BIKE - Bit-Flipping Key Encapsulation

Presented to the NIST Post-Quantum Cryptography Standardization Conference
April, 13th 2018, Fort Lauderdale, Florida, USA

Authors:	**Affiliations:**
Nicolas Aragon	University of Limoges, France
Paulo S. L. M. Barreto	University of Washington Tacoma, USA
Slim Bettaieb	Worldline, France
Loïc Bidoux	Worldline, France
Olivier Blazy	University of Limoges, France
Jean-Christophe Deneuville	INSA-CVL Bourges and University of Limoges, France
Philippe Gaborit	University of Limoges, France
Shay Gueron	University of Haifa, and Amazon Web Services, Israel
Tim Güneysu	Ruhr-Universität Bochum, and DFKI, Germany,
Carlos Aguilar Melchor	University of Toulouse, France
Rafael Misoczki	Intel Corporation, USA
Edoardo Persichetti	Florida Atlantic University, USA
Nicolas Sendrier	INRIA, France
Jean-Pierre Tillich	INRIA, France
Gilles Zémor	IMB, University of Bordeaux, France

http://bikesuite.org – team@bikesuite.org

Agenda

1. BIKE Suite (BIKE-1, BIKE-2 and BIKE-3)
2. Decoding
3. Security
4. Performance
5. Final Remarks

BIKE Suite: BIKE-1, BIKE-2, BIKE-3

Ingredients:

- McEliece encryption framework
- QC-MDPC Codes [MTSB12]
- CAKE [BGGM17], Ouroboros [DGZ17]
- Ephemeral keys

Download:

http://bikesuite.org

Design rationale:

- Security:
  - Based on well-known coding problems (MDPC approximates distinguishing to decoding problem)
  - Ephemeral keys defeat recent reaction-attacks against probabilistic decoding [GJS16]
- Efficiency:
  - Quasi-cyclic property: ensures small public keys
  - Simple operations: product and addition of binary polynomials/vector
  - Decoding: bit-flipping decoding techniques
- Several <u>trade-offs</u> were possible, thus we decided to submit a cipher suite with <u>3 variants</u>

BIKE Suite: Message Protocol

Alice **Bob**

1. Generate Ephemeral QC-MDPC key pair (sk, pk)

2. Send (pk) →

3. Generate sparse error vector e
4. Derive symmetric key **K** from error vector e
5. Encrypt e using pk to produce ciphertext ct

6. Send (ct) ←

7. Decrypt ct using sk to recover e or $\perp$
8. Derive symmetric key **K** from error vector e

Differences Between BIKE Variants

Let $R = F_2[X]/(X^r - 1)$, r is prime such as $(X^r - 1)/(X - 1)$ is irreducible.
All quantities are elements of R. The function $\mathbf{K}$ is a hash function.

	BIKE-1	BIKE-2	BIKE-3
SK	(h_0, h_1) with $\lvert h_0 \rvert = \lvert h_1 \rvert = w/2$		
PK	$(f_0, f_1) \leftarrow (gh_1, gh_0)$	$(f_0, f_1) \leftarrow (1, h_1 h_0^{-1})$	$(f_0, f_1) \leftarrow (h_1 + gh_0, g)$
Enc	$(c_0, c_1) \leftarrow (mf_0 + e_0, mf_1 + e_1)$	$c \leftarrow e_0 + e_1 f_1$	$(c_0, c_1) \leftarrow (e + e_1 f_0, e_0 + e_1 f_1)$
	$K \leftarrow \mathbf{K}(e_0, e_1)$		
Dec	$s \leftarrow c_0 h_0 + c_1 h_1$; $u \leftarrow 0$	$s \leftarrow ch_0$; $u \leftarrow 0$	$s \leftarrow c_0 + c_1 h_0$; $u \leftarrow t/2$
	$(e'_0, e'_1) \leftarrow \texttt{Decode}(s, h_0, h_1, u)$		
	$K \leftarrow \mathbf{K}(e'_0, e'_1)$		

BIKE Suite: Different Trade-Offs

	Latency	*Communication*		*Decoding*		*Security*	
	Fast key generation (inversion-less)	*Message 1 (1 element of R)*	*Message 2 (1 element of R)*	*Bit Flipping Decoding*	*Noisy BF Decoding*	*Codeword Finding + Syndrome Decoding*	*Syndrome Decoding*
BIKE-1	✓			✓		✓	
BIKE-2	*	✓	✓	✓		✓	
BIKE-3	✓	✓**			✓		✓

* : BIKE-2 key generation can have the amortized cost significantly improved by means of batched process.

