MITACS 2009

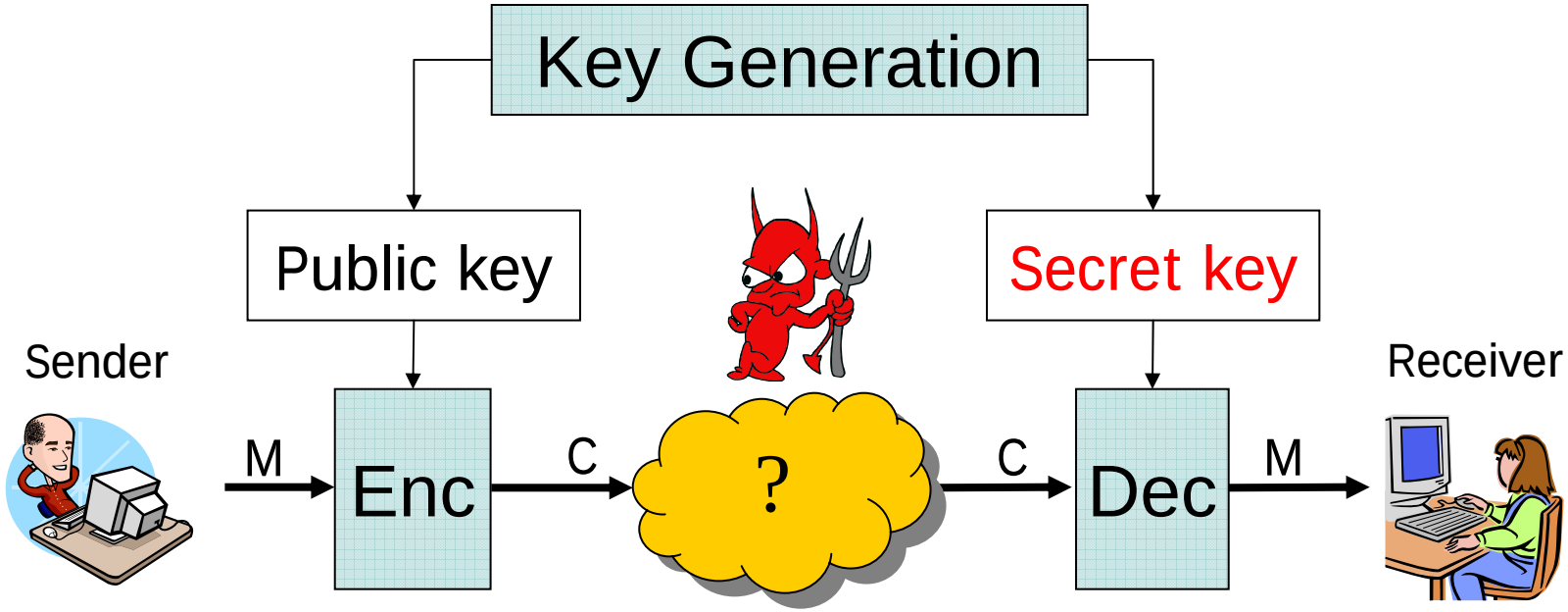
« Robust Public-key and Identity-based Encryption »

26 June 2009

Michel Abdalla
École normale supérieure & CNRS

Joint work with
Mihir Bellare, Chanathip Namprempre, Gregory Neven

Public-key encryption (PKE)



Security goals for PKE

- **Data Privacy**
  - Ciphertext should not reveal any partial information about the encrypted message
- **Key privacy** (a.k.a. anonymity)
  - Ciphertext should not reveal any partial information about the public key under which it was created

A practical scenario

- Suppose C is a ciphertext obtained by encrypting a message M under public key pk
- If C is decrypted using the secret key sk corresponding to pk, then the result is M
- However, what happens if C is decrypted using the secret key sk' corresponding to pk'$\neq$ pk?
- **Robustness**: The decryption algorithm should reject whenever the wrong decryption key is used

Why robustness?

- The primary security requirement for public key encryption is data privacy
- However, a growing number of applications (e.g., anonymous channels, electronic voting) also requires anonymity
- **Our thesis**: Anonymity without robustness is inadequate for most applications

Example 1: Auction protocol

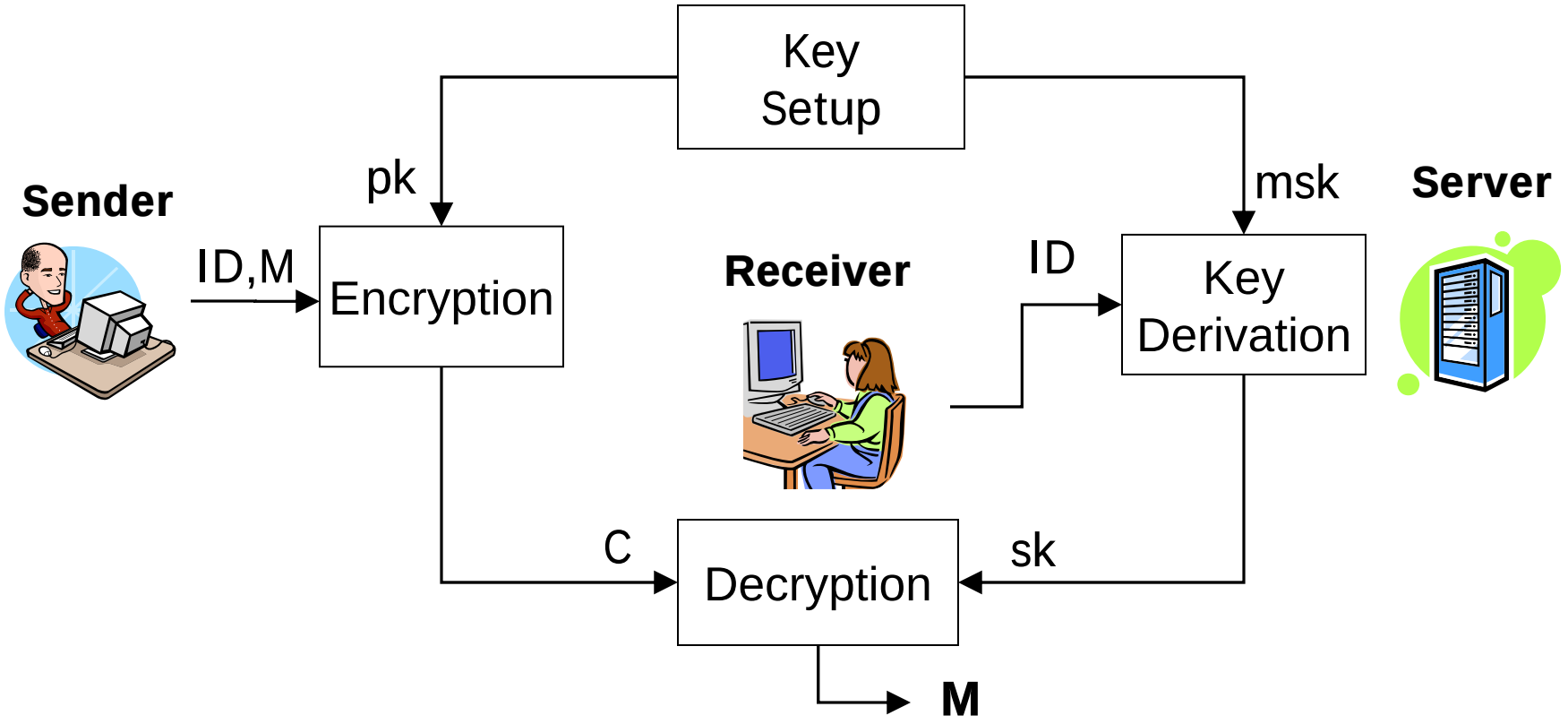
- Overall goal
  - Simulate a real-life auction based on sealed envelopes
- Correctness
  - The highest bid should be the winning bid
- Security goals
  - Only the highest bid should be revealed
  - The losing bids should remain secret
- Fairness
  - The scheme should remain secure even in the case of collusions between an auctioneer and a bidder.

