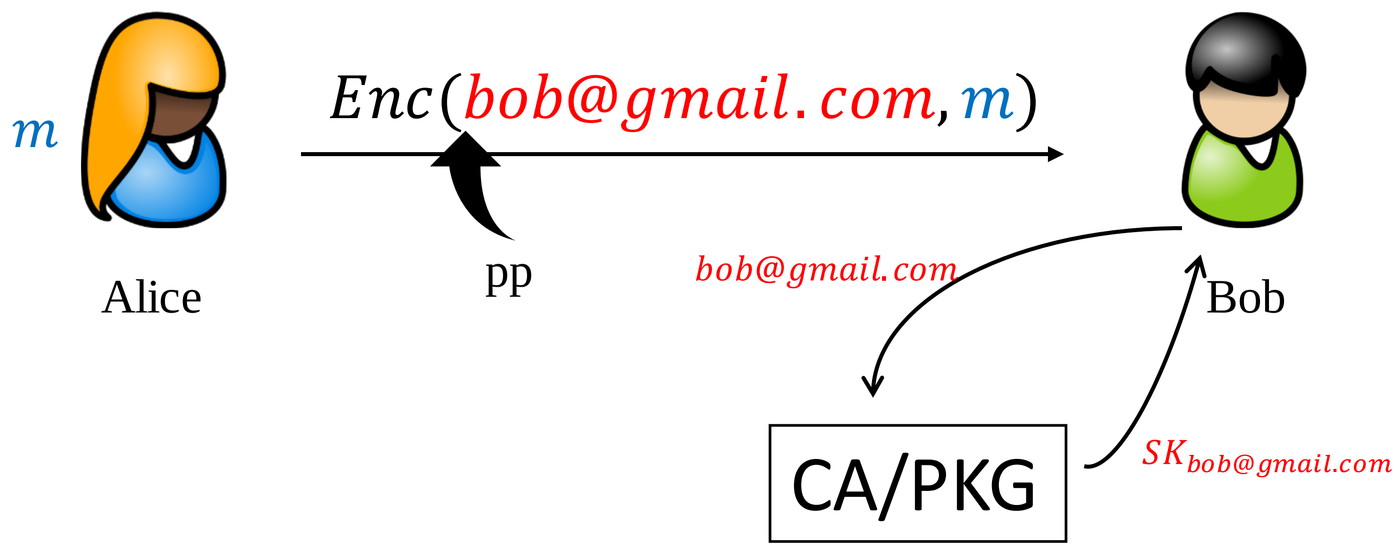
CS171: Cryptography

Lecture 18

Sanjam Garg

Identity-Based Encryption (IBE) [Shamir84]

Identity of the recipient used as the public key



Identity-Based Encryption (IBE) [Shamir84]

Four Algorithms: (S, K, E, D)

$S(1^\lambda) \rightarrow (pp, msk)$ — pp are public parameters; msk is the master secret-key

