Lecture 2
August 29, 13:40 – 15:40

- Public-key encryption with keyword search
- Anonymous identity-based encryption
- Identity-based encryption with wildcards

Public-key encryption with keyword search & anonymous IBE

Motivation

- Suppose Bob sends an encrypted email to Alice
- Alice's email gateway may want to test if the email contains the word "urgent", so that it could route the email accordingly
- Still, Alice does not want the gateway to be able to decrypt her messages
- **Public-key encryption with keyword search**: Enable gateway to test whether a given keyword is present in the email without learning anything else about the email

PEKS: Basic idea

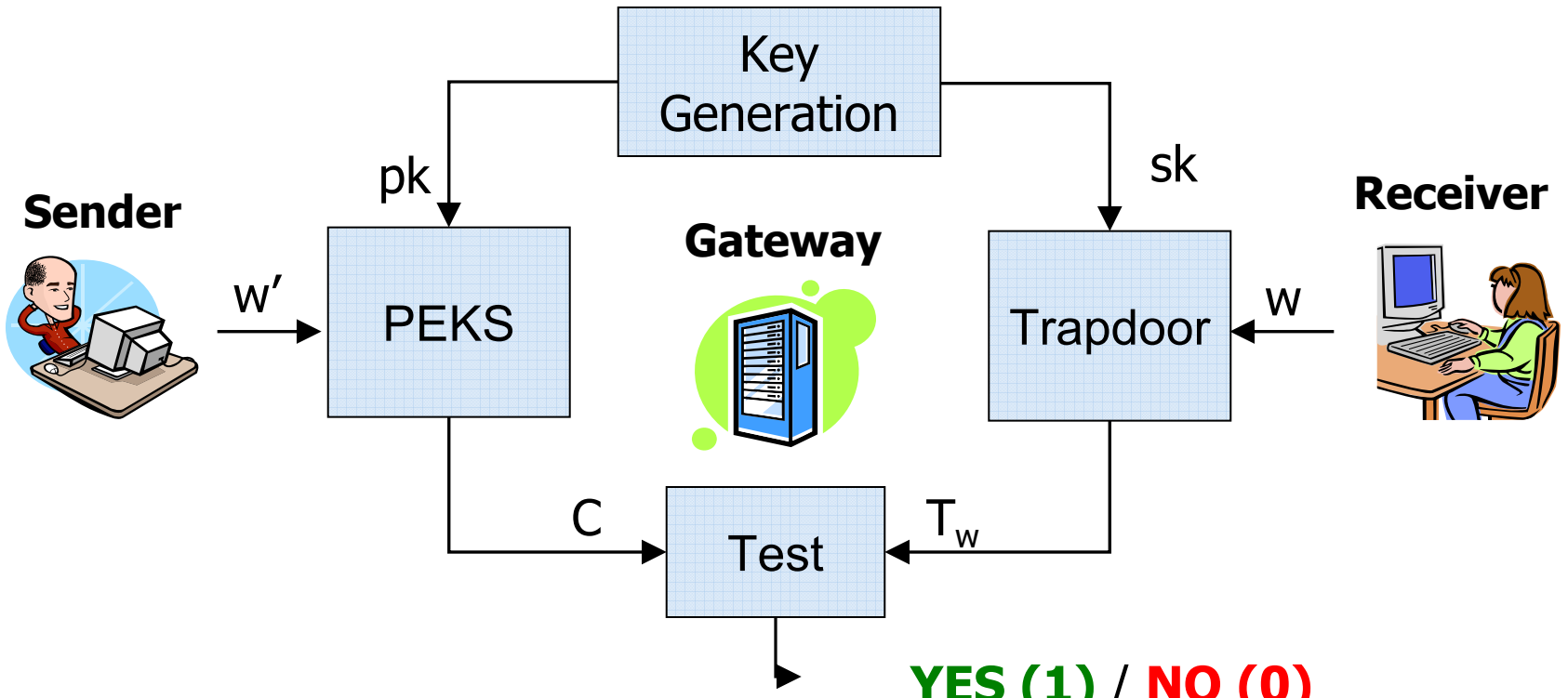
- Bob encrypts his email using a standard public-key encryption scheme PKE
- He then appends the public-key encryption with keyword search (PEKS) of each keyword

$$\text{Enc}(PK_{Alice}, \text{Email}) \,||\, \text{PEKS}(PK_{Alice}, W_1) \,||\, \ldots \,||\, \text{PEKS}(PK_{Alice}, W_m)$$

- **Main property**: Alice can give the gateway a trapdoor t_w that allows it to test whether $W_i = W$ for $i=1,\ldots,m$

PEKS: Public-key encryption with keyword search [BDOP04]

Goal: Allow gateway to test for the presence of keywords in ciphertexts



Consistency in cryptography

- **Every cryptographic primitive needs to satisfy two conditions**:
  - Security
  - Consistency
- **Example**: Public-key encryption
  - Security: Privacy (IND-CPA or IND-CCA)
  - Consistency: Decryption should reverse encryption
    - Let (sk,pk) be the output of the key generation
    - If C = Enc(pk,M), then Dec(sk,C) should return M

PEKS Security and consistency [BDOP04]

- **Security (IND-CPA)**
  - Ciphertext should not reveal any information about the encrypted keyword
  - The trapdoor for a keyword w should only allow the gateway to learn whether a given ciphertext contains w
- **Consistency**
  - Test should output 1 if and only if w'=w

Consistency of BDOP-PEKS

- In [BDOP04], the authors presented an efficient PEKS scheme (BDOP-PEKS) based on bilinear maps
  - Based on Boneh-Franklin's Basic IBE scheme [BF01]
- BDOP-PEKS does **NOT** meet their consistency notion
  - There are keywords w and w' such that **Trapdoor**(sk,w) = **Trapdoor**(sk,w')
  - Hence, **Test**(**Trapdoor**(sk,w),**PEKS**(pk,w'))=1
- Is there a weaker notion of consistency met by BDOP-PEKS which is still adequate in practice?

New notions of consistency

- A hierarchy of consistency notions
  - **Perfect** (BDOP04 consistency definition)
  - **Statistical**
  - **Computational** (achieved by BDOP-PEKS)

- Analogy to encryption case
  - **Perfect**: No decryption error
  - **Statistical**: Negligible probability of decryption error
  - **Computational**: Negligible probability of decryption error with respect to probabilistic polynomial time adversaries

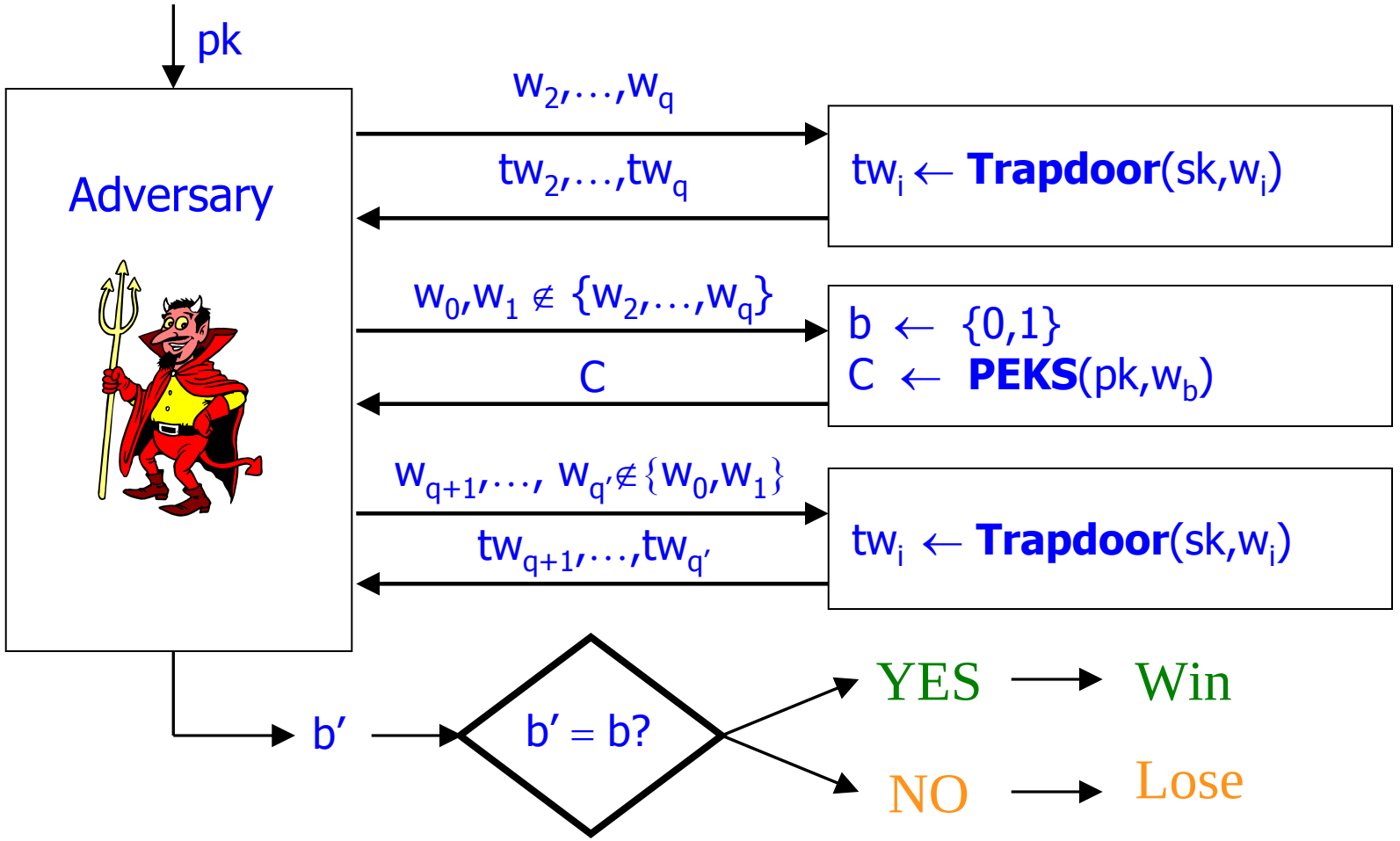
Outline

- **Definitions**
- PEKS constructions
- IBE-to-PEKS transformations
- Extensions
- Conclusion

