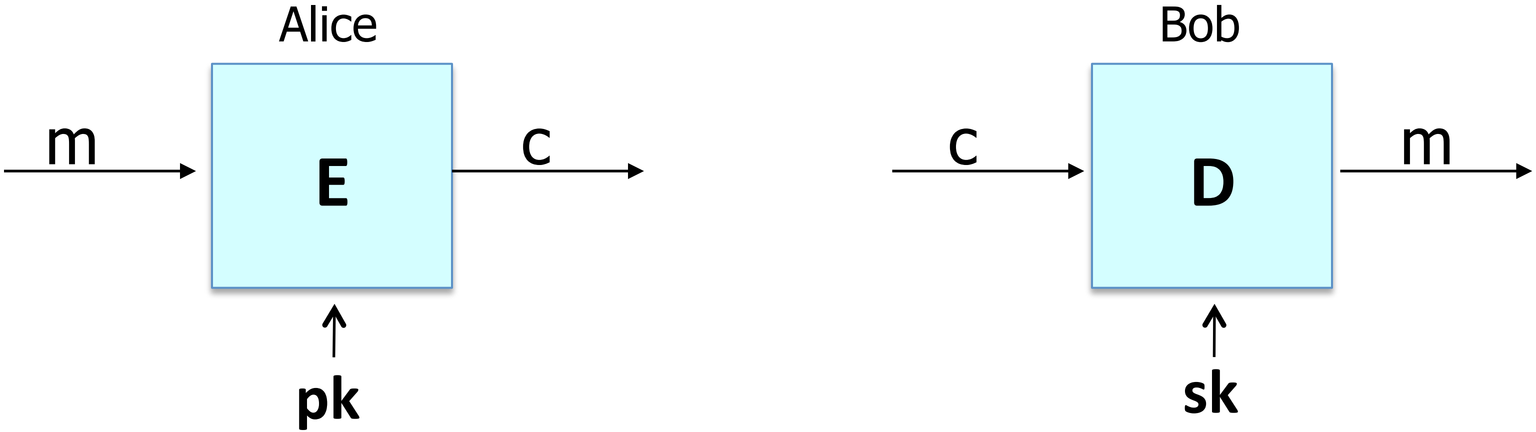
Public Key Encryption from trapdoor permutations

Public key encryption: definitions and security

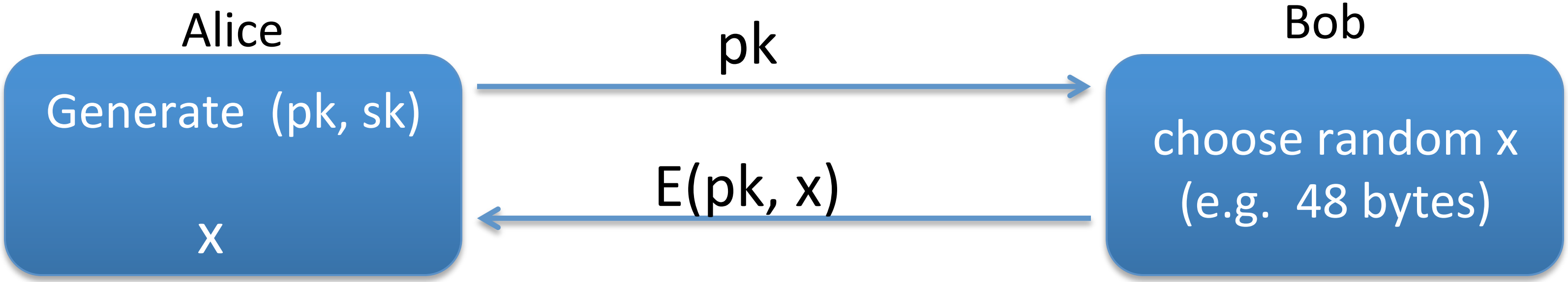
Public key encryption

Bob: generates (PK, SK) and gives PK to Alice



Applications

Session setup (for now, only eavesdropping security)



Non-interactive applications: (e.g. Email)

- Bob sends email to Alice encrypted using pk_{alice}
- Note: Bob needs pk_{alice} (public key management)

Public key encryption

<u>**Def**</u>: a public-key encryption system is a triple of algs. (G, E, D)