$K(msk, ID) \rightarrow sk_{ID}$ — sk_{ID} secret key for ID

$E(pp, ID, m) \rightarrow c$ — encrypt using pp and ID

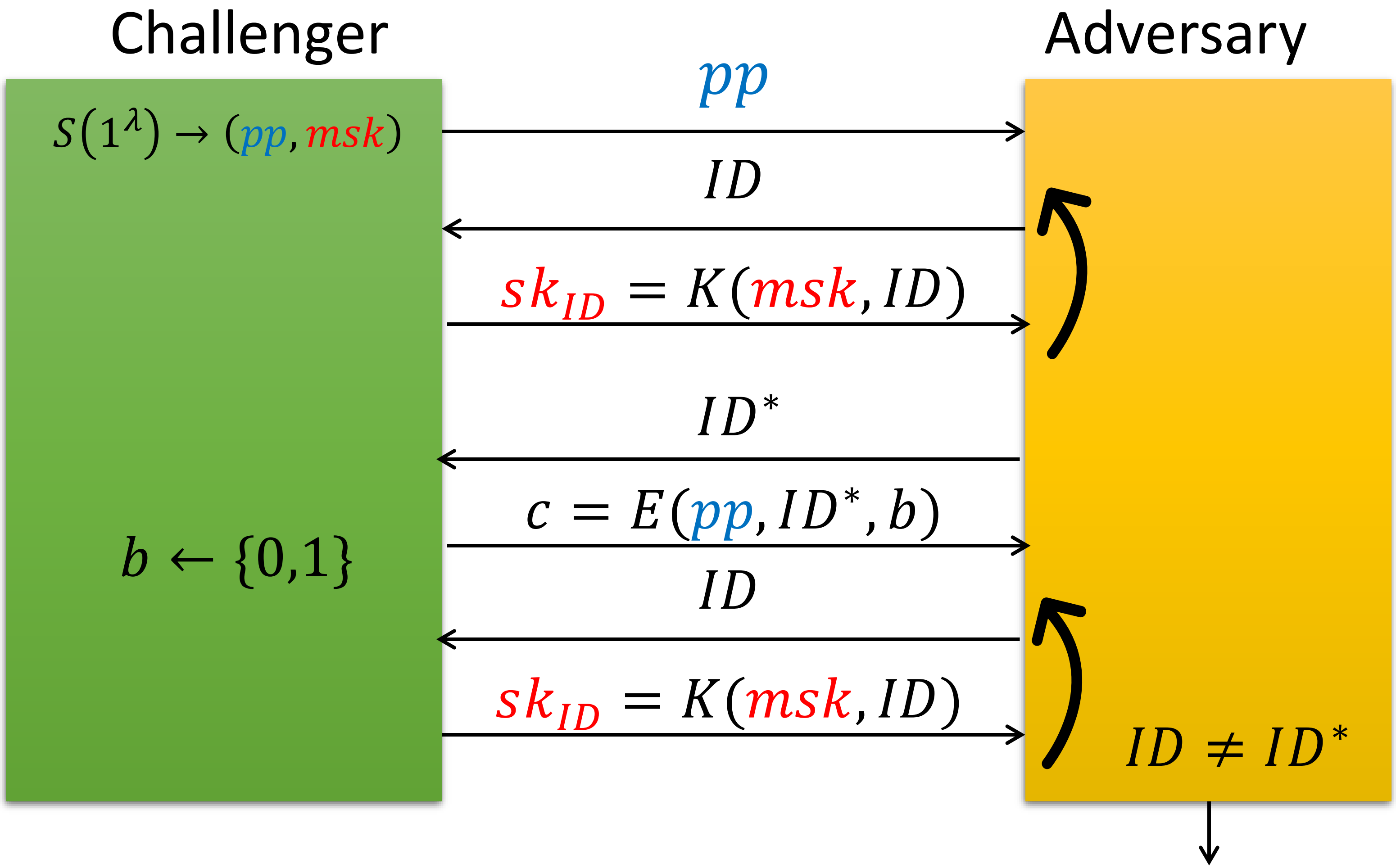
$D(sk_{ID}, c) \rightarrow m$ — decrypt c using sk_{ID}

Security of IBE [BF01]