**: BIKE-3 can send only 1 element of R plus a seed.

Decoding QC-MDPC is Simple

- Bit Flipping Decoding Algorithm [Gal63]:
  1. Compute syndrome from error vector: $s^T \leftarrow eH^T$
  2. Flip bits associated with most unsatisfied parity check equations
  3. Successfully terminate if syndrome is 0; restart from step 1 otherwise
- MDPC probabilistic decoding:
  - Decoding failure rate is small enough for practical purposes
  - Full theoretical analysis is difficult; recent promising results
- BIKE decoding approach:
  - New 2-Steps Bit-Flipping Variant
  - Recent reaction-attack [GJS16] on decoding failures: defeated by ephemeral keys

BIKE has Security Reductions to Well-Known Generic Hard Problems

Theorem:

1. **BIKE-1**, **BIKE-2** are IND-CPA secure in the Random Oracle Model under the **(2,1)-QCCF** and **(2,1)-QCSD** assumptions.
2. **BIKE-3** is IND-CPA secure in the Random Oracle Model under the **(3,1)-QCSD** and **(2,1)-QCSD** assumptions.

(2,1)-QC Codeword Finding:
Instance: $h \in R$, integer $t > 0$
Property: There exists $c_0, c_1 \in R$ such that $|c_0| + |c_1| = t$ and $c_0 + c_1 h = 0$

(2,1)-QC Syndrome Decoding:
Instance: $s, h \in R$, an integer $t > 0$
Property: There exists $e_0, e_1 \in R$ such that $|e_0| + |e_1| \leq t$ and $e_0 + e_1 h = s$

(3,1)-QC Syndrome Decoding:
Instance: $s_0, s_1, h_0, h_1 \in R$, $t > 0$
Property: There exists $e_0, e_1, e_2 \in R$ s.t. $|e_0| + |e_1| + |e_2| \leq 3t/2$ and $e_0 + e_2 h_0 = s_0$ and $e_1 + e_2 h_1 = s_1$

BIKE Enjoys a Straightforward Practical Security Assessment

- The best techniques to solve codeword finding and syndrome decoding:
  - Variants of Prange's Information Set Decoding (ISD) [Pran62]
  - Work factor of any ISD variant A to decode t errors in a (n, k)-binary code:

$$WF_A(n, k, t) = 2^{ct(1+o(1))}$$

- The quasi-cyclic case:
  - Codeword finding and decoding are a bit easier for quasi-cyclic codes.
  - Adversary gains a factor r for codeword finding and factor $\sqrt{r}$ for decoding
- The best quantum attack against BIKE:
  - Grover's algorithm [Gro96] applied to ISD

Affordable Latency

		Level 1					Level 3					Level 5				
		Reference	*Additional*		*Add. Const. Time*		*Reference*	*Additional*		*Add. Const. Time*		*Reference*	*Additional*		*Add. Const. Time*	
		--	*AVX2*	*AVX2512*	*AVX2*	*AVX512*	*--*	*AVX2*	*AVX512*	*AVX2*	*AVX512*	*--*	*AVX2*	*AVX512*	*AVX2*	*AVX512*
BIKE-1	KG	**0.73**	0.09	0.09	**0.20**	**0.19**	1.70	0.25	0.25	0.45	0.45	2.98	0.25	0.25	0.67	0.69
	Enc	**0.68**	0.11	0.11	**0.15**	**0.13**	1.85	0.28	0.27	0.36	0.33	3.02	0.29	0.27	0.42	0.36
	Dec	**2.90**	1.13	1.02	**5.30**	**4.86**	7.66	3.57	2.99	16.74	15.26	17.48	2.75	2.24	9.84	8.27
BIKE-2	KG	**6.38**	4.38	4.38	**4.46**	**4.45**	22.20	7.77	7.79	8.04	8.05	58.80	11.99	11.99	12.45	12.34
	Enc	**0.28**	0.09	0.08	**0.12**	**0.11**	0.71	0.17	0.18	0.27	0.23	1.20	0.27	0.25	0.39	0.34
	Dec	**2.67**	1.12	0.86	**5.55**	**5.12**	7.11	2.88	3.48	17.36	15.63	16.38	2.70	2.14	10.74	8.93

Performance in millions of cycles.
Implementation uses NTL Library.

