BIKE - Bit-Flipping Key Encapsulation

Presented to the 2nd NIST Post-Quantum Cryptography Standardization Conference
August, 24th 2019, Santa Barbara, California, USA

Authors:
Nicolas Aragon
Paulo S. L. M. Barreto
Slim Bettaieb
Loïc Bidoux
Olivier Blazy
Jean-Christophe Deneuville
Philippe Gaborit
Shay Gueron
Tim Güneysu
Carlos Aguilar Melchor
Rafael Misoczki (presenter)
Edoardo Persichetti
Nicolas Sendrier
Jean-Pierre Tillich
Valentin Vasseur (new member)
Gilles Zémor

Affiliations:
University of Limoges, France
University of Washington Tacoma, USA
Worldline, France
Worldline, France
University of Limoges, France
Federal University of Toulouse, ENAC, France
University of Limoges, France
University of Haifa, and Amazon Web Services, Israel
Ruhr-Universität Bochum, and DFKI, Germany,
University of Toulouse, France
Intel Corporation, USA
Florida Atlantic University, USA
INRIA, France
INRIA, France
INRIA, France
IMB, University of Bordeaux, France

https://bikesuite.org – team@bikesuite.org
BIKE Recap

• McEliece-like KEM with QC-MDPC Codes

• Well-Understood & Reliable Security
  • Theoretical Security: Reduction based on well-known coding-theory problems
  • Practical Security: ISD-based attacks [Pra62] whose work-factor* barely changed in ~50 years of research

• Performance
  • **Practical performance** for all KeyGen, Encaps, Decaps steps regarding both computational complexity and bandwidth

*Tip: Ideal usage as Ephemeral Key Exchange (e.g. SSL/TLS)*
"BIKE targets IND-CPA security and makes no attempt to make it difficult for an attacker to mount a chosen ciphertext attack if keys are reused. This design decision was made by the submitters, based on the difficulty of designing a bit-flipping decoder with a low enough decoding failure rate to allow an efficient IND-CCA2-secure construction."

NIST IR Report 8240, page 11, Section 3.12
New Backflip Decoder

• Context
  • Round 1 decoder: efficient but fails with non-negligible probability ($10^{-7}$)
  • To enable IND-CCA variants, negligible decoding probability was needed

• Backflip Rationale
  • Similar to Bit-Flipping
  • **Difference**: each bit flip keeps a time-to-live counter. After a given number of iterations, the bit flip reaches a time-to-death point and is flipped back
  • **Result**: Based on an extrapolation argument, it is possible to show that a certain parameter set attains an arbitrarily low failure rate using Backflip

https://bikesuite.org  —  team@bikesuite.org
New BIKE CCA Variants

• Core Ingredients
  • Backflip Decoder
  • [HHK17] like conversions (with bounds from [JZC18, JZM19])

• CPA→CCA conversion preserved the strong points of each variant

<table>
<thead>
<tr>
<th>Strong Points among CPA Variant</th>
<th>Strong Points among CCA Variant</th>
</tr>
</thead>
<tbody>
<tr>
<td>BIKE-1 Fastest KG+Encaps+Decaps among CPA variants</td>
<td>Fastest KG+Encaps+Decaps among CCA variants</td>
</tr>
<tr>
<td>BIKE-2 Smallest ciphertext among CPA variants</td>
<td>Smallest ciphertext among CCA variants</td>
</tr>
<tr>
<td>BIKE-3 Security reduction to single problem</td>
<td>Security reduction to single problem</td>
</tr>
</tbody>
</table>

• CCA Variants enable static keys. Current focus remains CPA Ephemeral

https://bikesuite.org – team@bikesuite.org
"BIKE offers key and ciphertext sizes and performance that are competitive with ring and module lattice schemes (especially at the lower security categories)."

NIST IR Report 8240, page 11, Section 3.12
BIKE CCA -- constant time implementations
Nir Drucker \(^1,2\), Shay Gueron \(^1,2\), Dusan Kostic \(^1,3\)
(1) Amazon  (2) University of Haifa  (3) EPFL

- New BIKE CCA implementation in constant time -- C, AVX2, AVX512
  - Constant time algorithm definition for the CCA decoder
  - Constant time implementation for the CCA BIKE flows

- Conclusions
  - It is possible to define and implement BIKE CCA in constant time
  - Performance costs are tolerable

- “Additional” code package & detailed report to be released/published soon

https://bikesuite.org  --  team@bikesuite.org
BIKE CCA -- constant time implementations
Nir Drucker $^{1,2}$, Shay Gueron $^{1,2}$, Dusan Kostic $^{1,3}$
(1) Amazon  (2) University of Haifa  (3) EPFL

