow[ll, dashed, "\varphi_{\mathsf{skEnc}}"']
\end{tikzcd}$$

$$\begin{tikzcd}
E_0 \arrow[rr, dashed, "\varphi_{\mathsf{skSig}}"] \arrow[dd, dashed, "\varphi_{\mathsf{com}}"] & & E_{\mathsf{sig}} \arrow[dd] \\
& & \\
E_{\mathsf{com}} \arrow[rr, dashed, "\sigma = (q, \varphi_{\mathsf{rsp}}(U), \varphi_{\mathsf{rsp}}(V))"] & & E_{\mathsf{chal}}
\end{tikzcd}$$

$\dashrightarrow$ non-rational

$\rightarrow$ secret

Snake Mackerel = POKÉ + SQIsignHD

$$\begin{tikzcd}
X_1 = [\alpha]\varphi_{\mathsf{skEnc}}(X_0) & X_0 \in E_0[D] & & \\
E_{\mathsf{enc}} \arrow[d, "\varphi'"] & E_0 \arrow[l, dashed, "\varphi_{\mathsf{skEnc}}"'] \arrow[d, "\varphi"'] & E_0 \arrow[r, dashed, "\varphi_{\mathsf{skSig}}"] \arrow[d, dashed, "\varphi_{\mathsf{com}}"] & E_{\mathsf{sig}} \arrow[d] \\
E_{\mathsf{ct}} & E_3 & E_{\mathsf{com}} \arrow[r, dashed, "{\sigma = (q, \varphi_{\mathsf{rsp}}(U), \varphi_{\mathsf{rsp}}(V))}"] & E_{\mathsf{chal}} \\
X_3 = [\beta]\varphi'(X_1) & X_2 = [\beta]\varphi(X_0) & &
\end{tikzcd}$$

$\dashrightarrow$ non-rational

$\longrightarrow$ secret

Snake Mackerel = POKÉ + SQIsignHD

$$X_1 = [\alpha]\varphi_{\mathsf{skEnc}}(X_0) \qquad X_0 \in E_0[D]$$

$$\begin{tikzcd}
E_{\mathsf{enc}} \arrow[d, "\varphi'"'] & E_0 \arrow[l, dashed, "\varphi_{\mathsf{skEnc}}"'] \arrow[d, "\varphi"'] & E_0 \arrow[r, dashed, "\varphi_{\mathsf{skSig}}"] \arrow[d, dashed, "\varphi_{\mathsf{com}}"] & E_{\mathsf{sig}} \arrow[d] \\
E_{\mathsf{ct}} & E_3 \arrow[l, dashed, "\varphi'_{\mathsf{skEnc}}"'] & E_{\mathsf{com}} \arrow[r, dashed, "\sigma = (q, \varphi_{\mathsf{rsp}}(U), \varphi_{\mathsf{rsp}}(V))"] & E_{\mathsf{chal}}
\end{tikzcd}$$

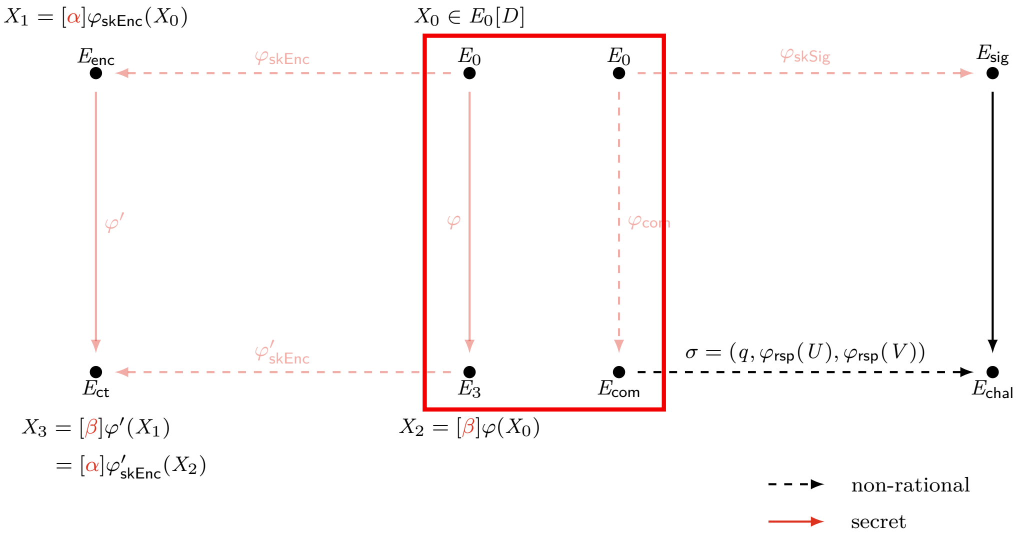
$$X_3 = [\beta]\varphi'(X_1) \qquad X_2 = [\beta]\varphi(X_0)$$

$$= [\alpha]\varphi'_{\mathsf{skEnc}}(X_2)$$

$\dashrightarrow$ non-rational

$\rightarrow$ secret

Snake Mackerel = POKÉ + SQIsignHD



$X_1 = [\alpha]\varphi_{\mathsf{skEnc}}(X_0)$

$X_0 \in E_0[D]$

E_{enc}

φ_{skEnc}

E_0

E_0

φ_{skSig}

E_{sig}

φ'

φ

φ_{com}

$\sigma = (q, \varphi_{\mathsf{rsp}}(U), \varphi_{\mathsf{rsp}}(V))$

$\varphi'_{\mathsf{skEnc}}$

E_{ct}

E_3

E_{com}

E_{chal}

$X_3 = [\beta]\varphi'(X_1)$

$= [\alpha]\varphi'_{\mathsf{skEnc}}(X_2)$

$X_2 = [\beta]\varphi(X_0)$

non-rational

secret

Snake Mackerel = POKÉ + SQIsignHD

$X_1 = [\alpha]\varphi_{\mathsf{skEnc}}(X_0)$

$X_0 \in E_0[D]$

$$\begin{tikzcd}
E_{\mathsf{enc}} \arrow[d, "\varphi'_{\mathsf{com}}"] & E_0 \arrow[l, dashed, "\varphi_{\mathsf{skEnc}}"'] \arrow[r, dashed, "\varphi_{\mathsf{skSig}}"] \arrow[d, "\varphi_{\mathsf{com}}"] & E_{\mathsf{sig}} \arrow[d, "{H(E_{\mathsf{com}}, X_3, \mathsf{cntxt})}"'] \\
E_{\mathsf{ct}} & E_{\mathsf{com}} \arrow[l, dashed, "\varphi'_{\mathsf{skEnc}}"'] \arrow[r, dashed, "{\sigma = (q, \varphi_{\mathsf{rsp}}(U), \varphi_{\mathsf{rsp}}(V))}"] & E_{\mathsf{chal}}
\end{tikzcd}$$

$$\begin{aligned} X_3 &= [\beta]\varphi'(X_1) \\ &= [\alpha]\varphi'_{\mathsf{skEnc}}(X_2) \end{aligned}$$

$X_2 = [\beta]\varphi(X_0)$

- - -► non-rational

──► secret

Compatibility

SQIsignHD and POKÉ use primes $p = c2^a3^b - 1$, but with different **sizes**

POKÉ: $3^b \approx 2^{2\lambda}$ SQIsignHD: $3^b \approx 2^{\lambda}$

Compatibility

SQIsignHD and POKÉ use primes $p = c2^a3^b - 1$, but with different **sizes**

$$\text{POKÉ: } 3^b \approx 2^{2\lambda} \qquad \text{SQIsignHD: } 3^b \approx 2^{\lambda}$$

Use **B-SIDH** approach for a more compact scheme with $p \in \mathcal{O}(2^{2\lambda})$

$$(p+1)(p-1) = 2^a ND, \qquad 2^a \in \mathcal{O}(2^{\lambda}), \qquad N = \prod \ell_i \in \mathcal{O}(2^{2\lambda}), \qquad D = q_1 q_2 q_3 \in \mathcal{O}(2^{\lambda})$$

Compatibility

SQIsignHD and POKÉ use primes $p = c2^a3^b - 1$, but with different **sizes**

$$\text{POKÉ: } 3^b \approx 2^{2\lambda} \qquad \text{SQIsignHD: } 3^b \approx 2^{\lambda}$$

