Adapting Identity-based Encryption with Wildcards to Access Control

Anaïs Barthoulot[1], Sébastien Canard[2], Jacques Traoré[3]

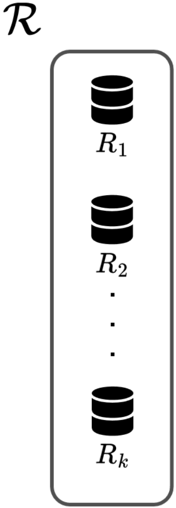
[1]University of Montpellier, LIRMM
[2]Télécom Paris
[3]Orange

Cryptography Seminar IRMAR - IRISA - DGA

4th October 2024

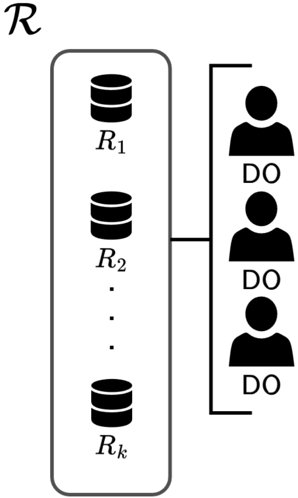
Table of contents

System Architecture



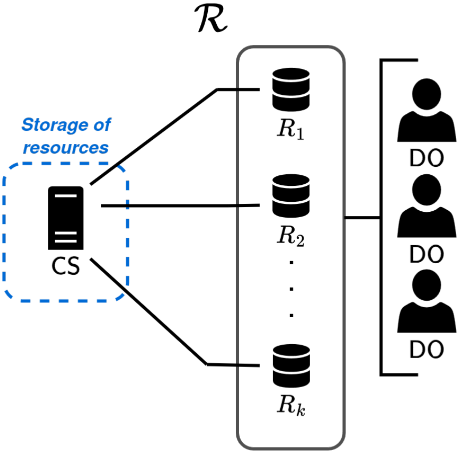
Resources (R): Sensitive data generated by connected objects

System Architecture



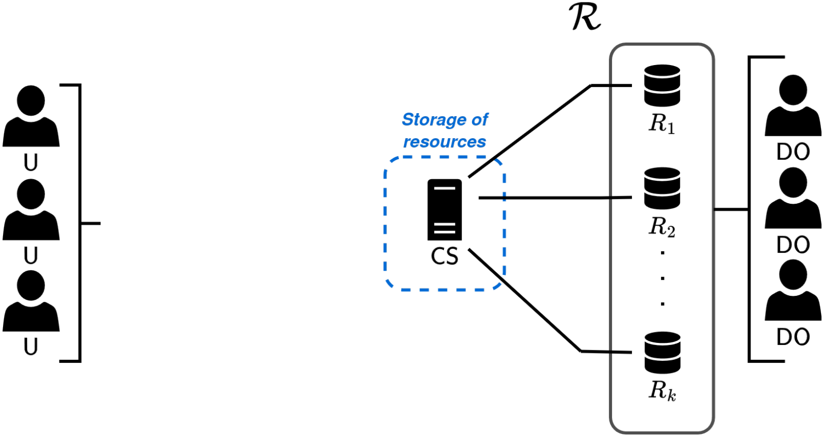
Data Owners (DO): Own and manage access to resources by defining access policies

System Architecture



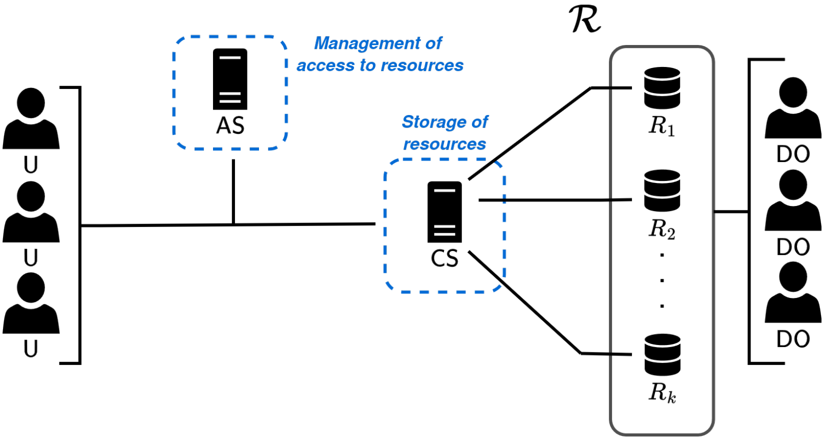
Central Server (CS): Stores the resources and provides access only when a valid token is presented

System Architecture



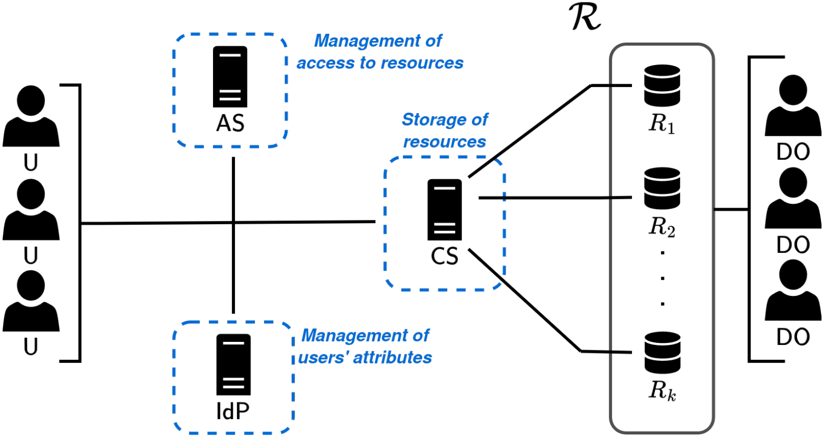
Users (U): Request access to resources and must provide valid tokens

System Architecture



Authorization Server (AS): Issues access tokens based on access policies defined by the Data Owners

System Architecture



Identity Provider (IdP): Manages user attributes to ensure the integrity of the access process

Procedures

- **Setup** (AS, IdP): generate all required parameters and keys

Procedures

- **Setup** (AS, IdP): generate all required parameters and keys
- **StoreResource** (DO, CS, AS): DO stores a new resource R_i on CS and sends associated access policy π_i to AS

Procedures

- **Setup** (AS, IdP): generate all required parameters and keys
- **StoreResource** (DO, CS, AS): DO stores a new resource R_i on CS and sends associated access policy π_i to AS
- **KeyQuery** (U, IdP, AS): U asks AS for a secret key related to her attributes A_{U}; IdP identifies U and verifies her attributes

Procedures

- **Setup** (AS, IdP): generate all required parameters and keys
- **StoreResource** (DO, CS, AS): DO stores a new resource R_i on CS and sends associated access policy π_i to AS
- **KeyQuery** (U, IdP, AS): U asks AS for a secret key related to her attributes A_{U}; IdP identifies U and verifies her attributes
- **ResourceQuery** (U, AS, CS): U queries AS and CS for access to R_i

Security Requirements

- **No unauthorized access:**

  Users without valid attributes must not access the resource R_i

Security Requirements

- **No unauthorized access:**

  Users without valid attributes must not access the resource R_i

- **Attribute protection:**

  Attributes A_{U} must not be revealed to AS or CS, even if they verify π_i

Security Requirements

- **No unauthorized access:**

  Users without valid attributes must not access the resource R_i

- **Attribute protection:**

  Attributes A_{U} must not be revealed to AS or CS, even if they verify π_i

- **Access policy confidentiality:**

  Only AS and DO should know the access policy π_i

Assumptions

- **DO is honest**

Assumptions

- **DO is honest**
- **AS, IdP and CS are honest but curious**

Assumptions

- **DO is honest**
- **AS, IdP and CS are honest but curious**
- **Users are dishonest**

Summary of the knowledge of each actor.