Parameter sets targeting NIST Security Level 1

<table>
<thead>
<tr>
<th>BIKE</th>
<th>CPA Cycles</th>
<th>CCA Cycles</th>
<th>CCA/CPA</th>
<th>CPA Bandwidth</th>
<th>CCA Bandwidth</th>
<th>CCA-CPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>BIKE1</td>
<td>KeyGen</td>
<td>250K</td>
<td>270K</td>
<td>1.08x</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Encaps</td>
<td>180K</td>
<td>210K</td>
<td>1.17x</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Decaps</td>
<td>1.9M</td>
<td>8.1M</td>
<td>4.18x</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\rightarrow$ 2.48KB</td>
<td>$\leftarrow$ 2.48KB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\rightarrow$ 2.88KB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\leftarrow$ 2.88KB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\Delta=400B$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BIKE2</td>
<td>KeyGen</td>
<td>4.8M</td>
<td>6.3M</td>
<td>1.30x</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Encaps</td>
<td>140K</td>
<td>170K</td>
<td>1.21x</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Decaps</td>
<td>1.9M</td>
<td>9.0M</td>
<td>4.64x</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\rightarrow$ 1.24KB</td>
<td>$\leftarrow$ 1.24KB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\rightarrow$ 1.44KB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\leftarrow$ 1.47KB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\Delta=400B$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BIKE3</td>
<td>KeyGen</td>
<td>220K</td>
<td>250K</td>
<td>1.14x</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Encaps</td>
<td>220K</td>
<td>280K</td>
<td>1.27x</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Decaps</td>
<td>2.5M</td>
<td>13.8M</td>
<td>5.58x</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\rightarrow$ 2.69KB</td>
<td>$\leftarrow$ 2.69KB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\rightarrow$ 2.69KB</td>
<td>$\leftarrow$ 2.69KB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\Delta=310B$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In red message size for
BIKE-3 with compressed $g$
BIKE Real Experiment with s2n AWS TLS library

• **s2n** is an AWSLabs open source library for TLS
  • Small, fast, with simplicity as a priority
  • Removes a lot of cruft that has built-up in libssl
  • Currently handles all of the S3 traffic today

• **PQ-TLS 1.2 – hybrid key exchange in s2n**
  • Added SIKE and BIKE (reference code) into the s2n code base
  • Added a hybrid key exchange cipher suites into s2n
    • TLS_ECDHE_BIKE_RSA_WITH_AES_256_GCM_SHA384
    • TLS_ECDHE_SIKE_RSA_WITH_AES_256_GCM_SHA384
  • Applied the same rigor to this new code as in all of s2n
  • Open Source implementation will be released soon

**Conclusion:** feasible to use “Classical + BIKE” hybrid in a real networking application

http://bikesuite.org – team@bikesuite.org
"Security strengths are based on information-set-decoding attacks, which have a long history of analysis during which the complexity of such attacks have not greatly changed."

NIST IR Report 8240, page 11, Section 3.12

"Possible areas for further analysis related to BIKE include ... investigating the effect, if any, of the quasi-cyclic code structure on security."

NIST IR Report 8240, page 12, Section 3.12
The Effect of Quasi-Cyclic Code Structure on Security

• QC-MDPC Parameters are selected considering three ISD-related attacks
  • Key distinguishing attack: Exhibit one codeword of $C^\perp$ of weight $w$
  • Key recovery attack: Exhibit $r$ codewords of $C^\perp$ of weight $w$
  • Decoding attack: Decode $t$ errors in a $(n, n - r)$-linear code.

• ISD algorithms assume a list of solution candidates of size $L$. Each candidate has a probability $P$ to produce a solution. Under optimal conditions: $WF_{ISD} (n, r, t) \approx L/P$

• [Sen11] shows that the gain when the decoding problem has $N_s$ solutions and when $N_i$ instances are treated simultaneously is: $N_s/\sqrt{N_i}$

<table>
<thead>
<tr>
<th></th>
<th>MDPC</th>
<th>QC-MDPC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key distinguishing</td>
<td>$\frac{1}{r}WF_{isd} (n, n - r, w)$</td>
<td>$\frac{1}{r}WF_{isd} (n, n - r, w)$</td>
</tr>
<tr>
<td>Key recovery</td>
<td>$WF_{isd} (n, n - r, w)$</td>
<td>$\frac{1}{r}WF_{isd} (n, n - r, w)$</td>
</tr>
<tr>
<td>Decoding</td>
<td>$WF_{isd} (n, r, t)$</td>
<td>$\frac{1}{\sqrt{r}}WF_{isd} (n, r, t)$</td>
</tr>
</tbody>
</table>

See [Mis13] for a detailed analysis
Smaller Updates & Final Remarks

• Smaller updates
  • BIKE-3 variant that generates $g$ from a seed, saving almost 50% communication
  • Fixed decoding threshold computation in reference & optimization code, which now matches the spec, accelerating decoding. No changes in additional code;
  • Fixed buffer overflows in reference & optimization code;

• Final remarks
  • BIKE has well-understood, reliable security & practical performance
  • BIKE is particularly appealing for low-level security (e.g. Level 1)
  • Given CPA focus, variants with fast key generation (e.g. BIKE-1, BIKE-3) are our priority
  • NIST Report 8240 already highlights benefits of BIKE and the team addressed requests
References


https://bikesuite.org

Authors:
Nicolas Aragon
Paulo S. L. M. Barreto
Slim Bettaieb
Loïc Bidoux
Olivier Blazy
Jean-Christophe Deneuville
Philippe Gaborit
Shay Gueron
Tim Güneysu
Carlos Aguilar Melchor
Rafael Misoczki (presenter)
Edoardo Persichetti
Nicolas Sendrier
Jean-Pierre Tillich
Valentin Vasseur (new member)
Gilles Zémor

Affiliation:
University of Limoges, France
University of Washington Tacoma, USA
Worldline, France
Worldline, France
University of Limoges, France
Federal University of Toulouse, ENAC, France
University of Limoges, France
University of Haifa, and Amazon Web Services, Israel
Ruhr-Universität Bochum, and DFKI, Germany,
University of Toulouse, France
Intel Corporation, USA
Florida Atlantic University, USA
INRIA, France
INRIA, France
INRIA, France
IMB, University of Bordeaux, France