Padding Oracle Attack

LINUX
D A Y
I T A L I A

28 ottobre

Una piccola presentazione

0xfederico

Scuola: ITIS Belluzzi-Fioravanti

Titolo di studio: triennale Unimore Scienze Informatica

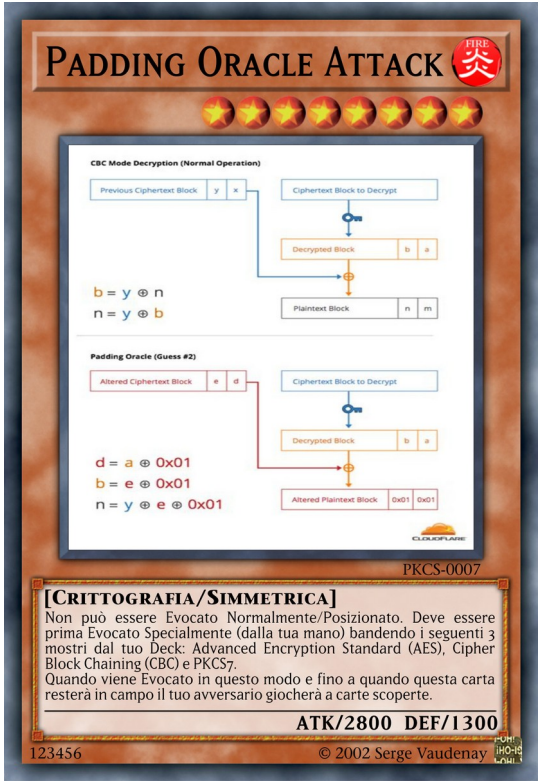
Cosa faccio ora: magistrale Unimore Scienze Informatica

Blog: 0xhacks.gitlab.io

Riccardo Fasolo

Scuola: ITIS Belluzzi-Fioravanti

Attacco padding che?



Inventore: Serge Vaudenay (2002)

Istanze dell'attacco:

- contro protocolli: SSL(2003), IPSEC(2007)
- contro web frameworks: JavaServer Faces, Ruby on Rails(2010), ASP.NET(2010)
- contro hardware security devices: security keys, hardware security modules (HSMs)(2012)
- contro software famosi: Steam(2016)

Oggi è una minaccia? da TLS 1.3 con l'utilizzo di AEAD (Authenticated Encryption with Additional Data) **l'attacco è mitigato**.
Prima di tale versione c'erano ancora varianti dell'attacco funzionanti come: Poodle (Padding Oracle On Downgraded Legacy Encryption) in SSL3.0 (2014) e Lucky Thirteen (2016).

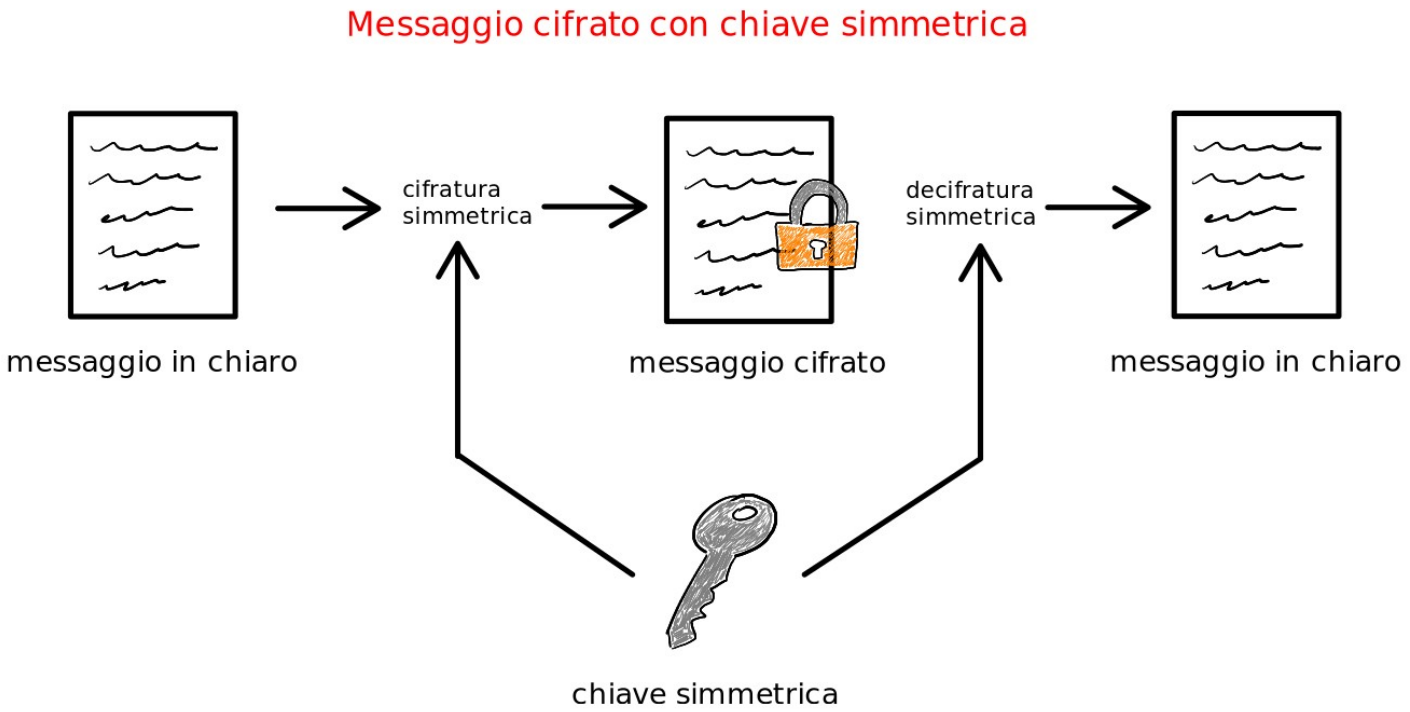
Fonti:

- *https://en.wikipedia.org/wiki/Padding_oracle_attack#Attacks_using_padding_oracles*
- *https://www.techtarget.com/searchsecurity/answer/How-concerned-should-I-be-about-a-padding-oracle-attack*

Troppa roba, facciamo un passo indietro

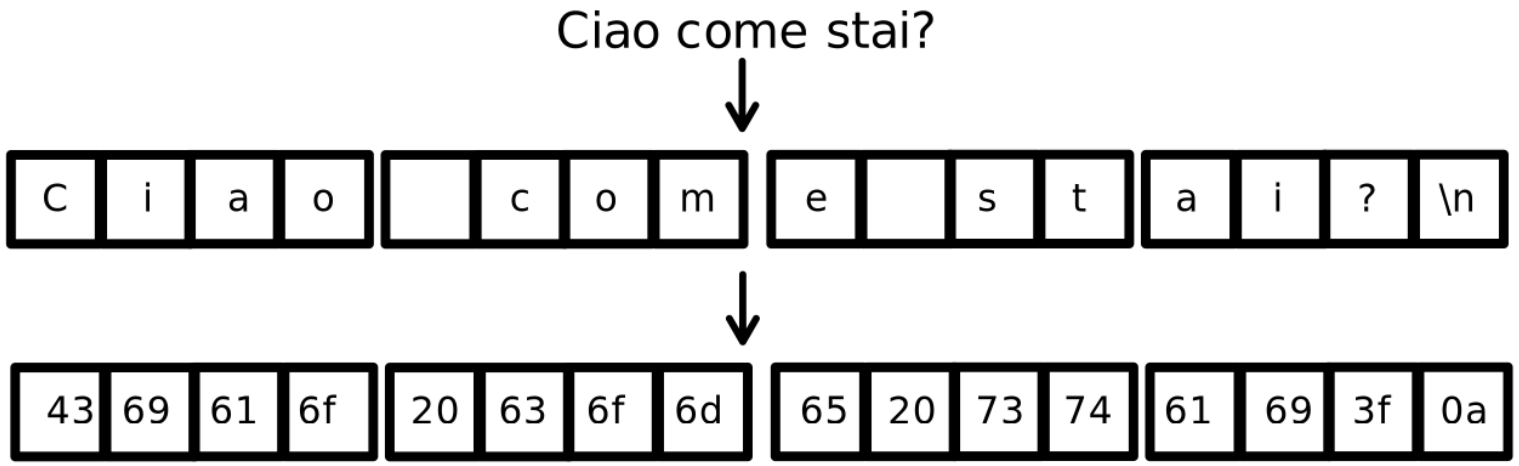
When you try to teach your parents how to use technology

Crittografia simmetrica

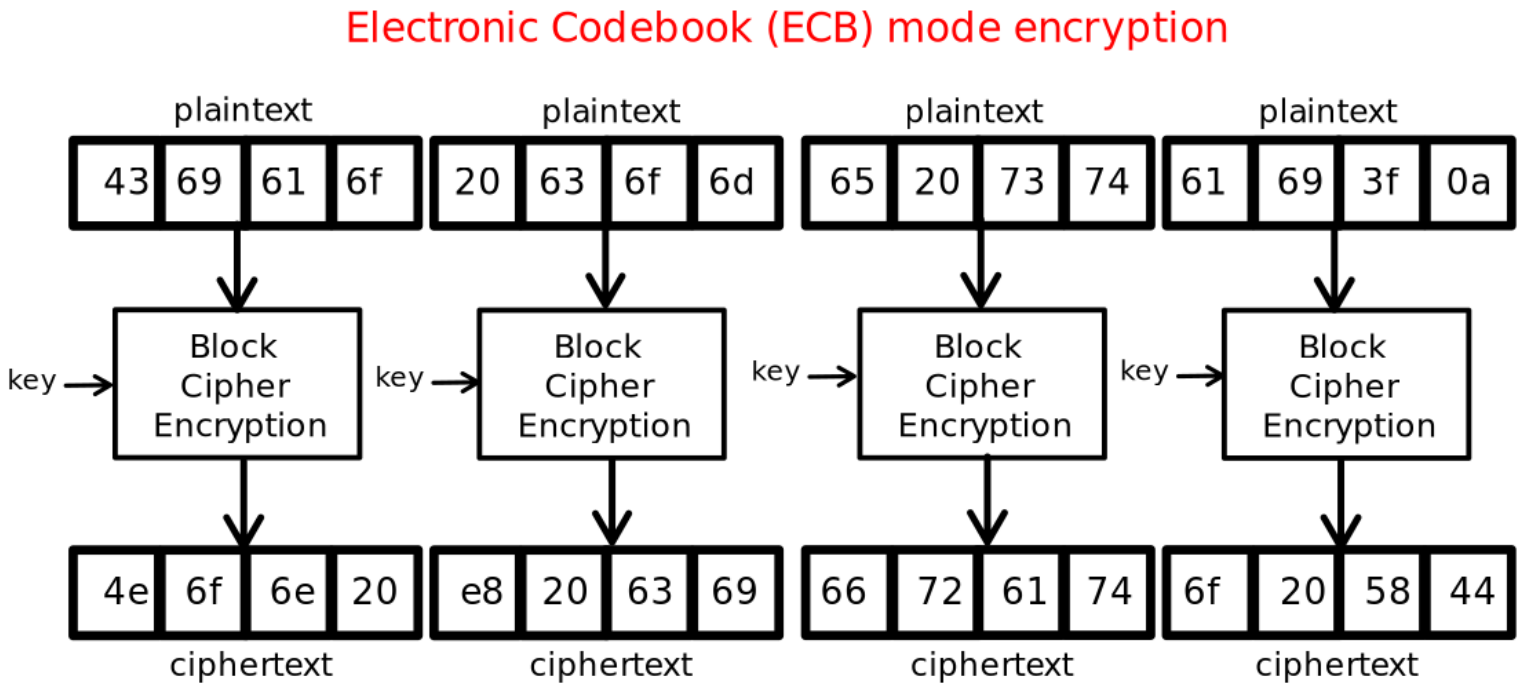


Crittografia simmetrica a blocchi e mode of operation (1/3)

