Public Key Cryptography

Rohit Musti

CUNY - Hunter College

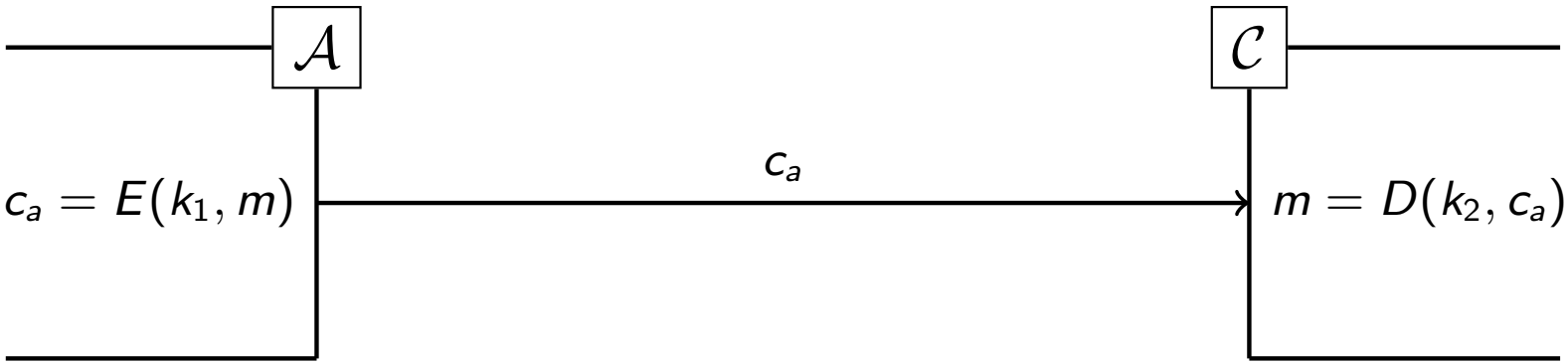
March 2, 2022

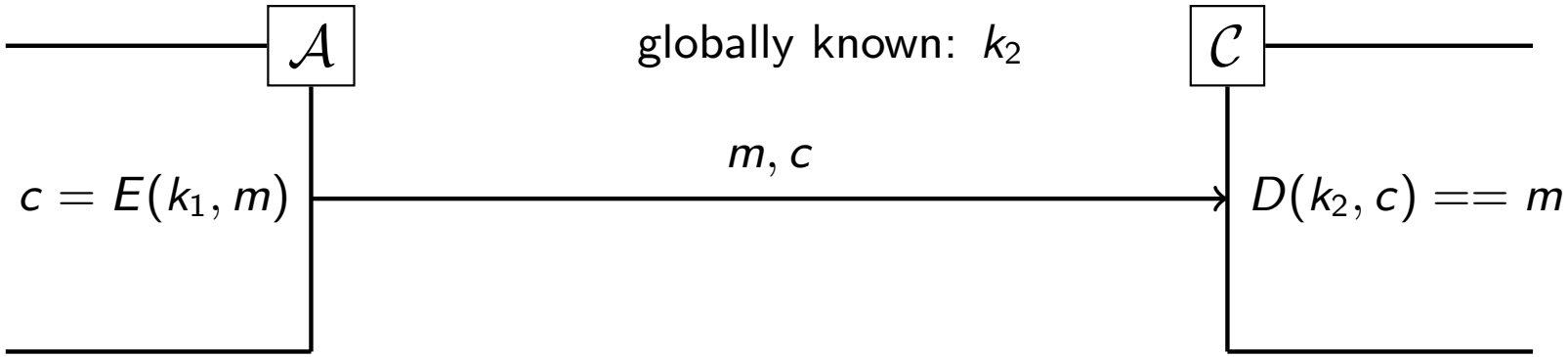
Outline

Overview

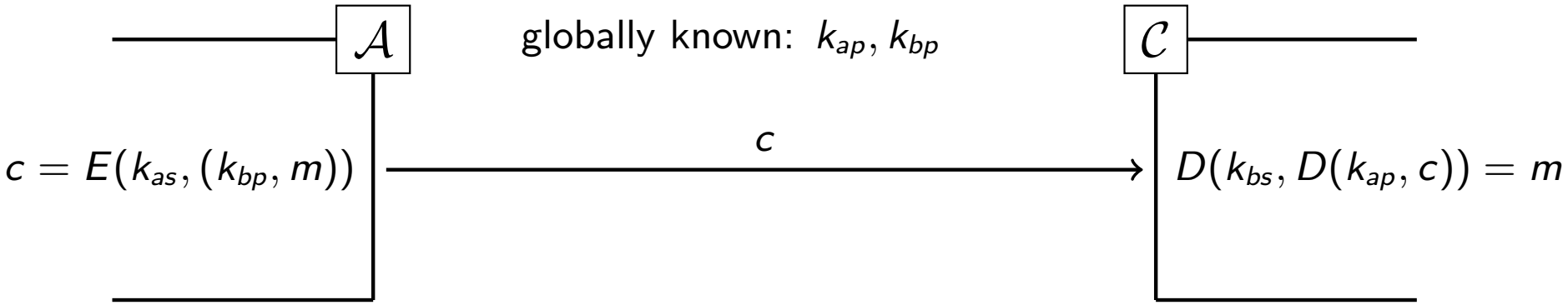
- Up until now we have considered systems that have a shared secret key
- Essentially you use the same key for encrypting and decrypting. Problem! You need to share the Key
- Enter Asymmetric cryptography where you have two separate but linked keys.

Asymmetric Keys: Secrecy + Integrity





Asymmetric Keys: Authenticated Encryption



Example: Email

- With email it is important to be able to easily verify that the email wasn't tampered with and not read by any bad actors.
- It is easy to see how using public key encryption systems, you could design an email protocol to handle sending secure information. (HINT: future hw problem)

Example: Sharing Encrypted Files

- How would you efficiently encrypt files on a shared file system where you wanted to grant access to specific users?
- Encrypt the file f with a symmetric cipher using a secret key k.
- To share that file with someone else, encrypt the key used to encrypt a particular file with that person's public key and store the result in the file header
- You have encrypted the file once, scaling easily to many users, and you can easily add new users in the optimal space complexity.

Example:

- How would you efficiently encrypt files on a shared file system where you wanted to grant access to specific users?
- Encrypt the file f with a symmetric cipher using a secret key k.
- To share that file with someone else, encrypt the key used to encrypt a particular file with that person's public key and store the result in the file header
- You have encrypted the file once, scaling easily to many users, and you can easily add new users in the optimal space complexity.