

Code-Based PKCs And Their Applications

Kazukuni Kobara
RCIS/AIST

ICITS 2009

No. 1

PKCs can be divided into

Number Theoretic (Cyclic) Problem

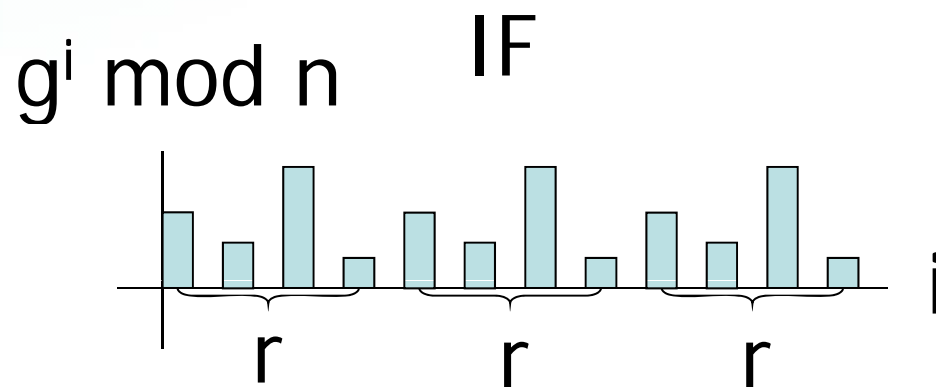
- Integer Factoring Based
 - RSA[RSA78]
 - Rabin[Ra79]
 - Okamoto-Uchiyama[OU98]
 - Paillier[Pa99]
 - S-Paillier[CHGN01]
 - ...
- Discrete Logarithm Based
 - Diffie-Hellman[DH77]
 - ElGamal[El84]
 - ECC[Mi85][Ko87]
 - XTR[LV00]
 - Cramer-Shoup[CS03]
 - Kurosawa-Desmedt[KD04]
 - ...

Combinatorial Problem

- Code Based
 - McEliece[Mc78]
 - Niederreiter[Ni86]
- Lattice Based
 - NTRU[HS96]
 - AjtaiDwork[AD97]
 - Goldreich-Goldwasser-Halevi [GGH97]
 - Ajtai[Ajt05]
 - Regev[Reg03,Reg05]
 - Peikert[Pei09]
 - ...
- Subset Sum Based
 - Okamoto-Tanaka-Uchiyama[OTU00]

Cyclic Problem: Integer Factoring (IF)

- Given a positive integer n , find its prime factor p_i
- Equivalent to finding r s.t. $g^r \equiv 1 \pmod{n}$
 - for a g of $\text{GCD}(g, n) = 1$ and $g \not\equiv \pm 1 \pmod{n}$
- Since $\text{GCD}(g^{r/2} - 1, n)$ or $\text{GCD}(g^{r/2} + 1, n)$ is p_i



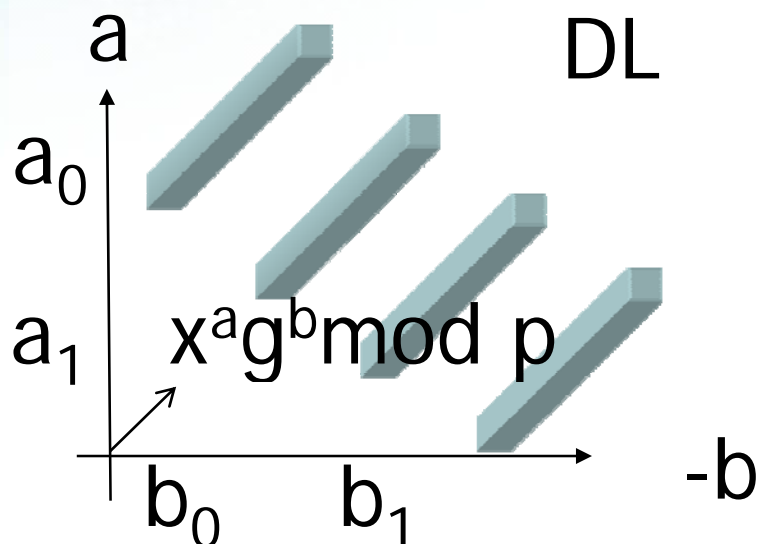
This cycle can be determined in poly time with **1D-QFT**, a.k.a. Shor's quantum algorithm.

QFT : Quantum Fourier Transform
No. 3

Cyclic Problem: Discrete-Log (DL)

- Given p , g and y , find r s.t. $x \equiv g^r \pmod{p}$
 - p : prime, g : generator of \mathbb{Z}_p^* , x a member of \mathbb{Z}_p^* except 1, $p-1$.
- **Equivalent to** finding a_0, a_1, b_0 and b_1 s.t.

$$x^{a_0} g^{b_0} \equiv x^{a_1} g^{b_1} \pmod{p}$$
 - Since $x^{(a_1-a_0)} \equiv g^{r(a_1-a_0)} \equiv g^{(b_0-b_1)} \pmod{p}$

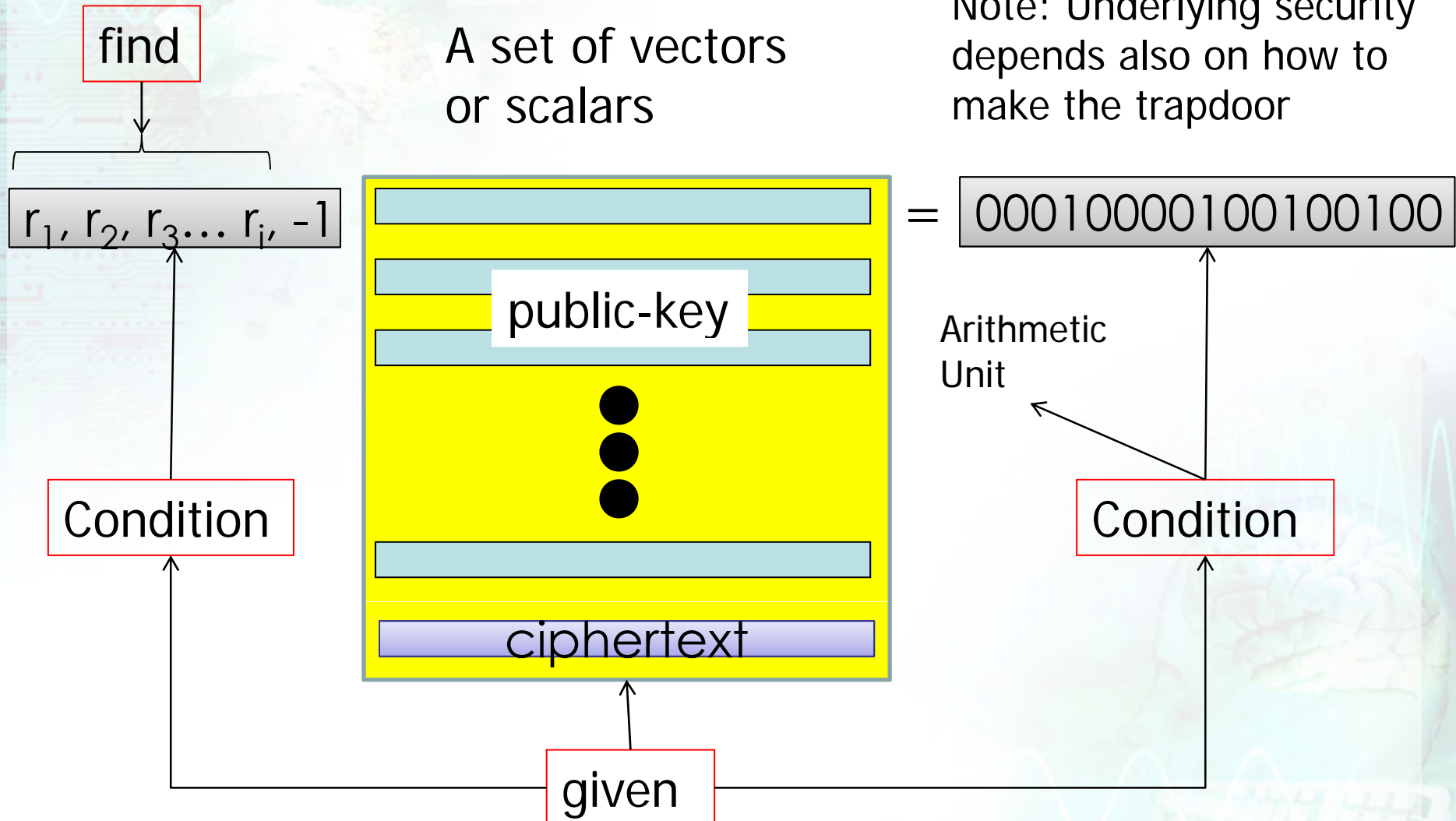


$$r \equiv \frac{(b_0 - b_1)}{(a_1 - a_0)} \pmod{p-1}$$

This inclination can be determined in poly time with **2D-QFT**, a.k.a. Shor's quantum algorithm.