Use **B-SIDH** approach for a more compact scheme with $p \in \mathcal{O}(2^{2\lambda})$

$$(p+1)(p-1) = 2^a ND, \quad \underbrace{2^a \in \mathcal{O}(2^{\lambda})}_{\text{HD representation}}, \quad \underbrace{N = \prod \ell_i \in \mathcal{O}(2^{2\lambda})}_{\text{rational isogenies}}, \quad \underbrace{D = q_1 q_2 q_3 \in \mathcal{O}(2^{\lambda})}_{\text{shared key}}$$

Compatibility

SQIsignHD and POKÉ use primes $p = c2^a3^b - 1$, but with different **sizes**

$$\text{POKÉ: } 3^b \approx 2^{2\lambda} \qquad \text{SQIsignHD: } 3^b \approx 2^{\lambda}$$

Use **B-SIDH** approach for a more compact scheme with $p \in \mathcal{O}(2^{2\lambda})$

$$(p+1)(p-1) = 2^a ND, \quad \underbrace{2^a \in \mathcal{O}(2^{\lambda}),}_{\text{HD representation}} \quad \underbrace{N = \prod \ell_i \in \mathcal{O}(2^{2\lambda}),}_{\text{rational isogenies}} \quad \underbrace{D = q_1 q_2 q_3 \in \mathcal{O}(2^{\lambda})}_{\text{shared key}}$$

- Commitment isogeny is now **rational**
  - $N \in \mathcal{O}(p)$ to ensure **good distribution** of the commitment curve

Compatibility

SQIsignHD and POKÉ use primes $p = c2^a3^b - 1$, but with different **sizes**

$$\text{POKÉ: } 3^b \approx 2^{2\lambda} \qquad \text{SQIsignHD: } 3^b \approx 2^{\lambda}$$

Use **B-SIDH** approach for a more compact scheme with $p \in \mathcal{O}(2^{2\lambda})$

$$(p+1)(p-1) = 2^a ND, \quad \underbrace{2^a \in \mathcal{O}(2^{\lambda}),}_{\text{HD representation}} \quad \underbrace{N = \prod \ell_i \in \mathcal{O}(2^{2\lambda}),}_{\text{rational isogenies}} \quad \underbrace{D = q_1 q_2 q_3 \in \mathcal{O}(2^{\lambda})}_{\text{shared key}}$$

- Commitment isogeny is now **rational**
  - $N \in \mathcal{O}(p)$ to ensure **good distribution** of the commitment curve
- D smooth enough to allow for **point compression**

Compatibility

SQIsignHD and POKÉ use primes $p = c2^a3^b - 1$, but with different **sizes**

$$\text{POKÉ: } 3^b \approx 2^{2\lambda} \qquad \text{SQIsignHD: } 3^b \approx 2^{\lambda}$$

Use **B-SIDH** approach for a more compact scheme with $p \in \mathcal{O}(2^{2\lambda})$

$$(p+1)(p-1) = 2^a ND, \quad \underbrace{2^a \in \mathcal{O}(2^{\lambda}),}_{\text{HD representation}} \quad \underbrace{N = \prod \ell_i \in \mathcal{O}(2^{2\lambda}),}_{\text{rational isogenies}} \quad \underbrace{D = q_1 q_2 q_3 \in \mathcal{O}(2^{\lambda})}_{\text{shared key}}$$

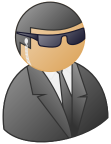
- Commitment isogeny is now **rational**
    - $N \in \mathcal{O}(p)$ to ensure **good distribution** of the commitment curve
- D smooth enough to allow for **point compression**

$$p = 2^{133} \cdot 3^6 \cdot 7^2 \cdot 17^4 \cdot 47^2 \cdot 311^2 \cdot 367^2 \cdot 439^2 \cdot 1049^2 \cdot 1373 - 1$$
$$\log p = 247, \qquad \max\{\ell_i\} = 1373, \qquad \max\{\log q_i\} = 39$$

Security

The Best out of Both Worlds?

Confidentiality: Ins-CCA, simplified

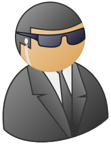


Challenger

Adversary

Confidentiality: Ins-CCA, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

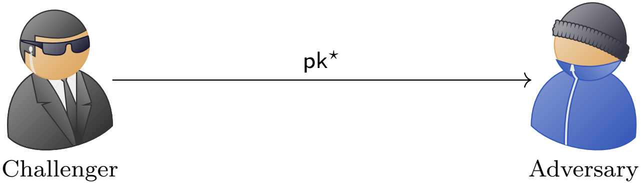


Challenger

Adversary

Confidentiality: Ins-CCA, simplified

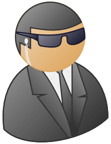
$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$



Confidentiality: Ins-CCA, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\mathsf{pk}^\star$



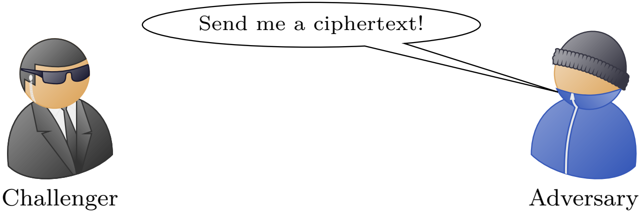
Challenger

Adversary

Confidentiality: Ins-CCA, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

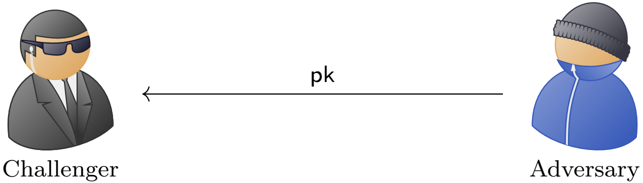
$\mathsf{pk}^\star$



Confidentiality: Ins-CCA, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\mathsf{pk}^\star$



pk

Challenger

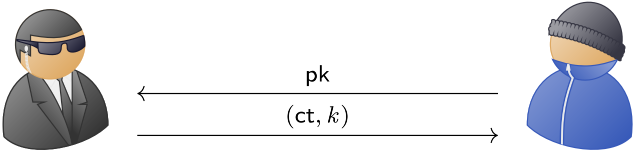
Adversary

Confidentiality: Ins-CCA, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\mathsf{pk}^\star$

$(\mathsf{ct}, k) \xleftarrow{\$} \mathsf{Encaps}(\mathsf{sk}^\star, \mathsf{pk})$



$\xleftarrow{\mathsf{pk}}$

$\xrightarrow{(\mathsf{ct}, k)}$

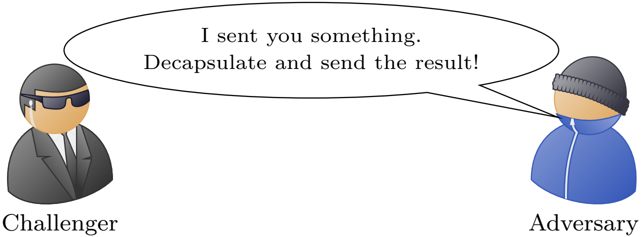
Challenger

Adversary

Confidentiality: Ins-CCA, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

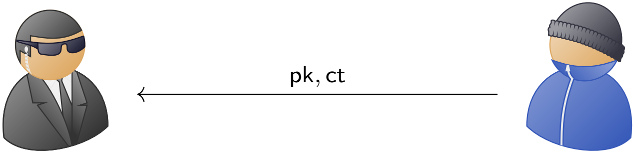
$\mathsf{pk}^\star$



Confidentiality: Ins-CCA, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\mathsf{pk}^\star$



pk, ct

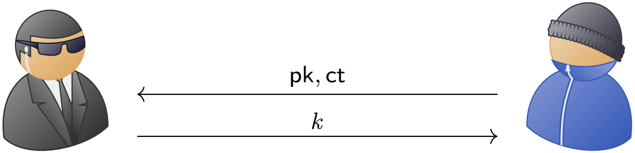
Challenger

Adversary

Confidentiality: Ins-CCA, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\mathsf{pk}^\star$



$k \leftarrow \mathsf{Decaps}(\mathsf{pk}, \mathsf{sk}^\star, \mathsf{ct})$

$\xleftarrow{\mathsf{pk}, \mathsf{ct}}$

$\xrightarrow{k}$

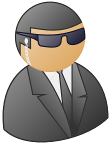
Challenger

Adversary

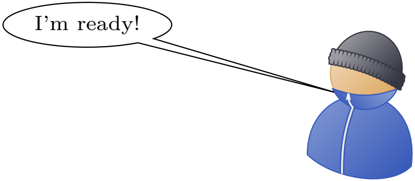
Confidentiality: Ins-CCA, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\mathsf{pk}^\star$