Example 1: Auction protocol [Sako2000]

- Setup
  - Secret Key: $v_1,\ldots,v_N \in Z_p$
  - Public Key: $g, X_1=g^{v1}, \ldots, X_N=g^{vN}, M$
- Bidding on a value $v \in \{1,\ldots,N\}$
  - $C = Enc(X_v,M) = (g^r,(X_v)^r M)$
- Opening bids $(C_1,\ldots,C_L)$
  - Set i=N and S = {}
  - For j=1,…,L, if $Dec(C_j)=M$, then $S = S \cup \{i\}$
  - If S = {}, then i = i-1

Example 2: Identity-based encryption [Shamir,BF01]

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



Example 3: Searchable Encryption [BDOP04]

- 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 (PEKS)**: Enable gateway to test whether a given keyword is present in the email without learning anything else about the email

Example 3: Searchable Encryption [BDOP04]

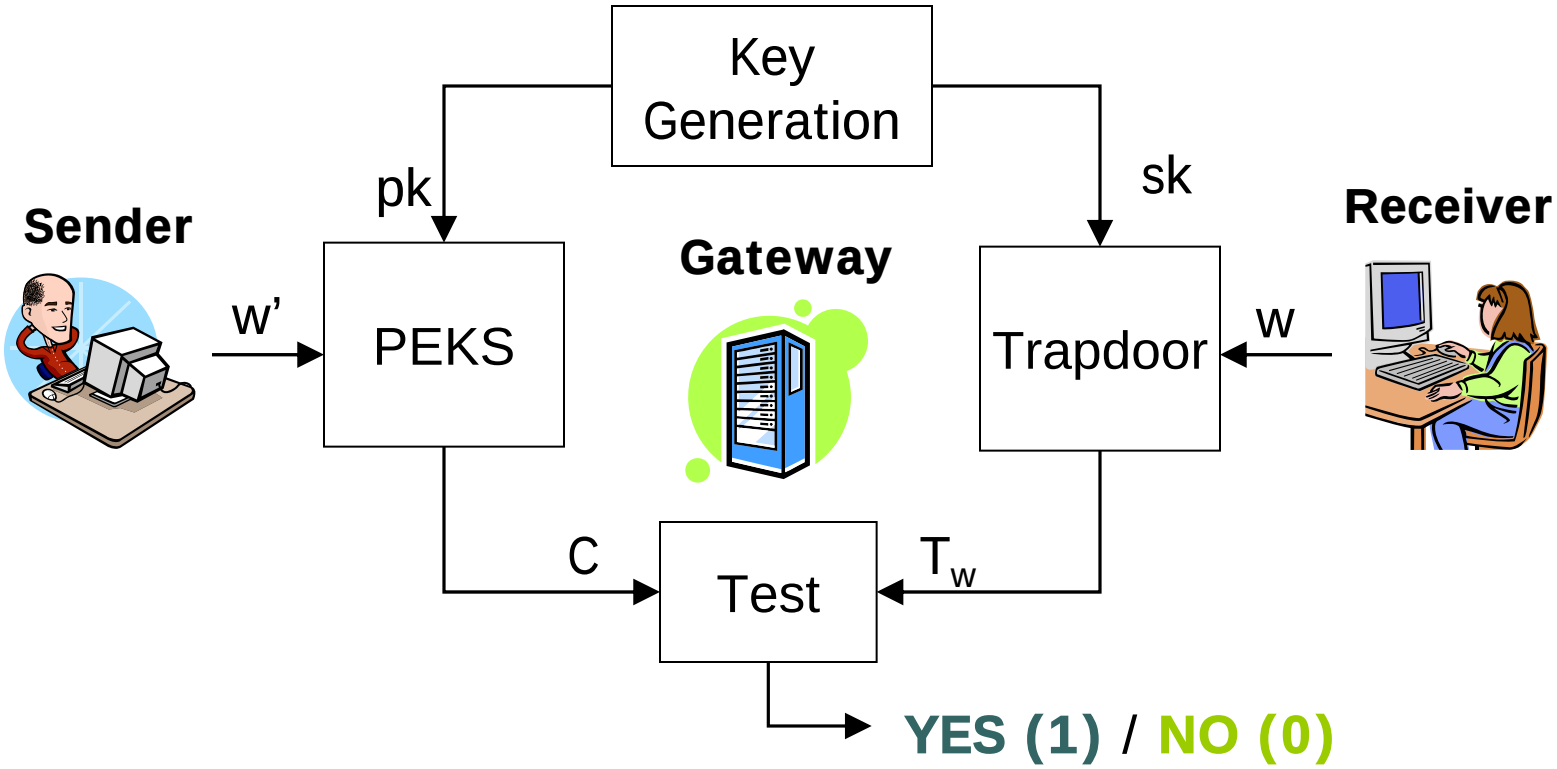
- Bob encrypt his email using a standard public-key encryption scheme PKE
- He then appends the public-key encryption with keyword search (PEKS) of each keyword

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

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

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

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



An IBE-based scheme [BDOP04]

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

Can robustness be trivially achieved?

- Is robustness implied by existing notions?
- If not, is there an easy way to make an encryption scheme robust?
- What about specific schemes such as DHIES or Cramer-Shoup?

Our results

- **Negative results**
  - We show that robustness is not implied by existing notions such as privacy or anonymity under chosen-ciphertext attacks
  - Adding redundancy to plaintext (e.g., encrypting PK and M) does not work in general
- **Positive results**
  - We provide a general transform that makes any existing PKE scheme without sacrificing its anonymity
  - We also show that some existing schemes (e.g., Cramer-Shoup) can be proven robust