	DO	CS	AS	U	IdP
Resource	x	x		x	
Access policy	x		x		
User's attributes				x	x
User has access		x		x	
Which requested resource		x	x	x	

Table of contents

Patterns

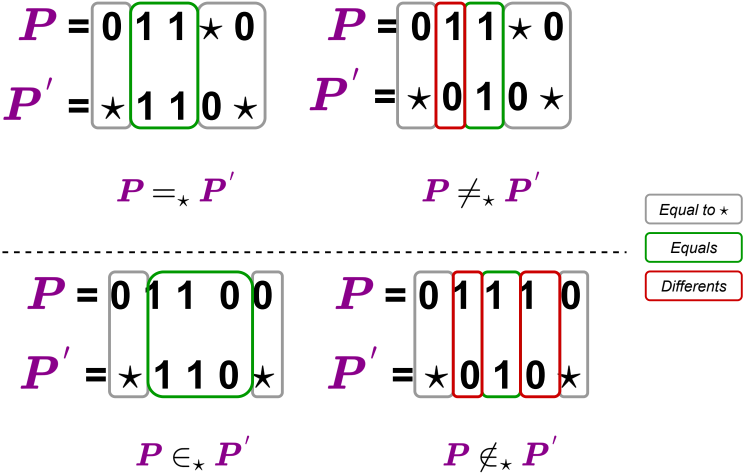
Patterns

- Pattern $\boldsymbol{P} = (P_1, \cdots, P_L) \in \mathcal{U}^L$, where
  - $\mathcal{U}$: set with a special wildcard symbol "$\star$",
  - $L \in \mathbb{N}$

- $\boldsymbol{P}^{'} = (P_1^{'}, \cdots, P_L^{'})$ and $\boldsymbol{P} = (P_1, \cdots, P_L)$:
  - $\boldsymbol{P}$ **belongs** to $\boldsymbol{P}^{'}$, denoted $\boldsymbol{P} \in_\star \boldsymbol{P}^{'}$, iff $\forall i \in \{1, \cdots, L\}$, $(P_i^{'} = P_i) \vee (P_i^{'} = \star)$
  - $\boldsymbol{P}$ **matches** $\boldsymbol{P}^{'}$, denoted $\boldsymbol{P} =_\star \boldsymbol{P}^{'}$, iff $\forall i \in \{1, \cdots, L\}$, $(P_i^{'} = P_i) \vee (P_i = \star) \vee (P_i^{'} = \star)$

Patterns: Example

$\mathcal{U} = \{0, 1, \star\}$



Identity-Based Encryption With Wildcard (WIBE)

Identity-Based Encryption with Wildcards (WIBE) [2]

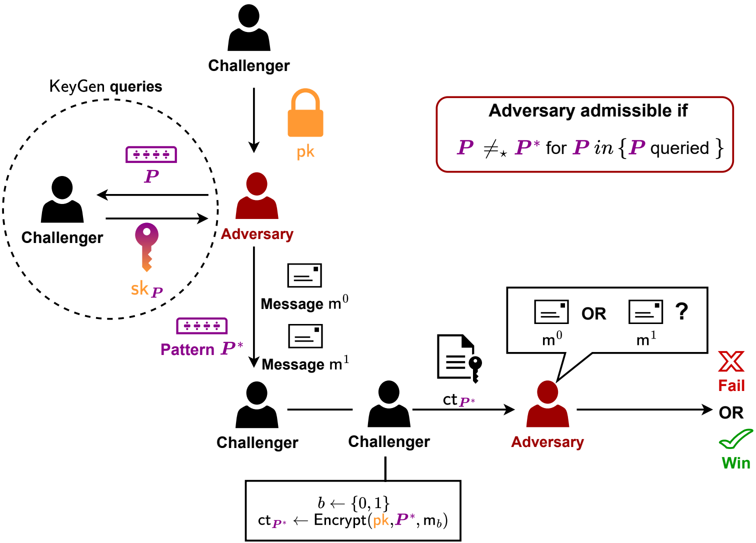
- $\mathsf{Setup}(1^\lambda, L) \rightarrow (\mathsf{pk}, \mathsf{msk})$
- $\mathsf{KeyGen}(\mathsf{msk}, \boldsymbol{P}) \rightarrow \mathsf{sk}_{\boldsymbol{P}}$
- $\mathsf{Encrypt}(\mathsf{pk}, \boldsymbol{P}^{'}, \mathsf{m}) \rightarrow \mathsf{ct}_{\boldsymbol{P}^{'}}$
- $\mathsf{Decrypt}(\mathsf{sk}_{\boldsymbol{P}}, \boldsymbol{P}, \mathsf{ct}_{\boldsymbol{P}^{'}}, (\boldsymbol{P}^{'})) \rightarrow \mathsf{m}^{'}$

Correctness:

For all $\lambda, L \in \mathbb{N}$, for $(\mathsf{pk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda, L)$ **honestly generated** and for all patterns $\boldsymbol{P}, \boldsymbol{P}^{'}$ such that $\boldsymbol{P} =_\star \boldsymbol{P}^{'}$:

$$\mathsf{Decrypt}(\mathsf{KeyGen}(\mathsf{msk}, \boldsymbol{P}), \boldsymbol{P}, \mathsf{Encrypt}(\mathsf{pk}, \boldsymbol{P}^{'}, \mathsf{m}), (\boldsymbol{P}^{'})) = \mathsf{m}$$

WIBE Security (1): Indistinguishability



WIBE Security (2): Pattern-Hiding

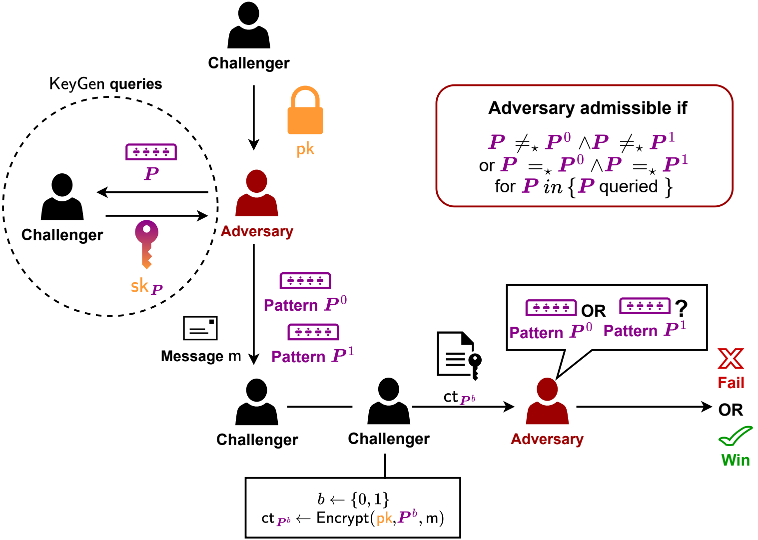


Table of contents

Access Policies, Attributes and Patterns

- $\pi = (\cdots \wedge \cdots) \vee (\cdots \wedge \cdots) \vee \cdots$
- $c = (a_k \wedge a_l \wedge \cdots \wedge a_z) \rightarrow \mathbf{P}^{'} \in \{0,1\}^L$:
  - $P_i^{'} = 1$ if $a_i \in c$
  - $P_i^{'} = 0$ otherwise, $i = 1, \cdots, L$
- $\text{AP2Pattern}(\pi, L) \rightarrow \mathbf{P}^{'}$: Produces a pattern from access policy π
- $A_{\text{U}} \rightarrow \mathbf{P} \in \{0, \star\}^L$
  - $P_i = \star$ if $a_i \in A_{\text{U}}$
  - $P_i = 0$ otherwise, $i = 1, \cdots, L$
- $\text{Att2Pattern}(A_{\text{U}}, L) \rightarrow \mathbf{P}_{\text{U}}$: Produces a pattern from attribute set A_{U}

