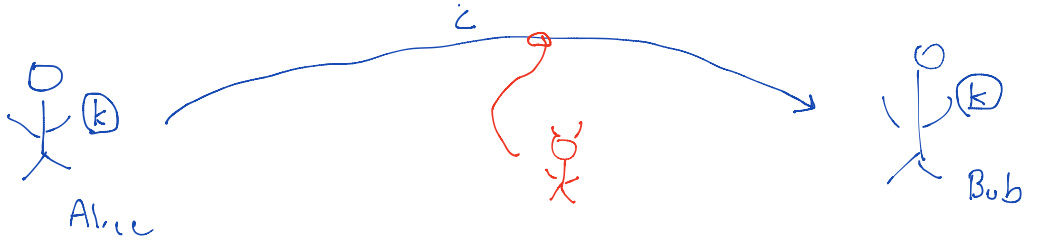
Lecture 5



$c = Enc_k(m)$

$m = Dec_k(c)$

Block Cipher

AES

$\{0,1\}^{128}$

$\{0,1\}^{128} \rightarrow c = Enc_k(m) \in \{0,1\}^{128}$

$Enc(k, m)$

key (same)

$|k| \geq 112$ bits.

$m = Dec_k(c)$

Block Cipher + Mode of Operation

$\{0,1\}^*$

$|c| = |m|$

$c = Enc\_Mode_k(m)$ $\quad m \in \{0,1\}^*$

$m = Dec\_Mode_k(c)$

- ECB → weak (OTS)
- CBC, CTR → med. (CPA)
- GCM → high (CCA)

Properties of a block ciphers (informal)

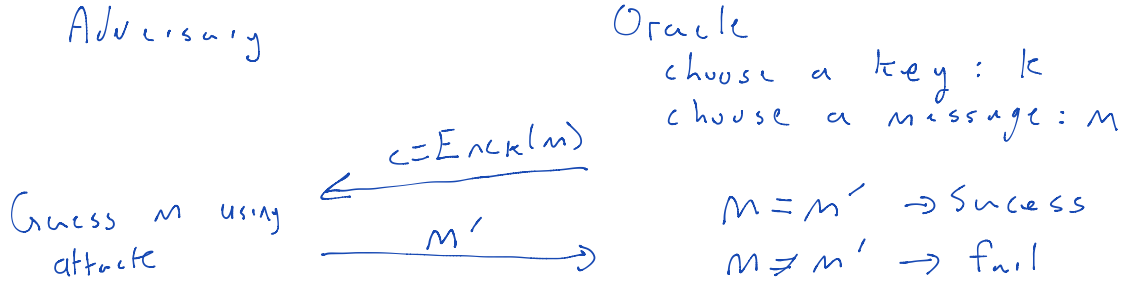
* Encryption is reversible iff you know the secret key.
* Encryption is irreversible iff you don't know the key.
* Key cannot be guessed by exhaustive search (too big: $|K| \geq 112$ bits)
* Avalanche effect: if you change a single bit of input to Enc or Dec, the output block changes randomly

Defining Security more formally

↳ Security game-based def'n is the most commonly used.

↳ 2 Players: Attacker / Adversary, Oracle / Challenger

First Attempt



Def'n: Encryption is secure if any adversary cannot win the game.
↑ computationally-bounded
win ← how often succeed?

Weakness of this definition
↳ it permits encryption schemes that leak some but not all bits of the message to be "secure"

OTS Game (One-time secure)

Adversary		Oracle
		* Choose key k.
* Choose any two messages m_1 & m_2 s.t. $m_0 \neq m_1$ and $\|m_0\| = \|m_1\|$	m_0, m_1 →	* Flip a coin $b \in_R \{0,1\}$ (random choose)
	← c_b	* $c_b = Enc_k(m_b)$
guess = { $b' = 0$ / $b' = 1$	b' →	{ $b = b'$ success / $b \neq b'$ fail