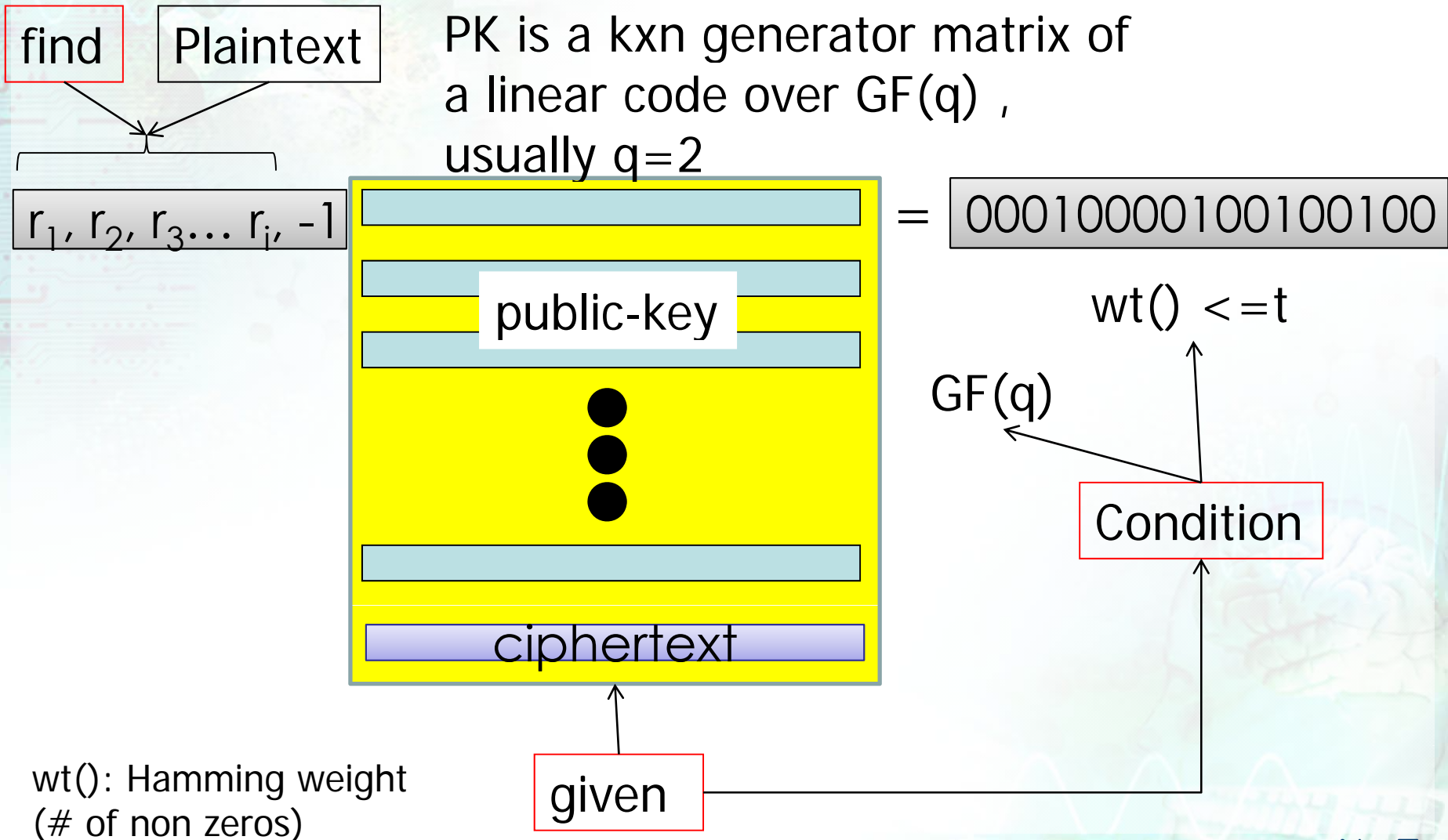
Lesson to learn

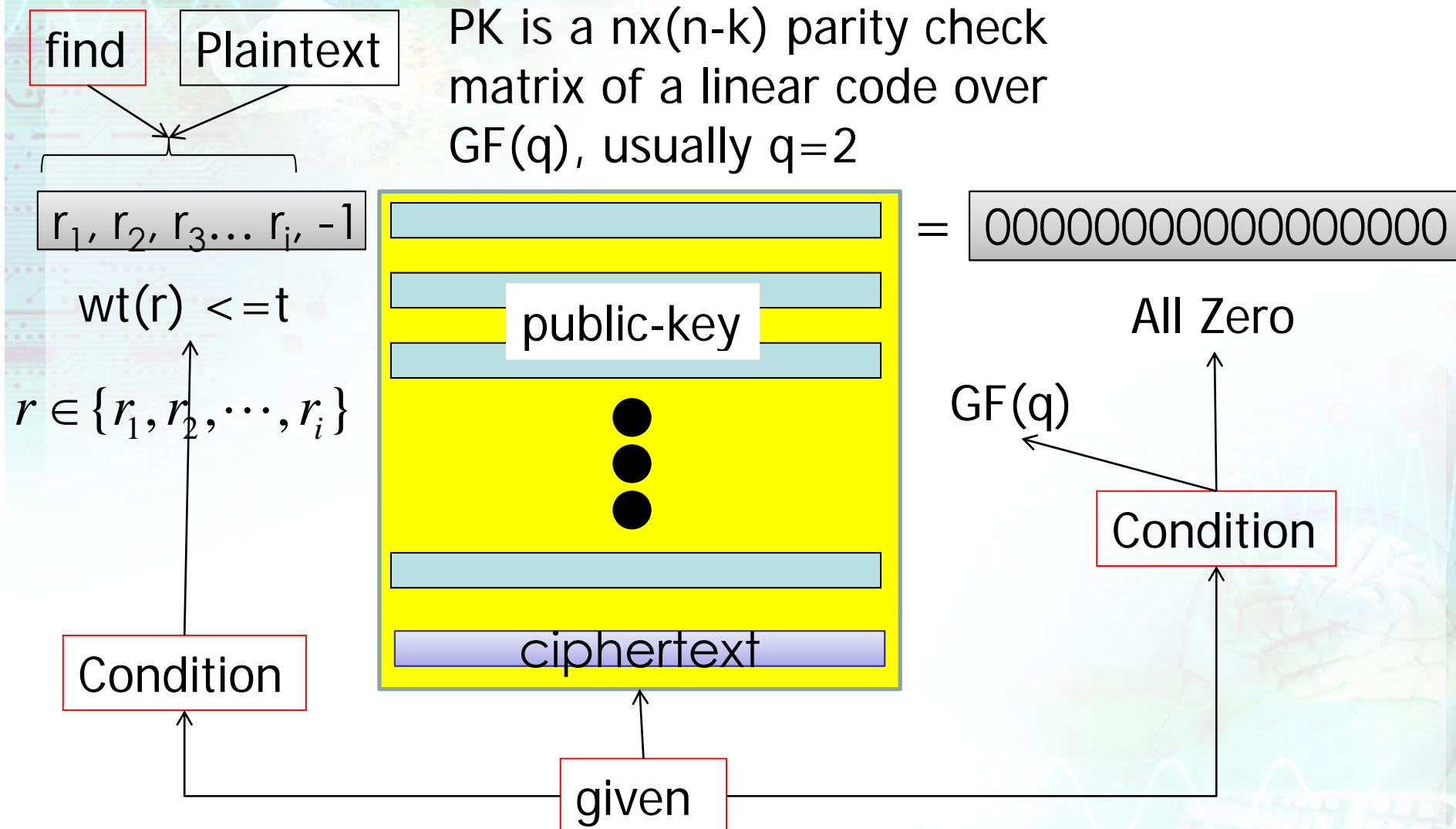
- If a problem is written in the form of determining a **cycle**,
- Then it can be solved in **polynomial time** using a quantum computer.



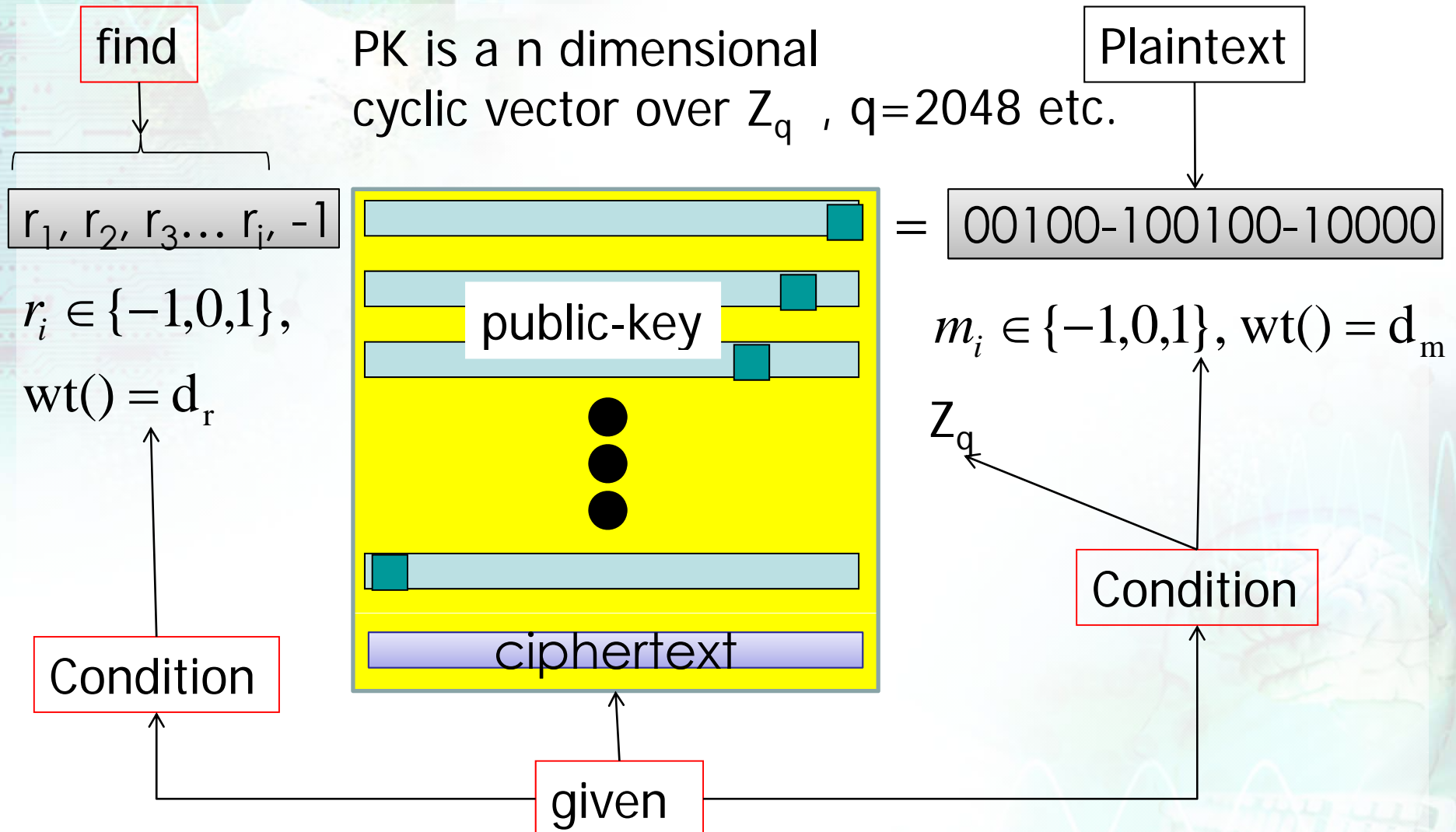
Note: Underlying security depends also on how to make the trapdoor

