CSE 564

Some Crypto Background

Winter 2019

Franziska (Franzi) Roesner
franzi@cs.washington.edu

Thanks to Dan Boneh, Dieter Gollmann, Dan Halperin, Yoshi Kohno, Ada Lerner, John Manferdelli, John Mitchell, Vitaly Shmatikov, Bennet Yee, and many others for sample slides and materials ...

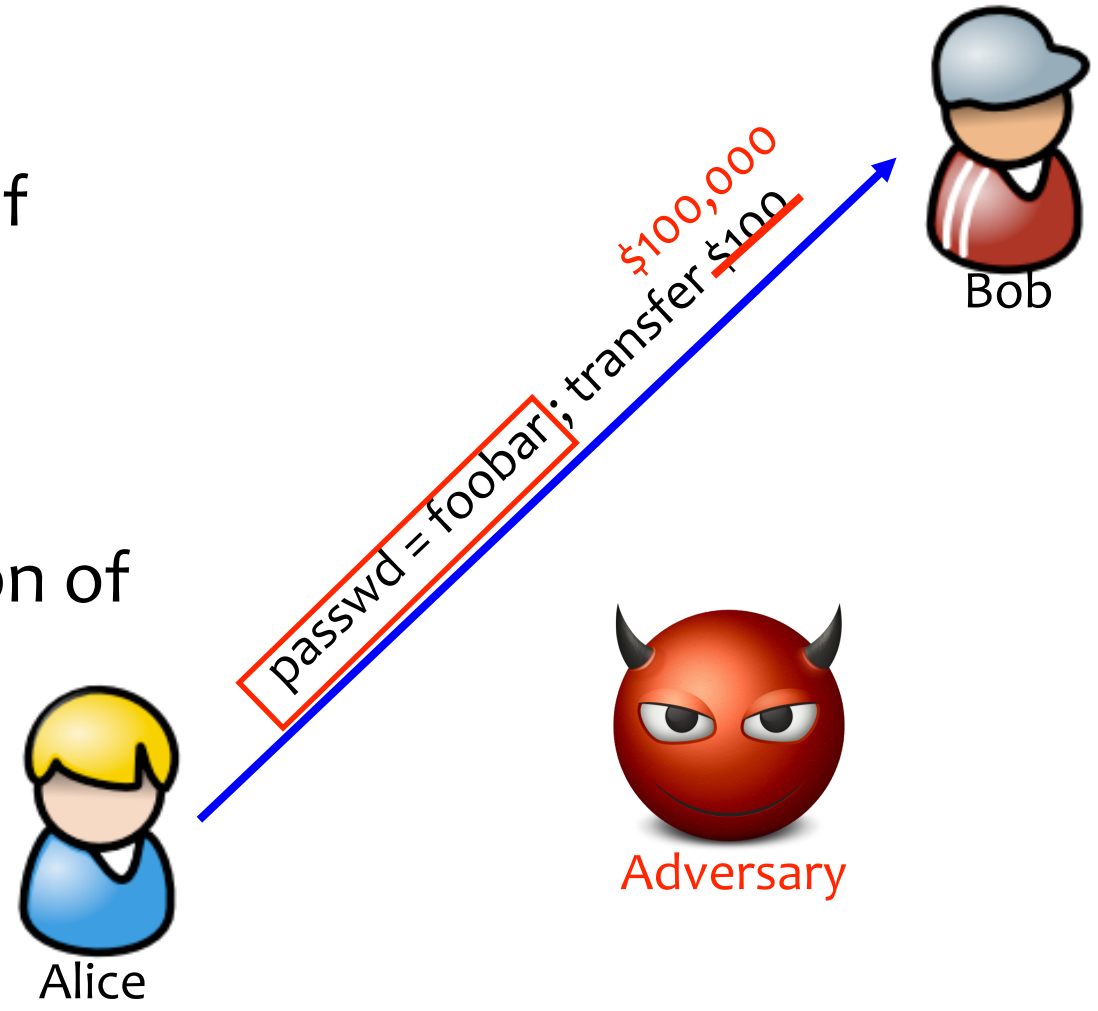
Common Communication Security Goals

Privacy of data:

Prevent exposure of information

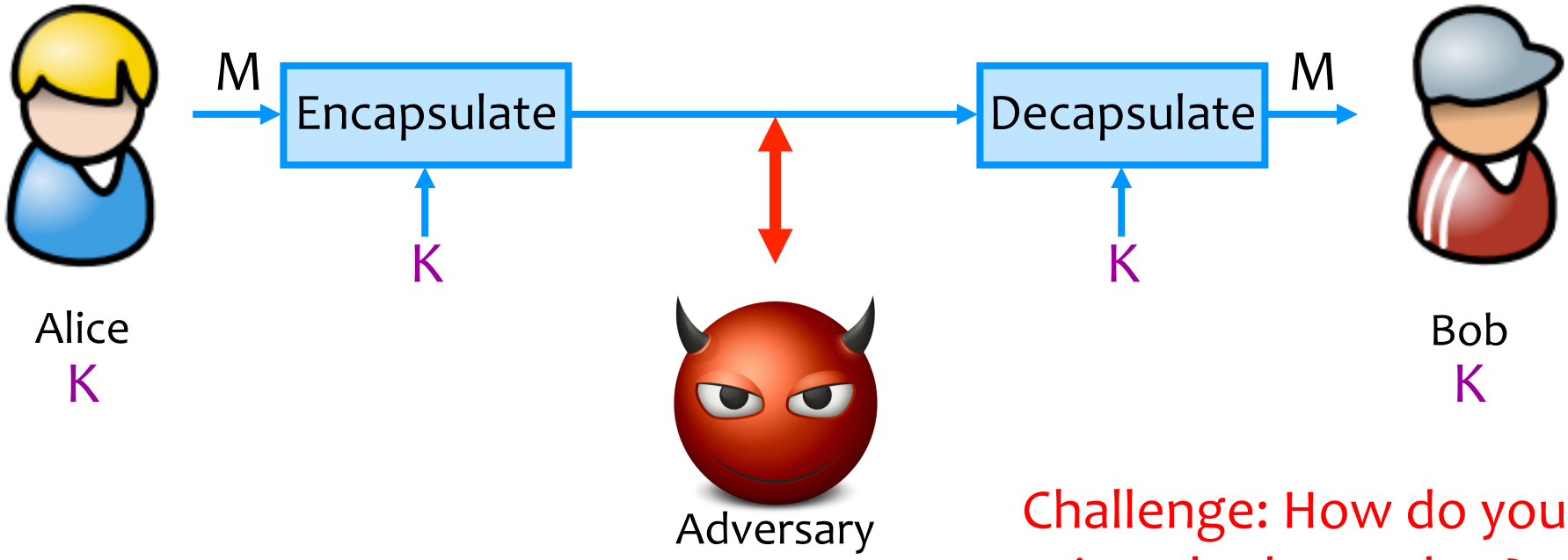
Integrity of data:

Prevent modification of information



Symmetric Setting

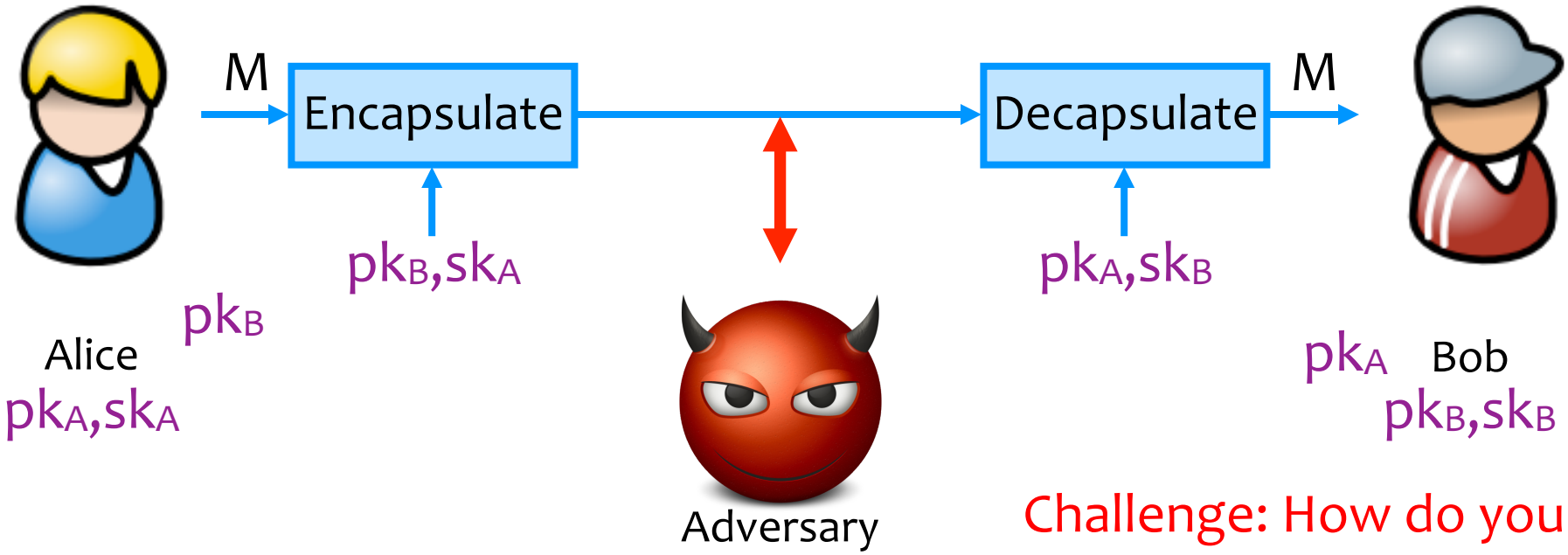
Both communicating parties have access to a shared random string K, called the key.



Challenge: How do you privately share a key?

Asymmetric Setting

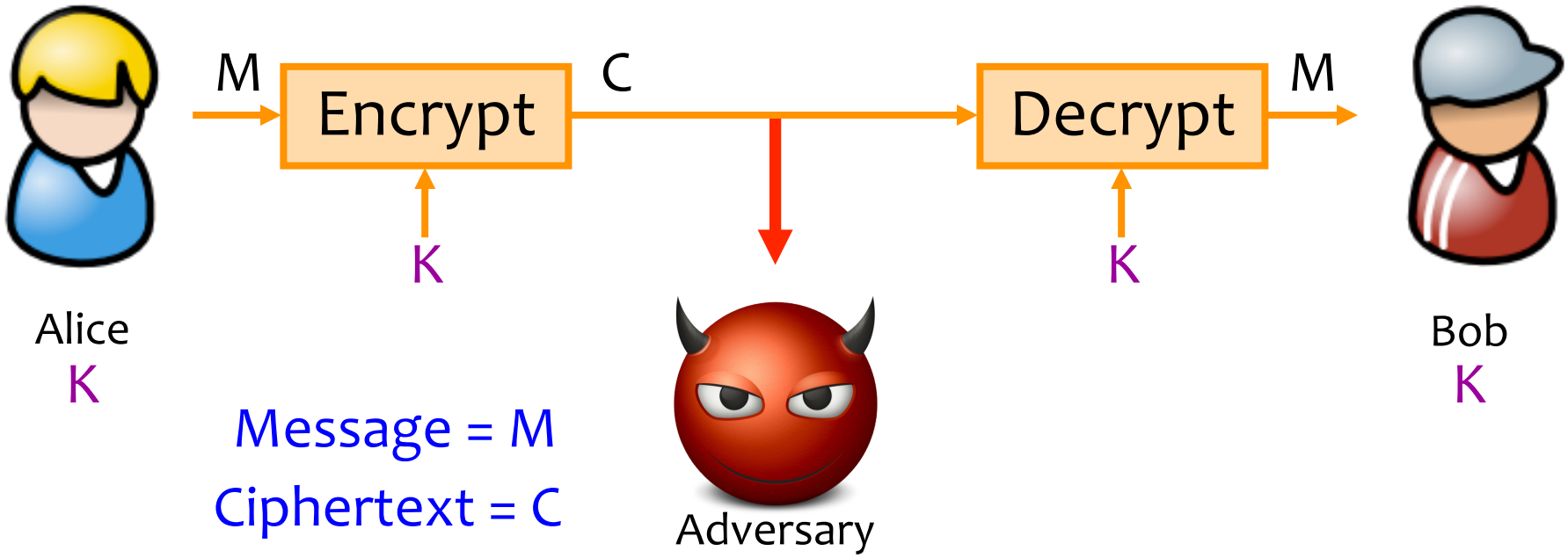
Each party creates a public key pk and a secret key sk.



Challenge: How do you validate a public key?

