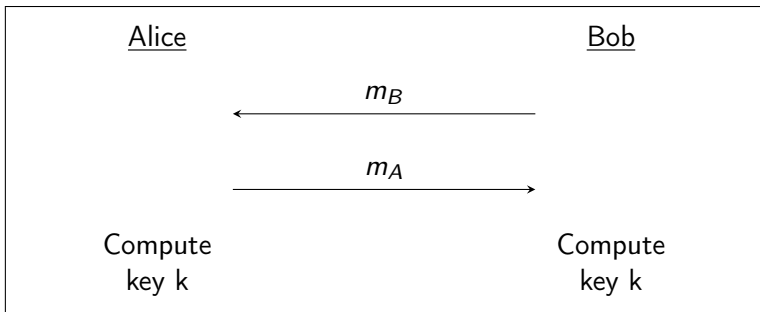


Lecture 29: 2-round Key Agreement and Public-key Encryption

- Suppose there is a 2-round Key-Agreement protocol. This means that there exists a protocol where
 - Bob sends the first message m_B
 - Alice sends the second message m_A
 - Now, parties can compute a secret key key that is hidden from an eavesdropper (who got to see the first message by Bob and the second message by Alice)
 - For example, Diffie-Hellman key-exchange protocol. Bob sends $m_B = g^b$, Alice sends $m_A = g^a$, and both parties compute the key $key = g^{ab}$, but it remains hidden from the adversary.
- Using this 2-round key-agreement protocol we can construct a public-key encryption scheme. For example, using the Diffie-Hellman key-exchange protocol, we shall construct the ElGamal public-key encryption scheme

First Component: 2-round Key-Agreement Protocol I

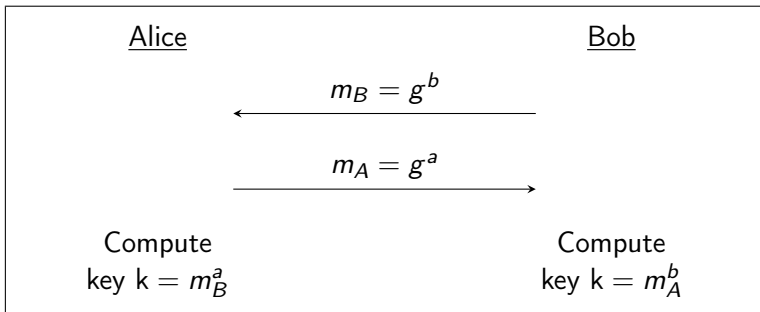
- Suppose we have a protocol Π_{2-KA} , which is a 2-round key-agreement protocol that looks like the following



- Note that Π_{2-KA} can be any 2-round key-agreement protocol. One such example is the Diffie-Hellman key-agreement protocol. The next slide presents this protocol in this template.

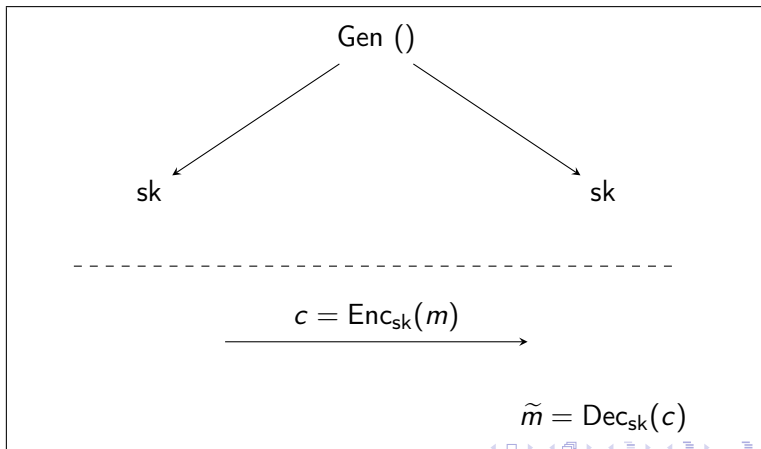
First Component: 2-round Key-Agreement Protocol II

- For example, we consider $\Pi_{2\text{-KA}}$ to be the Diffie-Hellman key agreement protocol