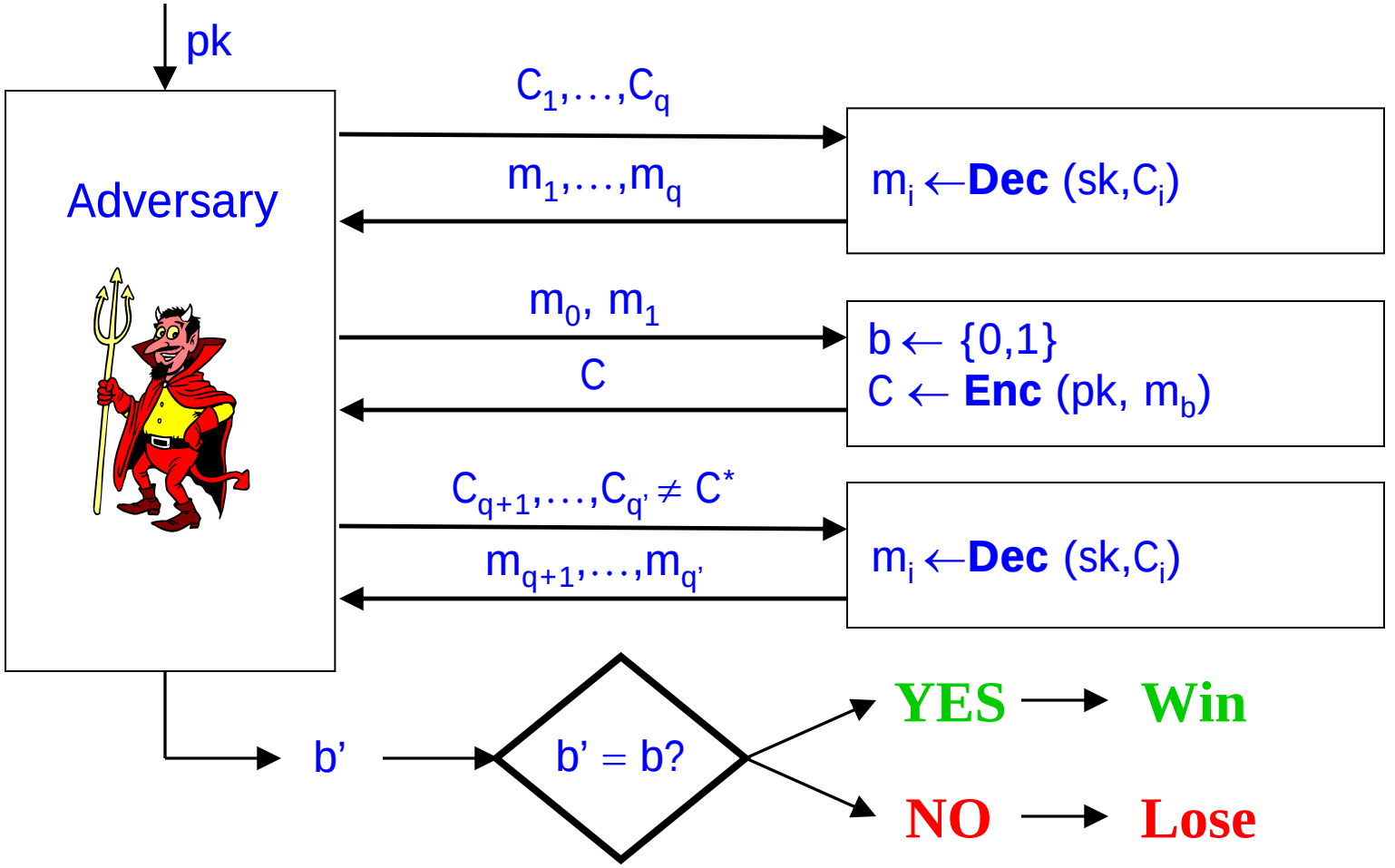
Plan

- **Security notions for PKE**
- Redundancy-based transform
- A commitment-based transform
- Robustness of specific schemes
- Extensions to the ID-based setting
- Concluding remarks

IND-CCA: privacy against chosen-ciphertext attack

- A scheme is IND-CCA secure when, given a ciphertext C whose decryption is $M=D_{sk}(C)$:
  - Adversary does not get partial information about M from C (e.g., the most significant bit of M)
  - Even when it's allowed to see the encryption $C'=E_{pk}(M')$ of messages M' of its choice
  - And the decryption $M'=D_{sk}(C')$ of ciphertexts $C' \neq C$ of its choice

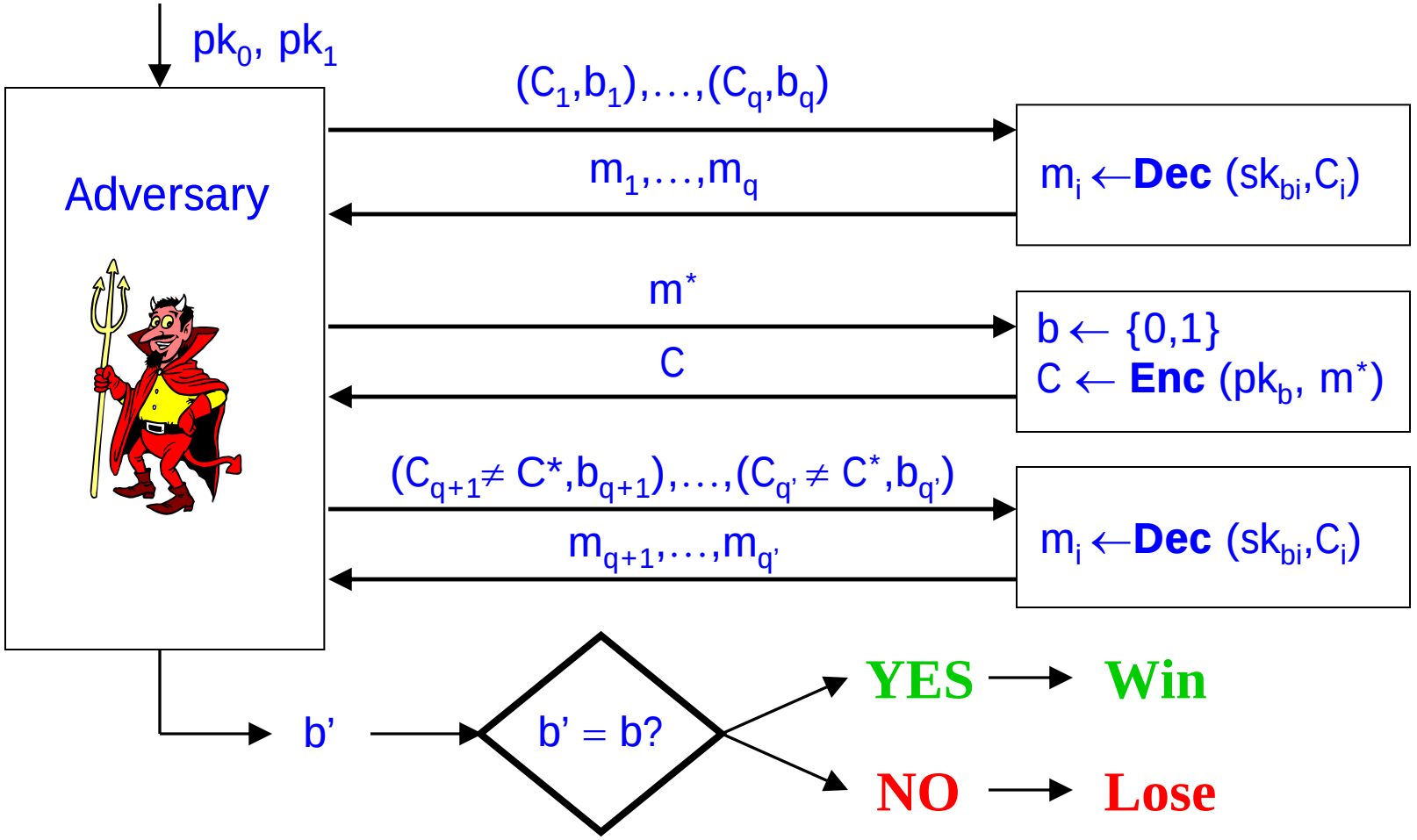
IND-CCA security experiment



ANO-CCA: anonymity against chosen-ciphertext attack

- A scheme is ANO-CCA secure when, given a message M* chosen by an adversary and two public-keys pk_0 and pk_1:
  - Adversary cannot tell apart the encryption of M* under public key pk_0 from the encryption of M* under public key pk_1
  - Even when it's allowed to see the decryptions $M_0=D_{sk0}(C')$ and $M_1=D_{sk1}(C')$ of ciphertexts $C'\neq C$ of its choice