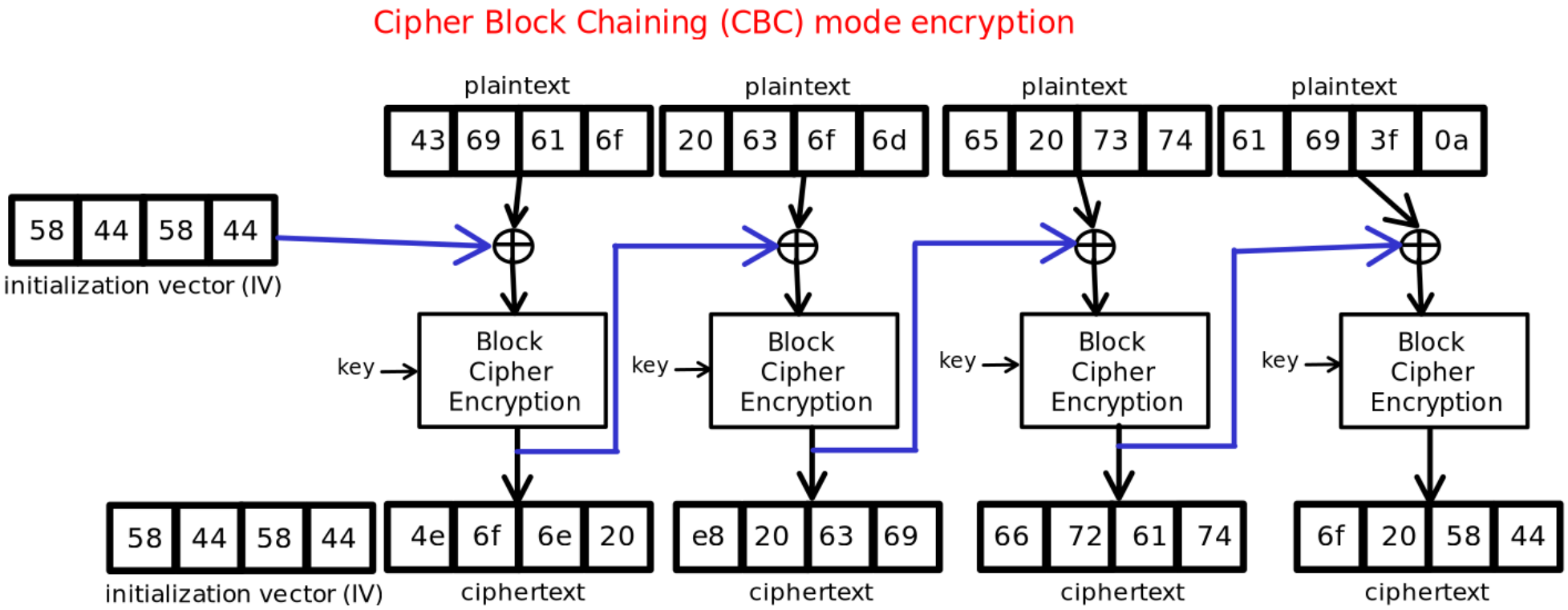
Suddivisione di un messaggio in blocchi da 4 byte



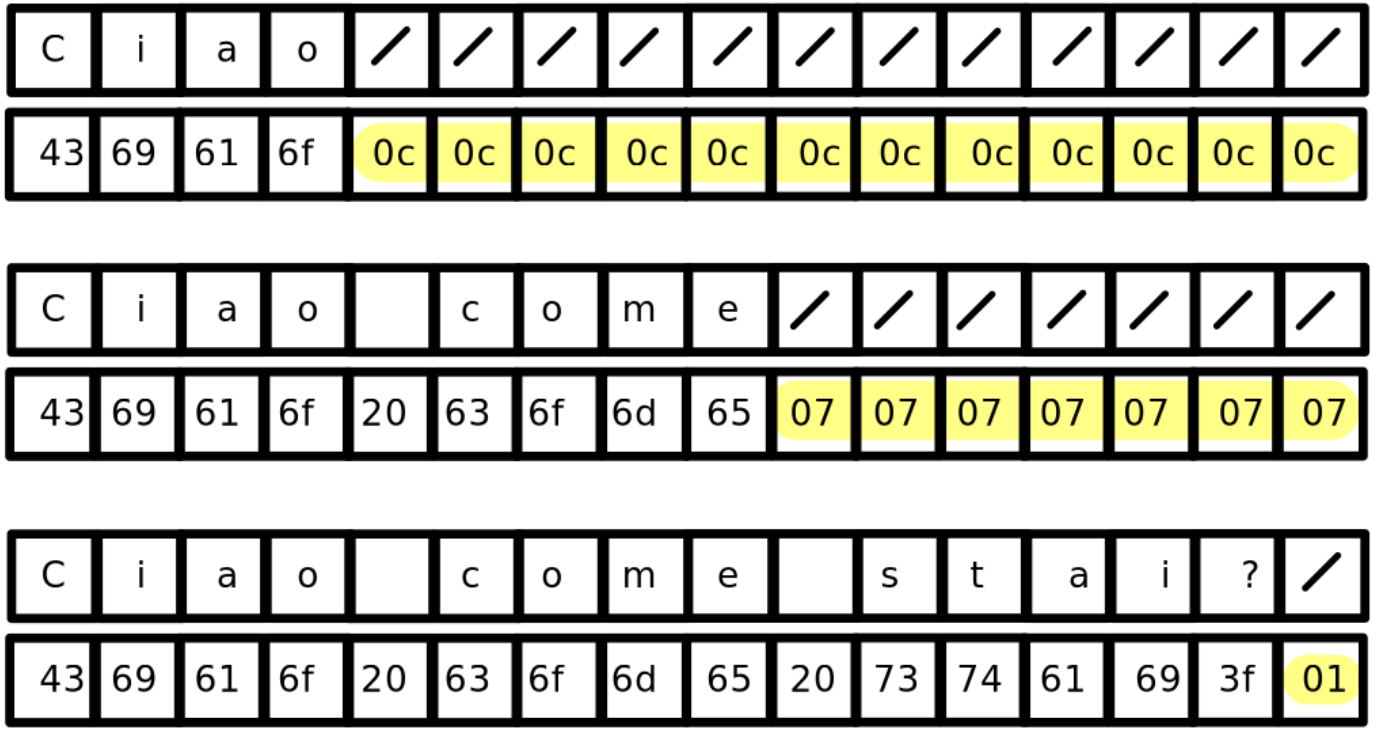
Crittografia simmetrica a blocchi e mode of operation (2/3)



