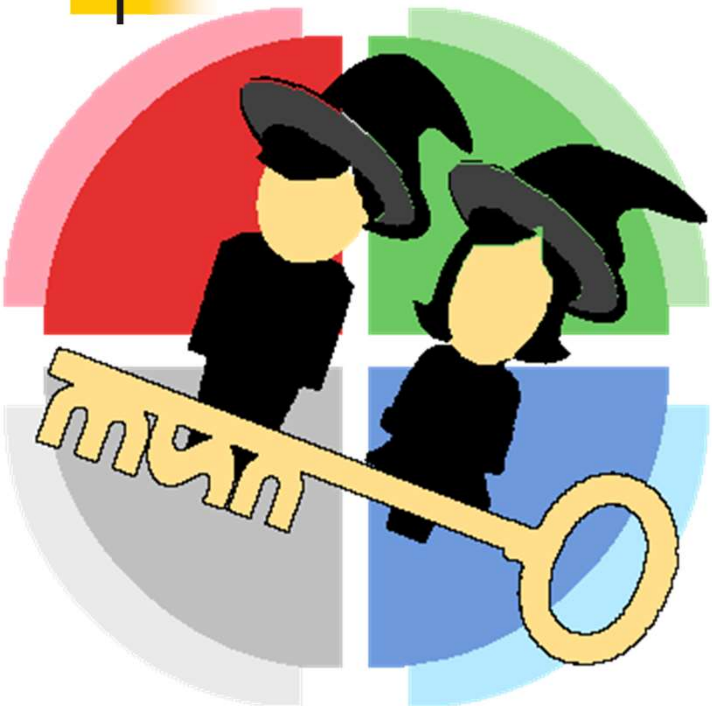
中国剩余定理在信息科学的应用



张真诚
台湾逢甲大学

- 1. Chinese Remainder Theorem (中国剩余定理)
- 2. Ordered Minimal Perfect Hashing Functions (OMPHF)
- 3. Data Field Protection
- 4. Access Control
- 5. Conclusions

Chinese Remainder Theorem (中国剩余定理)

- 韩信点兵问题：「今有物不知其数，三三数之剩二，五五数之剩三，七七数之剩二，问物几何？」～(孙武 孙子算经)

  亦即 求正整数C，使得

$$C \equiv 2(\mathrm{mod}3)$$

$$C \equiv 3(\mathrm{mod}5)$$

$$C \equiv 2(\mathrm{mod}7)$$

  两个疑问？

  (1)C是否存在？

  (2) 如何求C？

- 回答(1)：中国剩余定理
- Let $r_1, r_2, \cdots, r_n$ be integers.
- $\exists$ an integer C . st.

$$
\begin{aligned}
C &\equiv r_1 (\bmod m_1) \\
C &\equiv r_2 (\bmod m_2) \\
&\vdots \\
C &\equiv r_n (\bmod m_n)
\end{aligned}
$$

If $(m_i, m_j) = 1$, $\forall i \neq j$

- EX： 令 $m_1 = 3, m_2 = 5, m_3 = 7$,且令

  $r_1 = 2, r_2 = 3, r_3 = 2,$

  $\exists C = 23,$ s.t.

$$C \bmod m_1 = 23 \bmod 3 = 2$$

$$C \bmod m_2 = 23 \bmod 5 = 3$$

$$C \bmod m_3 = 23 \bmod 7 = 2$$

- 回答(2):
- 「三人同行七十稀
  五树梅花廿一枝
  七子团圆正半月
  除百零五便得知。」

～(程大位 算法统宗(1593))

