CS 442
Introduction to Cryptography

Lecture 22: Example Problems on PKE

Instructor: Aarushi Goel

Spring 2026

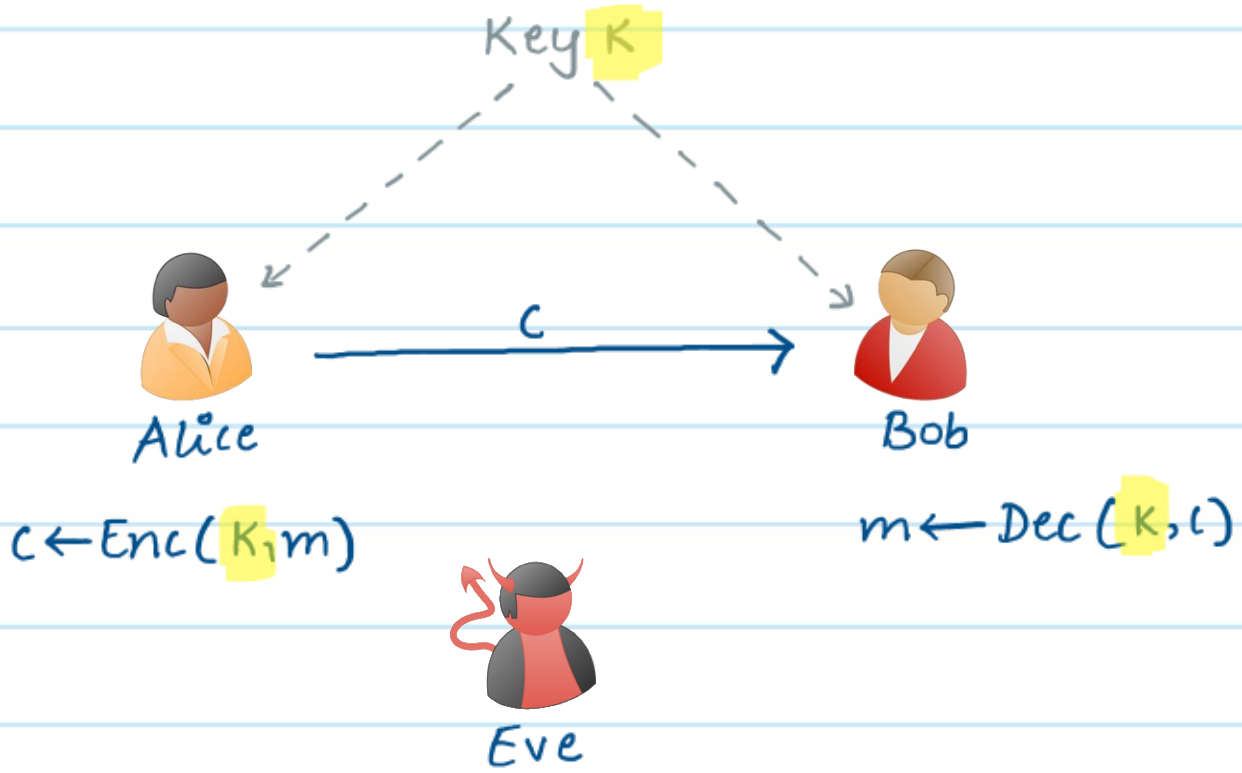
Agenda

* Recap of Public-Key Encryption.
* Example Problems.

* HW4 is due on April 19.