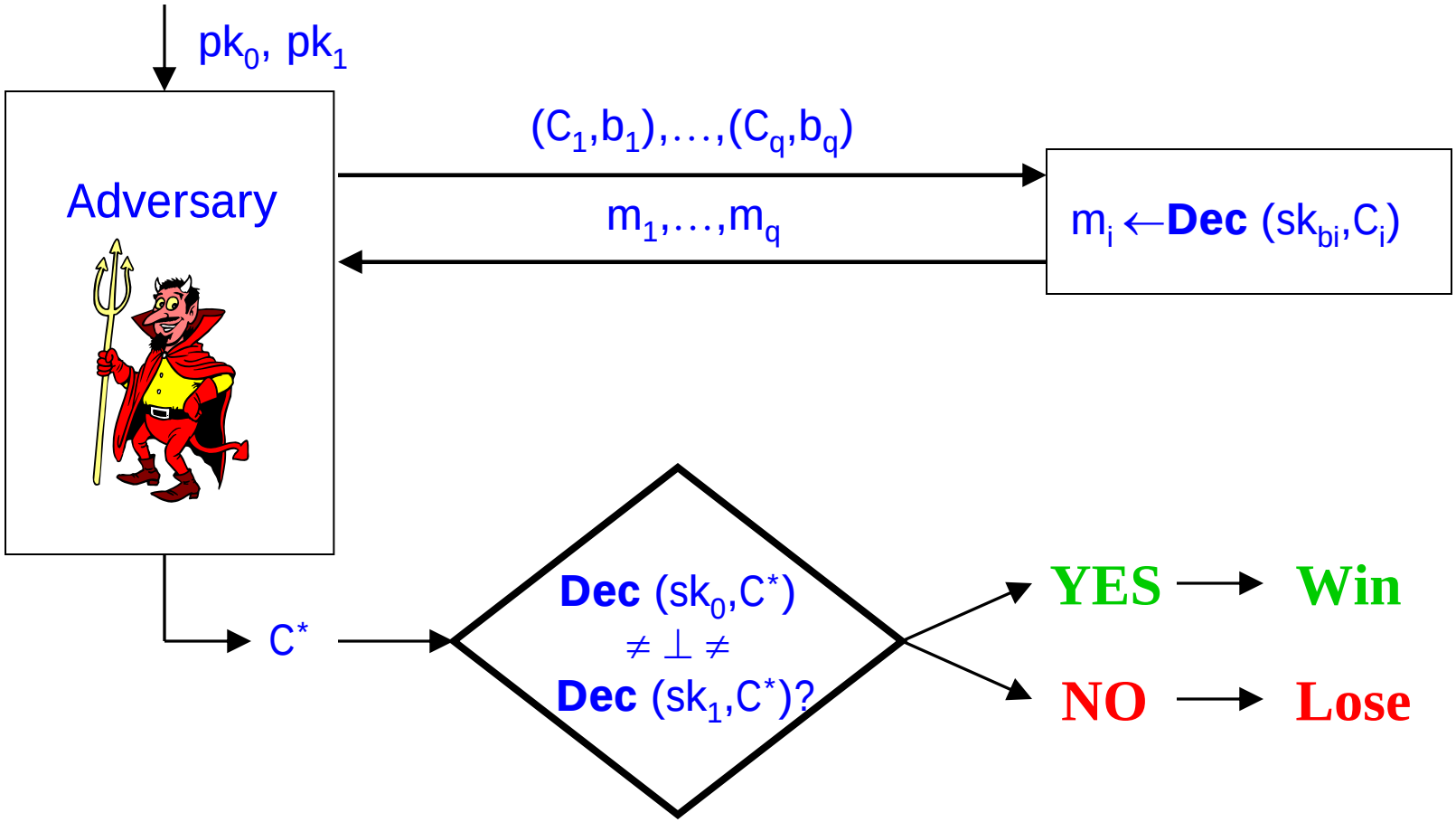
ANO-CCA security experiment



ROB-CCA: robustness against chosen-ciphertext attack

- A scheme is ROB-CCA secure when, given two public-keys pk_0 and pk_1:
  - Adversary cannot generate a ciphertext C* such that $D_{sk0}(C') \neq \perp$ and $D_{sk1}(C') \neq \perp$ simultaneously
  - Even when it's allowed to see the decryptions $M_0=D_{sk0}(C')$ and $M_1=D_{sk1}(C')$ of ciphertexts C' of its choice

ROB-CCA security experiment



Relation with existing notions

- **Theorem**: There are PKE schemes which are IND-CCA and ANO-CCA, but **not** ROB-CPA
- **Proof**:
  Given PKE = (K,E,D), build PKE' = (K,E',D') where
  - E'(PK,M)
    Return (l,C) = (|M|, E(PK,M))
  - D'(SK,(l,C))
    x = Dec(SK,C)
    If $x \neq \perp$ and $|x|=l$ return x else return 0^l

Plan

- Security notions for PKE
- **Redundancy-based transform**
- A commitment-based transform
- Robustness of specific schemes
- Extensions to the ID-based setting
- Concluding remarks

Redundancy-based transforms

- **Idea**: Add redundancy to plaintext and check upon decryption if redundancy is present
- **Intuition**: Decryption under the wrong key should look random, hence redundancy would be rarely present
- Examples of redundancy
  - Fixed string: $Enc(M||0^l)$
  - Public key: $Enc(M||PK)$
  - Hash of message and PK: $Enc(M||H(M||PK))$

Redundancy codes

- A redundancy code R=(RC,RV) is a pair of algorithms where
  - RC(x) computes a redundancy r for input x
  - RV(x,r) checks whether r is present in x
  - For all x, RV(x,RC(x))=1
- Examples
  - RC(PK,M) = 0^l
  - RC(PK,M) = PK
  - RC(PK,M) = K || H(K,pk||M) where $K \leftarrow \{0,1\}^k$

Redundancy-based transform

- Let R=(RC,RV) be a redundancy code
- Let PKE = (K,E,D) be an encryption scheme
- Transform outputs PKE'= (K,E',D') where:
  - E'(PK,M) = E(PK, M || RC(PK||M))
  - D'(SK,C')
    M || r ← D(SK,C')
    If RV(PK||M,r)=1 then return M else return ⊥

Adding redundancy fails

- **Theorem**: There exist encryption schemes PKE such that, for any redundancy code RC, the resulting encryption scheme PKE' is not ROB-CPA.