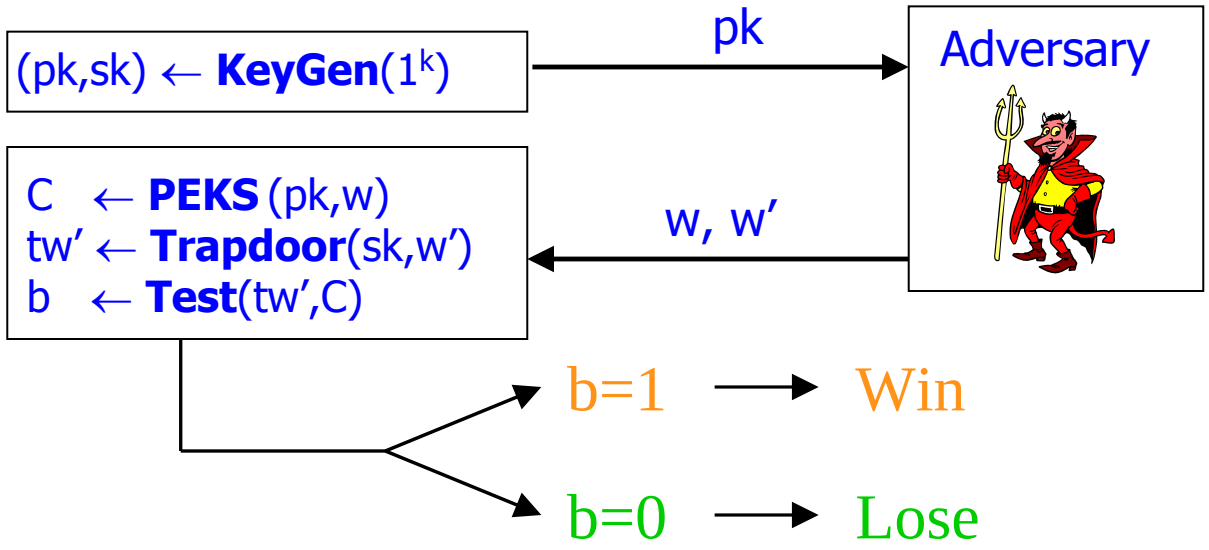
PEKS-IND-CPA: Privacy under chosen-plaintext attacks [BDOP04]

- A PEKS scheme is IND-CPA-secure if, for keywords w_0 and w_1 chosen by an adversary:
  - The adversary cannot tell apart the encryption **PEKS**(pk,w_0) of keyword w_0 from the encryption **PEKS**(pk,w_1) of keyword w_1
  - Even when it's allowed to see the trapdoor t_w=**Trapdoor**(sk,w) for keywords $w \neq \{w_0,w_1\}$ of its choice

PEKS-IND-CPA security experiment [BDOP04]



Consistency of PEKS schemes



Consistency	Adversary type	Success prob.
Perfect	Unbounded	0
Statistical	Unbounded	Negligible
Computational	PPT	Negligible

Tools and assumptions

- **Basic tool**: Bilinear maps
  - Let G_1 an additive group of prime order p and generator P
  - Let G_2 be a multiplicative group of prime order p
  - e is said to be a bilinear map $G_1 \times G_1 \rightarrow G_2$ if
    - **bilinear**: $\forall U,V \in G_1$, $\forall a,b \in Z_p$: $e(aU,bV)=e(U,V)^{ab}$
    - **Non-degenerate**: $e(P,P) \neq 1$
    - **Efficiency**: e can be efficiently computed

- **Basic assumption**: BDH assumption
  - Given $P,aP,bP,cP \in G_1$, it's hard to compute $e(P,P)^{abc}$

Outline

- Definitions
- **PEKS constructions**
- Identity-based encryption (IBE)
- IBE-to-PEKS transformations
- Extensions
- Conclusion

The BDOP-PEKS scheme

Key Generation (1^k)
$pk \leftarrow (1^k, P, sP, G_1, G_2, p, e)$
$sk \leftarrow (s, pk)$

Trapdoor (sk, w)
$t_w \leftarrow (pk, sH_1(w))$

PEKS (pk, w)
$r \leftarrow Z_p$
$T \leftarrow e(sP, H_1(w))^r$
$K \leftarrow H_2(T)$
$C \leftarrow (rP, K)$

Test $(t_w, C=(rP,K))$
$T \leftarrow e(rP, sH_1(w))$
$K' \leftarrow H_4(T)$
if $(K'=K)$
then return 1 else return 0

Computational consistency of BDOP-PEKS

- **Theorem**: BDOP-PEKS is computationally consistent in the random oracle model

PEKS-STAT: Our statistically-consistent PEKS

- **Main Idea**: Encryption method depends on keyword length
- Let $f(k) = k^{\log(k)}$ be a function which is super-poly and sub-exp
- **$|w| < f(k)$**
  - Use "highly-injective" random oracles to ensure that **Test**(t_w,**PEKS**(pk,w'))=1 with negligible probability for $w' \neq w$
- **$|w| \geq f(k)$**
  - Encryption returns w
  - Privacy is not affected because f(k) is super-polynomial

The PEKS-STAT Construction

Key Generation (1^k)
$pk \leftarrow (1^k, P, sP, G_1, G_2, p, e)$
$sk \leftarrow (s, pk)$

PEKS (pk, w) $[|w| < f(k)]$
$T \leftarrow e(sP, H_1(w))^r$
$K_1 \leftarrow H_4(T)$
$K \leftarrow \{0,1\}^k$
$c \leftarrow K_1 \oplus K$
$K_2 \leftarrow H_2(T)$
$t \leftarrow H_3(K \| w)$
$C \leftarrow (rP, c, t, K_2)$

Trapdoor (sk, w)
$t_w \leftarrow (pk, sH_1(w), w)$

Test $(t_w, C=(rP,c,t,K_2))$ $[|w| < f(k)]$
$T \leftarrow e(rP, sH_1(w))$
$K_1 \leftarrow H_4(T)$
$K \leftarrow K_1 \oplus c$
$K'_2 \leftarrow H_2(T)$
$t' \leftarrow H_3(K \| w)$
if $(K'_2 = K_2)$ and $(t' = t)$
then return 1 else return 0

Security and consistency of PEKS-STAT

- **Security:**
  PEKS-STAT is IND-CPA-secure in the random oracle model if the BDH assumption holds

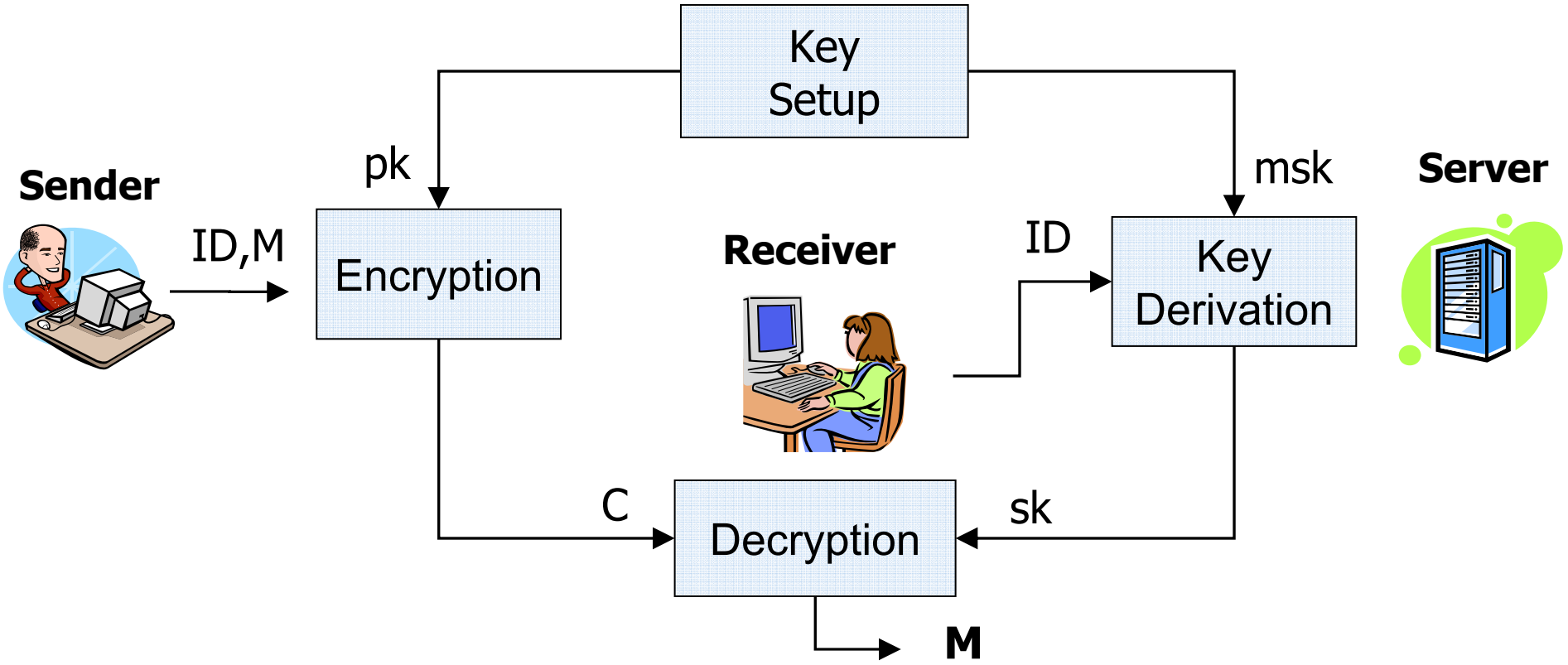
- **Consistency**:
  PEKS-STAT is statistically consistent in the random oracle model

Outline

- Definitions
- PEKS constructions
- **Identity-based encryption (IBE)**
- IBE-to-PEKS transformations
- Extensions
- Conclusion

IBE: Identity-based encryption [Shamir,BF01]