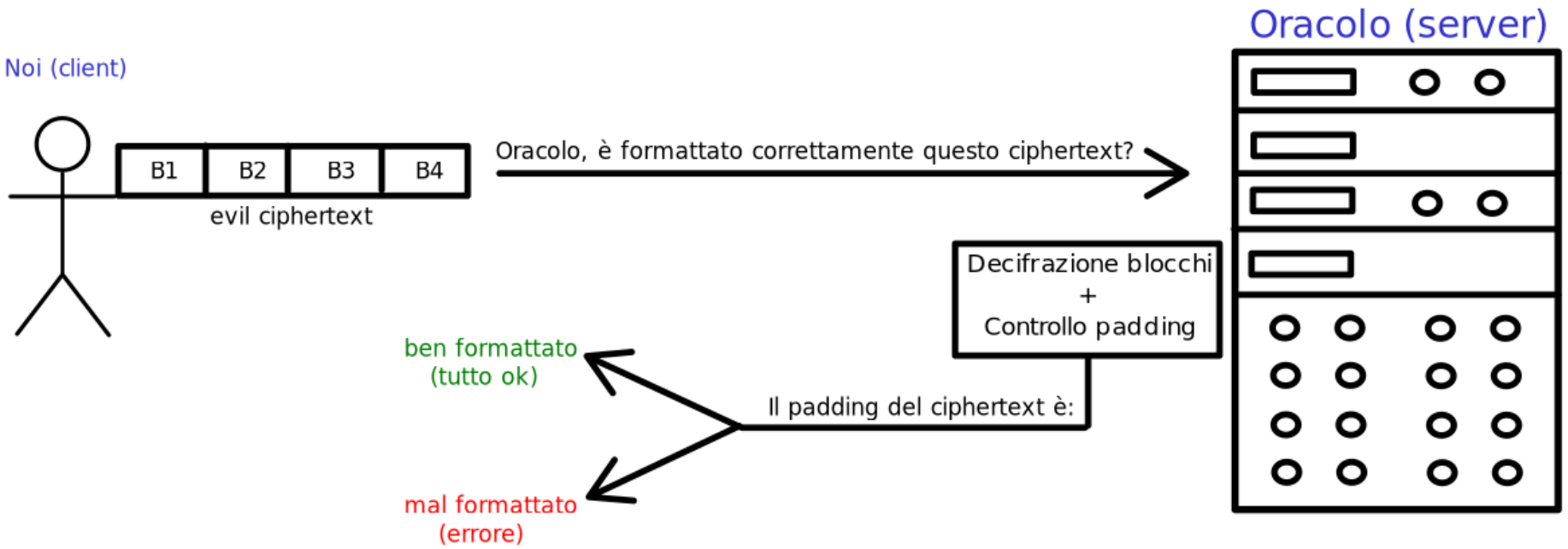
Crittografia simmetrica a blocchi e mode of operation (3/3)



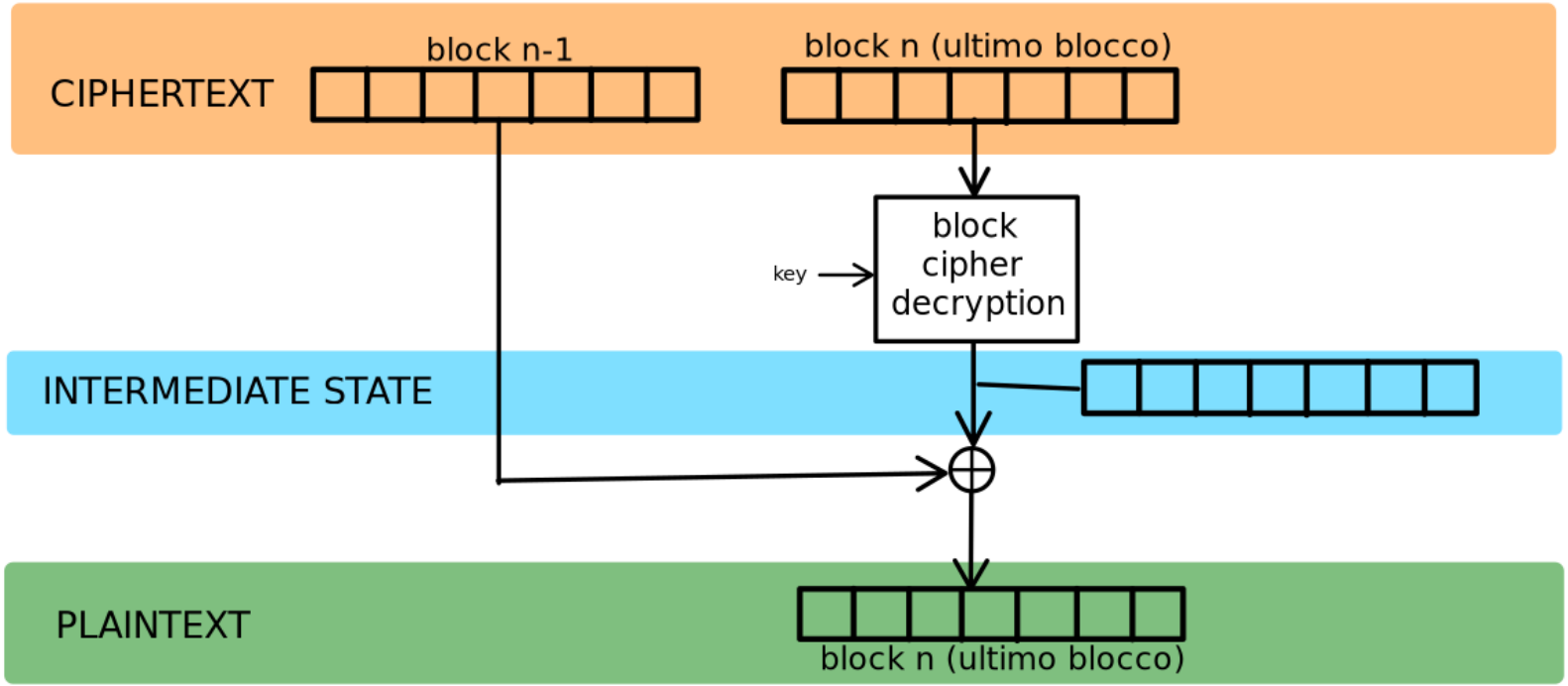
Padding (PKCS#7)