Counter example

- Let PKE*=(K*,E*,D*) be an IND-CCA and ANO-CCA encryption scheme

- Build PKE=(K*,E,D) where
  - E(PK,M) = 1 || E*(PK,M)
  - D(SK,b||C)
    If b=1, then return D*(SK,C)
    Else return M* || RC(PK||M*;$0^{l(|PK||M*|)}$)

Plan

- Security notions for PKE
- Redundancy-based transform
- **A commitment-based transform**
- Robustness of specific schemes
- Extensions to the ID-based setting
- Concluding remarks

Commitment schemes

- A commitment scheme CMT=(PG,Com,Open) is a triple of algorithms where
  - PG returns common parameters pars
  - Com(pars,x) computes a commitment com for x and the decommitment key dec
  - Open(pars,com,dec) returns either x or $\perp$
- **Correctness**
  - $\forall$x, $\forall$pars $\in$ PG, $\forall$(com,dec) $\in$ Com(pars,x): Open(pars,com,dec) = x

Commitment security properties

- **Hiding**
  - cpars $\leftarrow$ PG; b $\leftarrow$ {0,1}
  - $(x_0,x_1) \leftarrow$ Adversary(cpars)
  - (com,dec) $\leftarrow$ Com(cpars,x_b)
  - b' $\leftarrow$ Adversary(com)
  - If (b=b') then return 1 else return 0
- **Binding**
  - cpars $\leftarrow$ PG;
  - $(com,dec_0,dec_1) \leftarrow$ Adversary(cpars)
  - $x_0 \leftarrow$ Open(cpars,com,dec_0)
  - $x_1 \leftarrow$ Open(cpars,com,dec_1)
  - If ($x_0 \neq x_1$ and $x_0 \neq \perp$ and $x_1 \neq \perp$) then return 1 else return 0

A commitment-based transform

- **Idea**: Add a commitment of the public key to the ciphertext and encrypt decommitment key together with message
- **Intuition**: When decrypting with the wrong key, the probability that the decommitment key will open the commitment correctly is negligible

Our new commitment-based transform

- Let **Com = (CPG,Com,Open)** be a commitment scheme and **PKE = (PG,K,E,D)** an encryption scheme.
- We can construct a robust encryption scheme **PKE'=(PG',K',E',D')** as follows:

PG'(1^k)

pars ← **PG**(1^k)
cpars ← **CPG**(1^k)
R←{0,1}^{k}
return (pars,cpars,R)

K (pars,cpars,R)

(pk,sk) ← **K**(pars)
return (pk,sk)

E ((pars,cpars,R),pk, M)

(com,dec) ← **Com** (cpars,pk)
C ← **E** (pars,pk, M||dec||R)
return (com,C)

D ((pars,cpars),pk, sk,(com,C))

M||dec||R' ← **D**(C)
If Open(cpars,com,dec)=pk and R'=R
then return M else ⊥

Robustness of resulting PKE

- **Theorem**: If the commitment scheme Com is binding and the encryption scheme PKE is IND-CPA, then PKE' is ROB-CCA.
- **Proof**:
  - BindingAdversary(cpars)
    - pars$\leftarrow$PG(1^k); (sk_b,pk_b) $\leftarrow$K(pars) for b=0,1
    - R $\leftarrow\{0,1\}^k$
    - (com,C) $\leftarrow$ RobustAdversary(cpars,pars,R,pk_0,pk_1)
    - (m_b,dec_b,R') $\leftarrow$D(cpars,pars,sk_b,C) for b=0,1
    - Return (com,dec_0,dec_1)

Transform is CPA-preserving

- **Theorem**
  - If the encryption scheme PKE is IND-CPA, then PKE' is IND-CPA.
  - If PKE is ANO-CPA and IND-CPA and the commitment scheme Com is hiding, then PKE' is ANO-CPA.

An additional security property

- **Copy resistance**
  - cpars $\leftarrow$ PG;
  - x $\leftarrow$ Adversary(cpars)
  - (com,dec) $\leftarrow$ Com(cpars,x)
  - com' $\leftarrow$ adversary(com,dec)
  - If (com$\neq$com' and Open(cpars,com',dec)=x) then return 1 else return 0

Transform is CCA-preserving

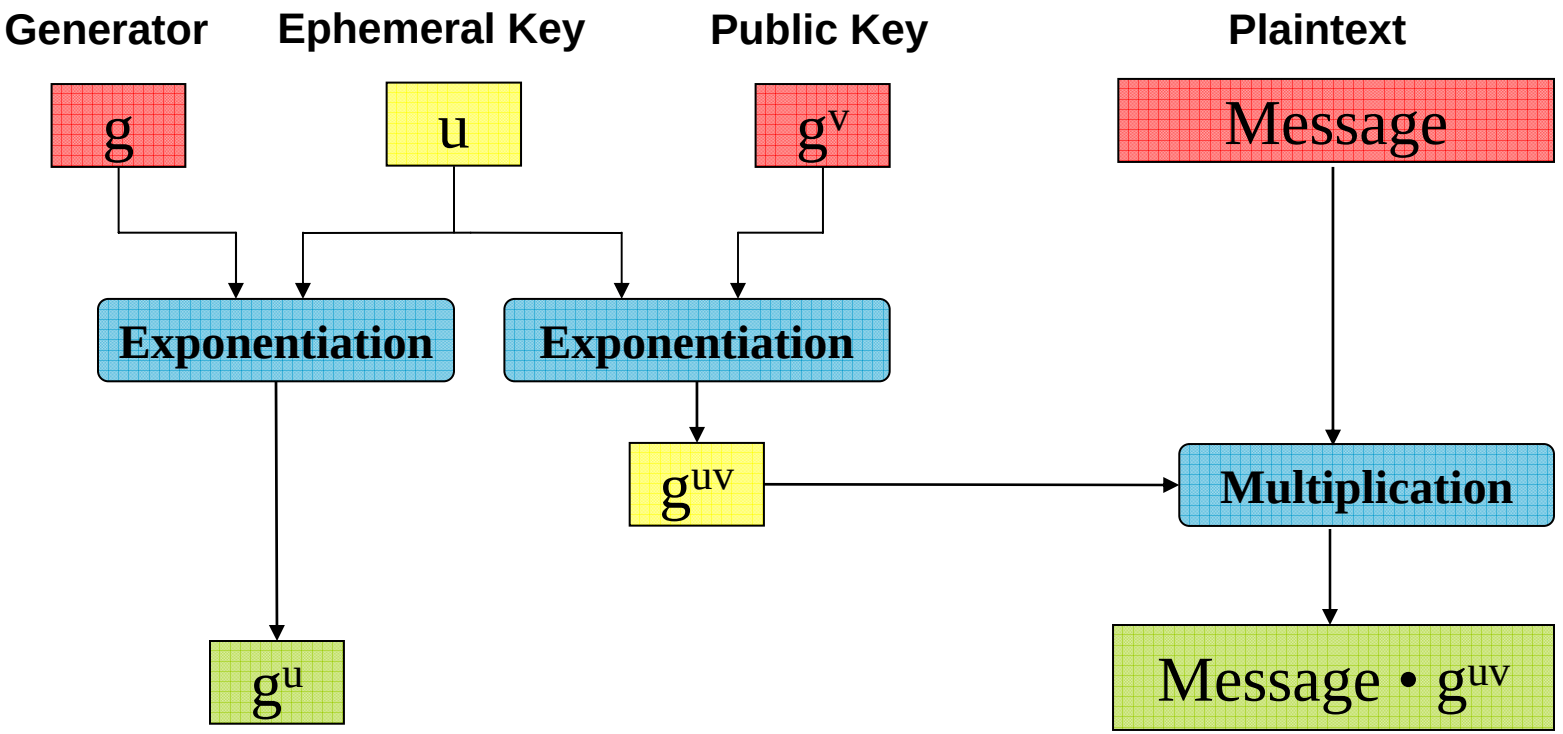
- **Theorem**
  - If the encryption scheme PKE is IND-CCA and the commitment scheme Com is copy-resistant, then PKE' is IND-CCA.
  - If PKE is ANO-CCA and IND-CCA and Com is hiding and copy-resistant, then PKE' is ANO-CCA.