Challenger



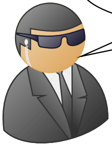
Adversary

Confidentiality: Ins-CCA, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\mathsf{pk}^\star$

I will send an honest message on your behalf to myself



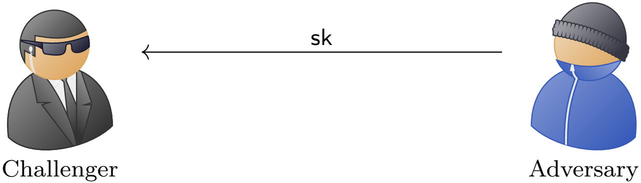
Challenger

Adversary

Confidentiality: Ins-CCA, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\mathsf{pk}^\star$



sk

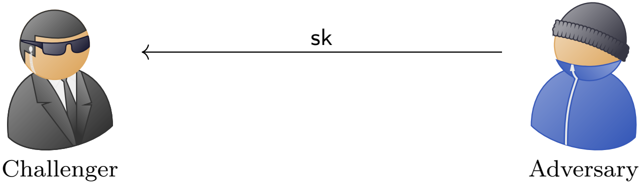
Challenger

Adversary

Confidentiality: Ins-CCA, simplified

```
$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\beta \xleftarrow{\$} \{0, 1\}$
$(\mathsf{ct}, k) \xleftarrow{\$} \mathsf{Encaps}(\mathsf{sk}, \mathsf{pk}^\star)$
**if** $\beta = 1$
    $k \leftarrow \$$
```

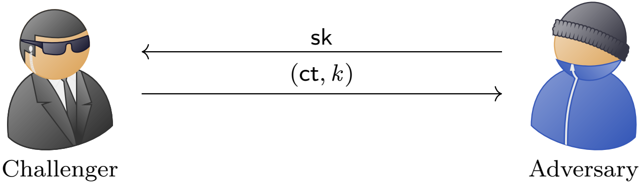
$\mathsf{pk}^\star$



Confidentiality: Ins-CCA, simplified

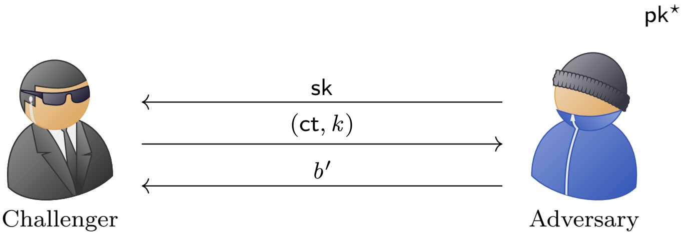
```
$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\beta \xleftarrow{\$} \{0, 1\}$
$(\mathsf{ct}, k) \xleftarrow{\$} \mathsf{Encaps}(\mathsf{sk}, \mathsf{pk}^\star)$
$\textbf{if } \beta = 1$
    $k \leftarrow \$$
```

$\mathsf{pk}^\star$



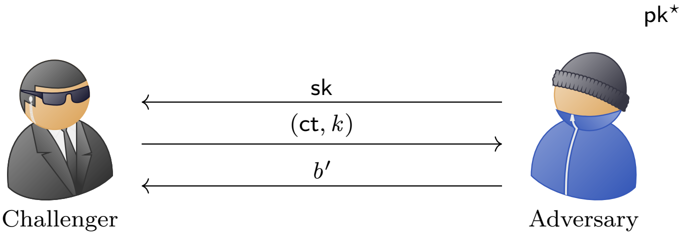
Confidentiality: Ins-CCA, simplified

```
$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\beta \xleftarrow{\$} \{0,1\}$
$(\mathsf{ct}, k) \xleftarrow{\$} \mathsf{Encaps}(\mathsf{sk}, \mathsf{pk}^\star)$
$\textbf{if } \beta = 1$
    $k \leftarrow \$$
$\textbf{win if } b = b'$
```



Confidentiality: Ins-CCA, simplified

```
$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\beta \xleftarrow{\$} \{0,1\}$
$(\mathsf{ct}, k) \xleftarrow{\$} \mathsf{Encaps}(\mathsf{sk}, \mathsf{pk}^\star)$
$\textbf{if } \beta = 1$
    $k \leftarrow \$$
$\textbf{win if } b = b'$
```



Note

sk is used for the **signature** and should not help to decapsulate the **KEM** part of ct

Confidentiality of SnakeM

Theorem

For any $\mathsf{Ins\text{-}CCA}$ adversary $\mathcal{A}$ against SnakeM, there exist an adversary $\mathcal{B}$ against $\mathsf{OW\text{-}KCA}$ of POKÉ such that

$$\mathsf{Adv}^{\mathsf{Ins\text{-}CCA}}_{\text{SnakeM}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{OW\text{-}KCA}}_{\text{POKÉ}}(\mathcal{B}) + \delta.$$

Confidentiality of SnakeM

Theorem

For any $\mathsf{Ins\text{-}CCA}$ adversary $\mathcal{A}$ against SnakeM, there exist an adversary $\mathcal{B}$ against $\mathsf{OW\text{-}KCA}$ of POKÉ such that

$$\mathsf{Adv}^{\mathsf{Ins\text{-}CCA}}_{\mathrm{SnakeM}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{OW\text{-}KCA}}_{\mathrm{POKÉ}}(\mathcal{B}) + \delta.$$

- $\mathsf{OW\text{-}KCA}$: Compute the shared key given access to an **key-checking** oracle

$$\mathcal{O}^{\mathsf{kc}}(\mathsf{ct}, k) \to 1 \quad \text{if } \mathsf{ct} \text{ contains key } k$$

Confidentiality of SnakeM

Theorem

For any $\mathsf{Ins\text{-}CCA}$ adversary $\mathcal{A}$ against SnakeM, there exist an adversary $\mathcal{B}$ against $\mathsf{OW\text{-}KCA}$ of POKÉ such that

$$\mathsf{Adv}^{\mathsf{Ins\text{-}CCA}}_{\mathrm{SnakeM}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{OW\text{-}KCA}}_{\mathrm{POKÉ}}(\mathcal{B}) + \delta.$$

- $\mathsf{OW\text{-}KCA}$: Compute the shared key given access to an **key-checking** oracle

$$\mathcal{O}^{\mathsf{kc}}(\mathsf{ct}, k) \rightarrow 1 \quad \text{if } \mathsf{ct} \text{ contains key } k$$

Why $\mathsf{OW\text{-}KCA}$ when POKÉ is $\mathsf{IND\text{-}CCA}$ secure?

Confidentiality of SnakeM

Theorem

For any Ins-CCA adversary $\mathcal{A}$ against SnakeM, there exist an adversary $\mathcal{B}$ against OW-KCA of POKÉ such that

$$\mathsf{Adv}^{\mathsf{Ins\text{-}CCA}}_{\mathrm{SnakeM}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{OW\text{-}KCA}}_{\mathrm{POK\acute{E}}}(\mathcal{B}) + \delta.$$

- OW-KCA: Compute the shared key given access to an **key-checking** oracle

$$\mathcal{O}^{\mathsf{kc}}(\mathsf{ct}, k) \to 1 \quad \text{if } \mathsf{ct} \text{ contains key } k$$

Why OW-KCA when POKÉ is IND-CCA secure? We cannot use **Fujisaki-Okamoto Transform** :(

Confidentiality of SnakeM

Theorem

For any Ins-CCA adversary $\mathcal{A}$ against SnakeM, there exist an adversary $\mathcal{B}$ against OW-KCA of POKÉ such that

$$\mathsf{Adv}^{\mathsf{Ins\text{-}CCA}}_{\mathrm{SnakeM}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{OW\text{-}KCA}}_{\mathrm{POK\acute{E}}}(\mathcal{B}) + \delta.$$

- OW-KCA: Compute the shared key given access to an **key-checking** oracle

$$\mathcal{O}^{\mathsf{kc}}(\mathsf{ct}, k) \to 1 \quad \text{if } \mathsf{ct} \text{ contains key } k$$

Why OW-KCA when POKÉ is IND-CCA secure? We cannot use **Fujisaki-Okamoto Transform** :(

