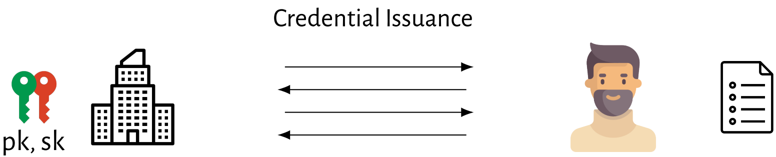
# Lattice-Based Accumulator and Application to Anonymous Credential Revocation

Victor Youdom Kemmoe    Anna Lysyanskaya    Ngoc Khanh Nguyen

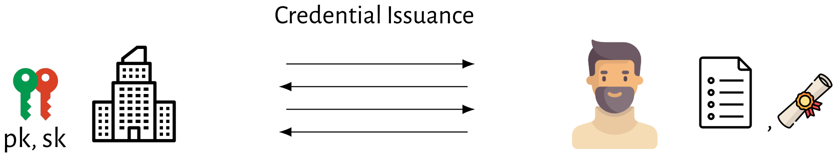
# Motivation

## Anonymous Credentials [CL02 a; BBC+24 ]



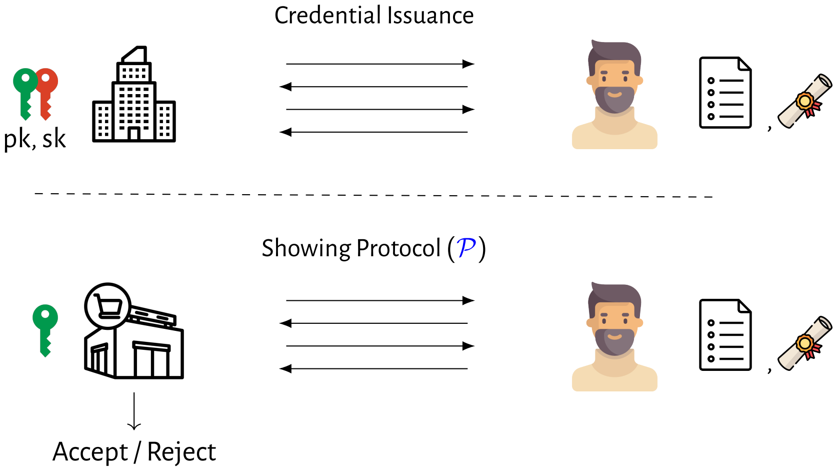
# Motivation

## Anonymous Credentials [CL02 a; BBC+24 ]



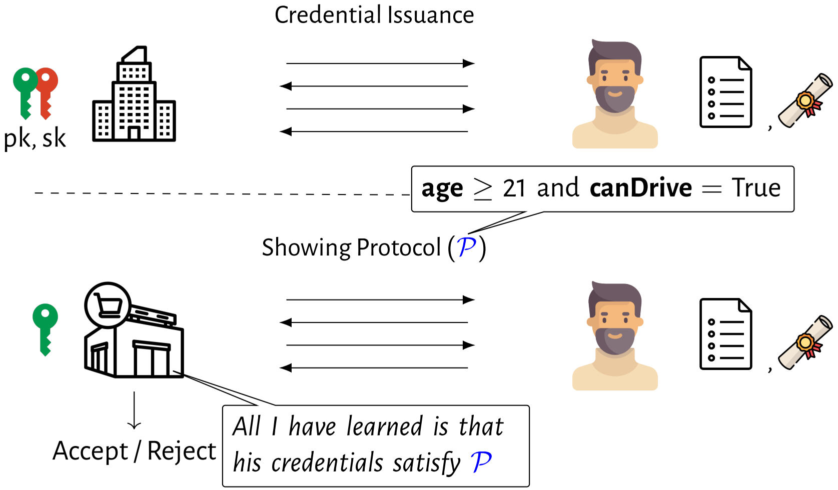
# Motivation

## Anonymous Credentials [CL02 a; BBC+24 ]



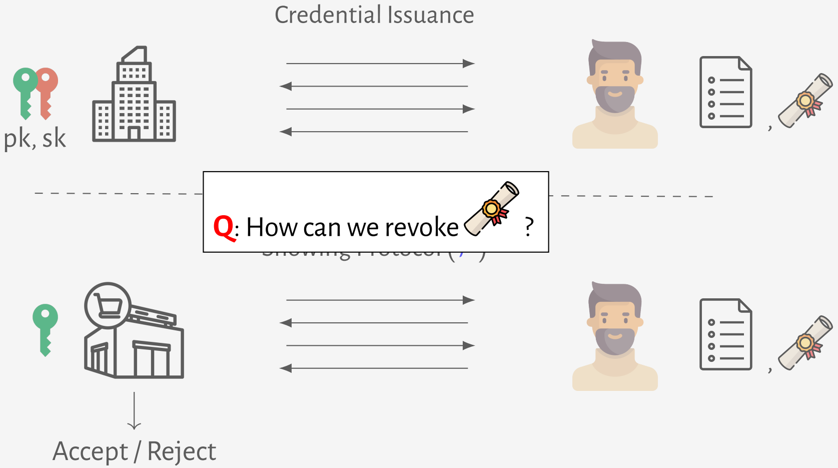
# Motivation

## Anonymous Credentials [CL02 a; BBC+24 ]



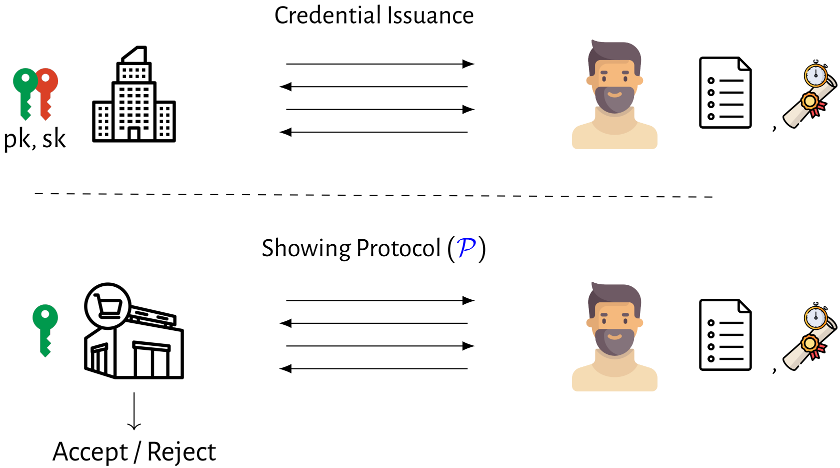
# Motivation

## Anonymous Credentials [CL02 a; BBC+24 ]



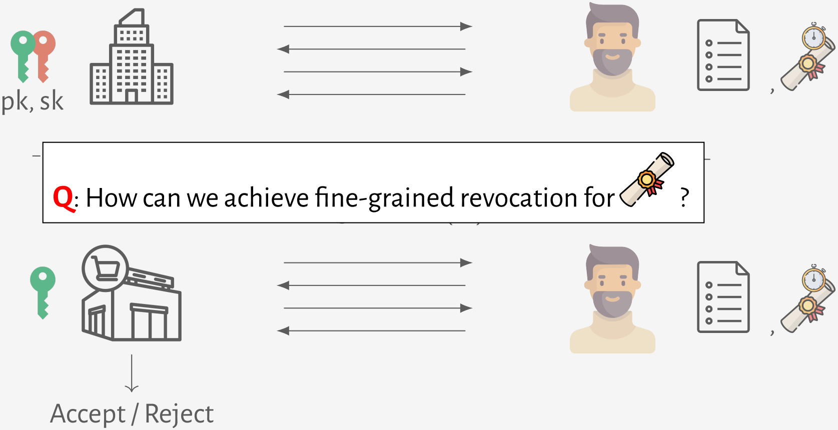
# Motivation

## Anonymous Credentials [CL02 a; BBC+24 ]



# Motivation

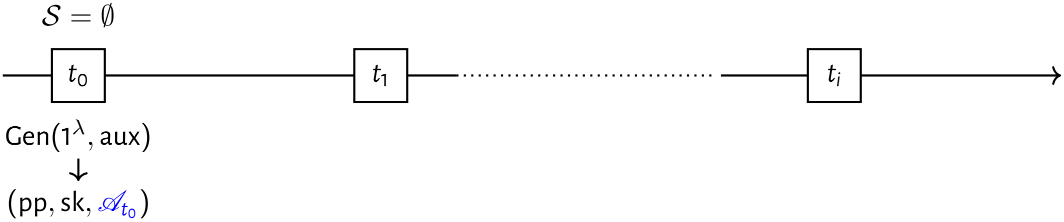
## Anonymous Credentials [CL02 a; BBC+24]