Digital Signature Σ

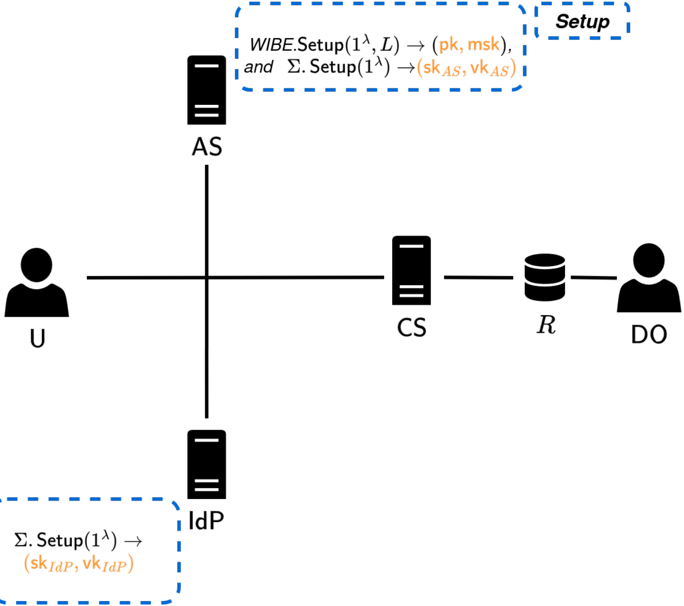
Definition

- $\mathsf{Setup}(1^\lambda) \rightarrow (\mathsf{sk}, \mathsf{vk})$
- $\mathsf{Sign}(\mathsf{sk}, \mathsf{m}) \rightarrow \sigma$
- $\mathsf{Verify}(\mathsf{vk}, \mathsf{m}, \sigma) \rightarrow 0/1$

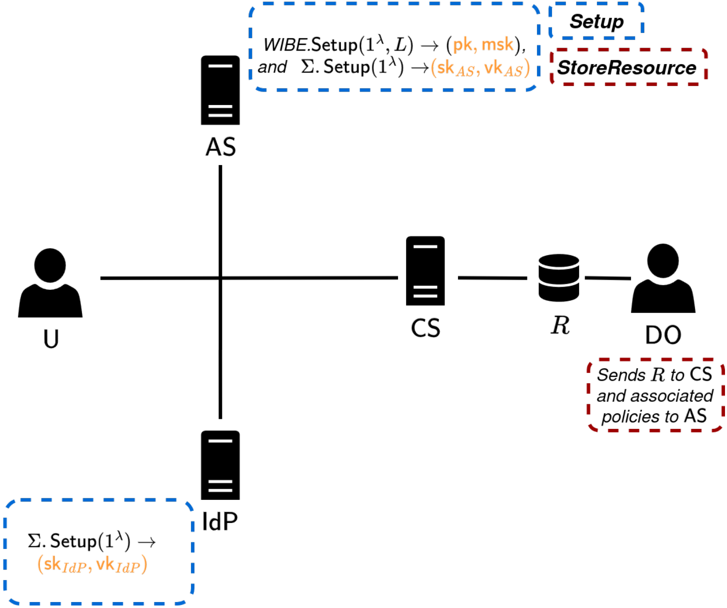
Security (informally)

Adversary **cannot forge** a signature for chosen message m

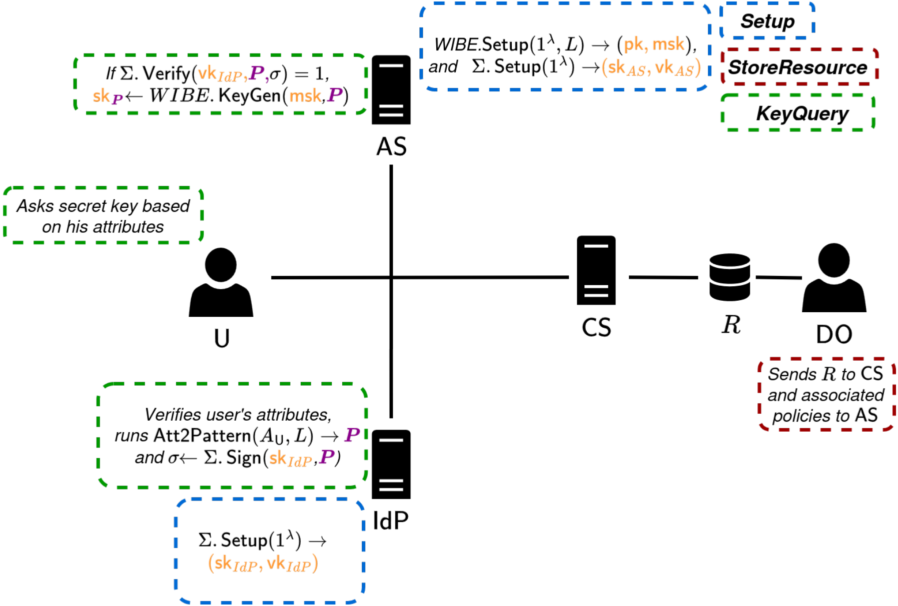
Our Use Case With a WIBE



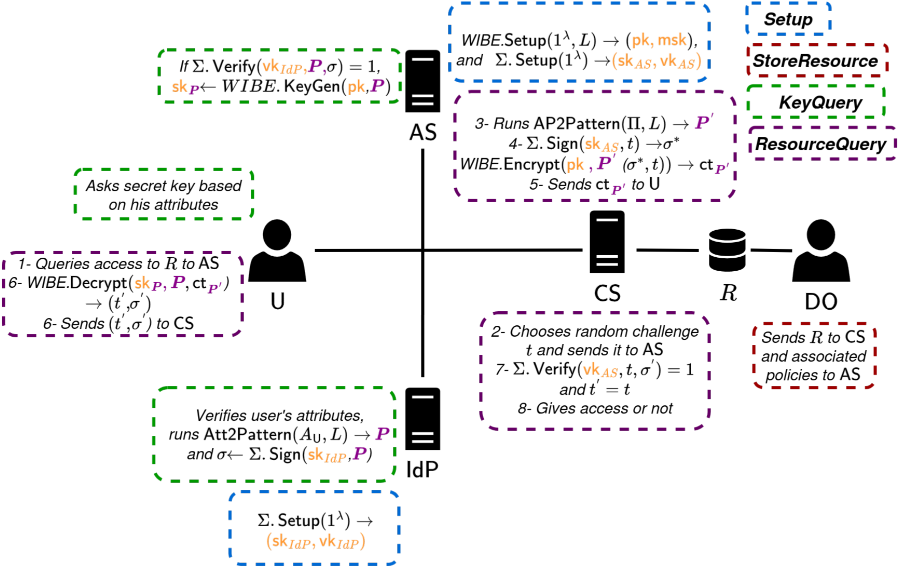
Our Use Case With a WIBE



Our Use Case With a WIBE



Our Use Case With a WIBE



Our Use Case With a WIBE

What about security requirements?