Fujisaki-Okamoto Transform [FO99, HHK17]

$$\mathsf{IND\text{-}CPA} \xRightarrow{\text{T-Transform}} \mathsf{OW\text{-}KCA} \xRightarrow{\text{U-Transform}} \mathsf{IND\text{-}CCA}$$

Confidentiality of SnakeM

Theorem

For any $\mathsf{Ins\text{-}CCA}$ adversary $\mathcal{A}$ against SnakeM, there exist an adversary $\mathcal{B}$ against $\mathsf{OW\text{-}KCA}$ of POKÉ such that

$$\mathsf{Adv}^{\mathsf{Ins\text{-}CCA}}_{\mathrm{SnakeM}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{OW\text{-}KCA}}_{\mathrm{POKÉ}}(\mathcal{B}) + \delta.$$

- $\mathsf{OW\text{-}KCA}$: Compute the shared key given access to an **key-checking** oracle

$$\mathcal{O}^{\mathsf{kc}}(\mathsf{ct}, k) \to 1 \quad \text{if } \mathsf{ct} \text{ contains key } k$$

Why $\mathsf{OW\text{-}KCA}$ when POKÉ is $\mathsf{IND\text{-}CCA}$ secure? We cannot use **Fujisaki-Okamoto Transform** :(

Fujisaki-Okamoto Transform [FO99, HHK17]

$$\mathsf{IND\text{-}CPA} \xrightarrow{\text{T-Transform}} \mathsf{OW\text{-}KCA} \xrightarrow{\text{U-Transform}} \mathsf{IND\text{-}CCA}$$

- T-Transform makes the encryption randomness **explicit** $\Longrightarrow$ leaks **commitment**

Confidentiality of SnakeM

$$\begin{tikzcd}
E_{\mathsf{enc}} \arrow[d, "\varphi'"] & E_0 \arrow[l, "\varphi_{\mathsf{skEnc}}"'] \arrow[d, "\varphi"'] & E_0 \arrow[r, "\varphi_{\mathsf{skSig}}"] \arrow[d, "\varphi_{\mathsf{com}}"] & E_{\mathsf{sig}} \arrow[d] \\
E_{\mathsf{ct}} & E_3 \arrow[l, "\varphi'_{\mathsf{skEnc}}"'] & E_{\mathsf{com}} \arrow[r, "\sigma = (q, \varphi_{\mathsf{rsp}}(U), \varphi_{\mathsf{rsp}}(V))"] & E_{\mathsf{chal}}
\end{tikzcd}$$

Confidentiality of SnakeM

Theorem

For any Ins-CCA adversary $\mathcal{A}$ against SnakeM, there exist an adversary $\mathcal{B}$ against OW-KCA of POKÉ such that

$$\mathsf{Adv}^{\mathsf{Ins\text{-}CCA}}_{\mathrm{SnakeM}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{OW\text{-}KCA}}_{\mathrm{POK\acute{E}}}(\mathcal{B}) + \delta.$$

- OW-KCA: Compute the shared key given access to an **key-checking** oracle

$$\mathcal{O}^{\mathsf{kc}}(\mathsf{ct}, k) \to 1 \quad \text{if } \mathsf{ct} \text{ contains key } k$$

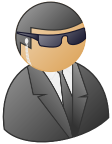
Why OW-KCA when POKÉ is IND-CCA secure? We cannot use **Fujisaki-Okamoto Transform** :(

Fujisaki-Okamoto Transform [FO99, HHK17]

$$\mathsf{IND\text{-}CPA} \xRightarrow{\text{T-Transform}} \mathsf{OW\text{-}KCA} \xRightarrow{\text{U-Transform}} \mathsf{IND\text{-}CCA}$$

- T-Transform makes the encryption randomness **explicit** $\Longrightarrow$ leaks **commitment**
- We include checks to avoid adaptive attacks like [GPST16, MOXZ24]

Authenticity: Ins-Auth, simplified

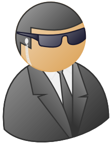


Challenger

Adversary

Authenticity: Ins-Auth, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

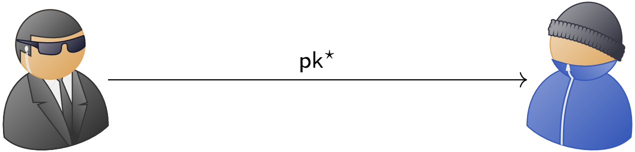


Challenger

Adversary

Authenticity: Ins-Auth, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$



$\mathsf{pk}^\star$

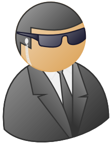
Challenger

Adversary

Authenticity: Ins-Auth, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \overset{\$}{\leftarrow} \mathsf{Gen}$

$\mathsf{pk}^\star$



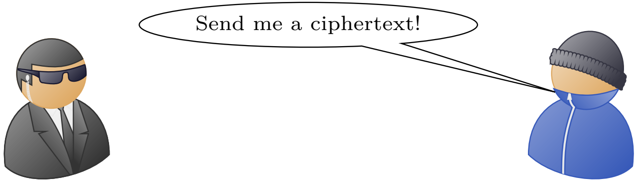
Challenger

Adversary

Authenticity: Ins-Auth, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\mathsf{pk}^\star$



Send me a ciphertext!

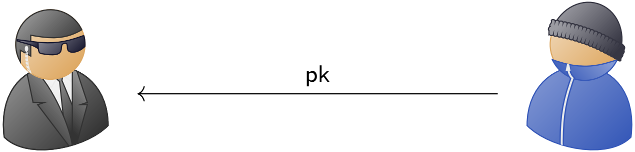
Challenger

Adversary

Authenticity: Ins-Auth, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\mathsf{pk}^\star$



pk

Challenger

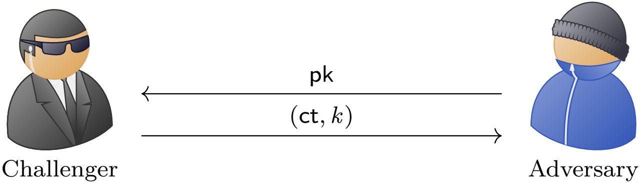
Adversary

Authenticity: Ins-Auth, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\mathsf{pk}^\star$

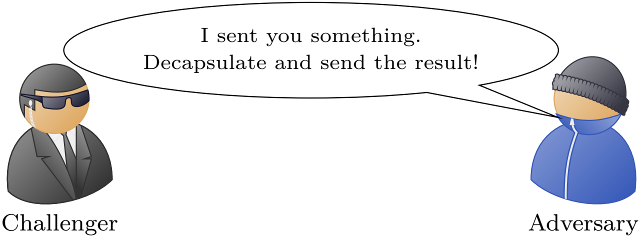
$(\mathsf{ct}, k) \xleftarrow{\$} \mathsf{Encaps}(\mathsf{sk}^\star, \mathsf{pk})$



Authenticity: Ins-Auth, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

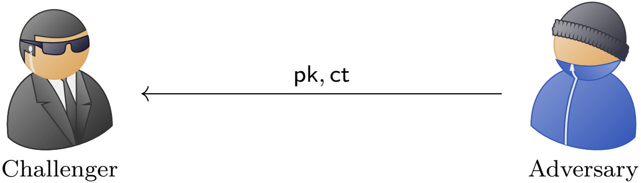
$\mathsf{pk}^\star$



Authenticity: Ins-Auth, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\mathsf{pk}^\star$

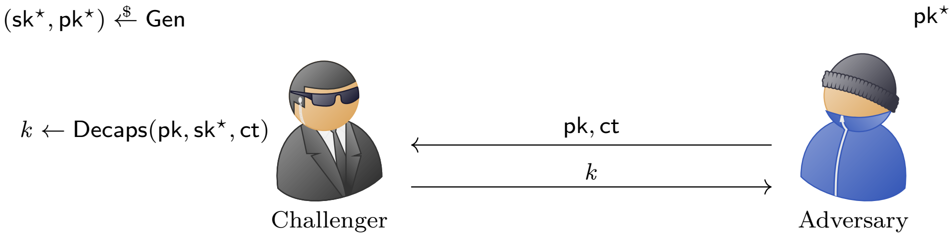


pk, ct

Challenger

Adversary

Authenticity: Ins-Auth, simplified



$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\mathsf{pk}^\star$

$k \leftarrow \mathsf{Decaps}(\mathsf{pk}, \mathsf{sk}^\star, \mathsf{ct})$

pk, ct

k

Challenger

Adversary

Authenticity: Ins-Auth, simplified

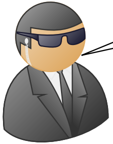
$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\mathsf{pk}^\star$

Authenticity: Ins-Auth, simplified

$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

Try to send a fresh ciphertext on my behalf!