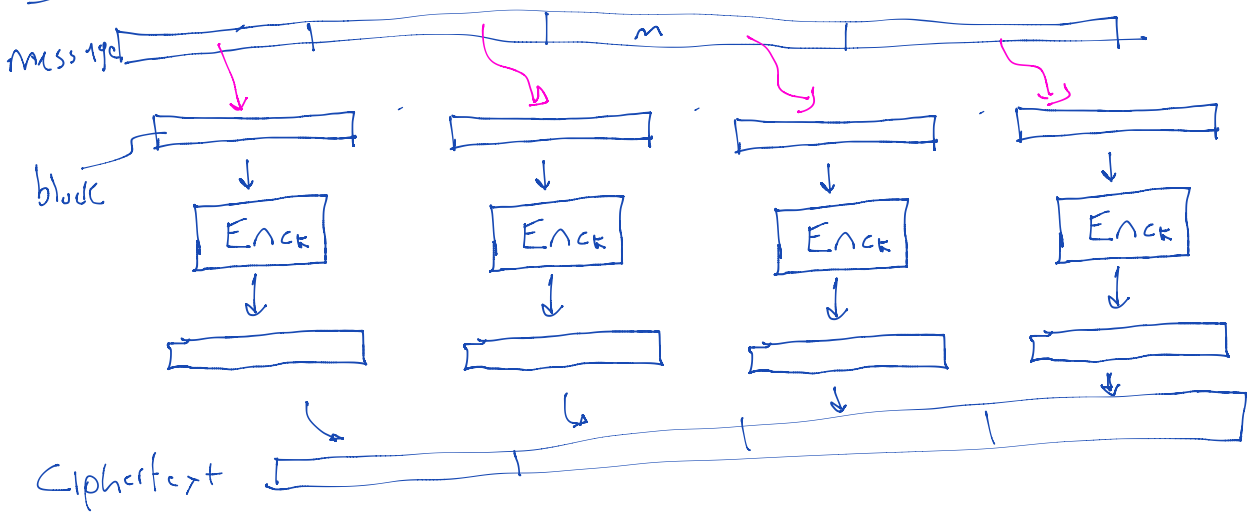
Def'n Encryption scheme is OTS-secure if no computationally bounded adversary can win the OTS game with probability greater than $1/2 + \varepsilon$

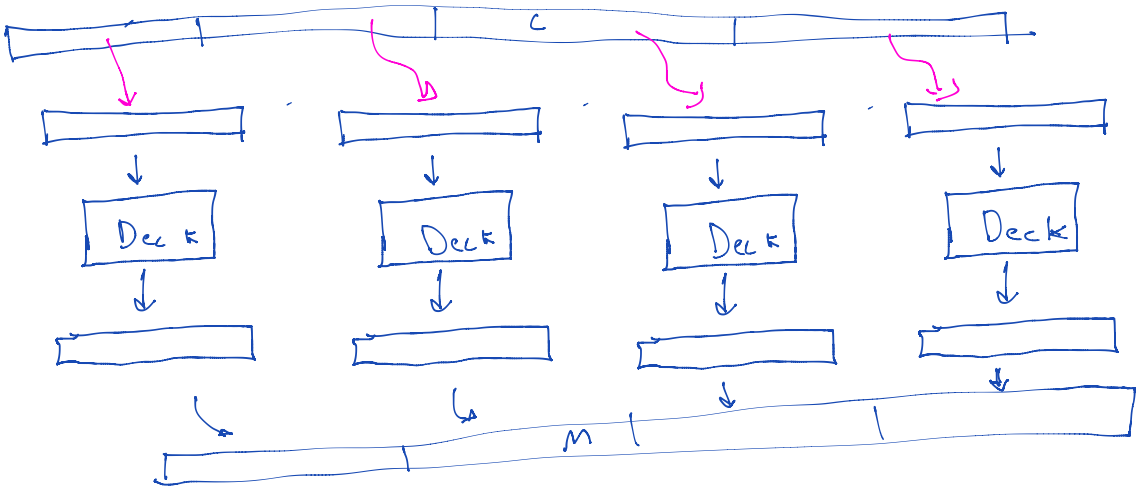
$\varepsilon \rightarrow$ negligible function

↳ if you play the game enough, ε approaches zero.

* Every time we play the game, the oracle chooses a new key → hence "one-time" security.