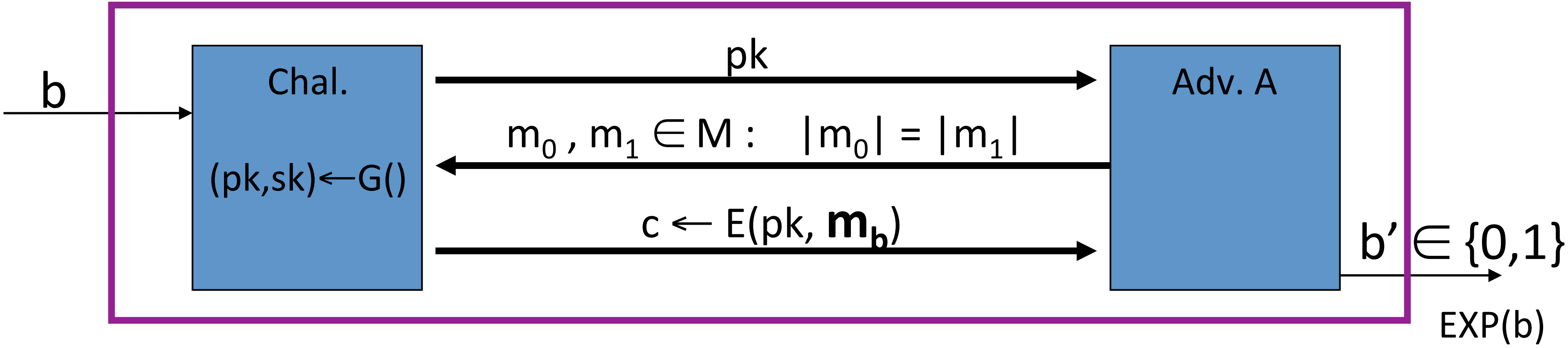
- G(): randomized alg. outputs a key pair (pk, sk)
- E(pk, m): randomized alg. that takes $m \in M$ and outputs $c \in C$
- D(sk,c): det. alg. that takes $c \in C$ and outputs $m \in M$ or $\perp$

Consistency: $\forall$(pk, sk) output by G :

$$\forall m \in M: \quad D(sk, E(pk, m)) = m$$

Security: eavesdropping

For b=0,1 define experiments EXP(0) and EXP(1) as:


b
Chal.
(pk,sk)⟵G()
pk
m_0 , m_1 ∈ M : $|m_0| = |m_1|$
c ⟵ E(pk, $\mathbf{m_b}$)
Adv. A
b' ∈ {0,1}
EXP(b)

Def: $\mathbb{E}$ =(G,E,D) is sem. secure (a.k.a IND-CPA) if for all efficient A:

$$\text{Adv}_{\text{SS}}[\text{A},\mathbb{E}] = \Big| \Pr[\text{EXP}(0)=1] - \Pr[\text{EXP}(1)=1] \Big| < \text{negligible}$$

Dan Boneh

Relation to symmetric cipher security

Recall: for symmetric ciphers we had two security notions:

- One-time security and many-time security (CPA)
- We showed that one-time security $\not\Rightarrow$ many-time security

For public key encryption:

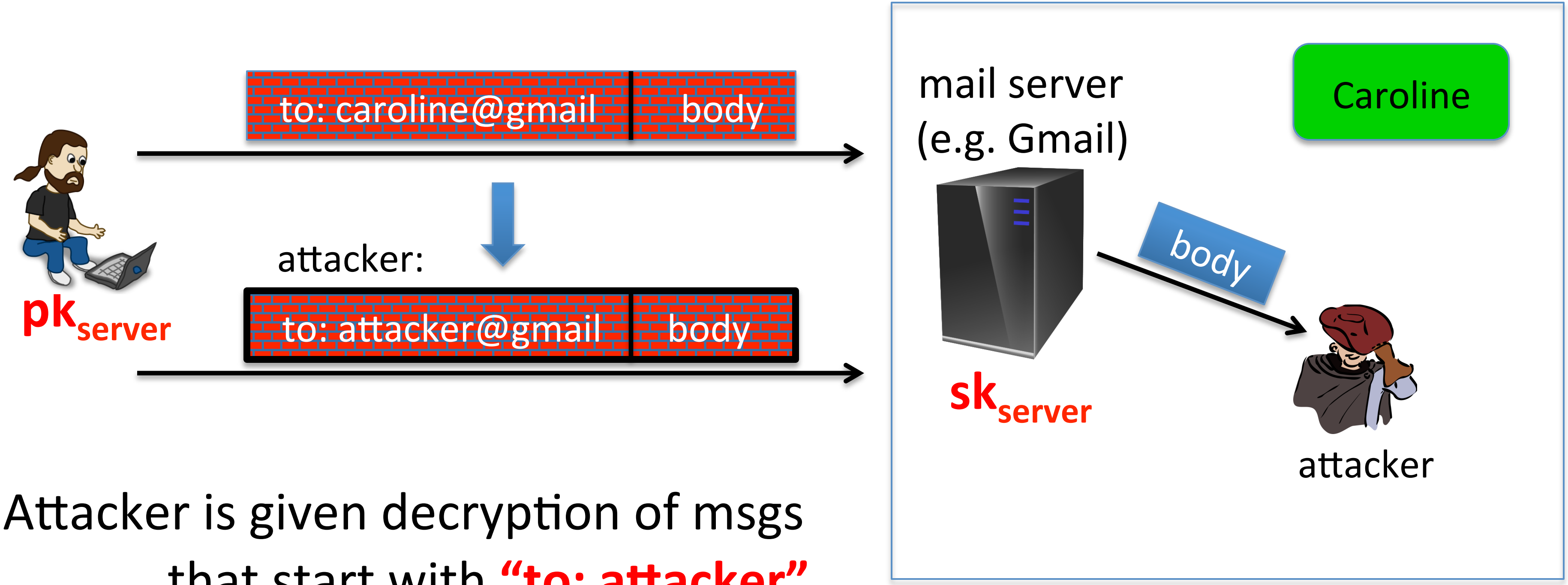
- One-time security $\Rightarrow$ many-time security (CPA)