- Attacker has access to *any* number of keys for identities of his choice

- Attacker cannot break security for any *other* identity

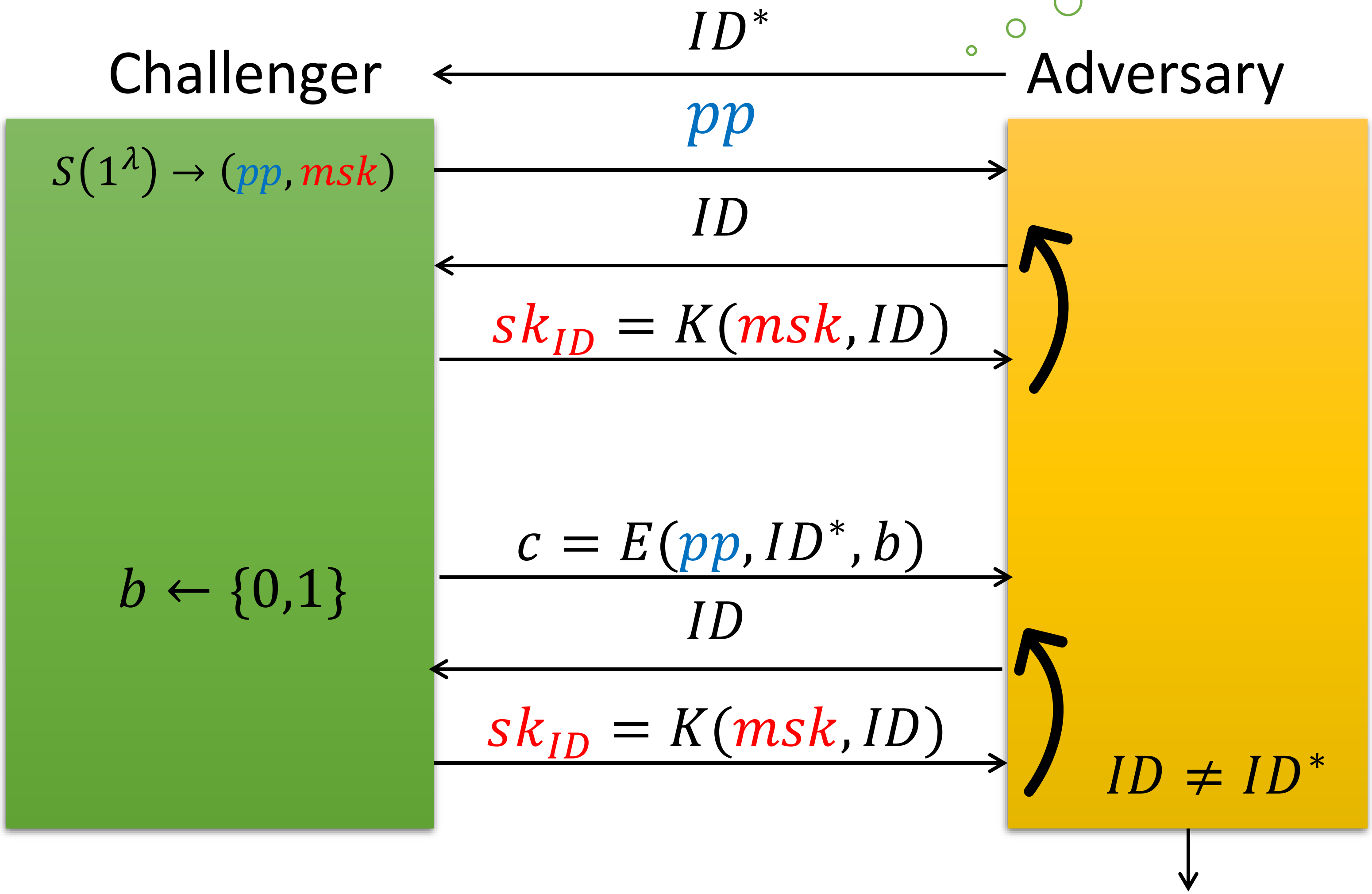
Security of IBE [BF01]



$b' \in \{0,1\}$

$$|\Pr[b = b'] - 1/2| \approx neg(\lambda)$$

Security of IBE [BF01]



$$|\Pr[b = b'] - 1/2| \approx neg(\lambda)$$

$$b' \in \{0,1\}$$

Bilinear Groups

- High level: Groups where CDH is hard but DDH is easy
- Consider group G of prime order q and generator g
- Comes with a - Bilinear map e
  - e: $G \times G \rightarrow G_T$
  - If g is a generator of G then $e(g, g)$ is a generator of G_T
  - $\forall\, a, b \in Z_q^*, e\left(g^a, g^b\right) = e(g, g)^{ab}$
- DDH is easy: how?
  - A, B, C is a DDH tuple if and only if $e(A, B) = e(g, C)$
- CDH is hard: how?
  - Cannot prove! Assume as no attacks are known.

