Quasi-Adaptive NIZK for Linear Subspaces Revisited

Eike Kiltz (**RUB**)

Hoeteck Wee (**ENS**)

non-interactive zero knowledge

[Blum Feldman Micali 88, Groth Sahai 08]

... algebraic relations in a group **impact** → pairing-based crypto

non-interactive zero knowledge

[Blum Feldman Micali 88, Groth Sahai 08]

... algebraic relations in a group **impact** → pairing-based crypto

i. **better** efficiency

– group signatures [**G06, G07**]

non-interactive zero knowledge

[Blum Feldman Micali 88, Groth Sahai 08]

... algebraic relations in a group **impact** → pairing-based crypto

i. **better** efficiency

– group signatures [**G06, G07**]

ii. **stronger** security

– anonymous credentials [**BCLK08, BCCKLS09, F11**]

non-interactive zero knowledge

[Blum Feldman Micali 88, Groth Sahai 08]

... algebraic relations in a group **impact** → pairing-based crypto

i. **better** efficiency

– group signatures [**G06, G07**]

ii. **stronger** security

– anonymous credentials [**BCLK08, BCCKLS09, F11**]

iii. **less** interaction

– two-party protocols [**KV11, FLM11**]

non-interactive zero knowledge

[Blum Feldman Micali 88, Groth Sahai 08]

- ... linear subspaces over a group

quasi-adaptive nizk

[Jutla Roy 13]

... linear subspaces over a group, **where**

- crs may depend on linear subspace
- computational soundness (but adaptive)

quasi-adaptive nizk

[Jutla Roy 13]

... linear subspaces over a group, **where**

– crs may depend on linear subspace

– computational soundness (but adaptive)

constant-size proofs [**Libert Peters Joye Yung 14, Jutla Roy 14, Abdalla Benhamouda Pointcheval 15**]

– independent of dimensions of subspace

quasi-adaptive nizk

[Jutla Roy 13]

... linear subspaces over a group, **where**

– crs may depend on linear subspace

– computational soundness (but adaptive)

constant-size proofs [**Libert Peters Joye Yung 14, Jutla Roy 14, Abdalla Benhamouda Pointcheval 15**]

– independent of dimensions of subspace

– extensions to simulation-soundness (OTSS, USS)

quasi-adaptive nizk

[Jutla Roy 13]

... linear subspaces over a group, **where**

– crs may depend on linear subspace

– computational soundness (but adaptive)

constant-size proofs [**Libert Peters Joye Yung 14, Jutla Roy 14, Abdalla Benhamouda Pointcheval 15**]

applications. shorter kdm-cca encryption, cca-secure IBE, ...

this work

better identity-based encryption (IBE)

better quasi-adaptive nizk

this work

1. simpler + improved quasi-adaptive nizk for linear subspaces

$$(\mathbf{xM})\mathbf{K} \quad = \mathbf{x}(\mathbf{MK})$$

this work

1. simpler + improved quasi-adaptive nizk for linear subspaces

$$(\mathbf{xM})\mathbf{KA} = \mathbf{x}(\mathbf{MK})\mathbf{A}$$

this work

1. simpler + improved quasi-adaptive nizk for linear subspaces

soundness.	AS^{-}	AS	OTSS	USS
[LPJY14]		3	4	20
[JR14]	2			
[ABP15]	2	3	3	

this work

1. simpler + improved quasi-adaptive nizk for linear subspaces

soundness.	AS^{-}	AS	OTSS	USS
[LPJY14]		3	4	20
[JR14]	2			
[ABP15]	2	3	3	
this	2	3	3	6

this work

1. simpler + improved quasi-adaptive nizk for linear subspaces

soundness.	AS^{-}	AS	OTSS	USS
[LPJY14]		3	4	20
[JR14]	1			
[ABP15]	1	2	2	
this	1	2	2	4

this work

1. simpler + improved quasi-adaptive nizk for linear subspaces

soundness.	AS^{-}	AS	OTSS	USS
[LPJY14]		3	4	20
[JR14]	1			
[ABP15]	1	2	2	
this	1	2	2	4

better assumptions / **shorter** parameters / **fewer** pairings

this work

1. simpler + improved quasi-adaptive nizk for linear subspaces

soundness.	AS^{-}	AS	OTSS	USS
[LPJY14]		3	4	20
[JR14]	1			
[ABP15]	1	2	2	
this	1	2	2	4

better assumptions / **shorter** parameters / **fewer** pairings

2. ... linearly homomorphic structure-preserving signatures **[LPJY13]**

this work

private-verifier nizk

(information-theoretic security)

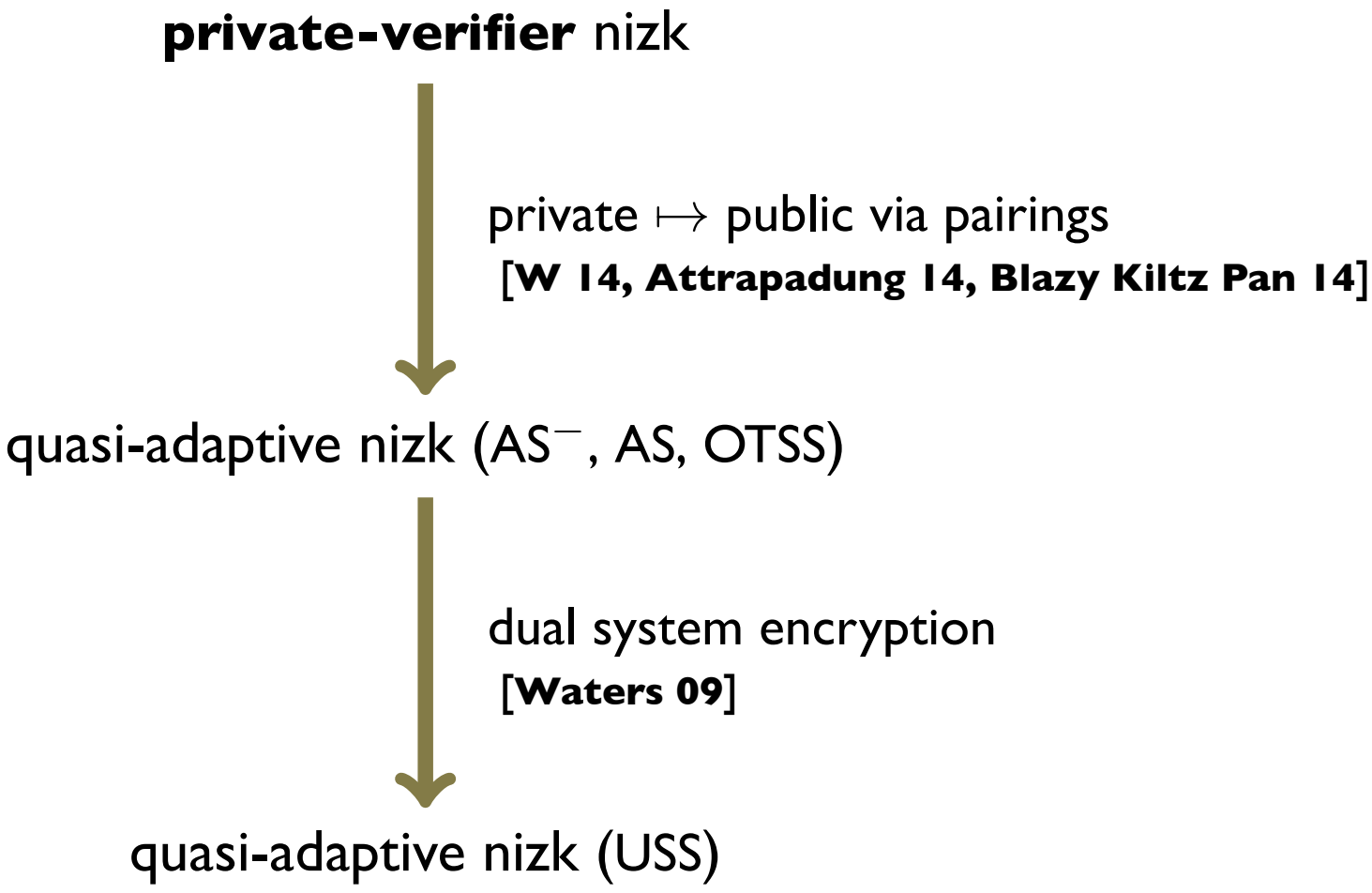
this work

private-verifier nizk

private $\mapsto$ public via pairings
[**W 14, Attrapadung 14, Blazy Kiltz Pan 14**]

quasi-adaptive nizk (AS^{-}, AS, OTSS)

this work


private-verifier nizk
private $\mapsto$ public via pairings
[**W 14, Attrapadung 14, Blazy Kiltz Pan 14**]
quasi-adaptive nizk (AS^{-}, AS, OTSS)
dual system encryption
[**Waters 09**]
quasi-adaptive nizk (USS)

bilinear groups

def. $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ cyclic groups of prime order q

