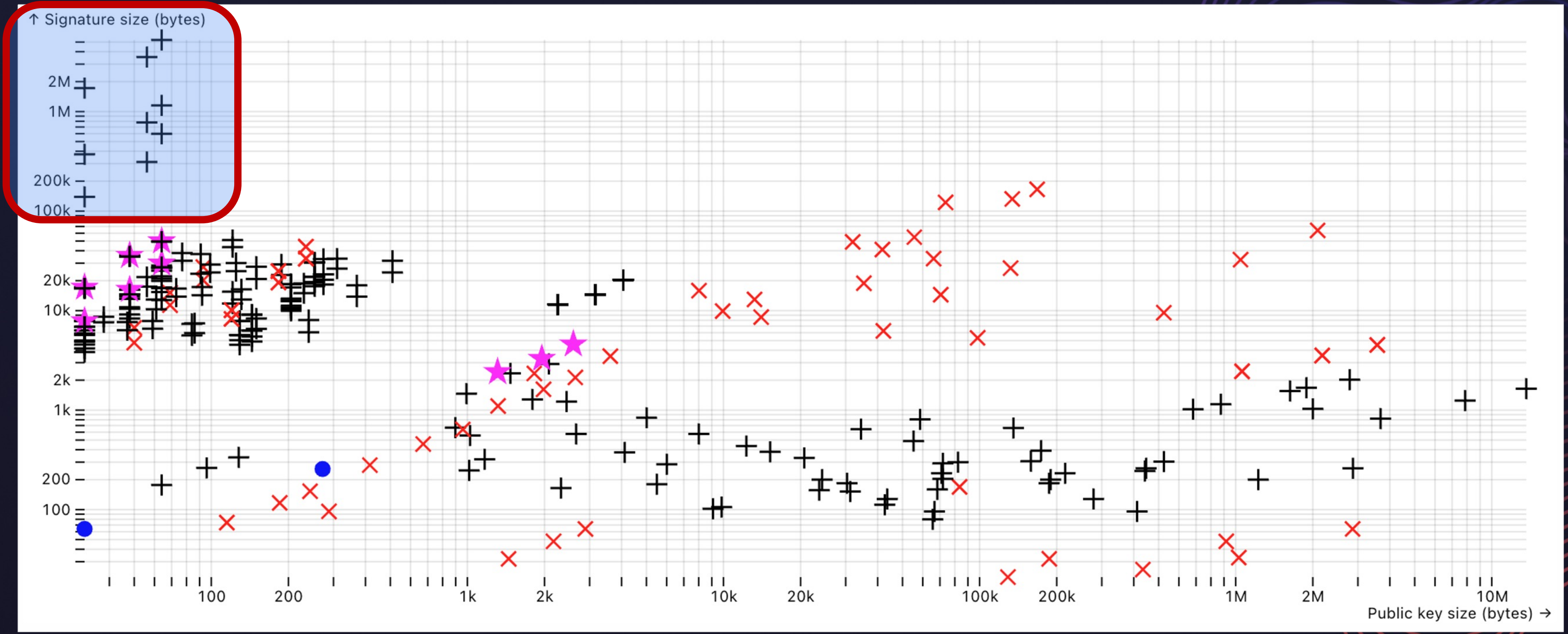
preon

Digital Signature from zk-SNARK

Wei-Chih Hong @BTQ

With Ming-Shing Chen, Yu-Shian Chen, Chen-Mou Cheng, Shiuan Fu, Jen-Hsuan Hsiang, Jimmy Hu, Po-Chun Kuo, Wei-Bin Lee, Feng-Hao Liu and Justin Thaler

Why “preon”?



https://pqshield.github.io/nist-sigs-zoo/wide.html

preon: beyond digital signature

- Standard digital signature on message x :
  - "I swear that the signer said x."

