NIST 5th PQC Standardization Conference

April 8-10, 2024

A lean BIKE KEM design for ephemeral key agreement

Nir Drucker [1], Shay Gueron [2, 3], Dusan Kostic [4]

[1] IBM Research; [2] University of Haifa; [3] Meta; [4] AWS

shay.gueron@gmail.com

BIKE - Bit Flipping Key Encapsulation

BIKE team: Nicolas Aragon, Paulo L. Barreto, Slim Bettaieb. Loïc Bidoux, Olivier Blazy. Jean-Christophe Deneuville, Philippe Gaborit, Santosh Ghosh, Shay Gueron, Tim Güneysu, Carlos Aguilar Melchor, Rafael Misoczki, Edoardo Persichetti, Jan Richter-Brockmann, Nicolas Sendrier, Jean-Pierre Tillich, Valentin Vasseur, Gilles Zémor

$(sk, h, \sigma) \leftarrow KeyGen(\cdot)$

$\sigma \xleftarrow{\$} \{0,1\}^{256}$

$h_0, h_1 \xleftarrow{D_1} S^2_{r,d}$

$sk = (h_0, h_1, \sigma)$

$pk = h = h_1 h_0^{-1}$

Return (sk, pk, σ)

$(C, K) \leftarrow Encaps(h)$

$m \xleftarrow{\$} \{0,1\}^{256}$

$e = (e_0, e_1) \xleftarrow{D_2} S_{2r,t}(m||h)$

$C = (c_0, c_1) = \left(e_0 + e_1 h, m \oplus \mathbf{L}(e_0, e_1)\right)$

$K = \mathbf{K}(m, C)$

Return (C, K)

BIKE(r,d)

$m = Decaps(sk, \sigma, h, C)$

$e' = (e'_0, e'_1) = Decode(sk, C)$

$m' = c_1 \oplus \mathbf{L}(e')$

If $S_{n,t}(m'||h) \neq e'$ then $m' = \sigma$

Return $\mathbf{K}(m', C)$

BIKE: a **Code-based** KEM

NIST seeks a **non-lattice** KEM alternative

Round 4 standardization (alternative)

Let's think of different bikes

The state of the BIKE (& some other code-based KEMs) Decoding Failure Rates: where are we now?

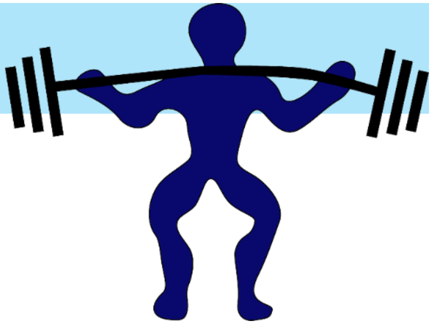
- Two NIST's code-based KEM candidates, HQC and BIKE, require a decoder with a sufficiently low DFR
- Current methods to study DFR on a given decoder rely on an assumption(s) and then some empirical estimates backed up by (extensive) simulations & extrapolations.
- This gives a **solid indication & convincing evidence** to a low DFR but not a proven upper bound of, say, 2^{-128}
- **We tested with 2^{45} messages (no decoding failure)**

Can't brush under the rug

Proven: If BIKE is used with a decoder that has sufficiently low DFR (e.g., < 2^{-128}) then BIKE has CCA security

[4] DGKP, "On the applicability of the Fujisaki-Okamoto transformation to the BIKE KEM"

CCA security is a heavy lifting!



The impact of taking on CCA challenge so

- Requires very careful constant time implementations
- All side channel attacks (HQC / BIKE) targeted a fixed key reused multiple times (CCA scenario)
- Higher complexity → Implementation mistakes
- Strict decoder specification

Why *always* pay the full cost of (hopeful) CCA when many (most?) usages settle with CPA security?

Hmmm

Forward secrecy needs ephemeral key agreement

(using key pair is used once)

BIKE CPA security has a proven reduction to a hard decoding problem

We already have support!