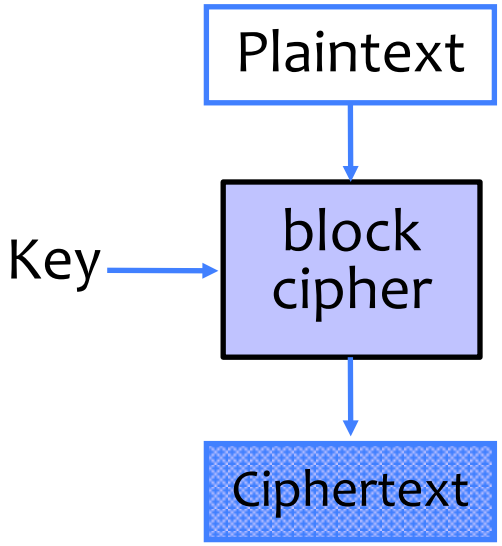
Achieving Privacy (Symmetric)

Encryption schemes: A tool for protecting privacy.



Block Ciphers

- Operates on a single chunk ("block") of plaintext
  - For example, 64 bits for DES, 128 bits for AES
  - Each key defines a different permutation
  - Same key is reused for each block (can use short keys)

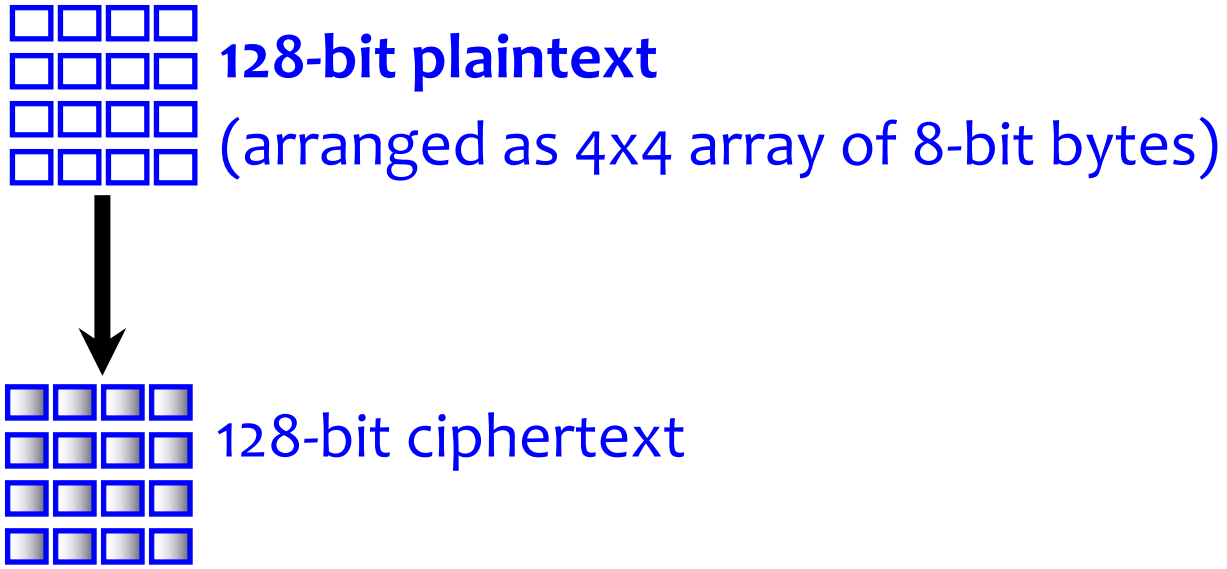


Standard block ciphers:

- DES (deprecated)
- AES

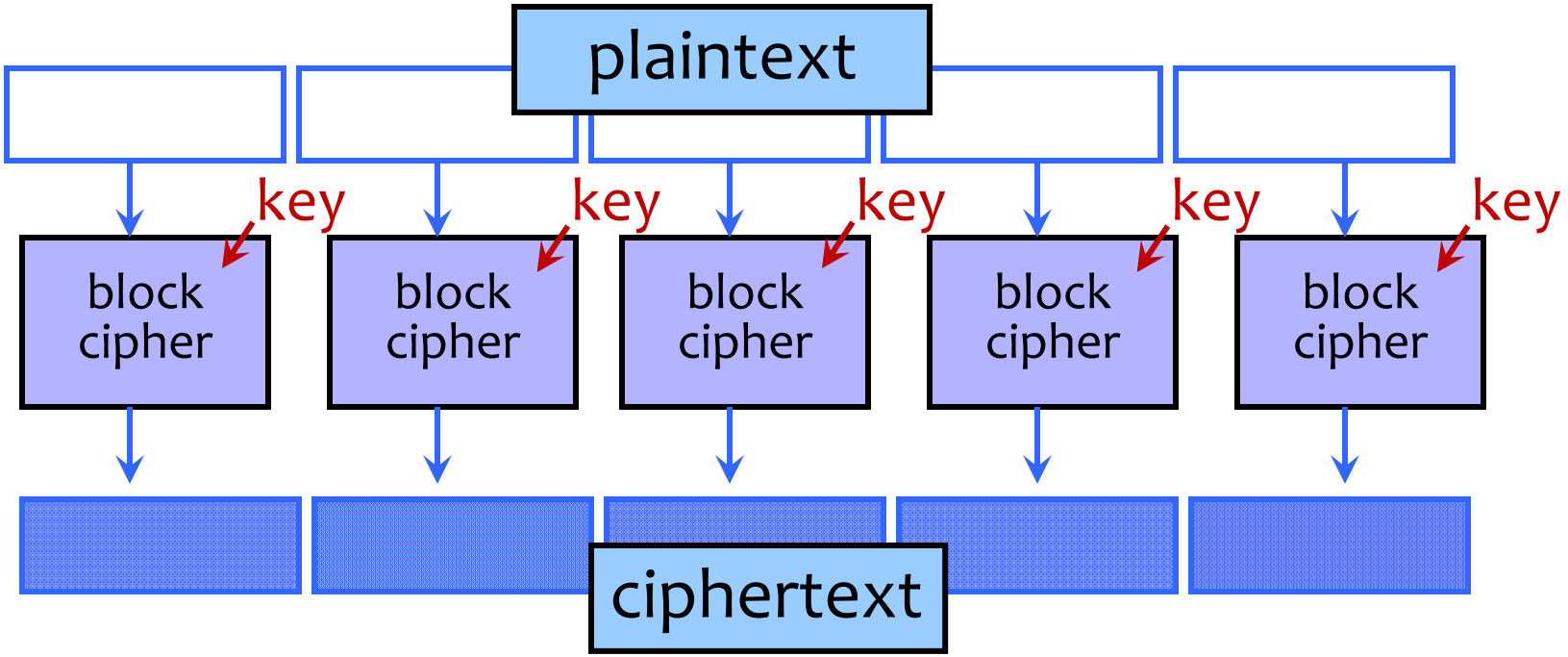
Encrypting a Large Message

- So, we've got a good block cipher, but our plaintext is larger than 128-bit block size