- Beyond digital signature:
  - "The signer said x, which I'm not going to disclose here, but I swear that $f(x) = y$."
  - Selective reveal
    - E.g. prove "I'm a citizen and more than 18 years old." without revealing further information in one's digital ID

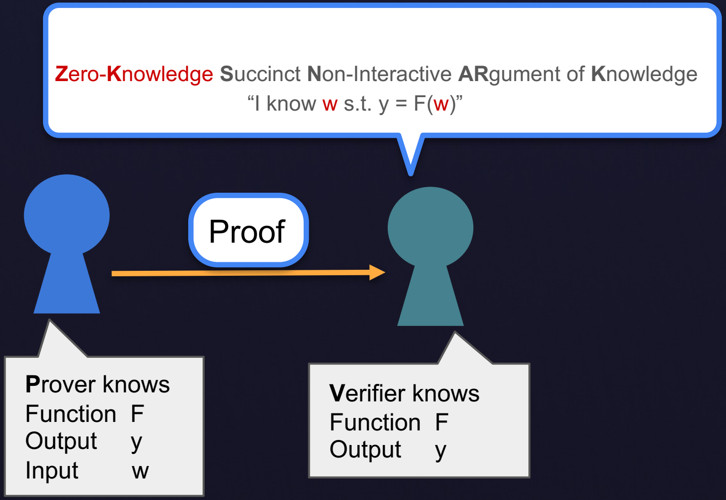
preon: overview

Provides the flexibility to prove any properties of msg

Provides the security to our signature

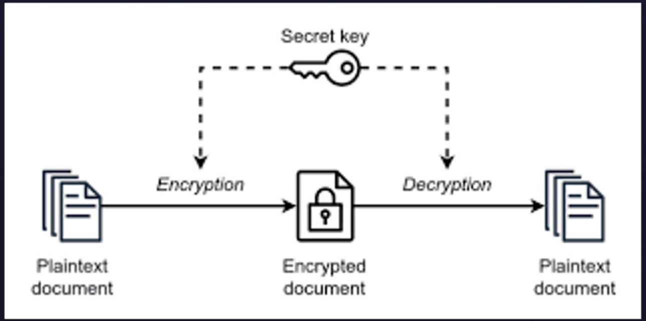
zk-SNARK + OWF => Signature scheme

Aurora



AES

The most widely used symmetric cipher standard in the world



preon: under the hood

- preon ≈ Aurora + AES
  - Aurora: post-quantum zk-SNARK
  - AES as one-way function
    - Public key is a pair of ciphertext and plaintext encrypted under the private key
- Optimization: replace prime field with binary field
  - greatly reduced number of constraints
  - faster arithmetic, additive FFT
  - about 20% smaller signature

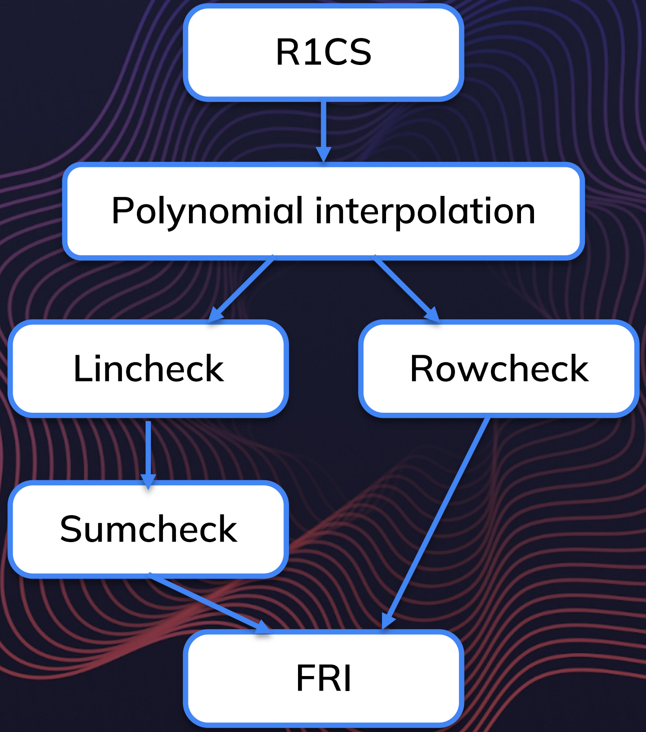
zk-SNARK

Zero-Knowledge Succinct Non-Interactive ARgument of Knowledge

- Prover P computes a proof π which convinces verifier V that P knows a ω such that $y = f(x, \omega)$.
- π is "small" compared with f, e.g. $|\pi| = O(\log|f|)$