BIKE CPA security has a proven reduction to a hard decoding problem

$\underline{\boldsymbol{(sk, h, \sigma) \leftarrow KeyGen(\cdot)}}$

$\sigma \xleftarrow{\$} \{0,1\}^{256}$

$h_0, h_1 \xleftarrow{D_1} S_{r,d}^2$

$sk = (h_0, h_1, \sigma)$

$pk = h = h_1 h_0^{-1}$

Return (sk, pk, σ)

$\underline{\boldsymbol{(C, K) \leftarrow Encaps(h)}}$

$m \xleftarrow{\$} \{0,1\}^{256}$

$e = (e_0, e_1) \xleftarrow{D_2} S_{2r,t}(m||h)$

$C = (c_0, c_1) = \left(e_0 + e_1 h, m \oplus \boldsymbol{L}(e_0, e_1)\right)$

$K = \boldsymbol{K}(m, C)$

Return (C, K)

This is BIKE

$\underline{\boldsymbol{m = Decaps(sk, \sigma, h, C)}}$

$e' = (e'_0, e'_1) = Decode(sk, C)$

$m' = c_1 \oplus \boldsymbol{L}(e')$

If $S_{n,t}(m'||h) \neq e'$ then $m' = \sigma$

Return $\boldsymbol{K}(m', C)$

$(sk, h, \sigma) \leftarrow KeyGen(\cdot)$

$\sigma \overset{\$}{\leftarrow} \{0,1\}^{256}$

$h_0, h_1 \overset{D_1}{\leftarrow} S_{r,d}^2$

$sk = (h_0, h_1, \sigma)$

$pk = h = h_1 h_0^{-1}$

Return (sk, pk, σ)

$(C, K) \leftarrow Encaps(h)$

$m \overset{\$}{\leftarrow} \{0,1\}^{256}$

$e = (e_0, e_1) \overset{D_2}{\leftarrow} S_{2r,t}(m||h)$

$C = (c_0, c_1) = \left(e_0 + e_1 h, m \oplus \boldsymbol{L}(e_0, e_1)\right)$

$K = \boldsymbol{K}(m, C)$

Return (C, K)

This is what we pay for a CCA claim for BIKE (assuming a low DFR decoder)

$m = Decaps(sk, \sigma, h, C)$

$e' = (e'_0, e'_1) = Decode(sk, C)$

$m' = c_1 \oplus \boldsymbol{L}(e')$

If $S_{n,t}(m'||h) \neq e'$ then $m' = \sigma$

Return $\boldsymbol{K}(m', C)$

Coming soon: + binding to public key

Lean BIKE

An optimized type of BIKE design
with the minimum needed for CPA security
(to be used with ephemeral keys)

What can we peel off from BIKE

To get a Lean BIKE

- **No FO transform**
- **No re-encryption**
- **No CT-sampling**
- **No binding**
- **Choice of seed-to-PRF expansion**
- **BYOD (Bring Your Own Decoder)**

$\underline{(\boldsymbol{sk}, \boldsymbol{h}) \leftarrow \boldsymbol{KeyGen}(\cdot)}$

$h_0, h_1 \stackrel{D_1}{\leftarrow} S^2_{r,d}$

$sk = (h_0, h_1)$

$pk = h = h_1 h_0^{-1}$

Return (sk, pk, σ)

$\underline{(\boldsymbol{C}, \boldsymbol{K}) \leftarrow \boldsymbol{Encaps}(\boldsymbol{h})}$

$e = (e_0, e_1) \stackrel{D_2}{\leftarrow} S_{2r,t}$

$C = e_0 + e_1 h$

$K = \boldsymbol{K}(e, C)$

Return (C, K)

Lean BIKE

$\underline{\boldsymbol{m} = \boldsymbol{Decaps}(\boldsymbol{sk}, \boldsymbol{h}, \boldsymbol{C})}$

$e' = (e'_0, e'_1) = Decode(sk, C)$

Return $\boldsymbol{K}(m', C)$

Engineering DFR concept

- Real systems & ephemeral keys (CPA security):
  - DFR tolerance level is much more lenient than for CCA security.

