HAETAE Update to v2.0

Jung Hee Cheon[1,2], **Hyeongmin Choe**[1], Julien Devevey, Tim Güneysu[3, 4], Dongyeon Hong, Markus Krausz[3], Georg Land[3], Junbum Shin[2], Damien Stehlé[2], MinJune Yi[1]

[1]Seoul National University, [2]CryptoLab Inc.,
[3]Ruhr Universität Bochum, [4]DFKI

7th KpqC workshop
November 13, 2023

HAETAE
HEAAN
CRYPTO LAB

Introduction to HAETAE v2.0

HAETAE will be updated soon!

Based on the feedback from

- 5th & 6th KpqC Workshops
- 1st & 2nd KIAS-JBNU KpqC Workshops

- KpqC Bulletin (`https://groups.google.com/g/kpqc-bulletin`)
- PQC Forum (`https://groups.google.com/a/list.nist.gov/g/pqc-forum`)

- Many researchers from the crypto/security community who are interested in HAETAE

Introduction to HAETAE v2.0

HAETAE will be updated soon!

Based on the feedback from

- 5th & 6th KpqC Workshops
- 1st & 2nd KIAS-JBNU KpqC Workshops

- KpqC Bulletin (`https://groups.google.com/g/kpqc-bulletin`)
- PQC Forum (`https://groups.google.com/a/list.nist.gov/g/pqc-forum`)

- Many researchers from the crypto/security community who are interested in HAETAE

Introduction to HAETAE v2.0

HAETAE will be updated soon!

Based on the feedback from

- 5th & 6th KpqC Workshops
- 1st & 2nd KIAS-JBNU KpqC Workshops

- KpqC Bulletin (`https://groups.google.com/g/kpqc-bulletin`)
- PQC Forum (`https://groups.google.com/a/list.nist.gov/g/pqc-forum`)

- Many researchers from the crypto/security community who are interested in HAETAE

New Specification

Our specification document is significantly updated to better **specify** HAETAE!

- Easy to understand!
  - Linking theorems, description, and reference code,
  - Changed variable/function names
- Complete security proof
  - Missing details for QROM security reductions,
- Implementation-friendly!
  - Implementation-oriented specification,
  - AVX2 optimization,
  - Memory optimization for embedded devices.

New Specification

Our specification document is significantly updated to better **specify** HAETAE!

- Easy to understand!
  - Linking theorems, description, and reference code,
  - Changed variable/function names
- Complete security proof
  - Missing details for QROM security reductions,
- Implementation-friendly!
  - Implementation-oriented specification,
  - AVX2 optimization,
  - Memory optimization for embedded devices.

New Specification

Our specification document is significantly updated to better **specify** HAETAE!

- Easy to understand!
  - Linking theorems, description, and reference code,
  - Changed variable/function names
- Complete security proof
  - Missing details for QROM security reductions,
- Implementation-friendly!
  - Implementation-oriented specification,
  - AVX2 optimization,
  - Memory optimization for embedded devices.

Bug Fixes

Some reported/unreported implementation bugs are fixed.

- Hyperball sampler
  - bug fix for parameters of HAETAE-260.
- rANS encoding
  - some implementation-specific attacks [Saa23, Tea23],
  - now fixed.
- Verification key compression
  - bug fix for the parameters enabling compressions.

Bug Fixes

Some reported/unreported implementation bugs are fixed.

- Hyperball sampler
  - bug fix for parameters of HAETAE-260.
- rANS encoding
  - some implementation-specific attacks [Saa23, Tea23],
  - now fixed.
- Verification key compression
  - bug fix for the parameters enabling compressions.

Bug Fixes

Some reported/unreported implementation bugs are fixed.