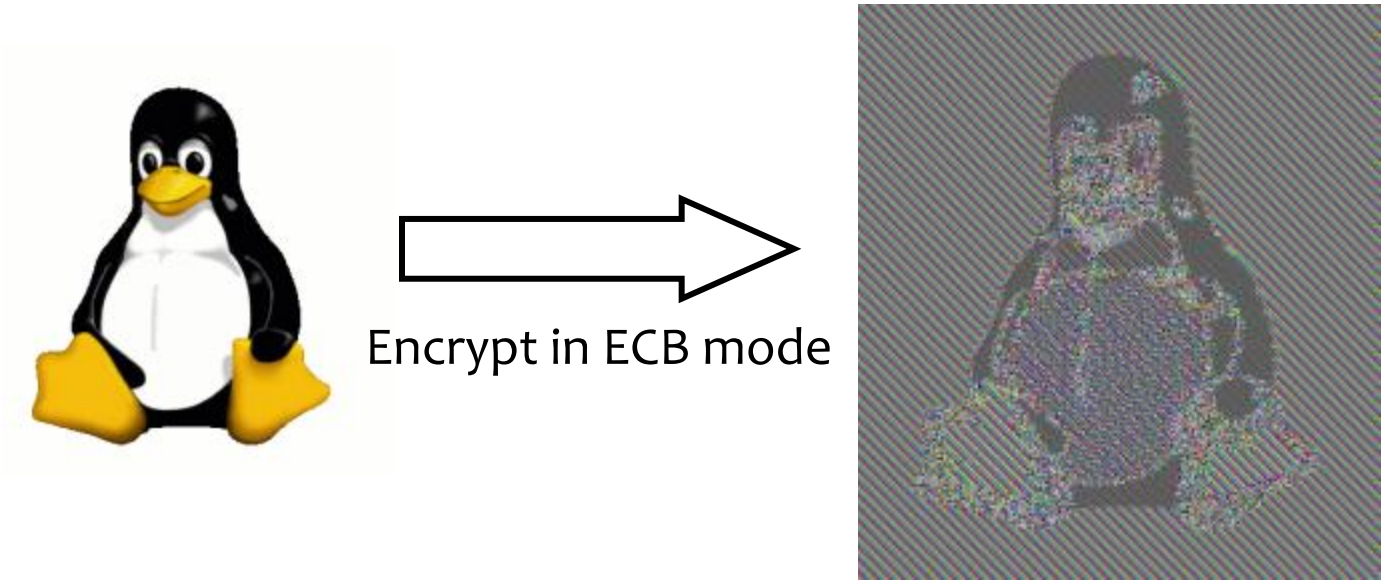
- What should we do?

Electronic Code Book (ECB) Mode



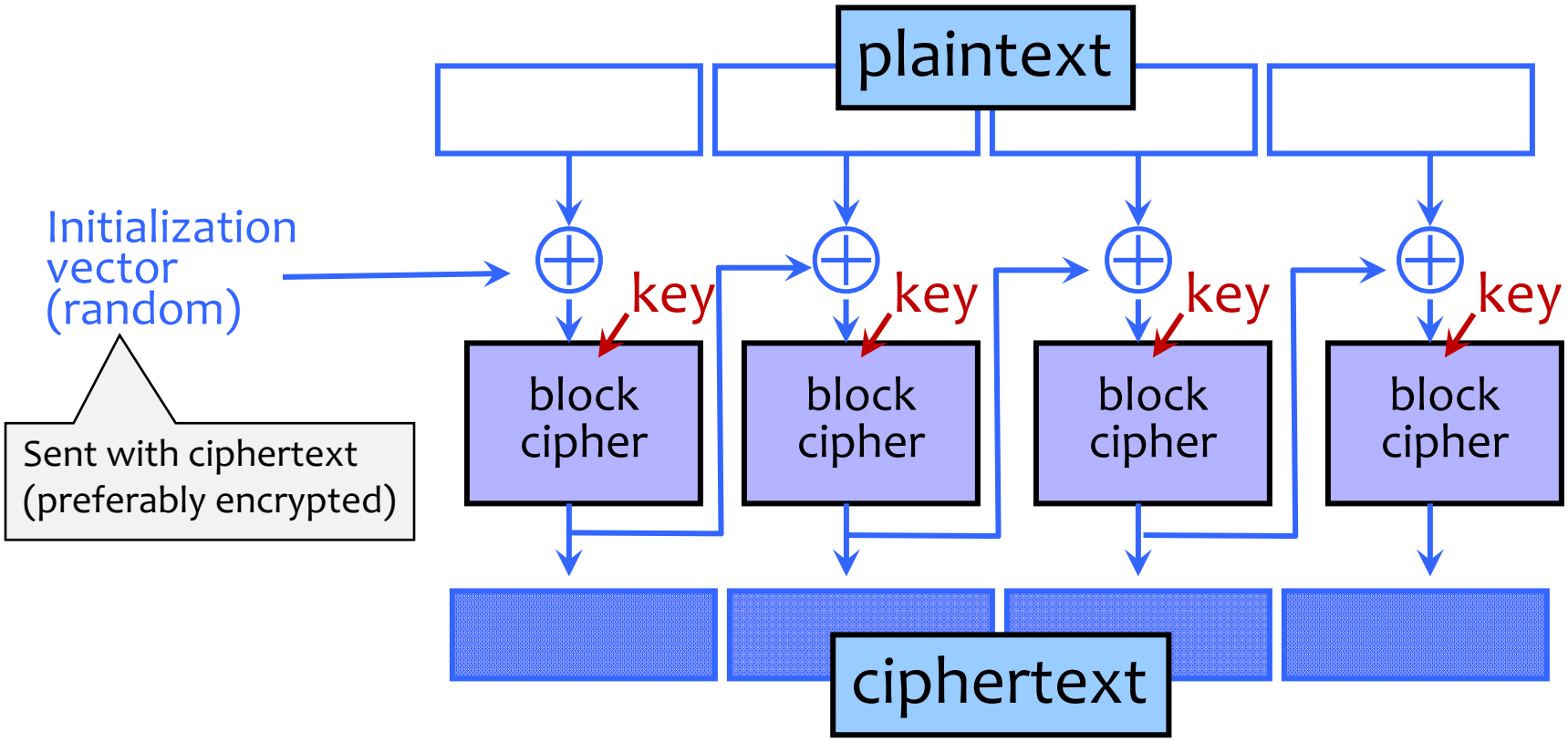
- Identical blocks of plaintext produce identical blocks of ciphertext
- No integrity checks: can mix and match blocks