Aurora

- Encode $y = f(x, \omega)$ into R1CS: $\mathbf{Az} \circ \mathbf{Bz} = \mathbf{Cz}$, where
  - $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ are matrices depending on f
  - $\mathbf{z} = [1, \mathbf{v}, \mathbf{w}]^{\mathrm{T}}$
  - Lincheck RS-encoded IOP: $\mathbf{a} = \mathbf{Az}$, $\mathbf{b} = \mathbf{Bz}$, $\mathbf{c} = \mathbf{Cz}$
  - Rowcheck RS-encoded IOP: $\mathbf{a} \circ \mathbf{b} = \mathbf{c}$
  - FRI low degree test on a single random linear combination of the committed polynomials in lincheck and rowcheck
  - BCS transform to turn a public-coin IOP into a NIROP



RS-encoded IOP

- Think of $z \in V$ as a function $H \to F_q$ for $|H| = \dim V$
- Encode z as f_z via Lagrange interpolation
  - $f_z(X) = r(X) + s(X) \prod_{h \in H} (X - h)$, such that $\forall h \in H, r(h) = f_z(h)$
  - $s(X)$ is randomly sampled masking polynomial with bounded degree
- Verifier can now query $f_z(x)$ for some $x \in L$
  - Intuition: ZK if $H \cap L = \varnothing$ and $\deg s$ is large enough
- Run FRI to check if $\deg f_z$ is small enough (low degree test)
- Example: for Preon128A, $|H| = 2^{12}$, $|L| = 2^{19}$, FRI test for degree lower than 2^{14}

Part 1

$P(\mathbf{w}, \mathbf{v}, \mathbf{A}, \mathbf{B}, \mathbf{C})$ $\qquad$ $V(\mathbf{v}, \mathbf{A}, \mathbf{B}, \mathbf{C})$

compute $f_{\mathbf{w}}, f_{\mathbf{Az}}, f_{\mathbf{Bz}}, f_{\mathbf{Cz}}$

$$\xrightarrow{\hat{f}_{\mathbf{w}}, \hat{f}_{\mathbf{Az}}, \hat{f}_{\mathbf{Bz}}, \hat{f}_{\mathbf{Cz}}}$$

Part 2, for indices $i \in \{1, ..., \lambda_i\}$, do the following independently

P $\qquad$ V

sample $\alpha_i, s_{i,1}, s_{i,2}, s_{i,3} \overset{\$}{\leftarrow} \mathbb{F}$

$$\xleftarrow{\alpha_i, s_{i,1}, s_{i,2}, s_{i,3}}$$

compute g_i, h_i
from formula (10)

$$\xrightarrow{\hat{h}_i}$$

Part 3, for indices $j \in \{1, \ldots, \lambda_i'\}$, do the following independently

P V

sample $\mathbf{y}_j \xleftarrow{\$} \mathbb{F}^{5+3\lambda_i}$

$\xleftarrow{\mathbf{y}_j}$

P and V run an FRI protocol for
$\deg(f_{j,0}) < 2\max\{m, n+1\}$
where $f_{j,0}$ is defined by formula (11)

$\longrightarrow$

$$f_{j,0} := \mathbf{y}_{j,1} \cdot f_{\mathbf{w}} + \mathbf{y}_{j,2} \cdot f_{\mathbf{Az}} + \mathbf{y}_{j,3} \cdot f_{\mathbf{Bz}} + \mathbf{y}_{j,4} \cdot f_{\mathbf{Cz}} + \mathbf{y}_{j,5} \cdot \frac{f_{\mathbf{Az}} \cdot f_{\mathbf{Bz}} - f_{\mathbf{Cz}}}{Z_{H_1}}$$

$$+ \sum_{i=1}^{\lambda_i} (\mathbf{y}_{j,5+i} \cdot h_i) + \sum_{i=1}^{\lambda_i} (\mathbf{y}_{j,5+\lambda_i+i} \cdot g_i)$$

$$+ \sum_{i=1}^{\lambda_i} (\mathbf{y}_{j,5+2\lambda_i+i} \cdot X^{(2\max\{m,n+1\})-(\max\{m,n+1\}-1)} \cdot g_i)$$

