HuFu

Yang Yu, Huiwen Jia, Leibo Li, Delong Ran,
Zhiyuan Qiu, Shiduo Zhang, Xiuhan Lin, Xiaoyun Wang

The 2nd Oxford Post-Quantum Cryptography Summit

What is HuFu?

虎符 ("HuFu") is a tally in the shape of a tiger.

- an authentication mechanism in ancient China

HuFu stands for

Hash-and-Sign Signatures From Powerful Gadgets

- hash-and-sign paradigm
- gadget-based GPV instantiation
- security based on plain LWE and SIS

Hash-and-Sign Lattice Signatures

Public key: $\mathbf{P}$ is a bad representation of $\mathcal{L}$
Secret key: $\mathbf{T}$ is a good representation of $\mathcal{L}$, called trapdoor

Sign

1. Hash the message to a random vector $\mathbf{m}$
2. Find some $\mathbf{v} \in \mathcal{L}$ close to $\mathbf{m}$ (using $\mathbf{T}$)

Verify

1. Check $\mathbf{v} \in \mathcal{L}$ (using $\mathbf{P}$)
2. Check $\mathbf{v}$ is close to $\mathbf{m}$

GPV Framework

Early hash-and-sign schemes were broken due to secret leakage from signatures[1].

[1] Learning a parallelepiped: Cryptanalysis of GGH and NTRU signatures. Eurocrypt'06. Nguyen, Regev.

GPV Framework

Early hash-and-sign schemes were broken due to secret leakage from signatures[1].

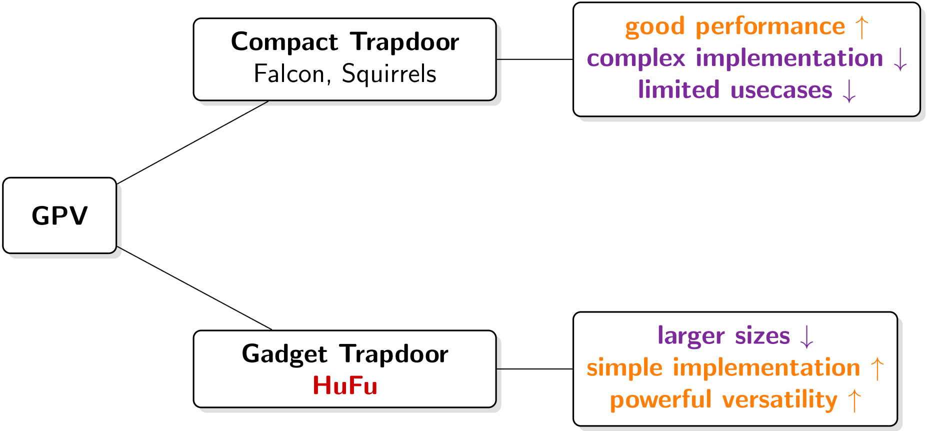
In 2008, Gentry, Peikert and Vaikuntanathan proposed a provably secure hash-and-sign framework[2].

- signatures follow some Gaussian distribution independent of $\mathbf{T}$
- zero-knowledge property $\Rightarrow$ security proof

[1] Learning a parallelepiped: Cryptanalysis of GGH and NTRU signatures. Eurocrypt'06. Nguyen, Regev.

[2] Trapdoors for Hard Lattices and New Cryptographic Constructions. STOC'08. Gentry, Peikert, Vaikuntanathan.

GPV Instantiations



Key Pair

HuFu uses the compact gadget framework[3]

Secret key: $\mathbf{S} \leftarrow \chi^{n\times m}, \mathbf{E} \leftarrow \chi^{m\times m}$ where χ is the LWE error distribution
Public key: $\hat{\mathbf{A}} \leftarrow U(\mathbb{Z}_Q^{m\times n})$ and $\mathbf{B} = p\mathbf{I} - (\hat{\mathbf{A}}\mathbf{S} + \mathbf{E}) \bmod Q$

- $\mathbf{A} = [\mathbf{I}, \hat{\mathbf{A}}, \mathbf{B}]$ can be seen as a random HNF under LWE assumption
- $\mathbf{A} \cdot \mathbf{T} = p\mathbf{I}$ where $\mathbf{T} = \begin{pmatrix} \mathbf{E} \\ \mathbf{S} \\ \mathbf{I} \end{pmatrix}$

[3] Compact Lattice Gadget and Its Applications to Hash-and-Sign Signatures. Crypto'23. Yu, Jia, Wang.