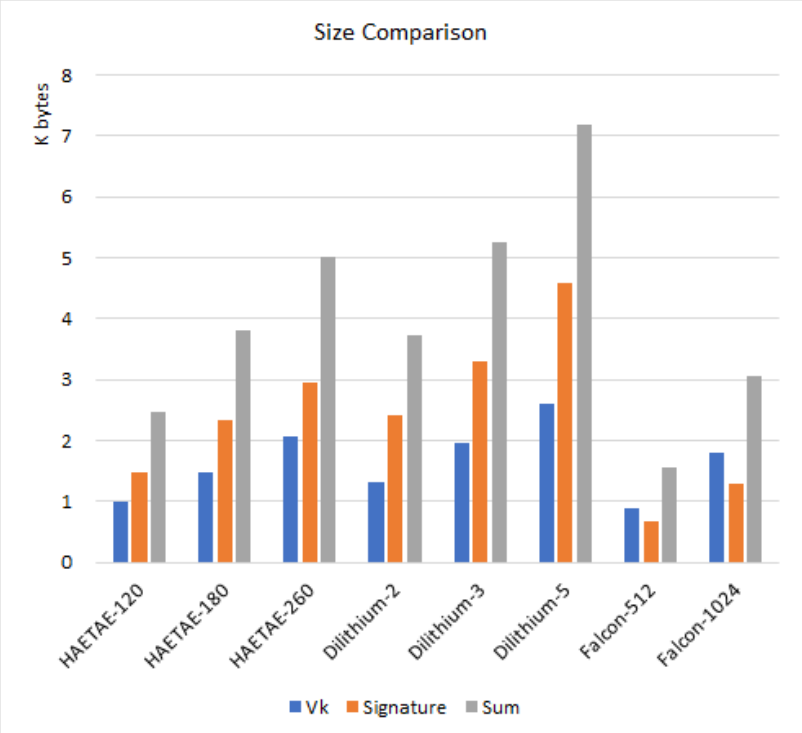
- Hyperball sampler
  - bug fix for parameters of HAETAE-260.
- rANS encoding
  - some implementation-specific attacks [Saa23, Tea23],
  - now fixed.
- Verification key compression
  - bug fix for the parameters enabling compressions.

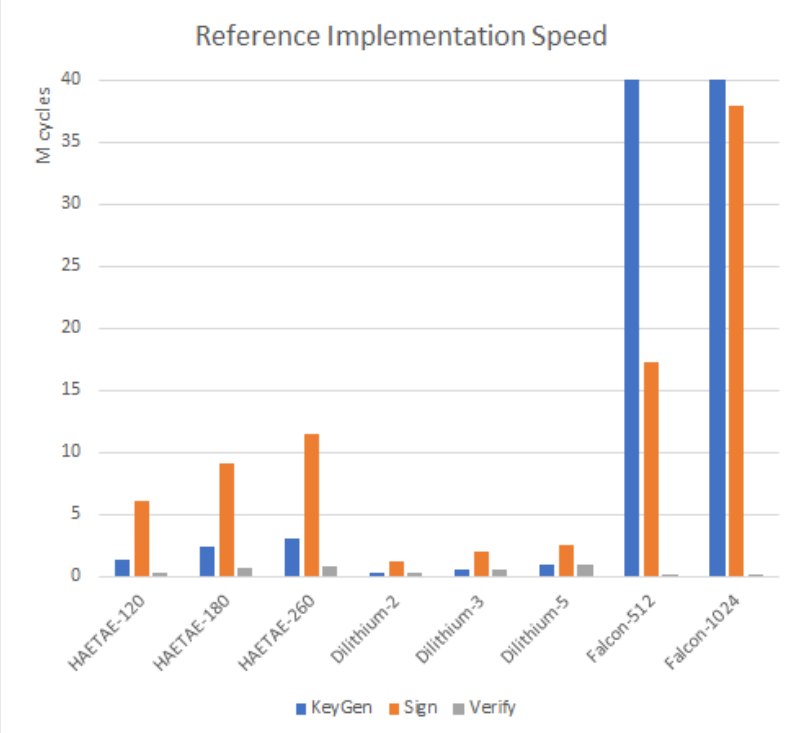
Numbers - Updated Reference Implementation