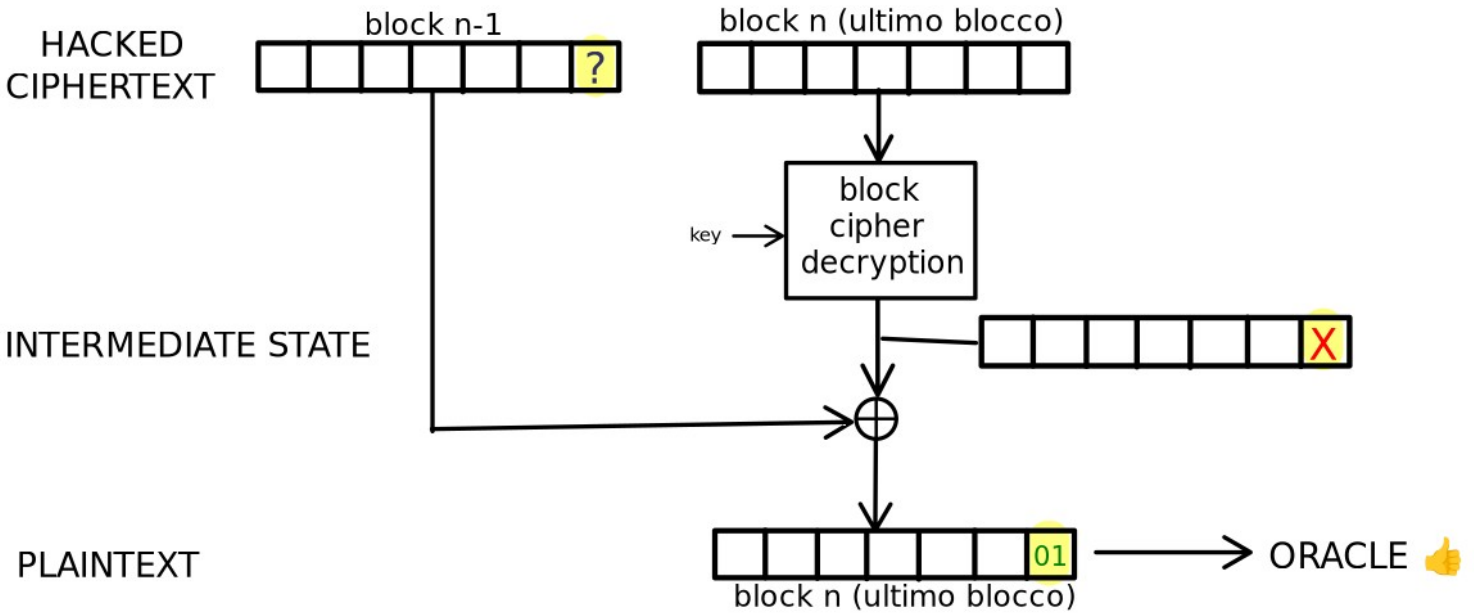
Ci manca solo un ultimo ingrediente: un oracolo!



AES + CBC + PKCS#7 + Oracle = *a lot of fun* (1/7)

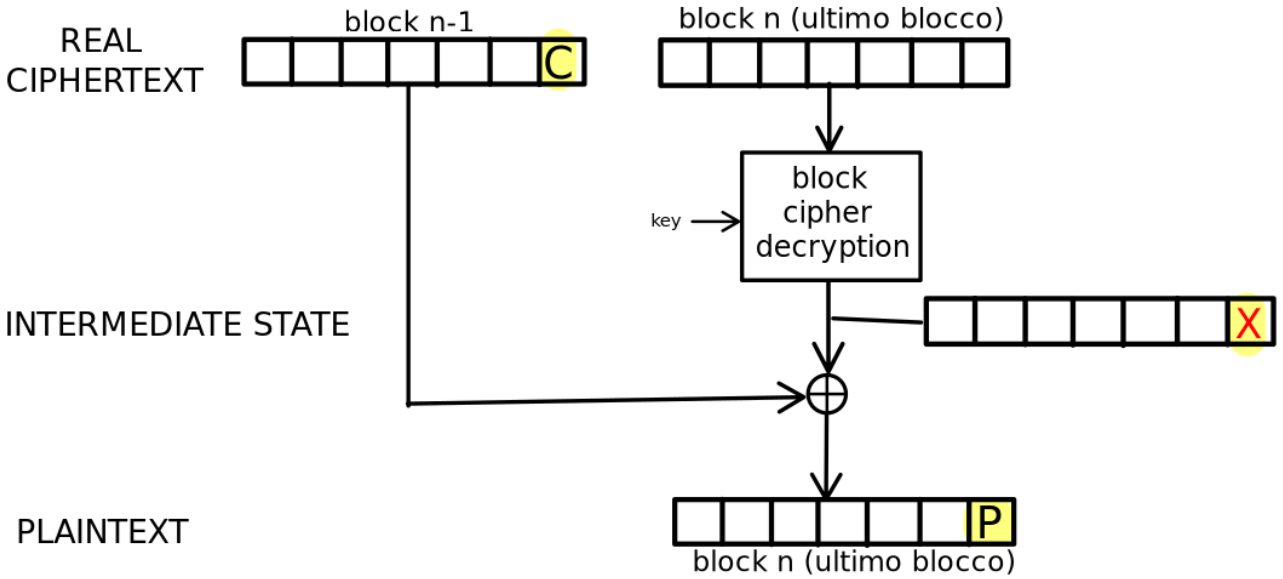


AES + CBC + PKCS#7 + Oracle = *a lot of fun* (2/7)



