Dubhe: Succinct Zero-Knowledge Proofs for Standard AES and Related Applications

Changchang Ding and Yan Huang

USENIX Security 2023

INDIANA UNIVERSITY

Zero-Knowledge Proofs of Knowledge

Allow a prover $\mathcal{P}$ to convince a verifier $\mathcal{V}$ that $\mathcal{P}$ holds secret witnesses w

s.t. $C(x, w) = 1$, without revealing w

Succinct: proof size / verifier time sublinear in
- $|C|$ (Weak Succinct)
- $|C| + |w|$ (Strong Succinct)

Transparent: no trusted setup

State-of-the-art Transparent ZKPs

MPC-in-the-Head based protocols:

- KKW [KKW18], Limbo [dOT21], etc.
- Pros: Support arbitrary fields.
- Cons: Not succinct.

Virgo [ZXZS19] / Virgo++ [ZLWZ+21]

- Using GKR [GKR08] and LDT
- Pros: Strong succinctness
- Cons: Constraints on choices of fields

Question:

Is it possible to construct a **concretely-efficient succinct ZKPoK** that can easily support computations on **arbitrary fields?**

Dubhe: Summary of Contributions

1. Succinct proof in the number of gates.
2. No restriction on the underlying fields.
3. Applications:
   - Identification / digital signature schemes
   - Ring identification / signature schemes

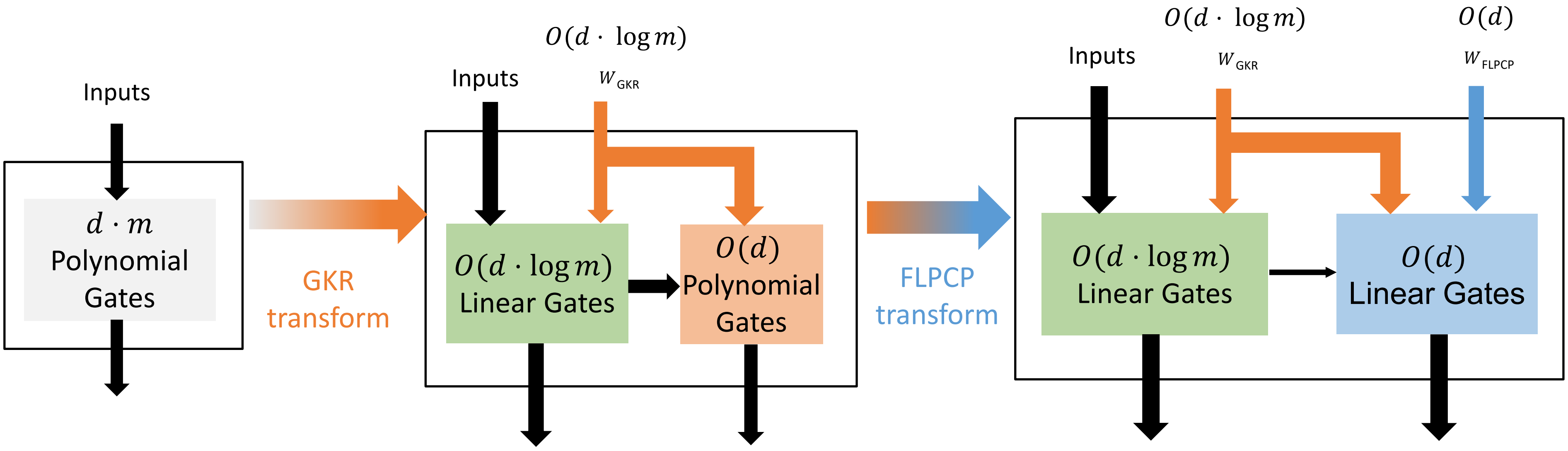
All schemes based on unmodified use of AES

Observations and Dubhe's Goals

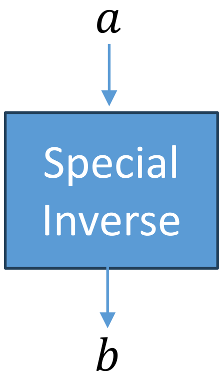
😣 ☹ ☺ 🙂
bad → good

	ZK	Fast Verifier	Short Proof	Non-linear gates	Linear gates
KKW	🙂	☹	😣	😣	🙂
GKR	😣	🙂	🙂	☹	
FLPCP	☺	☹	☺	☺	N/A
Dubhe	🙂	🙂	🙂	☺	🙂

Dubhe's Approach



Proof of AES



The only non-linear operation: special inverse in SubBytes

$$\begin{cases} b = a^{-1}, & a \neq 0 \\ b = 0, & a = 0 \end{cases}$$

Banquet [BdKO+21] / Limbo's approach: required **non-zero** inputs to all SubBytes