- Engineering DFR: target system operational reliability

- 5 nines reliability (99.999%) gold standard of system availability
  - Translates to a DFR $\leq 2^{\log_2 10^{-5}} = 2^{-16.61}$
- 6 nines reliability (99.9999%) ➔ $2^{-19.93}$
- 7 nines reliability (99.99999%) ➔ $2^{-23.25}$

(Network errors occur at higher rates)

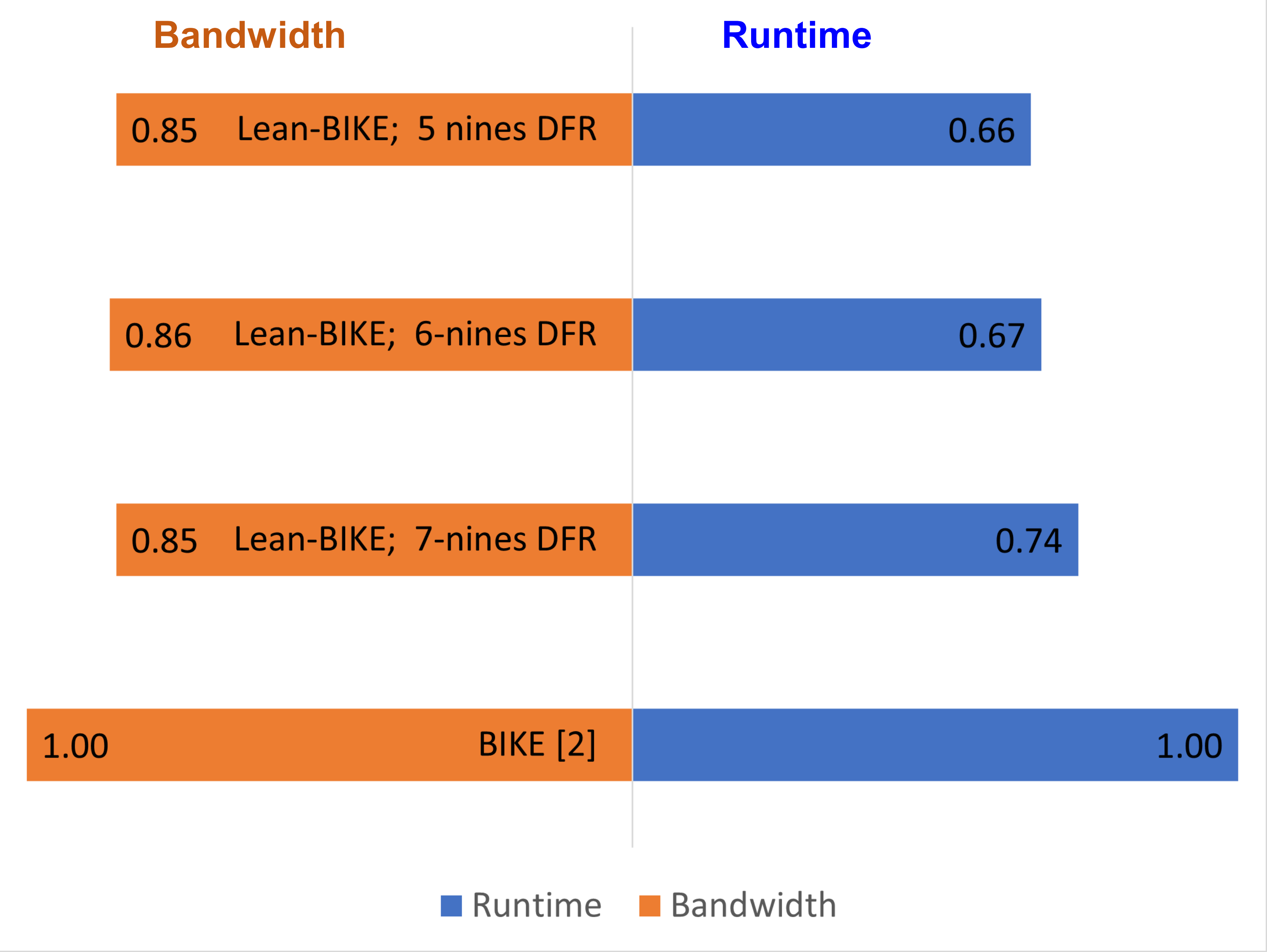
BIKE & Lean BIKE - numbers

Lean-BIKE vs. BIKE 5.1

(security Level 1)

Three levels of
Engineering DFR

And more savings are also possible



BIKE - Bit Flipping Key Encapsulation

Our concrete proposal

Standardize **both** BIKE **and** a Lean-BIKE version

Forward secrecy seeking (ephemeral key) usages

need not pay the toll

for trying to achieve CCA security

(when this is not really needed)

Lean-BIKE is available at

Drucker, Gueron, Kostic, “Additional implementation of BIKE”

https://github.com/awslabs/bike-kem

Thank you

References (1)

[1] Drucker, N., Gueron, S., Kostic, D.: On Constant-Time QC-MDPC Decoders with Negligible Failure Rate. In: Baldi, M., Persichetti, E., Santini, P. (eds.) Code- Based Cryptography. pp. 50–79. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-54074-6_4

[2] Drucker, N., Gueron, S., Kostic, D.: QC-MDPC Decoders with Several Shades of Gray. In: Ding, J., Tillich, J.P. (eds.) Post-Quantum Cryptography. pp. 35–50. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-44223-1_3