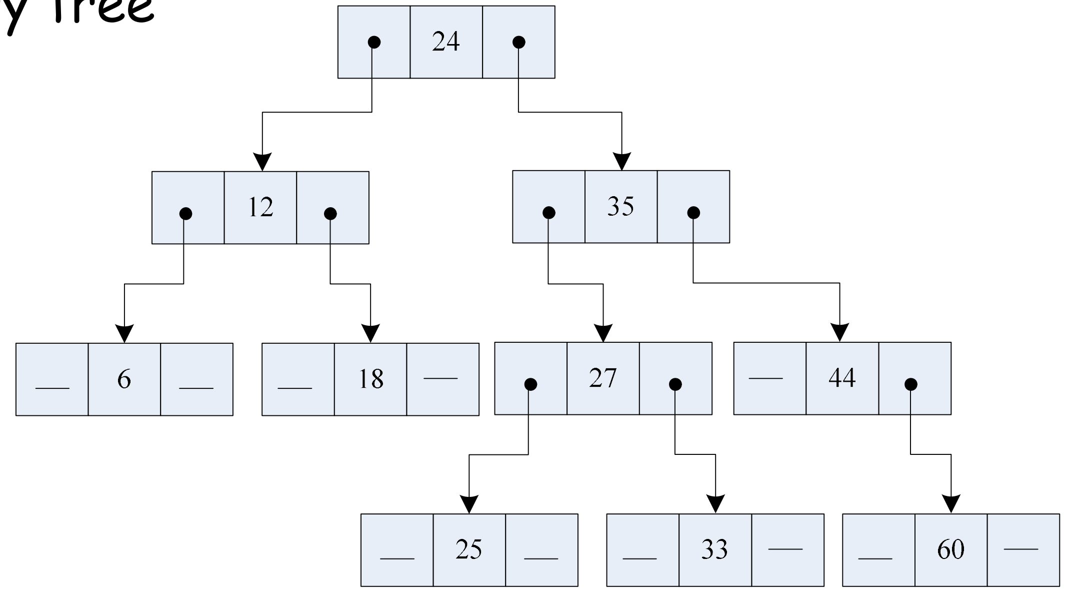
- 亦即

$$\begin{aligned} & 70 \times 2 + 21 \times 3 + 15 \times 2 \\ = \ & 140 + 63 + 30 \\ = \ & 233 \end{aligned}$$

$$233 \div 105 = 3 \text{ 余 } 23$$

Ordered Minimal Perfect Hashing Functions

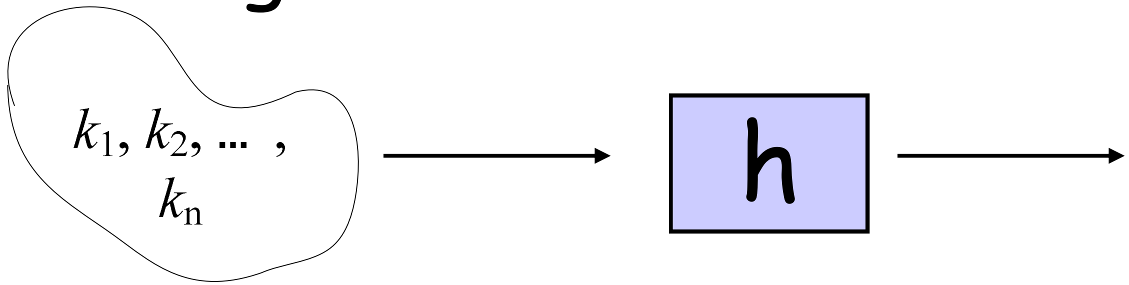
- Non-linear Organization
  - Ex: binary tree