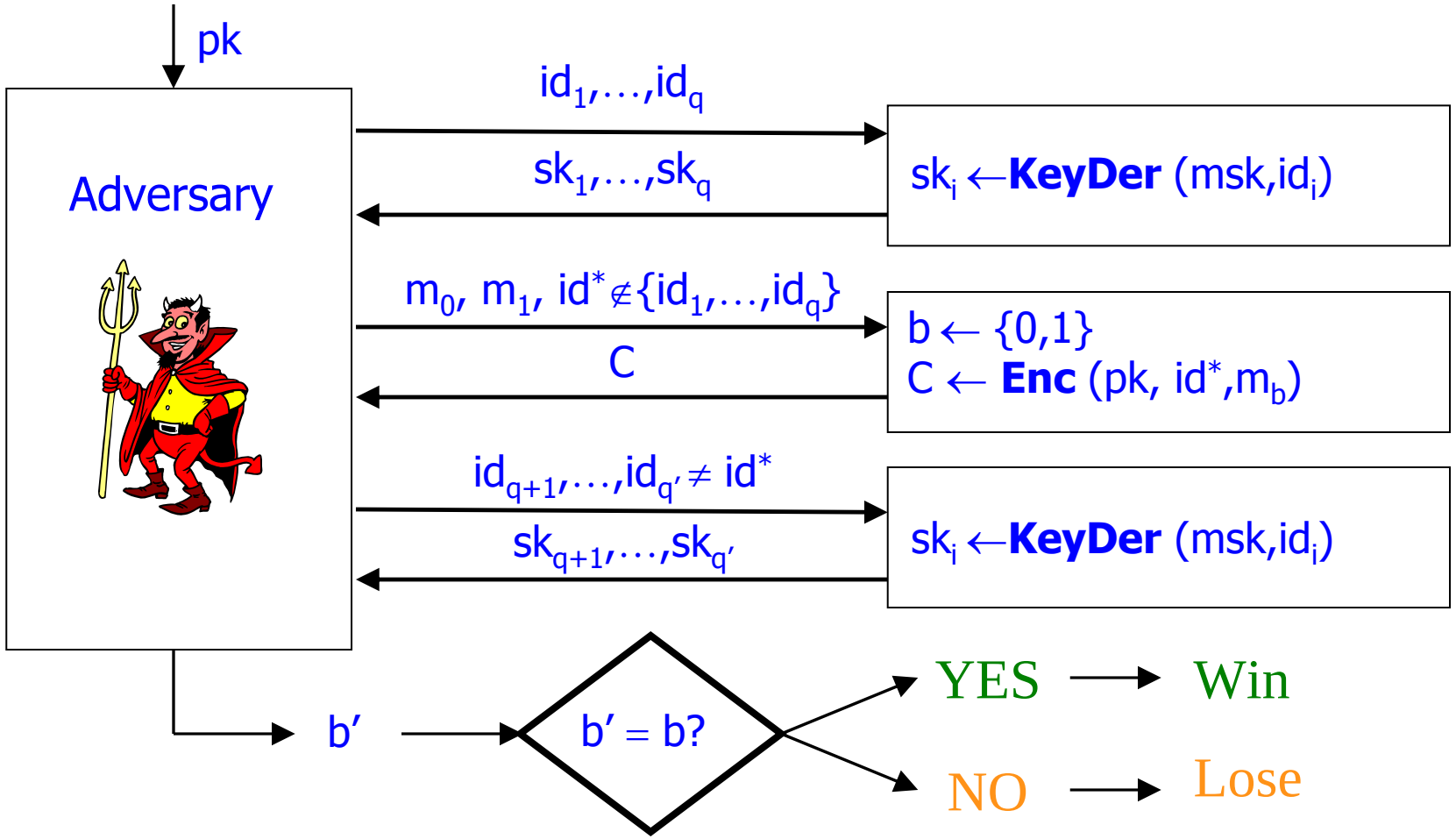
Goal: Allow sender to encrypt messages based on the receiver's identity



IBE-IND-CPA: privacy against chosen-plaintext attack [BF01]

- A scheme is IBE-IND-CPA secure if, for messages M_0 and M_1 and identity ID^* chosen by an adversary:
  - The adversary cannot tell apart the encryption of M_0 from the encryption of M_1 for identity ID^*
  - Even when it's allowed to see secret keys $sk=KeyDerivation(msk,ID)$ for identities $ID \neq ID^*$ of its choice

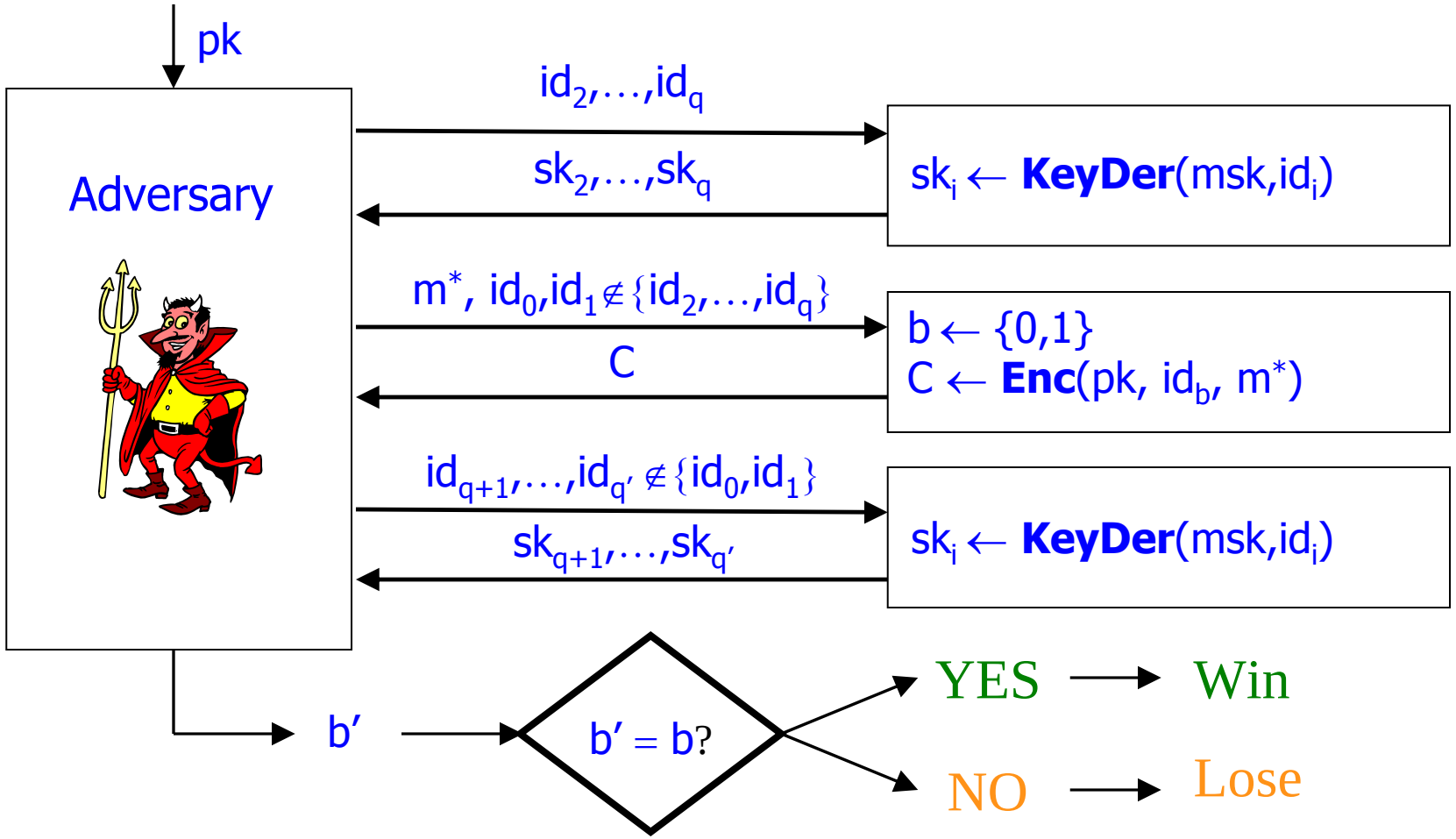
IBE-IND-CPA security experiment [BF01]



Anonymous IBE (ANO-CPA)

- Following [BBDP01], an IBE scheme is ANO-CPA-secure if, for identities ID_0 and ID_1 and message M^* chosen by an adversary:
  - The adversary cannot tell apart the encryption of M^* for identity ID_0 from the encryption of M^* for identity ID_1
  - Even when it's allowed to see secret keys $sk=KeyDerivation(msk,ID)$ for identities $ID \neq \{ID_0, ID_1\}$ of its choice

IBE-ANO-CPA security experiment



Boneh-Franklin Basic IBE scheme

Key Setup (1^k)
$pk \leftarrow (1^k, P, sP, G_1, G_2, p, e)$
$msk \leftarrow (s, pk)$

Key Derivation (msk, ID)
$sk \leftarrow (pk, sH_1(ID))$

Encryption (pk, ID, M)
$T \leftarrow e(sP, H_1(ID))^r$
$K \leftarrow H_2(T)$
$c \leftarrow M \oplus K$
$C \leftarrow (rP, c)$

Decryption (sk, C=(rP,c))
$T \leftarrow e(rP, sH_1(ID))$
$K \leftarrow H_2(T)$
$M \leftarrow K \oplus c$

Anonymity of Boneh-Franklin Basic IBE

- **Theorem**: The Boneh-Franklin Basic IBE scheme is anonymous in the random oracle if the BDH assumption holds.

Proof idea

- Let (m^*, id_0, id_1) be the values returned by the adversary in the challenge phase
- Define sequence of games $G_0,\ldots,G_3$
- G_0: $C \leftarrow \mathbf{Enc}(pk, id_0, m^*)$
- G_1: $C \leftarrow \mathbf{Enc}(pk, id_0, \$)$
- G_2: $C \leftarrow \mathbf{Enc}(pk, id_1, \$)$
- G_3: $C \leftarrow \mathbf{Enc}(pk, id_1, m^*)$

Follows from IND-CPA

Statistically negligible

Follows from IND-CPA

Waters IBE scheme [W05]

Key Generation (1^k)
$(G_1,G_2,p,e) \leftarrow \ldots$
$P,Q \leftarrow G_1$; $E \leftarrow e(P,Q)$
$\mathbf{U}[0,\ldots,N] \leftarrow G_1^{N+1}$
$pk \leftarrow (P,\mathbf{U},E,G_1,G_2,p,e)$
$msk \leftarrow (Q,pk)$

Key Derivation (msk, **ID**)
$r \leftarrow Z_p$
$V \leftarrow \mathbf{U}[0] + \Sigma\, \mathbf{U}[i] \circ \mathbf{ID}[i]$
$sk[\mathbf{ID}] \leftarrow (pk,\ rP,\ rV+Q)$

Encryption (pk, **ID**, M)
$\alpha \leftarrow Z_p$; $T \leftarrow E^{\alpha}$
$V \leftarrow \mathbf{U}[0] + \Sigma\, \mathbf{U}[i] \circ \mathbf{ID}[i]$
$c \leftarrow M \circ T$
$C \leftarrow (c,\ \alpha P,\ \alpha V)$

Decryption (sk, C)
$T \leftarrow e(\alpha P, rV+Q) \,/\, e(rP, \alpha V)$
$M \leftarrow c \,/\, T$

Anonymity of Waters IBE scheme

- **Theorem**: The Waters IBE scheme is NOT anonymous.