- No unauthorized access: YES

Our Use Case With a WIBE

What about security requirements?

- No unauthorized access: YES
- Access policy confidentiality: YES

Our Use Case With a WIBE

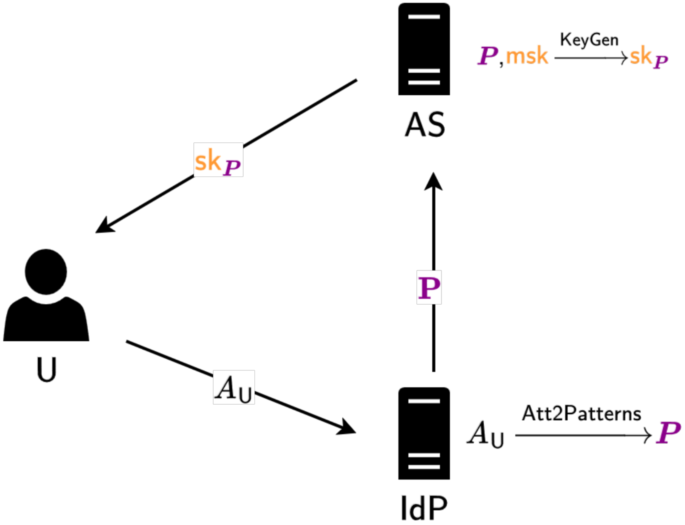
What about security requirements?

- No unauthorized access: YES
- Access policy confidentiality: YES
- Attribute protection: NO

Table of contents

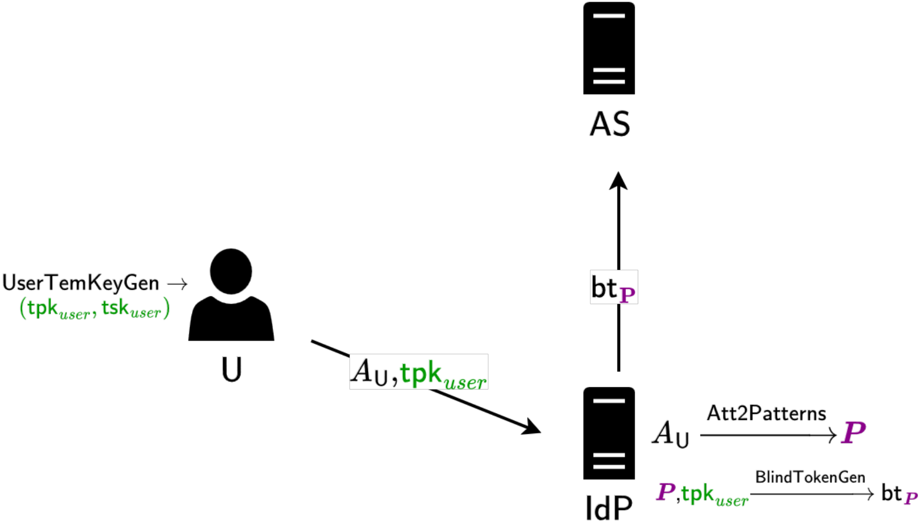
Replacing KeyGen by an Interactive Protocol

Original Key Generation:

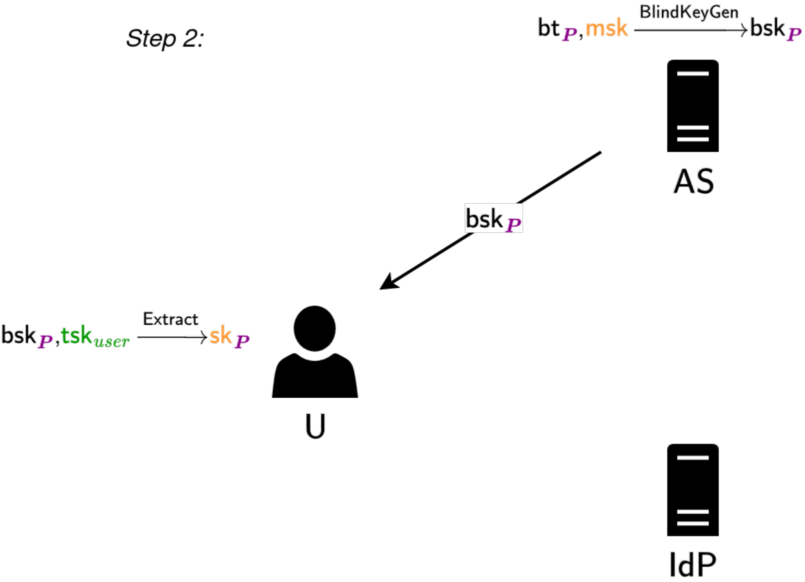


Replacing KeyGen by an Interactive Protocol

Step 1:



Replacing KeyGen by an Interactive Protocol



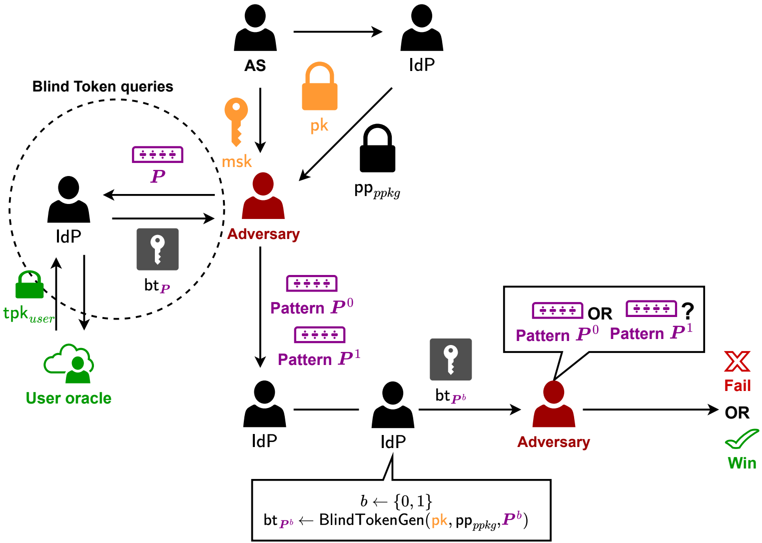
Privacy-Preserving Key Generation WIBE

Definition PPKG WIBE

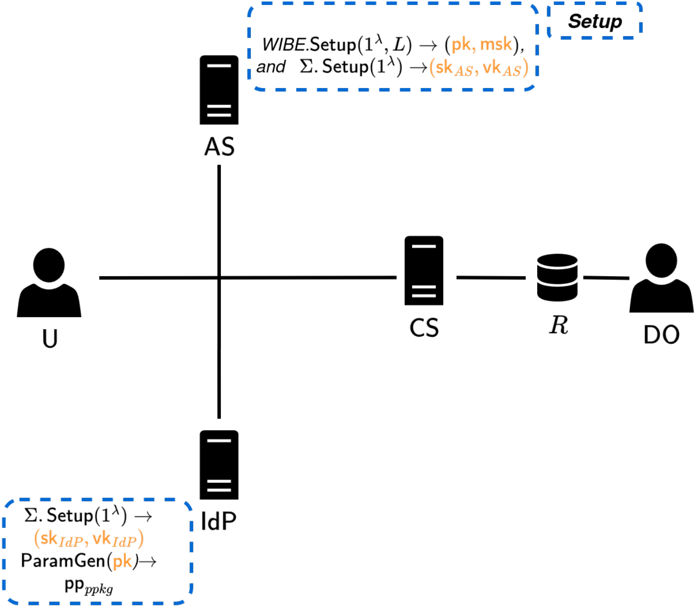
WIBE=(Setup, KeyGen, Encrypt, Decrypt) and