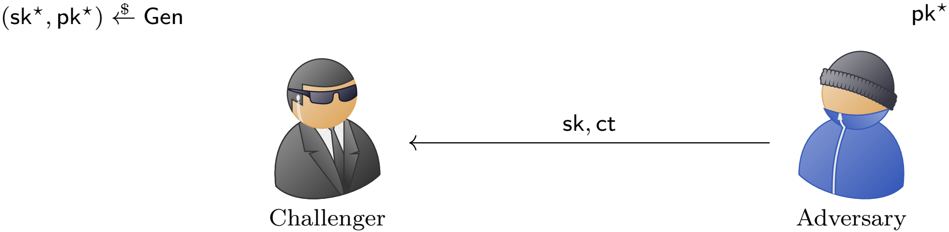


Challenger

$\mathsf{pk}^\star$

Adversary

Authenticity: Ins-Auth, simplified



$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

$\mathsf{pk}^\star$

sk, ct

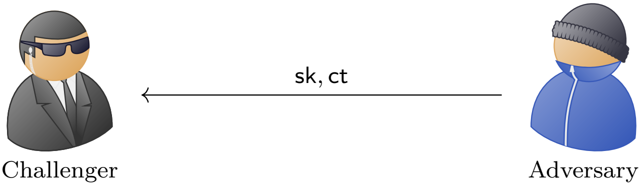
Challenger

Adversary

Authenticity: Ins-Auth, simplified

```
$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$
if $\mathsf{ct}$ not fresh:
    abort
$k \xleftarrow{\$} \mathsf{Decaps}(\mathsf{pk}^\star, \mathsf{sk}, \mathsf{ct})$
win if $k \neq \bot$
```

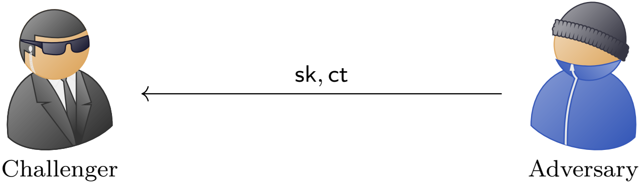
$\mathsf{pk}^\star$



Authenticity: Ins-Auth, simplified

```
$(\mathsf{sk}^\star, \mathsf{pk}^\star) \xleftarrow{\$} \mathsf{Gen}$

if ct not fresh:
    abort
$k \xleftarrow{\$} \mathsf{Decaps}(\mathsf{pk}^\star, \mathsf{sk}, \mathsf{ct})$
win if $k \neq \bot$
```

$\mathsf{pk}^\star$



Note

An honest Decaps checks the **signature** against $\mathsf{pk}^\star$ and returns $\bot$ if the signature is invalid

Non-Malleability: Return of the Lollipop

Observation

For Ins-Auth the signature needs to be **non-malleable**

$$\mathsf{ct} = (\mathsf{ct}_{\mathsf{KEM}}, \sigma) \qquad \Longrightarrow \qquad \mathsf{ct}' = (\mathsf{ct}_{\mathsf{KEM}}, \sigma')$$

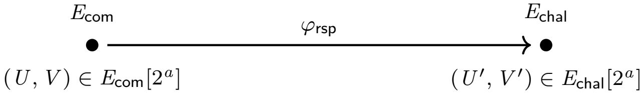
Non-Malleability: Return of the Lollipop

Observation

For Ins-Auth the signature needs to be **non-malleable**

$$\mathsf{ct} = (\mathsf{ct}_{\mathsf{KEM}}, \sigma) \quad \Longrightarrow \quad \mathsf{ct}' = (\mathsf{ct}_{\mathsf{KEM}}, \sigma')$$

In SQIsignHD, the signature is **interpolation data** $\sigma = (q, U', V')$


E_{com}
φ_{rsp}
E_{chal}
$(U, V) \in E_{\mathsf{com}}[2^a]$
$(U', V') \in E_{\mathsf{chal}}[2^a]$

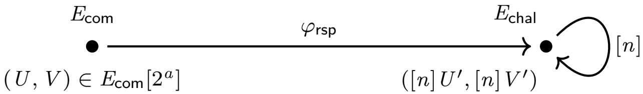
Non-Malleability: Return of the Lollipop

Observation

For Ins-Auth the signature needs to be **non-malleable**

$$\mathsf{ct} = (\mathsf{ct}_{\mathsf{KEM}}, \sigma) \qquad \Longrightarrow \qquad \mathsf{ct}' = (\mathsf{ct}_{\mathsf{KEM}}, \sigma')$$

In SQIsignHD, the signature is **interpolation data** $\sigma = (q, U', V')$



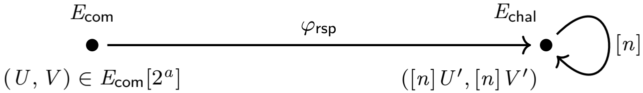
Non-Malleability: Return of the Lollipop

Observation

For Ins-Auth the signature needs to be **non-malleable**

$$\mathsf{ct} = (\mathsf{ct}_{\mathsf{KEM}}, \sigma) \qquad \Longrightarrow \qquad \mathsf{ct}' = (\mathsf{ct}_{\mathsf{KEM}}, \sigma')$$

In SQIsignHD, the signature is **interpolation data** $\sigma = (q, U', V')$



- If $n^2 q \leq 2^a$, then $\sigma' = (\mathbf{n^2} q, [\mathbf{n}]U', [\mathbf{n}]V')$ is a valid signature too

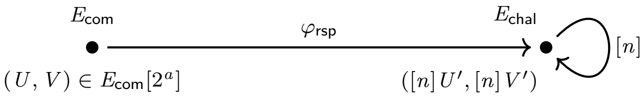
Non-Malleability: Return of the Lollipop

Observation

For Ins-Auth the signature needs to be **non-malleable**

$$\mathsf{ct} = (\mathsf{ct}_{\mathsf{KEM}}, \sigma) \quad \Longrightarrow \quad \mathsf{ct}' = (\mathsf{ct}_{\mathsf{KEM}}, \sigma')$$

In SQIsignHD, the signature is **interpolation data** $\sigma = (q, U', V')$



- If $n^2 q \leq 2^a$, then $\sigma' = (\mathbf{n^2} q, [\mathbf{n}]U', [\mathbf{n}]V')$ is a valid signature too
- Checking square-freeness of q is not enough as φ_{rsp} may contain a **cyclic** ℓ^m-isogeny

Non-Malleability: Return of the Lollipop

Observation

For Ins-Auth the signature needs to be **non-malleable**

$$\mathsf{ct} = (\mathsf{ct}_{\mathsf{KEM}}, \sigma) \qquad \Longrightarrow \qquad \mathsf{ct}' = (\mathsf{ct}_{\mathsf{KEM}}, \sigma')$$

In SQIsignHD, the signature is **interpolation data** $\sigma = (q, U', V')$

$$\begin{tikzcd}
E_{\mathsf{com}} \arrow[rr, "\varphi_{\mathsf{rsp}}"] & & E_{\mathsf{chal}} \arrow[loop right, "{[n]}"] \\
(U, V) \in E_{\mathsf{com}}[2^a] & & ([n]\,U', [n]\,V')
\end{tikzcd}$$

- ▶ If $n^2 q \leq 2^a$, then $\sigma' = (\mathbf{n^2}\,q, [\mathbf{n}]\,U', [\mathbf{n}]\,V')$ is a valid signature too
- ▶ Checking square-freeness of q is not enough as φ_{rsp} may contain a **cyclic** ℓ^m-isogeny

⇒ Non-Malleable version of SQIsignHD?

Non-Malleable SQIsignHD

It would be desirable to **check cyclicity** of HD-represented isogenies

Non-Malleable SQIsignHD

It would be desirable to **check cyclicity** of HD-represented isogenies

Bad News

So far, a generic cyclicity check seems out of reach

Non-Malleable SQIsignHD

It would be desirable to **check cyclicity** of HD-represented isogenies

Bad News

So far, a generic cyclicity check seems out of reach

Good News

We don't need a generic cyclicity check!