- **Proof**: We can check which identity was encrypted via the bilinear map

  - Choose $\mathbf{M}$, $\mathbf{ID_0}$, and $\mathbf{ID_1 \neq ID_0}$ and return $\mathbf{(M, ID_0, ID_1)}$
  - Let $\mathbf{C = (C_1, C_2 = \alpha P, C_3 = \alpha V_b)}$ where $\mathbf{V_b \leftarrow U[0] + \Sigma\, U[i] \circ ID_b[i]}$
  - If $\mathbf{e(C_2, V_0) = e(C_3, P)}$ then return 0 else return 1

Outline

- Definitions
- PEKS constructions
- Identity-based encryption (IBE)
- **IBE-to-PEKS transformations**
- Extensions
- Conclusion

An IBE-2-PEKS transformation [BDOP04]

PEKS = IBE-2-PEKS[**IBE**] (**KeyGen, Trapdoor, PEKS, Test**)	**IBE** (**Setup, KeyDer, Enc, Dec**)
pk	pk
sk	msk
Keyword **w**	Identity **w**
Trapdoor $t_{\mathbf{w}}$	User secret key sk[**w**]
PEKS (pk, **w**)	C $\leftarrow$ **Enc** (pk, **w**, $\mathbf{0}^k$)
Test ($t_{\mathbf{w}}$, C)	**Dec** (sk[**w**], C)= $\mathbf{0}^k$?

Consistency of IBE-2-PEKS transformation

If the underlying **IBE** is ANO-CPA-secure, then **PEKS** = IBE-2-PEKS[**IBE**] is IND-CPA-secure, but …

- **Theorem**: There exist ANO-CPA and IND-CPA **IBE** schemes for which **PEKS** = IBE-2-PEKS[**IBE**] is NOT computationally consistent

The NEW-IBE-2-PEKS transformation

PEKS = NEW-IBE-2-PEKS[**IBE**] (**KeyGen, Trapdoor, PEKS, Test**)	**IBE** (**Setup, KeyDer, Enc, Dec**)
pk	pk
sk	msk
Keyword **w**	Identity **w**
Trapdoor $t_{\mathbf{w}}$	User secret key sk[**w**]
PEKS (pk, **w**)	$C_1 \leftarrow \{0,1\}^k$; $C_2 \leftarrow$ **Enc** (pk, **w**, C_1)
Test ($t_{\mathbf{w}}$, (C_1,C_2))	**Dec** (sk[**w**], C_2)= C_1 **?**

Security and consistency of new transformation

- **Theorem 1**: If **IBE** is ANO-CPA-secure, then **PEKS**=NEW-IBE-2-PEKS[**IBE**] is IND-CPA-secure.

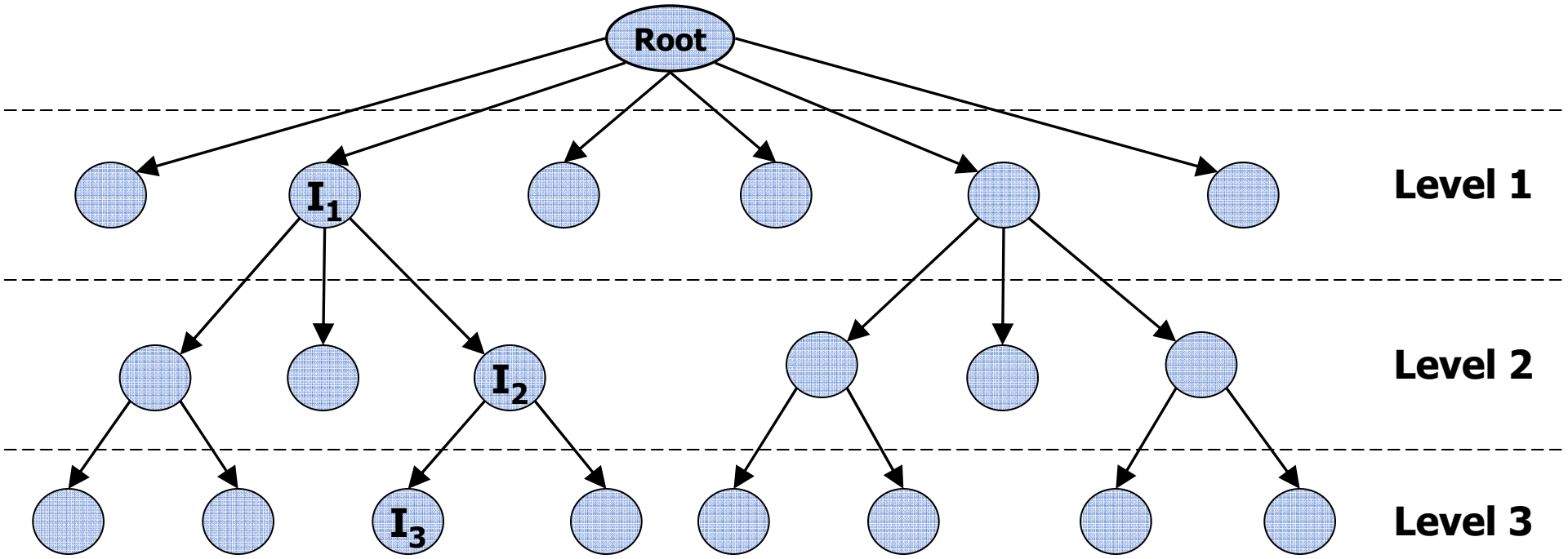
- **Theorem 2**: If **IBE** is IND-CPA-secure, then **PEKS**=NEW-IBE-2-PEKS[**IBE**] is computationally consistent.

Outline

- Definitions
- PEKS constructions
- IBE-to-PEKS transformations
- **Extensions**
- Conclusion

Hierarchical IBE (HIBE) [HL02,GS02]

Generalization of IBE schemes for hierarchical structures

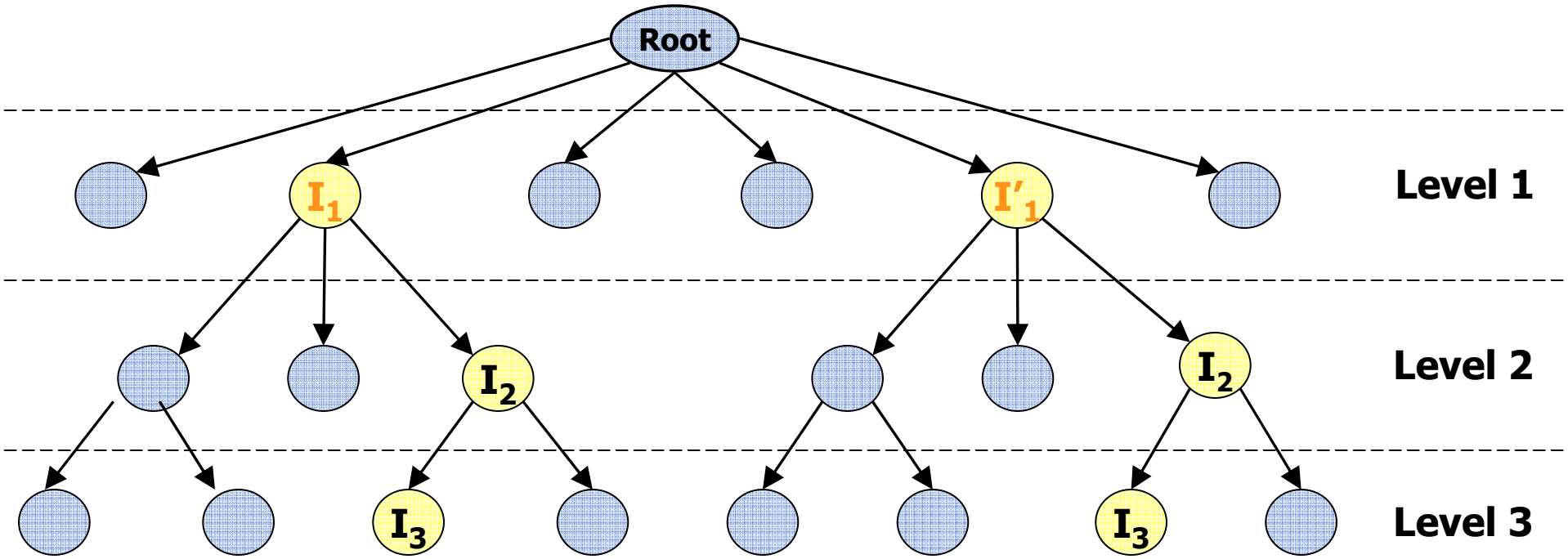


$ID = (I_1, I_2, I_3)$

Anonymous HIBE

- Anonymity based on levels
- An HIBE is anonymous at level L if
  - The adversary cannot tell apart the encryption of M for identity ID_0 from the encryption of M for identity ID_1
  - ID_0 and ID_1 are vectors that differ only in the L-th component