Mode of Operations: #1 ECB

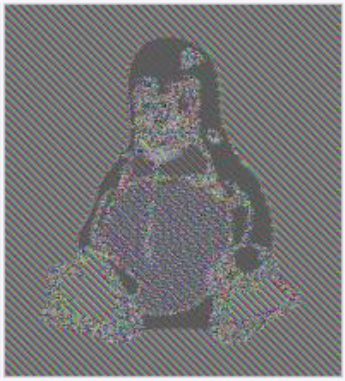




Security of ECB

* Weakness: same input block of message results in same output block.
* Example from wikipedia:

Original image



Encrypted using ECB mode

Modes other than ECB result in pseudo-randomness

The image on the right is how the image might appear encrypted with CBC, CTR or any of the other more secure modes—indistinguishable from random noise. Note that the random appearance of the image on the right does not ensure that the image has been securely encrypted; many kinds of insecure encryption have been developed which would produce output just as "random-looking".

Security Def'n Revisited

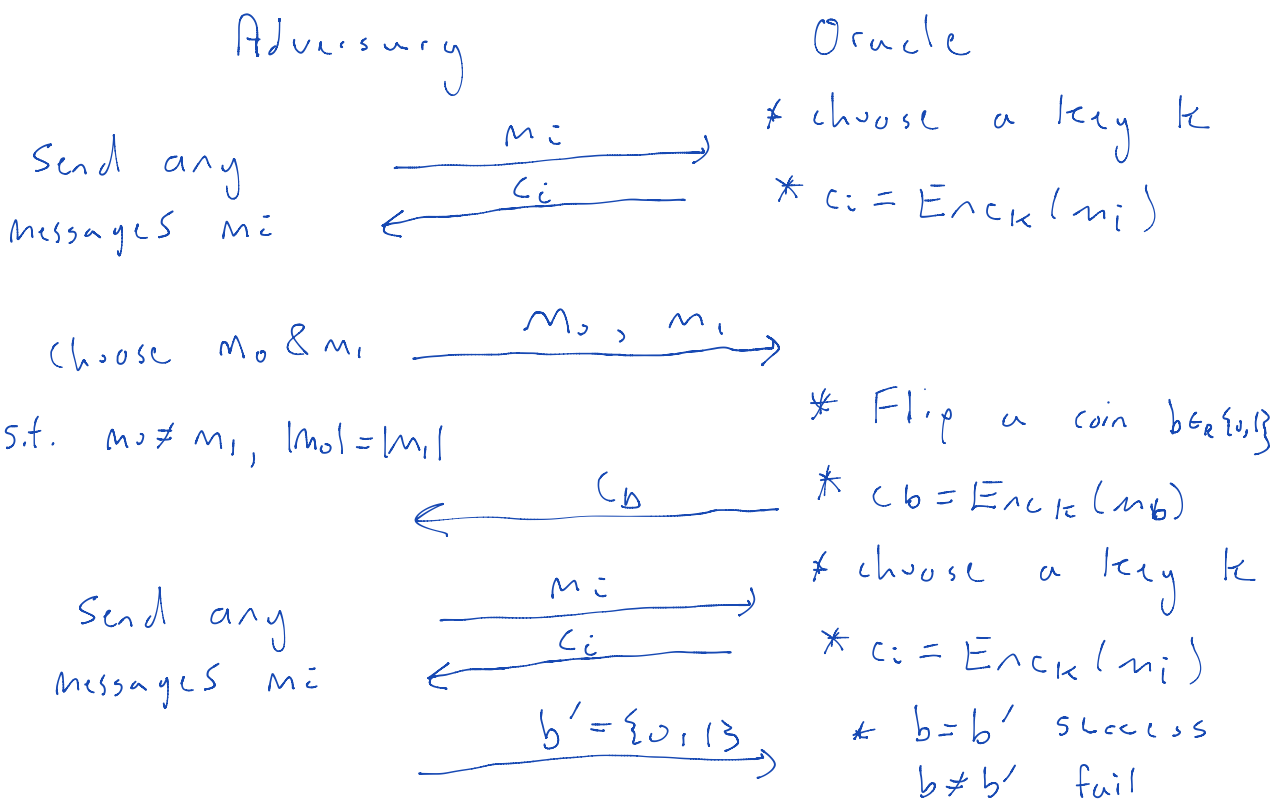
↳ extend the OTS game to include more than one encryption under the same key.