Information Leakage in ECB Mode



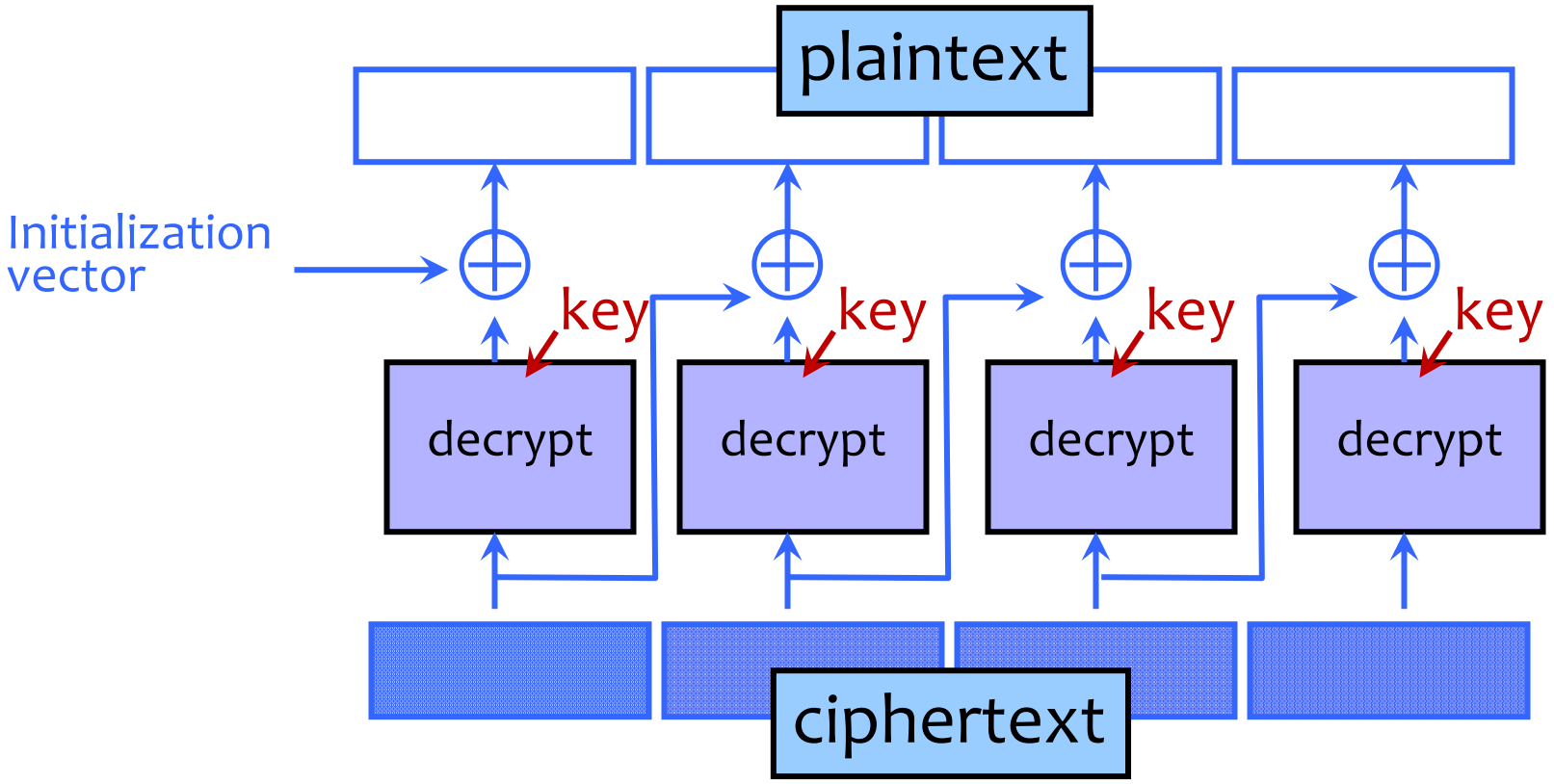
[Wikipedia]