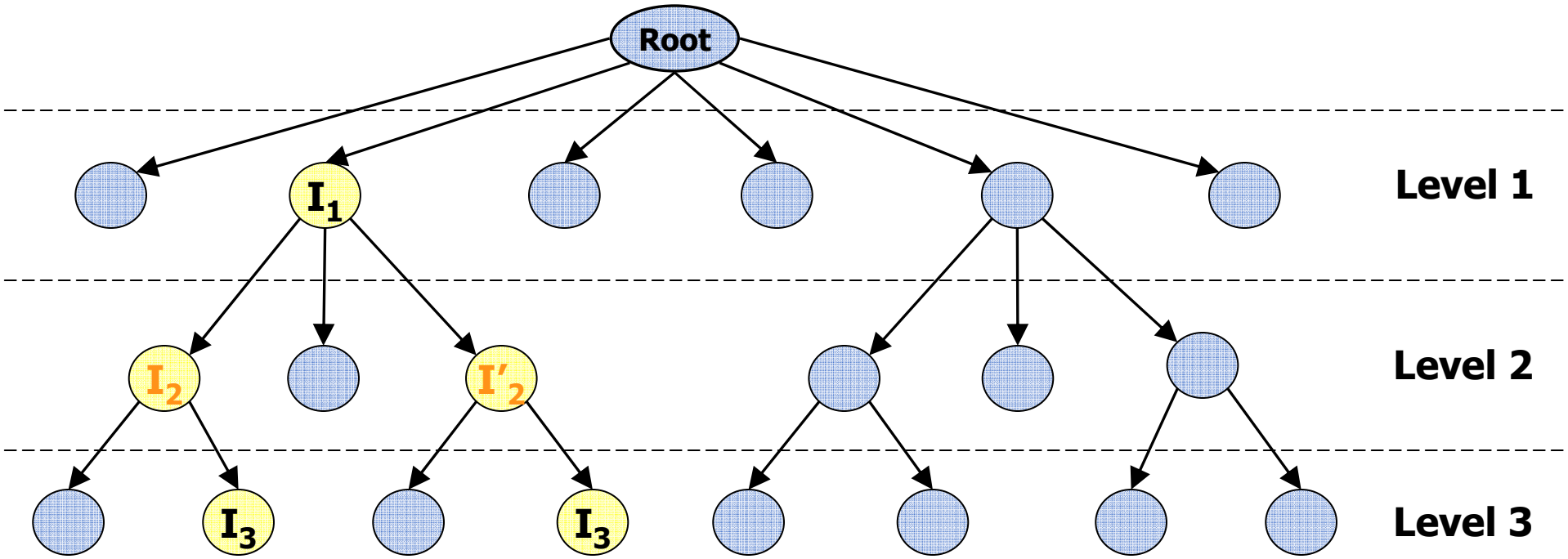
Level-1 Anonymous HIBE



$ID_0 = (I_1, I_2, I_3)$

$ID_1 = (I'_1, I_2, I_3)$

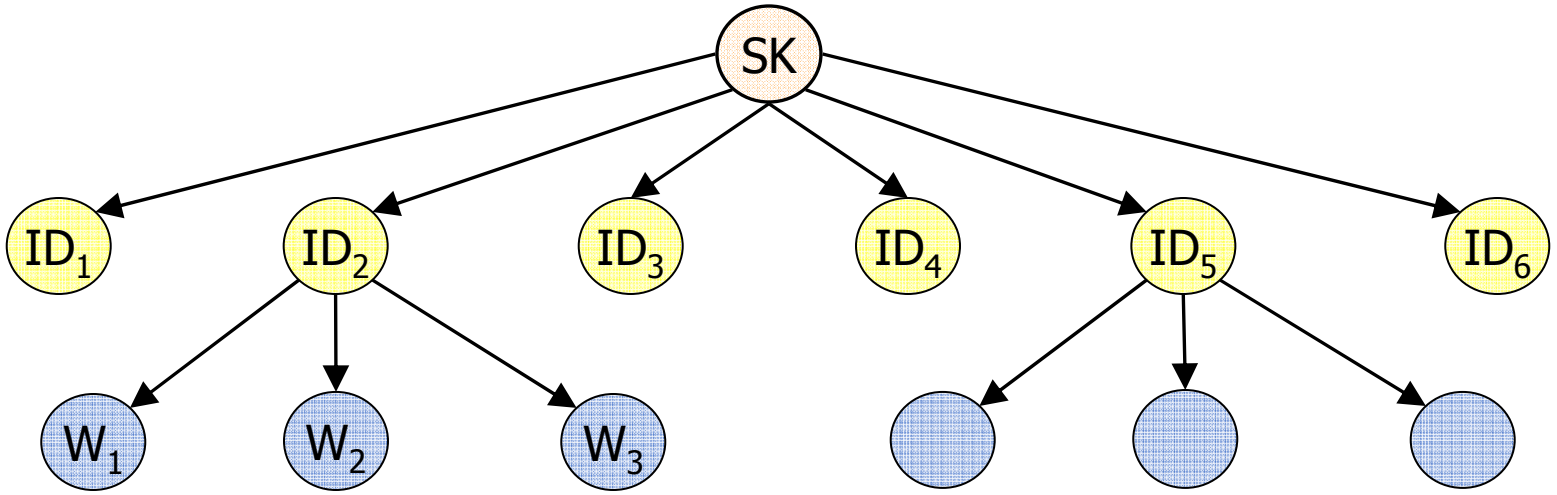
Level-2 Anonymous HIBE



$ID_0=(I_1,I_2,I_3)$ $ID_1=(I_1,I'_2,I_3)$

IBEKS: Identity-based encryption with keyword search

- **Idea**: Combine the concepts of IBE and PEKS
- **Generic construction from Hierarchical IBE**:
  - Identities at level 1
  - Keywords at level 2



The HIBE-2-IBEKS transformation

IBEKS = HIBE-2-IBEKS[**HIBE**] (**KeyGen, KeyDer, Trapd, IBEKS, Test**)	**HIBE** (**Setup, KeyDer, Enc, Dec**)
pk	pk
msk	msk
Identity **ID**	Identity **ID** at level 1
Keyword **w**	Identity **w** at level 2
User secret key sk[**ID**]	sk[**ID**]
Trapdoor $t_{w,ID}$ for keyword **w** and user **ID**	sk[**ID**,**w**]
IBEKS (pk, **ID**, **w**)	$C_1 \leftarrow \{0,1\}^k$; $C_2 \leftarrow$ **Enc** (pk, (**ID**, **w**), C_1)
Test ($t_{w,ID}$, (C_1,C_2))	**Dec** (sk[**ID**,**w**], C_2)= C_1 **?**

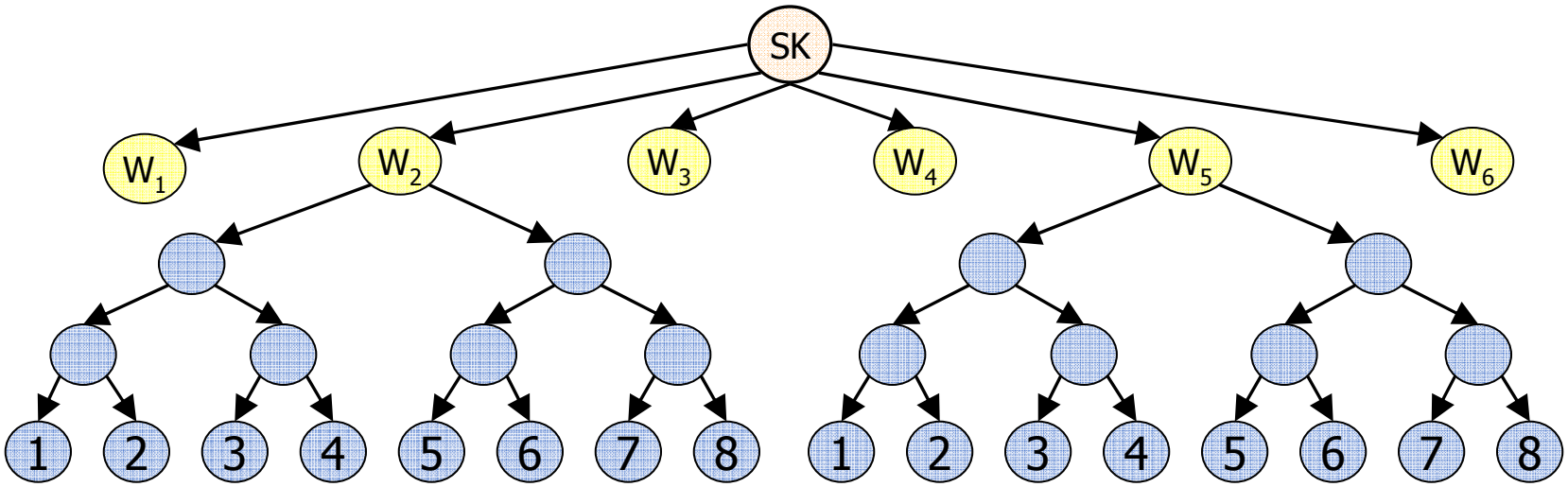
Security and consistency of HIBE-2-IBEKS transformation

- **Security:**
  If HIBE is anonymous at level 2,
  then IBEKS is IND-CPA-secure

- **Consistency**:
  If HIBE is IND-CPA-secure,
  then IBEKS is computationally consistent

PETKS: Public-key encryption with temporary keyword search

- **Idea**: Allow the testing of a keyword w across multiple time periods using a single temporary trapdoor for that interval
- **Generic construction from HIBE schemes**:
  - Keywords at level 1
  - Binary tree of time periods at levels 2..d [CHK03,BM99]



The HIBE-2-PETKS transformation

PETKS = HIBE-2-PETKS[**HIBE**] (**KeyGen, Trapdoor, PETKS, Test**)	**HIBE** (**Setup, KeyDer, Enc, Dec**)
pk	pk
sk	msk
Keyword **w**	Identity **w** at level 1
Time period **j**	Identity **j** at level d
Trapdoor $t_{\mathbf{w}}$**[s,e]** for keyword **w** and time interval **[s,e]**	secret key for nodes of the binary tree rooted at **w** corresponding to interval **[s,e]**
PETKS (pk, **w**, **j**)	$C_1 \leftarrow \{0,1\}^k$; $C_2 \leftarrow$ **Enc** (pk, (**w**, ⟨**j**⟩), C_1)
Test ($t_{\mathbf{w}}$**[s,e]**, (C_1,C_2))	**Dec** (sk[(**w**, ⟨**j**⟩)], C_2)= C_1 **?**

Security and consistency of HIBE-2-PETKS transformation

- **Security:**
  If HIBE is anonymous at level 1,
  then PETKS is IND-CPA-secure

- **Consistency**:
  If HIBE is IND-CPA-secure,
  then PETKS is computationally consistent

Instantiations

- **Anonymous IBE** (for basic PEKS)
  - Boneh-Franklin Basic IBE in the ROM [BF01]
- **HIBE anonymous at level 1** (for PETKS)
  - Modified version of GS-HIBE in the ROM [GS02]
- **HIBE anonymous at level 2** (for IBEKS):
  - No known instantiations even in the ROM

PEKS: Open problems

- More efficient constructions
- Other extensions:
  - Search using more expressive formulas
  - Fuzzy PEKS

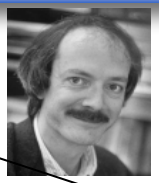
Identity-based encryption with wildcard key derivation

Identity-based encryption with wildcards (WIBE)

- Identities are vectors ($ID_1,\ldots,ID_L$)
- Hierarchical key derivation
- Encryption: receiver identity can contain "wildcards"
- Decryption by any "matching" identity

  e.g. C = Enc(mpk, (ID_1,$\star$,ID_3), M) can be decrypted by any (id_1,id_2,id_3) where id_1=ID_1 and id_3=ID_3
- ... but by nobody else

Usage example (1)



ECRYPT

STVL

AZTEC

PROVILAB

WG1

WG2

WG3

WG4

Michel

Dario

Alex

John

Greg

Usage example (1)

To: ECRYPT.*

ogether!