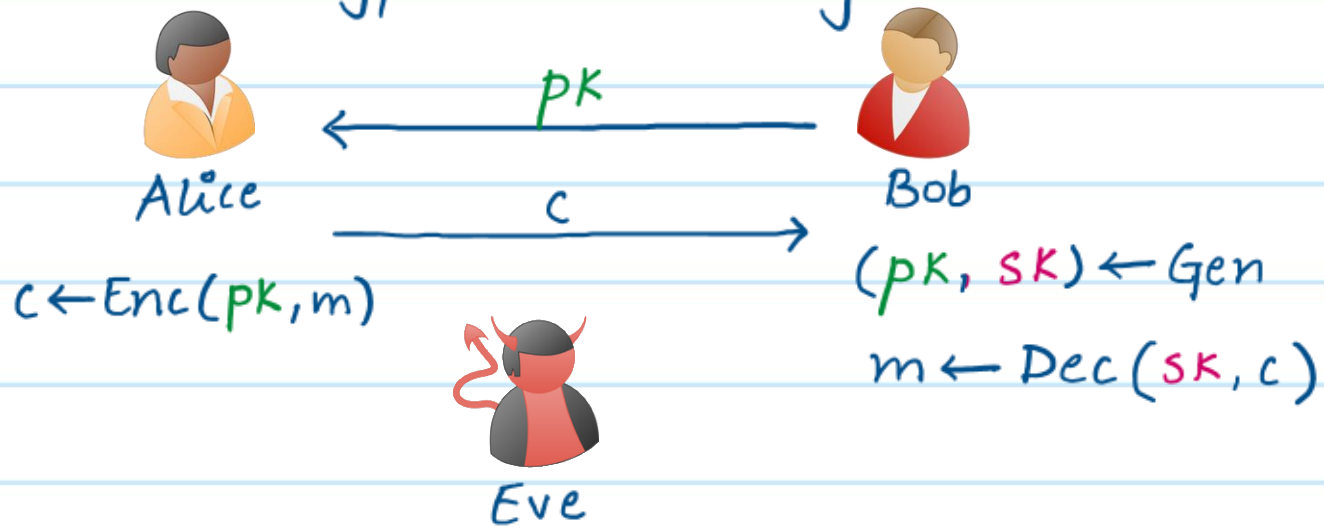
Secret-Key Encryption (SKE)

* So far, when discussing encryption schemes, we assumed that Alice and Bob use the same *secret-key* for both encryption and decryption.
* Such schemes are called secret-key encryption or symmetric-key encryption.



Public-Key Encryption

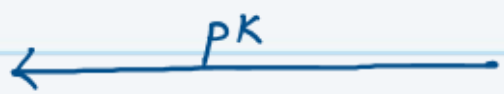
* Encryption Key = public Key (pk)
* Decryption Key = secret Key (sk)
* If Bob wants to receive a private message from Alice, he can generate a (pk, sk) pair and send pk to Alice.
* Alice uses pk to encrypt the message.
* Bob uses sk to decrypt the message

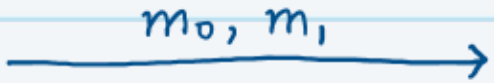


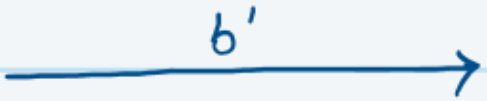
One-message Security

Adversary

Challenger







$$Pr[b = b'] \leq \frac{1}{2} + \nu(n).$$

OR equivalently: $\{pk, Enc(pk, m_0)\} \approx_c \{pk, Enc(pk, m_1)\}$

$\forall m_0, m_1 \in \mathcal{M}.$

Multi-Message Security

Adversary

Challenger

pk

$(pk, sk) \leftarrow Gen(1^n)$

m_0^1, m_1^1

$b \xleftarrow{\$} \{0,1\}$

c^1

$c^1 = Enc(pk, m_b^1)$

m_0^2, m_1^2

$c^2 = Enc(pk, m_b^2)$

c^2

$\vdots$

$\vdots$

m_0^k, m_1^k

$c^k = Enc(pk, m_b^k)$

c^k

b'

$$Pr[b = b'] \leq \frac{1}{2} + \nu(n)$$

Recall Properties of PKE

* The encryption algorithm in any PKE scheme cannot be deterministic.

* It is impossible to design a perfectly secure PKE scheme.
(secure against unbounded adversaries)

* Any PKE that is one-message secure must also be multi-message secure.

i.e., for PKE schemes one-message security ⇒ multi-message security.

* Recall that this not true for secret-key encryption schemes, where we can design schemes (eg. OTP) that are one-message secure but not multi-message secure.

Problem #1

Ques Recall the ElGamal PKE scheme for encrypting messages $m \in \mathbb{G}$.