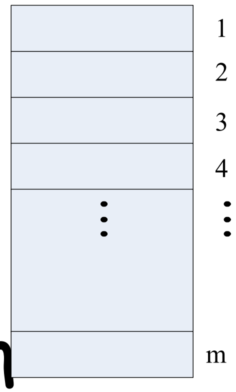


  - Linear Organization

6	12	18	24	25	27	33	35	44	60

- hashing

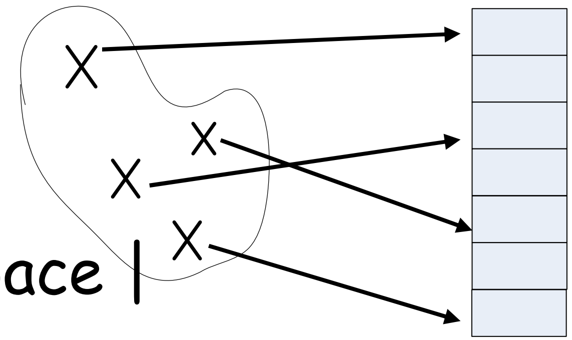




- key to address transformation
  - problem: collision
  - particular cases

    (1) one-to-one mapping and

    | key space | ≤ | address space |

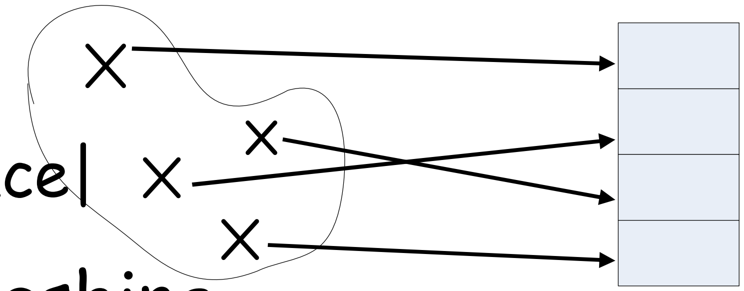


➡ Perfect hashing

Hashing

(2)one-to-one mapping and

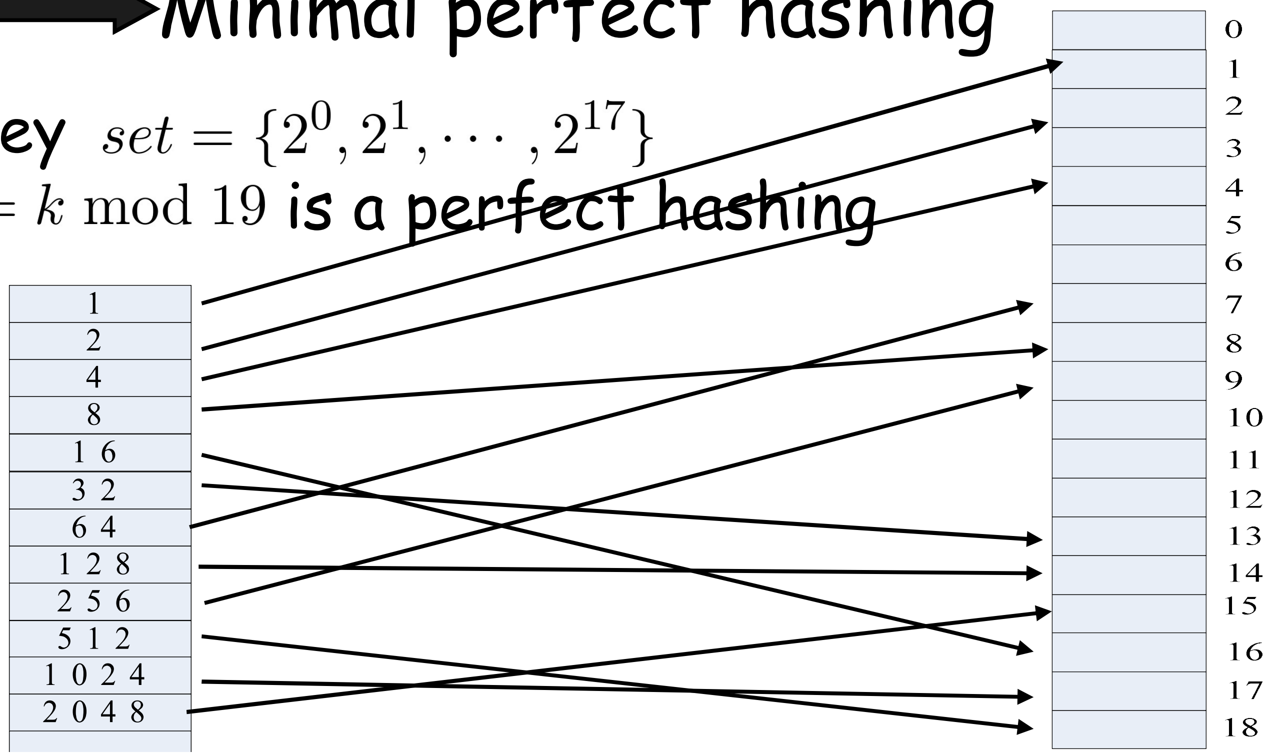
| key space | = | address space|



→ Minimal perfect hashing

Ex: key $set = \{2^0, 2^1, \cdots, 2^{17}\}$

$h(k) = k \bmod 19$ is a perfect hashing



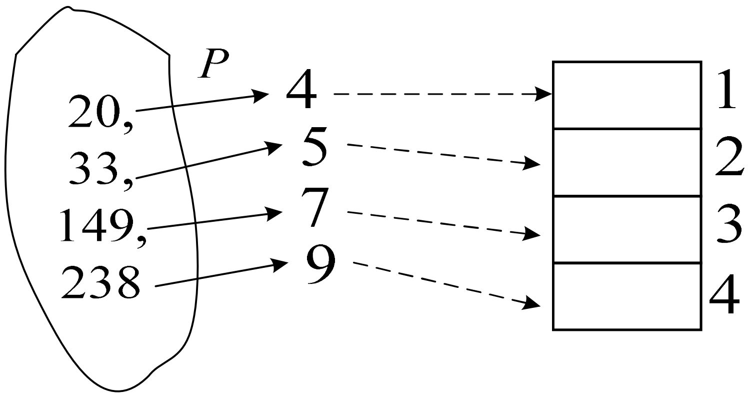
Minimal Perfect Hashing

$$K = \{k_1, k_2, \cdots, k_n\}$$

$h(k_i)$ is defined as $C \bmod P(k_i)$

$$Condition : (P(k_i), P(k_j)) = 1,$$

$$h(k) = 157 \bmod p(k)$$



since

$$\frac{157}{p(20)} = \frac{157}{4} =? \cdots\cdots 1$$

$$\frac{157}{p(33)} = \frac{157}{5} =? \cdots\cdots 2$$

$$\frac{157}{p(149)} = \frac{157}{7} =? \cdots\cdots 3$$

$$\frac{157}{p(238)} = \frac{157}{9} =? \cdots\cdots 4$$

Minimal Perfect Hashing (Cont.)

- Some questions

(1) Why $h(k_i) = C \bmod p(k_j)$ can be"1-1" and "onto"?

(2) Is there a method to convert

$$\{k_1, k_2, \cdots, k_n\}$$

$$\Downarrow$$

$\{P(k_1), P(k_2), \cdots, P(k_n)\}$ such that $(P(k_i), P(k_j)) = 1$

(3) How to obtain C ?

Minimal Perfect Hashing (Cont.)

- Ans(1): Chinese Remainder Theorem

Let $r_1, r_2, \cdots, r_n$ be an integers.

