Accordo su chiavi

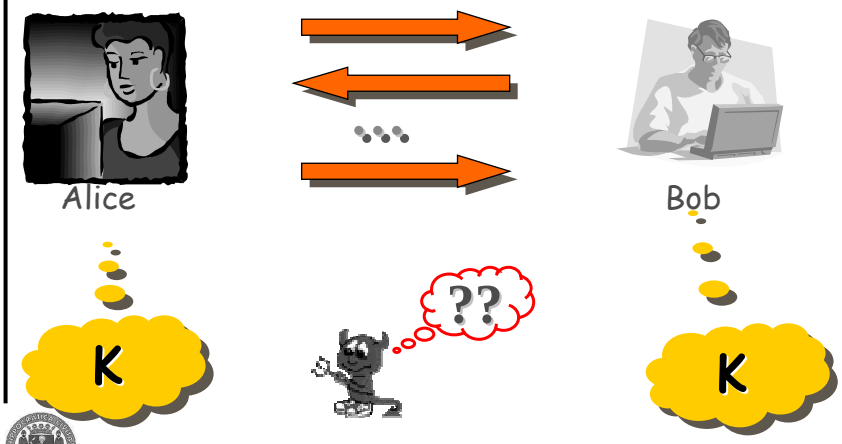
Barbara Masucci

Dipartimento di Informatica ed Applicazioni
Università di Salerno

masucci@dia.unisa.it

http://www.dia.unisa.it/professori/masucci

Accordo su una chiave



Accordo su chiavi

Vedremo due schemi:

- Diffie-Hellman
  - Basato sull'intrattabilità del problema del logaritmo discreto
- Puzzle di Merkle
  - Non basato su alcuna assunzione computazionale

Diffie-Hellman [1976]

primo p, generatore g di Z_p^*

Alice

Bob

Generatori di Z_p^*

g è generatore di Z_p^* se $\{g^i | 1 \le i \le p-1\} = Z_p^*$

Esempio:

g = 2 è un generatore di Z_{11}^*

$2^{10} = 1024 = 1 \bmod 11$
$2^1 = 2 \bmod 11$
$2^8 = 256 = 3 \bmod 11$
$2^2 = 4 \bmod 11$
$2^4 = 16 = 5 \bmod 11$
$2^9 = 512 = 6 \bmod 11$
$2^7 = 128 = 7 \bmod 11$
$2^3 = 8 \bmod 11$
$2^6 = 64 = 9 \bmod 11$
$2^5 = 32 = 10 \bmod 11$

Potenze in Z_{19}^*

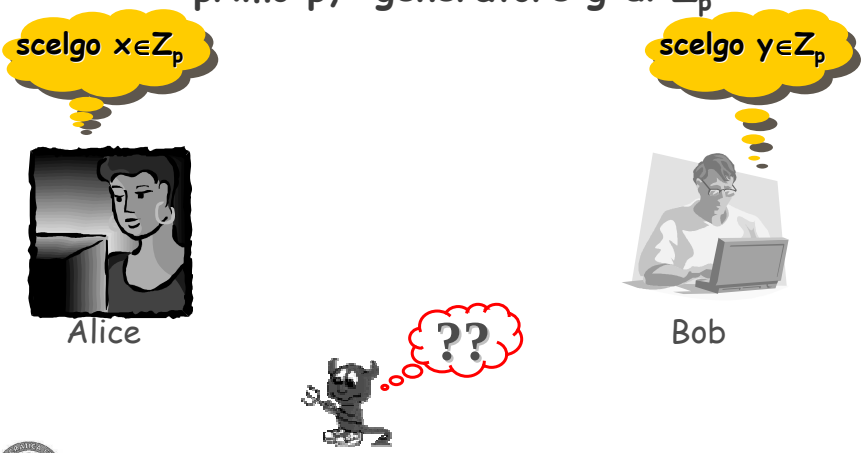
a	a^2	a^3	a^4	a^5	a^6	a^7	a^8	a^9	a^{10}	a^{11}	a^{12}	a^{13}	a^{14}	a^{15}	a^{16}	a^{17}	a^{18}
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	4	8	16	13	7	14	9	18	17	15	11	3	6	12	5	10	1
3	9	8	5	15	7	2	6	18	16	10	11	14	4	12	17	13	1
4	16	7	9	17	11	6	5	1	4	16	7	9	17	11	6	5	1
5	6	11	17	9	7	16	4	1	5	6	11	17	9	7	16	4	1
6	17	7	4	5	11	9	16	1	6	17	7	4	5	11	9	16	1
7	11	1	7	11	1	7	11	1	7	11	1	7	11	1	7	11	1
8	7	18	11	12	1	8	7	18	11	12	1	8	7	18	11	12	1
9	5	7	6	16	11	4	17	1	9	5	7	6	16	11	4	17	1
10	5	12	6	3	11	15	17	18	9	14	7	13	16	8	4	2	1
11	7	1	11	7	1	11	7	1	11	7	1	11	7	1	11	7	1
12	11	18	7	8	1	12	11	18	7	8	1	12	11	18	7	8	1
13	17	12	4	14	11	10	16	18	6	2	7	15	5	8	9	3	1
14	6	8	17	10	7	3	4	18	5	13	11	2	9	12	16	15	1
15	16	12	9	2	11	13	5	18	4	3	7	10	17	8	6	14	1
16	9	11	5	4	7	17	6	1	16	9	11	5	4	7	17	6	1
17	4	11	16	6	7	5	9	1	17	4	11	16	6	7	5	9	1
18	1	18	1	18	1	18	1	18	1	18	1	18	1	18	1	18	1