Code Based: McEliece

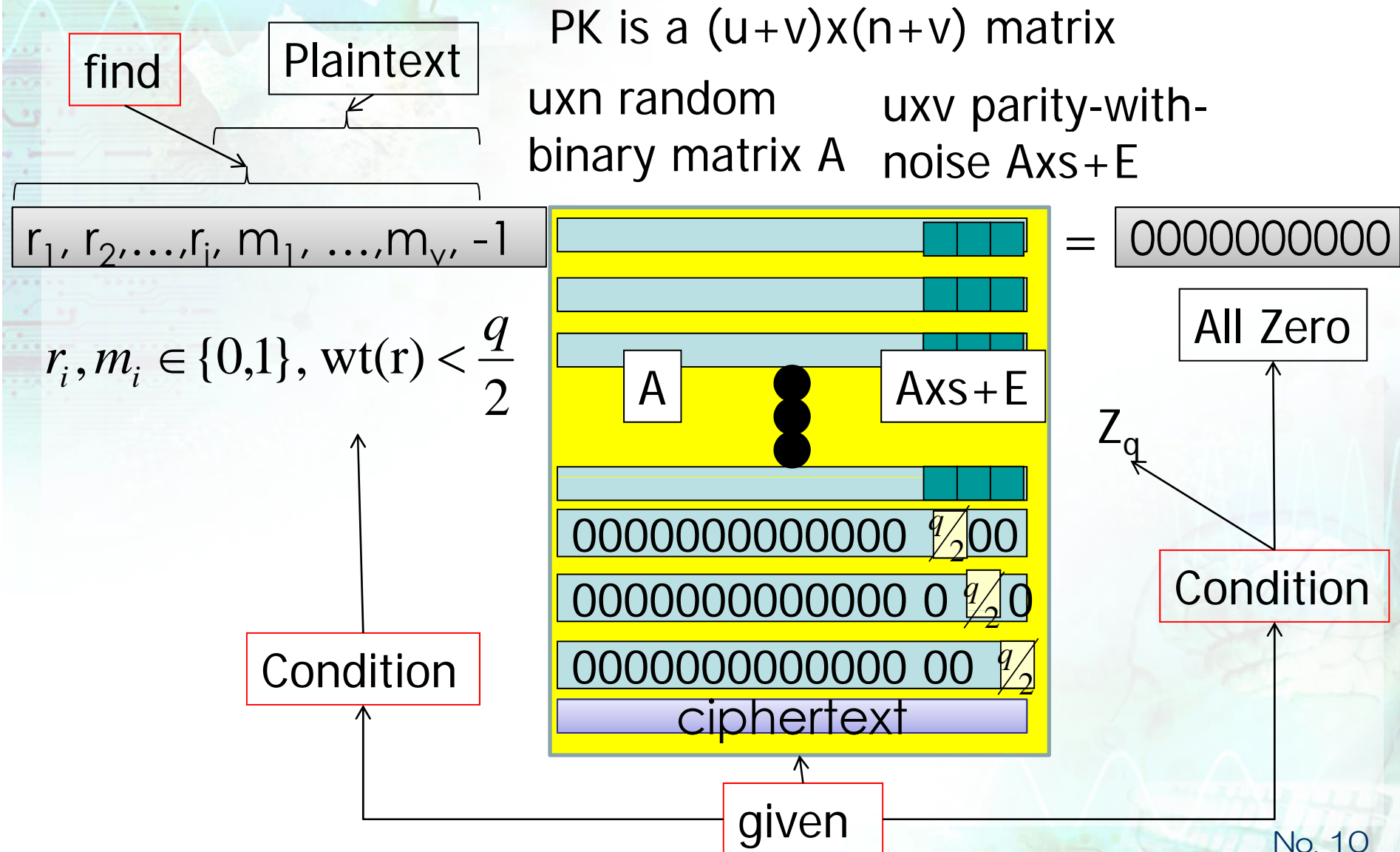




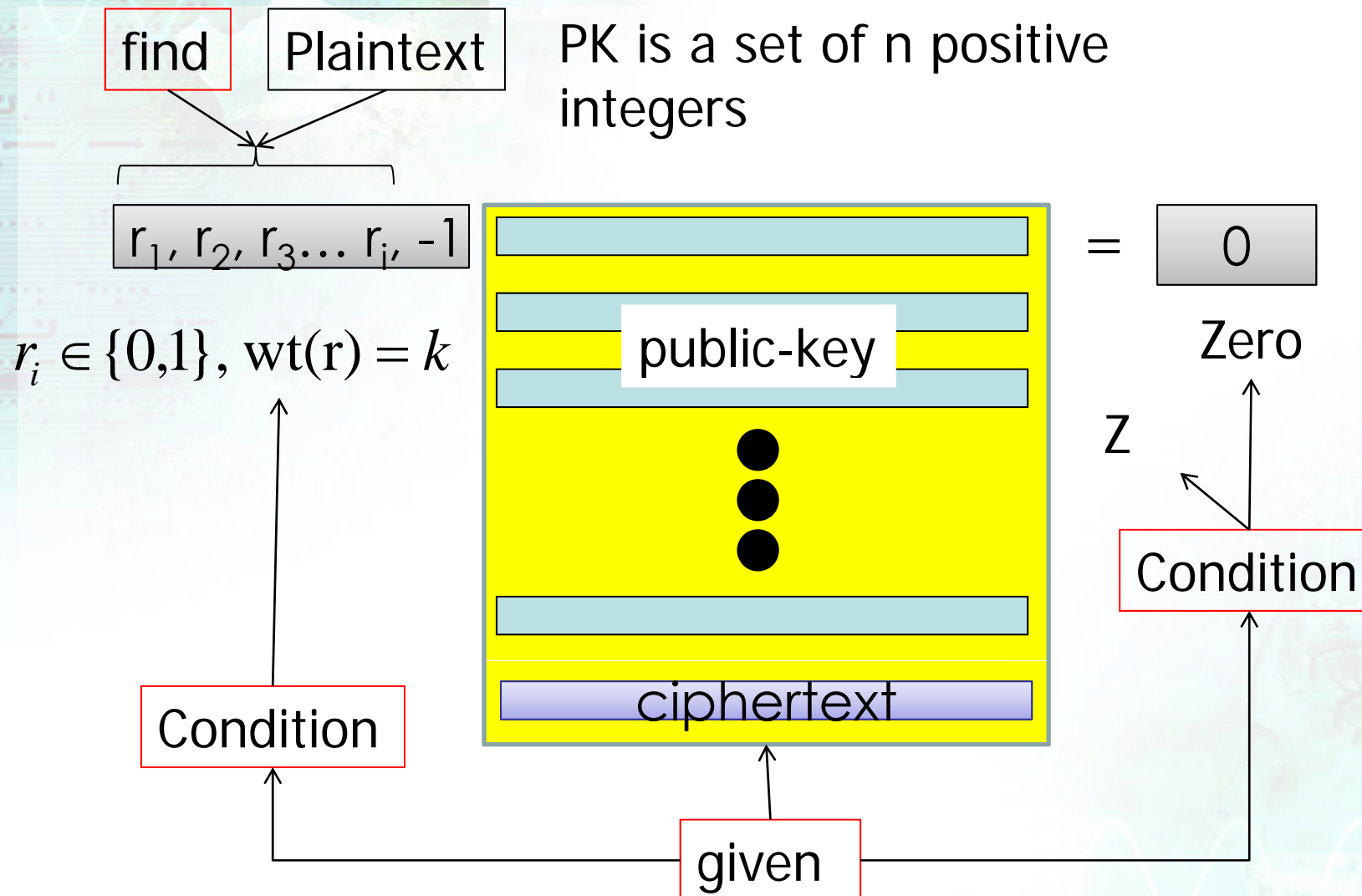
Lattice Based: NTRU



Lattice Based: Regev



Subset Sum Problem: OTU



So far it is not known

- To reduce combinatorial problems to a cyclic problem
- The best attack on combinatorial problem is:
 - Grover's algorithm
- Grover's algorithm can reduce
 - Running time to $\sqrt{\text{states}}$ but it is still **exponential** to its input size