↳ Stronger Def'n: CPA (chosen plaintext attack)

↳ ECB is OTS-secure

↳ ECB is not CPA-secure.

CPA-Game



Fact: ECB is not CPA secure.

Proof: ① Adv can ask for encryption of either in the challenge:

$m_0 = A$

$m_1 = B$

and receive c_b ($b=0$ or $b=1$)

② Adv asks for encryption of A and receives c_A

③ If $c_b = c_A \rightarrow b=0$

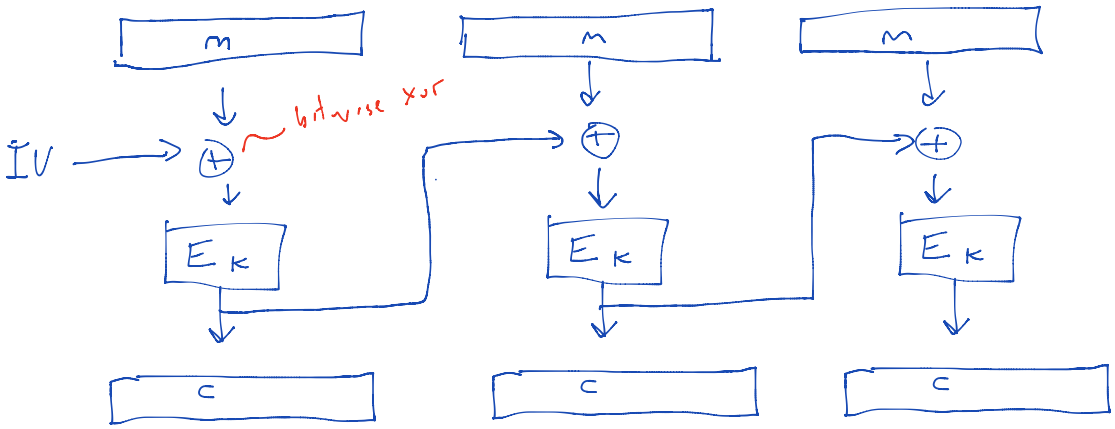
$c_b \neq c_A \rightarrow b=1$

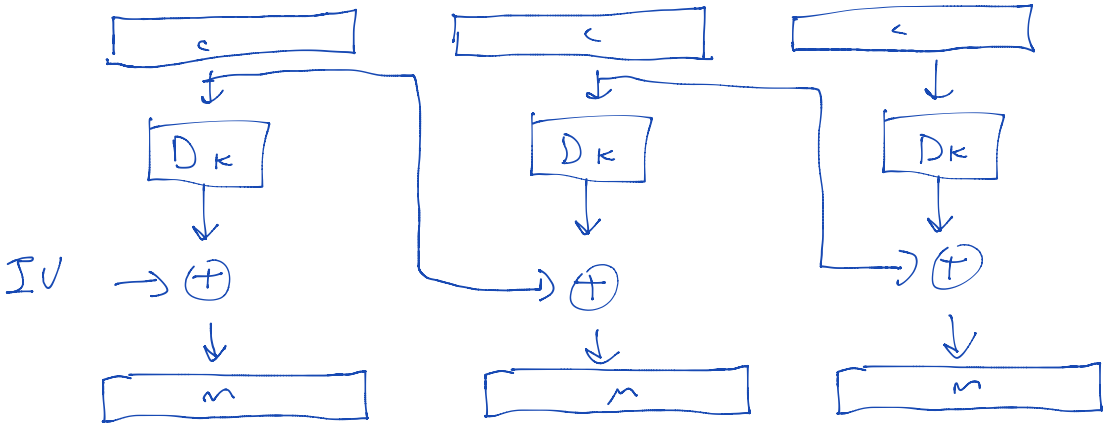
} win 100%

Fact: no deterministic encryption scheme is CPA-secure.