– $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T, \quad e(g_1^x, g_2^y) = e(g_1, g_2)^{xy}$

bilinear groups

def. $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ cyclic groups of prime order q

– $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T, \quad e([x]_1, [y]_2) = [xy]_T$

bilinear groups

def. $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ cyclic groups of prime order q

– $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T, \quad e([\mathbf{X}]_1, [\mathbf{Y}]_2) = [\mathbf{XY}]_T$

bilinear groups

def. $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ cyclic groups of prime order q

– $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T, \quad e([\mathbf{X}]_1, [\mathbf{Y}]_2) = [\mathbf{XY}]_T$

matrix assumptions [EHKRV13, MRV15]:

– diffie-hellman assumptions (e.g. SXDH, DLIN, k-LIN)

$$\left(\boxed{\mathbf{A}}, \boxed{\mathbf{A}}\boxed{\mathbf{s}} \right) \approx_c \left(\boxed{\mathbf{A}}, \boxed{\$} \right)$$

bilinear groups

def. $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ cyclic groups of prime order q

– $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T, \quad e([\mathbf{X}]_1, [\mathbf{Y}]_2) = [\mathbf{XY}]_T$

matrix assumptions [EHKRV13, MRV15]:

– diffie-hellman assumptions (e.g. SXDH, DLIN, k-LIN)

$$\left(\boxed{\mathbf{A}}, \boxed{\mathbf{A}}\boxed{\mathbf{s}} \right) \approx_c \left(\boxed{\mathbf{A}}, \boxed{\$} \right)$$

e.g. $\mathbf{A} = \begin{pmatrix} 1 \\ a_1 \end{pmatrix}$; $\mathbf{A} = \begin{pmatrix} 1 & 1 \\ a_1 & 0 \\ 0 & a_2 \end{pmatrix}$

bilinear groups

def. $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ cyclic groups of prime order q

– $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T, \quad e([\mathbf{X}]_1, [\mathbf{Y}]_2) = [\mathbf{XY}]_T$

matrix assumptions [EHKRV13, MRV15]:

– diffie-hellman assumptions (e.g. SXDH, DLIN, k-LIN)

$$\left(\boxed{\mathbf{A}}, \boxed{\mathbf{A}}\boxed{\mathbf{s}} \right) \approx_c \left(\boxed{\mathbf{A}}, \boxed{\$} \right)$$

– kernel assumption (e.g. SDP)

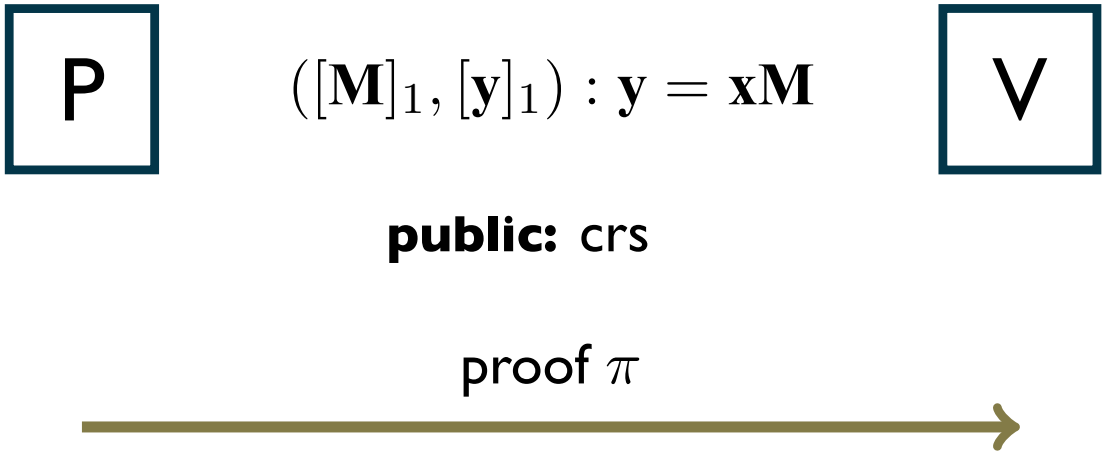
given $[\mathbf{A}]_2$, hard to find $[\mathbf{z}]_1 \neq [\mathbf{0}]_1$ s.t. $\mathbf{zA} = \mathbf{0}$