Advantages of Combinatorial Based

■ Quantum Tolerant:

- No polynomial time algorithm is known so far even on quantum computers

■ Arithmetic unit is small:

- for encryption (and signature verification)
- Usually **xors** or additions in a small Field/Ring
- They are highly **parallelizable**
- No heavy multi-precision modular exponentiation

■ **Information Ratio**, i.e.
$$\frac{(\text{Plaintext Size})}{(\text{Ciphertext Size})}$$
 is better

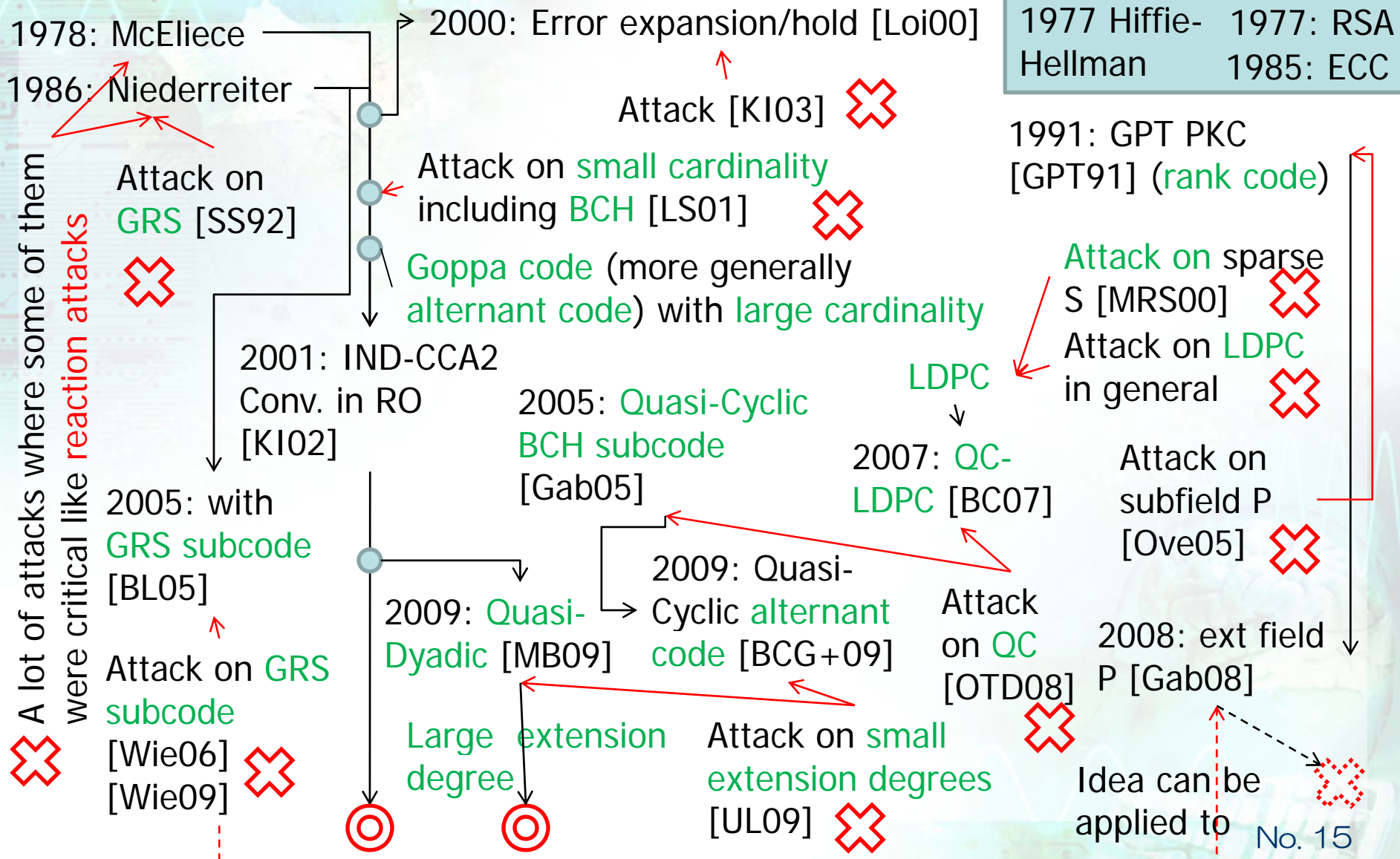
- **Arithmetic unit is smaller:**
- for encryption and signature verification
 - Usually xors

=> **Suitable for low computational power or
Ubiquitous devices**

History of Code-Based PKEs

Code-based

Number Theoretic




How to see the history

■ **Green Letters:** Underlying Liner Code

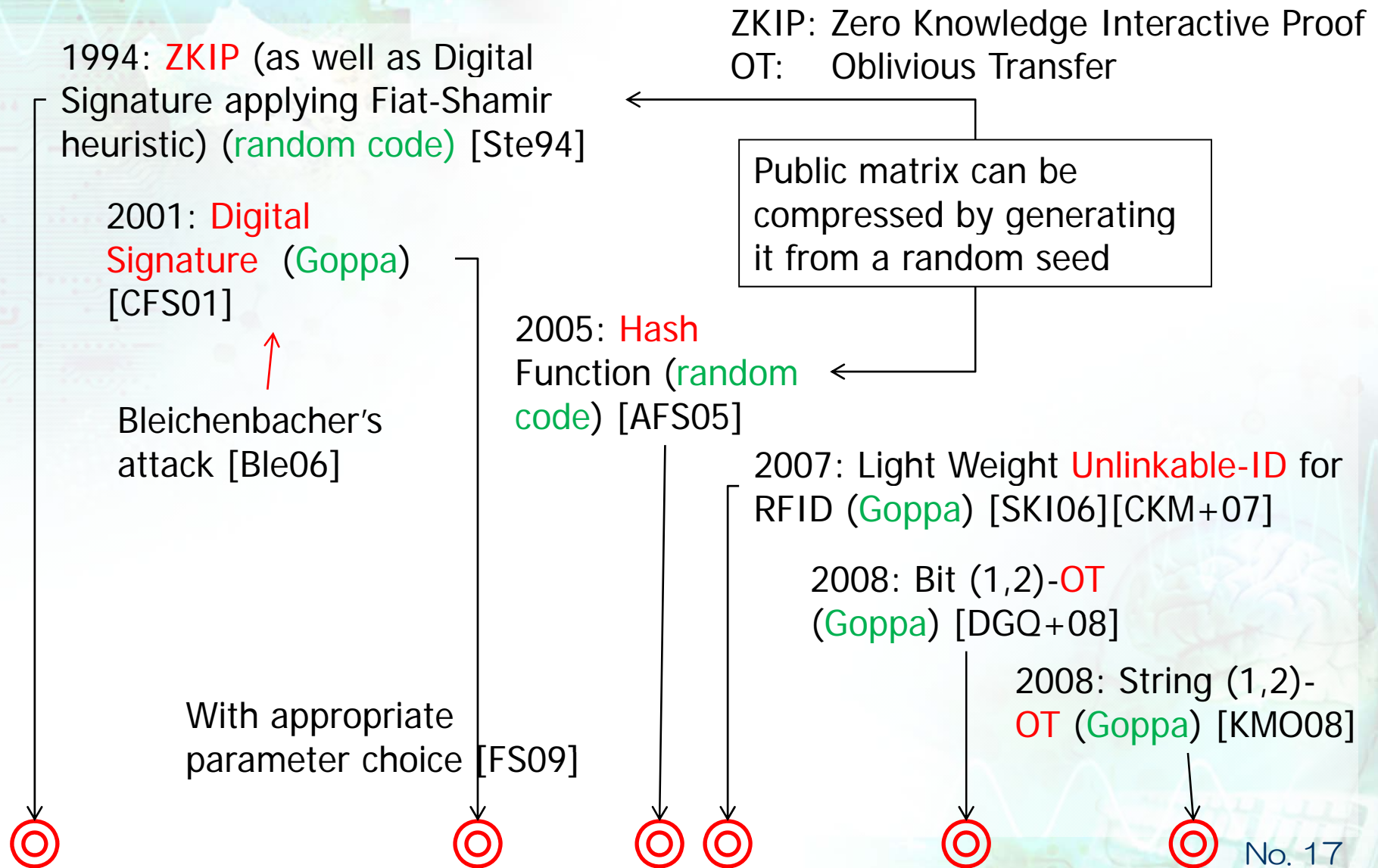
■  : Attack

■  : Already Broken

■  : Not Yet Broken

(as far as I know as of Dec. 2009)