To: ECRYPT.AZTEC.*

We have to organize meetings to do research together.

S

To: ECRYPT.AZTEC.WG1.*

We have to meet in Porto in July to do research together.

PROVILAB

WG1

WG2

WG3

WG4

ALRIGHT!!!

Michel

Dario

Alex

John

Greg

Usage example (2)

Structured email addresses name@dept.univ.edu

Send identity-based encrypted email to

- individual users: JohnSmith@cs.univ.edu
- computer science department: ✶@cs.univ.edu
- entire university: ✶@✶.univ.edu
- all computer science departments: ✶@cs.✶.edu
- all sysadmins: sysadmin@✶.univ.edu
- spammers' dream: ✶@✶.✶.✶

Generic construction from any HIBE

- Given HIBE = (Setup, KeyDer, Enc, Dec)
  Consider WIBE = (Setup, KeyDer', Enc', Dec'):
  - KeyDer':

    special wildcard string "$\star$"

    $sk'_{(ID1,ID2)} = \{ sk_{(ID1,ID2)}, sk_{(\text{“}\star\text{”},ID2)}, sk_{(ID1,\text{“}\star\text{”})}, sk_{(\text{“}\star\text{”},\text{“}\star\text{”})} \}$
  - Enc':

    Enc substituting "$\star$" for each wildcard
  - Dec':

    select correct key from list and apply Dec
- Major drawback: $|sk| = O(2^{\ell})$
- Schemes with efficiency polynomial in all parameters?

Waters' HIBE scheme

- Setup:
  Let L = max hierarchy depth, n = identity bit length
  $g_1, g_2 \leftarrow G$; $\alpha \leftarrow Z_p$; $h_1 \leftarrow g_1^{\alpha}$; $h_2 \leftarrow g_2^{\alpha}$
  For i = 1,…,L and j = 0,…,n do $u_{i,j} \leftarrow G$
  $mpk \leftarrow (g_1, g_2, h_1, u_{1,0},\ldots,u_{L,n})$; $msk \leftarrow h_2$

- Enc(mpk, $(ID_1,\ldots,ID_\ell)$, M):
  Let $ID_i = ID_{i,1},\ldots,ID_{i,n}$; Let $H_i(ID_i) = u_{i,0} \prod_{ID_{i,j}=1} u_{i,j}$
  $t \leftarrow Z_p$
  $C_1 \leftarrow g_1^t$
  $C_2 \leftarrow (C_{2,i})_{i=1,\ldots,\ell}$ where $C_{2,i} = H_i(ID_i)^t$
  $C_3 \leftarrow M \cdot e(h_1,g_2)^t$
  Return $C = (C_1, C_2, C_3)$

- Key derivation and decryption: also work ☺

Waters WIBE scheme

- Setup:

  Let L = max hierarchy depth, n = identity bit length $

  $g_1, g_2 \leftarrow G$; $\alpha \leftarrow Z_p$; $h_1 \leftarrow g_1^{\alpha}$; $h_2 \leftarrow g_2^{\alpha}$

  For $i = 1,\ldots,L$ and $j = 0,\ldots,n$ do $u_{i,j} \leftarrow G$

  $mpk \leftarrow (g_1, g_2, h_1, u_{1,0},\ldots,u_{L,n})$; $msk \leftarrow h_2$

- Enc(mpk, $(ID_1,\ldots,ID_\ell)$, M):

  Let $ID_i = ID_{i,1},\ldots,ID_{i,n}$; Let $H_i(ID_i) = u_{i,0} \prod_{ID_{i,j}=1} u_{i,j}$

  $t \leftarrow Z_p$

  $C_1 \leftarrow g_1^t$

  $C_2 \leftarrow (C_{2,i})_{i=1,\ldots,\ell}$ where $C_{2,i} = H_i(ID_i)^t$ if $ID_i \neq \star$

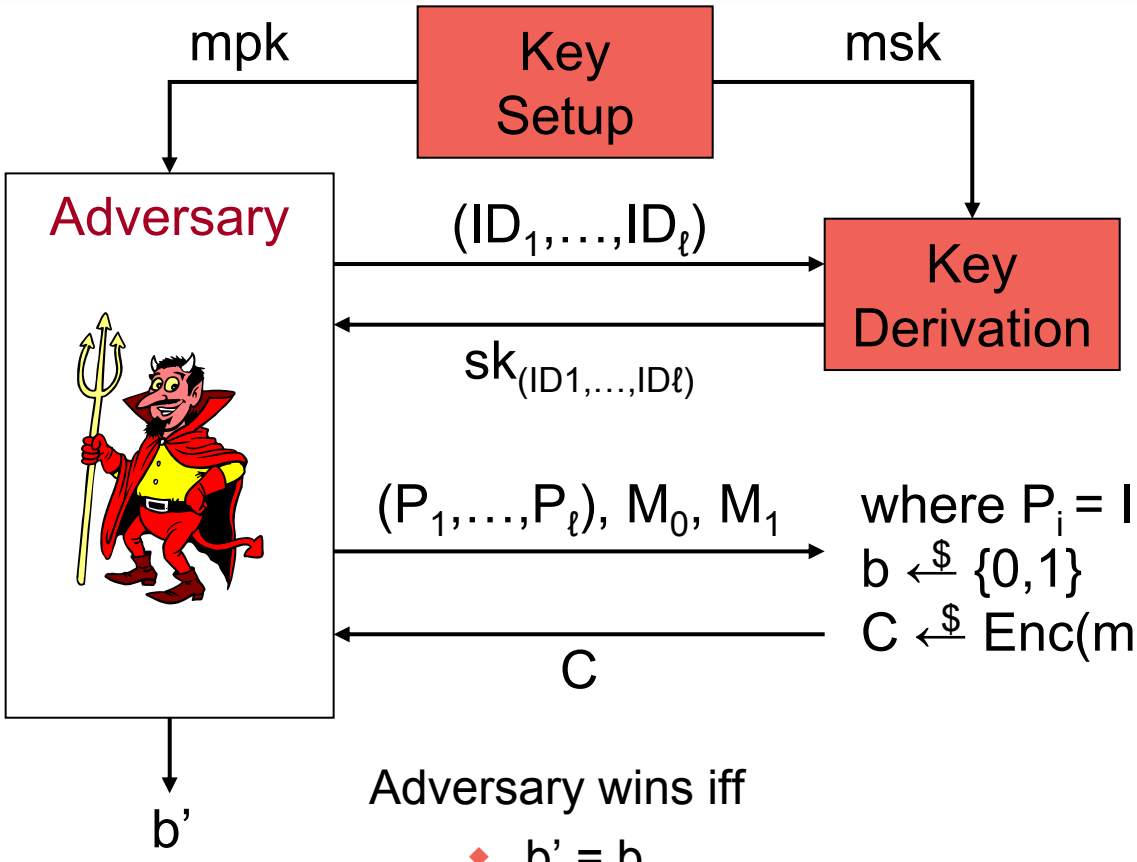
  $= (C_{2,i,j} = u_{i,j}^t)_{j=0,\ldots,n}$ if $ID_i = \star$

  $C_3 \leftarrow M \cdot e(h_1,g_2)^t$

  Return $C = (C_1, C_2, C_3)$

- Decryption: recompute $C_{2,i} = C_{2,i,0} \prod_{ID_{i,j}=1} C_{2,i,j}$ if $ID_i = \star$

Security notion: IND-WID-CPA



Adversary wins iff

- b' = b
- never queried key of (any ancestor of) any identity matching $(P_1,\ldots,P_\ell)$