Cipher Block Chaining (CBC) Mode: Encryption

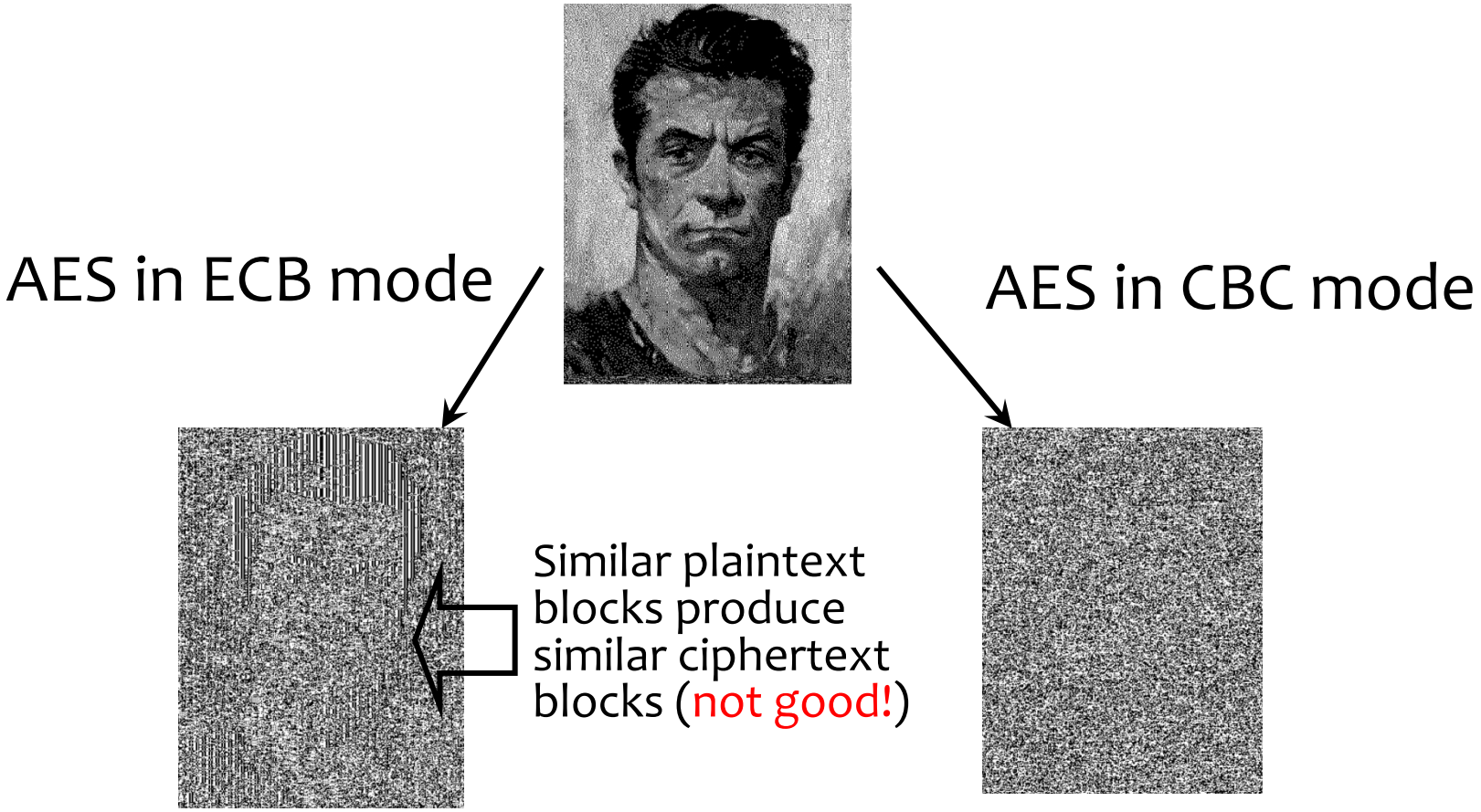


- Identical blocks of plaintext encrypted differently
- Last cipherblock depends on entire plaintext
  - Still does not guarantee integrity