History of Non PKE Code-Based Primitives

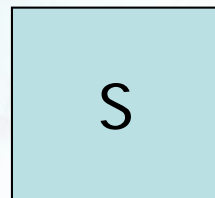


Construction of Code-Based PKCs

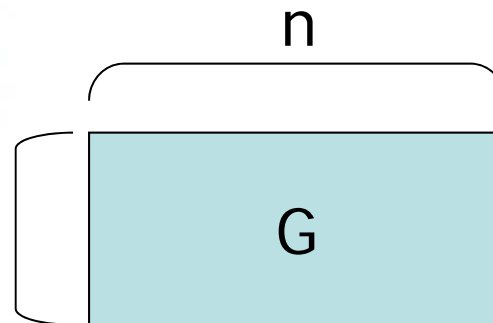
Keys for McEliece PKE

■ Secret key

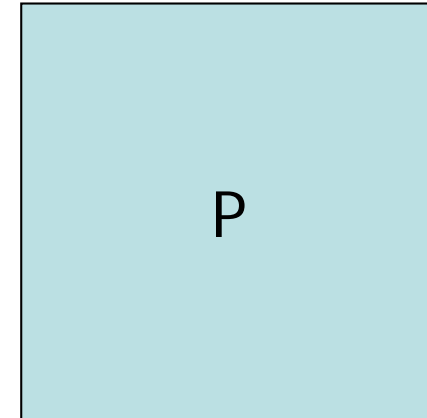
Non-Singular Matrix



k



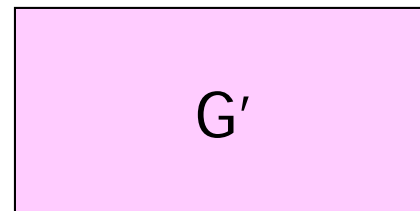
Random Permutation Matrix



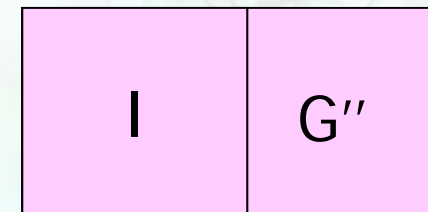
Generator matrix of ECC
 which can correct up to t -error symbols

■ Public key

||



May be
 systematic if an
 appropriate IND-
 CCA2 conversion
 is applied



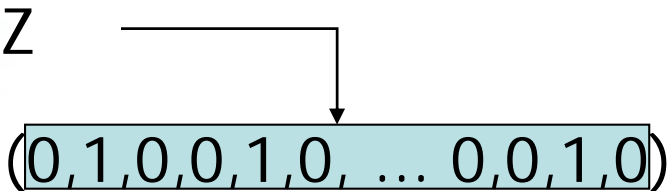
Encryption of Primitive McEliece PKE

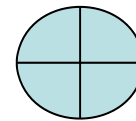
$$C = MG' + Z$$

Plaintext is a k
dimensional vector
over $GF(2)$

M

Random vector with weight t

Z




MG'

C

Ciphertext

G'

Decryption of Primitive McEliece PKE

■ 1. $CP^{-1} = MSG + ZP^{-1}$

C

P^{-1}

■ 2. Correct the t -error bits using error-correction algorithm and obtain MS

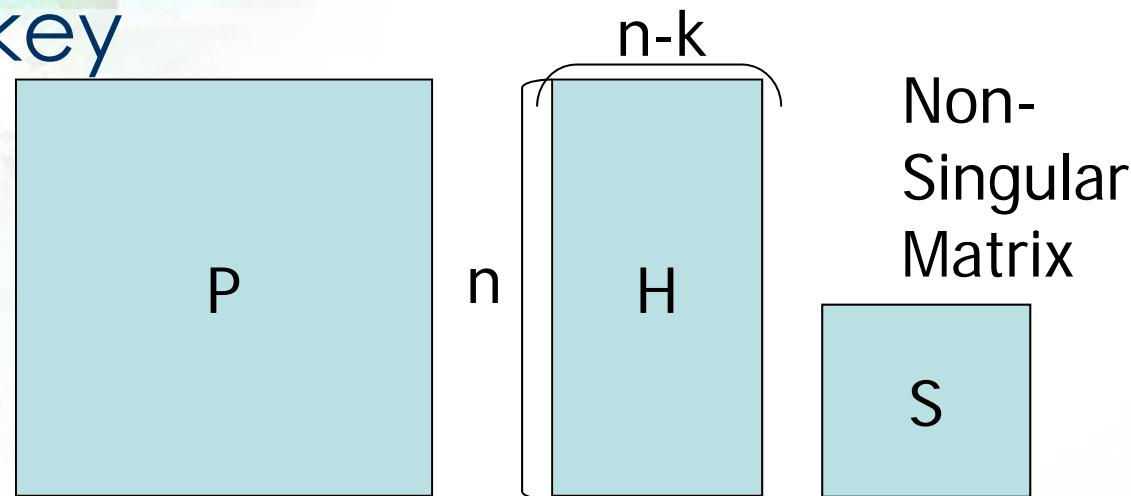
■ 3. $M = MS S^{-1}$

MS S^{-1} = M

Keys for Niederreiter PKE

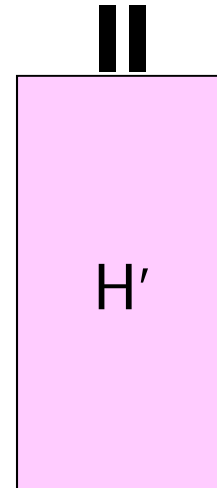
■ Secret key

Random
Permutation
Matrix

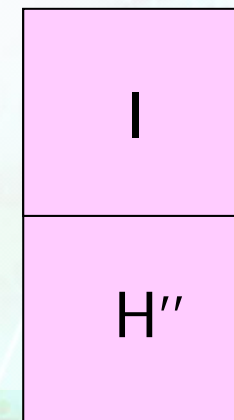


Parity check matrix of ECC
which can correct up to t -error symbols

■ Public key



May be
systematic if an
appropriate IND-
CCA2 conversion
is applied



Encryption of Primitive Niederreiter PKE

■ $C = ZH'$

Plaintext is an n dimensional
vector of weight t

Z

$(0, 1, 0, 0, 1, 0, \dots, 0, 0, 1, 0)$

H'

$=$

C

ZH'

Ciphertext

Decryption of Primitive Niederreiter PKE

■ 1. $CS^{-1} = ZH'S^{-1} = ZPH \quad SS^{-1} = (ZP)H$

C

S^{-1}

■ 2. Correct the t-error bits using error-correction algorithm and obtain ZP

■ 3. $Z = ZP P^{-1}$

ZP

P^{-1}

=

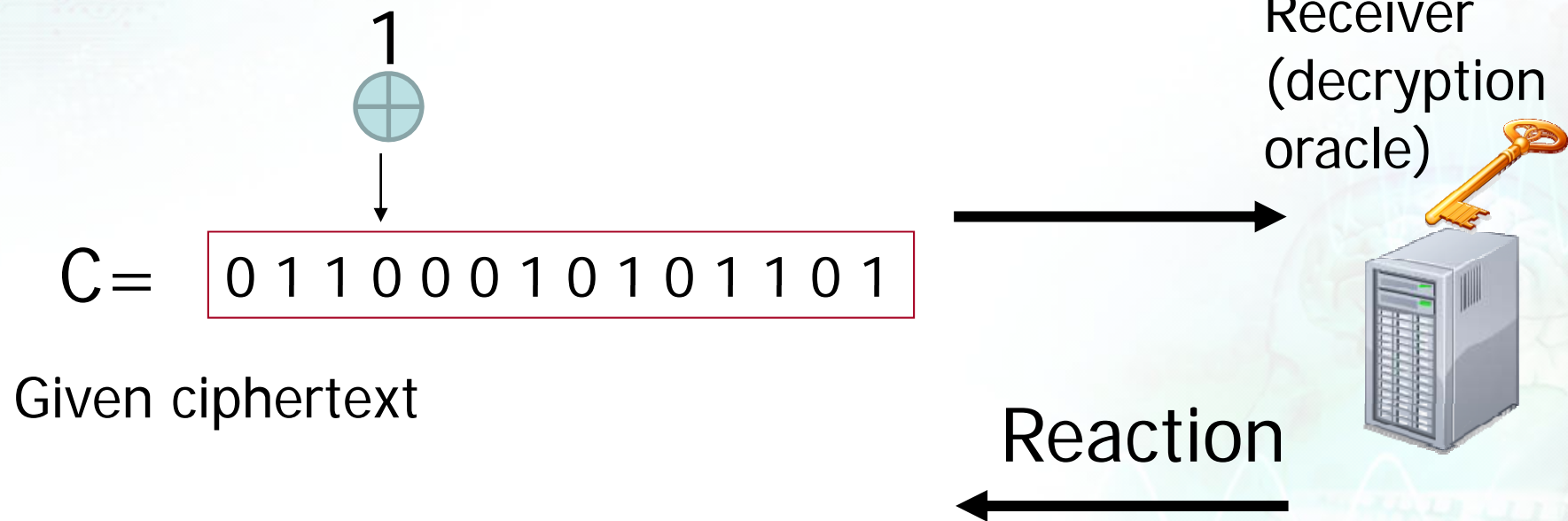
Z

Unfortunately

- Due to the simple structure of a linear code, a lot of practical attacks are known
 - This is the reason why some people still think that code-based PKEs have already been broken
- Ex.)
 - Reaction attack can decrypt a given ciphertext in $O(k)$.

Reaction Attack on McEliece PKC

- Flip one bit of the given ciphertext
- Let the receiver decrypt it
- If its reaction is normal, error is within the error correction bound.