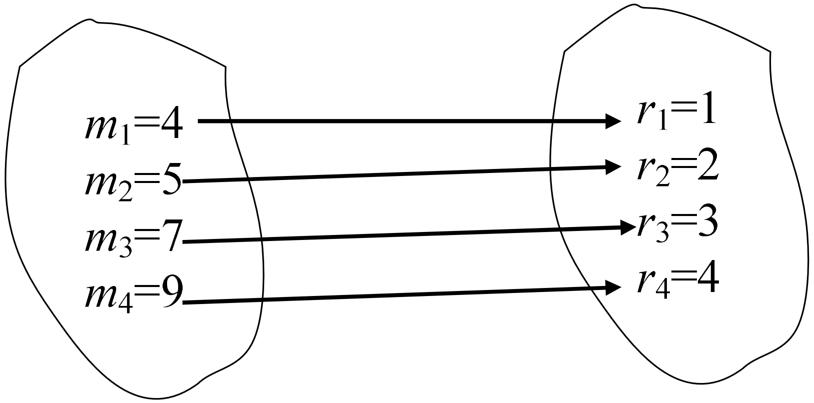
$\exists$ an integer c s.t.

$$\frac{C}{m_1} = ? \cdots r_1$$

$$\frac{C}{m_2} = ? \cdots r_2$$

$$\vdots$$

$$\frac{C}{m_n} = ? \cdots r_n$$



If $(m_i, m_j) = 1, \forall i \neq j, \exists C = 157$

Minimal Perfect Hashing (Cont.)

- Ans(2): Use Prime Number Functions

$$
\begin{aligned}
p(x) &= x^2 - x + 17, \text{for} \quad 1 \le x \le 16 \\
p(x) &= x^2 - x + 41, \text{for} \quad 1 \le x \le 40 \\
p(x) &= x^2 - 81x + 1681, \text{for} \quad 41 \le x \le 80 \\
p(x) &= x^2 + x + 41, \text{for} \quad 1 \le x \le 39 \\
p(x) &= x^2 - 79x + 1601, \text{for} \quad 40 \le x \le 79
\end{aligned}
$$

Ans(3):

$$C = \sum_{i=1}^{n} b_i M_i i \bmod M$$

$$where \quad M_i = \prod_{\substack{j=1 \\ j \ne i}}^{n}, M = \prod_{j=1}^{n} m_i$$

$$and \quad b_i \ni M_i b_i \equiv 1 (\bmod m_i)$$

THEOREM 22.10 Let $K=\{k_1,k_2,\cdots,k_n\}$ be a set of n distinct positive integers with $k_i<k_{i+1}$ .if $\{t_1,t_2,\cdots,t_s\}=\{k_i-k_j:1\leq j<i\leq n\}$ is the set of $s=n(n-1)/2$ differences ,then $D=w\cdot lcm(t_1,t_2,\cdots,t_s)$ where w is any positive integer, has the property that $Dk_1+1,Dk_2+1,\cdots,Dk_n+1$ are pairwise relatively prime.

Minimal Perfect Hashing (Cont.)

- Ex: $m_1 = 4, m_2 = 5, m_3 = 7, m_4 = 9$

$$\begin{aligned} M_1 &= 5 \times 7 \times 9 = 315 \\ M_2 &= 4 \times 7 \times 9 = 252 \\ M_3 &= 4 \times 5 \times 9 = 180 \\ M_4 &= 4 \times 5 \times 7 = 140 \end{aligned}$$

$$\begin{aligned} \mathrm{M}_1 b_1 &\equiv 1 (\mod m_1) \\ M_2 b_2 &\equiv 1 (\mod m_2) \\ M_3 b_3 &\equiv 1 (\mod m_3) \\ M_4 b_4 &\equiv 1 (\mod m_4) \end{aligned}$$

$$\begin{aligned} 315 \mathrm{b}_1 &\equiv 1 (\mod 4) \\ 252 b_2 &\equiv 1 (\mod 5) \\ 180 b_3 &\equiv 1 (\mod 7) \\ 140 b_4 &\equiv 1 (\mod 9) \end{aligned}$$

$$\begin{aligned} \mathrm{b}_1 &= -1 \\ b_2 &= -2 \\ b_3 &= 3 \\ b_4 &= 2 \end{aligned}$$

$$\begin{aligned} C' &= \sum_{i=1}^{4} b_1 M_i i \\ &= (-1) \times 315 \times 1 \\ &+ (-2) \times 252 \times 2 \\ &+ 3 \times 180 \times 3 \\ &+ 2 \times 140 \times 4 \\ &= 1417 \end{aligned}$$

$$\begin{aligned} C &= 1417 \bmod (m_1 \times m_2 \times m_3 \times m_4) \\ &= 1417 \bmod 1260 = 157 \end{aligned}$$

Applications-12 months English identifiers

JANUARY	JULY
FEBRUARY	AUGUST
MARCH	SEPTEMBER
APRIL	OCTOBER
MAY	NOVEMBER
JUNE	DECEMBER

Applications-12 months English identifiers (Cont.)

- Extract (The 2nd char., The 3rd char.)