**Q**: How can we achieve fine-grained revocation for ?

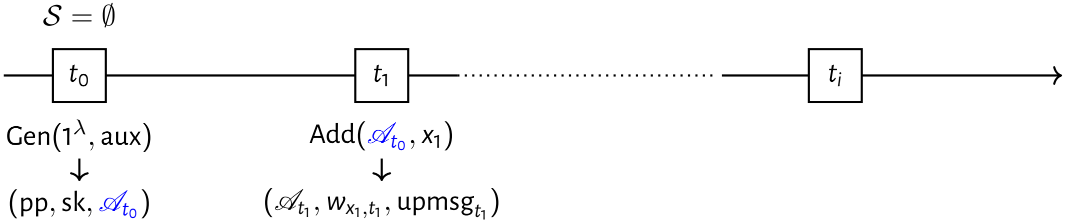
# Positive Dynamic Accumulator

## Syntax [BCD+17 ; DHS15 ; KL24 ]



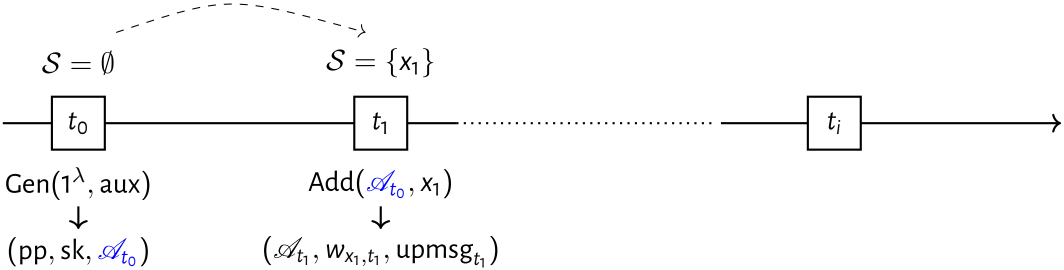
# Positive Dynamic Accumulator

## Syntax [BCD+17 ; DHS15 ; KL24 ]



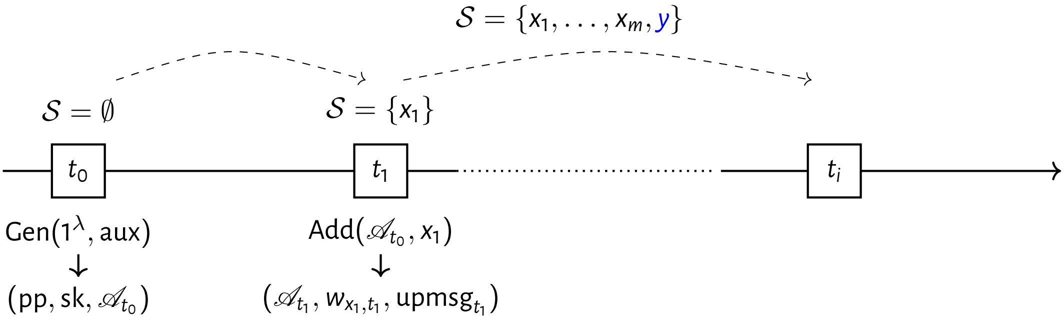
# Positive Dynamic Accumulator

## Syntax [BCD+17 ; DHS15 ; KL24 ]



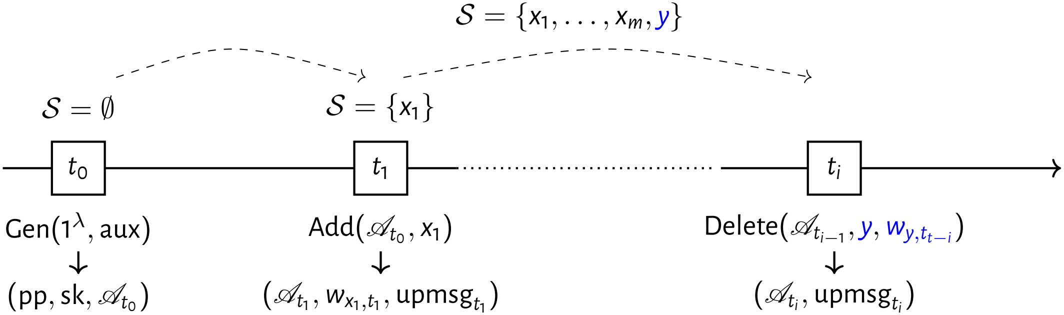
# Positive Dynamic Accumulator

## Syntax [BCD+17 ; DHS15 ; KL24 ]



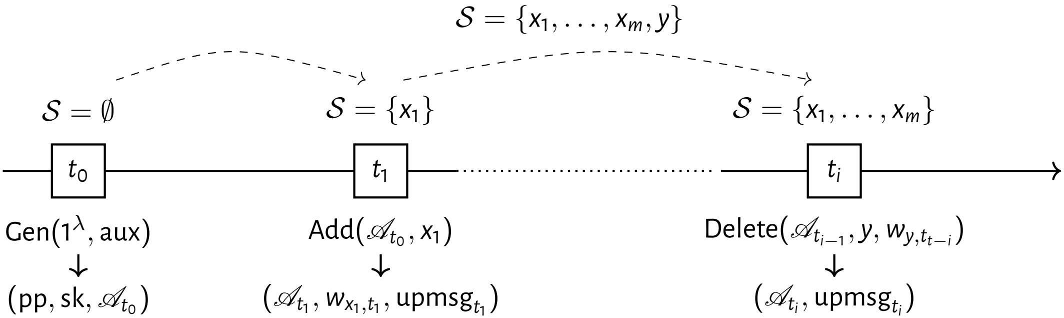
# Positive Dynamic Accumulator

## Syntax [BCD+17 ; DHS15 ; KL24 ]



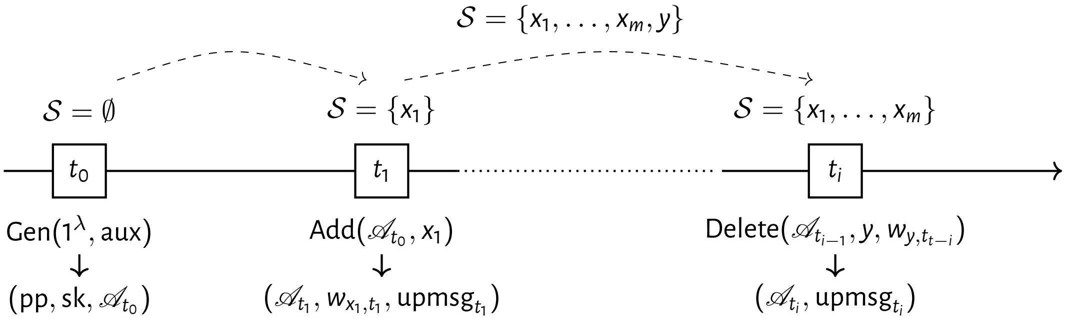
# Positive Dynamic Accumulator

## Syntax [BCD+17 ; DHS15 ; KL24 ]



# Positive Dynamic Accumulator

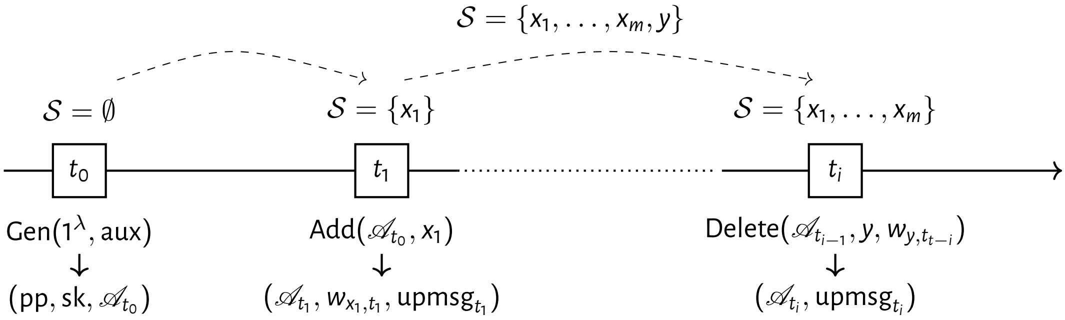
## Syntax [BCD+17 ; DHS15 ; KL24 ]




- $\mathsf{MemWitUp}(x, w_{x,t}, \mathsf{upmsg}_{t+1}) \rightarrow w_{x,t+1}$
- $\mathsf{MemVerify}(\mathscr{A}_t, x, w_{x,t}) \rightarrow \mathsf{Accept}/\mathsf{Reject}$

# Positive Dynamic Accumulator

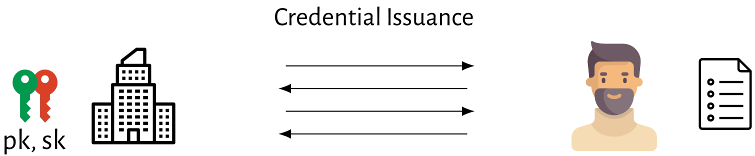
**Syntax [BCD+17 ; DHS15 ; KL24 ]**



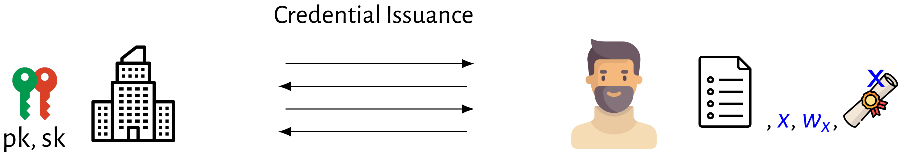

- $\mathsf{MemWitUp}(x, w_{x,t}, \mathsf{upmsg}_{t+1}$
- $\mathsf{MemVerify}(\mathscr{A}_t, x, w_{x,t}) \to$ Ac

- **Compactness**: $|\mathscr{A}| = \mathsf{poly}(\lambda)$, $|w_{x,t}| = \mathsf{poly}(\lambda, |x|)$
- **Security**: *Hard* to produce a $w_x$ for $x \notin \mathcal{S}$
- **Communication efficiency**: $|\mathsf{upmsg}| = O(\#\mathsf{Del})$

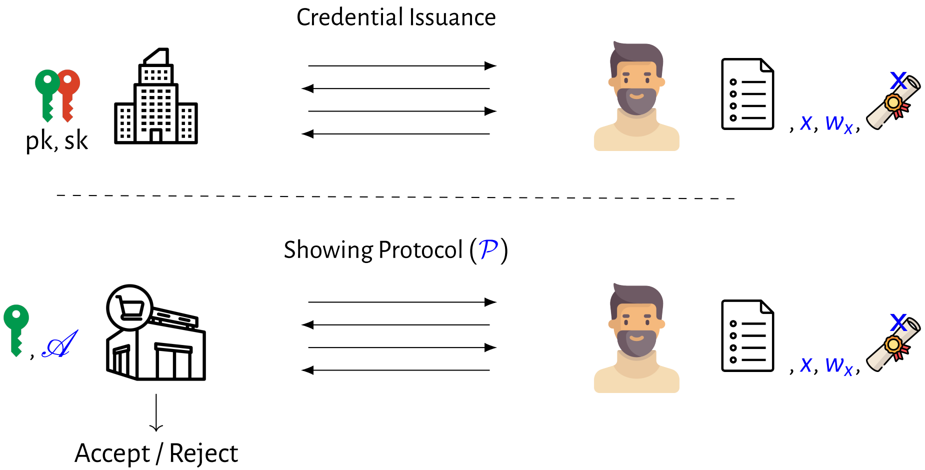
# Positive Dynamic Accumulator in ACs revocation



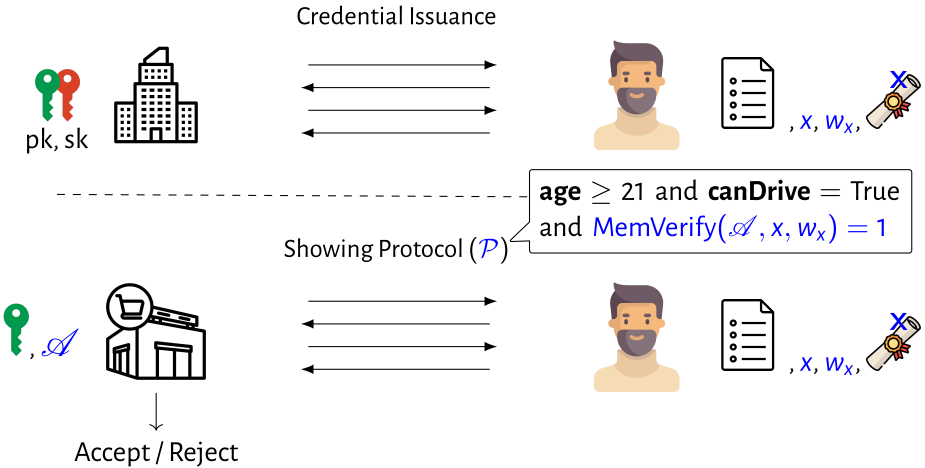
# Positive Dynamic Accumulator in ACs revocation



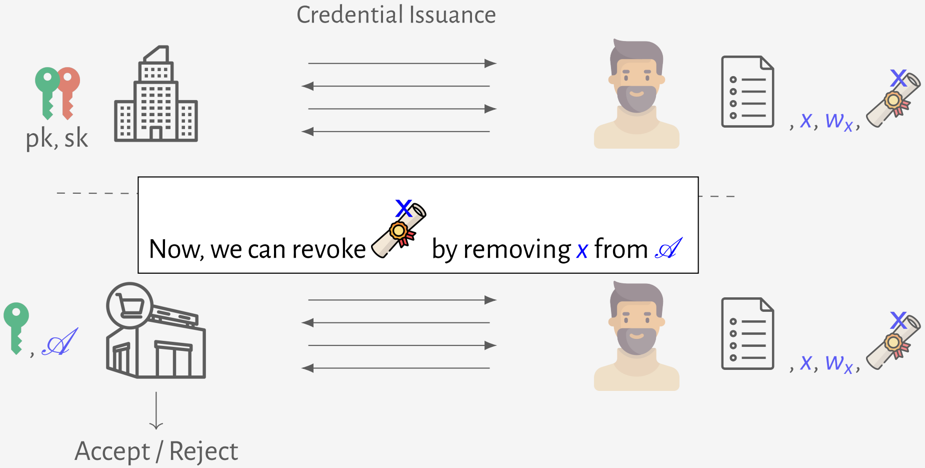
# Positive Dynamic Accumulator in ACs revocation