Decisional Bilinear Diffie-Hellman Assumption

$\mathrm{DBDH}_{A,\mathcal{G}}(n)$

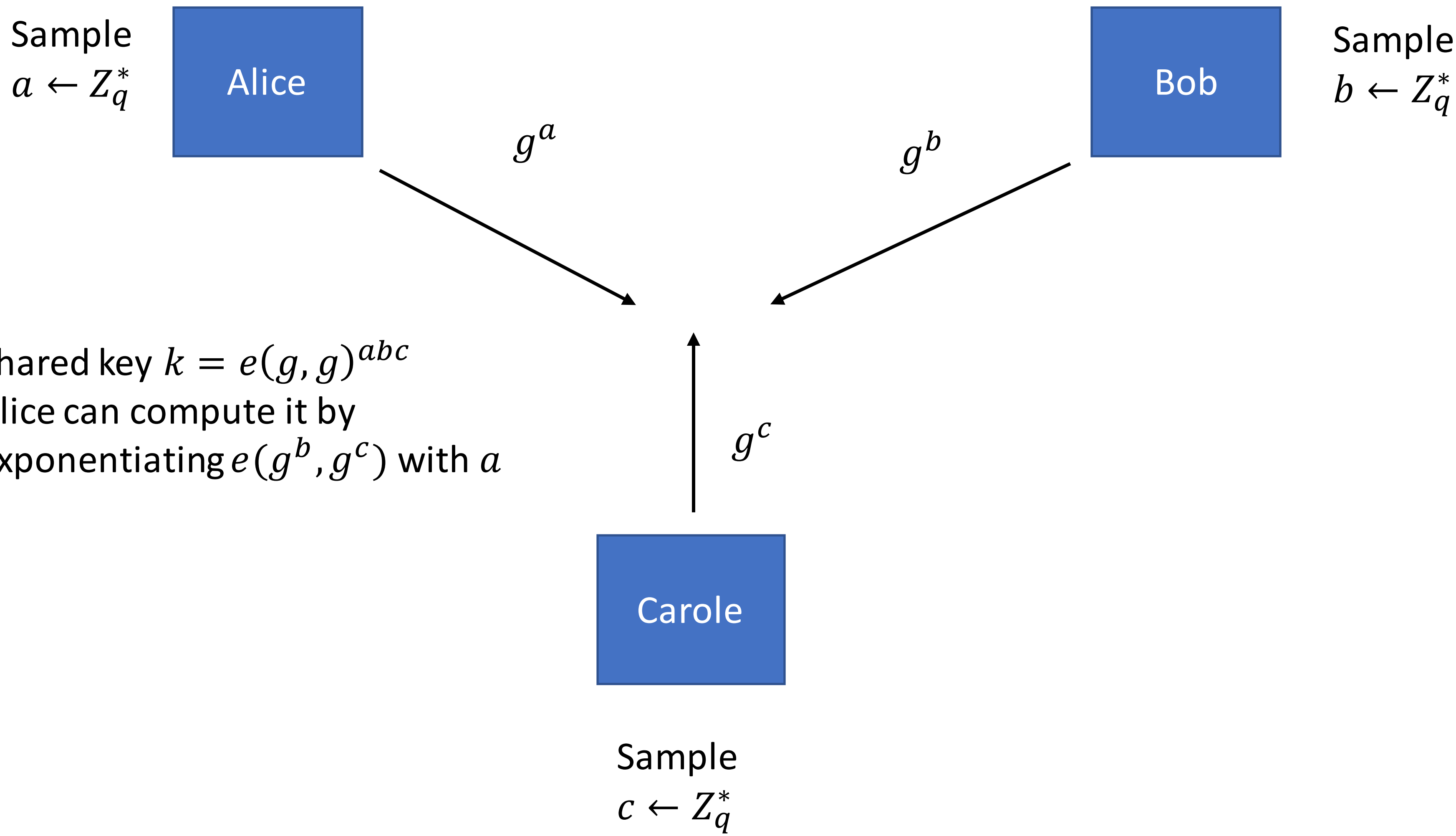
1. Run $\mathcal{G}(1^n)$ to obtain (G, G_T, g, q, e).
2. $a, b, c, r \leftarrow Z_q^*$ and $\beta \leftarrow \{0,1\}$.
3. A is given (G, G_T, g, q) and $(g^a, g^b, g^c, e(g,g)^{abc+\beta r})$ outputs β'.
4. Output 1 if $\beta = \beta'$ and 0 otherwise

DBDH is hard relative to $\mathcal{G}$ if

$\forall$ $PPT\ A$ $\exists$ $negl$ such that:

$\left|\Pr\left[\mathrm{DBDH}_{A,\mathcal{G}}(n) = 1\right] - 1/2\right| \leq \mathrm{negl}(n).$

Three party Non-Interactive Key-Exchange



Shared key $k = e(g,g)^{abc}$
Alice can compute it by exponentiating $e(g^b, g^c)$ with a

IBE Construction

Let $H : \{0,1\}^* \to G$ be a hash function

- $S(1^n)$: Output $mpk = g^a$ and $msk = a$
- $K(msk, id)$: Output $sk_{id} = H(id)^a$
- $E(mpk, id, m \in G)$: Sample $r \leftarrow Z_q^*$ and output $c_0 = g^r, c_1 = m \cdot e\big(mpk, H(id)\big)^r$
- $D\big(sk_{id}, (c_0, c_1)\big)$: Output $\frac{c_1}{e(c_0, sk_{id})}$

- Correctness: Follows by a simple check
- Security: Given g^a, g^r and $H(id)$, $e\big(g, H(id)\big)^{ar}$ is indistinguishable from uniform.