$? \oplus X = 01$ allora... $X = ? \oplus 01$

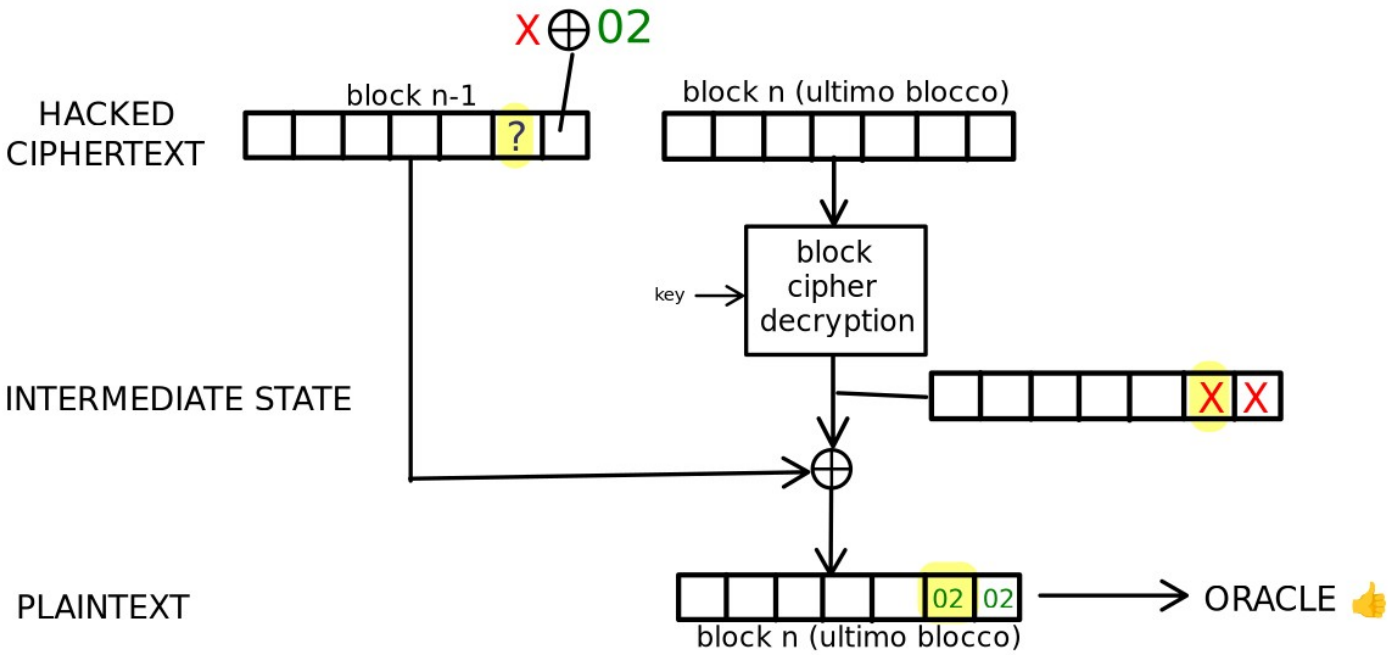
AES + CBC + PKCS#7 + Oracle = *a lot of fun* (3/7)



avendo trovato il valore intermedio possiamo ricavare il plaintext!

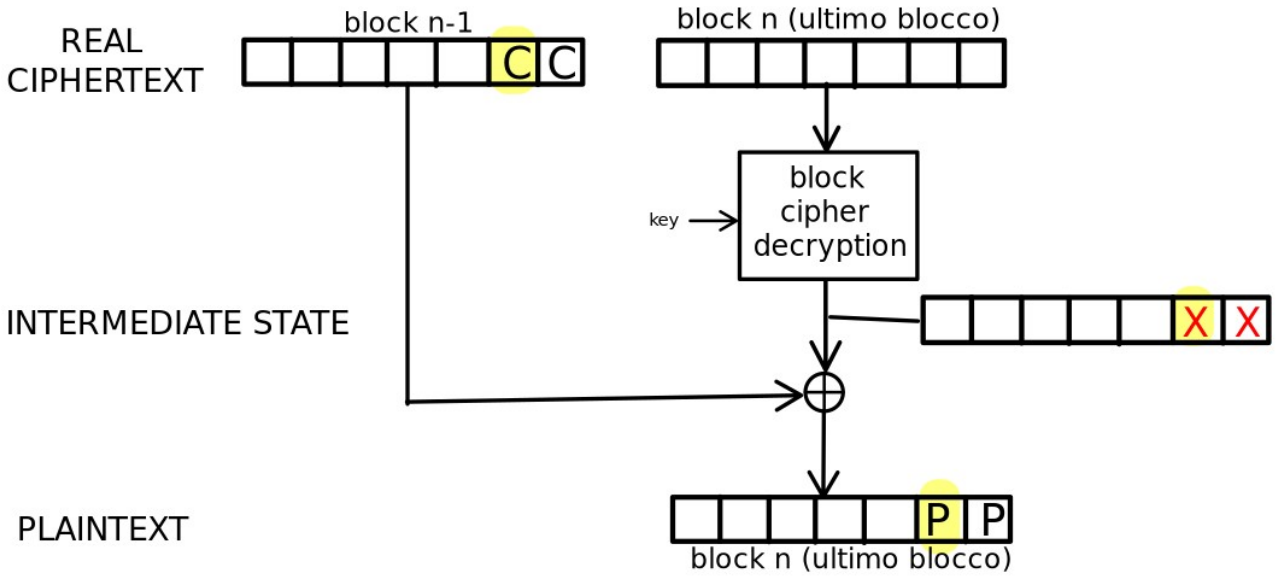
$$C \oplus X = P$$

AES + CBC + PKCS#7 + Oracle = *a lot of fun* (4/7)