# Positive Dynamic Accumulator in ACs revocation



# Positive Dynamic Accumulator in ACs revocation



# Prior works on Positive Dynamic Accumulators

| Scheme | Assumption | $\|w\|$ | $\|\mathsf{upmsg}\|_{\mathsf{Add}}$ | $\|\mathsf{upmsg}\|_{\mathsf{Del}}$ | $\|\mathsf{pp}\|$ |
|---|---|---|---|---|---|
| [CL02 b; LLX07 ; KL24 ] | Strong RSA | $\ell \cdot \mathsf{poly}(\lambda)$ | $\ell^*$ | $\ell$ | $\mathsf{poly}(\lambda)$ |
| [BCD+17 ; KL24 ] | Strong RSA | $\ell \cdot \mathsf{poly}(\lambda)$ | — | $\ell$ | $\mathsf{poly}(\lambda)$ |
| [Ngu05 ; ATS+09 ; CKS09 ] | $q$-Strong DH | $\mathsf{poly}(\lambda)$ | $\ell^*$ | $\ell$ | $s \cdot \mathsf{poly}(\lambda)$ |
| [KB21 ; JML24 ] | $q$-Strong DH | $\mathsf{poly}(\lambda)$ | — | $\ell$ | $\mathsf{poly}(\lambda)$ |
| [PST+13 ; YAY+18 ; LLN+23 ] | M-SIS | $\mathsf{poly}(\lambda) \cdot \log s$ | $\mathsf{poly}(\lambda) \cdot \log s^*$ | $\mathsf{poly}(\lambda) \cdot \log s$ | $\mathsf{poly}(\lambda)$ |
| [ZYH24 ] | M-SIS | $\mathsf{poly}(\lambda)$ | $\mathsf{poly}(\lambda)^*$ | $\mathsf{poly}(\lambda)$ | $\mathsf{poly}(\lambda) \cdot s \log s$ |
| [CP23 ] | M-SIS | $\ell \cdot \mathsf{poly}(\lambda)$ | $\ell^*$ | $\ell$ | $\ell \cdot \mathsf{poly}(\lambda)$ |
| [CP23 ]+ [WW23 ] | $\ell$-Succinct M-SIS | $\mathsf{poly}(\lambda)$ | $\ell^*$ | $\ell$ | $\ell^2 \cdot \mathsf{poly}(\lambda)$ |
| **Our work** | M-SIS | $\ell \cdot \mathsf{poly}(\lambda)$ | — | $\ell$ | $\ell \cdot \mathsf{poly}(\lambda)$ |
| **Our work** | $\ell$-Succinct M-SIS | $\mathsf{poly}(\lambda)$ | — | $\ell$ | $\ell^2 \cdot \mathsf{poly}(\lambda)$ |

- $\ell$: Input's bit length
- $*$: $\|\mathsf{upmsg}\| = 0$ for a fix set in pre-processing
- $s$: Size of the set

# Prior works on Positive Dynamic Accumulators

| Scheme | Assumption | $\lvert w\rvert$ | $\lvert \mathsf{upmsg}\rvert_{\mathsf{Add}}$ | $\lvert \mathsf{upmsg}\rvert_{\mathsf{Del}}$ | $\lvert \mathsf{pp}\rvert$ |
|---|---|---|---|---|---|
| [CL02 b; LLX07 ; KL24 ] | Strong RSA | $\ell \cdot \mathsf{poly}(\lambda)$ | $\ell^*$ | $\ell$ | $\mathsf{poly}(\lambda)$ |
| [BCD+17 ; KL24 ] | Strong RSA | $\ell \cdot \mathsf{poly}(\lambda)$ | — | $\ell$ | $\mathsf{poly}(\lambda)$ |
| [Ngu05 ; ATS+09 ; CKS09 ] | $q$-Strong DH | $\mathsf{poly}(\lambda)$ | $\ell^*$ | $\ell$ | $s \cdot \mathsf{poly}(\lambda)$ |
| [KB21 ; JML24 ] | $q$-Strong DH | $\mathsf{poly}(\lambda)$ | — | $\ell$ | $\mathsf{poly}(\lambda)$ |
| [PST+13 ; YAY+18 ; LLN+23 ] | M-SIS | $\mathsf{poly}(\lambda) \cdot \log s$ | $\mathsf{poly}(\lambda) \cdot \log s^*$ | $\mathsf{poly}(\lambda) \cdot \log s$ | $\mathsf{poly}(\lambda)$ |
| [ZYH24 ] | M-SIS | $\mathsf{poly}(\lambda)$ | $\mathsf{poly}(\lambda)^*$ | $\mathsf{poly}(\lambda)$ | $\mathsf{poly}(\lambda) \cdot s \log s$ |
| [CP23 ] | M-SIS | $\ell \cdot \mathsf{poly}(\lambda)$ | $\ell^*$ | $\ell$ | $\ell \cdot \mathsf{poly}(\lambda)$ |
| [CP23 ]+ [WW23 ] | $\ell$-Succinct M-SIS | $\mathsf{poly}(\lambda)$ | $\ell^*$ | $\ell$ | $\ell^2 \cdot \mathsf{poly}(\lambda)$ |
| **Our work** | M-SIS | $\ell \cdot \mathsf{poly}(\lambda)$ | — | $\ell$ | $\ell \cdot \mathsf{poly}(\lambda)$ |
| **Our work** | $\ell$-Succinct M-SIS | $\mathsf{poly}(\lambda)$ | — | $\ell$ | $\ell^2 \cdot \mathsf{poly}(\lambda)$ |

- $\ell$: Input's bit length
- $*$: $\lvert \mathsf{upmsg}\rvert = 0$ for a fix set in pre-processing
- $s$: Size of the set

# Prior works on Positive Dynamic Accumulators

| Scheme | Assumption | $\|w\|$ | $\|\text{upmsg}\|_{\text{Add}}$ | $\|\text{upmsg}\|_{\text{Del}}$ | $\|\text{pp}\|$ |
|---|---|---|---|---|---|
| [CL02 b; LLX07 ; KL24 ] | Strong RSA | $\ell \cdot \text{poly}(\lambda)$ | $\ell^*$ | $\ell$ | $\text{poly}(\lambda)$ |
| [BCD+17 ; KL24 ] | Strong RSA | $\ell \cdot \text{poly}(\lambda)$ | – | $\ell$ | $\text{poly}(\lambda)$ |
| [Ngu05 ; ATS+09 ; CKS09 ] | $q$-Strong DH | $\text{poly}(\lambda)$ | $\ell^*$ | $\ell$ | $s \cdot \text{poly}(\lambda)$ |
| [KB21 ; JML24 ] | $q$-Strong DH | $\text{poly}(\lambda)$ | – | $\ell$ | $\text{poly}(\lambda)$ |
| [PST+13 ; YAY+18 ; LLN+23 ] | M-SIS | $\text{poly}(\lambda) \cdot \log s$ | $\text{poly}(\lambda) \cdot \log s^*$ | $\text{poly}(\lambda) \cdot \log s$ | $\text{poly}(\lambda)$ |
| [ZYH24 ] | M-SIS | $\text{poly}(\lambda)$ | $\text{poly}(\lambda)^*$ | $\text{poly}(\lambda)$ | $\text{poly}(\lambda) \cdot s \log s$ |
| [CP23 ] | M-SIS | $\ell \cdot \text{poly}(\lambda)$ | $\ell^*$ | $\ell$ | $\ell \cdot \text{poly}(\lambda)$ |
| [CP23 ]+ [WW23 ] | $\ell$-Succinct M-SIS | $\text{poly}(\lambda)$ | $\ell^*$ | $\ell$ | $\ell^2 \cdot \text{poly}(\lambda)$ |
| **Our work** | M-SIS | $\ell \cdot \text{poly}(\lambda)$ | – | $\ell$ | $\ell \cdot \text{poly}(\lambda)$ |
| **Our work** | $\ell$-Succinct M-SIS | $\text{poly}(\lambda)$ | – | $\ell$ | $\ell^2 \cdot \text{poly}(\lambda)$ |

- $\ell$: Input's bit length
- $*$: $|\text{upmsg}| = 0$ for a fix set in pre-processing
- $s$: Size of the set

# Digital Signature

Let $\Sigma = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Verify})$ be a digital signature

- $\mathsf{Gen}(1^\lambda) \to (\mathsf{pk}, \mathsf{sk})$
- $\mathsf{Sign}(\mathsf{sk}, m) \to \sigma$
- $\mathsf{Verify}(\mathsf{pk}, m, \sigma) \to 1/0$

**Security**

It should be hard for an adversary to generate $(m^*, \sigma^*)$ given pk and $\{(m_i, \sigma_i)\}$ where $m^* \neq m_i$ for all $i$.

# Positive Dynamic Accumulator from Digital Signature

Let $\Sigma = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Verify})$ be a digital signature. In addition, suppose $\Sigma$ supports the following operations:

- $\mathsf{UpdatePK}(\mathsf{pk}, \mathsf{sk}, \bar{m}) \rightarrow (\mathsf{pk}', \mathsf{upmsg})$
- $\mathsf{UpdateSig}(m, \sigma_m, \mathsf{upmsg}) \rightarrow \sigma'_m$

# Positive Dynamic Accumulator from Digital Signature

Let $\Sigma = (\text{Gen}, \text{Sign}, \text{Verify})$ be a digital signature. In addition, suppose $\Sigma$ supports the following operations:

- $\text{UpdatePK}(\text{pk}, \text{sk}, \bar{m}) \to (\text{pk}', \text{upmsg})$
- $\text{UpdateSig}(m, \sigma_m, \text{upmsg}) \to \sigma'_m$

**Desiderata**

- $\text{Verify}(\text{pk}', m, \sigma'_m) = 1$ with overwhelming probability for any $m \neq \bar{m}$
- $\text{Verify}(\text{pk}', \bar{m}, \sigma'_{\bar{m}}) = 0$ with overwhelming probability*

# Positive Dynamic Accumulator from Digital Signature

Let $\Sigma = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Verify})$ be a digital signature. In addition, suppose $\Sigma$ supports the following operations:

- $\mathsf{UpdatePK}(\mathsf{pk}, \mathsf{sk}, \bar{m}) \to (\mathsf{pk}', \mathsf{upmsg})$
- $\mathsf{UpdateSig}(m, \sigma_m, \mathsf{upmsg}) \to \sigma'_m$

**Desiderata**

- $\mathsf{Verify}(\mathsf{pk}', m, \sigma'_m) = 1$ with overwhelming probability for any $m \neq \bar{m}$
- $\mathsf{Verify}(\mathsf{pk}', \bar{m}, \sigma'_{\bar{m}}) = 0$ with overwhelming probability*

UpdatePK allows to *revoke* signatures on messages.