- $\mathsf{ParamGen}(\mathsf{pk}) \rightarrow \mathsf{pp}_{\mathsf{ppkg}}$ (IdP)
- $\mathsf{UserTemKeyGen}(\mathsf{pk}, \mathsf{pp}_{\mathsf{ppkg}}) \rightarrow (\mathsf{tpk}_{user}, \mathsf{tsk}_{user})$ (U)
- $\mathsf{BlindTokenGen}(\mathsf{pk}, \mathsf{pp}_{\mathsf{ppkg}}, \mathbf{P}, \mathsf{tpk}_{user}) \rightarrow \mathsf{bt}_{\mathbf{P}}$ (IdP)
- $\mathsf{BlindKeyGen}(\mathsf{msk}, \mathsf{bt}_{\mathbf{P}}) \rightarrow \mathsf{bsk}_{\mathbf{P}}$ (AS)
- $\mathsf{KeyExtract}(\mathsf{bsk}_{\mathbf{P}}, \mathsf{tsk}_{user}) \rightarrow \mathsf{sk}_{\mathbf{P}}$ (U)

Privacy-Preserving Key Generation WIBE



Back to Our Access Control Use Case



Back to Our Access Control Use Case

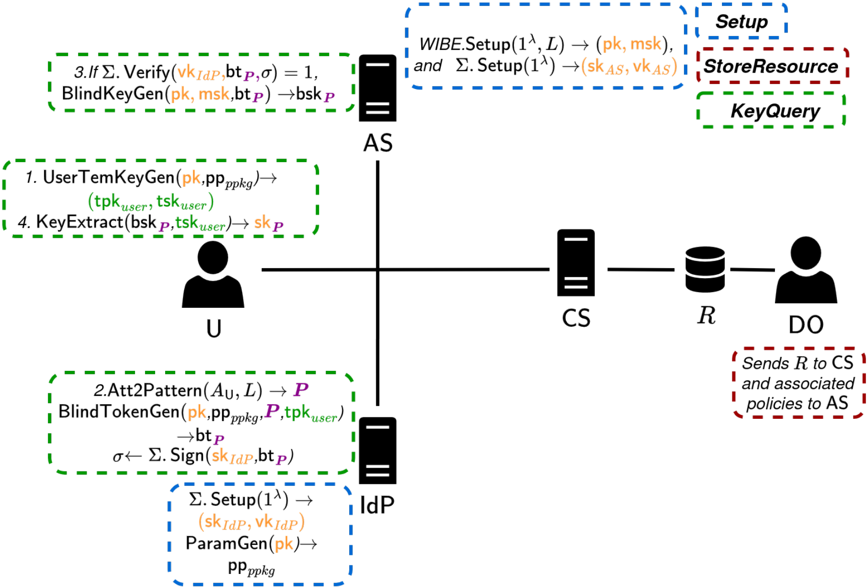


Table of contents

Auxiliary Primitive: Inner Product Encryption (IPE)

Vectors $\mathbf{x}, \mathbf{y}$ of length $N \in \mathbb{N}$

Inner Product Encryption Scheme [5]

- $\mathsf{Setup}(1^\lambda, N) \rightarrow (\mathsf{pk}, \mathsf{msk})$
- $\mathsf{KeyGen}(\mathsf{msk}, \mathbf{y}) \rightarrow \mathsf{sk}_{\mathbf{y}}$
- $\mathsf{Encrypt}(\mathsf{pk}, \mathbf{x}, \mathsf{m}) \rightarrow \mathsf{ct}_{\mathbf{x}}$
- $\mathsf{Decrypt}(\mathsf{sk}_{\mathbf{y}}, \mathsf{ct}_{\mathbf{x}}) \rightarrow \mathsf{m}' = m$ if $\langle \mathbf{x}, \mathbf{y} \rangle = 0$

Security (informally)

- *payload-hiding*: ciphertexts hide messages
- weak/strong *attribute-hiding*: ciphertexts hide vectors $\mathbf{x}$

Generic Construction [1]

WIBE with patterns length L from IPE with vectors length $N = 2L$

Generic Construction [1]

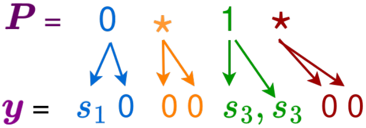
WIBE with patterns length L from IPE with vectors length $N = 2L$

Algorithm ExtendingKeyPatternRandomized

Input: key pattern $\boldsymbol{P}$ of length L

Output: randomized vector $\mathbf{y}$ of length $N = 2L$

$i \leftarrow 1$, $j \leftarrow 1$
while $i \leq L$, $j \leq 2L$ **do**
  if $P_i \neq \star$ **then**
    $y_j \leftarrow s_i$ and $y_{j+1} \leftarrow s_i \cdot P_i$ for $s_i \leftarrow \mathbb{Z}_p$
  else
    $y_j \leftarrow 0$ and $y_{j+1} \leftarrow 0$
  end if
  $j \leftarrow j + 2$, $i \leftarrow i + 1$
end while
return y



Generic Construction [1]

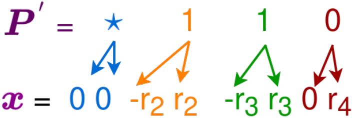
WIBE with patterns length L from IPE with vectors length $N = 2L$

Algorithm ExtendingCtPattern

Input: ciphertext pattern $\boldsymbol{P}$ of length L

Output: vector $\boldsymbol{x}$ of length $2L$

$i \leftarrow 1$, $j \leftarrow 1$
while $i \leq L$, $j \leq 2L$ **do**
  if $P_i \neq \star$ **then**
    $x_j \leftarrow -r_i \cdot P_i$, $x_{j+1} \leftarrow r_i$ for $r_i \leftarrow \mathbb{Z}_p$
  else
    $x_j \leftarrow 0$ and $x_{j+1} \leftarrow 0$
  end if
  $j \leftarrow j + 2$, $i \leftarrow i + 1$
end while
return $\boldsymbol{x}$



Generic Construction [1]

WIBE with patterns length L from IPE with vectors length $N = 2L$

Correctness: $\langle \mathbf{x}, \mathbf{y} \rangle = 0 \iff \mathbf{P} =_\star \mathbf{P}'$

$$\boldsymbol{P} = \begin{bmatrix} 0 & \star & 1 & \star \end{bmatrix}$$

$$\boldsymbol{P}' = \begin{bmatrix} \star & 1 & 1 & 0 \end{bmatrix}$$

$$\boldsymbol{y} = s_1\ 0\quad 0\ 0\quad s_3\ s_3\quad 0\ 0$$

$$\boldsymbol{x} = 0\ 0\quad -r_2\ r_2\quad -r_3\ r_3\quad 0\ r_4$$

$$\boldsymbol{P} =_\star \boldsymbol{P}'$$

$$\langle \mathbf{x}, \mathbf{y} \rangle = s_1.0 + 0.0 - r_2.0 + r_2.0 - r_3.\,s_3 + r_3.\,s_3 + 0.0 + 0.r_4 = 0$$

Generic Construction [1]

WIBE with patterns length L from IPE with vectors length $N = 2L$

Security:

- payload-hiding IPE $\Longrightarrow$ indistinguisable WIBE [1]
- *weak* attribute-hiding IPE $\Longrightarrow$ *anonymous* WIBE [1]
- **strong** attribute-hiding IPE $\Longrightarrow$ **pattern-hiding** WIBE