CBC Mode: Decryption

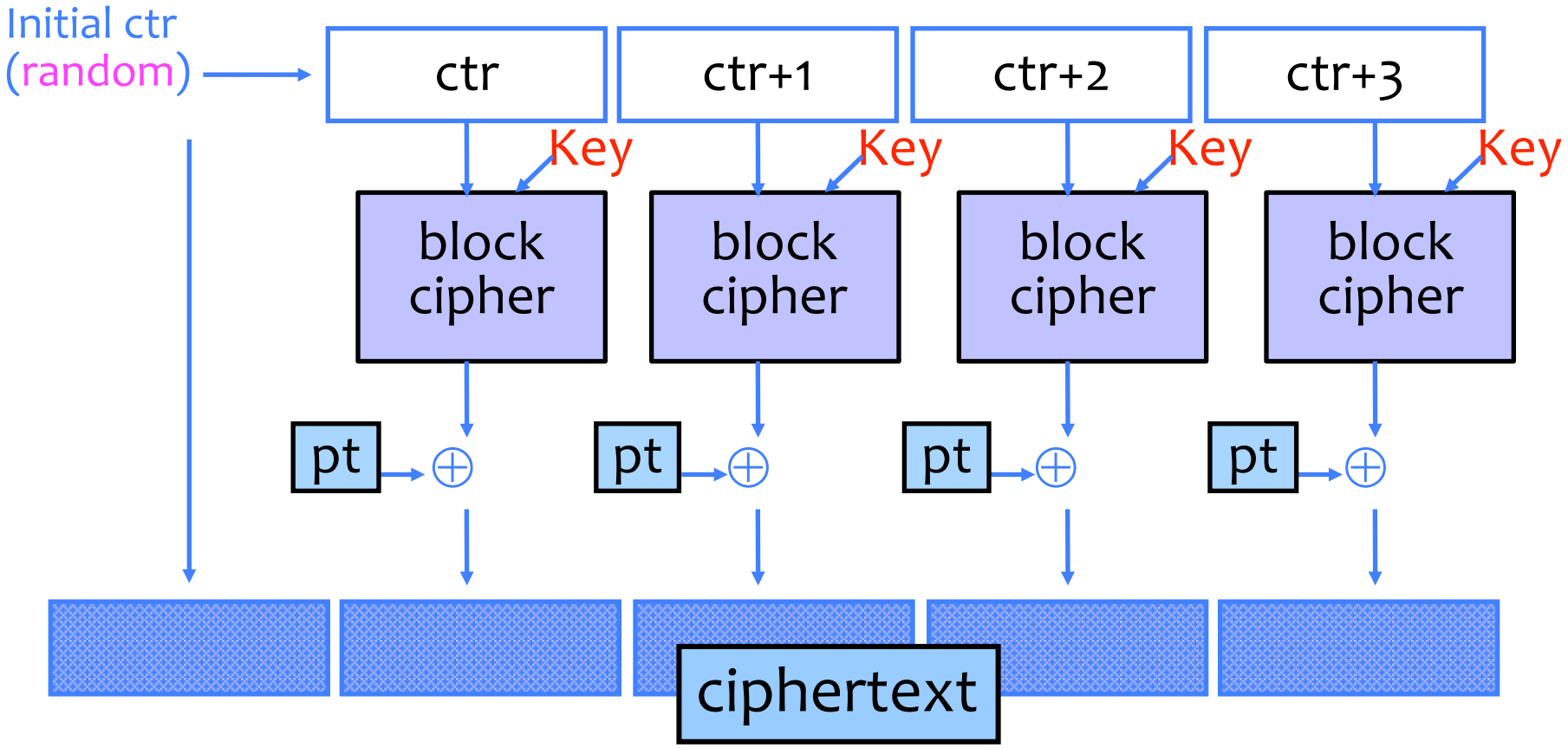


ECB vs. CBC



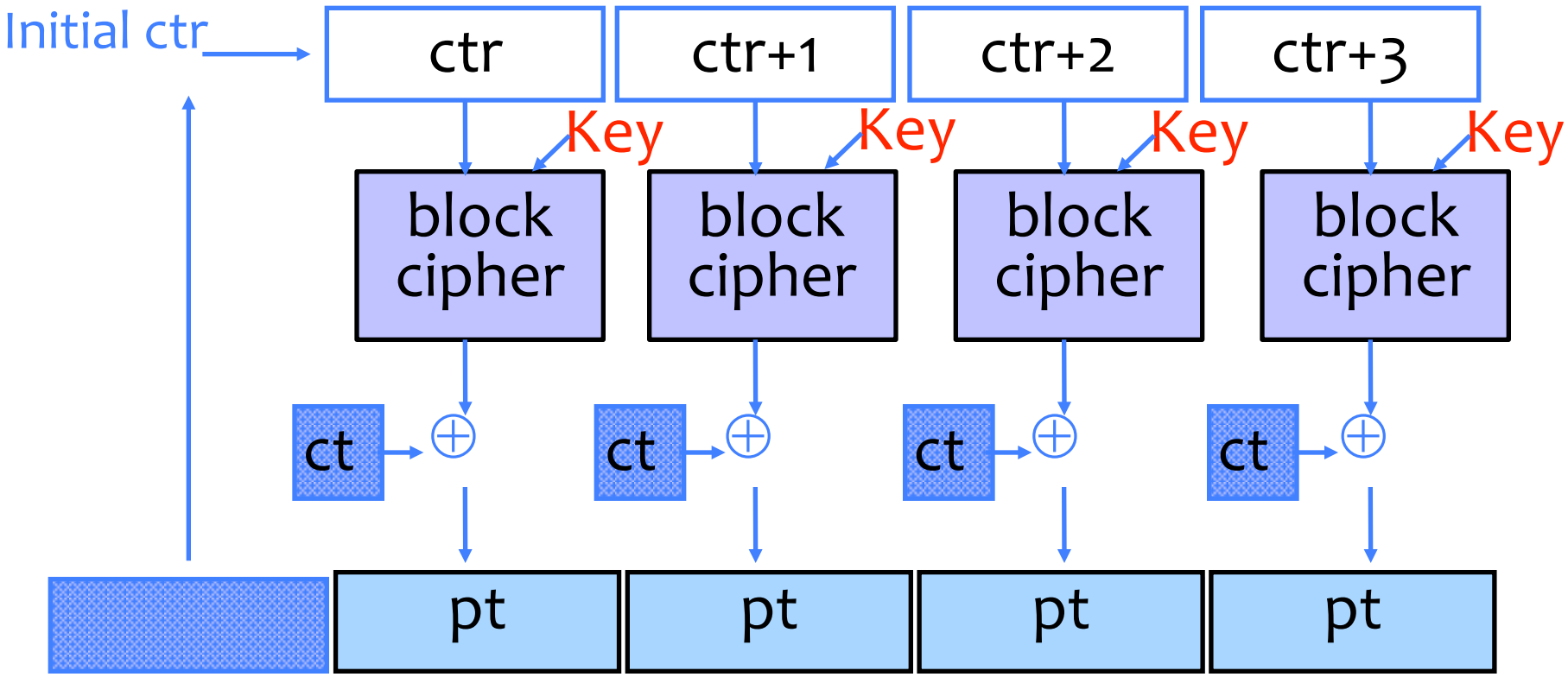
[Picture due to Bart Preneel]

Counter Mode (CTR): Encryption



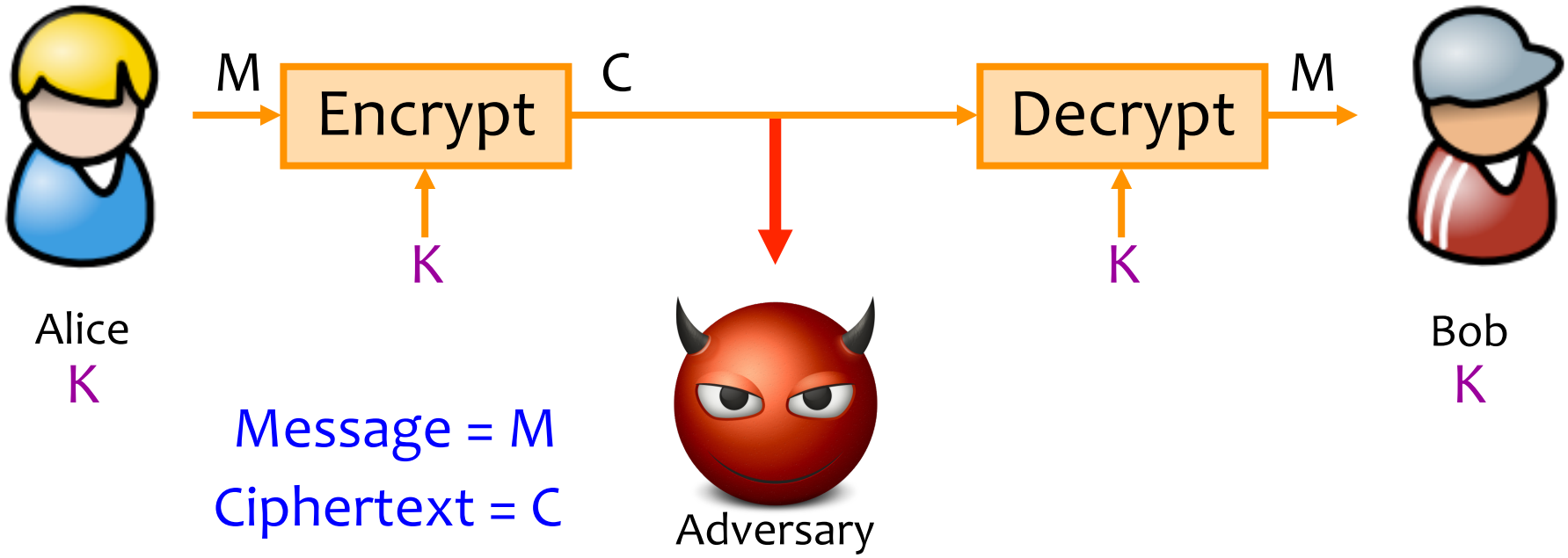
- Identical blocks of plaintext encrypted differently
- Still does not guarantee integrity; Fragile if ctr repeats