$$\boxed{\mathbf{z}}\quad\boxed{\mathbf{A}}$$

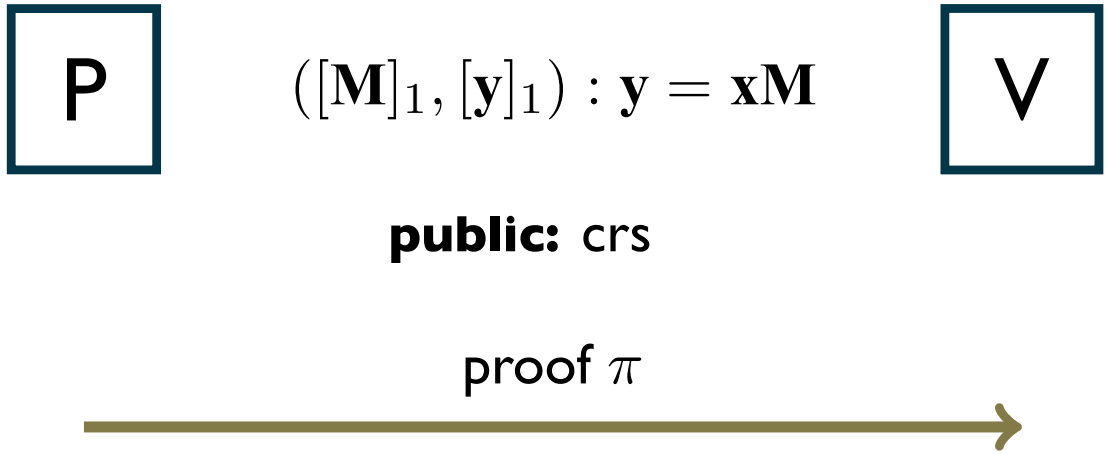
non-interactive zero knowledge

$$([\mathbf{M}]_1, [\mathbf{y}]_1) : \mathbf{y} = \mathbf{x}\mathbf{M}$$

non-interactive zero knowledge

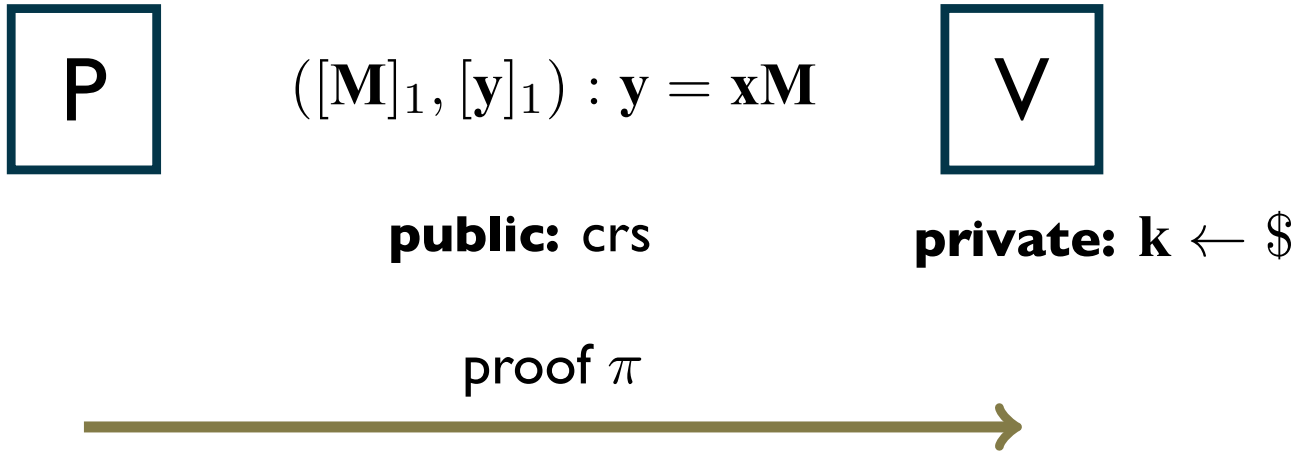


non-interactive zero knowledge



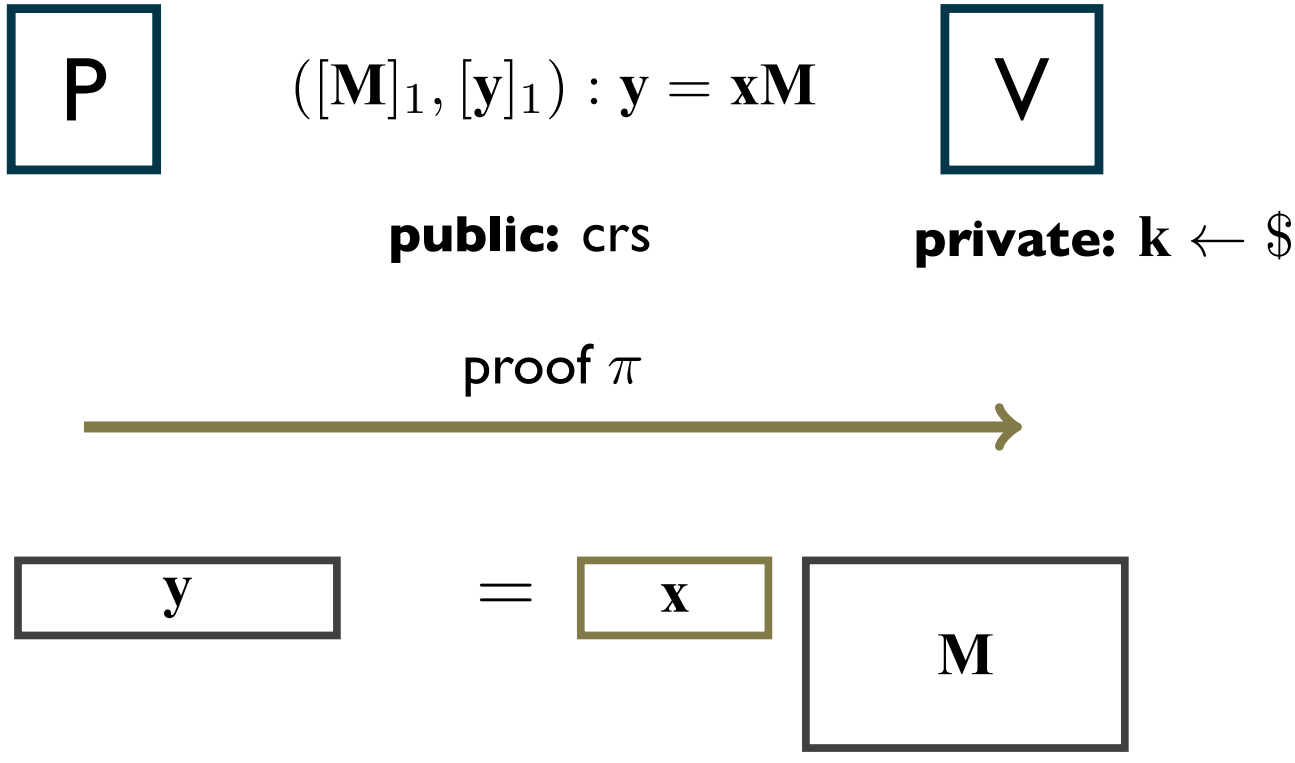
- **completeness.** if $\mathbf{y} = \mathbf{xM}$, then V accepts π
- **soundness.** if $\mathbf{y} \notin \text{span}(\mathbf{M})$, then V rejects w.h.p.
- **zero knowledge.** (crs, π) $\equiv$ simulator$([\mathbf{M}]_1, [\mathbf{y}]_1)$

non-interactive zero knowledge

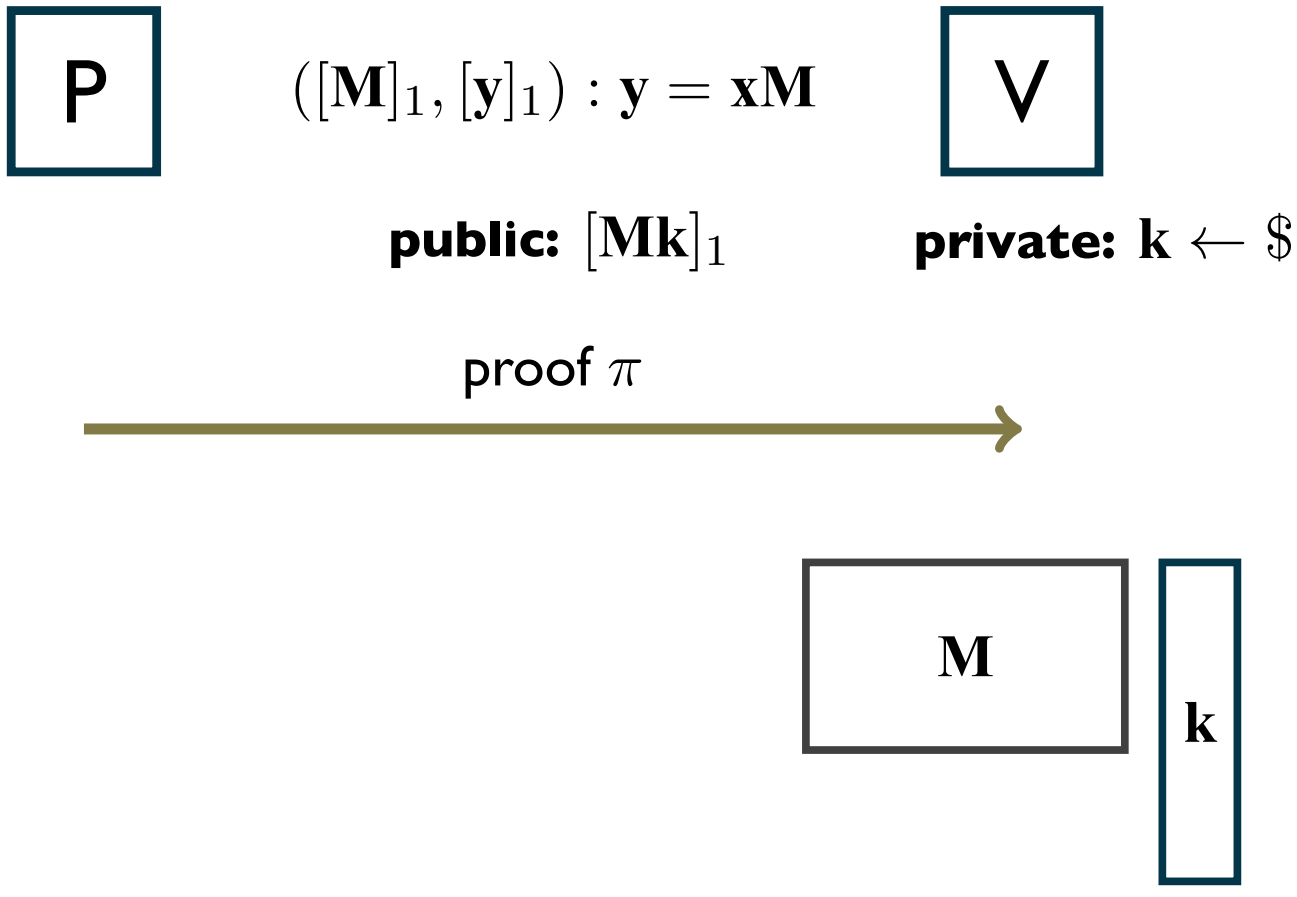


- **completeness.** if $\mathbf{y} = \mathbf{xM}$, then V accepts π
- **soundness.** if $\mathbf{y} \notin \text{span}(\mathbf{M})$, then V rejects w.h.p.
- **zero knowledge.** (crs, π) $\equiv$ simulator$([\mathbf{M}]_1, [\mathbf{y}]_1)$