PPKG IPE $\Longrightarrow$ PPKG WIBE

Table of contents

Mathematical Background and Notations

Asymmetric Bilinear Pairing Group

$\Gamma = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e)$ where

- prime p
- cyclic groups $\mathbb{G}_1, \mathbb{G}_2$, $\mathbb{G}_T$, of order p
- $\langle g_1 \rangle = \mathbb{G}_1$, $\langle g_2 \rangle = \mathbb{G}_2$
- polynomial-time computable non-degenerate bilinear pairing e
  - $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$
  - $e(g_1^s, g_2^t) = e(g_1, g_2)^{st}$, $s, t \in \mathbb{Z}_p$
  - $e(g_1, g_2) \neq 1$

Notation

Vectors $\mathbf{u}, \mathbf{v} \in \mathbb{Z}_p^N$

- $g_1^{\mathbf{u}} = (g_1^{u_1}, g_1^{u_2}, \cdots, g_1^{u_N})$
- $e(g_1^{\mathbf{u}}, g_2^{\mathbf{v}}) = \prod_{i=1}^N e(g_1^{u_i}, g_2^{v_i}) = e(g_1, g_2)^{\sum_{i=1}^N u_i \cdot v_i}$

A Strong Attribute-Hiding IPE Scheme

Chen *et al.* [4]'s modified private-key IPE

- Setup($1^\lambda, N$): Γ, $\boldsymbol{B} \leftarrow \mathbb{Z}_p^{4\times 2}$, $\mathbf{u}, \mathbf{w}_1, \cdots, \mathbf{w}_N \leftarrow \mathbb{Z}_p^{1\times 4}, \alpha \leftarrow \mathbb{Z}_p$

$$\mathsf{msk} = (\Gamma, \alpha, \mathbf{u}, \mathbf{w}_1, \cdots, \mathbf{w}_N, \mathbf{B}), \mathsf{pp} = \Gamma$$

A Strong Attribute-Hiding IPE Scheme

Chen *et al.* [4]'s modified private-key IPE

- $\mathsf{Setup}(1^\lambda, N)$: Γ, $\boldsymbol{B} \leftarrow \mathbb{Z}_p^{4\times 2}$, $\mathbf{u}, \mathbf{w}_1, \cdots, \mathbf{w}_N \leftarrow \mathbb{Z}_p^{1\times 4}, \alpha \leftarrow \mathbb{Z}_p$

$$\mathsf{msk} = (\Gamma, \alpha, \mathbf{u}, \mathbf{w}_1, \cdots, \mathbf{w}_N, \mathbf{B}), \mathsf{pp} = \Gamma$$

- $\mathsf{KeyGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{y} = (y_1, \cdots, y_N) \in \mathbb{Z}_p^N)$: $\mathbf{r} \leftarrow \mathbb{Z}_p^{2\times 1}$,

$$\mathsf{sk}_\mathbf{y} = (K_0, K_1) = (g_2^{\alpha + (\sum_{i=1}^N y_i \mathbf{w}_i)\mathbf{Br}}, g_2^{\mathbf{Br}})$$

A Strong Attribute-Hiding IPE Scheme

Chen *et al.* [4]'s modified private-key IPE

- Setup($1^\lambda, N$): Γ, $\boldsymbol{B} \leftarrow \mathbb{Z}_p^{4\times 2}$, $\mathbf{u}, \mathbf{w}_1, \cdots, \mathbf{w}_N \leftarrow \mathbb{Z}_p^{1\times 4}, \alpha \leftarrow \mathbb{Z}_p$

$$\mathsf{msk} = (\Gamma, \alpha, \mathbf{u}, \mathbf{w}_1, \cdots, \mathbf{w}_N, \mathbf{B}), \mathsf{pp} = \Gamma$$

- KeyGen(pp, msk, $\mathbf{y} = (y_1, \cdots, y_N) \in \mathbb{Z}_p^N$): $\mathbf{r} \leftarrow \mathbb{Z}_p^{2\times 1}$,

$$\mathsf{sk}_{\mathbf{y}} = (K_0, K_1) = (g_2^{\alpha + (\sum_{i=1}^N y_i \mathbf{w}_i)\mathbf{Br}}, g_2^{\mathbf{Br}})$$

- Encrypt(pp, msk, $\mathbf{x} = (x_1, \cdots, x_N) \in \mathbb{Z}_p^N, \mathsf{m} \in \mathbb{G}_T$): $s \leftarrow \mathbb{Z}_p$,

$$\mathsf{ct}_{\mathbf{x}} = (\{C_i\}_{i=1}^N, C, C_{aux}) = (\left\{g_1^{s(\mathbf{u}x_i + \mathbf{w}_i)}\right\}_{i=1}^N, e(g_1, g_2)^{\alpha s}\mathsf{m}, g_1^{-s})$$

A Strong Attribute-Hiding IPE Scheme

Chen *et al.* [4]'s modified private-key IPE

- Setup($1^\lambda, N$): Γ, $\boldsymbol{B} \leftarrow \mathbb{Z}_p^{4\times 2}$, $\mathbf{u}, \mathbf{w}_1, \cdots, \mathbf{w}_N \leftarrow \mathbb{Z}_p^{1\times 4}, \alpha \leftarrow \mathbb{Z}_p$

$$\mathsf{msk} = (\Gamma, \alpha, \mathbf{u}, \mathbf{w}_1, \cdots, \mathbf{w}_N, \mathbf{B}), \mathsf{pp} = \Gamma$$

- KeyGen(pp, msk, $\mathbf{y} = (y_1, \cdots, y_N) \in \mathbb{Z}_p^N$): $\mathbf{r} \leftarrow \mathbb{Z}_p^{2\times 1}$,

$$\mathsf{sk}_{\mathbf{y}} = (K_0, K_1) = (g_2^{\alpha + (\sum_{i=1}^N y_i \mathbf{w}_i)\mathbf{Br}}, g_2^{\mathbf{Br}})$$

- Encrypt(pp, msk, $\mathbf{x} = (x_1, \cdots, x_N) \in \mathbb{Z}_p^N, \mathsf{m} \in \mathbb{G}_T$): $s \leftarrow \mathbb{Z}_p$,

$$\mathsf{ct}_{\mathbf{x}} = (\{C_i\}_{i=1}^N, C, C_{aux}) = (\left\{g_1^{s(\mathbf{u}x_i + \mathbf{w}_i)}\right\}_{i=1}^N, e(g_1, g_2)^{\alpha s}\mathsf{m}, g_1^{-s})$$

- Decrypt($\mathsf{sk}_{\mathbf{y}}, \mathsf{ct}_{\mathbf{x}}$): $\mathsf{m}' = C \cdot e(\prod_{i=1}^N C_i^{y_i}, K_1) \cdot e(C_{aux}, K_0)$

Correctness of the IPE

$$C \cdot e(\prod_{i=1}^{N} C_i^{y_i}, K_1) \cdot e(C_{aux}, K_0)$$

$$= e(g_1, g_2)^{\alpha s}\mathsf{m} \cdot e(g_1, g_2)^{s\mathbf{u}\langle \mathbf{x}, \mathbf{y}\rangle \mathbf{Br} + s\sum_{i=1}^{N} y_i \mathbf{w}_i \mathbf{Br}} \cdot e(g_1, g_2)^{-s\alpha - s\sum_{i=1}^{N} y_i w_i \mathbf{Br}}$$

$$= \mathsf{m} \cdot e(g_1, g_2)^{s\mathbf{u}\langle \mathbf{x}, \mathbf{y}\rangle \mathbf{Br}}$$

$$= \mathsf{m}$$

if $\langle \mathbf{x}, \mathbf{y} \rangle = 0$

Adding PPKG to our IPE

$$\mathsf{sk}_y = (K_0, K_1) = \left(g_2^{\alpha + \left(\sum_{i=1}^{N} y_i \mathbf{w}_i\right)\mathbf{Br}}, g_2^{\mathbf{Br}} \right)$$