# Positive Dynamic Accumulator from Digital Signature

Given $\Sigma = (\text{Gen}, \text{Sign}, \text{UpdatePK}, \text{UpdateSig}, \text{Verify})$ we construct a positive dynamic accumulator as follows:

# Positive Dynamic Accumulator from Digital Signature

Given $\Sigma = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{UpdatePK}, \mathsf{UpdateSig}, \mathsf{Verify})$ we construct a positive dynamic accumulator as follows:

- $\mathsf{Add}(\mathsf{pk}, \mathsf{sk}, x)$:
  1. Compute $\sigma_x \leftarrow \Sigma.\mathsf{Sign}(\mathsf{pp}, \mathsf{sk}, x)$.
  2. Return $\sigma_x$ as $w_x$

# Positive Dynamic Accumulator from Digital Signature

Given $\Sigma = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{UpdatePK}, \mathsf{UpdateSig}, \mathsf{Verify})$ we construct a positive dynamic accumulator as follows:

- $\mathsf{Add}(\mathsf{pk}, \mathsf{sk}, x)$:
  1. Compute $\sigma_x \leftarrow \Sigma.\mathsf{Sign}(\mathsf{pp}, \mathsf{sk}, x)$.
  2. Return $\sigma_x$ as $w_x$

- $\mathsf{Delete}(\mathsf{pk}, \mathsf{sk}, y)$:
  1. Compute $(\mathsf{pk}', \mathsf{upmsg}) \leftarrow \Sigma.\mathsf{UpdatePK}(\mathsf{pk}, \mathsf{sk}, y)$.
  2. Return $(\mathsf{pk}', \mathsf{upmsg})$

# Positive Dynamic Accumulator from Digital Signature

Given $\Sigma = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{UpdatePK}, \mathsf{UpdateSig}, \mathsf{Verify})$ we construct a positive dynamic accumulator as follows:

- $\mathsf{Add}(\mathsf{pk}, \mathsf{sk}, x)$:
  1. Compute $\sigma_x \leftarrow \Sigma.\mathsf{Sign}(\mathsf{pp}, \mathsf{sk}, x)$.
  2. Return $\sigma_x$ as $w_x$
- $\mathsf{Delete}(\mathsf{pk}, \mathsf{sk}, y)$:
  1. Compute $(\mathsf{pk}', \mathsf{upmsg}) \leftarrow \Sigma.\mathsf{UpdatePK}(\mathsf{pk}, \mathsf{sk}, y)$.
  2. Return $(\mathsf{pk}', \mathsf{upmsg})$
- $\mathsf{MemWitUp}(x, w_x, \mathsf{upmsg})$:
  1. Parse $w_x$ as $\sigma_x$.
  2. Compute $\sigma'_x \leftarrow \Sigma.\mathsf{UpdateSig}(x, \sigma_x, \mathsf{upmsg})$.
  3. Return $\sigma'_x$ as $w'_x$.

# Positive Dynamic Accumulator from Digital Signature

Given $\Sigma = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{UpdatePK}, \mathsf{UpdateSig}, \mathsf{Verify})$ we construct a positive dynamic accumulator as follows:

- $\mathsf{Add}(\mathsf{pk}, \mathsf{sk}, x)$:
  1. Compute $\sigma_x \leftarrow \Sigma.\mathsf{Sign}(\mathsf{pp}, \mathsf{sk}, x)$.
  2. Return $\sigma_x$ as $w_x$
- $\mathsf{Delete}(\mathsf{pk}, \mathsf{sk}, y)$:
  1. Compute $(\mathsf{pk}', \mathsf{upmsg}) \leftarrow \Sigma.\mathsf{UpdatePK}(\mathsf{pk}, \mathsf{sk}, y)$.
  2. Return $(\mathsf{pk}', \mathsf{upmsg})$
- $\mathsf{MemWitUp}(x, w_x, \mathsf{upmsg})$:
  1. Parse $w_x$ as $\sigma_x$.
  2. Compute $\sigma'_x \leftarrow \Sigma.\mathsf{UpdateSig}(x, \sigma_x, \mathsf{upmsg})$.
  3. Return $\sigma'_x$ as $w'_x$.
- $\mathsf{MemVerify}(\mathsf{pk}, x, w_x)$:
  1. Parse $w_x$ as $\sigma_x$.
  2. Return $\Sigma.\mathsf{Verify}(\mathsf{pk}, x, \sigma_x)$.

# Positive Dynamic Accumulator from Digital Signature

Given $\Sigma = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{UpdatePK}, \mathsf{UpdateSig}, \mathsf{Verify})$ we construct a positive dynamic accumulator as follows:

- $\mathsf{Add}(\mathsf{pk}, \mathsf{sk}, x)$:
  1. Compute $\sigma_x \leftarrow \Sigma.\mathsf{Sign}(\mathsf{pp}, \mathsf{sk}, x)$.
  2. Return $\sigma_x$ as $w_x$
- $\mathsf{Delete}(\mathsf{pk}, \mathsf{sk}, y)$:
  1. Compute $(\mathsf{pk}', \mathsf{upmsg}) \leftarrow \Sigma.\mathsf{UpdatePK}(\mathsf{pk}, \mathsf{sk}, y)$.
  2. Return $(\mathsf{pk}', \mathsf{upmsg})$
- $\mathsf{MemWitUp}(x, w_x, \mathsf{upmsg})$:
  1. Parse $w_x$ as $\sigma_x$.
  2. Compute $\sigma'_x \leftarrow \Sigma.\mathsf{UpdateSig}(x, \sigma_x, \mathsf{upmsg})$.
  3. Return $\sigma'_x$ as $w'_x$.
- $\mathsf{MemVerify}(\mathsf{pk}, x, w_x)$:
  1. Parse $w_x$ as $\sigma_x$.
  2. Return $\Sigma.\mathsf{Verify}(\mathsf{pk}, x, \sigma_x)$.

This construction is communication efficient, i.e., $|\mathsf{upmsg}| = O(\#\mathsf{Del})$.

# Gadget Matrix

[MP12]

Let $R_q \supseteq \mathbb{Z}_q$ be a ring such that $R_q^m$ admits an $\ell_\infty$-norm

$$\mathbf{G} = \begin{bmatrix} 1, 2, 4, \ldots, 2^{k-1} & & & \\ & 1, 2, 4, \ldots, 2^{k-1} & & \\ & & \ddots & \\ & & & 1, 2, 4, \ldots, 2^{k-1} \end{bmatrix} \in R_q^{n \times nk}$$

# Gadget Matrix

[MP12]

Let $R_q \supseteq \mathbb{Z}_q$ be a ring such that $R_q^m$ admits an $\ell_\infty$-norm

$$\mathbf{G} = \begin{bmatrix} 1, 2, 4, \ldots, 2^{k-1} & & & \\ & 1, 2, 4, \ldots, 2^{k-1} & & \\ & & \ddots & \\ & & & 1, 2, 4, \ldots, 2^{k-1} \end{bmatrix} \in R_q^{n \times nk}$$

- $k = \lceil \log q \rceil$.
- There exists a *decomposition* function $\mathbf{G}^{-1} : R_q^n \to R_q^{nk}$ such that for any $\mathbf{u} \in R_q^n$, we have $\mathbf{G} \cdot \mathbf{G}^{-1}(\mathbf{u}) = \mathbf{u}$ and $\|\mathbf{G}^{-1}(\mathbf{u})\|_\infty = 1$

# Homomorphic Operations on Matrices

**[GSW13 ; BGG+14 ; CP23 ]**

For any $\ell \in \mathbb{N}$, let $\mathcal{F} = \{f_i : \{0,1\}^\ell \to \{0,1\}\}_{i \in \mathbb{N}}$ be a family of Boolean circuits. Then, there exist efficient algorithm EvalF and EvalFX such that for any $\mathbf{B} \in R_q^{n \times \ell m}$, $f \in \mathcal{F}$, and $x \in \{0,1\}^\ell$:

- $\mathsf{EvalF}(f, \mathbf{B}) \to \mathbf{B}_f$
- $\mathsf{EvalFX}(f, \mathbf{B}, x) \to \mathbf{H}_{f,x}$ with $\|\mathbf{H}_{f,x}\|_\infty = 1$

$$\text{s.t.} \quad (\mathbf{B} - x \otimes \mathbf{G}) \cdot \mathbf{H}_{f,x} = \mathbf{B}_f - f(x) \cdot \mathbf{G}$$

# Homomorphic Operations on Matrices

[GSW13 ; BGG+14 ; CP23 ]

For any $\ell \in \mathbb{N}$, let $\mathcal{F} = \{f_i : \{0,1\}^\ell \to \{0,1\}\}_{i\in\mathbb{N}}$ be a family of Boolean circuits. Then, there exist efficient algorithm EvalF and EvalFX such that for any $\mathbf{B} \in R_q^{n\times \ell m}$, $f \in \mathcal{F}$, and $x \in \{0,1\}^\ell$:

- $\mathsf{EvalF}(f, \mathbf{B}) \to \mathbf{B}_f$
- $\mathsf{EvalFX}(f, \mathbf{B}, x) \to \mathbf{H}_{f,x}$ with $\|\mathbf{H}_{f,x}\|_\infty = 1$

$$\text{s.t.} \quad (\mathbf{B} - x \otimes \mathbf{G}) \cdot \mathbf{H}_{f,x} = \mathbf{B}_f - f(x) \cdot \mathbf{G}$$

# Homomorphic Operations on Matrices

[GSW13 ; BGG+14 ; CP23 ]

For any $\ell \in \mathbb{N}$, let $\mathcal{F} = \{f_i : \{0,1\}^\ell \to \{0,1\}\}_{i \in \mathbb{N}}$ be a family of Boolean circuits. Then, there exist efficient algorithm EvalF and EvalFX such that for any $\mathbf{B} \in R_q^{n \times \ell m}$, $f \in \mathcal{F}$, and $x \in \{0,1\}^\ell$:

- $\mathsf{EvalF}(f, \mathbf{B}) \to \mathbf{B}_f$
- $\mathsf{EvalFX}(f, \mathbf{B}, x) \to \mathbf{H}_{f,x}$ with $\|\mathbf{H}_{f,x}\|_\infty = 1$

$$\text{s.t.} \quad (\mathbf{B} - x \otimes \mathbf{G}) \cdot \mathbf{H}_{f,x} = \mathbf{B}_f - f(x) \cdot \mathbf{G}$$

$$\mathcal{F}_{\text{Indicator}} : \{\mathbb{1}_y : \{0,1\}^\ell \to \{0,1\}\}, \text{ where } \mathbb{1}_y(x) = \begin{cases} 1 \text{ if } x = y \\ 0 \text{ otherwise} \end{cases}$$

# Our Construction

## Communication efficient accumulator

$$\mathsf{pp} = (\mathbf{A} \in R_q^{n \times \bar{m}}, \mathbf{B} \in R_q^{n \times \ell m}), \mathsf{sk} = \mathbf{T}_{\mathbf{A}}, \mathscr{A}_0 \leftarrow\!\!\$\; R_q^{n \times m}$$

sk *allows to compute a low-norm matrix* $\mathbf{V} \leftarrow \mathsf{SamplePre}_{\mathsf{sk}}([\mathbf{A} \mid \bar{\mathbf{B}}], \mathbf{U})$ *s.t.* $[\mathbf{A} \mid \bar{\mathbf{B}}] \cdot \mathbf{V} = \mathbf{U}$ *for any* $\bar{\mathbf{B}}$.