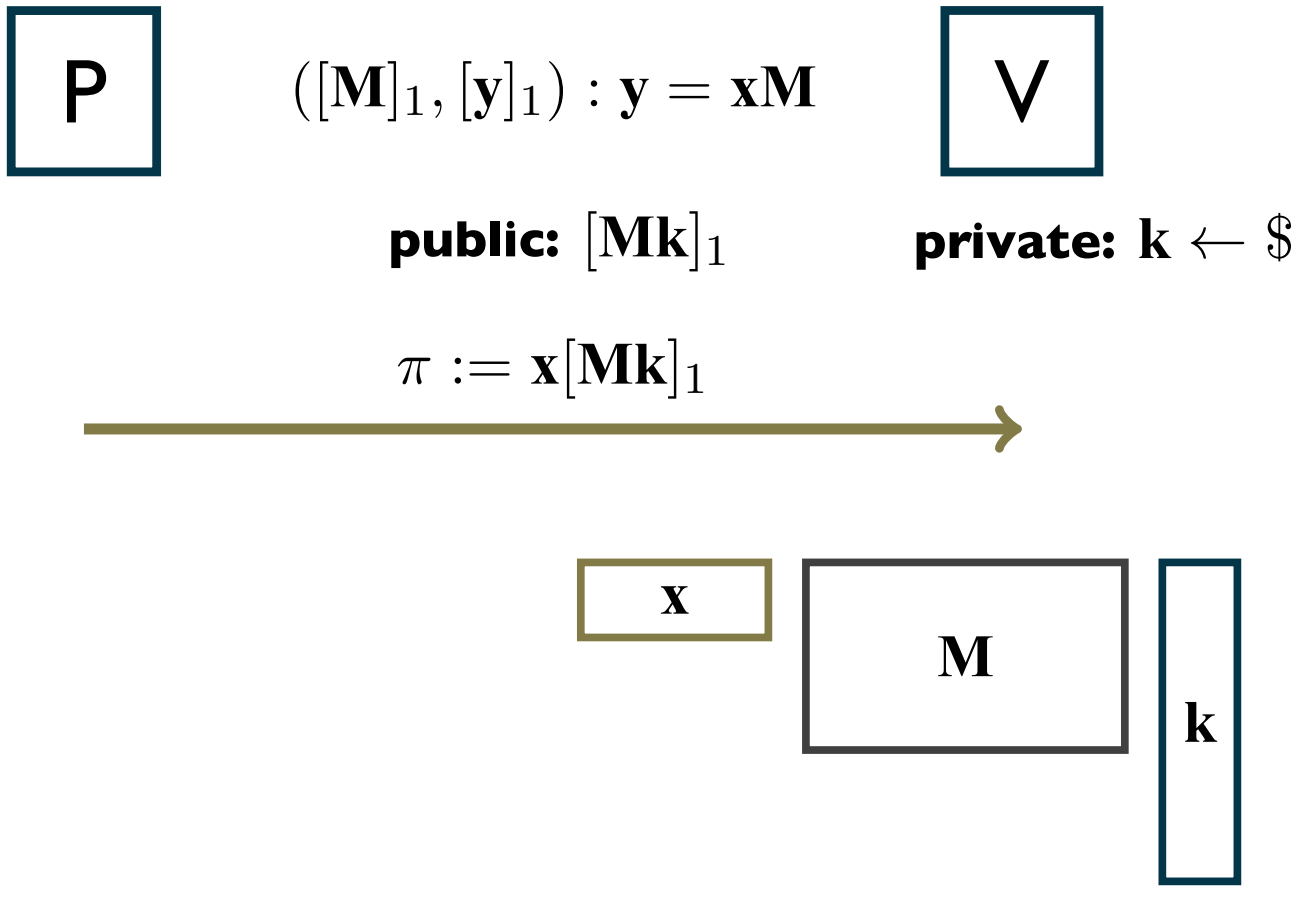
our construction



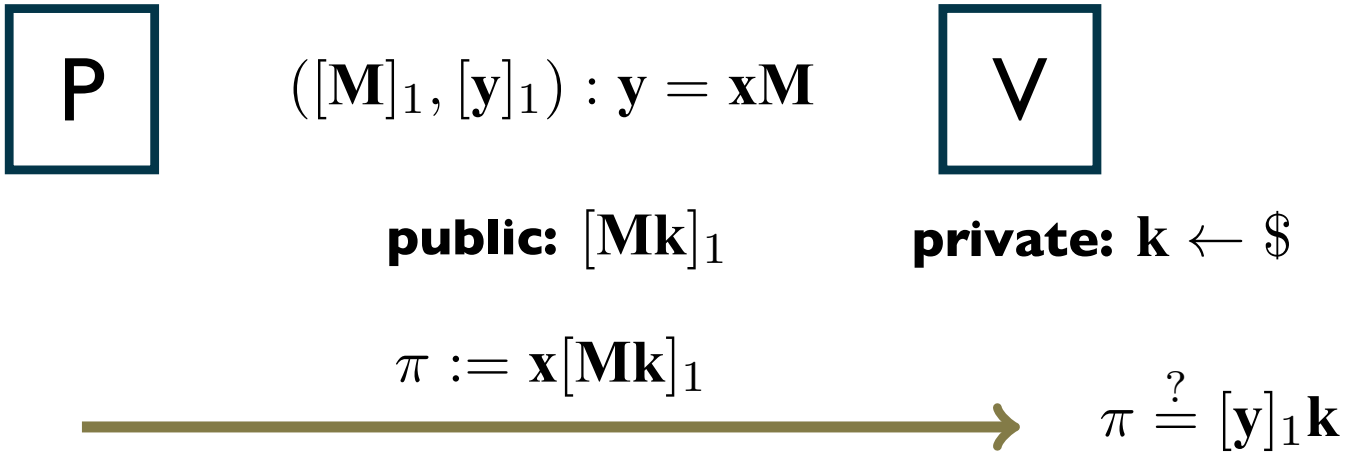
our construction

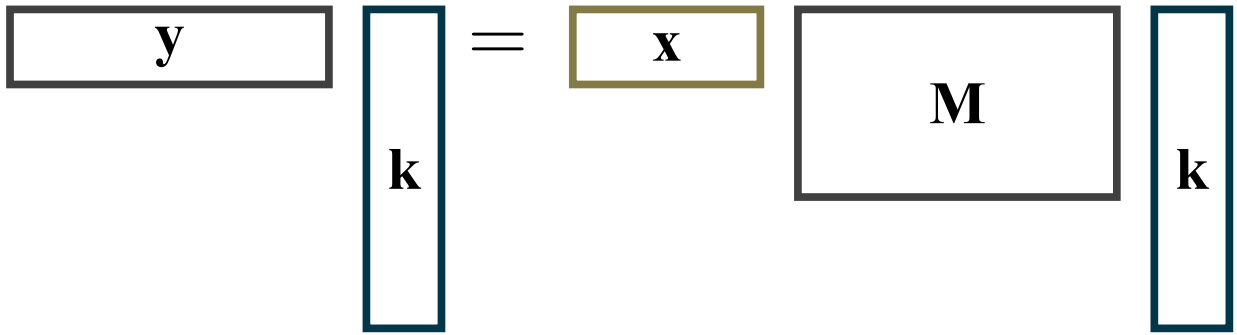


our construction



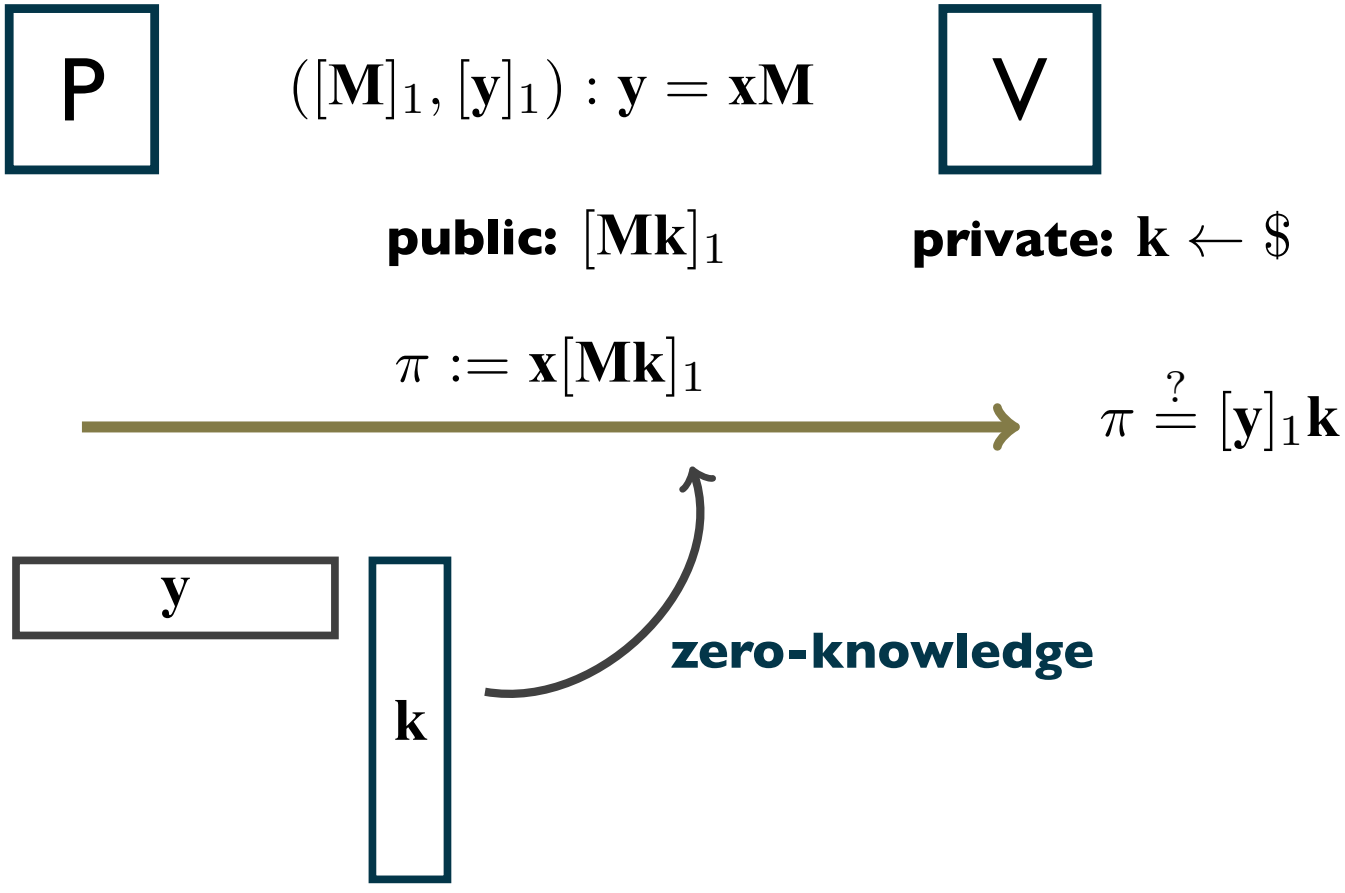
our construction



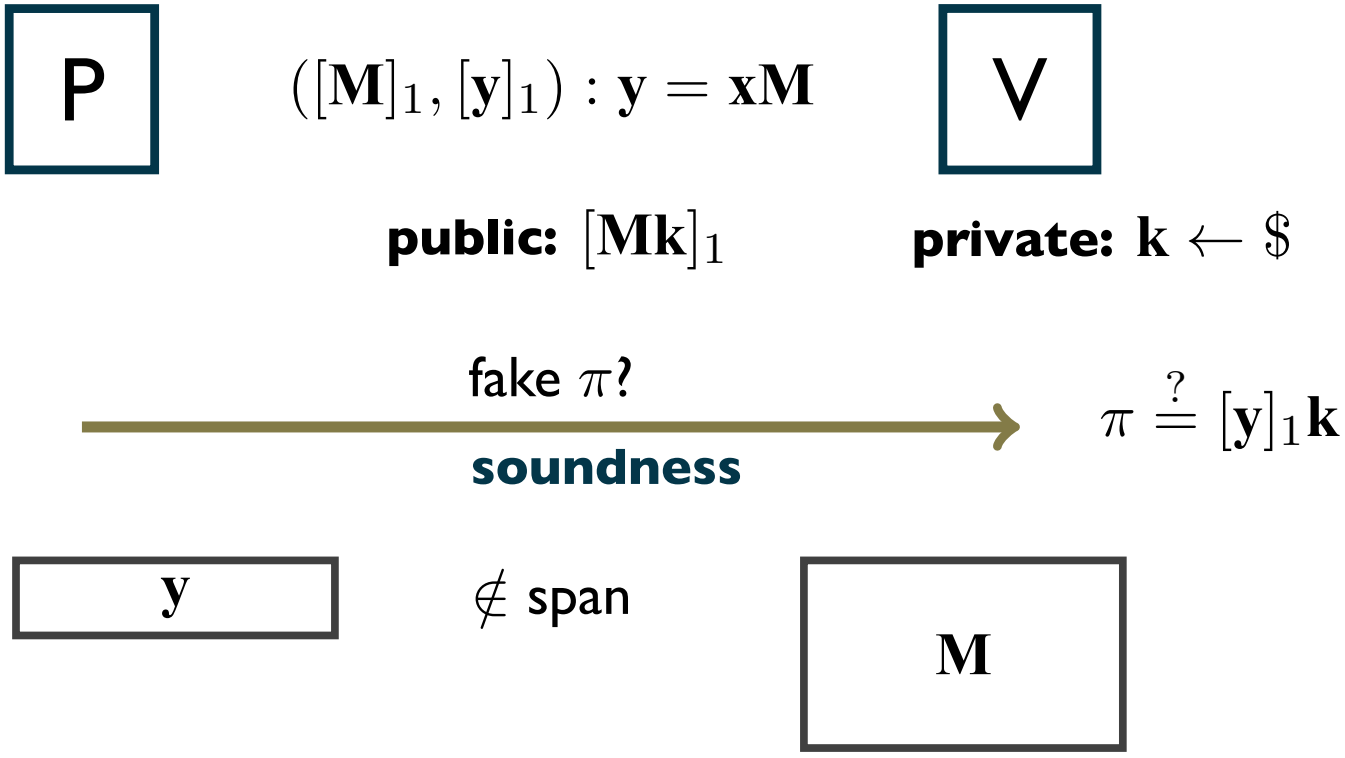


completeness. $(\mathbf{xM})\mathbf{k} = \mathbf{x}(\mathbf{Mk})$