$? \oplus X = 02$ allora... $X = ? \oplus 02$

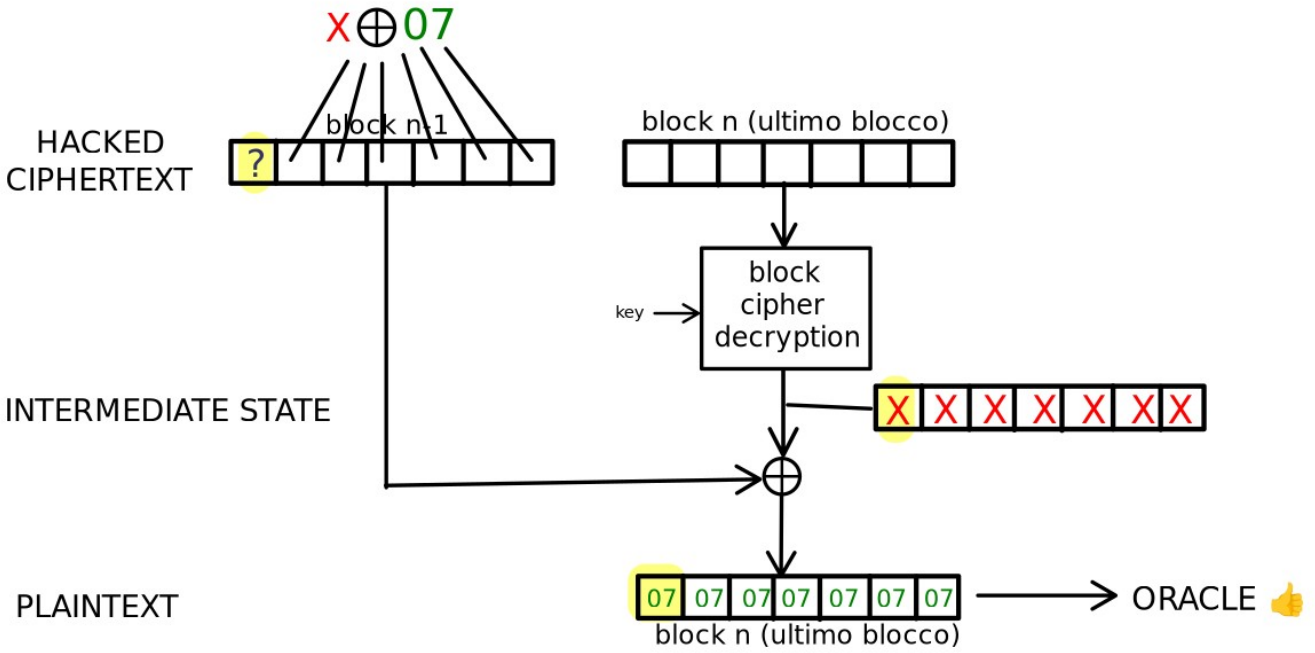
AES + CBC + PKCS#7 + Oracle = *a lot of fun* (5/7)



avendo trovato il valore intermedio possiamo ricavare il plaintext!

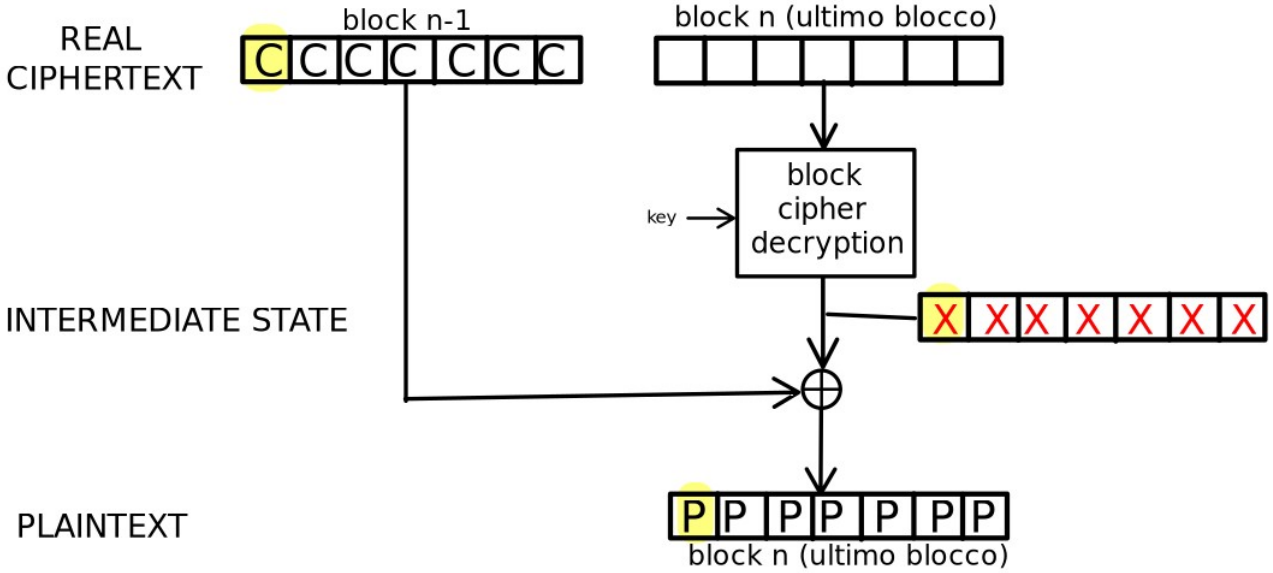
$$C \oplus X = P$$

AES + CBC + PKCS#7 + Oracle = *a lot of fun* (6/7)