  (follows from the fact that attacker can encrypt by himself)

- Public key encryption **must** be randomized

Security against active attacks

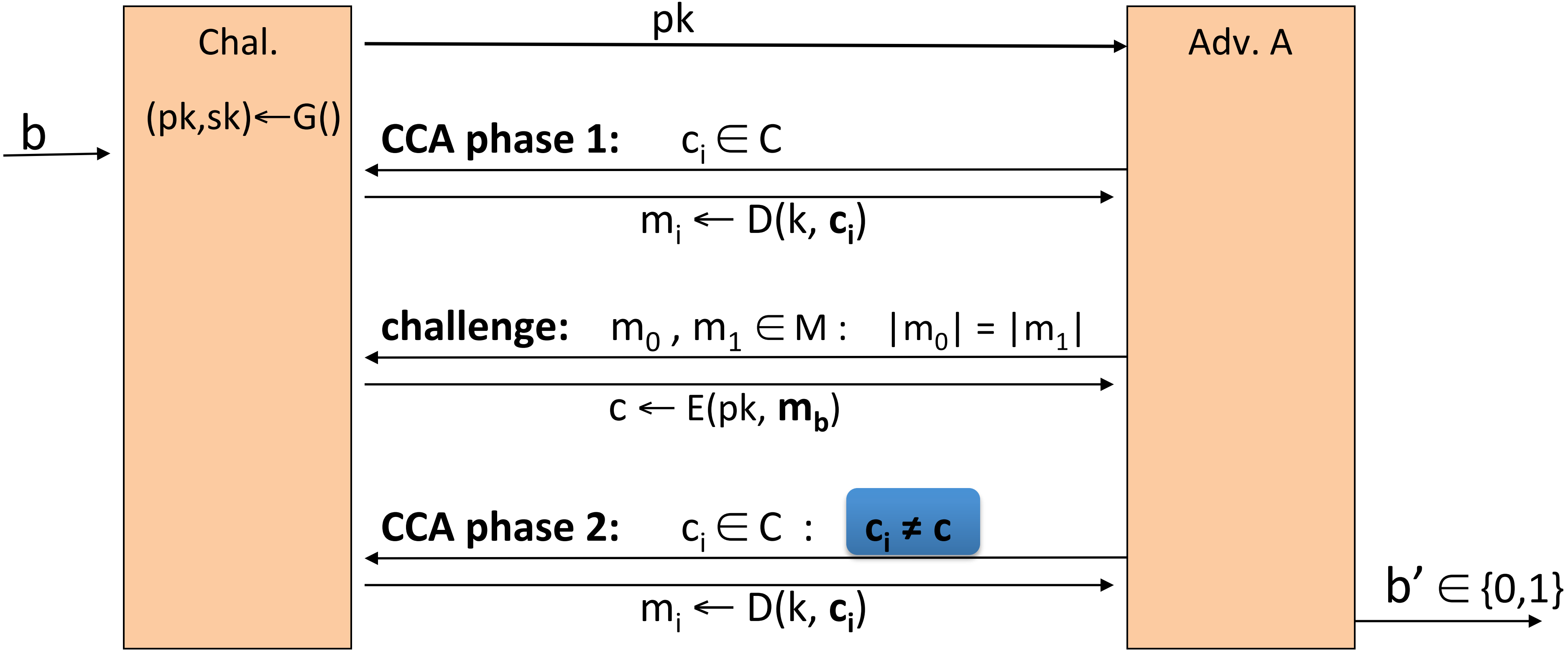
What if attacker can tamper with ciphertext?