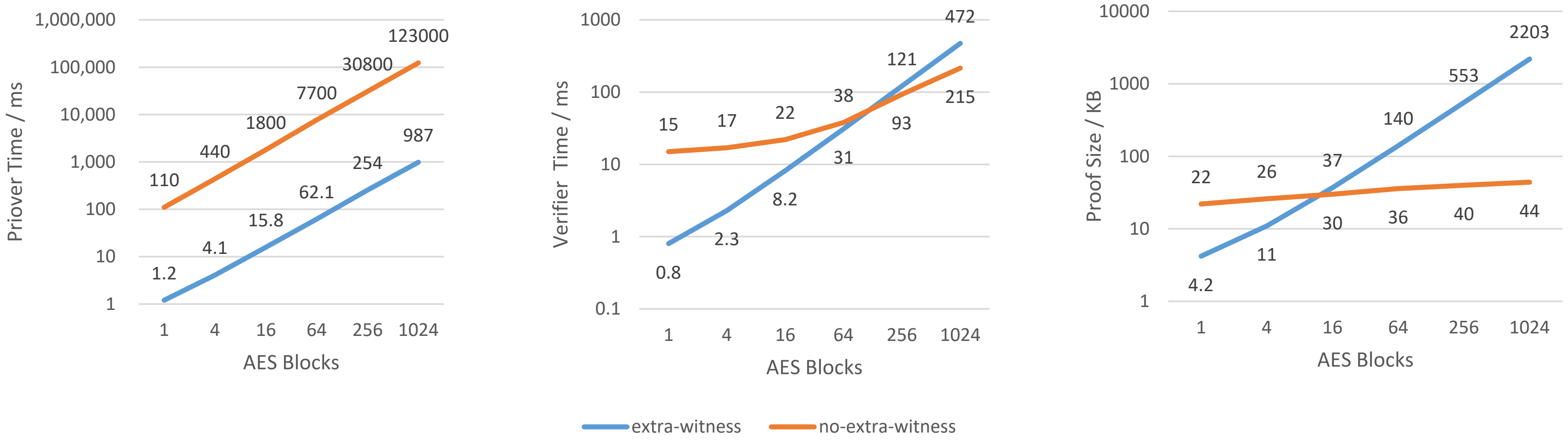
With extra witness:

$$a \cdot b = 1 \lor (a = 0 \land b = 0) \Leftrightarrow a(ab + 1) = 0 \land b(ba + 1) = 0$$

Without extra witness:

- Treat SubBytes as 8 table-lookups, each has 256 entries.
- Encode each table as an 8-variate polynomial.

Counter-Mode AES



AES based Identification / Signature

Identification (interactive):

- Keygen: $pk \leftarrow AES_{sk}(ID_u)$
- Proof of Identity: $\mathrm{w} = sk,\ x = (ID_u, pk)$
- Circuit: AES with extra witness

Signature (non-interactive through Fiat-Shamir Transform):

- Keygen: $pk \leftarrow AES_{sk}(ID_u)$
- Signature: a ZKP of $\mathrm{w} = sk,\ x = (ID_u, pk)$ with $H(m)$ as randoms.
- Circuit: AES with extra witness, skip GKR to reduce number of rounds.

AES based Identification (100-bit stat. sec.)

Identification	P time (ms)	V time (ms)	Comm. (KB)
QuickSilver [YSWW21]	334	334	1644
Virgo++	751	36	132
Virgo	2265	21.4	174
Limbo (n=16)	2.7	2.5	10
Dubhe (n=16)	**2.8**	**2.0**	**9.2**
Limbo (n=256)	12	11	5.8
Dubhe (n=256)	**6.6**	**6.0**	**6.1**

AES based Signature (~128-bit comp. sec.)

Sec.	Signature	S time (ms)	V time (ms)	Sign. size (KB)
101	Virgo	2265	21	174
103	Virgo++ (243 layers)	49	55	775
101	Virgo++ (9 layers)	409	32	129
127	Limbo	3.6	2.5	21
128	**Dubhe**	**4.8**	**4.0**	**30**
133	SPHINCS+-128 (smaller)	164	0.4	29
128	SPHINCS+-128 (faster)	17	0.7	49

Ring Identification / Signature

Ring Identification: Prove in ZK one's identity belongs to a predefined group.