Scheme	KeyGen	Sign	Verify
HAETAE-120	1,403,402	6,039,674	376,486
HAETAE-180	2,368,038	9,161,312	691,652
HAETAE-260	3,101,280	11,444,678	895,098
Dilithium-2	343,222	1,191,218	376,008
Dilithium-3	630,170	2,061,816	612,538
Dilithium-5	945,776	2,522,834	987,154
Falcon-512	53,778,476	17,332,716	103,056
Falcon-1024	154,298,384	38,014,050	224,378

Table: Reference implementation speeds. Median cycle counts of 1000 executions for HAETAE, Dilithium, and Falcon. Cycle counts were obtained on one core of an Intel Core i7-10700k, with TurboBoost and hyperthreading disabled.

Numbers - Updated Reference Implementation



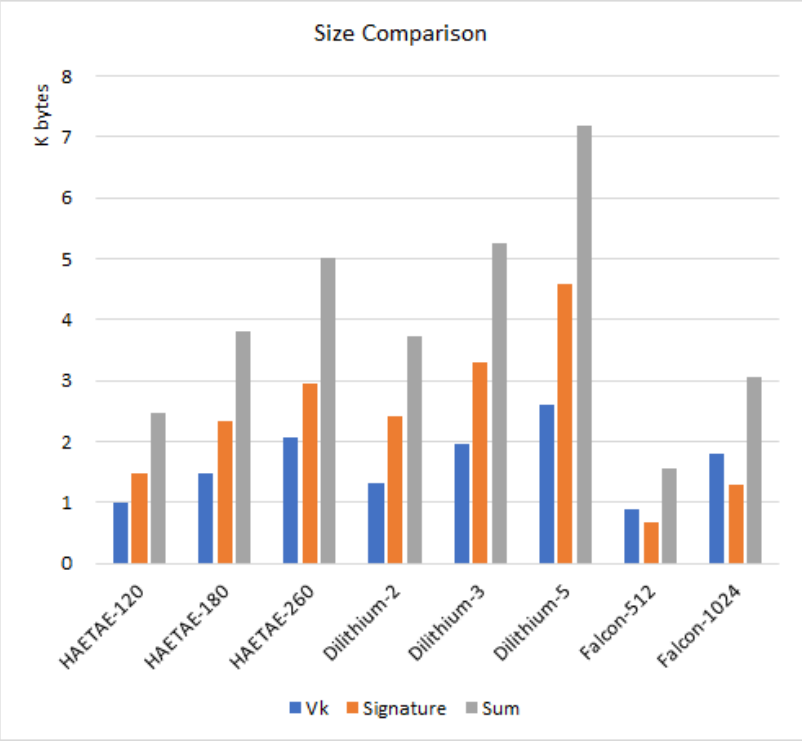


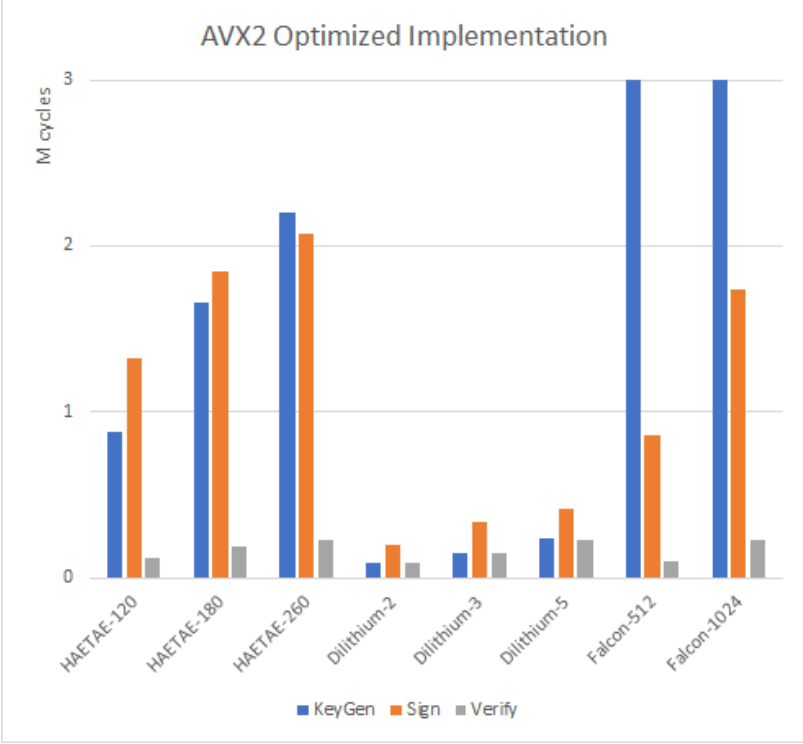
Numbers - AVX2-optimized Implementation

Scheme	KeyGen	Sign	Verify
HAETAE-120	882,350	1,323,118	115,638
HAETAE-180	1,654,464	1,844,610	183,920
HAETAE-260	2,199,678	2,069,734	223,852
Dilithium-2	87,020	200,242	92,148
Dilithium-3	146,560	334,898	148,810
Dilithium-5	233,976	415,228	232,146