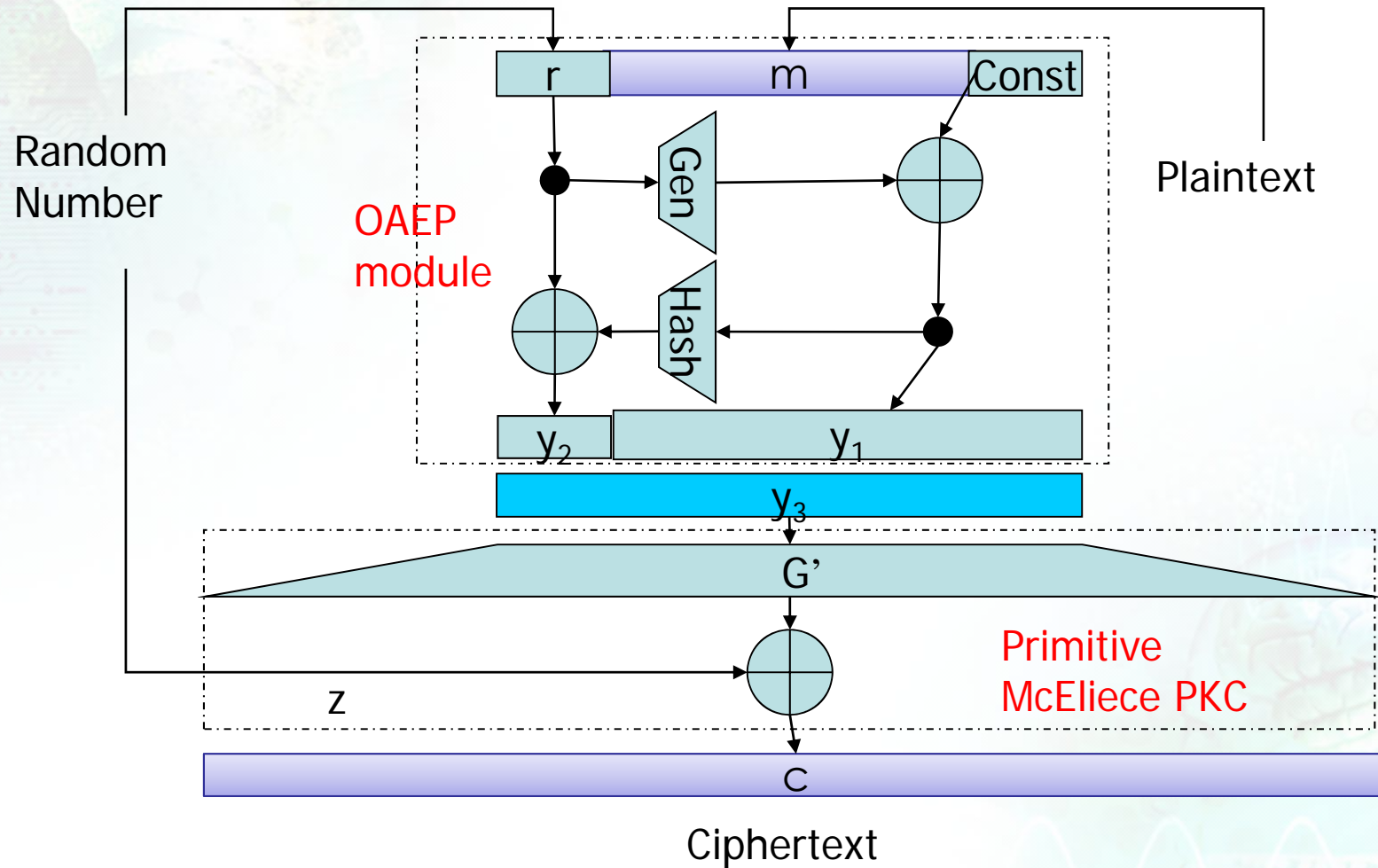
■ Attacks

- Using decryption oracles
- Against indistinguishability or non-malleability

■ Can be prevented applying an “appropriate” conversion scheme

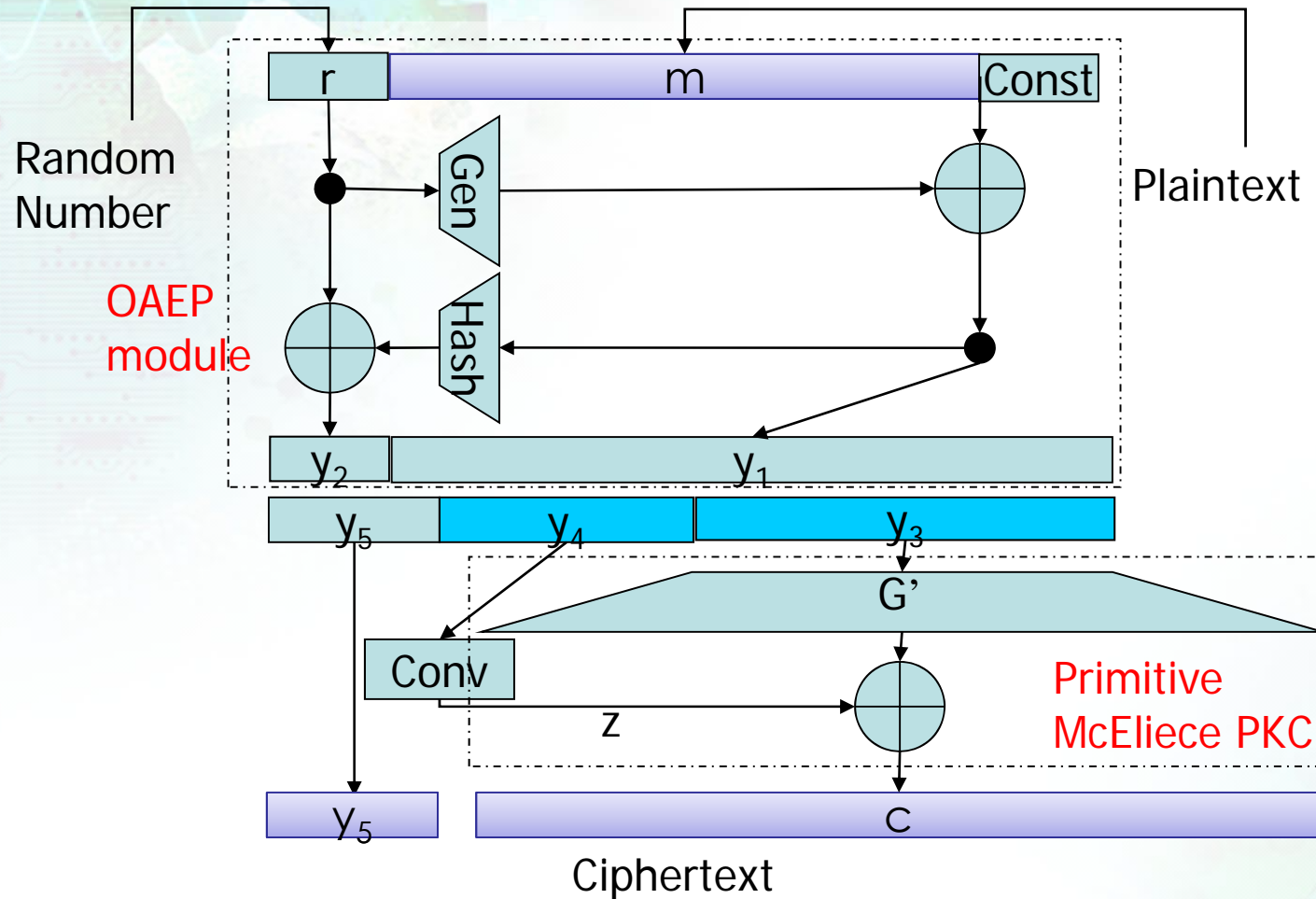
- Naïve application sometimes does not work