Reference implementation measured on Intel® Core™ i5-6260U CPU @1.80GHz. Additional implementation measured on Intel® Core™ Intel® Xeon® Platinum 8124M CPU @3GHz.

		Level 1	Level 3	Level 5
BIKE-3	KeyGen	**0.43**	1.10	2.30
	Enc	**0.57**	1.46	3.25
	Dec	**3.43**	7.73	18.04

Affordable Bandwidth

BIKE-1

	Size	Level 1	Level 3	Level 5
Message 1	n	2.48	5.34	7.99
Message 2	n	2.48	5.34	7.99

BIKE-2

	Size	Level 1	Level 3	Level 5
Message 1	r	1.24	2.67	3.99
Message 2	r	1.24	2.67	3.99

BIKE-3

	Size	Level 1	Level 3	Level 5
Message 1	r+256	1.37	2.67	4.44
Message 2	n	2.69	5.29	8.82

Message sizes given in KBytes.

Final Remarks

- BIKE Suite provides 3 code-based key encapsulation schemes
- Underlying security: well-known hard-problems from coding-theory
- Affordable latency and communication bandwidth
- Simple operations: product/addition of binary vectors + bit flipping

The BIKE suite is <u>secure</u>, simple, efficient and flexible

It offers the various trade-offs needed to meet the heterogeneous requirements of modern cryptographic applications (IoT devices, Internet, Data Centers, …)

http://bikesuite.org

Authors:	**Affiliation:**
Nicolas Aragon	University of Limoges, France
Paulo S. L. M. Barreto	University of Washington Tacoma, USA
Slim Bettaieb	Worldline, France
Loïc Bidoux	Worldline, France
Olivier Blazy	University of Limoges, France
Jean-Christophe Deneuville	INSA-CVL Bourges and University of Limoges, France
Philippe Gaborit	University of Limoges, France
Shay Gueron	University of Haifa, and Amazon Web Services, Israel
Tim Güneysu	Ruhr-Universität Bochum, and DFKI, Germany,
Carlos Aguilar Melchor	University of Toulouse, France
Rafael Misoczki	Intel Corporation, USA
Edoardo Persichetti	Florida Atlantic University, USA
Nicolas Sendrier	INRIA, France
Jean-Pierre Tillich	INRIA, France
Gilles Zémor	IMB, University of Bordeaux, France

References

- [BGGM17]: Paulo S. L. M. Barreto, Shay Gueron, Tim Guneysu, Rafael Misoczki, Edoardo Persichetti, Nicolas Sendrier, and Jean-Pierre Tillich. CAKE: Code-based Algorithm for Key Encapsulation. 16th IMA International Conference on Cryptography and Coding. 2017.
- [DGZ17]: Jean-Christophe Deneuville, Philippe Gaborit, Gilles Zémor. Ouroboros: A Simple, Secure and Efficient Key Exchange Protocol Based on Coding Theory. PQCrypto 2017: 18-34
- [Gal63]: R. G. Gallager. Low-Density Parity-Check Codes. PhD thesis, M.I.T., 1963.
- [GJS16]: Qian Guo, Thomas Johansson, and Paul Stankovski. A Key Recovery Attack on MDPC with CCA Security Using Decoding Errors, pages 789, 815. Springer Berlin Heidelberg, Berlin, Heidelberg, 2016
- [Gro96]: Grover L.K.: A fast quantum mechanical algorithm for database search, Proceedings, 28th Annual ACM Symposium on the Theory of Computing, (May 1996) p. 212
- [MTSB12]: R. Misoczki, J.-P. Tillich, N. Sendrier, and P. L.S.M. Barreto. MDPC McEliece: New McEliece variants from moderate density parity-check codes. In IEEE International Symposium on Information Theory, ISIT'2013, pages 2069, 2073, Istanbul, Turkey, 2013.
- [Pra62]: Eugene Prange. The use of information sets in decoding cyclic codes. IRE Transactions, IT-8:S5 S9, 1962.