# Our Construction

## Communication efficient accumulator

$$\mathsf{pp} = (\mathbf{A} \in R_q^{n \times \bar{m}}, \mathbf{B} \in R_q^{n \times \ell m}), \mathsf{sk} = \mathbf{T}_{\mathbf{A}}, \mathscr{A}_0 \leftarrow\!\!\$\; R_q^{n \times m}$$

sk *allows to compute a low-norm matrix* $\mathbf{V} \leftarrow \mathsf{SamplePre}_{\mathsf{sk}}([\mathbf{A} \mid \bar{\mathbf{B}}], \mathbf{U})$ *s.t.* $[\mathbf{A} \mid \bar{\mathbf{B}}] \cdot \mathbf{V} = \mathbf{U}$ *for any* $\bar{\mathbf{B}}$.

- $\mathsf{Add}(\mathsf{pp}, \mathsf{sk}, \mathscr{A}, x)$:
  1. Sample $\mathbf{S}_x \leftarrow \mathsf{SamplePre}_{\mathsf{sk}}([\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}], \mathscr{A})$
  2. Return $\mathbf{S}_x$ as $w_x$

*Agrawal-Boneh-Boyen [ABB10 ] signature*

# Our Construction

## Communication efficient accumulator

$$\mathsf{pp} = (\mathbf{A} \in R_q^{n\times \bar{m}}, \mathbf{B} \in R_q^{n\times \ell m}), \mathsf{sk} = \mathbf{T}_{\mathbf{A}}, \mathscr{A}_0 \leftarrow\!\!\$\ R_q^{n\times m}$$

sk *allows to compute a low-norm matrix* $\mathbf{V} \leftarrow \mathsf{SamplePre}_{\mathsf{sk}}([\mathbf{A} \mid \bar{\mathbf{B}}], \mathbf{U})$ *s.t.* $[\mathbf{A} \mid \bar{\mathbf{B}}] \cdot \mathbf{V} = \mathbf{U}$ *for any* $\bar{\mathbf{B}}$.

- $\mathsf{Add}(\mathsf{pp}, \mathsf{sk}, \mathscr{A}, x)$:
  1. Sample $\mathbf{S}_x \leftarrow \mathsf{SamplePre}_{\mathsf{sk}}([\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}], \mathscr{A})$
  2. Return $\mathbf{S}_x$ as $w_x$

  *Agrawal-Boneh-Boyen [ABB10 ] signature*

- $\mathsf{Delete}(\mathsf{pp}, \mathscr{A}, y)$:
  1. Compute $\mathbf{B}_{\mathbb{1}_y} \leftarrow \mathsf{EvalF}(\mathbb{1}_y, \mathbf{B})$
  2. Compute $\mathscr{A}' \leftarrow \mathscr{A} + \mathbf{B}_{\mathbb{1}_y}$
  3. Return $(\mathscr{A}', \mathsf{upmsg} = \{y\})$

# Our Construction

## Communication efficient accumulator

$$\mathsf{pp} = (\mathbf{A} \in R_q^{n\times\bar{m}}, \mathbf{B} \in R_q^{n\times \ell m}), \mathsf{sk} = \mathbf{T}_\mathbf{A}, \mathscr{A}_0 \leftarrow\!\!\$\; R_q^{n\times m}$$

sk *allows to compute a low-norm matrix* $\mathbf{V} \leftarrow \mathsf{SamplePre}_{\mathsf{sk}}([\mathbf{A} \mid \bar{\mathbf{B}}], \mathbf{U})$ *s.t.* $[\mathbf{A} \mid \bar{\mathbf{B}}] \cdot \mathbf{V} = \mathbf{U}$ *for any* $\bar{\mathbf{B}}$.

- $\mathsf{Add}(\mathsf{pp}, \mathsf{sk}, \mathscr{A}, x)$:
  1. Sample $\mathbf{S}_x \leftarrow \mathsf{SamplePre}_{\mathsf{sk}}([\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}], \mathscr{A})$
  2. Return $\mathbf{S}_x$ as $w_x$

  *Agrawal-Boneh-Boyen [ABB10 ] signature*

- $\mathsf{MemWitUp}(\mathsf{pp}, x, w_x, \mathsf{upmsg} = \{y\})$:
  1. Compute $\mathbf{H}_{\mathbb{1}_y,\mathbf{B},x} \leftarrow \mathsf{EvalFX}(\mathbb{1}_y, \mathbf{B}, x)$
  2. Compute $w'_x \leftarrow w_x + \begin{bmatrix} \mathbf{0} \\ \mathbf{H}_{\mathbb{1}_y,\mathbf{B},x} \end{bmatrix}$
  3. Return $w'_x$

- $\mathsf{Delete}(\mathsf{pp}, \mathscr{A}, y)$:
  1. Compute $\mathbf{B}_{\mathbb{1}_y} \leftarrow \mathsf{EvalF}(\mathbb{1}_y, \mathbf{B})$
  2. Compute $\mathscr{A}' \leftarrow \mathscr{A} + \mathbf{B}_{\mathbb{1}_y}$
  3. Return $(\mathscr{A}', \mathsf{upmsg} = \{y\})$

# Our Construction

## Communication efficient accumulator

$$\mathsf{pp} = (\mathbf{A} \in R_q^{n\times\bar{m}}, \mathbf{B} \in R_q^{n\times \ell m}), \mathsf{sk} = \mathbf{T}_{\mathbf{A}}, \mathscr{A}_0 \leftarrow\!\!\$\ R_q^{n\times m}$$

sk *allows to compute a low-norm matrix* $\mathbf{V} \leftarrow \mathsf{SamplePre}_{\mathsf{sk}}([\mathbf{A} \mid \bar{\mathbf{B}}], \mathbf{U})$ *s.t.* $[\mathbf{A} \mid \bar{\mathbf{B}}] \cdot \mathbf{V} = \mathbf{U}$ *for any* $\bar{\mathbf{B}}$.

- $\mathsf{Add}(\mathsf{pp}, \mathsf{sk}, \mathscr{A}, x)$:
  1. Sample $\mathbf{S}_x \leftarrow \mathsf{SamplePre}_{\mathsf{sk}}([\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}], \mathscr{A})$
  2. Return $\mathbf{S}_x$ as $w_x$

  *Agrawal-Boneh-Boyen [ABB10] signature*

- $\mathsf{Delete}(\mathsf{pp}, \mathscr{A}, y)$:
  1. Compute $\mathbf{B}_{\mathbb{1}_y} \leftarrow \mathsf{EvalF}(\mathbb{1}_y, \mathbf{B})$
  2. Compute $\mathscr{A}' \leftarrow \mathscr{A} + \mathbf{B}_{\mathbb{1}_y}$
  3. Return $(\mathscr{A}', \mathsf{upmsg} = \{y\})$

- $\mathsf{MemWitUp}(\mathsf{pp}, x, w_x, \mathsf{upmsg} = \{y\})$:
  1. Compute $\mathbf{H}_{\mathbb{1}_y,\mathbf{B},x} \leftarrow \mathsf{EvalFX}(\mathbb{1}_y, \mathbf{B}, x)$
  2. Compute $w_x' \leftarrow w_x + \begin{bmatrix} \mathbf{0} \\ \mathbf{H}_{\mathbb{1}_y,\mathbf{B},x} \end{bmatrix}$
  3. Return $w_x'$

- $\mathsf{MemVerify}(\mathsf{pp}, \mathscr{A}, x, w_x)$:
  1. Check if $[\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}] \cdot w_x = \mathscr{A}$ and $\|w_x\|_\infty$ is small

# Our Construction

## Communication efficient accumulator – Correctness

Let $x \in \{0,1\}^{\ell}$ with an updated witness $w'_x$ that was generated after deleting $y \neq x \in \{0,1\}^{\ell}$.
We have $\mathscr{A}' = \mathscr{A} + \mathbf{B}_{\mathbb{1}_y}$.

# Our Construction

## Communication efficient accumulator – Correctness

Let $x \in \{0,1\}^\ell$ with an updated witness $w'_x$ that was generated after deleting $y \neq x \in \{0,1\}^\ell$.
We have $\mathscr{A}' = \mathscr{A} + \mathbf{B}_{\mathbb{1}_y}$.

- $w'_x = w_x + \begin{bmatrix} \mathbf{0} \\ \mathbf{H}_{\mathbb{1}_y,\mathbf{B},x} \end{bmatrix}$, where $w_x = \mathbf{S}_x \leftarrow \mathsf{SamplePre}_{\mathsf{sk}}([\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}], \mathscr{A})$

# Our Construction

## Communication efficient accumulator – Correctness

Let $x \in \{0,1\}^\ell$ with an updated witness $w'_x$ that was generated after deleting $y \neq x \in \{0,1\}^\ell$.
We have $\mathscr{A}' = \mathscr{A} + \mathbf{B}_{\mathbb{1}_y}$.

- $w'_x = w_x + \begin{bmatrix} \mathbf{0} \\ \mathbf{H}_{\mathbb{1}_y,\mathbf{B},x} \end{bmatrix}$, where $w_x = \mathbf{S}_x \leftarrow \mathsf{SamplePre}_{\mathsf{sk}}([\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}], \mathscr{A})$
- Therefore,

$$
\begin{aligned}
[\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}] \left( \mathbf{S}_x + \begin{bmatrix} \mathbf{0} \\ \mathbf{H}_{\mathbb{1}_y,\mathbf{B},x} \end{bmatrix} \right) &= \mathscr{A} + (\mathbf{B} - x \otimes \mathbf{G}) \cdot \mathbf{H}_{\mathbb{1}_y,\mathbf{B},x} \\
&= \mathscr{A} + \mathbf{B}_{\mathbb{1}_y} - \mathbb{1}_y(x)\mathbf{G} \\
&= \mathscr{A}' \quad (\text{Since } \mathbb{1}_y(x) = 0)
\end{aligned}
$$

# Our Construction

## Communication efficient accumulator – Correctness

Let $x \in \{0,1\}^\ell$ with an updated witness $w'_x$ that was generated after deleting $y \neq x \in \{0,1\}^\ell$.
We have $\mathscr{A}' = \mathscr{A} + \mathbf{B}_{\mathbb{1}_y}$.

- $w'_x = w_x + \begin{bmatrix} \mathbf{0} \\ \mathbf{H}_{\mathbb{1}_y,\mathbf{B},x} \end{bmatrix}$, where $w_x = \mathbf{S}_x \leftarrow \mathsf{SamplePre}_{\mathsf{sk}}([\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}], \mathscr{A})$
- Therefore,

$$\begin{aligned}[\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}]\left(\mathbf{S}_x + \begin{bmatrix} \mathbf{0} \\ \mathbf{H}_{\mathbb{1}_y,\mathbf{B},x} \end{bmatrix}\right) &= \mathscr{A} + (\mathbf{B} - x \otimes \mathbf{G}) \cdot \mathbf{H}_{\mathbb{1}_y,\mathbf{B},x} \\ &= \mathscr{A} + \mathbf{B}_{\mathbb{1}_y} - \mathbb{1}_y(x)\mathbf{G} \\ &= \mathscr{A}' \quad (\text{Since } \mathbb{1}_y(x) = 0)\end{aligned}$$

- $\|w'_x\|_\infty = \|w_x\|_\infty + \left\|\mathbf{H}_{\mathbb{1}_y,\mathbf{B},x}\right\|_\infty = \|w_x\|_\infty + 1$

By setting the *noise* budget accordingly, we can support poly deletions.