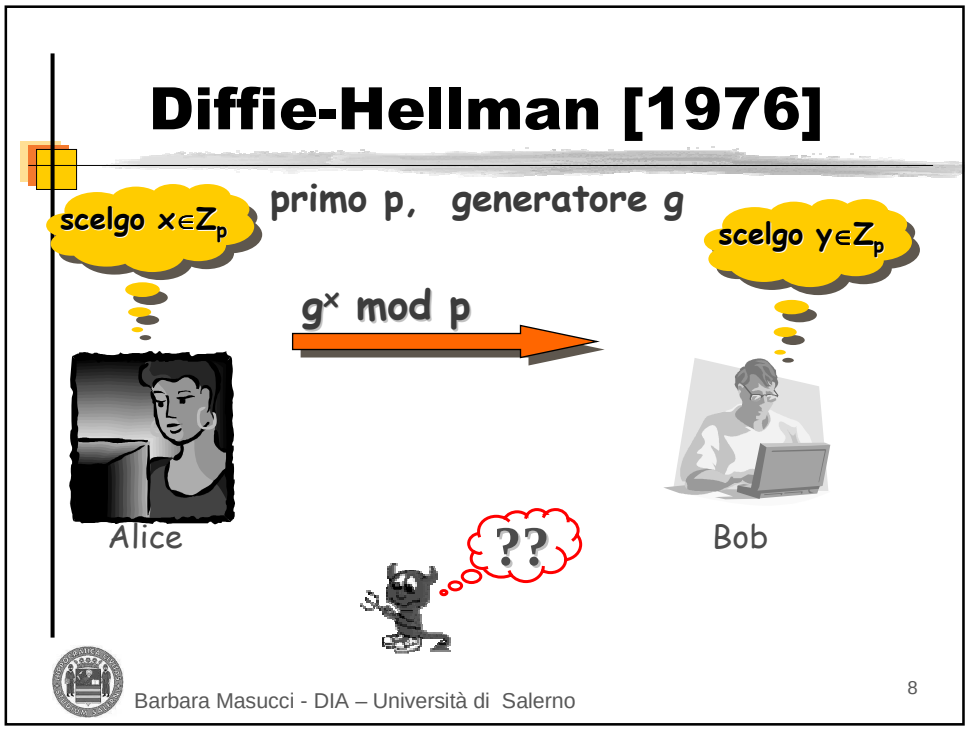
Generatori di Z_n^*

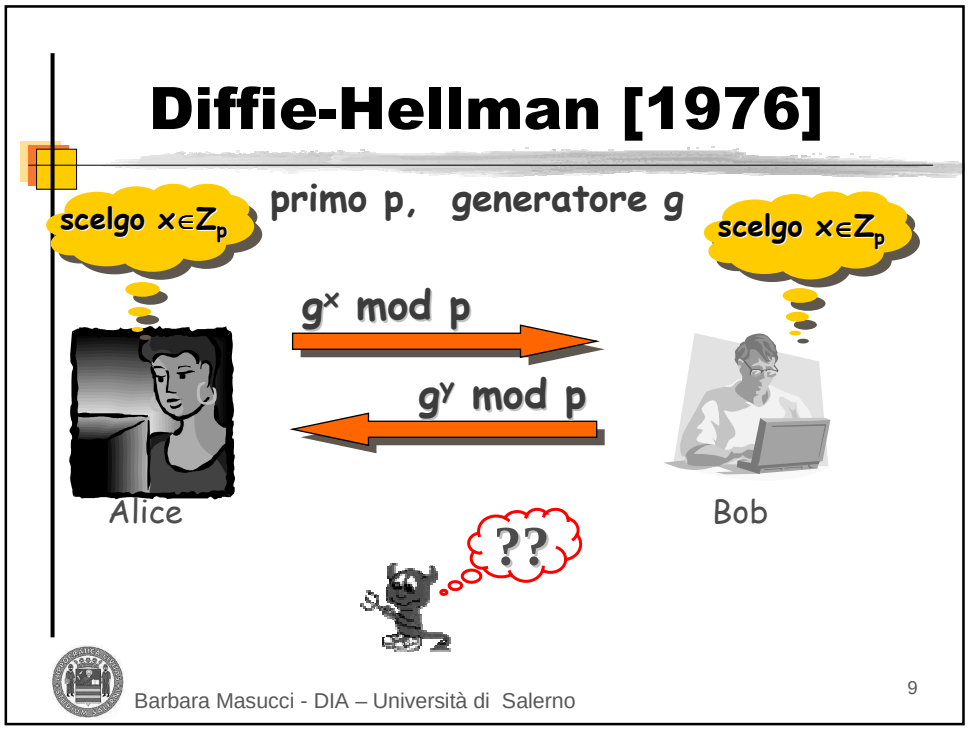
- Z_n^* ha un generatore $\Leftrightarrow$ $n = 2,4,p^k,2p^k$, con p primo e $k \geq 1$