Naïve application of OAEP is vulnerable



Slight Modification Makes it Provably Secure and Compact

Encryption

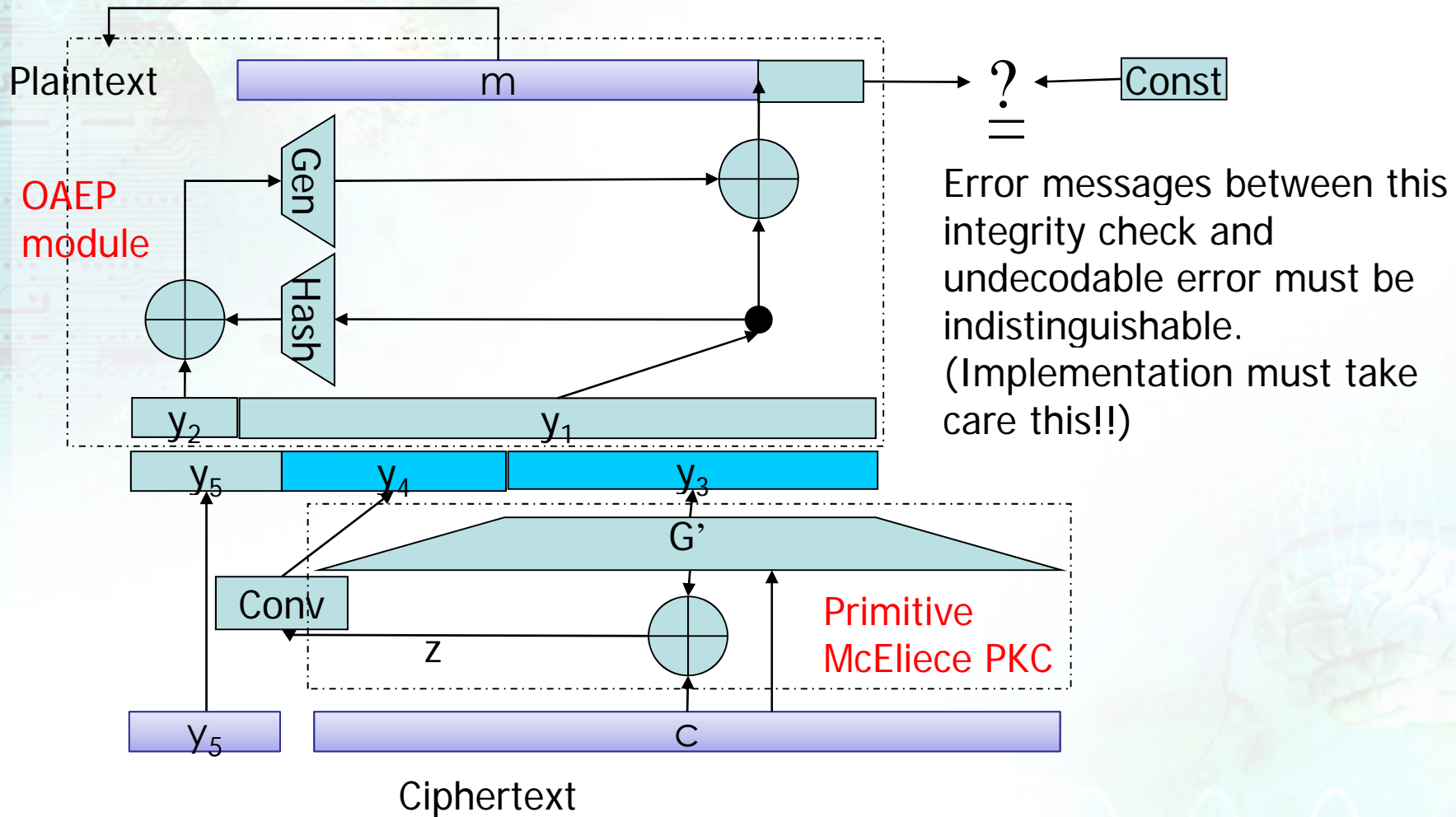


$(Len(y_2) + Len(y_1) \geq Len(y_4) + Len(y_3))$ and

$((Len(y_1) \geq Len(y_3)) \text{ or } (Len(y_1) \geq Len(y_4) \text{ by swapping } y_3 \text{ and } y_4))$

must hold

Decryption of Conversion



In The Case of Niederreiter

- Make the ciphertext most compact and **provably secure** in the RO
 - OAEP+ for short plaintext
 - OAEP++ for long plaintext
- **No proof** but no attack
 - OAEP
 - SAEP+
- **Insecure** (since Niederreiter primitive is position-wise malleable)
 - SAEP

- 2007: IND-CPA in the standard model
[NIK+07]
- 2008: IND-CCA2 in the standard model
[DQN08]

IND: Indistinguishability

CPA: Chosen Plaintext Attack

CCA2: Adaptive Chosen Ciphertext Attack

Secure Constructions are Available

- As long as the primitive McEliece/Niederreiter PKC satisfies **OW-CPA**
- Parameters meeting OW-CPA against most powerful attacks **ISD** and **GBA** are estimated in [FS09]


OW-CPA: One-Wayness against Chosen Plaintext Attack

ISD: Information Set Decoding

GBA: Generalized Birthday Attack

Parameters for PKE

m	t	Binary work factor	PK size (KB)	Plaintext /Ciphertext size	
				McEliece	Niederreiter
11	32	$2^{86.8}$	72.9KB	1696/2048 bits	233/352bits
12	41	$2^{128.5}$	216.5KB	3604/4096 bits	327/492bits


 Public-key size is large compared to Number Theoretic ones.

Parameters for Digital Signature

m	t	Binary work factor	PK size (KB)	Iteration *1	Signature Size *2,*3
22	9	$2^{81.7}$	101,371KB	$2^{18.5}$	198~216.5 bits
15	12	$2^{81.5}$	716.0KB	$2^{28.8}$	180~208.8 bits
14	13	$2^{80.7}$	360.0KB	$2^{32.5}$	182~214.5 bits
13	14	$2^{80.0}$	178.0KB	$2^{36.4}$	182~218.4 bits
15	12	$2^{88.2}$	86.0KB	$2^{40.3}$	180~220.3 bits

*1: affects the signing cost

*2: Signature size depends on how to express the error pattern and it affects the verification speed.

*3: Signature size can be reduced further by removing some error positions while increasing the verification cost further [CFS01]

How to Reduce Public- Key Size

■ Increase the error correction capability

■ Capacity Approaching Codes

■ LDPC, QC-LDPC



■ List Decoding



■ corrects only a couple of more errors for practical parameters while increasing the decoding complexity

■ Compress the public-key

■ Quasi-Cyclic



■ Quasi-Dyadic

■ Small extension degree



■ Large extension degree



Exploits the fact the density of the secret matrix low.

$$\boxed{H'^T} = \boxed{S} \boxed{\begin{array}{l} H^T: \text{Parity Check} \\ \text{Matrix of LDPC} \\ \text{code} \end{array}} \boxed{P}$$

This may be hidden with S and P but can be recovered.

Let

$$\boxed{\begin{array}{l} H^T: \text{Parity Check} \\ \text{Matrix of LDPC} \\ \text{code} \end{array}} = \boxed{H_1} \boxed{H_2}$$

1. Make H' systematic

$$\begin{bmatrix} I & H''^T \end{bmatrix} = \begin{bmatrix} S_1 & H'^T \end{bmatrix}$$

2. Multiply a low density vector h_1 , and check if the density of h_2 is low.

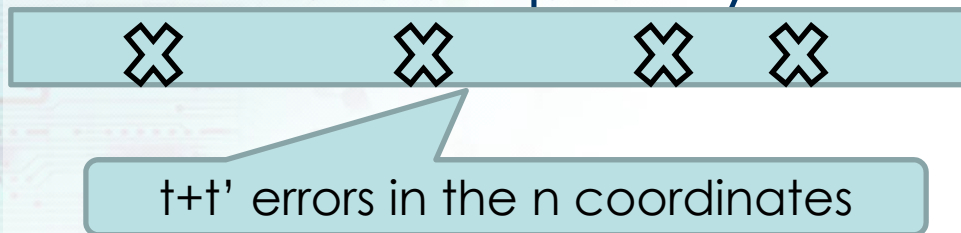
$$\begin{bmatrix} h_1 \\ H_1 \end{bmatrix} \begin{bmatrix} I & H''^T \end{bmatrix} \stackrel{?}{=} \begin{bmatrix} h_1 & h_2 \\ H_1 & H_2 \end{bmatrix}$$

Since # of candidates $C(n-k, w)$ where w is the Hamming weight of h_1 (and also the possibility of being another low density matrix is negligible), H can be recovered.

If one uses this direction, they must find a good code of Middle Density (we call MDPC).

Exhaustive Search

- Correct any t' more errors but with the complexity



$$O \left(\frac{\binom{n}{t'}}{\binom{t+t'}{t'}} \right)$$

$$\left(\frac{n-t'+1}{t+1} \right)^{t'} \geq \frac{\binom{n}{t'}}{\binom{t+t'}{t'}} \geq \left(\frac{n}{t+t'} \right)^{t'}$$

Bernstein's List Decoding [Ber08]

- t' more errors in poly time where

$$t' = n - \sqrt{n(n-2t-2)} - t$$

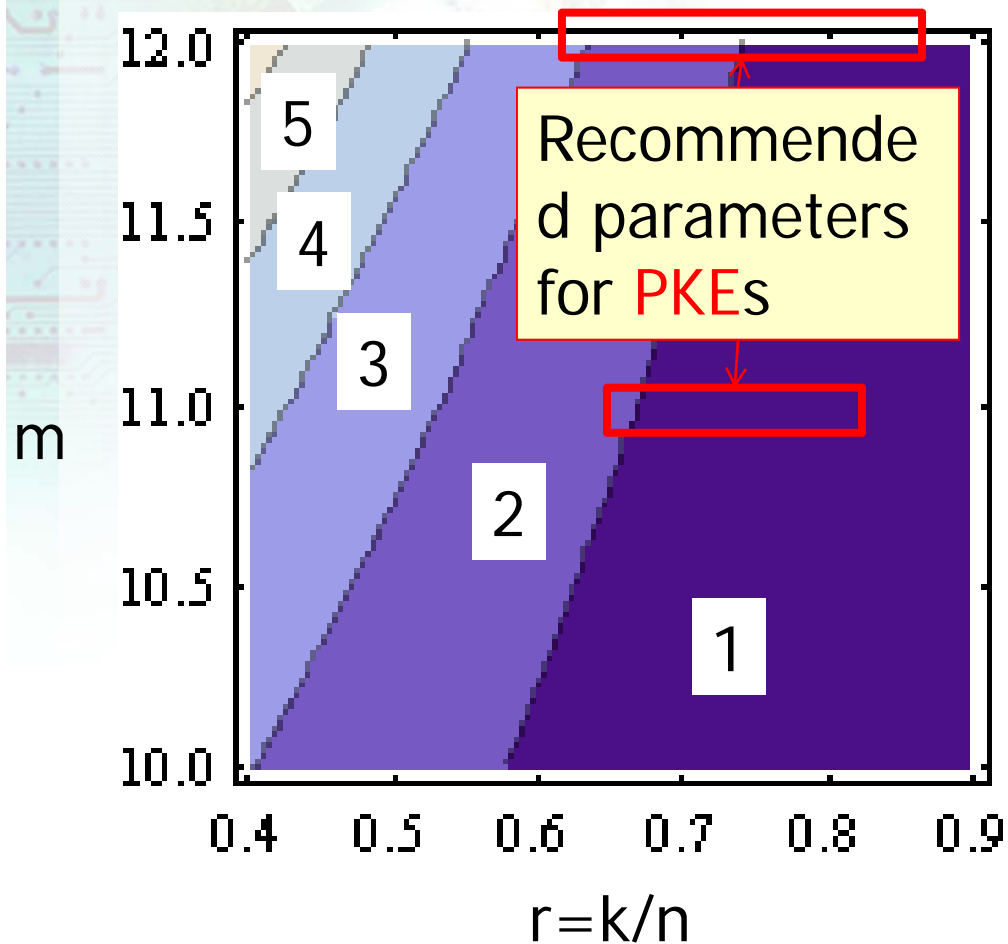
$$= 2^m - \frac{2^m(1-r)}{m} - \sqrt{2^m \left(2^m - 2 \left(\frac{2^m(1-r)}{m} \right) - 2 \right)}$$