Security of Waters WIBE

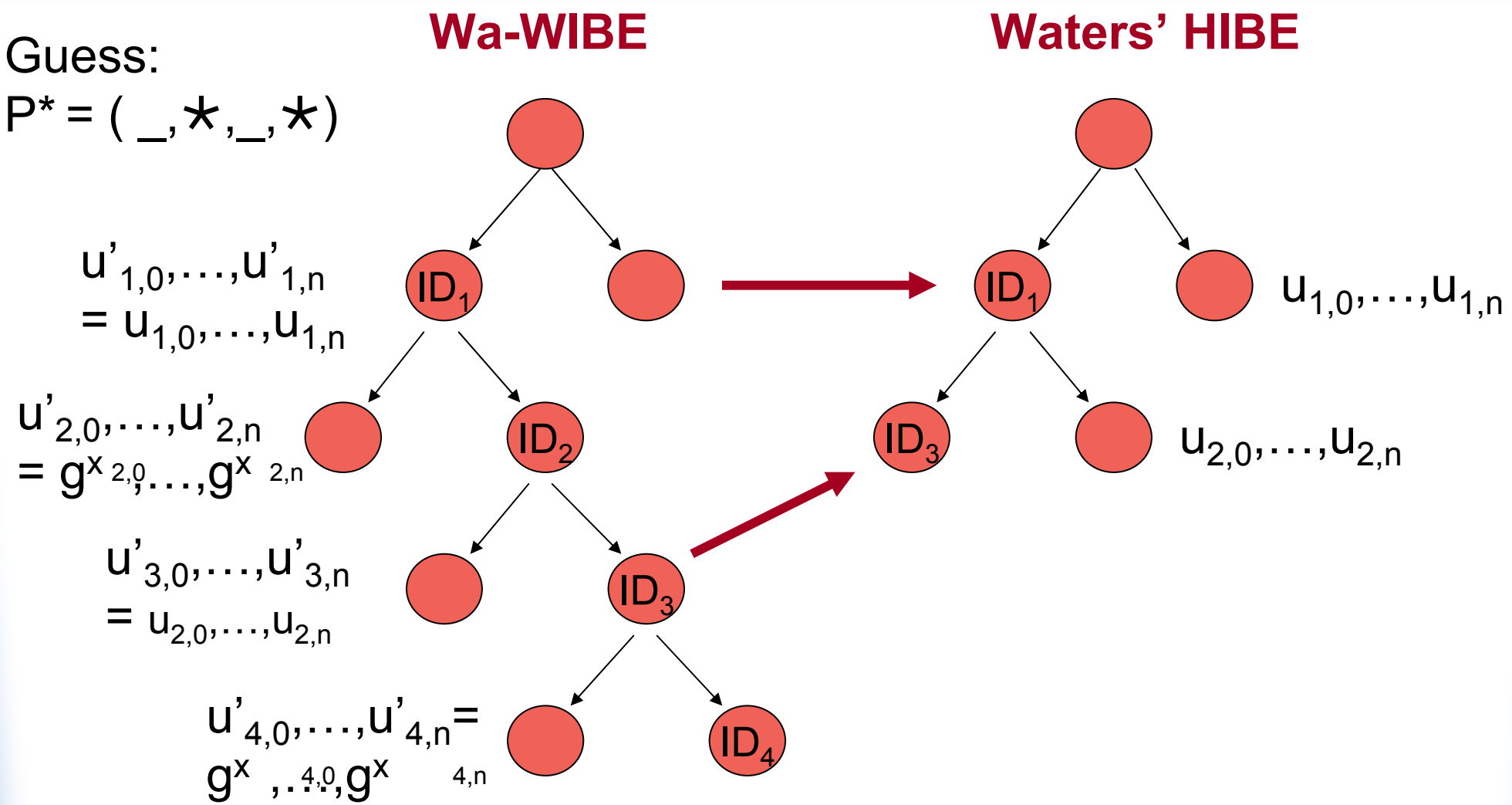
Theorem:

If Waters' HIBE is (t, q_K, ε) IND-HID-CPA secure,
then Waters WIBE is (t', q'_K, ε') IND-WID-CPA secure,
where $\varepsilon' \geq \varepsilon/2^L$, $q_K = q'_K$ and $t' = t + nL(1+q_K)\cdot t_{exp}$

Theorem [Wa05]:

If the BDDH problem is (t,ε)-hard
then Waters' HIBE is (t', q'_K, ε') IND-HID-CPA secure,
where $\varepsilon' \geq O(\varepsilon/(nq_K)^L)$ and $t' = O(t) + \ldots$

Proof idea


Wa-WIBE
Waters' HIBE
Guess:
P* = (_,*,_,*)
u'1,0,…,u'1,n
= u1,0,…,u1,n
u'2,0,…,u'2,n
= gx 2,0,…,gx 2,n
u'3,0,…,u'3,n
= u2,0,…,u2,n
u'4,0,…,u'4,n=
gx 4,0,…,gx 4,n
ID1
ID2
ID3
ID4
ID1
ID3
u1,0,…,u1,n
u2,0,…,u2,n

Alternative schemes

Scheme based on	|mpk| *# elems*	|sk| *# elems*	|C| *# elems*	Dec *# pairings*	Assmptn	RO?
any HIBE	$\|mpk_{HIBE}\|$	$2^L \|sk_{HIBE}\|$	$\|C_{HIBE}\|$	Dec_{HIBE}	IND_{HIBE}	No
Waters	(n+1)L+3	L+1	(n+1)L+2	L+1	BDDH	No
BB	2L+3	L+1	2L+2	L+1	BDDH	Yes
BBG	L+4	L+2	L+3	2	L-BDHI	Yes

L = maximal hierarchy depth ; n = identity length (bits)

Identity-based encryption with wildcards

Wildcard key derivation

- Limited key delegation [BBG05]: restrict depth

  e.g. (edu,univ,cs,⋆) can derive ⋆@cs.univ.edu,

  but not ⋆@⋆.cs.univ.edu

- Generalization: wildcards anywhere

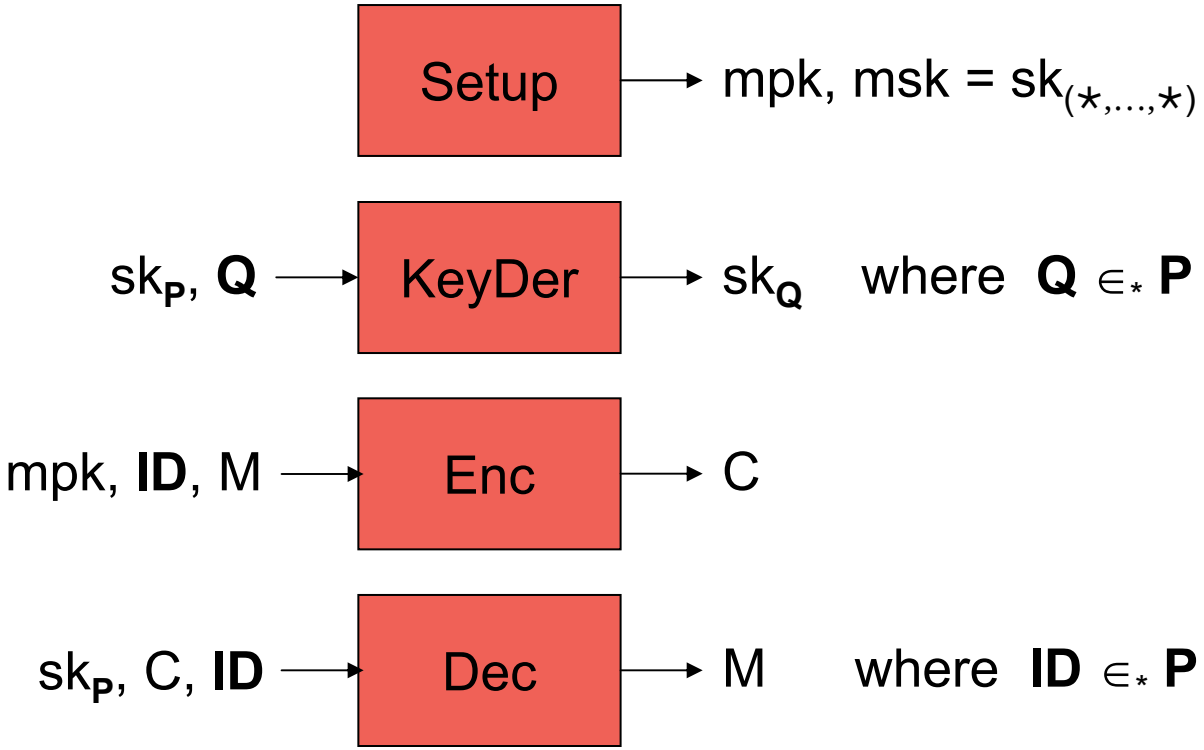
  e.g. sysadmin@⋆.univ.edu

  ⋆@google.⋆

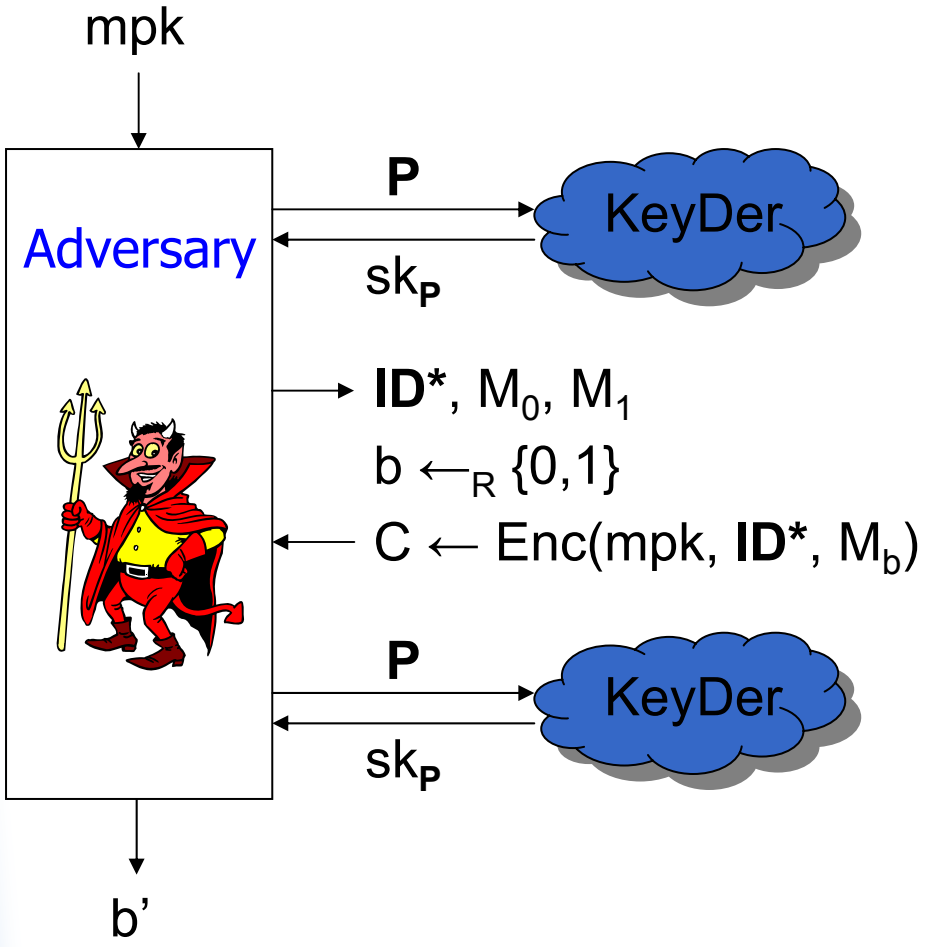
- IBE with wildcard key derivation (WKD-IBE)
  or **“wicked” IBE**

Wicked IBE (WKD-IBE)

- Pattern **P** = $(P_1,\ldots,P_\lambda)$ where $1 \le \lambda \le L$, $P_i \in \{0,1\}^* \cup \{*\}$
- Natural matching definition, denoted **Q** $\in_*$ **P**



Security of wicked IBE



- A wins iff
  - b' = b
  - never queried **P** such that **ID*** $\in_*$ **P**
- WKD-IBE is CPA-secure if no PPT adversary wins with non-negligible prob.
- Selective-identity: Adversary commits to **ID*** before seeing mpk.