Plan

- Security notions for PKE
- Redundancy-based transform
- A commitment-based transform
- **Robustness of specific schemes**
- Extensions to the ID-based setting
- Concluding remarks

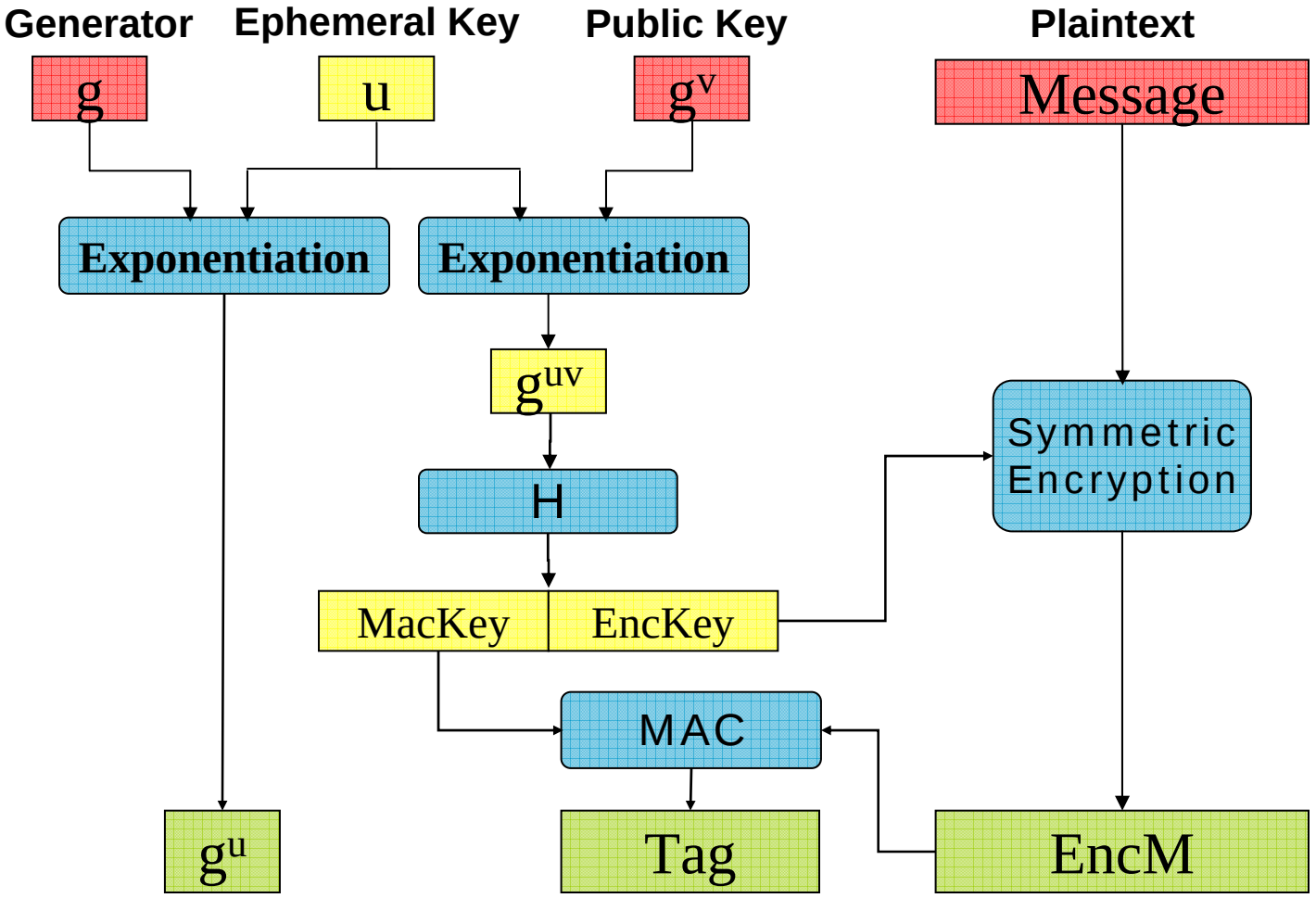
ElGamal encryption scheme

Secret Key: **v** **Public Key**: **g, g^v**



The DHIES Scheme

Secret Key: v **Public Key: g, g^v**



Cramer-Shoup encryption

PG(1^k)
$K \leftarrow Keys(H)$; $w \leftarrow Z_p^*$
$g_1 \leftarrow G^*$; $g_2 \leftarrow g_1^w$
$pars \leftarrow (g_1, g_2, K)$

KG (pars)
$x_1, x_2, y_1, y_2, z_1, z_2 \leftarrow Z_p$
$e \leftarrow g_1^{x1} g_2^{x2}$; $f \leftarrow g_1^{y1} g_2^{y2}$
$h \leftarrow g_1^{z1} g_2^{z2}$
Return $(pk=(e,f,h), sk=(x_1,x_2,y_1,y_2,z_1,z_2))$

ENC $((g_1,g_2,K), (e,f,h), M)$
$u \leftarrow Z_p^*$
$a_1 \leftarrow g_1^u$; $a_2 \leftarrow g_2^u$
$b \leftarrow h^u$
$c \leftarrow b \circ M$
$v \leftarrow H(K,(a_1,a_2,c))$
$d \leftarrow e^u f^{uv}$
$C \leftarrow (a_1, a_2, c, d)$

Dec $((g_1,g_2,K),(e,f,h),(x_1,x_2,y_1,y_2,z_1,z_2),C)$
$(a_1, a_2, c, d) \leftarrow C$
$v \leftarrow H(K,(a_1,a_2,c))$
$M \leftarrow c\, a_1^{-z1} a_2^{-z2}$
If $d \neq a_1^{x1+y1v}\, a_2^{x2+y2v}$ then $M \leftarrow \perp$
If $a_1 = 1$ then $M \leftarrow \perp$
Return M

Robustness of Cramer-Shoup

- **Theorem**: If the hash function family is pre-image resistant, then the Cramer-Shoup encryption scheme is ROB-CCA
- **Proof idea**:
  - First show that it is safe to reject any ciphertext (a_1,a_2,c,d) such that $a_2 \neq a_1^w$
  - If ciphertext is valid under pk_0 and pk_1, then $v=H(K,(a_1,a_2,c))$ must satisfy