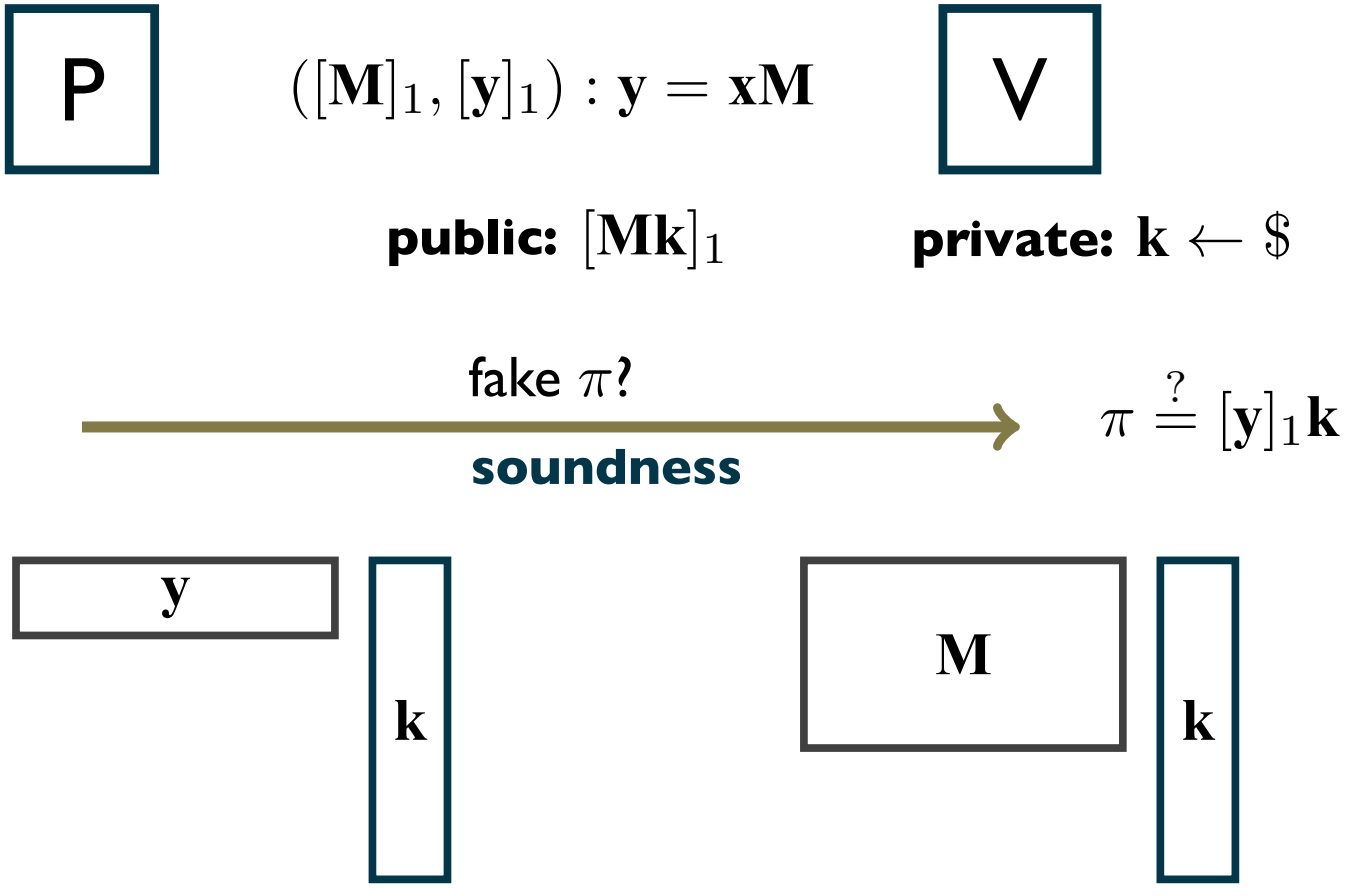
our construction



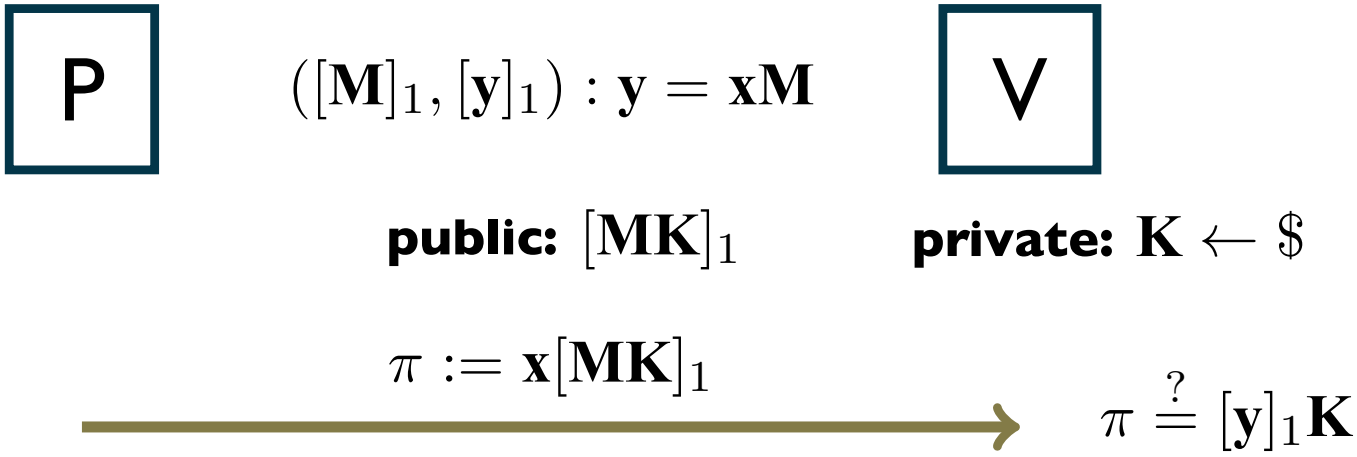
our construction



our construction



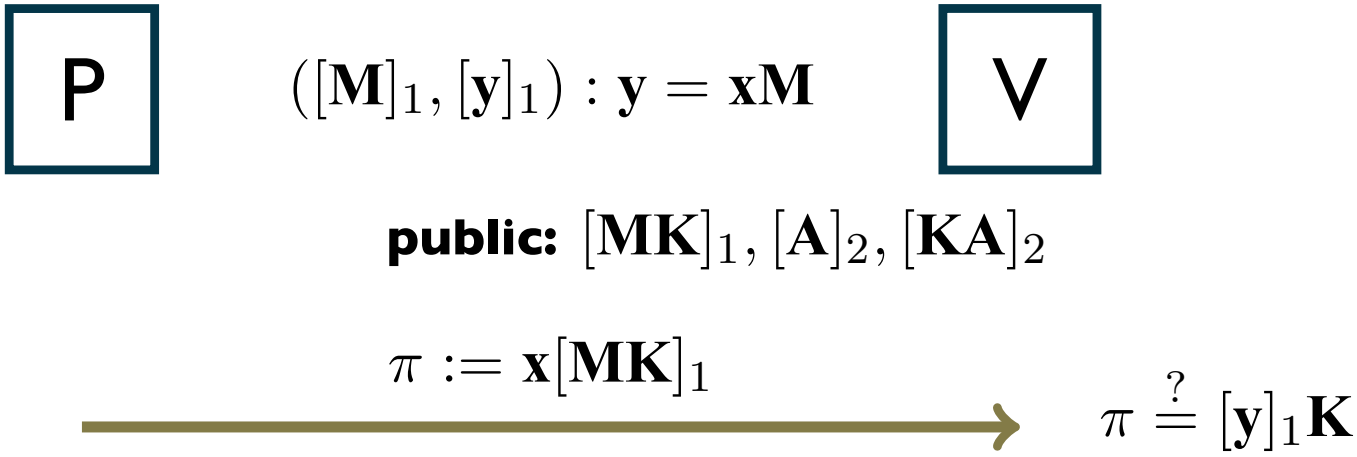
our construction



private $\mapsto$ public

(1) $\mathbf{k} \mapsto \mathbf{K}$

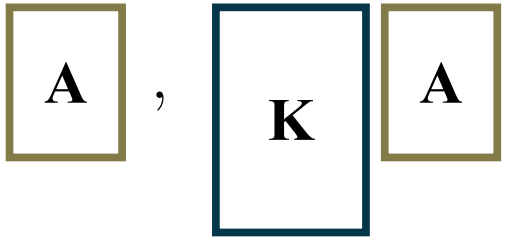
our construction



private $\mapsto$ public

(1) $\mathbf{k} \mapsto \mathbf{K}$

(2) publish $([\mathbf{A}]_2, [\mathbf{KA}]_2)$ in crs



our construction

$([\mathbf{M}]_1, [\mathbf{y}]_1) : \mathbf{y} = \mathbf{xM}$

public: $[\mathbf{MK}]_1, [\mathbf{A}]_2, [\mathbf{KA}]_2$

$\pi := \mathbf{x}[\mathbf{MK}]_1$

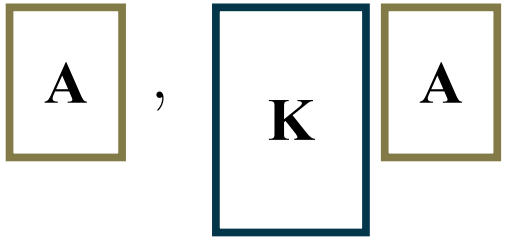
$\pi \stackrel{?}{=} [\mathbf{y}]_1\mathbf{K}$