BBG HIBE scheme [BBG05]

Key Generation (1^k)

$(G_1,G_2,p,e) \leftarrow \ldots$

$g, g_2, g_3, h_1,\ldots,h_L \leftarrow G_1^{L+3}$

$\alpha \leftarrow Z_p$; $g_1 \leftarrow g^\alpha$; $g_4 \leftarrow g_2^\alpha$

$pk \leftarrow (g,g_1,g_2,g_3,\mathbf{h},G_1,G_2,p,e)$

$msk \leftarrow (\alpha,pk)$

Key Derivation (msk, $\mathbf{ID=(I_1,\ldots,I_\lambda)}$)

$r \leftarrow Z_p$

$a_1 \leftarrow g^r$

$a_2 \leftarrow g_4 (g_3 \prod h_i^{\mathbf{I}[i]})^r$

$\mathbf{b} \leftarrow \{h_i^r\}_{i=\lambda+1,\ldots,L}$

$sk[\mathbf{ID}] \leftarrow (pk, a_1, a_2, \mathbf{b})$

Encryption (pk, **ID**, M)

$t \leftarrow Z_p$;

$c_1 \leftarrow g^t$; $c_2 \leftarrow (g_3 \prod h_i^{\mathbf{I}[i]})^t$

$T \leftarrow e(g_1,g_2)^t$

$c \leftarrow M \circ T$

$C \leftarrow (c, c_1, c_2)$

Decryption (sk, C)

$T \leftarrow e(c_1,a_2) / e(a_2,c_1)$

$M \leftarrow c / T$

Wicked IBE from BBG HIBE

Key Generation (1^k)

$(G_1,G_2,p,e) \leftarrow \ldots$

$g, g_2, g_3, h_1,\ldots,h_L \leftarrow G_1^{L+3}$

$\alpha \leftarrow Z_p$; $g_1 \leftarrow g^\alpha$; $g_4 \leftarrow g_2^\alpha$

$pk \leftarrow (g,g_1,g_2,g_3,\mathbf{h},G_1,G_2,p,e)$

$msk \leftarrow (\alpha,pk)$

Key Derivation (msk, $\mathbf{P=(P_1,\ldots,P_\lambda)}$)

$r \leftarrow Z_p$

$a_1 \leftarrow g^r$

$a_2 \leftarrow g_4 (g_3 \prod h_i^{\mathbf{P}[i]})^r$ $\quad (\mathbf{P_i \neq *})$

$\mathbf{b} \leftarrow \{h_i^r\}$ $\quad (\mathbf{P_i = *})$

$sk[\mathbf{P}] \leftarrow (pk, a_1, a_2, \mathbf{b})$

Encryption (pk, **ID**, M)

$t \leftarrow Z_p$;

$c_1 \leftarrow g^t$; $\quad c_2 \leftarrow (g_3 \prod h_i^{I[i]})^t$

$T \leftarrow e(g_1,g_2)^t$

$c \leftarrow M \circ T$

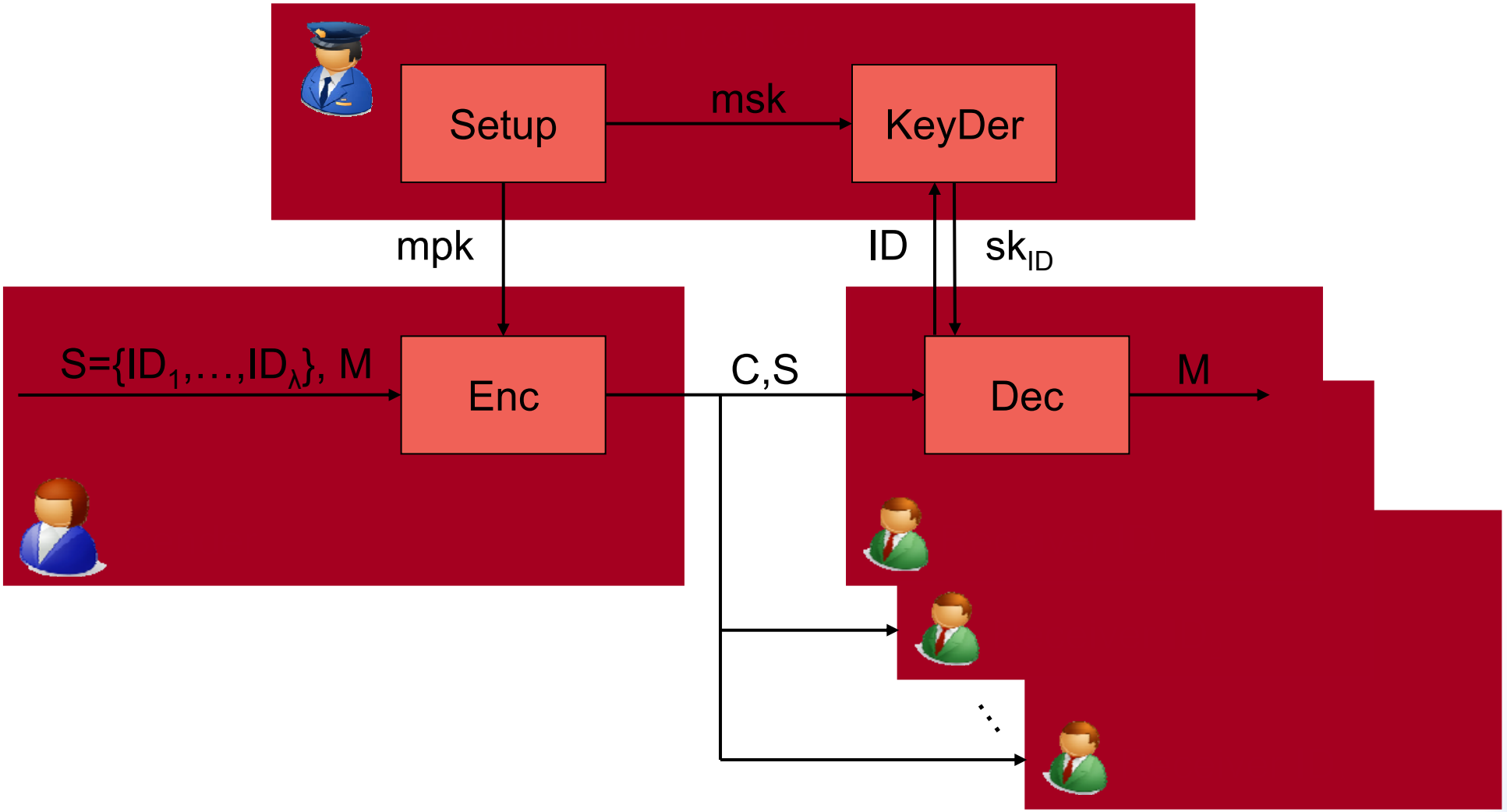
$C \leftarrow (c, c_1, c_2)$

Decryption (sk, C)

$T \leftarrow e(c_1,a_2) / e(a_2,c_1)$

$M \leftarrow c / T$

Identity-based broadcast encryption (IBBE)



IBBE: A trivial construction

- Given any IBE = (Setup, KeyDer, Enc, Dec), construct IBBE = (Setup, Keyder, BEnc, BDec) by concatenating ciphertexts:

  BEnc(mpk, $S = \{ID_1,\ldots,ID_\lambda\}$, M):

  For i = 1,…, λ do $C_i \leftarrow_R Enc(mpk,ID_i,M)$

  $\mathbf{C} \leftarrow (C_1,\ldots,C_\lambda)$

  BDec(sk_{ID}, C, $S = \{ID_1,\ldots,ID_\lambda\}$):

  $M \leftarrow Dec(sk_{ID}, C_i)$ where i such that $ID_i = ID$

  ☹ ciphertext length O(λ)

- Goal: outperform trivial construction

IBBE: Construction from any WKD-IBE

- Given any WKD-IBE = (Setup, WKeyDer, WEnc, WDec) consider IBBE = (Setup, BKeyDer, BEnc, BDec) where

  BKeyDer(msk, ID):

  For i = 1,…,L do $wsk_i \leftarrow \text{WKeyDer}(msk, (*,\ldots,*,\overleftrightarrow{i\text{-}1}\text{ID}, *,\ldots,*))$

  $sk_{ID} \leftarrow (wsk_1,\ldots,wsk_L)$

  BEnc(mpk, S = $\{ID_1,\ldots,ID_\lambda\}$, M):

  $C \leftarrow_R \text{WEnc}(mpk, (ID_1,\ldots,ID_\lambda), M)$

  BDec(sk_{ID}, C, S = $\{ID_1,\ldots,ID_\lambda\}$):

  Find i such that $ID_i = ID$

  $M \leftarrow \text{WDec}(wsk_i, C)$

- When instantiated with BBG scheme: ciphertext size O(1), independent of L secret key size $O(L^2)$

Wicked and wildcard signatures

- Wicked signatures
  wildcard key delegation for ID-based signatures
  L-level WKD-IBS from any (L+1)-level WKD-IBE
  (using extension of Naor's observation for IBE)

- Wildcard signatures
  message being signed contains wildcards
  wildcards can be instantiated without invalidating signature
  application: signed fill-out forms, limited signing delegation
  e.g. "State X certifies that person ⋆ has the right to drive a car."

- Wicked wildcard signatures

Other extensions

Attribute based encryption (ABE)

- Extension of identity-based encryption
  - Secret keys and ciphertexts are associated with a set of attributes instead of identities
- Two possible variations
  - Key policy ABE
    - Ciphertexts are associated with a set of attributes
    - Secret keys are associated with access structures
  - CIphertext policy ABE
    - The other way around
- Applications
  - Identity-based encryption based on biometrics (Fuzzy IBE)

Acknowledgements

- Some of the slides used in theses lectures were provided by Sara Miner (University of California at San Diego), Gregory Neven (K.U. Leuven), and David Pointcheval (Ecole normale superieure, Paris).