(A, N)	(U, L)
(E, B)	(U, G)
(A, R)	(E, P)
(P, R)	(C, T)
(A, Y)	(O, V)
(U, N)	(E, C)

Applications-12 months English identifiers (Cont.)

Group	Location	Extraction pair	Original key
1	1 2 3	(A, N) (A, R) (A, Y)	JANUARY MARCH MAY
2	4	(C, T)	OCTOBOR
3	5 6 7	(E, B) (E, C) (E, P)	FEBRUARY DECEMBER SEPTEMBER
4	8	(O, V)	NOVEMBER
5	9	(P, R)	APRIL
6	10 11 12	(U, G) (U, L) (U, N)	AUGUST JULY JUNE

Applications-12 months English identifiers (Cont.)

x	d(x)	c(x)	p(x)
A	0	161896	2
B			3
C	3	1	5
D			7
E	4	427	11
F			13
G			17
H			19
I			23
J			29
K			31
L			37
M			41

x	d(x)	c(x)	p(x)
N			43
O	7	1	47
P	8	1	53
Q			59
R			61
S			67
T			71
U	9	12989	73
V			79
W			83
X			89
Y			97
Z			101

$$\begin{aligned} H(P,R) &= d(P) + (c(P) \bmod p(R)) \\ &= 8 + (1 \bmod 61) \\ &= 8 + 1 \\ &= 9 \end{aligned}$$

36 Pascal Reserved Words

ARRAY, AND, BEGIN, CASE,

CONST, DOWNTO, DO, DIV, END,

ELSE, FUNCTION, FILE, FOR, GOTO,

IF, IN, LABEL, MOD, NIL, NOT,

OTHERWISE, OF, OR, PROCEDURE,

PROGRAM, PACKED, REPEAT, RECORD,

SET, TYPE, THEN, TO, UNTIL, VAR,

WITH, WHILE

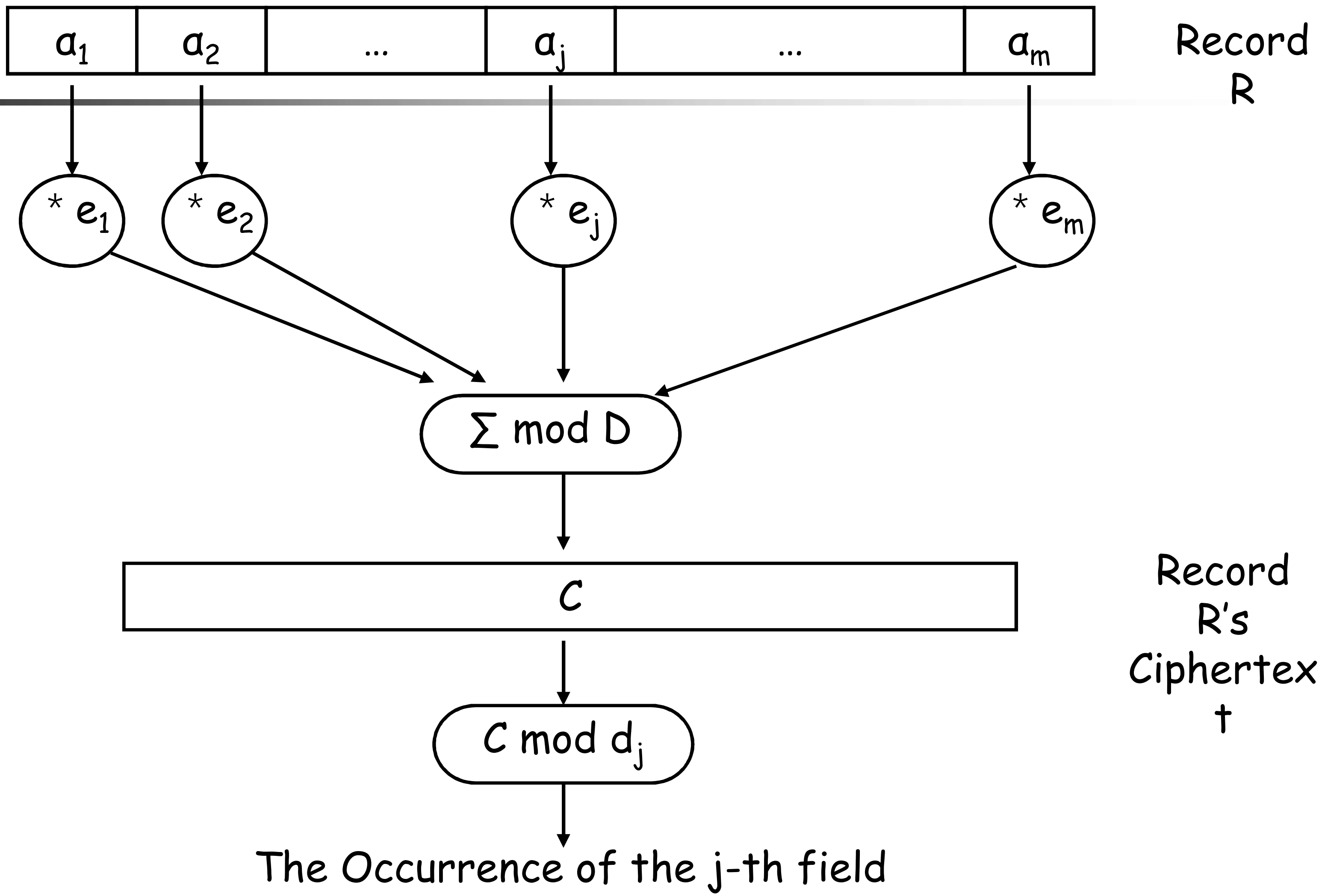
AA, AD, BI, CE, CS, DN
DO, DV, ED, EE, FC, FE
GO, IF, IN, LE, MD, NL
NT, OE, OF, OR, PC, PG
PK, RE, RO, ST, TE, TN
TO, UI, VR, WH, WL