# Our Construction

## Communication efficient accumulator – Instantiation

| **Scheme** | $q$ | #Add | #Del | $\lvert w_{\mathbf{x}} \rvert$ | $\lvert \text{upmsg} \rvert_{\text{Add}}$ | $\lvert \text{upmsg} \rvert_{\text{Del}}$ | $\lvert \mathscr{A} \rvert$ | $\lvert \text{pp} \rvert$ |
|---|---|---|---|---|---|---|---|---|
| [CP23 ] (M-SIS) | $\approx 2^{90}$ | $2^{32}$ | $2^{32}$ | 12MB | 4 B | 4 B | 45KB | 14.2MB |
| [CP23 ]+[WW23 ] ($\ell$-Succinct M-SIS) | $\approx 2^{150}$ | $2^{32}$ | $2^{32}$ | 5.5MB | 4 B | 4 B | 75KB | 77.3MB |
| Our work (M-SIS) | $\approx 2^{100}$ | — | $2^{32}$ | 14.72MB | — | 4 B | 50KB | 16.7MB |
| Our work ($\ell$-Succinct M-SIS) | $\approx 2^{162}$ | — | $2^{32}$ | 9.33MB | — | 4 B | 81KB | 171.7MB |

# Our Construction

## Communication efficient accumulator – Instantiation

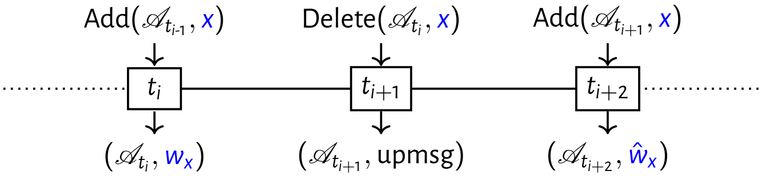
| **Scheme** | $q$ | #Add | #Del | $\lvert w_{\mathbf{x}} \rvert$ | $\lvert \text{upmsg} \rvert_{\text{Add}}$ | $\lvert \text{upmsg} \rvert_{\text{Del}}$ | $\lvert \mathscr{A} \rvert$ | $\lvert \text{pp} \rvert$ |
|---|---|---|---|---|---|---|---|---|
| [CP23] (M-SIS) | $\approx 2^{90}$ | $2^{32}$ | $2^{32}$ | 12MB | 4 B | 4 B | 45KB | 14.2MB |
| [CP23]+[WW23] ($\ell$-Succinct M-SIS) | $\approx 2^{150}$ | $2^{32}$ | $2^{32}$ | 5.5MB | 4 B | 4 B | 75KB | 77.3MB |
| Our work (M-SIS) | $\approx 2^{100}$ | — | $2^{32}$ | 14.72MB | — | 4 B | 50KB | 16.7MB |
| Our work ($\ell$-Succinct M-SIS) | $\approx 2^{162}$ | — | $2^{32}$ | 9.33MB | — | 4 B | 81KB | 171.7MB |

# Security Analysis

- **Replacement-free condition**: Cannot re-add $x$ after it was deleted.

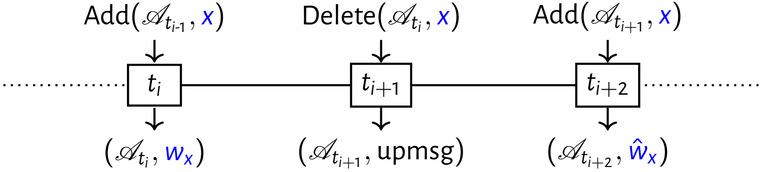
# Security Analysis

- **Replacement-free condition**: Cannot re-add $x$ after it was deleted.



# Security Analysis

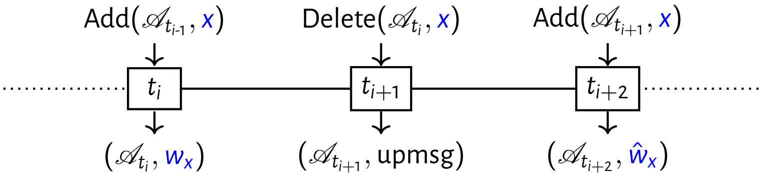
- **Replacement-free condition**: Cannot re-add $x$ after it was deleted.




By using EvalFX, we can compute $\tilde{w}_x$ from $w_x$ such that $[\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}] \cdot \tilde{w}_x = \mathscr{A}_{t_{i+2}} - \mathbf{G}$.
And $[\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}] \cdot \hat{w}_x = \mathscr{A}_{t_{i+2}}$.

# Security Analysis

- **Replacement-free condition**: Cannot re-add $x$ after it was deleted.



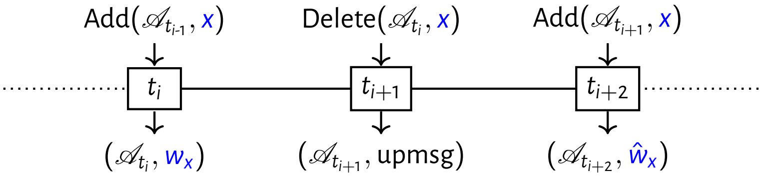

By using EvalFX, we can compute $\tilde{w}_x$ from $w_x$ such that $[\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}] \cdot \tilde{w}_x = \mathscr{A}_{t_{i+2}} - \mathbf{G}$.
And $[\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}] \cdot \hat{w}_x = \mathscr{A}_{t_{i+2}}$.

$$[\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}] \cdot (\hat{w}_x - \tilde{w}_x) = \mathbf{G}$$

**Note**: $\hat{w}_x - \tilde{w}_x$ can be used as a $\mathbf{G}$-trapdoor to forge membership witnesses for $x$.

# Security Analysis

- **Replacement-free condition**: Cannot re-add $x$ after it was deleted.




By using EvalFX, we can compute $\tilde{w}_x$ from $w_x$ such that $[\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}] \cdot \tilde{w}_x = \mathscr{A}_{t_{i+2}} - \mathbf{G}$.
And $[\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}] \cdot \hat{w}_x = \mathscr{A}_{t_{i+2}}$.

$$[\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}] \cdot (\hat{w}_x - \tilde{w}_x) = \mathbf{G}$$

**Note**: $\hat{w}_x - \tilde{w}_x$ can be used as a $\mathbf{G}$-trapdoor to forge membership witnesses for $x$.

**Theorem**

*If the replacement-free condition holds and the (module) Short Integer Solution problem is hard, then our construction is a selectively secure communication efficient positive dynamic* accumulator.*

# Security Analysis

**Short Integer Solution $(n, m, \beta)$**

Given $\bar{\mathbf{A}} \leftarrow\!\!\$\ R_q^{n\times m}$, find $\mathbf{v} \neq \mathbf{0}$ such that $\|\mathbf{v}\| \leq \beta$ and

- $\bar{\mathbf{A}}\mathbf{v} = \mathbf{0}$, for the homogeneous case.
- $\bar{\mathbf{A}}\mathbf{v} = \mathbf{t}$, for the inhomogeneous case w.r.t target $\mathbf{t} \neq 0$.

# Security Analysis

**Short Integer Solution $(n, m, \beta)$**

Given $\bar{\mathbf{A}} \leftarrow\!\!\$\ R_q^{n \times m}$, find $\mathbf{v} \neq \mathbf{0}$ such that $\|\mathbf{v}\| \leq \beta$ and

- $\bar{\mathbf{A}}\mathbf{v} = \mathbf{0}$, for the homogeneous case.
- $\bar{\mathbf{A}}\mathbf{v} = \mathbf{t}$, for the inhomogeneous case w.r.t target $\mathbf{t} \neq 0$.

Suppose a selective adversary $\mathcal{A}$ outputs a forgery $(x^*, w_{x^*})$

# Security Analysis

**Short Integer Solution $(n, m, \beta)$**

Given $\bar{\mathbf{A}} \leftarrow\!\!\$\ R_q^{n\times m}$, find $\mathbf{v} \neq \mathbf{0}$ such that $\|\mathbf{v}\| \leq \beta$ and

- $\bar{\mathbf{A}}\mathbf{v} = \mathbf{0}$, for the homogeneous case.
- $\bar{\mathbf{A}}\mathbf{v} = \mathbf{t}$, for the inhomogeneous case w.r.t target $\mathbf{t} \neq 0$.

Suppose a selective adversary $\mathcal{A}$ outputs a forgery $(x^*, w_{x^*})$

**Case 1:** $x^*$ was never added to the accumulator.

Then $[\mathbf{A} \mid \mathbf{B} - x^* \otimes \mathbf{G}] \cdot w_{x^*} = \mathscr{A}$.

Since $w_{x^*}$ is *short*, it is an inhomogeneous solution for $[\mathbf{A} \mid \mathbf{B} - x^* \otimes \mathbf{G}]$.

# Security Analysis

**Short Integer Solution $(n, m, \beta)$**

Given $\bar{\mathbf{A}} \leftarrow\$ R_q^{n\times m}$, find $\mathbf{v} \neq \mathbf{0}$ such that $\|\mathbf{v}\| \leq \beta$ and

- $\bar{\mathbf{A}}\mathbf{v} = \mathbf{0}$, for the homogeneous case.
- $\bar{\mathbf{A}}\mathbf{v} = \mathbf{t}$, for the inhomogeneous case w.r.t target $\mathbf{t} \neq 0$.

Suppose a selective adversary $\mathcal{A}$ outputs a forgery $(x^*, w_{x^*})$

**Case 1:** $x^*$ was never added to the accumulator.
Then $[\mathbf{A} \mid \mathbf{B} - x^* \otimes \mathbf{G}] \cdot w_{x^*} = \mathscr{A}$.
Since $w_{x^*}$ is *short*, it is an inhomogeneous solution for $[\mathbf{A} \mid \mathbf{B} - x^* \otimes \mathbf{G}]$.

**Case 2:** $x^*$ was added then remove from the accumulator.
Then there exists $\tilde{w}_{x^*} \neq w_{x^*}$ such that $[\mathbf{A} \mid \mathbf{B} - x^* \otimes \mathbf{G}] \cdot \tilde{w}_{x^*} = \mathscr{A} - \mathbf{G}$.

# Security Analysis

**Short Integer Solution $(n, m, \beta)$**

Given $\bar{\mathbf{A}} \leftarrow\!\!\$\ R_q^{n\times m}$, find $\mathbf{v} \neq \mathbf{0}$ such that $\|\mathbf{v}\| \leq \beta$ and

- $\bar{\mathbf{A}}\mathbf{v} = \mathbf{0}$, for the homogeneous case.
- $\bar{\mathbf{A}}\mathbf{v} = \mathbf{t}$, for the inhomogeneous case w.r.t target $\mathbf{t} \neq 0$.

Suppose a selective adversary $\mathcal{A}$ outputs a forgery $(x^*, w_{x^*})$

**Case 1:** $x^*$ was never added to the accumulator.
Then $[\mathbf{A} \mid \mathbf{B} - x^* \otimes \mathbf{G}] \cdot w_{x^*} = \mathscr{A}$.
Since $w_{x^*}$ is *short*, it is an inhomogeneous solution for $[\mathbf{A} \mid \mathbf{B} - x^* \otimes \mathbf{G}]$.