Falcon-512	24,663,306	863,076	100,540
Falcon-1024	71,013,520	1,740,188	228,086

Table: AVX2 optimized implementation speeds. Median cycle counts of 1000 executions for HAETAE, Dilithium, and Falcon. Cycle counts were obtained on one core of an Intel Core i7-10700k, with TurboBoost and hyperthreading disabled.

Numbers - AVX2 optimized Implementation





Numbers - Embedded Implementation on Cortex-M4

Scheme		KeyGen	Sign	Verify
HAETAE-120	*speed-opt*	19,796	54,564	22,532
	stack-opt	17,364	40,732	22,532
HAETAE-180	*speed-opt*	29,612	69,631	31,020
	stack-opt	22,444	57,116	31,020
HAETAE-260	*speed-opt*	34,108	102,964	36,428
	stack-opt	22,356	68,380	36,428
Dilithium-2		38,408	49,380	36,212
Dilithium-3		60,836	68,836	57,724
Dilithium-5		97,692	115,932	92,788
Falcon-512		18,416	42,508	4,724
Falcon-1024		36,296	82,532	8,820

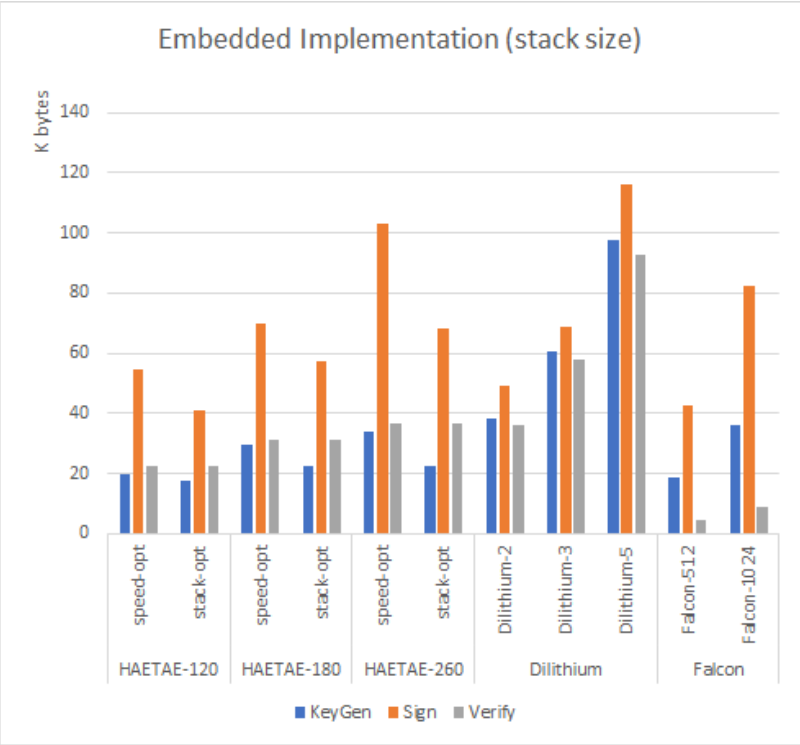
Table: Maximum stack size in bytes for Cortex-M4 implementations of HAETAE, Dilithium, and Falcon.

