

ATTACCHI : scelta di e

$$e \neq \frac{\phi(n)}{2} + 1$$

$$e \neq \frac{\phi(n)}{k} + 1$$

se  $k | p-1$  e  $k | q-1$

altrimenti

$$m^e \bmod n = m$$

( quando  $\text{MCD}(m, n) = 1$  )

## ATTACCHI CON LO STESSO VALORE di e

- almeno e utenti che hanno scelto lo stesso valore di e
- gli e utenti ricevono lo stesso messaggio m

U<sub>1</sub>

$$C_1 = m^e \bmod n_1$$

U<sub>2</sub>

$$C_2 = m^e \bmod n_2$$

⋮

U<sub>e</sub>

$$C_e = m^e \bmod n_e$$

Sia

$$n = \prod_{i=1}^e n_i$$

Th. Cinese del resto,  $\exists !$  t.c.

$$m' \equiv m^e \bmod n$$

$m'$  si può calcolare in tempo polinomiale

1 poteri

$\forall i, j \quad 1 \leq i < j \leq e$

$\text{GCD}(n_i, n_j) = 1$

$\forall i \quad 1 \leq i \leq e$

$m < n_i$

$$m^l \equiv m^e \pmod{n}$$

$$m^l = m^l \pmod{n} = m^e \pmod{n} = \cancel{m^e} \quad m^e < n$$

↑  
 $m^l < n$

$\forall i \quad m < n_i^c$

$$m^e = \underbrace{m \cdot m \cdot m \cdots m}_{e \text{ volte}} < n_1 \cdot n_2 \cdot \cdots \cdot n_e = n$$

$\Rightarrow$  la congruenza diventa

$$m^l \equiv m^e \quad \Rightarrow \quad m = \sqrt[e]{m^l}$$

 Padding 

## Attacco "Common Modulus"

Due utenti  $U_1, U_2$  con chiavi pubbliche

- $\langle e_1, n \rangle$        $\langle e_2, n \rangle$
- $\text{MCD}(e_1, e_2) = 1$

$$\begin{array}{ll} U_1 & C_1 = m^{e_1} \bmod n \\ U_2 & C_2 = m^{e_2} \bmod n \end{array}$$

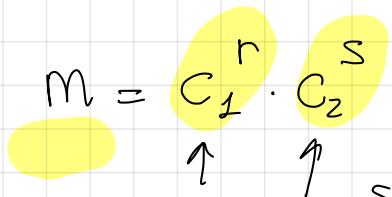
$\text{MCD}(e_1, e_2) = 1 \Rightarrow \exists r, s \text{ t.c. } re_1 + se_2 = 1$   
↳ e si calcolano i coefficienti  
polinomiale con EG

w.l.o.g

$$r < 0 \quad \text{e} \quad s > 0$$

$$\begin{aligned}
 m &= m^1 = m^{re_1 + se_2} = m^{re_1 + se_2} \bmod n = \\
 &= (m^{re_1} \bmod n)(m^{se_2} \bmod n) \bmod n \\
 &= (\underbrace{m^{e_1} \bmod n}_c)^r (\underbrace{m^{e_2} \bmod n}_c)^s \bmod n
 \end{aligned}$$

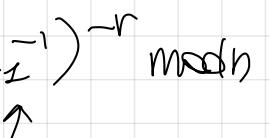
$$m = C_1^r \cdot C_2^s \bmod n$$



$s > 0$ ,  $C_2^s$  si calcola in tempi  
 polinomiale con l'algoritmo di  
 esponente totale veloce

$$C_1^r \bmod n$$

$$C_1^r \bmod n = (C_1^{-1})^{-r} \bmod n$$



$$-r > 0$$

$\Rightarrow \text{se } \text{HCD}(C_1, n) = 1$

$\Rightarrow$  we calculate  $C_1^{-1} \pmod{n}$  (traps poly.)  
on EE

$\Rightarrow$  calculate  $(C_1^{-1})^{-r}$  on alg. exp. voice  
(traps poly)

in fine there M come

$$M = (C_1^{-1})^{-r} (C_2)^s \pmod{n}$$

# Attacchi a tempo (DH, RSA)

Si basano sul tempo di esecuzione dell'algoritmo di decifrazione

## IDEA

determinare d analizzando il tempo impiegato per decifrare

- Quando viene eseguita l'esponenziazione modulare, si esegue una moltiplicazione ad ogni iterazione, più un'ulteriore moltiplicazione modulare per ciascun bit uguale a 1 in d
- **Rimedio:** aggiungere ritardo causale per confondere l'attaccante