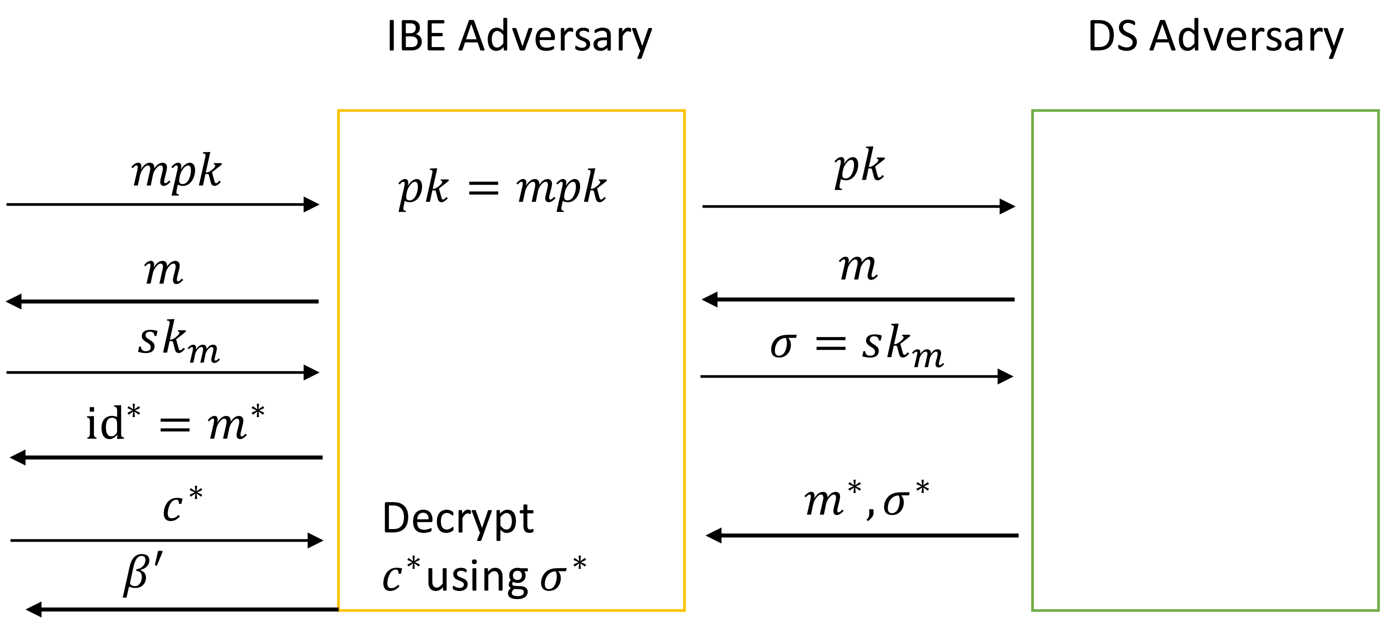
Digital Signatures from IBE

IBE => Digital Signatures

- $Gen(1^n)$: Sample $(mpk, msk) \leftarrow S(1^n)$ and output $pk = mpk, sk = msk$
- $Sign(sk, m)$: Output $\sigma \leftarrow K(msk, m)$
- $Vrfy(pk, m, \sigma)$: Output 1 if and only if for a random $h \leftarrow G$, we have that $\mathrm{D}\big(\sigma, E(mpk, m, h)\big) = h$

Proof

- Attackers ability to produce a forgery on a message m^* directly translates to breaking the security of the IBE on identity $id^* = m^*$.



CCA Security from IBE

CCA Security

Much harder in the PKE setting.