Non-Malleable SQIsignHD

It would be desirable to **check cyclicity** of HD-represented isogenies

Bad News

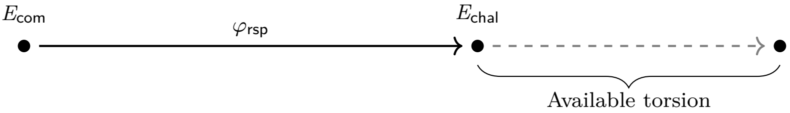
So far, a generic cyclicity check seems out of reach

Good News

We don't need a generic cyclicity check!

Idea:

- During signing: require **minimum length** $q \geq 2^a / \log p$



Non-Malleable SQIsignHD

It would be desirable to **check cyclicity** of HD-represented isogenies

Bad News

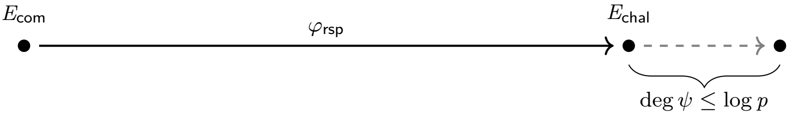
So far, a generic cyclicity check seems out of reach

Good News

We don't need a generic cyclicity check!

Idea:

- During signing: require **minimum length** $q \geq 2^a / \log p$



Non-Malleable SQIsignHD

It would be desirable to **check cyclicity** of HD-represented isogenies

Bad News

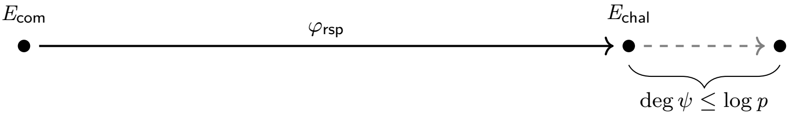
So far, a generic cyclicity check seems out of reach

Good News

We don't need a generic cyclicity check!

Idea:

- During signing: require **minimum length** $q \geq 2^a / \log p$
- During verification: evaluate φ_{rsp} on $E_{\mathsf{com}}[n]$ for all prime $n \leq \sqrt{\log p}$ and see if it vanishes



Non-Malleable SQIsignHD

It would be desirable to **check cyclicity** of HD-represented isogenies

Bad News

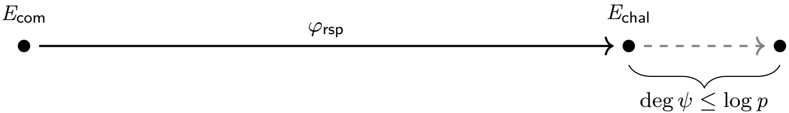
So far, a generic cyclicity check seems out of reach

Good News

We don't need a generic cyclicity check!

Idea:

- During signing: require **minimum length** $q \geq 2^a / \log p$
- During verification: evaluate φ_{rsp} on $E_{\mathsf{com}}[n]$ for all prime $n \leq \sqrt{\log p}$ and see if it vanishes
  - cannot append **small** scalar multiplication



Non-Malleable SQIsignHD

It would be desirable to **check cyclicity** of HD-represented isogenies

Bad News

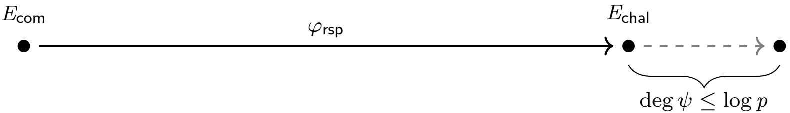
So far, a generic cyclicity check seems out of reach

Good News

We don't need a generic cyclicity check!

Idea:

- During signing: require **minimum length** $q \geq 2^a / \log p$
- During verification: evaluate φ_{rsp} on $E_{\mathsf{com}}[n]$ for all prime $n \leq \sqrt{\log p}$ and see if it vanishes
  - cannot append **small** scalar multiplication
- Large(r) scalar multiplication already **exceeds** the available 2^a-torsion



Non-Malleable SQIsignHD

It would be desirable to **check cyclicity** of HD-represented isogenies

Bad News

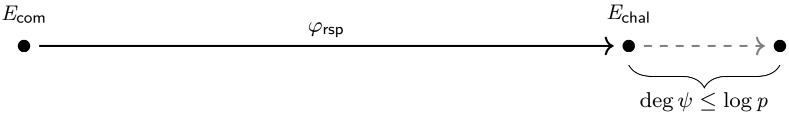
So far, a generic cyclicity check seems out of reach

Good News

We don't need a generic cyclicity check!

Idea:

- During signing: require **minimum length** $q \geq 2^a / \log p$
- During verification: evaluate φ_{rsp} on $E_{\mathsf{com}}[n]$ for all prime $n \leq \sqrt{\log p}$ and see if it vanishes
  - cannot append **small** scalar multiplication
- Large(r) scalar multiplication already **exceeds** the available 2^a-torsion
- Experiments suggest: rejection probability **1/1000**



Non-Malleable SQIsignHD

It would be desirable to **check cyclicity** of HD-represented isogenies

Bad News

So far, a generic cyclicity check seems out of reach

Good News

We don't need a generic cyclicity check!

Idea:

- During signing: require **minimum length** $q \geq 2^a / \log p$
- During verification: evaluate φ_{rsp} on $E_{\mathsf{com}}[n]$ for all prime $n \leq \sqrt{\log p}$ and see if it vanishes
  - cannot append **small** scalar multiplication
- Large(r) scalar multiplication already **exceeds** the available 2^a-torsion
- Experiments suggest: rejection probability **1/1000**

Non-Malleability of SQIsignHD

For any NM adversary $\mathcal{A}$ against a *slight modification* of SQIsignHD, there exist adversaries $\mathcal{B}$ against OneEnd and $\mathcal{C}$ against Cyclic RUGDIO indistinguishability (CR-IND) such that

$$\mathsf{Adv}^{\mathsf{NM}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{OneEnd}}(\mathcal{B}) + q_{\mathsf{Trans}} \cdot \mathsf{Adv}^{\mathsf{CR\text{-}IND}}(\mathcal{C}).$$

Authenticity of SnakeM

Theorem

For any $\mathsf{Ins\text{-}Aut}$ adversary $\mathcal{A}$ against SnakeM, there exist an adversary $\mathcal{B}$ against $\mathsf{SS\text{-}Enc}$ and an adversary $\mathcal{C}$ against $\mathsf{NM\text{-}Enc}$ such that

$$\mathsf{Adv}^{\mathsf{Ins\text{-}Aut}}_{\mathrm{SnakeM}}(\mathcal{A}) \leq +\mathsf{Adv}^{\mathsf{SS\text{-}Enc}}_{\mathrm{POKÉ},\mathrm{SQIsignHD}}(\mathcal{B}) + \mathsf{Adv}^{\mathsf{NM\text{-}Enc}}_{\mathrm{POKÉ},\mathrm{SQIsignHD}}(\mathcal{C}) + \delta.$$

Authenticity of SnakeM

Theorem

For any $\mathsf{Ins\text{-}Aut}$ adversary $\mathcal{A}$ against SnakeM, there exist an adversary $\mathcal{B}$ against $\mathsf{SS\text{-}Enc}$ and an adversary $\mathcal{C}$ against $\mathsf{NM\text{-}Enc}$ such that

$$\mathsf{Adv}^{\mathsf{Ins\text{-}Aut}}_{\mathrm{SnakeM}}(\mathcal{A}) \leq +\mathsf{Adv}^{\mathsf{SS\text{-}Enc}}_{\mathrm{POKÉ,SQIsignHD}}(\mathcal{B}) + \mathsf{Adv}^{\mathsf{NM\text{-}Enc}}_{\mathrm{POKÉ,SQIsignHD}}(\mathcal{C}) + \delta.$$

- NM: Given $\mathsf{pk}_{\mathsf{ID}}$ and transcripts $\mathcal{T} = \{(\mathsf{com}_i, \mathsf{chal}_i, \mathsf{rsp}_i)\}$, compute $(\mathsf{com}', \mathsf{chal}', \mathsf{rsp}') \notin \mathcal{T}$