Signing

The signing procedure can be done in two phases

- offline phase: samples $\mathbf{p} \leftarrow D_{\mathbb{Z}^{n+2m}, s^2\mathbf{I} - \mathbf{T}\mathbf{T}^t}$
- online phase
  1. compute small $(\mathbf{z}, \mathbf{e})$ such that $p\mathbf{z} + \mathbf{e} = H(m) - \mathbf{A}\mathbf{p} \mod Q$
  2. return $\mathbf{s} = \mathbf{T}\mathbf{z} + \mathbf{p}$

Correctness: $\mathbf{A}\mathbf{s} + \mathbf{e} = \mathbf{A}\mathbf{T}\mathbf{z} + \mathbf{A}\mathbf{p} + \mathbf{e} = p\mathbf{z} + \mathbf{e} + \mathbf{A}\mathbf{p} = H(m)$

Security: the signing is simulatable without knowing the trapdoor

- Forgery is hard under SIS assumption

Parameters and Performance

Security level	NIST-1	NIST-3	NIST-5
Dimensions (m, n)	(736, 848)	(1024, 1232)	(1312, 1552)
Modulus Q	2^{16}	2^{17}	2^{17}
Gadget param. (p, q)	$(2^{12}, 2^4)$	$(2^{13}, 2^4)$	$(2^{13}, 2^4)$
Acceptance bound B	62521	108493	130320
Sig. size (in bytes)	2455	3540	4520
PK size (in kilobytes)	1059	2177	3573
Key recovery (C/Q)	129/117	194/176	256/233
Forgery (C/Q)	128/116	192/175	258/234

- key size is fairly large, but signature size is comparable to Dilithium

Parameters and Performance

	NIST-I	NIST-III	NIST-V
Optimized implementation			
KeyGen	1,269,041	5,989,281	9,986,598
Sign (online)	942	1,458	3,891
Sign (offline)	8,919	14,811	37,060
Sign (total)	9861	16,269	40,951
Verify	1692	6515	11,310
AVX2 implementation			
KeyGen	819,865	2,962,178	5,930,716
Sign (online)	380	707	998
Sign (offline)	3,384	6809	10,873
Sign (total)	3,764	7,516	11,871
Verify	900	2,366	3,801

Table: Performance (in kilocycles) on a single core of Intel Core i9-12900K @ 3.20 GHz.

Attacks on HuFu Signature Encodings

Saarinen reported two security flaws of HuFu.

Bit-flipping Attack

By flipping some bits in HuFu signatures, an adversary can generate a new valid signature for the same message.

Length Modification Attack

An adversary can modify the length field in HuFu signatures to trigger buffer overflows.

Counetermeasures

Two attacks exploits the fact that there can be multiple encodings for the same signature.

Countermeasure against the bit-flipping attack

- Fix the encoder's initial state, and check if the decoder's final state matches that number
- Perform sanity check for decoder's initial state

Countermeasure against the length modification attack

- Remove the length field, pad the signature to a fixed length
- Resembles ISO/IEC 7816-4, but padding at the front of buffer to make it compatible with ANS

Both countermeasures come with very minor efficiency loss!

Future Investigations

Some recent techniques can improve the overall size of HuFu

- new gadget construction
- trapdoor generation

BUFF transformation[4] gives additional security properties to signatures

- transformation is direct, but overhead is great due to large key size
- Can we design a lightweight BUFF?

Current parameters fully avoid the small-modulus SIS attack[5]

- if taking a relaxed ℓ_2-norm condition while adding ℓ_∞-norm condition, we can reduce the overall size
- How far can we go?

---

[4] Buffing signature schemes beyond unforgeability and the case of postquantum signatures. S&P 2021, Cremers, Düzlü, Fiedler, Fischlin, Janson

[5] Finding short integer solutions when the modulus is small. Crypto'23. Ducas, Espitau, Postlethwaite

Thank you!