?⊕X = 07 allora... X = ?⊕07

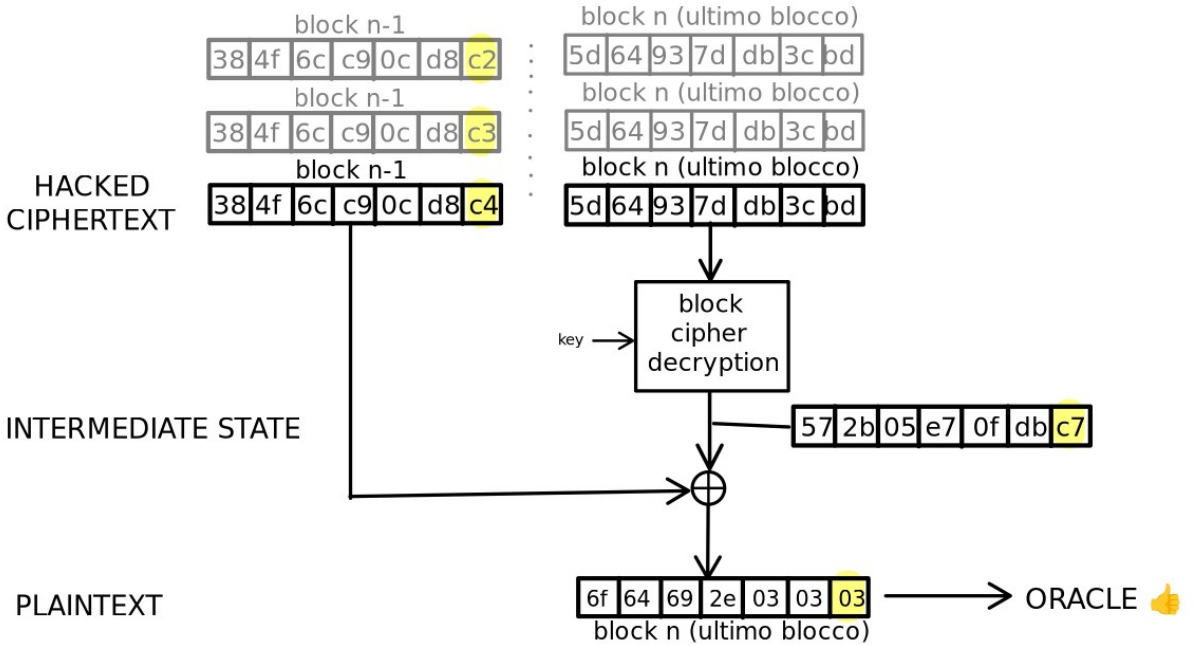
AES + CBC + PKCS#7 + Oracle = *a lot of fun* (7/7)



avendo trovato il valore intermedio possiamo ricavare il plaintext!

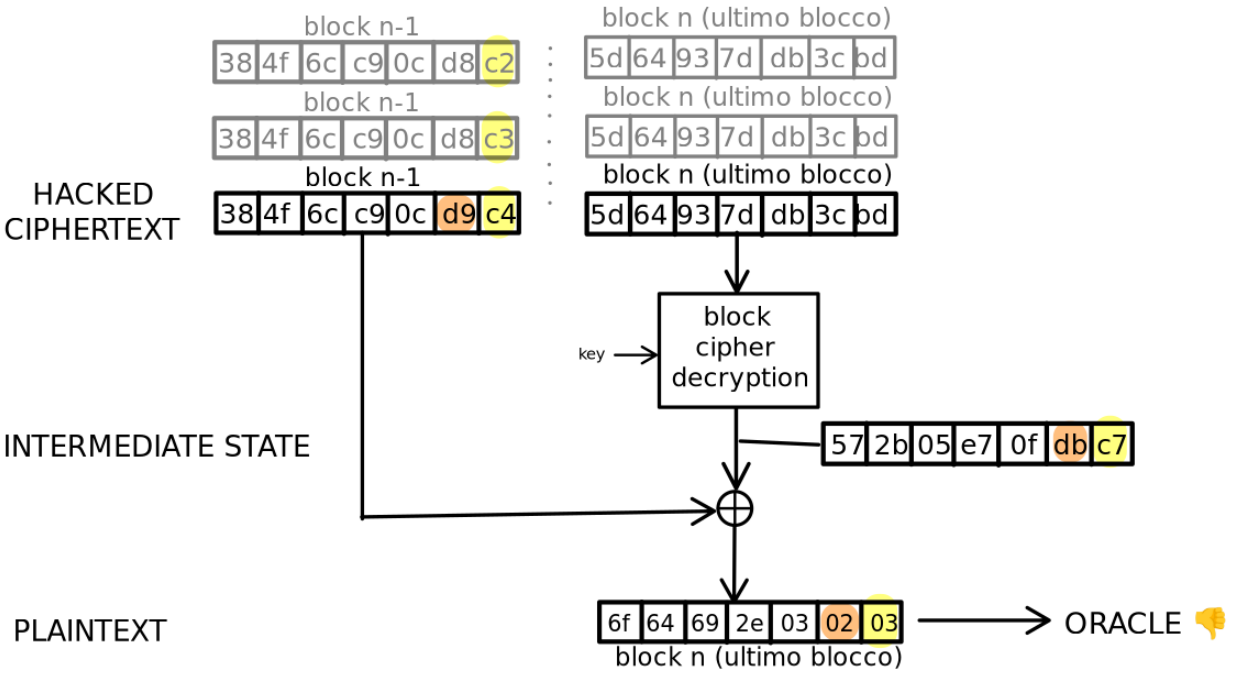
$$C \oplus X = P$$

Test della correttezza (1/2)



Aspetta un attimo, ma non dovremmo trovare un 01 in prima posizione? 🤔

Test della correttezza (2/2)



Che succede se modifichiamo il byte precedente del nostro ciphertext?
Se l'oracolo ci dice che il padding non va bene abbiamo trovato un falso positivo!

Belle le slide! Ma il codice?

“Talk is cheap. Show me the code.”

Linus Torvalds

Slides e codice disponibili su gitlab: https://gitlab.com/0xfederico/cryptography-attacks/

Per chi è arrivato fino a qui...

Grazie per l’attenzione,
adesso passiamo al codice!