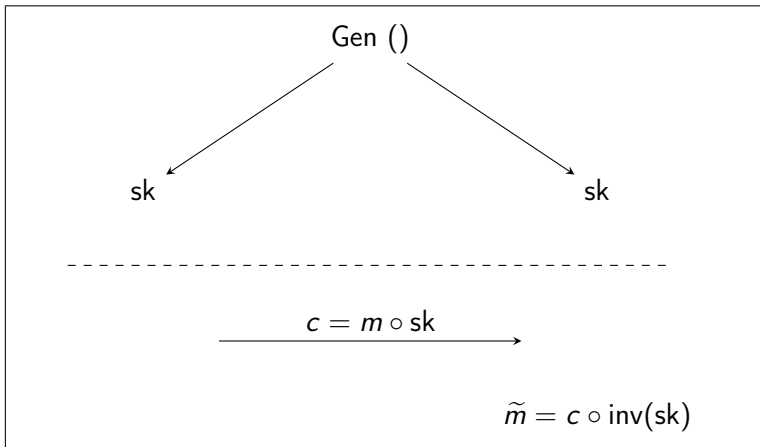
Second Component: Private-key Encryption I

- Suppose we have a private-key encryption scheme $(\text{Gen}, \text{Enc}, \text{Dec})$. Without loss of generality, we can assume that $\text{Gen}()$ outputs a uniformly random key sk from a set S . Recall that a private-key encryption scheme looks as follows



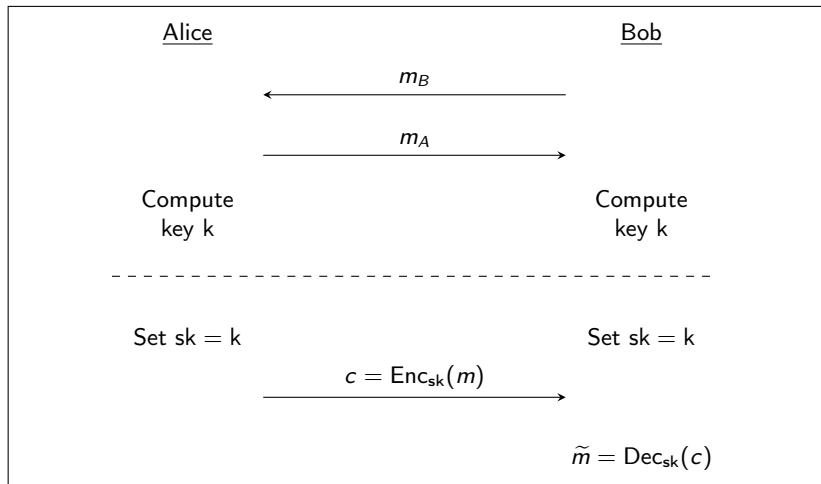
Second Component: Private-key Encryption II

- Consider, for example, the one-time pad encryption scheme



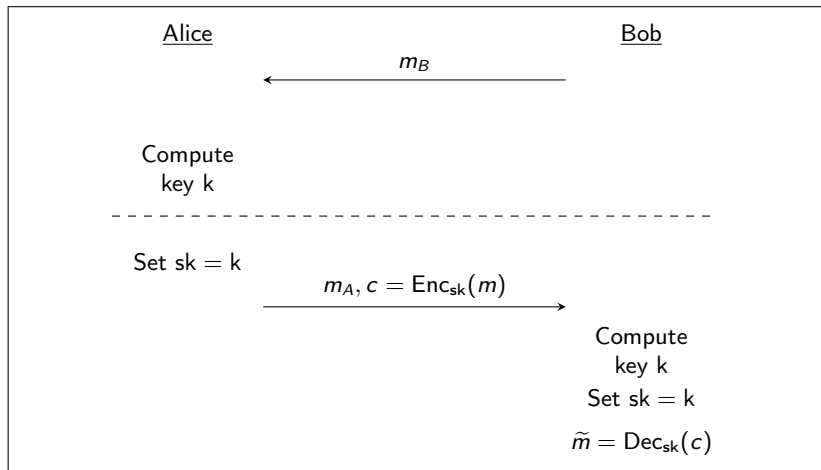
Combining to obtain a Public-key Encryption Scheme I

If the key of the first component is random over the set S (from which the private-key of the second-component is chosen) then we can stick together these two protocols as follows