  - Se p è primo, allora Z_p^* ha un generatore
- Il numero di generatori di Z_n^* è $\phi(\phi(n))$
  - Se p è primo, il numero di generatori di Z_p^* è $\phi(p-1)$

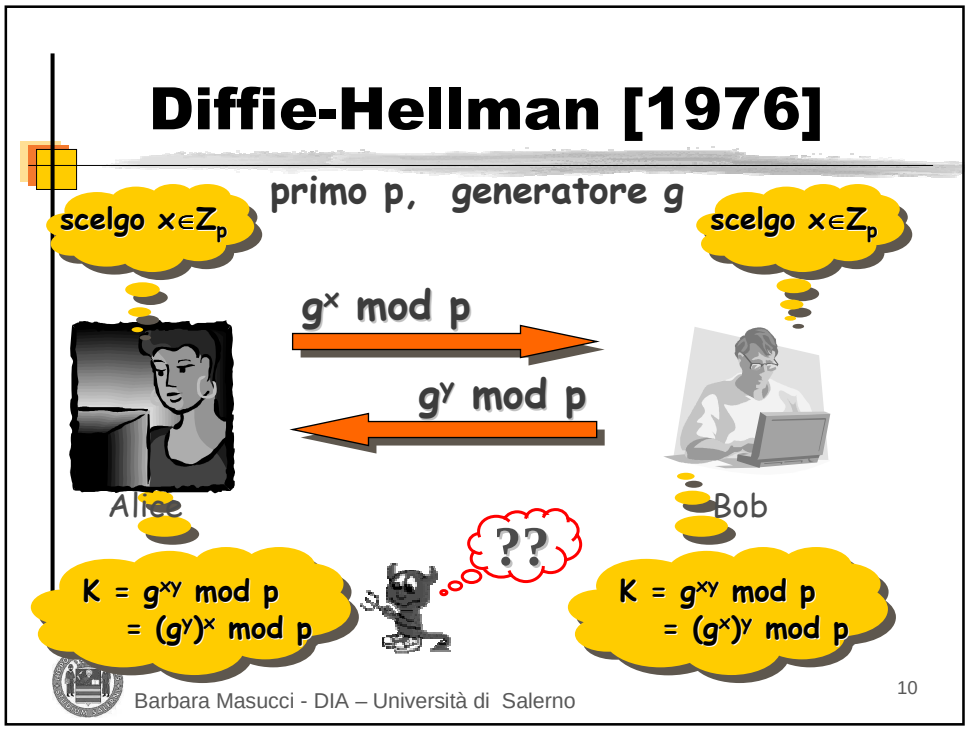
Diffie-Hellman [1976]