Mode of Operations: #2 CBC

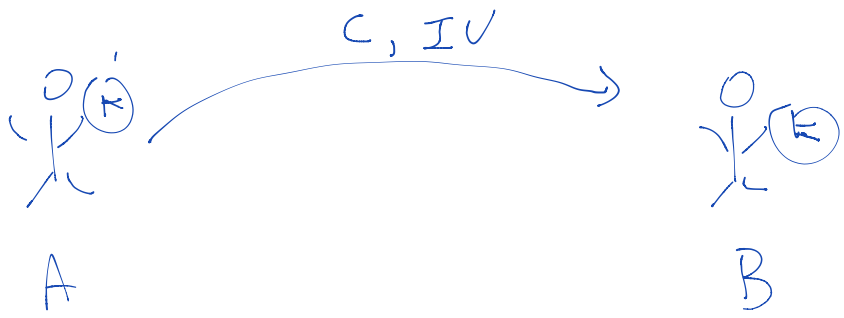
└ cipher block chaining





$$c = E_{nc-CBC_K}(IV, m)$$

$$m = E_{nc-CBC_K}(IV, c)$$



CBC uses initialization vectors (IV) to provide non-determinism.

↳ Fact: CBC is OTS-secure

↳ Fact: CBC is CPA-secure

↳ Lemma: ECB attack doesn't work on CBC-mode because you get a different ciphertext back everytime you ask for an encryption of the same message.

IVs are not secret but they can't be predictable

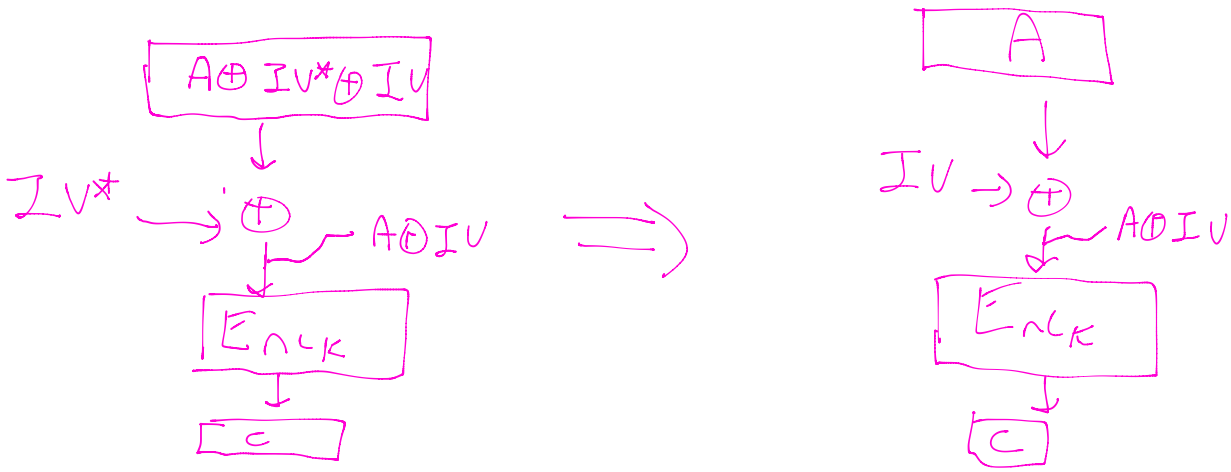
↳ can't know IV before choosing message to encrypt.

Example:

A → O: give me encryption of A

O → A: $\langle IV, C = Enc_K(A) \rangle$

A: know the next IV value will be IV*



$A \to O$: Challenges:

$$m_0 = A \oplus IV \oplus IV^*$$

$$m_1 = B$$

$O \to A$: c_b $\begin{cases} b=0 \to IV, c \\ b=1 \to \tilde{IV}, \tilde{c} \end{cases}$

$\tilde{c}$: not c

Fact: a predictable IV is not CPA-secure.