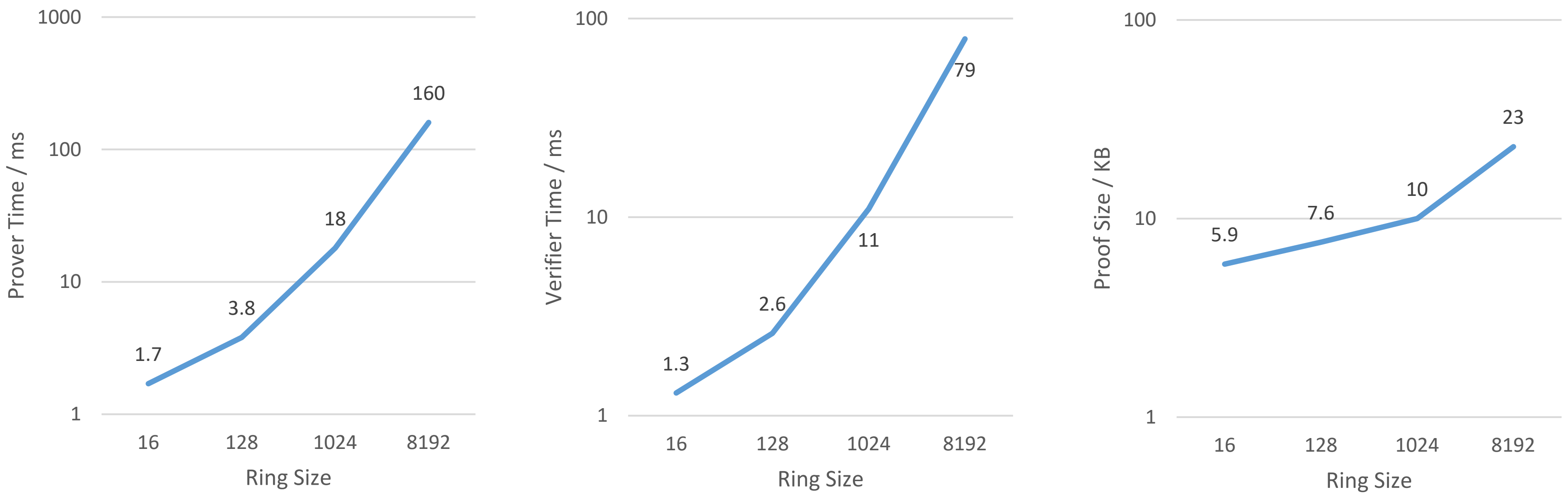
Ring Signature: Sign messages on behalf of a group without revealing the signer's identity

$$AES_{sk}(0) = ID \quad \wedge \quad ID \in \{ID_i, i \in [m]\}$$

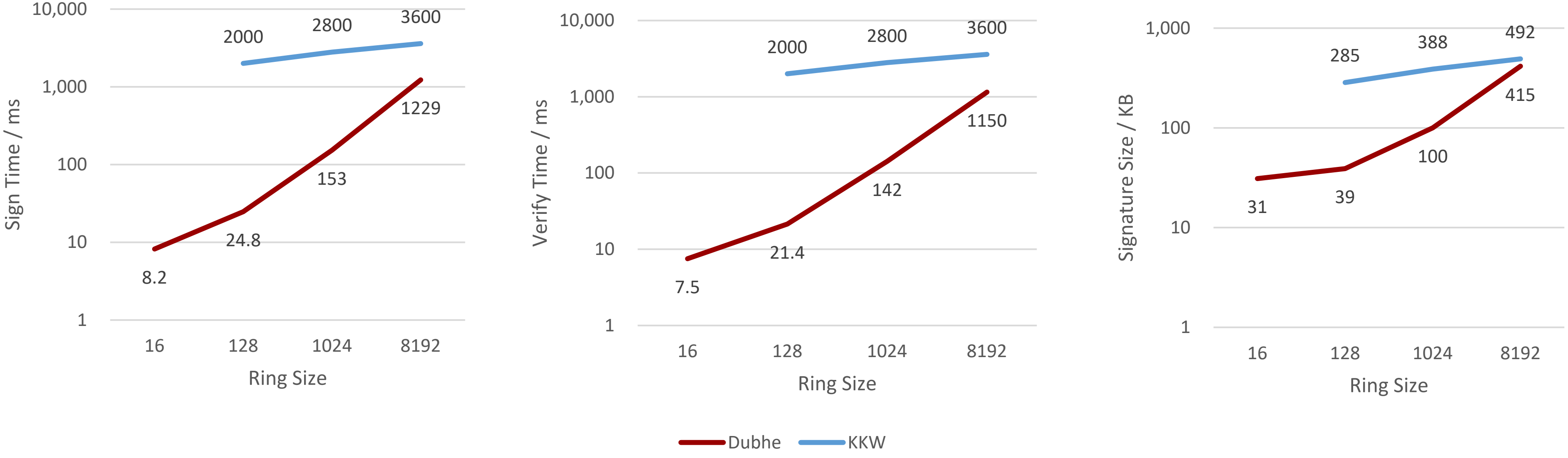
Proof of membership: multiplication tree.

$$ID \in \{ID_i, i \in [m]\} \Leftrightarrow \prod (ID - ID_i) = 0$$

Ring Identification



Ring Signature



Thank you!

We invite you to read our paper for details

and play with our implementation at

https://github.com/zkPrfs/dubhe