Adding PPKG to our IPE

$$\mathsf{sk}_y = (K_0, K_1) = \left(g_2^{\alpha + \left(\sum_{i=1}^{N} y_i \mathbf{w}_i\right)\mathbf{Br}}, g_2^{\mathbf{Br}} \right)$$

Sending $\left\{ g_2^{y_i + \gamma} \right\}_{i=1}^{N}$ instead of $\left\{ g_2^{y_i} \right\}_{i=1}^{N}$, $\gamma \in \mathbb{Z}_p$

Adding PPKG to our IPE

$$\mathsf{sk}_y = (K_0, K_1) = \left(g_2^{\alpha + \left(\sum_{i=1}^{N} y_i \mathbf{w}_i\right)\mathbf{Br}}, g_2^{\mathbf{Br}}\right)$$

Sending $\left\{g_2^{y_i+\gamma}\right\}_{i=1}^{N}$ instead of $\left\{g_2^{y_i}\right\}_{i=1}^{N}$, $\gamma \in \mathbb{Z}_p$

Problem: γ constant $\rightarrow$ KGC learns information.

Adding PPKG to our IPE

$$\mathsf{sk}_y = (K_0, K_1) = \left(g_2^{\alpha + \left(\sum_{i=1}^{N} y_i \mathbf{w}_i\right)\mathbf{Br}}, g_2^{\mathbf{Br}} \right)$$

Sending $\left\{ g_2^{y_i + \gamma_i} \right\}_{i=1}^{N}$, $\gamma_i \in \mathbb{Z}_p$

Thus,

$$g_2^{\alpha} \cdot \left(\prod_{i=1}^{N} g_2^{y_i + \gamma_i} \right)^{\mathbf{w}_i \mathbf{Br}}$$

$$= g_2^{\alpha + \left(\sum_{i=1}^{N} y_i \mathbf{w}_i\right)\mathbf{Br} + \left(\sum_{i=1}^{N} \gamma_i \mathbf{w}_i\right)\mathbf{Br}}$$

Adding PPKG to our IPE

$$\mathsf{sk}_y = (K_0, K_1) = \left(g_2^{\alpha + \left(\sum_{i=1}^{N} y_i \mathbf{w}_i\right)\mathbf{Br}}, g_2^{\mathbf{Br}} \right)$$

Sending $\left\{ g_2^{y_i + \gamma_i} \right\}_{i=1}^{N}$, $\gamma_i \in \mathbb{Z}_p$

Thus,

$$g_2^{\alpha} \cdot \left(\prod_{i=1}^{N} g_2^{y_i + \gamma_i} \right)^{\mathbf{w}_i \mathbf{Br}}$$

$$= g_2^{\alpha + \left(\sum_{i=1}^{N} y_i \mathbf{w}_i\right)\mathbf{Br} + \left(\sum_{i=1}^{N} \gamma_i \mathbf{w}_i\right)\mathbf{Br}}$$

Problem: Must remove $g_2^{\left(\sum_{i=1}^{n} \gamma_i \mathbf{w}_i\right)\mathbf{Br}}$

Adding PPKG to our IPE

$$\mathsf{sk}_{y} = (K_0, K_1) = \left(g_2^{\alpha + \left(\sum_{i=1}^{N} y_i \mathbf{w}_i\right)\mathbf{Br}}, g_2^{\mathbf{Br}} \right)$$

Sending $\left\{ g_2^{y_i + \gamma_i} \right\}_{i=1}^{N}$, $\gamma_i \in \mathbb{Z}_p$

Thus,

$$g_2^{\alpha} \cdot \left(\prod_{i=1}^{N} g_2^{y_i + \gamma_i} \right)^{\mathbf{w}_i \mathbf{Br}}$$

$$= g_2^{\alpha + \left(\sum_{i=1}^{N} y_i \mathbf{w}_i\right)\mathbf{Br} + \left(\sum_{i=1}^{N} \gamma_i \mathbf{w}_i\right)\mathbf{Br}}$$

Problem: Must remove $g_2^{\left(\sum_{i=1}^{n} \gamma_i \mathbf{w}_i\right)\mathbf{Br}}$

Including $\left\{ g_2^{\mathbf{w}_i \mathbf{Br}} \right\}_{i=1}^{N}$ in $\mathsf{bsk}_{\mathbf{y}}$

Adding PPKG to our IPE

$\mathsf{sk}_{y} = (K_0, K_1) = \left(g_2^{\alpha + \left(\sum_{i=1}^N y_i \mathbf{w}_i\right)\mathbf{Br}}, g_2^{\mathbf{Br}} \right)$

Sending $\left\{ g_2^{y_i + \gamma_i} \right\}_{i=1}^N$, $\gamma_i \in \mathbb{Z}_p$

Thus,

$$g_2^{\alpha} \cdot \left(\prod_{i=1}^N g_2^{y_i + \gamma_i} \right)^{\mathbf{w}_i \mathbf{Br}}$$

$$= g_2^{\alpha + \left(\sum_{i=1}^N y_i \mathbf{w}_i\right)\mathbf{Br} + \left(\sum_{i=1}^N \gamma_i \mathbf{w}_i\right)\mathbf{Br}}$$

Problem: Must remove $g_2^{\left(\sum_{i=1}^n \gamma_i \mathbf{w}_i\right)\mathbf{Br}}$

Including $\left\{ g_2^{\mathbf{w}_i \mathbf{Br}} \right\}_{i=1}^N$ in $\mathsf{bsk}_{\mathbf{y}}$

Problem: Knowing $\{\gamma_i, y_i\}_{i=1}^N$ and $\left\{ g_2^{\mathbf{w}_i \mathbf{Br}} \right\}_{i=1}^N \rightarrow$ recover g_2^{α}

Adding PPKG to our IPE

$$\mathsf{sk}_y = (K_0, K_1) = \left(g_2^{\alpha + \left(\sum_{i=1}^{N} y_i \mathbf{w}_i\right)\mathbf{Br}}, g_2^{\mathbf{Br}} \right)$$

Associate $\{\gamma_i\}_{i=1}^{N}$ with $\theta \in \mathbb{Z}_p$

Sending $\left\{ g_2^{y_i + \theta \gamma_i} \right\}_{i=1}^{N}$ and g_2^{θ}

Adding PPKG to our IPE

$$\mathsf{sk}_y = (K_0, K_1) = \left(g_2^{\alpha + \left(\sum_{i=1}^N y_i \mathbf{w}_i\right)\mathbf{Br}}, g_2^{\mathbf{Br}}\right)$$

Associate $\{\gamma_i\}_{i=1}^N$ with $\theta \in \mathbb{Z}_p$

Sending $\left\{g_2^{y_i + \theta\gamma_i}\right\}_{i=1}^N$ and g_2^{θ}

$$g_2^{\alpha + \left(\sum_{i=1}^N y_i \mathbf{w}_i\right)\mathbf{Br} + \theta\left(\sum_{i=1}^N \gamma_i \mathbf{w}_i\right)\mathbf{Br}}$$

Thus $\mathsf{bsk}_{\mathbf{y}}$ must include $\left\{g_2^{\theta \mathbf{w}_i \mathbf{Br}}\right\}_{i=1}^N$