Counter Mode (CTR): Decryption



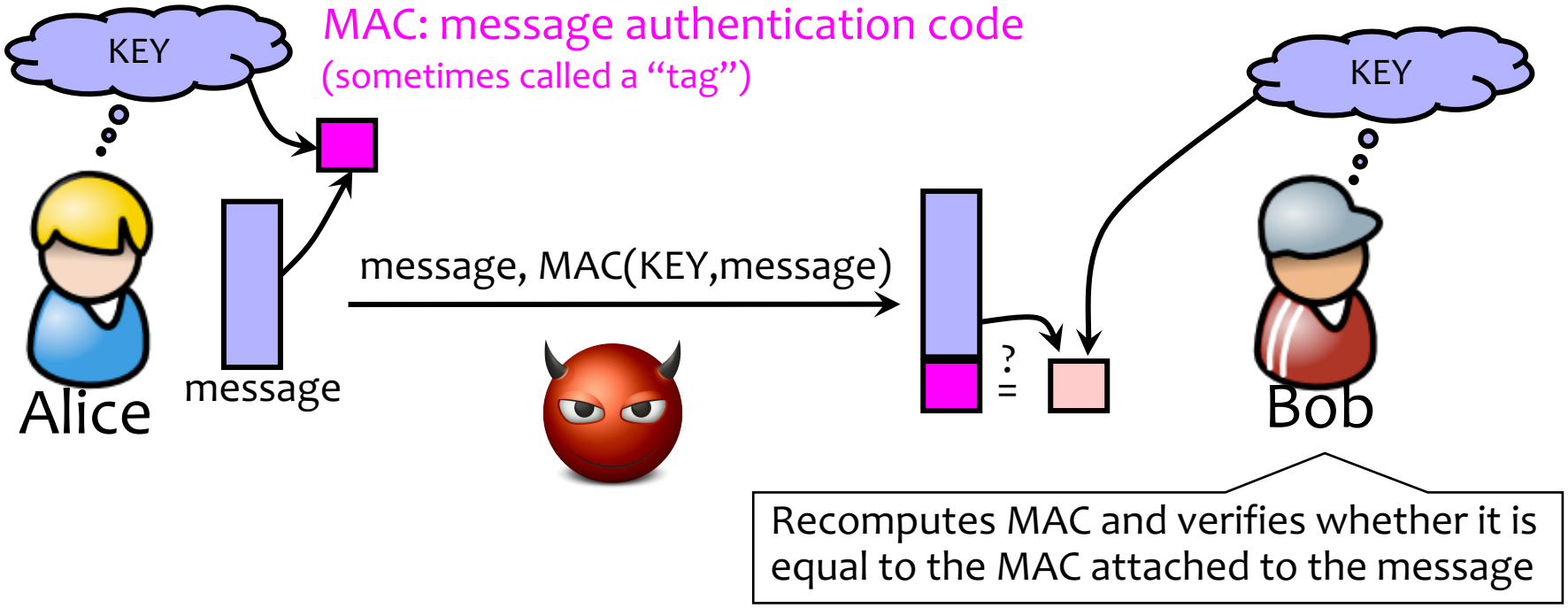
So Far: Achieving Privacy

Encryption schemes: A tool for protecting privacy.



Now: Achieving Integrity

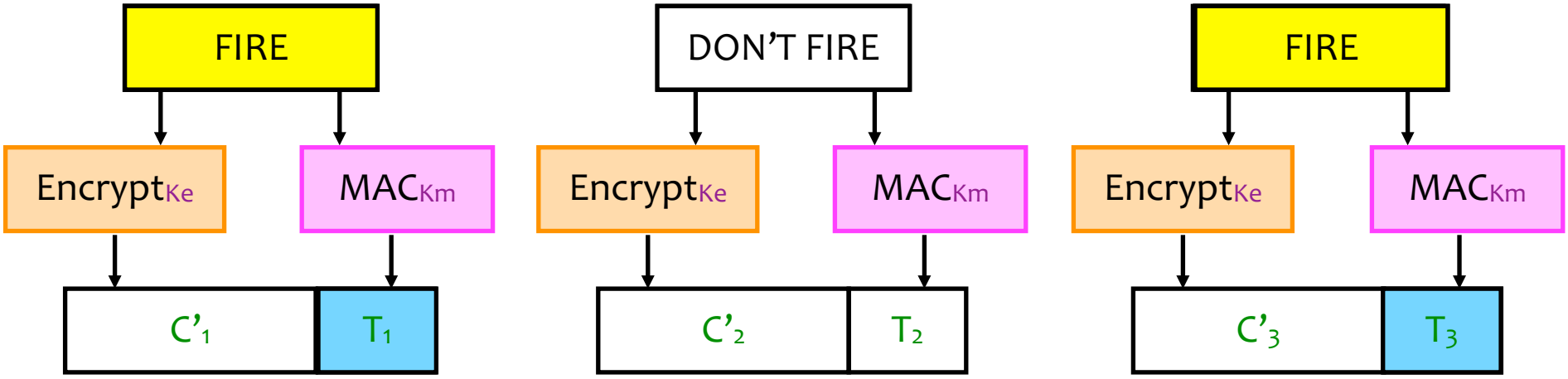
Message authentication schemes: A tool for protecting integrity.



Integrity and authentication: only someone who knows KEY can compute correct MAC for a given message.

Authenticated Encryption

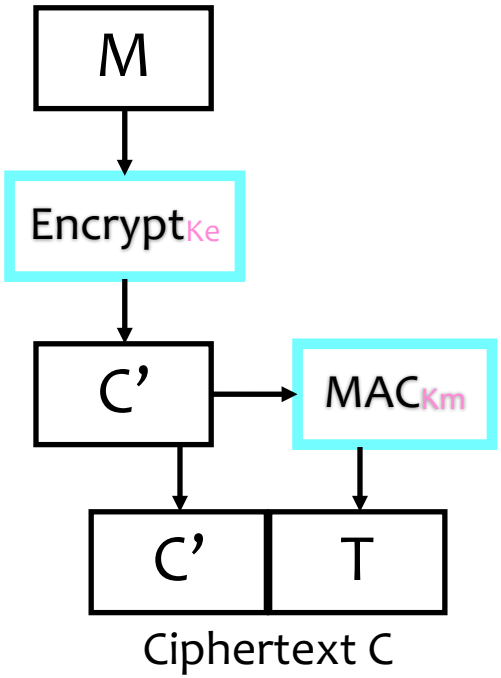
- What if we want <u>both</u> privacy and integrity?
- Natural approach: combine encryption scheme and a MAC.
- But be careful!
  - Obvious approach: Encrypt-and-MAC
  - Problem: MAC is deterministic! same plaintext → same MAC



Authenticated Encryption

- Instead:
  Encrypt *then* MAC.

- (Not as good:
  MAC-then-Encrypt)



Encrypt-then-MAC