* Gen: Sample a cyclic group $(\mathbb{G}, \cdot)$ of order q with generator g.
  Sample $x \xleftarrow{\$} \mathbb{Z}_q$. $pk = (\mathbb{G}, g, q, X = g^x)$, $sk = (\mathbb{G}, g, q, x)$

* Enc(pk, m): Sample $y \xleftarrow{\$} \mathbb{Z}_q$. $c = (Y = g^y, D = m \cdot X^y)$

* Dec(sk, c): Parse $c = (Y, D)$. Compute $m = D \cdot Y^{-x}$.

How can we modify it for encrypting messages $m \in \{0,1\}$?

Ans For $m \in \{0,1\}$:

* Gen: Sample a cyclic group $(\mathbb{G}, \cdot)$ of order q with generator g.
  Sample $x \xleftarrow{\$} \mathbb{Z}_q$. $pK = (\mathbb{G}, g, q, X = g^x)$, $sK = (\mathbb{G}, g, q, x)$

* Enc(pK, m): Sample $y \xleftarrow{\$} \mathbb{Z}_q$. $c = (Y = g^y, D = g^m \cdot X^y)$

* Dec(sK, c): Parse $c = (Y, D)$. Compute $M = D \cdot Y^{-x}$. If $M = g^1$, output $m = 1$, else if $M = g^0 = 1$, output $m = 0$.

Ques Can we use a similar idea for $m \in \mathbb{F}$? ($\mathbb{F}$: any field)

Problem #2

Ques Consider the following variant of ElGammal encryption for $m \in \mathbb{G}$.

* Gen: Sample a cyclic group $(\mathbb{G}, \cdot)$ of order q with generator g.
  Sample $x \xleftarrow{\$} \mathbb{Z}_q$. $pK = (\mathbb{G}, g, q, X = g^x)$, $sK = (\mathbb{G}, g, q, x)$

* Enc(pK, m): Sample $y \xleftarrow{\$} \mathbb{Z}_q$. $c = (Y = g^y, D = m \cdot X \cdot Y)$

* Dec(sK, c): Parse $c = (Y, D)$. Compute $m = D \cdot g^{-x} \cdot Y^{-1}$

Is this a secure PKE scheme?

Ans No! An adversary who has $PK = (G, g, q, X)$ can easily decrypt any ciphertext $c = (Y, D)$ by computing $m = D \cdot X^{-1} \cdot Y^{-1}$.

Problem #3

Ques Let (Gen_1, Enc_1, Dec_1) and (Gen_2, Enc_2, Dec_2) be two PKE schemes, only one of these is secure. But we don't know which one is secure.

Now consider the following scheme:

* Gen: $(pk_1, sk_1) \leftarrow Gen_1$, $(pk_2, sk_2) \leftarrow Gen_2$. Output $pk = (pk_1, pk_2)$, $sk = (sk_1, sk_2)$.
* $Enc(pk, m)$: $c = Enc_1(pk_1, Enc_2(pk_2, m))$

Propose a corresponding decryption algorithm. Is this a secure PKE?

Ans - $Dec(sk, c)$: $Dec_2(sk_2, Dec_1(sk_1, c)) = m$

- Yes, this is a secure scheme.

Why?

To show that this scheme is secure, we need to show:

$$\{pk, Enc_1(pk_1, Enc_2(pk_2, m_0))\} \approx_c \{pk, Enc_1(pk_1, Enc_2(pk_2, m_1))\}$$

Case1: Let's assume (Gen_1, Enc_1, Dec_1) is secure.

Let $c_0 = Enc_2(pk_2, m_0)$ & $c_1 = Enc_2(pk_2, m_1)$.

From security of (Gen_1, Enc_1, Dec_1) it then follows that:

$$\{pk, Enc_1(pk_1, c_0)\} \approx_c \{pk, Enc_1(pk_1, c_1)\}$$

Case2: Let's assume (Gen_2, Enc_2, Dec_2) is secure.