$\mathrm{PubK}^{\mathrm{CCA}}_{\mathrm{A},\Pi}(n)$

1. $(pk, sk) \leftarrow G(1^n)$ and give pk to A.
2. $A^{Dec(sk,\cdot)}$ outputs $m_0, m_1 \in \{0,1\}^*, |m_0| = |m_1|$.
3. b $\leftarrow$ {0,1}, $c \leftarrow Enc(pk, m_b)$
4. c is given to $A^{Dec(sk,\cdot)}$ and it outputs b' (query c not allowed)
5. Output 1 if $b = b'$ and 0 otherwise

Encryption scheme $\Pi = (Gen, Enc, Dec)$ is indistinguishable in the presence of a CCA attacker, or is *CCA-secure* if

$\forall$ PPT A it holds that:

$$\Pr\left[\mathrm{PubK}^{\mathrm{cca}}_{\mathrm{A},\Pi} = 1\right] \leq \frac{1}{2}$$

+ negl(n)

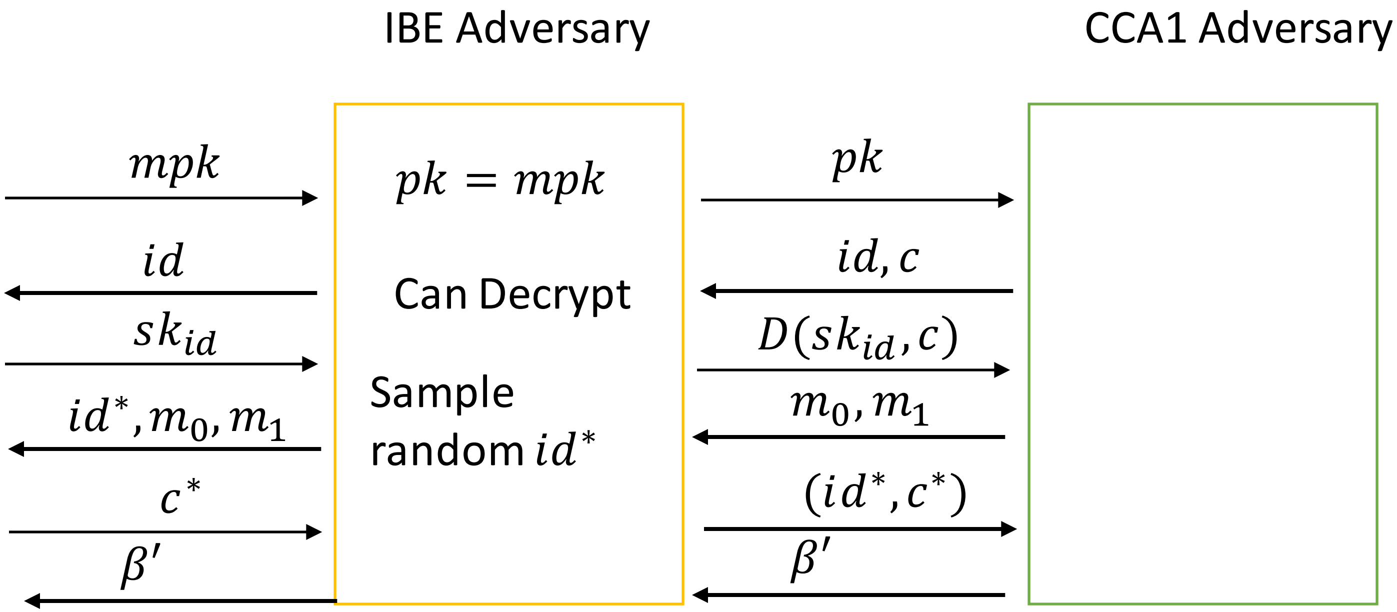
Let’s start with CCA1

CCA Security from IBE

IBE => CCA1 Secure PKE

- $Gen(1^n)$: Sample $(mpk, msk) \leftarrow S(1^n)$ and output $pk = mpk, sk = msk$
- $Enc(pk, m)$: Sample a random identity id. Output ciphertext as $(id, \mathrm{E(pk, id, m)})$
- $Dec(sk, (id, c))$: Output $D(K(sk, id), c)$