Adding PPKG to our IPE

$\mathsf{sk}_y = (K_0, K_1) = \left(g_2^{\alpha + \left(\sum_{i=1}^{N} y_i \mathbf{w}_i\right)\mathbf{Br}}, g_2^{\mathbf{Br}} \right)$

Associate $\{\gamma_i\}_{i=1}^{N}$ with $\theta \in \mathbb{Z}_p$

Sending $\left\{ g_2^{y_i + \theta\gamma_i} \right\}_{i=1}^{N}$ and g_2^{θ}

$$g_2^{\alpha + \left(\sum_{i=1}^{N} y_i \mathbf{w}_i\right)\mathbf{Br} + \theta\left(\sum_{i=1}^{N} \gamma_i \mathbf{w}_i\right)\mathbf{Br}}$$

Thus $\mathsf{bsk}_{\mathbf{y}}$ must include $\left\{ g_2^{\theta \mathbf{w}_i \mathbf{Br}} \right\}_{i=1}^{N}$

Problem: User knowing $\theta \to g_2^{\alpha}$

Adding PPKG to our IPE

$$\mathsf{sk}_y = (K_0, K_1) = \left(g_2^{\alpha + \left(\sum_{i=1}^{N} y_i \mathbf{w}_i\right)\mathbf{Br}}, g_2^{\mathbf{Br}}\right)$$

Associate $\{\gamma_i\}_{i=1}^{N}$ with $\theta \in \mathbb{Z}_p$

Sending $\left\{g_2^{y_i + \theta\gamma_i}\right\}_{i=1}^{N}$ and g_2^{θ}

$$g_2^{\alpha + \left(\sum_{i=1}^{N} y_i \mathbf{w}_i\right)\mathbf{Br} + \theta\left(\sum_{i=1}^{N} \gamma_i \mathbf{w}_i\right)\mathbf{Br}}$$

Thus $\mathsf{bsk}_{\mathbf{y}}$ must include $\left\{g_2^{\theta \mathbf{w}_i \mathbf{Br}}\right\}_{i=1}^{N}$

Problem: User knowing $\theta \rightarrow g_2^{\alpha}$

Solution: IdP selects $\tilde{g}_2 = g_2^{\theta}$ **randomly**

Table of contents

Results

Implementation[1] done in *C* with *Relic* [3]

Used curves: two 128-bits security level curves

- BLS24-P317 as elements in $\mathbb{G}_1$ are small
- BLS12-P381 as pairing is fast

$L = 100$ and $N = 200$

Key generation, privacy-preserving key generation, encryption and decryption execution time. Time is in milliseconds and rounded up.

Algorithms / Curves	KeyGen	Encrypt	Decrypt	**PPKG**
BLS24-P317	22	31	13	**330**
BLS12-P381	9	31	12	**140**

[1] Thanks to Cyril Bouvier

Results

Implementation[1] done in *C* with *Relic* [3]

Used curves: two 128-bits security level curves

- BLS24-P317 as elements in $\mathbb{G}_1$ are small
- BLS12-P381 as pairing is fast

$L = 100$ and $N = 200$

Table: (Theoretical) Sizes in **kilo-bytes**, and rounded.

Elements / Curves	pp_{ppkg}	ct	sk	bt	tpk_{user}	bsk
BLS24-P317	0.3	8.7	1.6	**63**	**60**	**63.6**
BLS12-P381	0.1	10	0.5	**19**	**20**	**19.2**

[1] Thanks to Cyril Bouvier

Using a Trick: Group inversion

- In [4]'s scheme:
  - $|\mathsf{ct}| = O(N)$ elements of $\mathbb{G}_1$
  - $|\mathsf{sk}| = O(1)$ elements of $\mathbb{G}_2$

Using a Trick: Group inversion

- In [4]'s scheme:
  - $|\mathsf{ct}| = O(N)$ elements of $\mathbb{G}_1$
  - $|\mathsf{sk}| = O(1)$ elements of $\mathbb{G}_2$
- In our scheme:
  - $|\mathsf{ct}| = O(N)$ elements of $\mathbb{G}_1$
  - $|\mathsf{bsk}| = O(N)$ elements of $\mathbb{G}_2$

Using a Trick: Group inversion

- In [4]'s scheme:
  - $|\mathsf{ct}| = O(N)$ elements of $\mathbb{G}_1$
  - $|\mathsf{sk}| = O(1)$ elements of $\mathbb{G}_2$

- In our scheme:
  - $|\mathsf{ct}| = O(N)$ elements of $\mathbb{G}_1$
  - $|\mathsf{bsk}| = O(N)$ elements of $\mathbb{G}_2$

- Can switch elements from $\mathbb{G}_1$ and $\mathbb{G}_2$!

Using a Trick: Group inversion

Key generation, privacy-preserving key generation, encryption and decryption execution time. Time is in milliseconds and rounded up.

Algorithms / Curves	KeyGen	Encrypt	Decrypt	**PPKG**
BLS24-P317	4	224	51	**67**
BLS12-P381	4	85	22	**71**

Using a Trick: Group inversion

Table: (Theoretical) Sizes in **kilo-bytes**, and rounded.

Elements / Curves	$\mathsf{pp}_{\mathsf{ppkg}}$	ct	sk	bt	tpk_{user}	bsk
BLS24-P317	0.04	63.6	0.2	**8**	**8**	**8.1**
BLS12-P381	0.05	19.4	0.2	**9.6**	**10**	**9.7**

Table of contents

Conclusion

Contributions:

- New security property for IPE and WIBE: **PPKG**
- First PPKG IPE and WIBE construction (using pairings)
- Novel access control method using WIBE in a real-world use case

Future Works:

- Develop generic PPKG constructions
- Explore more complex access policies
- Tutorial for pairing-based protocol implementation

Any questions?

Thank you for your attention!

Bibliography I

Abdalla, M., Caro, A.D., Phan, D.H.: Generalized key delegation for wildcarded identity-based and inner-product encryption. IEEE Trans. Inf. Forensics Secur. **7**(6), 1695–1706 (2012). doi:10.1109/TIFS.2012.2213594, `https://doi.org/10.1109/TIFS.2012.2213594`

Abdalla, M., Catalano, D., Dent, A., Malone-Lee, J., Neven, G., Smart, N.: Identity-based encryption gone wild. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006, Part II. LNCS, vol. 4052, pp. 300–311. Springer, Heidelberg (Jul 2006). doi:10.1007/11787006_26

Aranha, D.F., Gouvêa, C.P.L.: RELIC is an Efficient LIbrary for Cryptography. `https://github.com/relic-toolkit/relic`

Bibliography II

Chen, J., Gong, J., Wee, H.: Improved inner-product encryption with adaptive security and full attribute-hiding. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018, Part II. LNCS, vol. 11273, pp. 673–702. Springer, Heidelberg (Dec 2018). doi:10.1007/978-3-030-03329-3_23

Katz, J., Sahai, A., Waters, B.: Predicate encryption supporting disjunctions, polynomial equations, and inner products. In: Smart, N.P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 146–162. Springer, Heidelberg (Apr 2008). doi:10.1007/978-3-540-78967-3_9

Related Works

Table: Comparison of the security properties satisfied by the different tools.

Tool	Privacy of access policies	Privacy of Attributes
Role-Based Access Control	$\surd$	$\times$
Anonymous Credential Systems	$\times$	$\times$
Attribute-Based Signatures	$\times$	$\surd$
Attribute Based Encryption	$\times$	$\surd$
	or $\surd$	$\times$