Authenticity of SnakeM

Theorem

For any $\mathsf{Ins\text{-}Aut}$ adversary $\mathcal{A}$ against SnakeM, there exist an adversary $\mathcal{B}$ against $\mathsf{SS\text{-}Enc}$ and an adversary $\mathcal{C}$ against $\mathsf{NM\text{-}Enc}$ such that

$$\mathsf{Adv}^{\mathsf{Ins\text{-}Aut}}_{\mathrm{SnakeM}}(\mathcal{A}) \leq +\mathsf{Adv}^{\mathsf{SS\text{-}Enc}}_{\mathrm{POKÉ,SQIsignHD}}(\mathcal{B}) + \mathsf{Adv}^{\mathsf{NM\text{-}Enc}}_{\mathrm{POKÉ,SQIsignHD}}(\mathcal{C}) + \delta.$$

- NM: Given $\mathsf{pk}_{\mathsf{ID}}$ and transcripts $\mathcal{T} = \{(\mathsf{com}_i, \mathsf{chal}_i, \mathsf{rsp}_i)\}$, compute $(\mathsf{com}', \mathsf{chal}', \mathsf{rsp}') \notin \mathcal{T}$
- NM-Enc: Additional Enc oracle that provides a consistent "POKÉ part" of the SnakeM ciphertext:

Authenticity of SnakeM

Theorem

For any $\mathsf{Ins\text{-}Aut}$ adversary $\mathcal{A}$ against SnakeM, there exist an adversary $\mathcal{B}$ against $\mathsf{SS\text{-}Enc}$ and an adversary $\mathcal{C}$ against $\mathsf{NM\text{-}Enc}$ such that

$$\mathsf{Adv}^{\mathsf{Ins\text{-}Aut}}_{\mathrm{SnakeM}}(\mathcal{A}) \leq +\mathsf{Adv}^{\mathsf{SS\text{-}Enc}}_{\text{POKÉ},\mathrm{SQIsignHD}}(\mathcal{B}) + \mathsf{Adv}^{\mathsf{NM\text{-}Enc}}_{\text{POKÉ},\mathrm{SQIsignHD}}(\mathcal{C}) + \delta.$$

- NM: Given $\mathsf{pk}_{\mathsf{ID}}$ and transcripts $\mathcal{T} = \{(\mathsf{com}_i, \mathsf{chal}_i, \mathsf{rsp}_i)\}$, compute $(\mathsf{com}', \mathsf{chal}', \mathsf{rsp}') \notin \mathcal{T}$
- NM-Enc: Additional Enc oracle that provides a consistent “POKÉ part” of the SnakeM ciphertext:

$$\begin{tikzcd}
E_0 & E_{\mathsf{sig}} \arrow[d, "\varphi_{\mathsf{chal}}"] \\
E_{\mathsf{com}} \arrow[r, "\sigma = (q, \varphi_{\mathsf{rsp}}(U), \varphi_{\mathsf{rsp}}(V))"] & E_{\mathsf{chal}}
\end{tikzcd}$$

Authenticity of SnakeM

Theorem

For any $\mathsf{Ins\text{-}Aut}$ adversary $\mathcal{A}$ against SnakeM, there exist an adversary $\mathcal{B}$ against $\mathsf{SS\text{-}Enc}$ and an adversary $\mathcal{C}$ against $\mathsf{NM\text{-}Enc}$ such that

$$\mathsf{Adv}^{\mathsf{Ins\text{-}Aut}}_{\mathrm{SnakeM}}(\mathcal{A}) \leq +\mathsf{Adv}^{\mathsf{SS\text{-}Enc}}_{\mathrm{POKÉ,SQIsignHD}}(\mathcal{B}) + \mathsf{Adv}^{\mathsf{NM\text{-}Enc}}_{\mathrm{POKÉ,SQIsignHD}}(\mathcal{C}) + \delta.$$

- NM: Given $\mathsf{pk}_{\mathsf{ID}}$ and transcripts $\mathcal{T} = \{(\mathsf{com}_i, \mathsf{chal}_i, \mathsf{rsp}_i)\}$, compute $(\mathsf{com}', \mathsf{chal}', \mathsf{rsp}') \notin \mathcal{T}$
- $\mathsf{NM\text{-}Enc}$: Additional Enc oracle that provides a consistent "POKÉ part" of the SnakeM ciphertext:

$$\begin{tikzcd}
(P_2, Q_2) \in E_{\mathsf{enc}} \arrow[d, dashed, "\varphi'_{\mathsf{com}}"] & (P_0, Q_0) \in E_0 \arrow[d, dashed, "\varphi_{\mathsf{com}}"] & & E_{\mathsf{sig}} \arrow[d, "\varphi_{\mathsf{chal}}"] \\
(P_3, Q_3) \in E_{\mathsf{ct}} & (P_2, Q_2) \in E_{\mathsf{com}} \arrow[rr, "\sigma = (q, \varphi_{\mathsf{rsp}}(U), \varphi_{\mathsf{rsp}}(V))"] & & E_{\mathsf{chal}}
\end{tikzcd}$$

Authenticity of SnakeM

Theorem

For any $\mathsf{Ins\text{-}Aut}$ adversary $\mathcal{A}$ against SnakeM, there exist an adversary $\mathcal{B}$ against $\mathsf{SS\text{-}Enc}$ and an adversary $\mathcal{C}$ against $\mathsf{NM\text{-}Enc}$ such that

$$\mathsf{Adv}^{\mathsf{Ins\text{-}Aut}}_{\text{SnakeM}}(\mathcal{A}) \leq +\mathsf{Adv}^{\mathsf{SS\text{-}Enc}}_{\text{POKÉ,SQIsignHD}}(\mathcal{B}) + \mathsf{Adv}^{\mathsf{NM\text{-}Enc}}_{\text{POKÉ,SQIsignHD}}(\mathcal{C}) + \delta.$$

- NM: Given $\mathsf{pk}_{\mathsf{ID}}$ and transcripts $\mathcal{T} = \{(\mathsf{com}_i, \mathsf{chal}_i, \mathsf{rsp}_i)\}$, compute $(\mathsf{com}', \mathsf{chal}', \mathsf{rsp}') \notin \mathcal{T}$
- $\mathsf{NM\text{-}Enc}$: Additional Enc oracle that provides a consistent "POKÉ part" of the SnakeM ciphertext:

$$\begin{tikzcd}
(P_2, Q_2) \in E_{\mathsf{enc}} \arrow[d, dashed, "\varphi'_{\mathsf{com}}"] \arrow[dr, phantom, "\text{Adversary can decrypt :(}"] & (P_0, Q_0) \in E_0 \arrow[l, dashed, "\varphi_{\mathsf{skEnc}}"'] \arrow[d, dashed, "\varphi_{\mathsf{com}}"] & E_{\mathsf{sig}} \arrow[d, "\varphi_{\mathsf{chal}}"] \\
(P_3, Q_3) \in E_{\mathsf{ct}} & (P_2, Q_2) \in E_{\mathsf{com}} \arrow[l, dashed, "\varphi'_{\mathsf{skEnc}}"'] \arrow[r, "\sigma = (q, \varphi_{\mathsf{rsp}}(U), \varphi_{\mathsf{rsp}}(V))"] & E_{\mathsf{chal}}
\end{tikzcd}$$

Compactness – Is It Worth It?

Scheme (variant)	Confidentiality	Authenticity	Deniability	PQ	Size (in bytes)	
					ct	pk
Group-based						
DH-AKEM [ABH$^+$21]	**Ins-CCA**	**Out-Aut**	**DR***	✗	32	32
Zheng [Zhe97, BSZ02]	**Ins-CCA**	**Ins-Aut**	**HR***	✗	64	64
Lattice-based						
EtStH-AKEM (BAT + Antrag) [AJKL23]	**Ins-CCA**	**Out-Aut**	—	✓	1 119	1 417
NIKE-AKEM (Swoosh) [AJKL23]	**Ins-CCA**	**Out-Aut**	**DR***	✓	$>$ 221 184	$>$ 221 184
EaNtH-AKEM (BAT + Swoosh)	**Ins-CCA**	**Out-Aut**	**DR***	✓	473	$>$ 221 705
FrodoKEX+ [CHN$^+$24b]	**IND-1BatchCCA**	**UNF-1KCA**	**DR**	✓	72	21 300
Den. AKEM (BAT + Gandalf) [GJK24]	**Ins-CCA**	**Out-Aut**	**HR & DR**	✓	1 749	1 417
Isogeny-based						
EtStH-AKEM (POKÉ + SQIsignHD) [AJKL23]	**Ins-CCA**	**Out-Aut**	—	✓	493	432
NIKE-AKEM (CSIDH) [AJKL23]	**Ins-CCA**	**Out-Aut**	**DR***	✓	256†	256†
EaNtH-AKEM (POKÉ + CSIDH)	**Ins-CCA**	**Out-Aut**	**DR***	✓	384	624
Den. AKEM (POKÉ + Erebor) [GJK24]	**Ins-CCA**	**Out-Aut**	**HR & DR**	✓	740	432
SnakeM	**Ins-CCA**	**Ins-Aut**	**HR**	✓	296	368

Open Questions

Cryptanalysis

- $\mathsf{OW\text{-}KCA}$ of POKÉ + Countermeasures
- Additional Enc oracle in SS and NM

Other Constructions

- Though there are already some ideas...