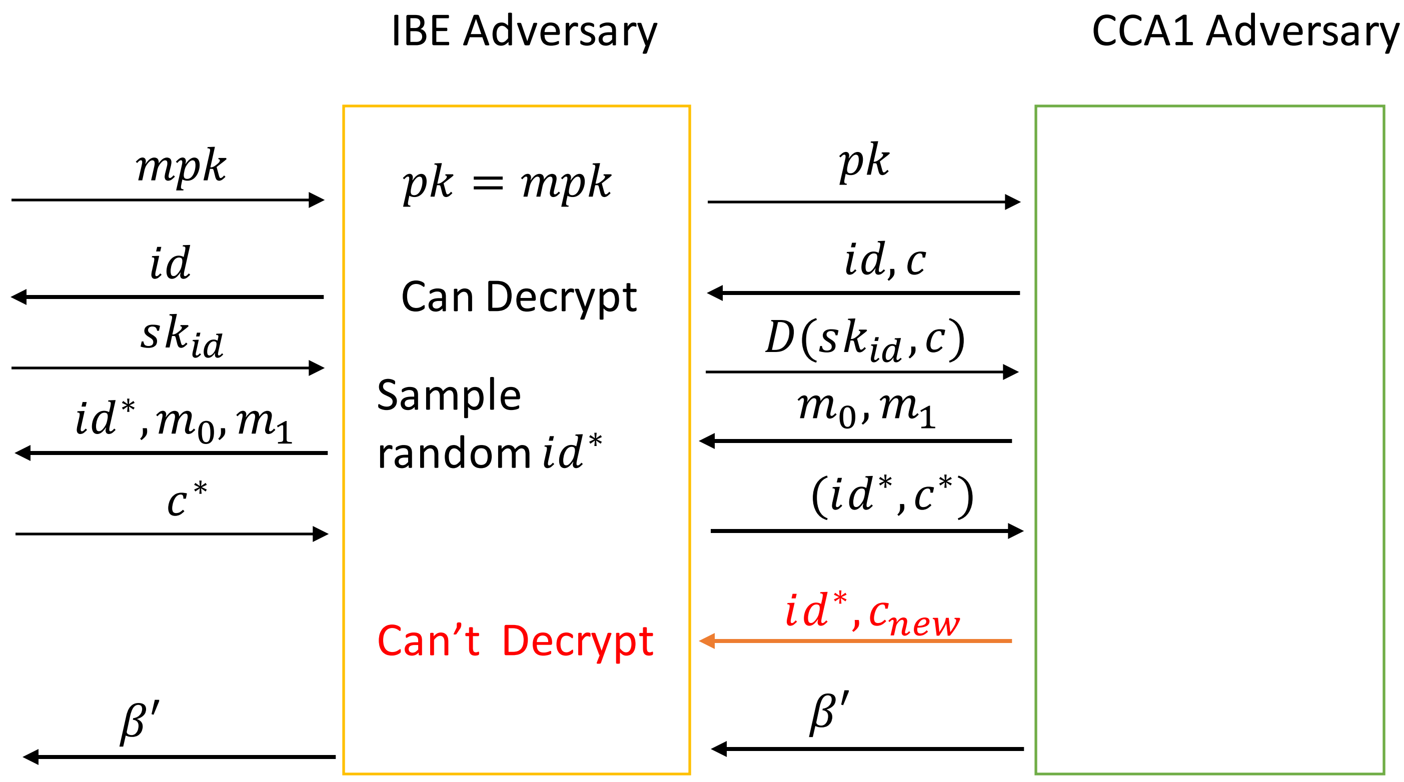
Proof



IBE Adversary
CCA1 Adversary
mpk
id
sk_{id}
id^*, m_0, m_1
c^*
β'
$pk = mpk$
Can Decrypt
Sample random id^*
pk
id, c
$D(sk_{id}, c)$
m_0, m_1
(id^*, c^*)
β'

What is the problem in getting CCA2?

- The adversary can generate ciphertexts for identity id^* that the IBE adversary (or reduction) will not be able to answer



How can we fix this?

- Two possibilities:
  - Develop a method to enable decryption of such new ciphers.
  - Prevent CCA2 attacker for asking such decryption queries.

- How can we prevent the attacker for asking such queries?