    $v(y_{01}+wy_{02}-y_{11}-wy_{12}) + (x_{01}+wx_{02}-x_{11}-wx_{12}) = 0$

Plan

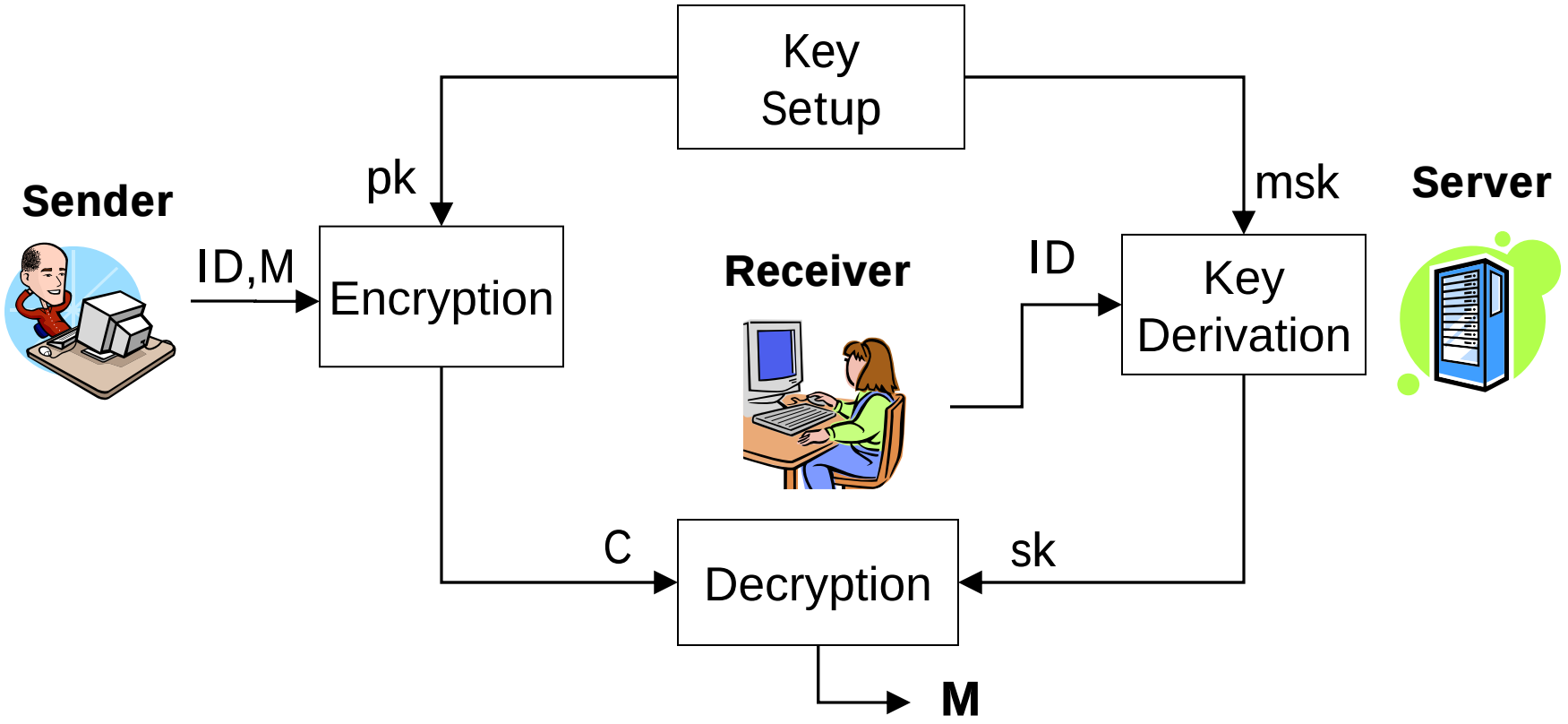
- Security notions for PKE
- Redundancy-based transform
- A commitment-based transform
- Robustness of specific schemes
- **Extensions to the ID-based setting**
- Concluding remarks

Extensions to the ID-based setting: Summary

- Definitions can be easily extended
  - Identities play the role of public keys
- The commitment-based transform also works in the IBE case, though some proofs are a bit trickier
- **IBE-2-PEKS transform**
  - PEKS consistency follows easily from robustness of the IBE
  - PEKS is IND-CCA if IBE is ANO-CCA

Identity-based encryption (IBE) [Shamir,BF01]