**Case 2:** $x^*$ was added then remove from the accumulator.
Then there exists $\tilde{w}_{x^*} \neq w_{x^*}$ such that
$[\mathbf{A} \mid \mathbf{B} - x^* \otimes \mathbf{G}] \cdot \tilde{w}_{x^*} = \mathscr{A} - \mathbf{G}$. Therefore,
$[\mathbf{A} \mid \mathbf{B} - x^* \otimes \mathbf{G}] \cdot (w_{x^*} - \tilde{w}_{x^*}) = \mathbf{G}$.
Hence, using $(w_{x^*} - \tilde{w}_{x^*})$ we can sample a short $\mathbf{v} \neq \mathbf{0}$ and $[\mathbf{A} \mid \mathbf{B} - x^* \otimes \mathbf{G}]\mathbf{v} = \mathbf{0}$

# Security Analysis

**Short Integer Solution $(n, m, \beta)$**

Given $\bar{\mathbf{A}} \leftarrow\!\!\$\ R_q^{n\times m}$, find $\mathbf{v} \neq \mathbf{0}$ such that $\|\mathbf{v}\| \leq \beta$ and

- $\bar{\mathbf{A}}\mathbf{v} = \mathbf{0}$, for the homogeneous case.
- $\bar{\mathbf{A}}\mathbf{v} = \mathbf{t}$, for the inhomogeneous case w.r.t target $\mathbf{t} \neq 0$.

Suppose a selective adversary $\mathcal{A}$ outputs a forgery $(x^*, w_{x^*})$

**Case 1:** $x^*$ was never added to the accumulator.
Then $[\mathbf{A} \mid \mathbf{B} - x^* \otimes \mathbf{G}] \cdot w_{x^*} = \mathscr{A}$.
Since $w_{x^*}$ is *short*, it is an inhomogeneous solution for $[\mathbf{A} \mid \mathbf{B} - x^* \otimes \mathbf{G}]$.

**Case 2:** $x^*$ was added then remove from the accumulator.
Then there exists $\tilde{w}_{x^*} \neq w_{x^*}$ such that
$[\mathbf{A} \mid \mathbf{B} - x^* \otimes \mathbf{G}] \cdot \tilde{w}_{x^*} = \mathscr{A} - \mathbf{G}$. Therefore,
$[\mathbf{A} \mid \mathbf{B} - x^* \otimes \mathbf{G}] \cdot (w_{x^*} - \tilde{w}_{x^*}) = \mathbf{G}$.
Hence, using $(w_{x^*} - \tilde{w}_{x^*})$ we can sample a short $\mathbf{v} \neq \mathbf{0}$ and $[\mathbf{A} \mid \mathbf{B} - x^* \otimes \mathbf{G}]\mathbf{v} = \mathbf{0}$

**Note:** Under the replacement-free condition, these two cases are sufficient.

# Security Analysis

- The accumulator needs to be replacement-free and is only selectively secure. Is that not undesirable?

# Security Analysis

- The accumulator needs to be replacement-free and is only selectively secure. Is that not undesirable?

**Theorem**

*Non-adaptively secure Positive Dynamic Accumulator* + *Adaptively secure Digital signature* $\overset{\text{[BCD+17]}}{\Longrightarrow}$ *Adaptively secure Positive Dynamic Accumulator*

# Security Analysis

- The accumulator needs to be replacement-free and is only selectively secure. Is that not undesirable?

**Theorem**

*Non-adaptively secure Positive Dynamic Accumulator (Communication efficient)* + *Adaptively secure Digital signature* $\overset{\text{[BCD+17]}}{\Longrightarrow}$ *Adaptively secure Positive Dynamic Accumulator (Communication efficient)*

# Security Analysis

- The accumulator needs to be replacement-free and is only selectively secure. Is that not undesirable?

**Theorem**

*Selectively secure Positive Replacement-free Accumulator (Communication efficient)* + *Adaptively secure Digital signature* $\overset{\text{[BCD+17]}}{\Longrightarrow}$ *Adaptively secure Positive Dynamic Accumulator (Communication efficient)*

# Security Analysis

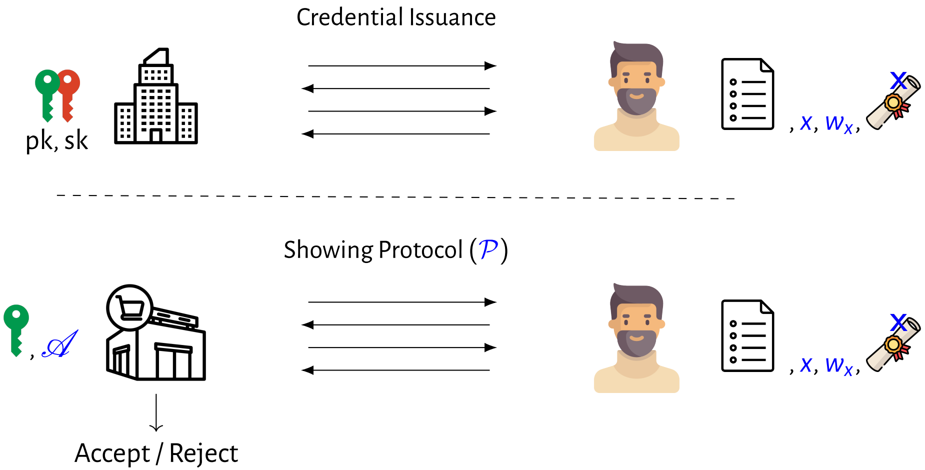
- The accumulator needs to be replacement-free and is only selectively secure. Is that not undesirable?

**Theorem**

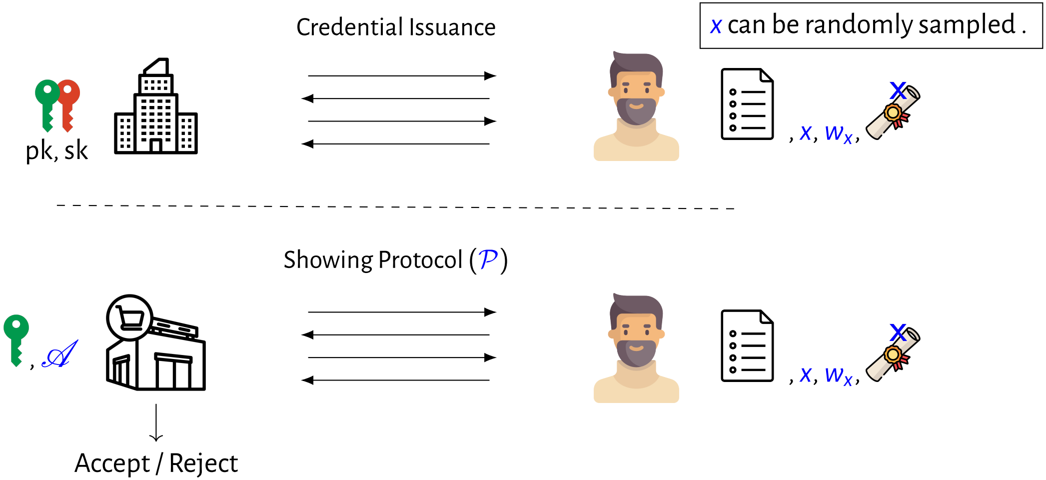
*Selectively secure Positive Replacement-free Accumulator (Communication efficient)* + *Adaptively secure Digital signature* $\overset{[BCD+17]}{\Longrightarrow}$ *Adaptively secure Positive Dynamic Accumulator (Communication efficient)*

**Note**: A replacement-free selectively secure accumulator is sufficient for Anonymous Credential Revocation.

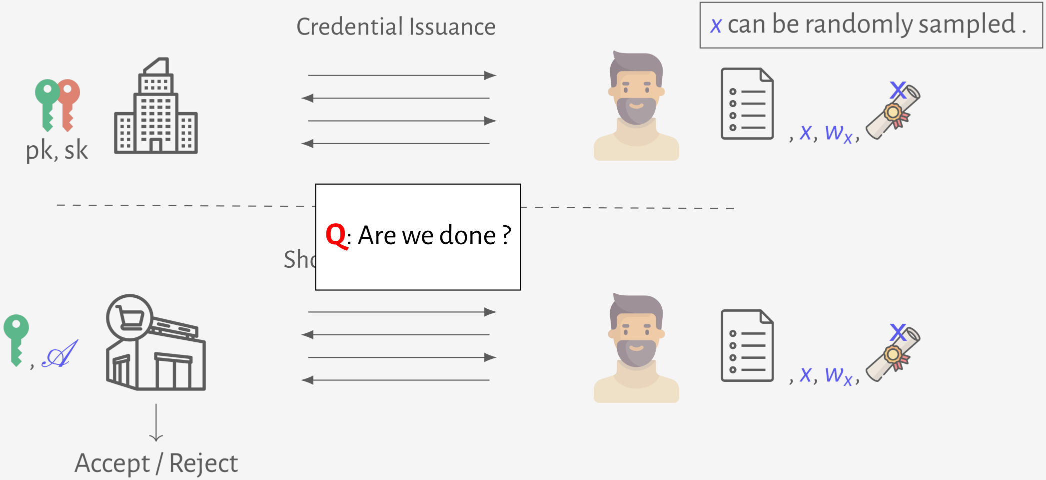
# Replacement-free Selectively Secure Accumulator in ACs revocation



# Replacement-free Selectively Secure Accumulator in ACs revocation

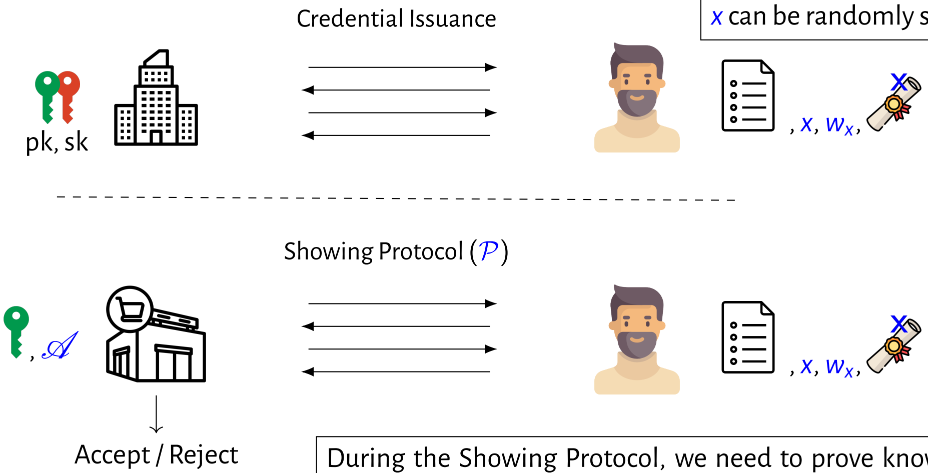


# Replacement-free Selectively Secure Accumulator in ACs revocation



# Replacement-free Selectively Secure Accumulator in ACs revocation

$x$ can be randomly sampled .




During the Showing Protocol, we need to prove knowledge of $x$ and $w_x$ s.t. MemVerify($\mathscr{A}$, $x$, $w_x$) $= 1$.