Expressiveness of R1CS

$y = x^3$	$x \cdot x = y$ $u \cdot x = y$
$0 \leq x < 8$	$1 \cdot (x_0 + 2x_1 + 4x_2) = x$ $x_0 \cdot x_0 = x_0$ $x_1 \cdot x_1 = x_1$ $x_2 \cdot x_2 = x_2$
$r = \text{if } b \text{ then } t \text{ else } f$	$(t - f) \cdot b = r - f$ $b \cdot b = b$

BTQ

R1CS constraints for AES

- Every byte variable in AES is a field element in $\mathbb{F}_2[x]/(x^8+x^4+x^3+x+1)$
- Adopting binary field in Aurora makes expressing AES encryption as R1CS constraints extremely efficient.
  - E.g. the inversion of a field element in the SubBytes step
  - $y = b^{-1} \bmod (x^8+x^4+x^3+x+1) \Leftrightarrow b \times y = 1 + h \times (x^8+x^4+x^3+x+1)$ with additional range check constraints on the bits of y and h
- Total number of constraints for AES-128: 14240[†] ⟶ 3656
  - Greatly reduces the size of matrices **A**, **B**, **C**

[†] https://github.com/akosba/xjsnark

Parameter selection

- Three sets of parameters, namely aggressive (A), balanced (B), and conservative (C), are selected for security level 1, 3, and 5.
- Only the aggressive and balanced parameter sets are recommended

Parameter set	$\lvert\mathbb{F}\rvert$	t	$\lvert\mathbb{L}\rvert$	b	ℓ	λ
Preon128A	2^{192}	2^{12}	2^{19}	1040	26	256
Preon128B	2^{192}	2^{12}	2^{19}	2320	58	384
Preon192A	2^{256}	2^{13}	2^{20}	1638	39	384
Preon192B	2^{256}	2^{13}	2^{20}	3654	87	512
Preon256A	2^{320}	2^{14}	2^{20}	2184	52	512
Preon256B	2^{320}	2^{14}	2^{20}	4956	118	512

preon: performance

Security	Size (Bytes)			Timing		
	Private key	Public key	Signature	Keygen	Sign	Verify
128-bit(A)	16	32	139K	2us	7s	209ms
128-bit(B)	16	32	372K	2us	7.3s	224ms
192-bit(A)	24	56	312K	2us	24.3s	1,867ms
192-bit(B)	24	56	778K	2us	25.6s	1,847ms
256-bit(A)	32	64	598K	2us	80.6s	8,724ms
256-bit(B)	32	64	1,157K	2us	80.2s	8,528ms

Summary of current results

- Can be easily extended to support advanced functions with R1CS
- Preliminary security analysis shows even Aggressive (A) meets NIST's requirement, with Balanced (B) as backup
- Small public/private keys but big signatures
- Current implementation is slow

Future directions

- In-depth security analysis and proofs
  - Working with experts on zk-SNARK
- Further optimized implementations
  - In progress on optimized CPU implementation
  - FPGA implementation
  - Open: GPU implementation
- Preon+ (Cantor basis + efficient R1CS encode)

Security	Private key	Public key	Signature	Keygen	Sign	Verify
Preon128-bit(A)	16	32	139KB	2us	7s	209ms
Preon+128-bit(A)	16	32	125 -> 90KB	2us	**628 ms**	**197 ms**

The preon team

Yu-Shian Chen
Hon Hai Research Institute

Wei-Bin Lee
Hon Hai Research Institute

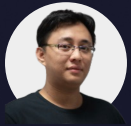
Chen Mou Cheng
BTQ

Shiuan Fu
BTQ

Po-Chun Kuo
BTQ

Wei-Chih Hong
BTQ

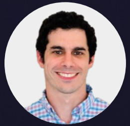
Jen-Hsuan Hsiang
BTQ



Sheng-Te Hu
BTQ

Ming-Shing Chen
Academia Sinica

Feng-Hao Liu
Washington State University



Justin Thaler
Georgetown University