↳ In practice: BEAST attack on TLS

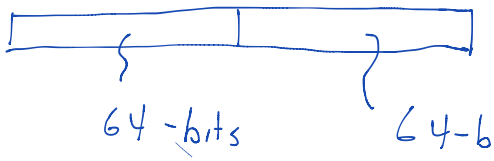
Counter Mode (CTR / CM)

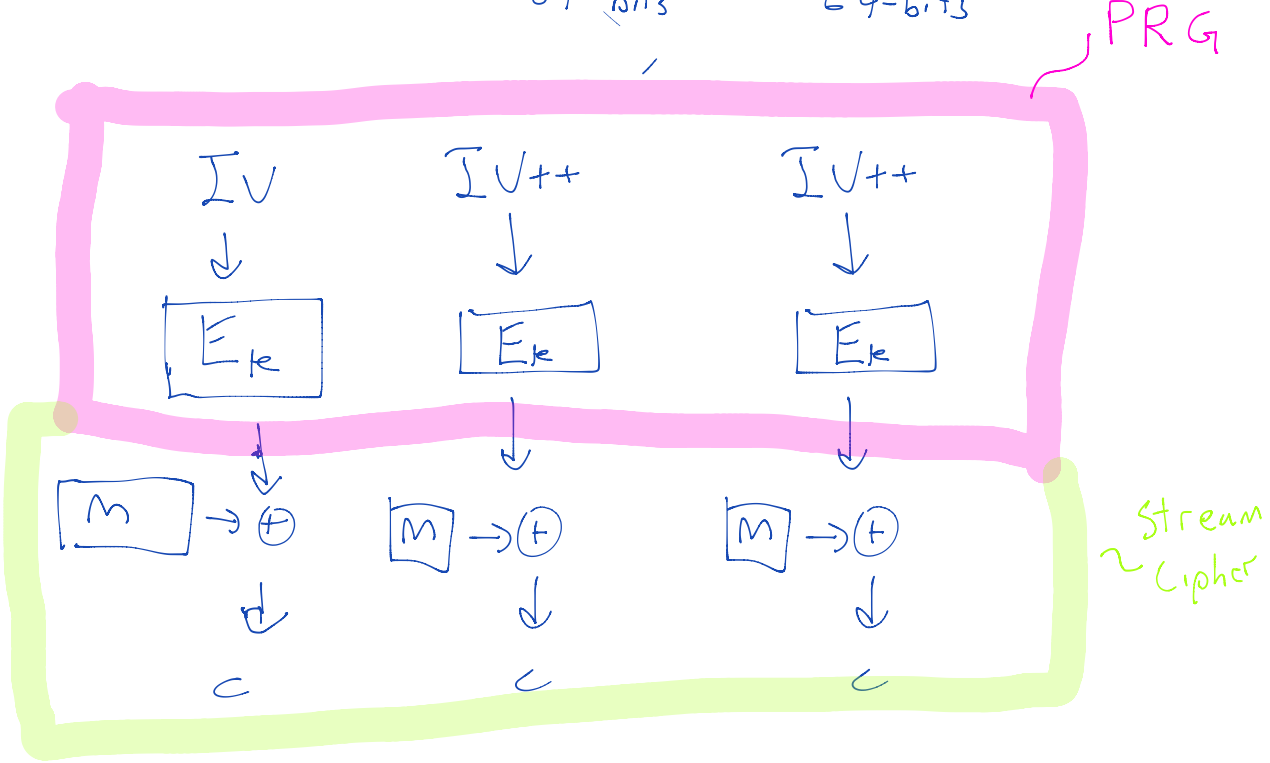
IV = nonce || counter

nonce: random number that is unpredictable.

counter: 0, 1, 2, 3, ...