# Replacement-free Selectively Secure Accumulator in ACs revocation

From Lattice-based zero-knowledge proofs [Lyu12 ; ENS20 ; LNP+21 ; LNP22 ; BS23 ], we know how to prove knowledge of **v** such that

$$\mathbf{C}\mathbf{v} = \mathbf{t}, \quad \|\mathbf{v}\| \leq \beta$$

# Replacement-free Selectively Secure Accumulator in ACs revocation

From Lattice-based zero-knowledge proofs [Lyu12 ; ENS20 ; LNP+21 ; LNP22 ; BS23 ], we know how to prove knowledge of **v** such that

$$\mathbf{C}\mathbf{v} = \mathbf{t}, \quad \|\mathbf{v}\| \leq \beta$$

For our construction, we need to prove knowledge of $(x, w_x)$ such that

$$[\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}] \cdot w_x = \mathscr{A}, \quad \|w_x\| \leq \beta' \tag{1}$$

# Replacement-free Selectively Secure Accumulator in ACs revocation

From Lattice-based zero-knowledge proofs [Lyu12 ; ENS20 ; LNP+21 ; LNP22 ; BS23 ], we know how to prove knowledge of **v** such that

$$\mathbf{C}\mathbf{v} = \mathbf{t}, \quad \|\mathbf{v}\| \le \beta$$

For our construction, we need to prove knowledge of $(x, w_x)$ such that

$$[\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}] \cdot w_x = \mathscr{A}, \quad \|w_x\| \le \beta' \tag{1}$$

# Replacement-free Selectively Secure Accumulator in ACs revocation

From Lattice-based zero-knowledge proofs [Lyu12 ; ENS20 ; LNP+21 ; LNP22 ; BS23 ], we know how to prove knowledge of $\mathbf{v}$ such that

$$\mathbf{C}\mathbf{v} = \mathbf{t}, \qquad \|\mathbf{v}\| \leq \beta$$

For our construction, we need to prove knowledge of $(x, w_x)$ such that

$$[\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}] \cdot w_x = \mathscr{A}, \qquad \|w_x\| \leq \beta' \tag{1}$$

How can we handle $x$?

- Compute a commitment $\mathrm{Com}(x; r)$ and produce a proof $\pi_{\mathrm{Com}} = (\mathbf{w}, c, z)$.
- From $z$, we can extract $z_x = y_x + c \cdot x$ such that

$$[c\mathbf{A} \mid c\mathbf{B} - z_x \otimes \mathbf{G}] \cdot w_x = c\underbrace{[\mathbf{A} \mid \mathbf{B} - x \otimes \mathbf{G}] \cdot w_x}_{\mathscr{A}} + [\mathbf{0} \mid -y_x \otimes \mathbf{G}] \cdot w_x$$

# Thank You!

https://ia.cr/2025/1099

Some icons were sourced from Flaticon.com

# Reference I


[ABB10] S. Agrawal, D. Boneh, and X. Boyen. "Efficient Lattice (H)IBE in the Standard Model". In: *Advances in Cryptology - EUROCRYPT 2010, 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Monaco / French Riviera, May 30 - June 3, 2010. Proceedings*. Ed. by H. Gilbert. Vol. 6110. Lecture Notes in Computer Science. Springer, 2010, pp. 553–572.

[ATS+09] M. H. Au et al. "Dynamic Universal Accumulators for DDH Groups and Their Application to Attribute-Based Anonymous Credential Systems". In: *Topics in Cryptology - CT-RSA 2009, The Cryptographers' Track at the RSA Conference 2009, San Francisco, CA, USA, April 20-24, 2009. Proceedings*. Ed. by M. Fischlin. Vol. 5473. Lecture Notes in Computer Science. Springer, 2009, pp. 295–308.

[BBC+24] C. Baum et al. *Cryptographers' Feedback on the EU Digital Identity's ARF*. https://github.com/user-attachments/files/15904122/cryptographers-feedback.pdf. 2024.

# Reference II


[BCD+17 ] F. Baldimtsi et al. “Accumulators with Applications to Anonymity-Preserving Revocation”. In: *2017 IEEE European Symposium on Security and Privacy, EuroS&P 2017, Paris, France, April 26-28, 2017*. IEEE, 2017, pp. 301–315.

[BGG+14 ] D. Boneh et al. “Fully Key-Homomorphic Encryption, Arithmetic Circuit ABE and Compact Garbled Circuits”. In: *Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings*. Ed. by P. Q. Nguyen and E. Oswald. Vol. 8441. Lecture Notes in Computer Science. Springer, 2014, pp. 533–556.

[BS23 ] W. Beullens and G. Seiler. “LaBRADOR: Compact Proofs for R1CS from Module-SIS”. In: *CRYPTO (5)*. Vol. 14085. Lecture Notes in Computer Science. Springer, 2023, pp. 518–548.

# Reference III


[CKS09] J. Camenisch, M. Kohlweiss, and C. Soriente. "An Accumulator Based on Bilinear Maps and Efficient Revocation for Anonymous Credentials". In: *Public Key Cryptography - PKC 2009, 12th International Conference on Practice and Theory in Public Key Cryptography, Irvine, CA, USA, March 18-20, 2009. Proceedings*. Ed. by S. Jarecki and G. Tsudik. Vol. 5443. Lecture Notes in Computer Science. Springer, 2009, pp. 481–500.

[CL02 a] J. Camenisch and A. Lysyanskaya. "A Signature Scheme with Efficient Protocols". In: *SCN*. Vol. 2576. Lecture Notes in Computer Science. Springer, 2002, pp. 268–289.

[CL02 b] J. Camenisch and A. Lysyanskaya. "Dynamic Accumulators and Application to Efficient Revocation of Anonymous Credentials". In: *Advances in Cryptology - CRYPTO 2002, 22nd Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 2002, Proceedings*. Ed. by M. Yung. Vol. 2442. Lecture Notes in Computer Science. Springer, 2002, pp. 61–76.

# Reference IV


[CP23 ] L. de Castro and C. Peikert. “Functional Commitments for All Functions, with Transparent Setup and from SIS”. In: *Advances in Cryptology – EUROCRYPT 2023: 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Lyon, France, April 23-27, 2023, Proceedings, Part III*. Lyon, France: Springer-Verlag, 2023, pp. 287–320. ISBN: 978-3-031-30619-8.

[DHS15 ] D. Derler, C. Hanser, and D. Slamanig. “Revisiting Cryptographic Accumulators, Additional Properties and Relations to Other Primitives”. In: *Topics in Cryptology —— CT-RSA 2015*. Ed. by K. Nyberg. Cham: Springer International Publishing, 2015, pp. 127–144. ISBN: 978-3-319-16715-2.

[ENS20 ] M. F. Esgin, N. K. Nguyen, and G. Seiler. “Practical Exact Proofs from Lattices: New Techniques to Exploit Fully-Splitting Rings”. In: *Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proceedings, Part II*. Ed. by S. Moriai and H. Wang. Vol. 12492. Lecture Notes in Computer Science. Springer, 2020, pp. 259–288.

# Reference V


[GSW13] C. Gentry, A. Sahai, and B. Waters. "Homomorphic Encryption from Learning with Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based". In: *Advances in Cryptology – CRYPTO 2013*. Ed. by R. Canetti and J. A. Garay. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 75–92. ISBN: 978-3-642-40041-4.

[JML24] S. Jaques, H. Montgomery, and M. Lodder. "ALLOSAUR: Accumulator with Low-Latency Oblivious Sublinear Anonymous credential Updates with Revocations". In: *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security, ASIA CCS 2024, Singapore, July 1-5, 2024*. Ed. by J. Zhou et al. ACM, 2024.

[KB21] I. Karantaidou and F. Baldimtsi. "Efficient Constructions of Pairing Based Accumulators". In: *34th IEEE Computer Security Foundations Symposium, CSF 2021, Dubrovnik, Croatia, June 21-25, 2021*. IEEE, 2021, pp. 1–16.

# Reference VI


[KL24] V. Y. Kemmoe and A. Lysyanskaya. "RSA-Based Dynamic Accumulator without Hashing into Primes". In: *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security, CCS 2024, Salt Lake City, UT, USA, October 14-18, 2024*. Ed. by B. Luo et al. ACM, 2024, pp. 4271–4285.

[LLN+23] B. Libert et al. "Zero-Knowledge Arguments for Lattice-Based Accumulators: Logarithmic-Size Ring Signatures and Group Signatures Without Trapdoors". In: *J. Cryptol.* 36.3 (2023), p. 23.

[LLX07] J. Li, N. Li, and R. Xue. "Universal Accumulators with Efficient Nonmembership Proofs". In: *Applied Cryptography and Network Security, 5th International Conference, ACNS 2007, Zhuhai, China, June 5-8, 2007, Proceedings*. Ed. by J. Katz and M. Yung. Vol. 4521. Lecture Notes in Computer Science. Springer, 2007, pp. 253–269.

[LNP+21] V. Lyubashevsky et al. "Shorter Lattice-Based Group Signatures via "Almost Free" Encryption and Other Optimizations". In: *ASIACRYPT (4)*. Vol. 13093. Lecture Notes in Computer Science. Springer, 2021, pp. 218–248.

# Reference VII


[LNP22] V. Lyubashevsky, N. K. Nguyen, and M. Plançon. "Lattice-Based Zero-Knowledge Proofs and Applications: Shorter, Simpler, and More General". In: *Advances in Cryptology – CRYPTO 2022: 42nd Annual International Cryptology Conference, CRYPTO 2022, Santa Barbara, CA, USA, August 15–18, 2022, Proceedings, Part II*. Santa Barbara, CA, USA: Springer-Verlag, 2022, pp. 71–101. ISBN: 978-3-031-15978-7.

[Lyu12] V. Lyubashevsky. "Lattice Signatures without Trapdoors". In: *Advances in Cryptology - EUROCRYPT 2012 - 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15-19, 2012. Proceedings*. Ed. by D. Pointcheval and T. Johansson. Vol. 7237. Lecture Notes in Computer Science. Springer, 2012, pp. 738–755.

[MP12] D. Micciancio and C. Peikert. "Trapdoors for lattices: simpler, tighter, faster, smaller". In: *Proceedings of the 31st Annual International Conference on Theory and Applications of Cryptographic Techniques*. EUROCRYPT'12. Cambridge, UK: Springer-Verlag, 2012, pp. 700–718. ISBN: 9783642290107.

# Reference VIII


[Ngu05] L. Nguyen. “Accumulators from Bilinear Pairings and Applications”. In: *Topics in Cryptology - CT-RSA 2005, The Cryptographers’ Track at the RSA Conference 2005, San Francisco, CA, USA, February 14-18, 2005, Proceedings*. Ed. by A. Menezes. Vol. 3376. Lecture Notes in Computer Science. Springer, 2005, pp. 275–292.

[PST+13] C. Papamanthou et al. “Streaming Authenticated Data Structures”. In: *Advances in Cryptology - EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings*. Ed. by T. Johansson and P. Q. Nguyen. Vol. 7881. Lecture Notes in Computer Science. Springer, 2013, pp. 353–370.

[WW23] H. Wee and D. J. Wu. “Lattice-Based Functional Commitments: Fast Verification and Cryptanalysis”. In: *Advances in Cryptology – ASIACRYPT 2023*. Ed. by J. Guo and R. Steinfeld. Singapore: Springer Nature Singapore, 2023, pp. 201–235. ISBN: 978-981-99-8733-7.

# Reference IX


[YAY+18 ] Z. Yu et al. “Lattice-Based Universal Accumulator with Nonmembership Arguments”. In: *Information Security and Privacy - 23rd Australasian Conference, ACISP 2018, Wollongong, NSW, Australia, July 11-13, 2018, Proceedings*. Ed. by W. Susilo and G. Yang. Vol. 10946. Lecture Notes in Computer Science. Springer, 2018, pp. 502–519.

[ZYH24 ] Y. Zhao, S. Yang, and X. Huang. “Lattice-based dynamic universal accumulator: Design and application”. In: *Comput. Stand. Interfaces* 89 (2024), p. 103807.