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

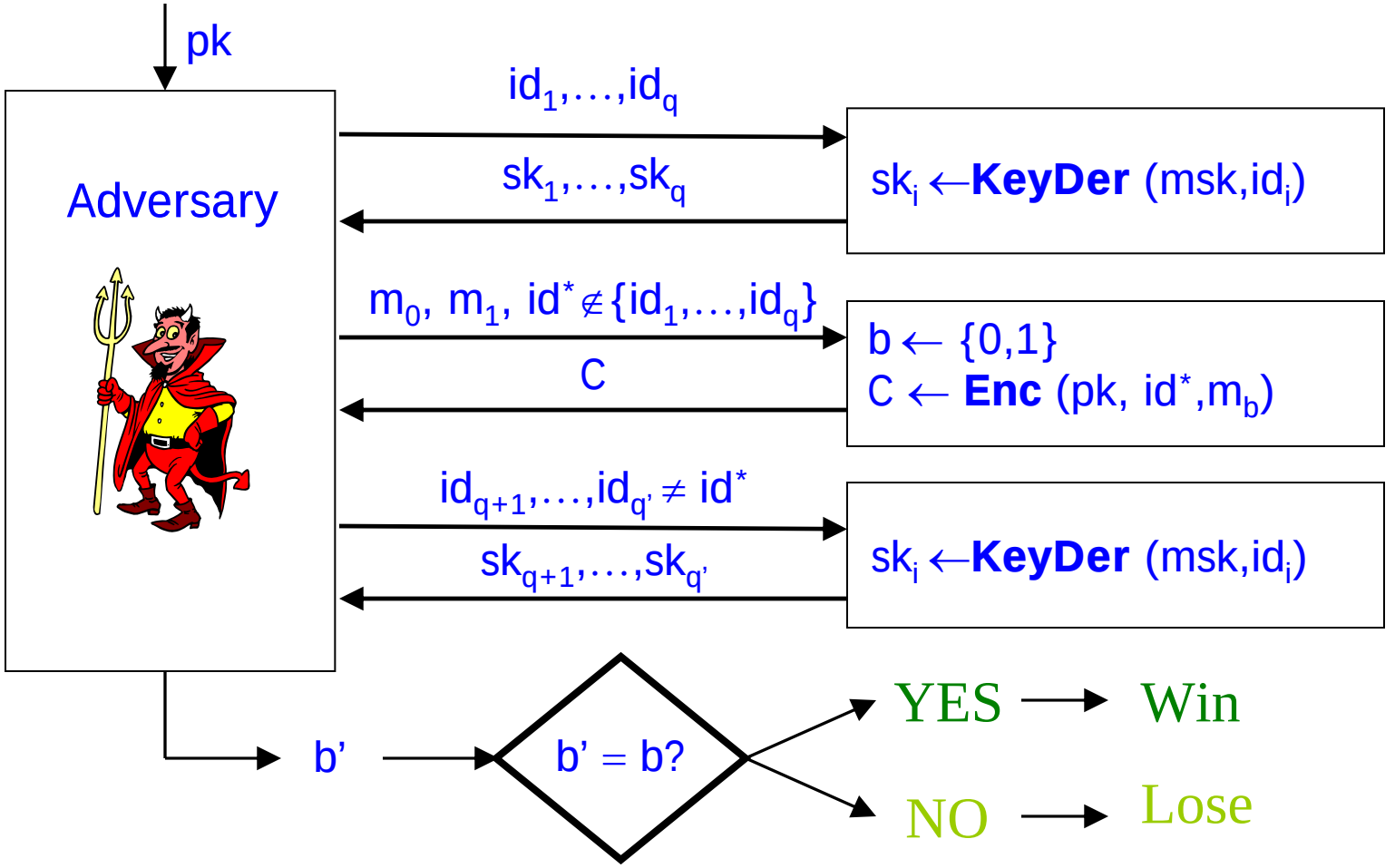


IBE-IND-CPA: privacy against chosen-plaintext attack [BF01]

An IBE 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]

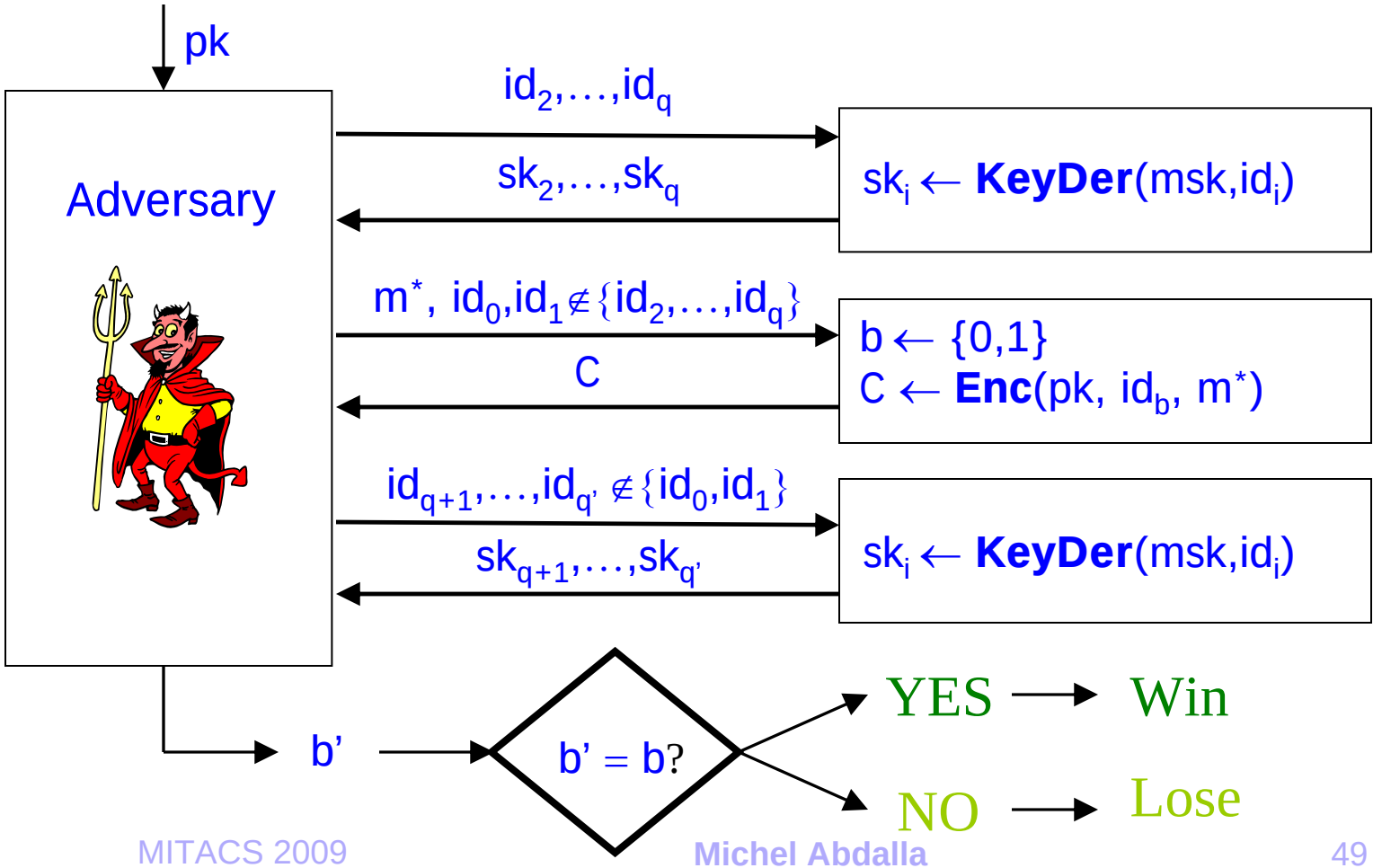


Anonymous IBE (ANO-CPA)

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



The commit-identity transform

- Let **Com = (CPG,Com,Open)** be a commitment scheme and let **IBE = (S,K,E,D)** be an IBE scheme.
- We can construct a robust IBE scheme **IBE'=(S',K',E',D')** as follows:

S'(1^k)

(pk,msk) ← **S**(1^k)
cpars ← **CPG**(1^k)
R ← $\{0,1\}^k$
pk' ← (pk,cpars,R)
return (pk',msk)

K ((pk,cpars,R),msk,ID)

sk ← **K**(pk,msk,ID)
return (sk)

E ((pk,cpars,R),ID, M)

(com,dec) ← **Com** (cpars,ID)
C ← **E** (pk,ID, M||dec||R)
return (com,C)

D ((pk,cpars,R),ID, sk,(com,C))

M||dec||R' ← **D**(mpk,ID,sk,C)
If Open(cpars,com,dec)=ID and R'=R
then return M else ⊥

Robustness of resulting IBE

- **Theorem**: If the commitment scheme Com is binding and the IBE scheme IBE is IND-CPA, then IBE' is ROB-CCA.
- **Proof**: Similar to PKE case.

Transform is CPA-preserving

- **Theorem**
  - If the IBE scheme IBE is IND-CPA, then IBE' is IND-CPA.
  - If IBE is ANO-CPA and IND-CPA and the commitment scheme Com is hiding, then IBE' is ANO-CPA.

Transform is CCA-preserving

- **Theorem**
  - If the IBE scheme IBE is IND-CCA and the commitment scheme Com is copy-resistant, then IBE' is IND-CCA.
  - If IBE is ANO-CCA and IND-CCA and Com is hiding and copy-resistant, then IBE' is ANO-CCA.

Concluding remarks

- Robustness is extremely important for the correctness of several applications
  - E.g., anonymous broadcast, auctions, PEKS
- Robustness has been considered informally in the cryptographic community for a while
  - This work makes it explicit and provides formal definitions for it
- Contrary to what seems intuitive, natural ways to confer robustness (such as adding redundancy) fail