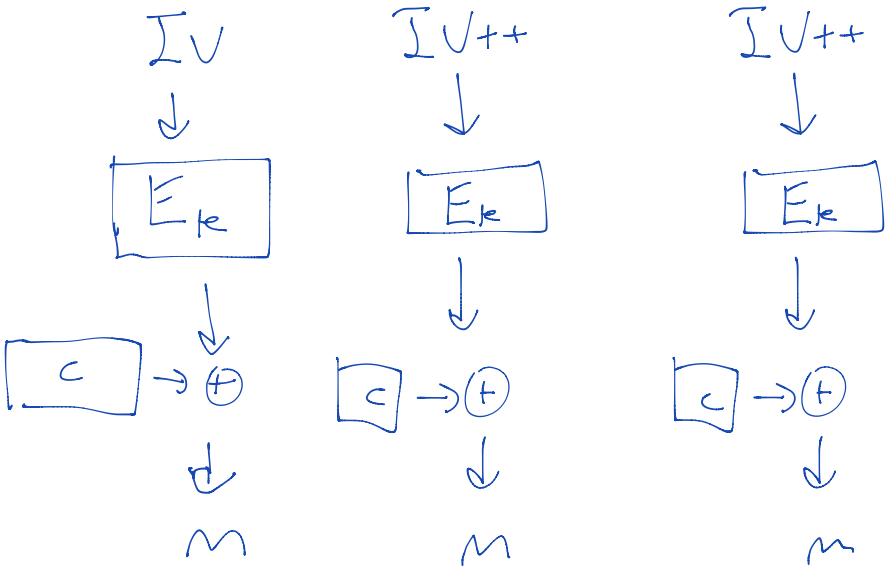
=





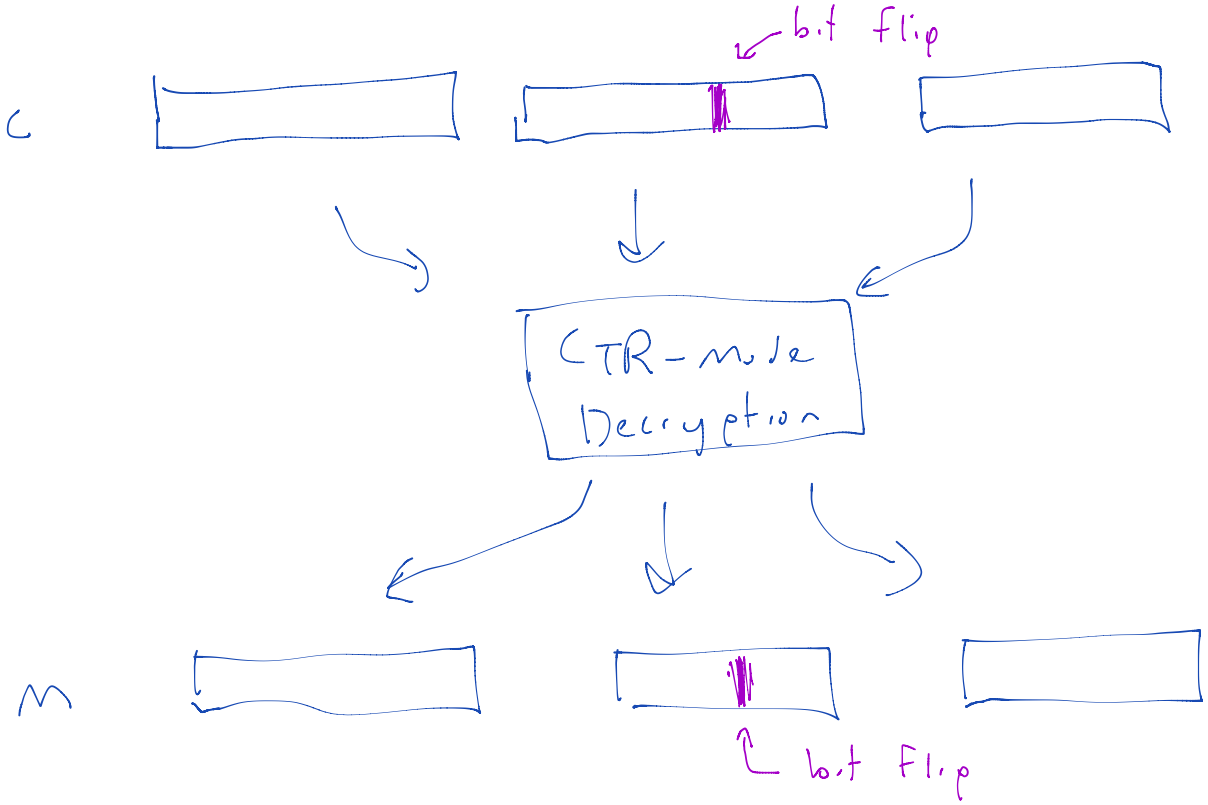
$$\langle c, IV \rangle = Enc\text{-}CTR_k(m, IV)$$