$\pi \quad \mathbf{A} \quad \stackrel{?}{=} \quad [\mathbf{y}]_1 \quad \mathbf{KA}$

private $\mapsto$ public

(1) $\mathbf{k} \mapsto \mathbf{K}$

(2) publish $([\mathbf{A}]_2, [\mathbf{KA}]_2)$ in crs



our construction

$([\mathbf{M}]_1, [\mathbf{y}]_1) : \mathbf{y} = \mathbf{xM}$

public: $[\mathbf{MK}]_1, [\mathbf{A}]_2, [\mathbf{KA}]_2$

$\pi := \mathbf{x}[\mathbf{MK}]_1$

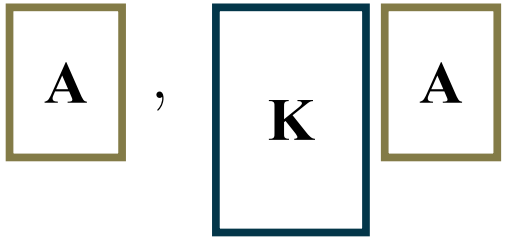
$\pi \stackrel{?}{=} [\mathbf{y}]_1\mathbf{K}$

$e(\pi, [\mathbf{A}]_2) \stackrel{?}{=} e([\mathbf{y}]_1, [\mathbf{KA}]_2)$

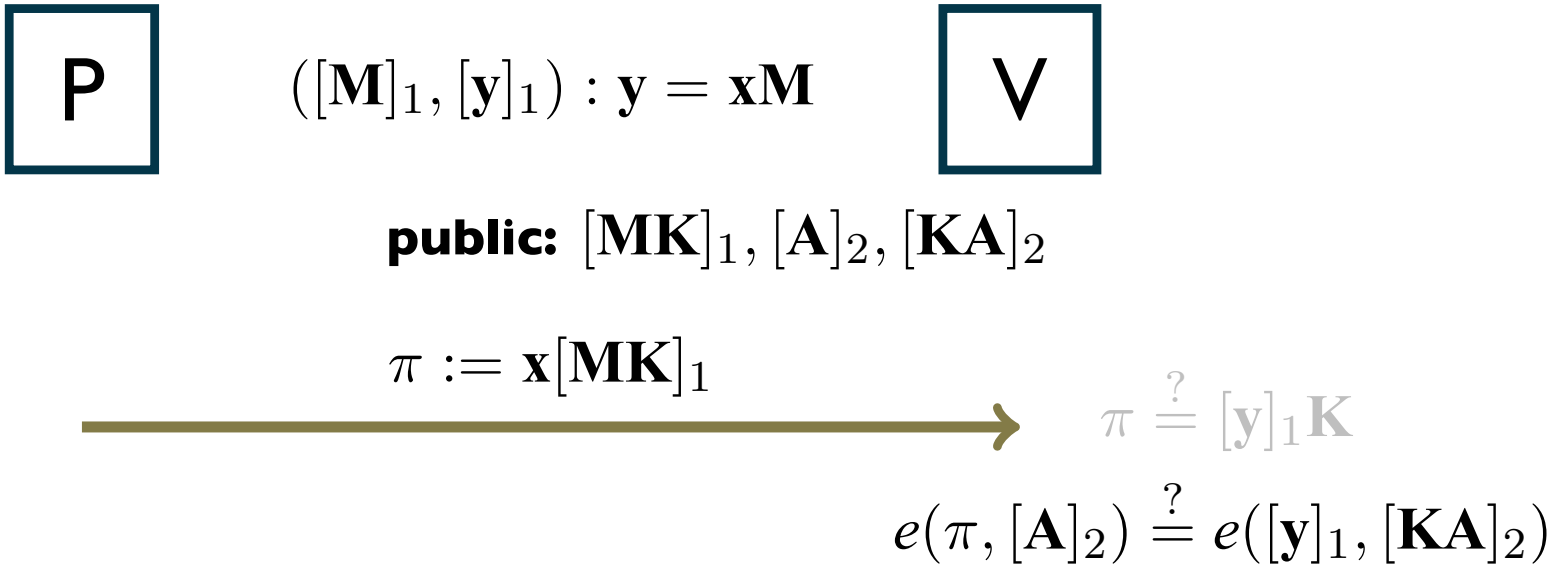
private $\mapsto$ public

(1) $\mathbf{k} \mapsto \mathbf{K}$

(2) publish $([\mathbf{A}]_2, [\mathbf{KA}]_2)$ in crs



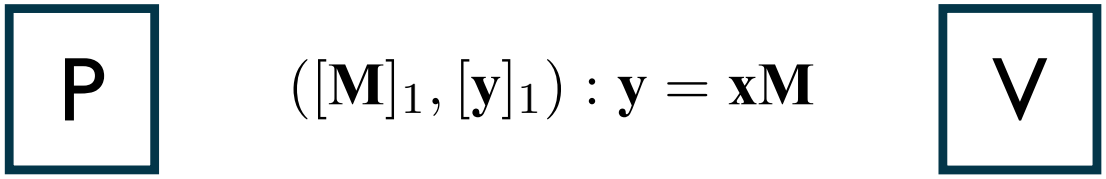
our construction



completeness. $\mathbf{x}(\mathbf{MK})\mathbf{A} = (\mathbf{xM})(\mathbf{KA})$

zero-knowledge. $\pi = [\mathbf{y}]_1\mathbf{K}$

our construction



public: $[\mathbf{MK}]_1, [\mathbf{A}]_2, [\mathbf{KA}]_2$

$\pi := \mathbf{x}[\mathbf{MK}]_1$

$\pi \stackrel{?}{=} [\mathbf{y}]_1\mathbf{K}$

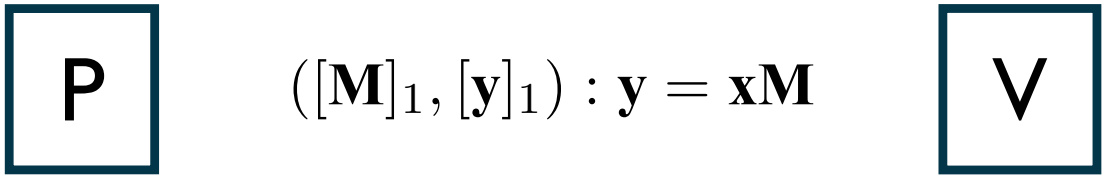
$e(\pi, [\mathbf{A}]_2) \stackrel{?}{=} e([\mathbf{y}]_1, [\mathbf{KA}]_2)$

soundness.

