

1)

(a) Fare delle sostituzioni preserva la frequenza delle lettere?

TRUE

(b) Fare delle trasposizioni preserva la frequenza delle lettere?

TRUE

(c) Qual'è il numero di trasposizioni possibili su una stringa IN^I ?

$N!$

(d) Qual'è il numero di sostituzioni possibili su una stringa IN^I ?

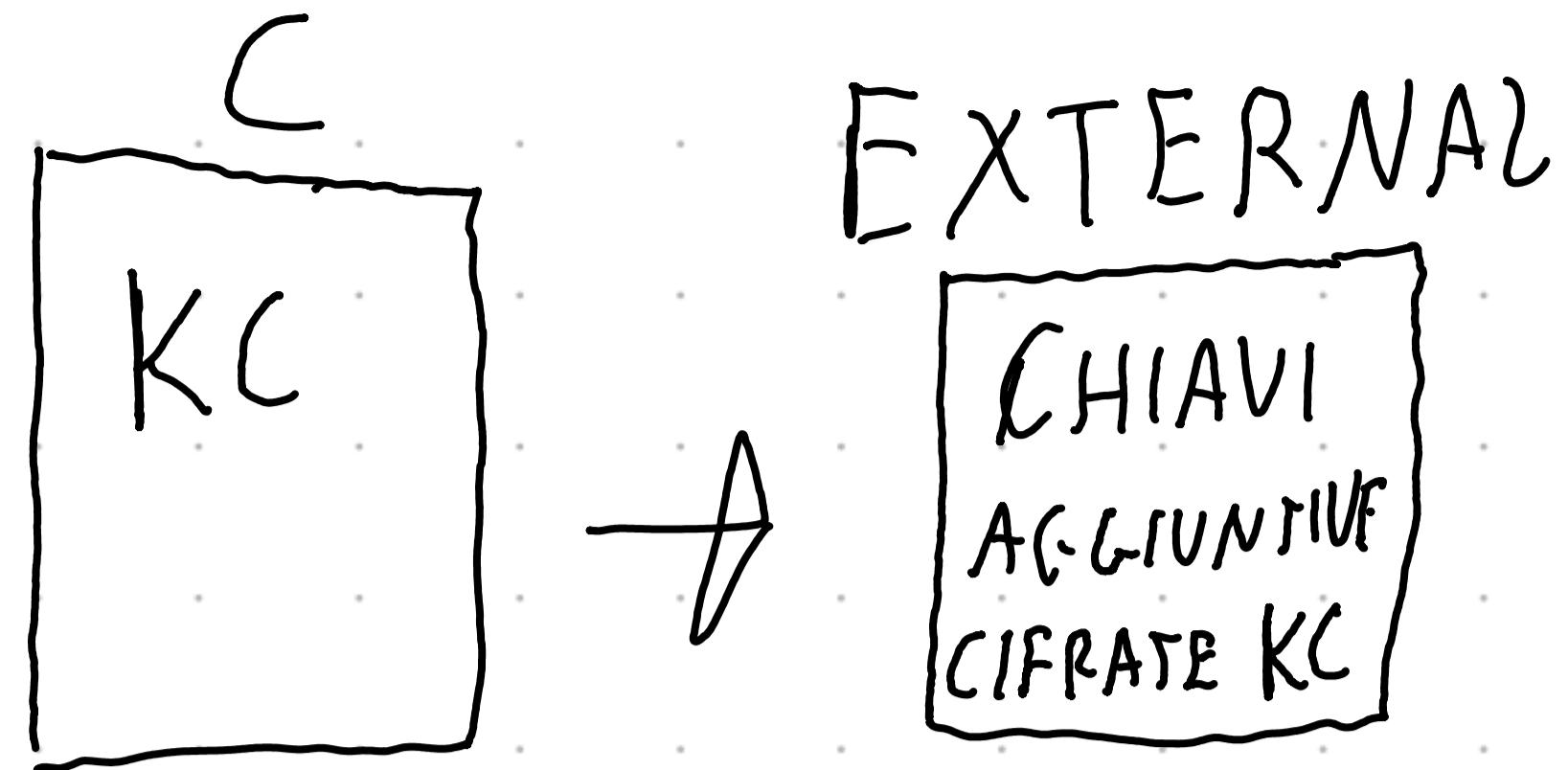
K^N dove K è la cardinalità dell'alfabeto



$$\text{SE BIT} = 2^N$$

$$\text{SE ALFABETO LATINO} = 22^N$$

2)



Ogni chiave K viene associata ad un vettore PK il quale indica come deve essere usata K , e C dove usarla nel modo descritto nel vettore

Avvertimenti: C non può esportare $\{ek\}_{Kc}, PK\}$ perché un attaccante potrebbe modificarlo

→ Si prova ad integrare PK dentro la cifratura fatta da C

(1)

Non è adeguato perché se un attaccante modifica il nostro PK , di conseguenza la nostra K cambierà quando facciamo il reverse dello XOR.