Attacker is given decryption of msgs
that start with **“to: attacker”**

(pub-key) Chosen Ciphertext Security: definition

$\mathbb{E}$ = (G,E,D) public-key enc. over (M,C). For b=0,1 define EXP(b):



Chal.

(pk,sk)⟵G()

b

pk

Adv. A

CCA phase 1: $c_i \in C$

$m_i \leftarrow D(k, \mathbf{c_i})$

challenge: $m_0, m_1 \in M : \ |m_0| = |m_1|$

$c \leftarrow E(pk, \mathbf{m_b})$

CCA phase 2: $c_i \in C$: $\mathbf{c_i \neq c}$

$m_i \leftarrow D(k, \mathbf{c_i})$

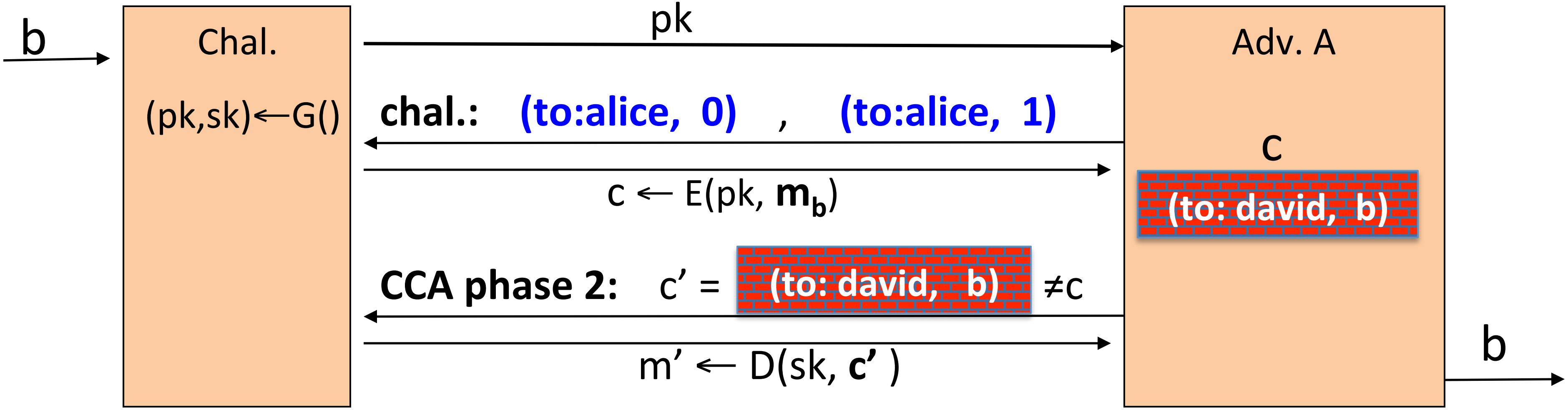
$b' \in \{0,1\}$

Chosen ciphertext security: definition

Def: E is CCA secure (a.k.a IND-CCA) if for all efficient A:

$$\mathrm{Adv}_{\mathrm{CCA}}[A,E] = \left| \Pr[\mathrm{EXP}(0)=1] - \Pr[\mathrm{EXP}(1)=1] \right|$$ is negligible.

Example: Suppose **(to: alice, body)** ⟶ **(to: david, body)**


b
Chal.
(pk,sk)⟵G()
pk
Adv. A
chal.: (to:alice, 0) , (to:alice, 1)
c ⟵ E(pk, $\mathbf{m_b}$)
c
(to: david, b)
CCA phase 2: c′ = (to: david, b) ≠c
m′ ⟵ D(sk, **c′**)
b

Active attacks: symmetric vs. pub-key

Recall: secure symmetric cipher provides **authenticated encryption**

[chosen plaintext security & ciphertext integrity **]**

- Roughly speaking: **attacker cannot create new ciphertexts**
- Implies security against chosen ciphertext attacks

In public-key settings:

- Attacker **can** create new ciphertexts using pk !!
- So instead: we directly require chosen ciphertext security

This and next module:

constructing CCA secure pub-key systems

End of Segment