[3] Drucker, N., Gueron, S., Kostic, D.: Fast Polynomial Inversion for Post Quantum QC-MDPC Cryptography. In: Dolev, S., Kolesnikov, V., Lodha, S., Weiss, G. (eds.) Cyber Security Cryptography and Machine Learning. pp. 110–127. Springer Inter- national Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-49785-9_8

[4] Drucker, N., Gueron, S., Kostic, D., Persichetti, E.: On the applicability of the Fujisaki-Okamoto transformation to the BIKE KEM. Int. J. Comput. Math. Comput. Syst. Theory 6(4), 364–374 (2021). https://doi.org/10.1080/23799927.2021.1930176

[5] Drucker, N., Gueron, S., Kostic, D.: Binding BIKE Errors to a Key Pair. In: Dolev, S., Margalit, O., Pinkas, B., Schwarzmann, A. (eds.) Cyber Security Cryptography and Machine Learning. pp. 275–281. Springer International Publishing, Cham (2021)

[6] Guo, Q., Hlauschek, C., Johansson, T., Lahr, N., Nilsson, A., Schro ̈der, R.L.: Don't Reject This: Key-Recovery Timing Attacks Due to Rejection-Sampling in HQC and BIKE. IACR Transactions on Cryptographic Hardware and Embedded Systems 2022(3), 223–263 (Jun 2022), https://doi.org/10.46586/tches.v2022.i3.223-263

[7] Drucker, N., Gueron, S., Kostic, D.: To Reject or Not Reject: That Is the Question. The Case of BIKE Post Quantum KEM. In: Latifi, S. (ed.) ITNG 2023 20th International Conference on Information Technology-New Generations. pp. 125–131. Springer International Publishing, Cham (2023)

[8] Wang, T., Wang, A., Wang, X.: Exploring decryption failures of bike: New class of weak keys and key recovery attacks. In: Handschuh, H., Lysyanskaya, A. (eds.) Ad- vances in Cryptology – CRYPTO 2023. pp. 70–100. Springer Nature Switzerland, Cham (2023)