x	d(x)	c(x)	p(x)
A	0	9	2
B	2	1	3
C	3	672	5
D	5	5001	7
E	8	0	11
F	10	57	13
G	13	3236	17
H	14	1	19
I	16	131	23
J	17	1	29
K		1	31
L			37
M			41

x	d(x)	c(x)	p(x)
N	18	1777	43
O	20	3785	47
P	23	726	53
Q			59
R	26	331	61
S	28	1	67
T	29	1565	71
U	32	4	73
V	33	1	79
W	34	1	83
X		39	89
Y			97
Z			101

$$\begin{aligned} H(W, L) &= d(W) + (c(W) \bmod p(L)) \\ &= 34 + (39 \bmod 37) \\ &= 34 + 2 \\ &= 36 \end{aligned}$$

Data Field Protection



Data Field Protection (Cont.)

- Example:

  Let $R = (4, 10, 2)$ be a record. Take $d_1 = 7, d_2 = 11$ and $d_3 = 5$ to be R's three deciphering keys. Then $D = d_1 \times d_2 \times d_3 = 7 \times 1 \times 5 = 385$

  The following are encryption keys

$$e_1 = (\frac{D}{d_1})b_1 = \frac{385}{7} \times 6 = 55 \times 6 = 330$$

$$e_2 = (\frac{D}{d_2})b_2 = \frac{385}{11} \times 6 = 35 \times 6 = 210$$

$$e_3 = (\frac{D}{d_3})b_3 = \frac{385}{5} \times 3 = 77 \times 3 = 231$$

Data Field Protection (Cont.)

- Because $a_1 = 4, a_2 = 10$ and $a_3 = 2$, we have $C = (e_1 a_1 + e_2 a_2 + e_3 a_3) \bmod D$. Thus R's ciphertext is $C = (330 \times 4 + 210 \times 10 + 231 \times 2) \bmod 385 = 32$.

  Then we have

$$a_1 = C \mod d_1 = 32 \bmod 7 = 4$$

$$a_2 = C \mod d_2 = 32 \bmod 11 = 10$$

$$a_3 = C \mod d_3 = 32 \bmod 5 = 2$$

Data Field Protection (Cont.)

- When a_2 is changed from 10 to 8, R's ciphertext becomes

$$
\begin{aligned}
C &= (C - e_2(C \bmod d_2) + e_2 \times 8) \bmod D \\
&= (32 - 210 \times 10 + 210 \times 8) \bmod 385 \\
&= -388 \bmod 385 \\
&= 382
\end{aligned}
$$