From security of (Gen_2, Enc_2, Dec_2) it follows that:

$$\{pk, Enc_2(pk_2, m_0)\} \approx_c \{pk, Enc_2(pk_2, m_1)\}$$

From closure property of computational indistinguishability,

we get $\{pk, Enc_1(pk_1, Enc_2(pk_2, m_0))\} \approx_c \{pk, Enc_1(pk_1, Enc_2(pk_2, m_1))\}$

Problem #4

Ques Let (Gen, Enc, Dec) be a secure PKE for encrypting messages $m \in \{0,1\}$. How can we use this to encrypt messages $m \in \{0,1\}^n$?

Ans Gen' : $(sk, pk) \leftarrow Gen$

$Enc'(pk, m)$: Let $m = m_1 \ldots m_n$. $\forall i \in [n]$: $c_i = Enc(pk, m_i)$.

Output $c = c_1, \ldots, c_n$

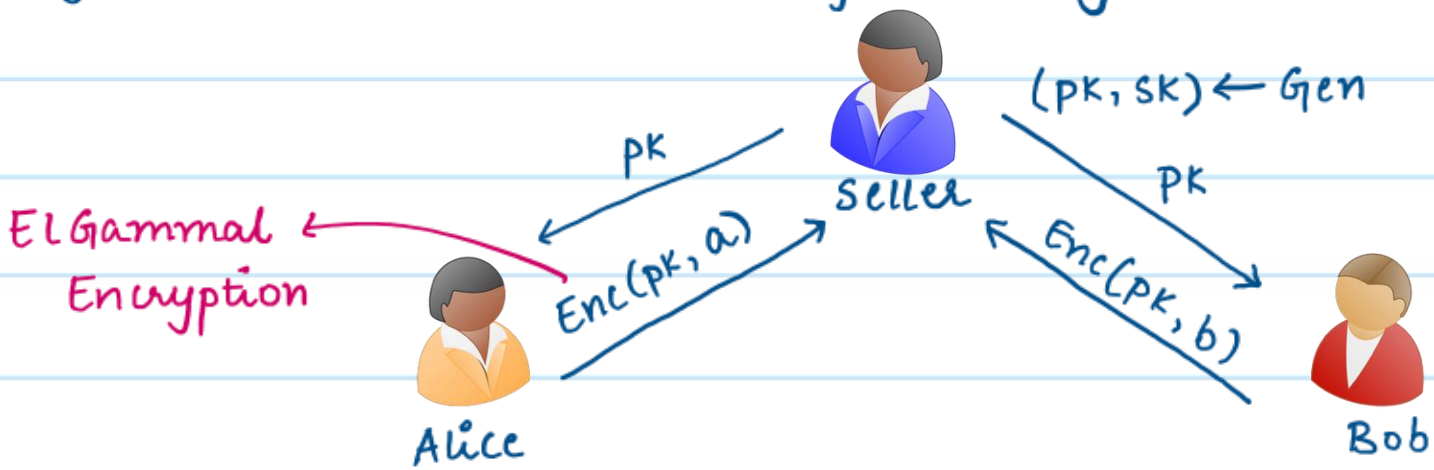
$Dec'(sk, c)$: Parse $c = c_1 \ldots, c_n$. $\forall i \in [n]$. $m_i = Dec(sk, c_i)$.

Output $m = m_1, \ldots, m_n$.

Security of this scheme follows from multi-message security of the underlying bit-encryption scheme (Gen, Enc, Dec).

Problem #5

Ques We want to design a sealed-bid auction. Assume the seller is honest and there are two potential buyers - Alice and Bob. First Alice submits her bid and then Bob submits his bid. A natural security requirement from a sealed-bid auction scheme is that Bob should not be able to choose his bid based on the bid made by Alice, since otherwise Bob can always outbid Alice. Is the following a secure sealed-bid auction?



(all these messages are sent over public communication channels)

Ans No!

* Let's assume Alice's ciphertext is of the form:
  $(Y = g^y, D = a \cdot X^y)$.

* An adversarial Bob can design a modified ciphertext:
  $(Y' = Y^2, D' = D^2)$
  this is a valid ElGammal encryption using random value $2y$ of message $b = a^2$.
* Bob using this strategy will always be able to outbid Alice.

Take Away: ElGammal Encryption is prone to malleability attacks.