cheating proof $([\mathbf{y}]_1, \pi)$ for public-verifier

$\Rightarrow$ cheating proof $([\mathbf{y}]_1, \pi)$ for private-verifier

our construction



public: $[\mathbf{MK}]_1, [\mathbf{A}]_2, [\mathbf{KA}]_2$

$\pi := \mathbf{x}[\mathbf{MK}]_1$

$\pi \stackrel{?}{=} [\mathbf{y}]_1\mathbf{K}$

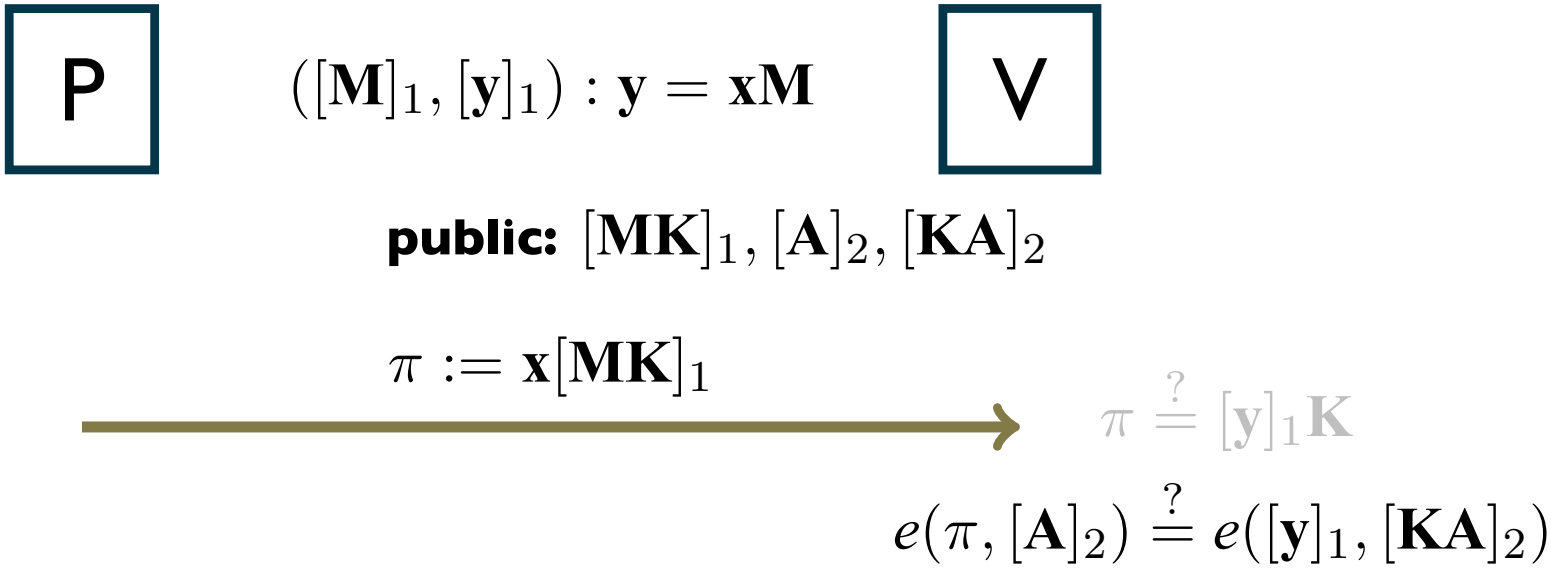
$e(\pi, [\mathbf{A}]_2) \stackrel{?}{=} e([\mathbf{y}]_1, [\mathbf{KA}]_2)$

soundness.

$e(\pi, [\mathbf{A}]_2) = e([\mathbf{y}]_1, [\mathbf{KA}]_2)$

$\Rightarrow \pi = [\mathbf{y}]_1\mathbf{K}$

our construction



soundness.

$$\pi \quad \mathbf{A} \quad = \quad [\mathbf{y}]_1 \quad \mathbf{KA}$$

$$\Rightarrow \pi = [\mathbf{y}]_1\mathbf{K}$$

our construction

$([\mathbf{M}]_1, [\mathbf{y}]_1) : \mathbf{y} = \mathbf{xM}$

public: $[\mathbf{MK}]_1, [\mathbf{A}]_2, [\mathbf{KA}]_2$

$\pi := \mathbf{x}[\mathbf{MK}]_1$

$\pi \stackrel{?}{=} [\mathbf{y}]_1\mathbf{K}$

$e(\pi, [\mathbf{A}]_2) \stackrel{?}{=} e([\mathbf{y}]_1, [\mathbf{KA}]_2)$

soundness.

$(\pi - [\mathbf{y}]_1\mathbf{K})\mathbf{A} = \mathbf{0}$

$\Rightarrow \pi = [\mathbf{y}]_1\mathbf{K}$

our construction

$([\mathbf{M}]_1, [\mathbf{y}]_1) : \mathbf{y} = \mathbf{xM}$

public: $[\mathbf{MK}]_1, [\mathbf{A}]_2, [\mathbf{KA}]_2$

$\pi := \mathbf{x}[\mathbf{MK}]_1$

$\pi \stackrel{?}{=} [\mathbf{y}]_1\mathbf{K}$

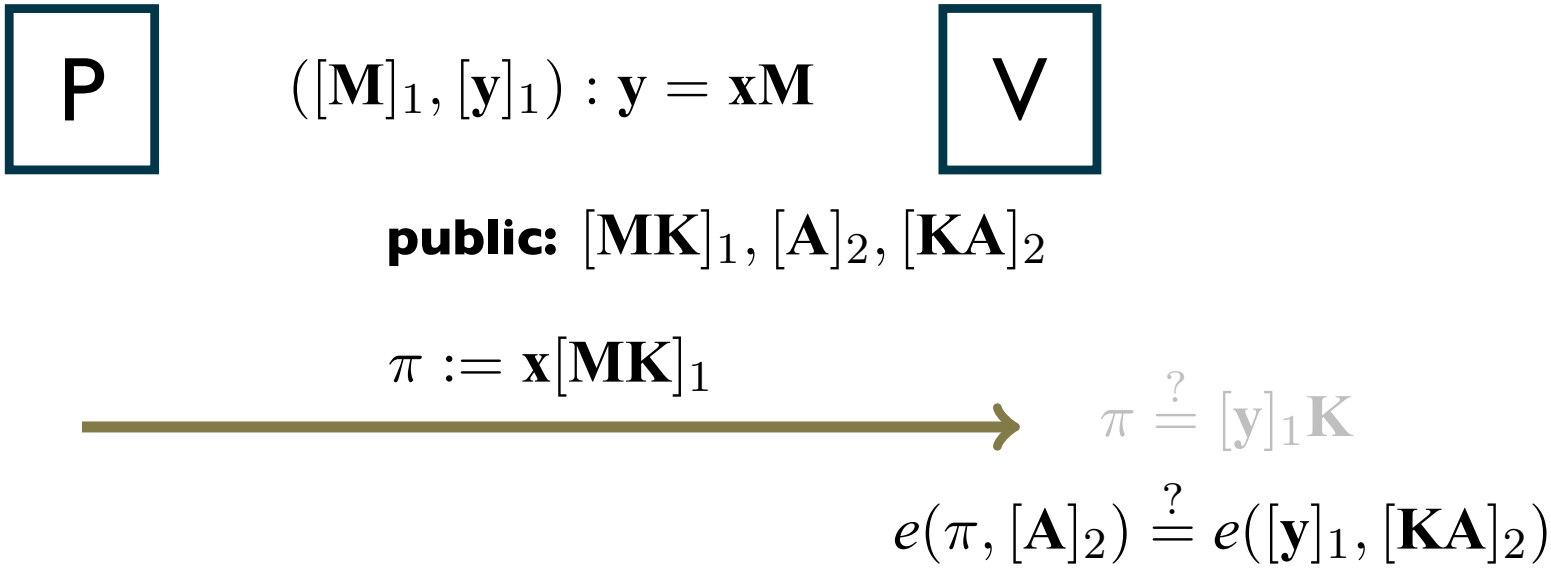
$e(\pi, [\mathbf{A}]_2) \stackrel{?}{=} e([\mathbf{y}]_1, [\mathbf{KA}]_2)$

soundness.

$(\pi - [\mathbf{y}]_1\mathbf{K}) \quad = \mathbf{0}$

$\Rightarrow \pi = [\mathbf{y}]_1\mathbf{K}$

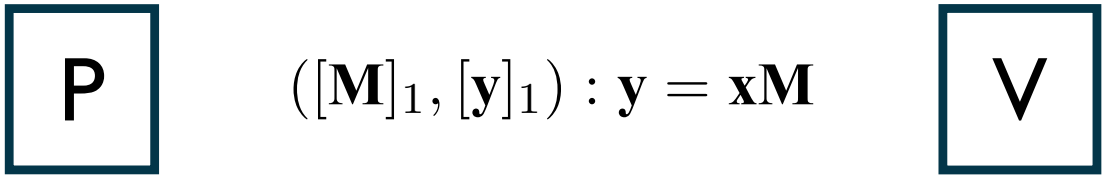
our construction



simulation-soundness.

– one-time. replace $\mathbf{K}$ by $\mathbf{K}_0 + \tau\mathbf{K}_1$ (w.r.t. tag τ)

our construction



public: $[\mathbf{MK}]_1, [\mathbf{A}]_2, [\mathbf{KA}]_2$

$\pi := \mathbf{x}[\mathbf{MK}]_1$

$\pi \stackrel{?}{=} [\mathbf{y}]_1\mathbf{K}$

$e(\pi, [\mathbf{A}]_2) \stackrel{?}{=} e([\mathbf{y}]_1, [\mathbf{KA}]_2)$

simulation-soundness.

– **one-time.** replace $\mathbf{K}$ by $\mathbf{K}_0 + \tau\mathbf{K}_1$ (w.r.t. tag τ)

– **unbounded.** (randomized) PRF from IBE **[BKP14, CGW15]**

conclusion

better identity-based encryption (IBE)

private $\mapsto$ public via pairings

better quasi-adaptive nizk