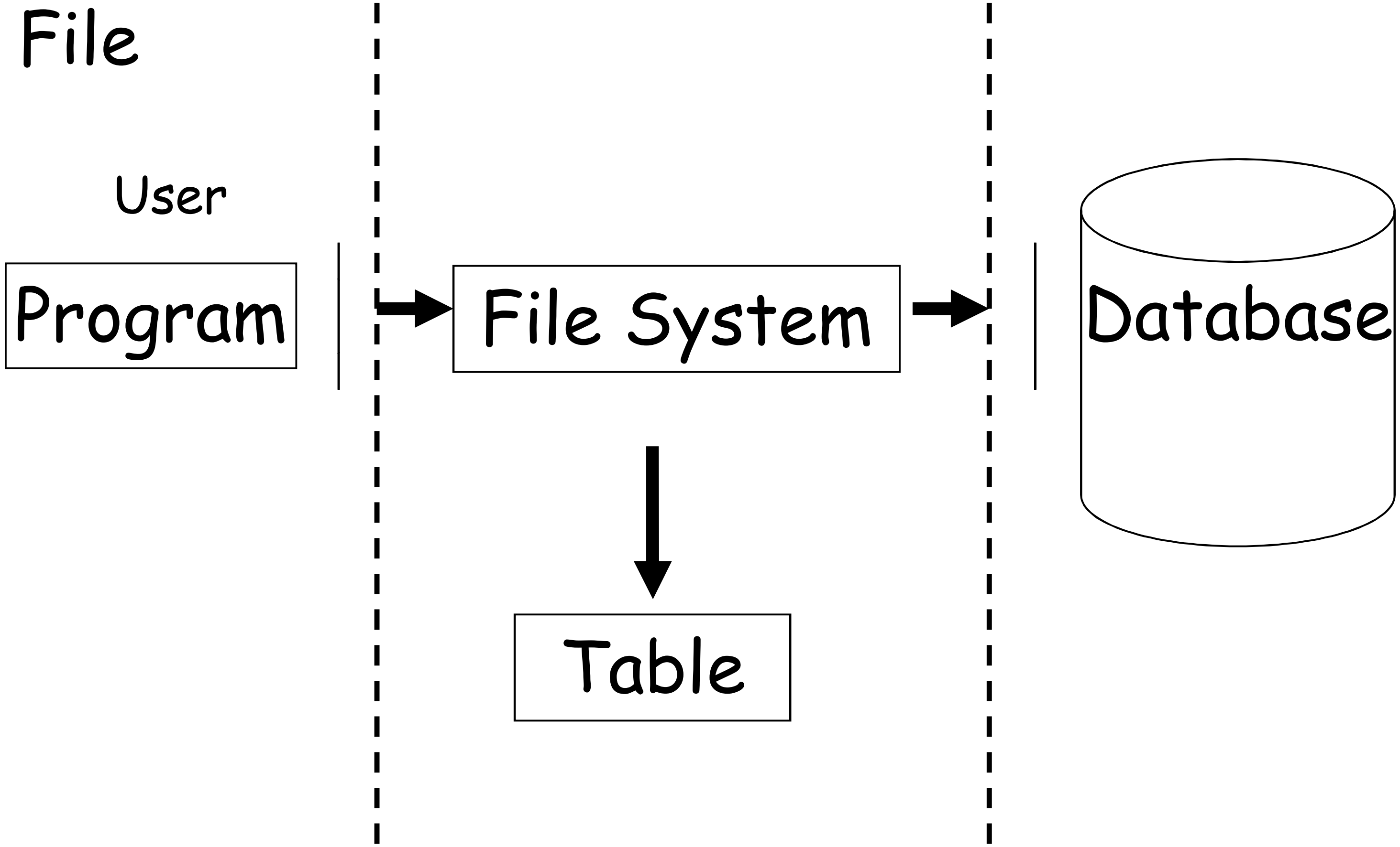
Access Control

	M_1	M_2	M_3	M_4
P1	R W E		Own R W	
P2		R W E		Own R E

Access Matrix

Access Control (Cont.)

- File



Access Control (Cont.)

User \ File	A	B	C	D	E	F
U1	4	4	1	2	0	0
U2	3	3	4	4	0	1
U3	1	3	1	3	4	4
U4	1	2	1	0	2	3
U5	0	1	2	0	3	3

0 : No access

1 : Execute

2 : Read

3 : Write

4 : Own

Key_Lock Pair

A. Wu and Hwang [1984]

$$a_{ij} = K_i * L_j$$

(1) * ： inner product of Galois Field GF(p)
(2) p>a_{ij} and p is a prime number

Step1:Find a 5*5 Non_Singular Matrix

$$T = \begin{bmatrix} 3 & 1 & 5 & 6 & 5 \\ 1 & 2 & 3 & 5 & 3 \\ 4 & 1 & 1 & 4 & 1 \\ 2 & 6 & 1 & 1 & 2 \\ 5 & 5 & 6 & 5 & 4 \end{bmatrix}$$

Key_Lock Pair (Cont.)

Step 2:

$$K_1 = (3, 1, 5, 6, 5)$$
$$K_2 = (1, 2, 3, 5, 3)$$
$$K_3 = (4, 1, 1, 4, 1)$$
$$K_4 = (2, 6, 1, 1, 2)$$
$$K_5 = (5, 5, 6, 5, 4)$$

Step 3:

$$L_1 = (X_1, X_2, X_3, X_4, X_5)$$
$$K_i * L_1 = a_{i1}, i = 1, 2, \cdots, 5$$

Key_Lock Pair (Cont.)

$$4 = 3X_1 + X_2 + 5X_3 + 6X_4 + 5X_5$$

$$3 = X_1 + 2X_2 + 3X_3 + 5X_4 + 3X_5$$

$$1 = 4X_1 + X_2 + X_3 + 4X_4 + X_5$$

$$1 = 2X_1 + 6X_2 + X_3 + X_4 + 2X_5$$

$$0 = 5X_1 + 5X_2 + 6X_3 + 6X_4 + 4X_5$$

$$\Downarrow$$

$$L_1 = (1, 3, 0, 1, 4)$$

$$\Downarrow$$

Key_Lock Pair (Cont.)