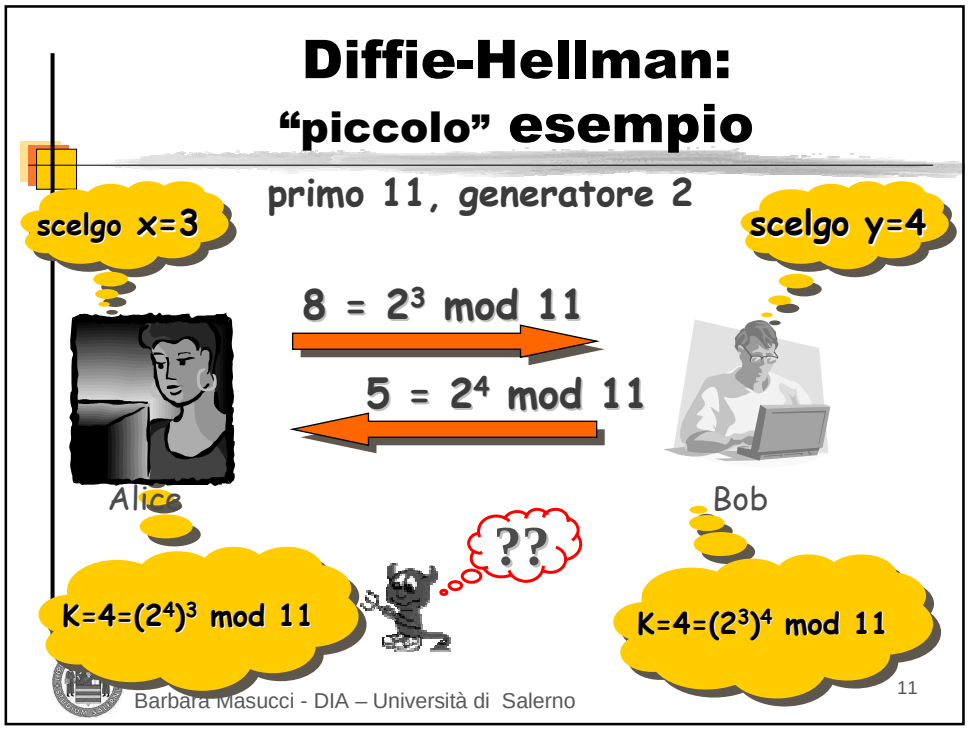
primo p, generatore g di Z_p^*




Diffie-Hellman [1976]
primo p, generatore g
scelgo x∈Zp
scelgo y∈Zp
gˣ mod p
Alice
Bob
??
Barbara Masucci - DIA – Università di Salerno
8


Diffie-Hellman [1976]
primo p, generatore g
scelgo x∈Zp
scelgo x∈Zp
gˣ mod p
gʸ mod p
Alice
Bob
??
Barbara Masucci - DIA – Università di Salerno
9


Diffie-Hellman [1976]
primo p, generatore g
scelgo x∈Zp
scelgo x∈Zp
g^x mod p
g^y mod p
Alice
Bob
??
K = g^xy mod p
= (g^y)^x mod p
K = g^xy mod p
= (g^x)^y mod p
Barbara Masucci - DIA – Università di Salerno
10


Diffie-Hellman:
"piccolo" esempio
primo 11, generatore 2
scelgo x=3
scelgo y=4
8 = 2^3 mod 11
5 = 2^4 mod 11
Alice
Bob
??
K=4=(2^4)^3 mod 11
K=4=(2^3)^4 mod 11
Barbara Masucci - DIA – Università di Salerno
11

Diffie-Hellman: esempio

scelgo x=3578

primo 25307, generatore 2

scelgo y=19956

$6113 = 2^{3578} \bmod 25307$

$7984 = 2^{19956} \bmod 25307$

Alice

Bob

$K=3694=7984^{3578}$

$K=3694=6113^{19956}$

Logaritmo discreto

La sicurezza di molte tecniche crittografiche si basa sulla intrattabilità del logaritmo discreto:

- Crittosistema ElGamal
- Accordo su chiavi Diffie-Hellman
- Firme digitali DSS

Dati a,n,b calcolare x tale che $a^x = b \bmod n$

Esempio: $3^x = 7 \bmod 13$ soluzione $x = 6$

Logaritmo discreto: Complessità algoritmi

Dati a,n,b calcolare x tale che $a^x = b \mod n$

Se n è primo, i migliori algoritmi hanno complessità

$$L_n[a,c] = O(e^{(c+o(1))(\ln n)^a(\ln\ln n)^{1-a}})$$

con $c > 0$ ed $0 < a < 1$

Miglior algoritmo: Number field sieve

tempo medio euristico $L_n[1/3, 1.923]$

Problema di Diffie-Hellman

Input: primo p, generatore g, $g^x \mod p$, $g^y \mod p$
Calcolare: $g^{xy} \mod p$

Il miglior algoritmo conosciuto calcola prima il logaritmo discreto $x \leftarrow \log_{g,p}(g^x \mod p)$

… ma non si sa se sono equivalenti!

Scelta dei parametri

- Come scegliere p e g?

Scelta di un generatore

Scegli_Generatore_Naive (p)

1. Scegli a caso g in Z_p^*
2. **If** $\{g^i | 1 \le i \le p-1\} = Z_p^*$
   then trovato
   else goto 1.

$\{g^i | 1 \le i \le p-1\} = Z_p^*$?

L'unico algoritmo efficiente necessita dei fattori primi di p-1

Scelta di un generatore

p primo, $p - 1 = p_1^{e_1} p_2^{e_2} \dots p_k^{e_k}$

g è un generatore di $Z_p^* \Leftrightarrow$

$$\begin{cases} g^{(p-1)/p_1} \neq 1 \mod p \\ \dots \\ g^{(p-1)/p_k} \neq 1 \mod p \end{cases}$$

Scelta di un generatore

p primo, $p - 1 = p_1^{e_1} p_2^{e_2} \dots p_k^{e_k}$

g è un generatore di $Z_p^* \Leftrightarrow$

$$\begin{cases} g^{(p-1)/p_1} \neq 1 \mod p \\ \dots \\ g^{(p-1)/p_k} \neq 1 \mod p \end{cases}$$

Esempio

- 11 primo, $p-1 = 10 = 2 \cdot 5$
- 2 è un generatore di Z_{11}^* perché

$2^{(11-1)/2} = 2^5 = 10 \neq 1 \mod 11$

$2^{(11-1)/5} = 2^2 = 4 \neq 1 \mod 11$

Scelta di un generatore

p primo, $p - 1 = p_1^{e_1} p_2^{e_2} \dots p_k^{e_k}$

g è un generatore di $Z_p^* \Leftrightarrow$

$$\begin{cases} g^{(p-1)/p_1} \neq 1 \mod p \\ \dots \\ g^{(p-1)/p_k} \neq 1 \mod p \end{cases}$$

Esempio

- 11 primo, $p-1 = 10 = 2 \cdot 5$
- 3 **non** è un generatore di Z_{11}^* perché

$3^{(11-1)/2} = 3^5 = 243 = 1 \mod 11$

$3^{(11-1)/5} = 3^2 = 9 \neq 1 \mod 11$

Scelta di un generatore

p primo, $p - 1 = p_1^{e_1} p_2^{e_2} \dots p_k^{e_k}$

g è un generatore di $Z_p^* \Leftrightarrow$

$$\begin{cases} g^{(p-1)/p_1} \neq 1 \mod p \\ \dots \\ g^{(p-1)/p_k} \neq 1 \mod p \end{cases}$$

Scegli_generatore (p, $(p_1,e_1,p_2,e_2,\dots,p_k,e_k)$)

1. g ← elemento scelto a caso in Z_p^*
2. **if** ($g^{(p-1)/p_1} \neq 1 \mod p$ **and** … **and** $g^{(p-1)/p_k} \neq 1 \mod p$)
   then esci — **trovato!**
   else go to 1.