Replace identity
with verification key
of a signature
scheme

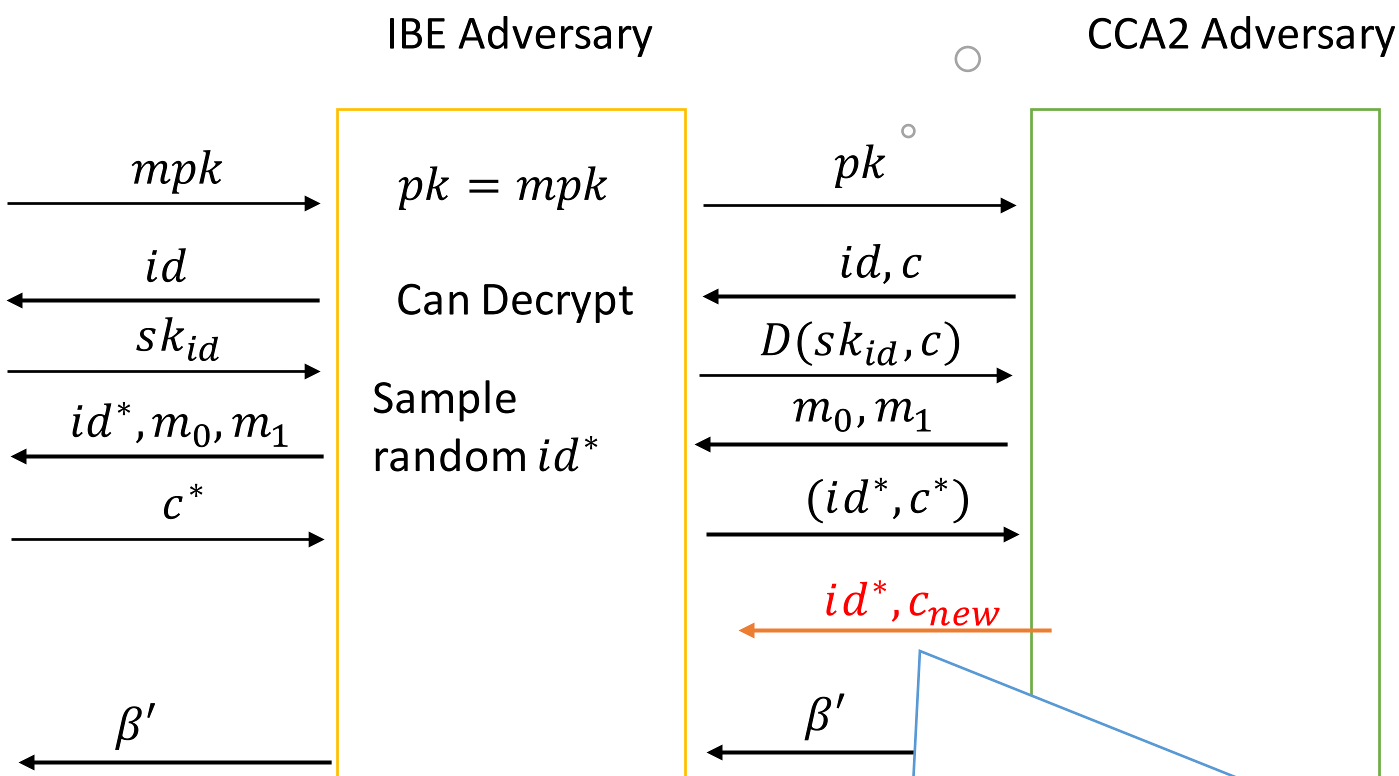
IBE => CCA2 Secure PKE

Strongly Unforgeable Digital Signature scheme

- $Gen(1^n)$: Sample $(mpk, msk) \leftarrow S(1^n)$ and output $pk = mpk, sk = msk$
- $Enc(pk, m)$: Sample $(\mathrm{pk_{sig}}, \mathrm{sk_{sig}}) \leftarrow Gen_{sig}(1^n)$.
  1. Set $id = pk_{sig}$
  2. $c \leftarrow E(pk, id, m)$
  3. $\sigma \leftarrow Sign(sk_{sig}, c)$
  4. Output ciphertext as (id, c, σ)
- $Dec(sk, (id, c))$: Output $D(K(sk, id), c)$ if $Ver(pk_{sig} = id, c, \sigma) = 1$ and error $\perp$ otherwise.

CCA2 secure PKE

One-time security for the signature scheme suffices.



The ciphertext $c_{new} = (id^*, c', \sigma')$ is such that $(c', \sigma') \neq (c^*, \sigma^*)$ and it needs to be decrypted only if $Vrfy(id^*, c', \sigma') = 1$. Specifically, IBE adversary can safely return $\perp$ if this test is the signature verification fails. However, if the signature verification success then (c', σ') is actually a forgery for the underlying signature scheme.

Thank You!