User	Key
U1	$K_1=(3, 1, 5, 6, 5)$
U2	$K_2=(1, 2, 3, 5, 3)$
U3	$K_3=(4, 1, 1, 4, 1)$
U4	$K_4=(2, 6, 1, 1, 2)$
U5	$K_5=(5, 5, 6, 6, 4)$

File	Lock
1	$L_1=(1, 3, 0, 1, 4)$
2	$L_2=(1, 2, 6, 0, 5)$
3	$L_3=(1, 2, 3, 2, 5)$
4	$L_4=(0, 5, 5, 2, 6)$
5	$L_5=(0, 5, 5, 0, 1)$
6	$L_6=(4, 2, 3, 4, 2)$

Key_Lock Pair (Cont.)

- Example:

$$K_4 * L_1 = (2,6,1,1,2) * (1,3,0,1,4)$$
$$= 2 + 18 + 0 + 1 + 8 = 29$$

$$29 \bmod 7 = 1 \ \#$$

- Disadvantage:

  (1) Space: O(m^2+mn) > O(mn)

  (2) Time: 1. m multiplications

  2. m-1 additions

  3. 1 divisions

  ※ m users and n files.

Key_Lock Pair (Cont.)

B. Chang [1985]

$a_{ij}=K_i \bmod L_j$

	File	A	B	C	D	E	F
User		5	6	7	11	13	17
U1	102544	4	4	1	2	0	0
U2	351663	3	3	4	4	0	1
U3	213711	1	3	1	3	4	4
U4	415976	1	2	1	0	2	3
U5	497695	0	1	2	0	3	3

Lock

Key

Example:

$$\begin{aligned} a_{45} &= K_4 \bmod L_5 \\ &= 415976 \bmod 13 \\ &= 2 \end{aligned}$$

Key_Lock Pair (Cont.)

Questions??

(1) Why $a_{ij} = K_i \bmod L_j$?

(2) How to find K_i's and L_j's ?

Answer (1)

Chinese Remainder Theorem

Let $r_1, r_2, \cdots, r_n$ be integers ,

$\exists$ an integer K s.t.

$$\frac{K}{L_1} =? \cdots r_1$$

$$\frac{K}{L_2} =? \cdots r_2$$

$$\vdots$$

$$\frac{K}{L_n} =? \cdots r_n$$

If $(L_i, L_j) = 1, \forall i \neq j$

Key_Lock Pair (Cont.)

Ex: $\exists$ K=102544

$L_1=5 \longrightarrow 4$

$L_2=6 \longrightarrow 4$

$L_3=7 \longrightarrow 1$

$L_4=11 \longrightarrow 2$

$L_5=13 \longrightarrow 0$

$L_6=17 \longrightarrow 0$

<u>Answer(2):</u>

Lock: $L_1, L_2, \ldots, L_n$, coprimes

Key_Lock Pair (Cont.)

- Key

$$K_i = \sum_{j=1}^{n} b_j * D_j * aij$$

(1) $$D_j = \prod_{\substack{i=1 \\ j \neq i}}^{n} L_i$$

(2) $$D_j b_j \equiv 1 (\bmod L_j), \forall j$$

- Example

$$L_1 = 5, L_2 = 6, L_3 = 7, L_4 = 11, L_5 = 13, L_6 = 17$$

$$D_1 = L_1 L_2 L_3 L_4 L_5 L_6 = 6 * 7 * 11 * 13 * 17 = 102102$$
$$D_2 = L_1 L_2 L_3 L_4 L_5 L_6 = 5 * 7 * 11 * 13 * 17 = 85085$$
$$D_3 = L_1 L_2 L_3 L_4 L_5 L_6 = 5 * 6 * 11 * 13 * 17 = 72930$$
$$D_4 = L_1 L_2 L_3 L_4 L_5 L_6 = 5 * 6 * 7 * 13 * 17 = 46410$$
$$D_5 = L_1 L_2 L_3 L_4 L_5 L_6 = 5 * 6 * 7 * 11 * 17 = 39270$$
$$D_6 = L_1 L_2 L_3 L_4 L_5 L_6 = 5 * 6 * 7 * 11 * 13 = 30030$$

Key_Lock Pair (Cont.)

$\because \; D_j b_j \equiv 1(\bmod L_j)$

$\therefore \; b_1 = 3, b_2 = -1, b_3 = -12,$
$\quad b_4 = 1, b_5 = 4, b_6 = 15$

$$\begin{aligned} \Rightarrow K_1 = \sum_{j=1}^{6} b_j D_j a_{1j} &= 3 * 102102 * 4 \\ &+ (-1) * 85085 * 4 \\ &+ (-12) * 72930 * 1 \\ &+ 1 * 46410 * 2 \\ &+ 4 * 39270 * 0 \\ &+ 15 * 30030 * 0 \\ &= 102544 \end{aligned}$$

If $Mb \equiv 1(\bmod m)$, where $(M, m) = 1$, is there any efficient way to find b?

$Mx + my = 1(e.g. 8x + 3y = 1)$, find $(x, y) = ?$

How to deal with large K_i ?

Conclusions

- Design a perfect hashing function to allow insertion and deletion of keys
- How to speed up the calculation of C?
- Multi-key hashing
- How to deal with large K_i problem?
- Apply Data Field Protection to Secure Broadcasting
- More applications?