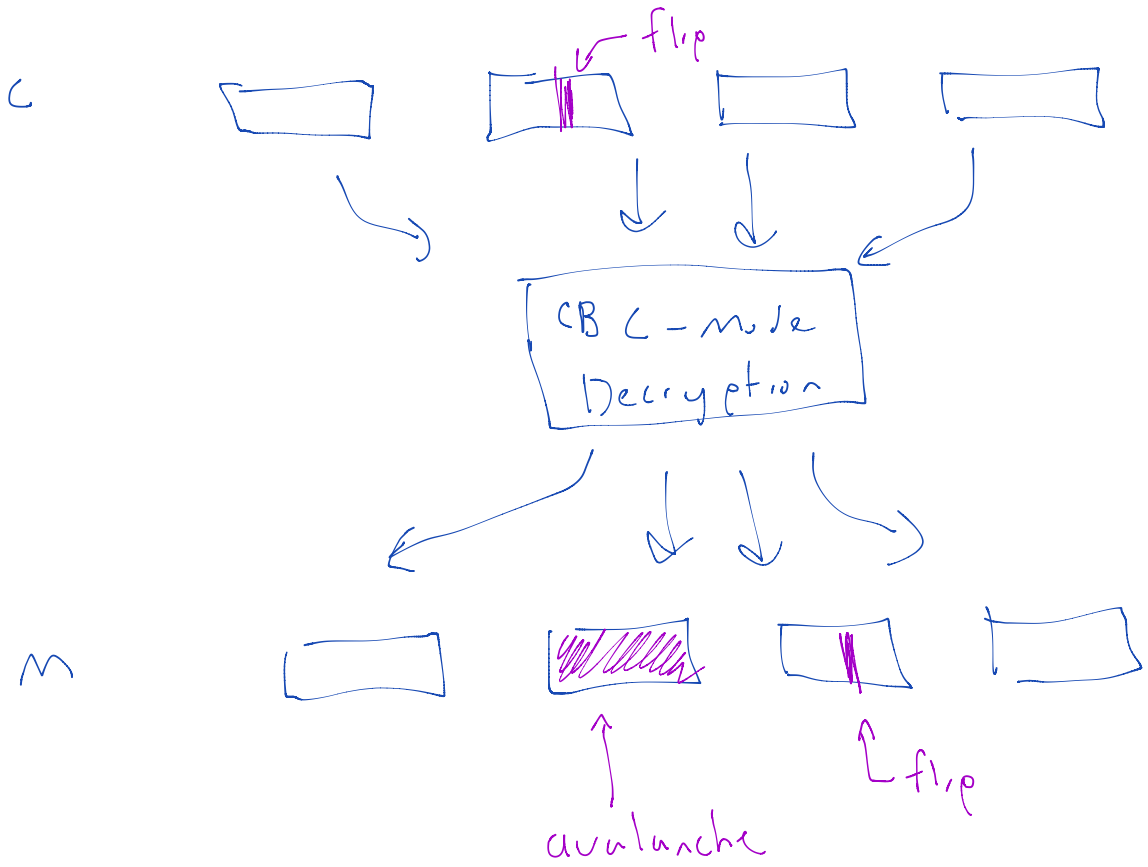
$$m = Dec\text{-}CTR_k(c, IV)$$



* CTR mode
  - ↳ only need encryption function
  - ↳ CTR mode is a stream cipher
  - ↳ CTR allows Enc operations to be pre-computed before you see ciphertext (but know IV)
  - ↳ CBC can parallelize decryption

* Malleability





Road Map:

* CBC & CTR are CPA-secure
* CBC & CTR are malleable

↓

change ciphertext and plaintext will change in a predictable way.

* We need a higher level of security that excludes malleability

↳ CCA-security.

* We need a better mode of operation that is CCA-secure.