Probabilità successo singola iterazione

- Numero di generatori modulo un primo p è

$$\phi(\phi(p)) = \phi(p-1)$$
$$> (p-1) / (6 \cdot \ln\ln(p-1))$$

per ogni intero $n \geq 5$, $\phi(n) > n/(6\ln\ln n)$

Probabilità successo singola iterazione

- Numero di generatori modulo un primo p è

$$\phi(\phi(p)) = \phi(p-1)$$
$$> (p-1) / (6 \cdot \ln\ln(p-1))$$

per ogni intero $n \geq 5$, $\phi(n) > n/(6\ln\ln n)$

- Probabilità che un elemento a caso in Z_p^* sia generatore

$$= \frac{\phi(\phi(p))}{\phi(p)} > \frac{p-1}{\phi(p) \cdot 6\ln\ln(p-1)} = \frac{1}{6 \cdot \ln\ln(p-1)}$$

Analisi di Scegli_generatore

Numero medio di iterazioni $< 6 \cdot \ln\ln(p-1)$

512 bit	$6 \cdot \ln\ln(2^{512}) \approx 35{,}23$
1024 bit	$6 \cdot \ln\ln(2^{1024}) \approx 39{,}38$
2048 bit	$6 \cdot \ln\ln(2^{2048}) \approx 43{,}54$

Generazione chiavi Diffie-Hellman

Scegli a caso 2 numeri primi p_1 p_2

$p \leftarrow 1 + 2p_1p_2$

Se p non è primo, go to 1.

$g \leftarrow$ **Scegli_generatore**$(p,(2,1,p_1,1,p_2,1))$

Schema di Merkle

- Non basato su assunzioni computazionali
- Alice genera n chiavi distinte e "nasconde" ogni chiave in un puzzle
  - Il puzzle contiene informazioni per il calcolo della chiave
  - La soluzione di un puzzle richiede un tempo ragionevole
  - La soluzione di tutti i puzzle richiede un tempo troppo elevato

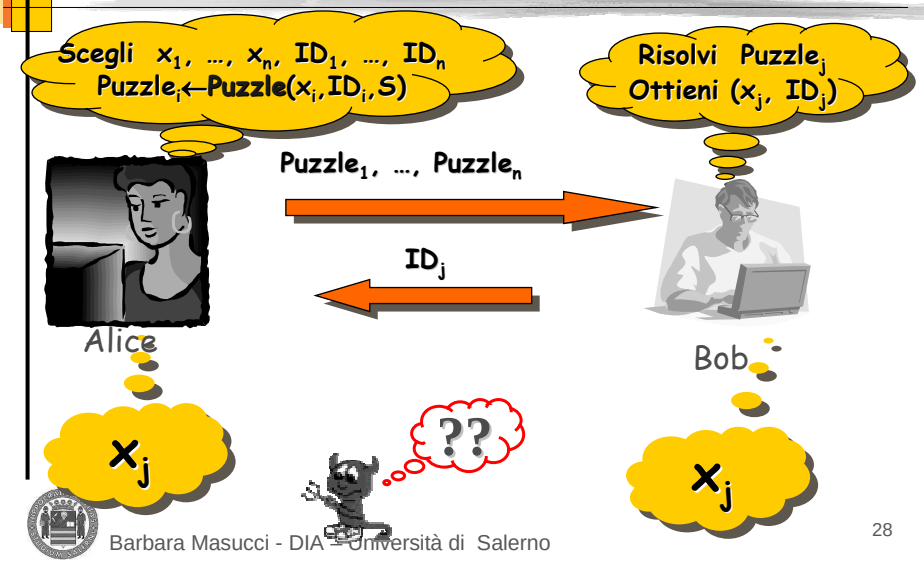
Puzzle di Merkle

- **Puzzle la cui soluzione richiede t operazioni**
- **Esempio:**

Puzzle (x, ID, S)
Scegli una chiave k
Computa $y \leftarrow CBC\text{-}DES_k(x, ID, S)$
return (y, primi 20 bit di k)

- **x è la soluzione del puzzle**
  - **Richiede 2^{35} operazioni in media**
- **ID è l'identificativo del puzzle**
  - **Unico per ciascun puzzle**
- **S è un valore noto**
  - **Serve per garantire l'unicità della soluzione del puzzle**
  - **Esempio: 32 bit nulli**

Puzzle di Merkle



Puzzle di Merkle

Computazioni di  :

- Costruzione di n puzzle — tempo $\theta(n)$

Computazioni di  :

- Risoluzione di un puzzle — tempo $\theta(t)$

Computazioni di  :

- Risoluzione di n/2 puzzle in media — tempo $\theta(t \cdot n)$

Puzzle di Merkle

Computazioni di  :

Se $n = \theta(t)$

- Costruzione di n puzzle — tempo $\theta(n)$

Computazioni di 

- Risoluzione di un puzzle — tempo $\theta(n)$

Computazioni di 

- Risoluzione di n/2 puzzle in media — tempo $\theta(n^2)$