3)

(a)

Si, il timestamp è necessario per impedire attacchi di tipo replay, infatti senza timestamp, un attaccante potrebbe rutilizzare il vecchio messaggio per rianticarsi.

(b)

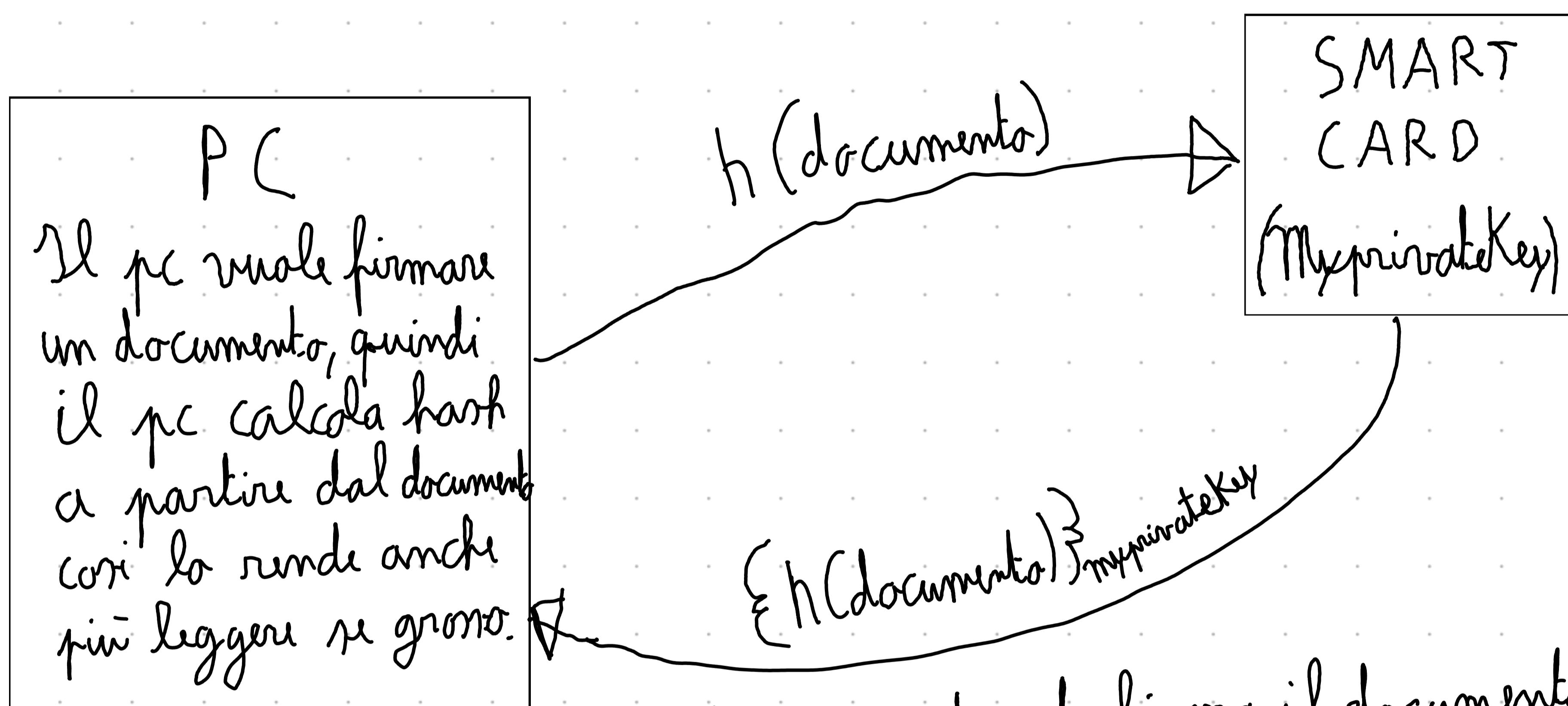
In questo caso il nonce viene utilizzato in maniera errata perché nel punto 3 "C" non ritorna il nonce a "S".

In più il nonce è inutile se viene già utilizzata il timestamp: il nonce potrebbe tornare utile in accoppiata col timestamp solo se ci fossero richieste multiple.

4)

(a)

Visto che la smartcard non può contenere tanti dati, essa viene utilizzata solo per contenere la chiave privata usata per criptare la firma, quindi calcola ed invia al reader solo la stringa di bit che rappresenta la firma.



La smartcard firma il documento, per farlo critta hash ricevuto con la sua chiave privata salvata dentro la SMARTCARD, la quale per nessun motivo deve abbandonare la SMARTCARD.

La SMARTCARD firma e non convalida la firma digitale, per convalidare la firma digitale serve la chiave pubblica