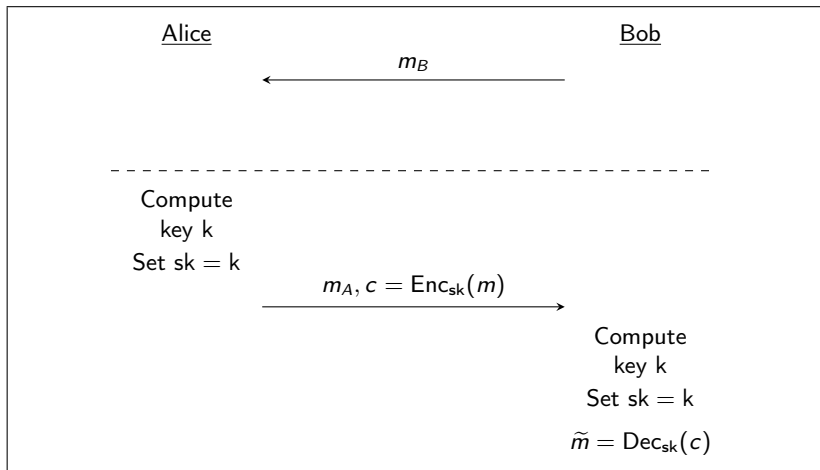
Combining to obtain a Public-key Encryption Scheme II

We can merge the message m_A and c into one-single message. And we get the following scheme.



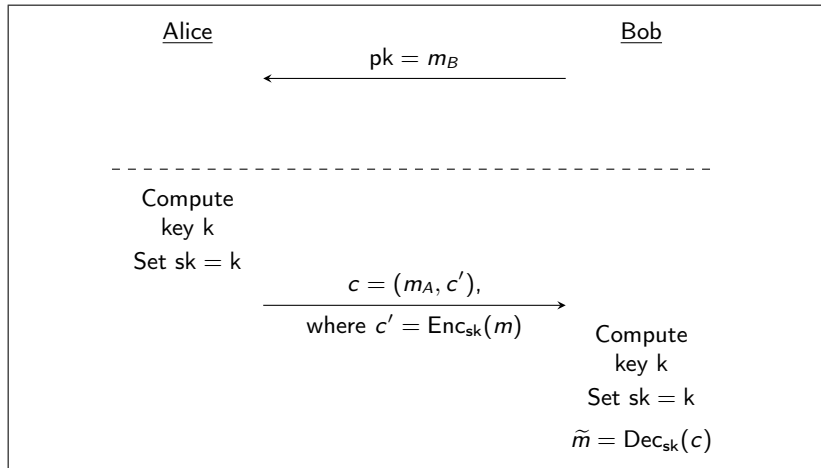
Combining to obtain a Public-key Encryption Scheme III

Every time we want to encrypt a message m , we calculate a fresh key k . And we get the following scheme.



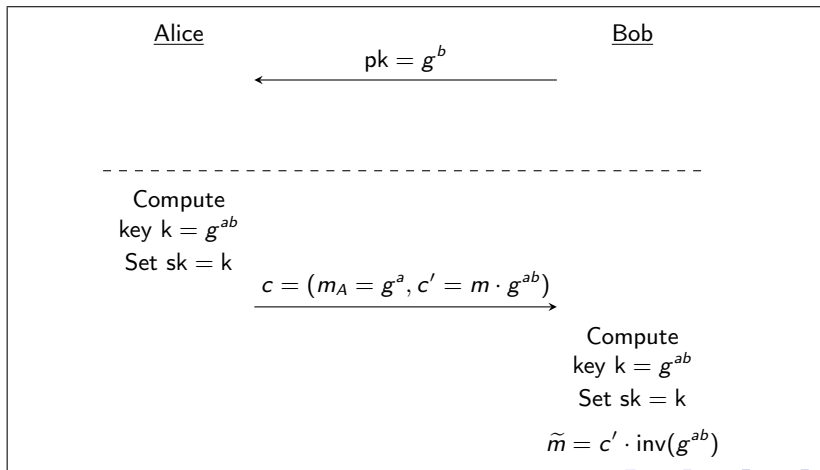
Combining to obtain a Public-key Encryption Scheme IV

Finally, we interpret the message m_B as the public-key for Bob. And the messages (m_A, c) as the encryption of the message m . This gives us our public-key encryption scheme!



Example 1

- Suppose our first component is Diffie-Hellman key-agreement protocol and the second component is one-time pad. Then we get the following public-key encryption scheme.



Example II

This is the ElGamal public-key encryption scheme!