Better Security Proof

- Reduce $\mathsf{NM\text{-}Enc}$ and $\mathsf{SS\text{-}Enc}$ to (more) standard assumptions
- Maybe in an Algebraic Isogeny Model

Questions?

☞ meers.org
✉ research@meers.org

References I

[ABF12] Afonso Arriaga, Manuel Barbosa, and Pooya Farshim. On the joint security of signature and encryption schemes under randomness reuse: Efficiency and security amplification. pages 206–223, 2012.

[ABH$^+$21] Joël Alwen, Bruno Blanchet, Eduard Hauck, Eike Kiltz, Benjamin Lipp, and Doreen Riepel. Analysing the HPKE standard. pages 87–116, 2021.

[AJKL23] Joël Alwen, Jonas Janneck, Eike Kiltz, and Benjamin Lipp. The pre-shared key modes of HPKE. pages 329–360, 2023.

[BSZ02] Joonsang Baek, Ron Steinfeld, and Yuliang Zheng. Formal proofs for the security of signcryption. pages 80–98, 2002.

[CHN$^+$24a] Daniel Collins, Loïs Huguenin-Dumittan, Ngoc Khanh Nguyen, Nicolas Rolin, and Serge Vaudenay. K-waay: Fast and deniable post-quantum X3DH without ring signatures. 2024.

[CHN$^+$24b] Daniel Collins, Loïs Huguenin-Dumittan, Ngoc Khanh Nguyen, Nicolas Rolin, and Serge Vaudenay. K-waay: Fast and deniable post-quantum X3DH without ring signatures. Cryptology ePrint Archive, Report 2024/120, 2024.

[FO99] Eiichiro Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. pages 537–554, 1999.

References II

[GJK24] Phillip Gajland, Jonas Janneck, and Eike Kiltz. Ring signatures for deniable AKEM: Gandalf's fellowship. pages 305–338, 2024.

[GPST16] Steven D. Galbraith, Christophe Petit, Barak Shani, and Yan Bo Ti. On the security of supersingular isogeny cryptosystems. pages 63–91, 2016.

[HHK17] Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz. A modular analysis of the Fujisaki-Okamoto transformation. pages 341–371, 2017.

[MOXZ24] Tomoki Moriya, Hiroshi Onuki, Maozhi Xu, and Guoqing Zhou. Adaptive attacks against FESTA without input validation or constant-time implementation. pages 3–19, 2024.

[Zhe97] Yuliang Zheng. Digital signcryption or how to achieve cost(signature & encryption) $\ll$ cost(signature) + cost(encryption). pages 165–179, 1997.

SnakeM in Detail

```
$\underline{\mathsf{SnakeM.Gen}}$
$(\mathsf{sk}_{\mathsf{KEM}}, \mathsf{pk}_{\mathsf{KEM}}) \xleftarrow{\$} \mathsf{KEM.Gen}$
$(\mathsf{sk}_{\mathsf{ID}}, \mathsf{pk}_{\mathsf{ID}}) \xleftarrow{\$} \mathsf{ID.Gen}$
$\mathsf{s} \xleftarrow{\$} \{0,1\}^{\eta}$
$\mathsf{sk} \leftarrow (\mathsf{sk}_{\mathsf{KEM}}, \mathsf{sk}_{\mathsf{ID}}, \mathsf{s})$
$\mathsf{pk} \leftarrow (\mathsf{pk}_{\mathsf{KEM}}, \mathsf{pk}_{\mathsf{ID}})$
$\textbf{return } (\mathsf{sk}, \mathsf{pk})$

$\underline{\mathsf{SnakeM.Encaps}(\mathsf{sk}_{\mathsf{SND}}, \mathsf{pk}_{\mathsf{RCV}})}$
$\textbf{parse } \mathsf{sk}_{\mathsf{SND}} = (\cdot, \mathsf{sk}_{\mathsf{ID}}, \cdot)$
$\textbf{parse } \mathsf{pk}_{\mathsf{RCV}} = (\mathsf{pk}_{\mathsf{KEM}}, \cdot)$
$\mathsf{pk}_{\mathsf{ID}} \leftarrow \mathsf{derive}(\mathsf{sk}_{\mathsf{ID}})$
$\mathsf{pk}_{\mathsf{SND}} \leftarrow \mathsf{derive}(\mathsf{sk}_{\mathsf{SND}})$
$(\mathsf{com}, R) \xleftarrow{\$} \mathsf{ID.Com}$    \\ $\mathsf{com} = \mathsf{ct}_0$
$(\mathsf{ct}_1, K) \xleftarrow{\$} \mathsf{KEM.Encaps}_1(\mathsf{pk}_{\mathsf{KEM}}, R)$
$(\mathsf{chl}, \mathsf{pad}) \leftarrow \mathsf{G}(\mathsf{pk}_{\mathsf{ID}}, \mathsf{com}, \mathsf{pk}_{\mathsf{RCV}}, \mathsf{ct}_1, K)$
$\mathsf{rsp} \xleftarrow{\$} \mathsf{ID.Rsp}(\mathsf{sk}_{\mathsf{ID}}, \mathsf{com}, \mathsf{chl}, R)$
$\mathsf{ct}_{\mathsf{rsp}} \leftarrow \mathsf{rsp} \oplus \mathsf{pad}$
$\mathsf{ct} \leftarrow (\mathsf{com}, \mathsf{ct}_1, \mathsf{ct}_{\mathsf{rsp}})$
$\mathsf{k} \leftarrow \mathsf{H}(K, \mathsf{com}, \mathsf{ct}_1, \mathsf{rsp}, \mathsf{pk}_{\mathsf{SND}}, \mathsf{pk}_{\mathsf{RCV}})$
$\textbf{return } (\mathsf{ct}, \mathsf{k})$

$\underline{\mathsf{SnakeM.Decaps}(\mathsf{pk}_{\mathsf{SND}}, \mathsf{sk}_{\mathsf{RCV}}, \mathsf{ct})}$
$\textbf{parse } \mathsf{pk}_{\mathsf{SND}} = (\cdot, \mathsf{pk}_{\mathsf{ID}})$
$\textbf{parse } \mathsf{sk}_{\mathsf{RCV}} = (\mathsf{sk}_{\mathsf{KEM}}, \cdot, \mathsf{s})$
$\textbf{parse } \mathsf{ct} = (\mathsf{com}, \mathsf{ct}_1, \mathsf{ct}_{\mathsf{rsp}})$
$\mathsf{pk}_{\mathsf{RCV}} \leftarrow \mathsf{derive}(\mathsf{sk}_{\mathsf{RCV}})$
$K \leftarrow \mathsf{KEM.Decaps}(\mathsf{sk}_{\mathsf{KEM}}, \mathsf{com}, \mathsf{ct}_1)$
$\textbf{if } K = \bot$    \\ Decaps may fail
    $K \leftarrow \mathsf{s}$
$(\mathsf{chl}, \mathsf{pad}) \leftarrow \mathsf{G}(\mathsf{pk}_{\mathsf{ID}}, \mathsf{com}, \mathsf{pk}_{\mathsf{RCV}}, \mathsf{ct}_1, K)$
$\mathsf{rsp} \leftarrow \mathsf{ct}_{\mathsf{rsp}} \oplus \mathsf{pad}$
$\textbf{if } \mathsf{ID.Ver}(\mathsf{pk}_{\mathsf{ID}}, \mathsf{com}, \mathsf{chl}, \mathsf{rsp}) = 1 :$
    $\mathsf{k} \leftarrow \mathsf{H}(K, \mathsf{com}, \mathsf{ct}_1, \mathsf{rsp}, \mathsf{pk}_{\mathsf{SND}}, \mathsf{pk}_{\mathsf{RCV}})$
    $\textbf{return } \mathsf{k}$
$\textbf{return } \bot$
```