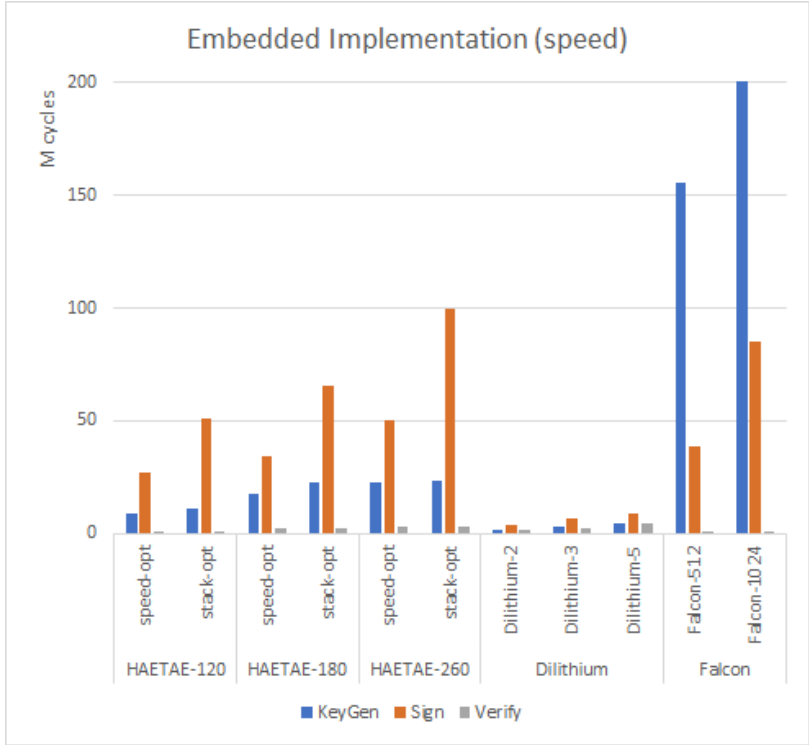
Numbers - Embedded Implementation on Cortex-M4

Scheme		KeyGen	Sign	Verify
HAETAE-120	*speed-opt*	8,904,552	27,311,965	1,054,478
	stack-opt	10,818,804	51,016,745	1,054,472
HAETAE-180	*speed-opt*	17,666,326	34,466,279	2,026,448
	stack-opt	22,859,766	65,854,630	2,026,454
HAETAE-260	*speed-opt*	22,850,880	50,174,603	2,733,469
	stack-opt	23,213,004	99,471,768	2,733,451
Dilithium-2		1,597,999	4,111,596	1,571,804
Dilithium-3		2,830,024	6,588,465	2,691,283
Dilithium-5		4,826,422	8,779,067	4,705,693
Falcon-512		155,757,768	38,979,435	481,452
Falcon-1024		480,071,949	85,125,001	994,972

Table: Average cycle counts of 1000 (100 for Falcon) executions on the Cortex-M4 for HAETAE, Dilithium, and Falcon.

Numbers - Embedded Implementation on Cortex-M4





Thanks!

A paper version of HAETAE v2.0 can be found at

`ia.cr/2023/624`.

Any question?

References I

[Saa23] Markku-Juhani O. Saarinen.

Buffer overflows in haetae / on crypto vs implementation errors, 2023.

Available at https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/bkJKBFq3TDY/m/lTCum6zgBQAJ.

[Tea23] HuFu Team.

rans signature compression done right, 2023.

Available at http://123.56.244.4/rANS.pdf.