$$n = 2^m$$

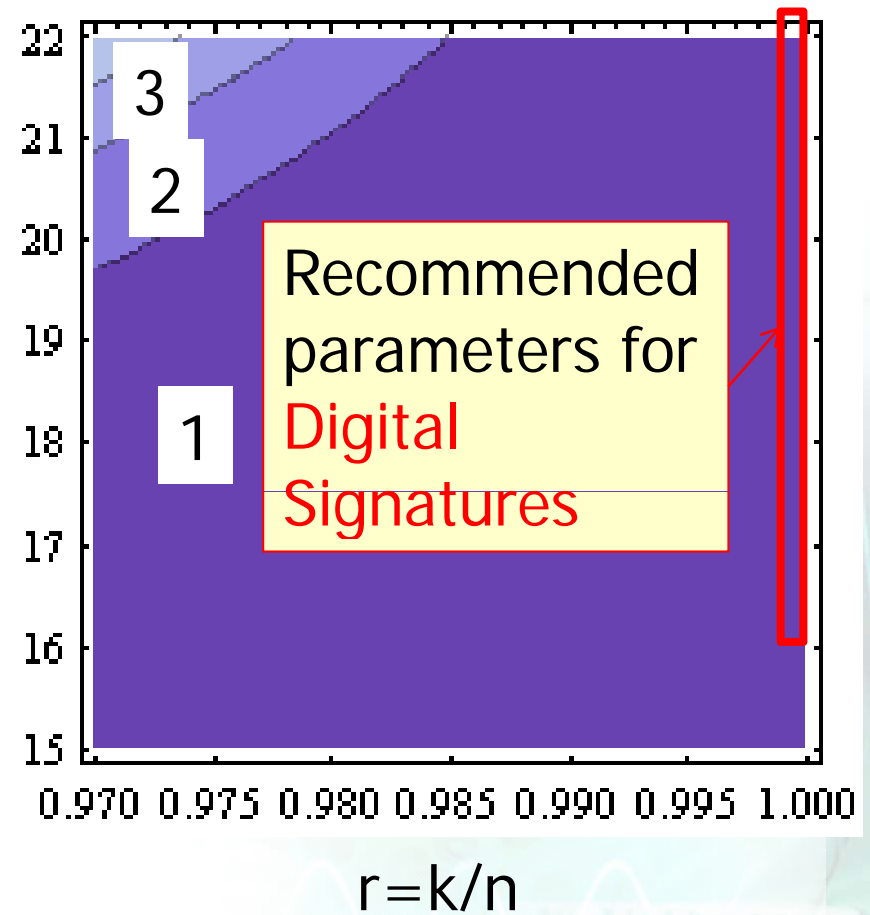
$$t = \frac{(n-k)}{m} = \frac{2^m(1-r)}{m}$$

of extra error-correctable bits

Extra error correction bits



Extra error correction bits



How to Reduce Public- Key Size

■ Increase the error correction capability

■ Capacity Approaching Codes

■ LDPC, QC-LDPC



■ List Decoding



■ corrects only a couple of more errors for practical parameters while increasing the decoding complexity

■ Compress the public-key

■ Quasi-Cyclic



■ Quasi-Dyadic

■ Small extension degree



■ Large extension degree

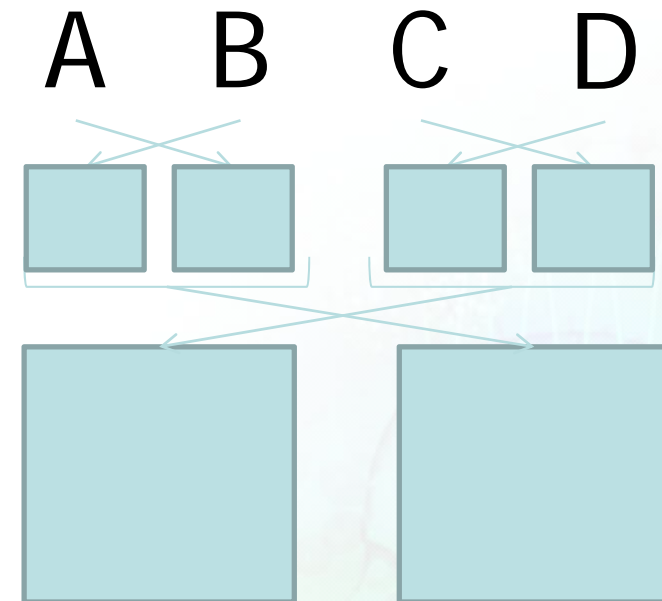


Dyadic Matrix

$$\begin{bmatrix} \alpha & \beta \\ \beta & \alpha \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} A & B \\ B & A \end{bmatrix} & \begin{bmatrix} C & D \\ D & C \end{bmatrix} \\ \begin{bmatrix} C & D \\ D & C \end{bmatrix} & \begin{bmatrix} A & B \\ B & A \end{bmatrix} \end{bmatrix}$$

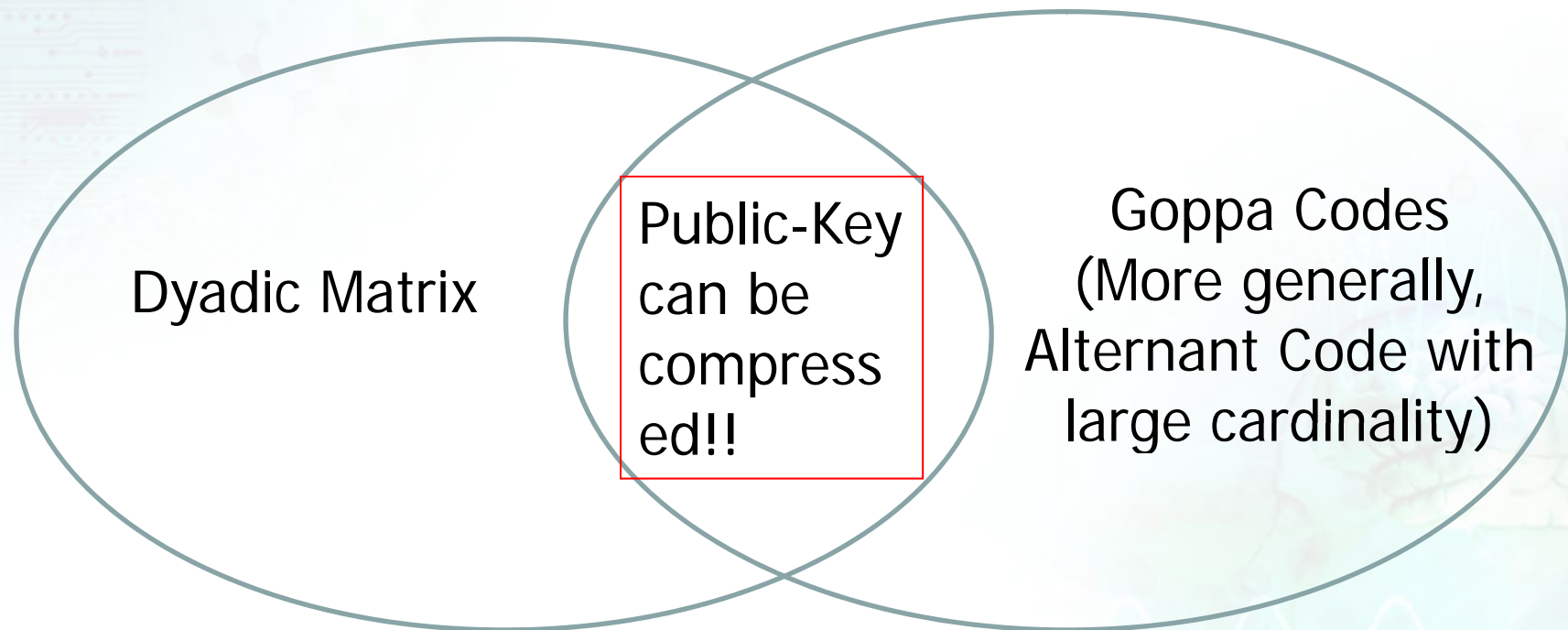
Advantage of Dyadic Matrix

$$\left[\begin{array}{cc} \begin{bmatrix} A & B \end{bmatrix} \\ \begin{bmatrix} B & A \end{bmatrix} \\ \begin{bmatrix} C & D \end{bmatrix} \\ \begin{bmatrix} D & C \end{bmatrix} \end{array} \right] \left[\begin{array}{cc} \begin{bmatrix} C & D \end{bmatrix} \\ \begin{bmatrix} D & C \end{bmatrix} \\ \begin{bmatrix} A & B \end{bmatrix} \\ \begin{bmatrix} B & A \end{bmatrix} \end{array} \right]$$



Dyadic Matrix and Goppa Codes

have an intersection over the extension field of $GF(2)$



Goppa Codes and Cauchy Matrix

■ Binary Goppa code is the set of all

$$c = (c_0 \quad c_1 \quad \cdots \quad c_{n-1}) \in GF(2)^n \quad \text{s.t.}$$

$$S_e(X) \equiv \sum_{i=0}^{n-1} \frac{c_i}{X - L_i} \equiv 0 \pmod{g(X)}$$

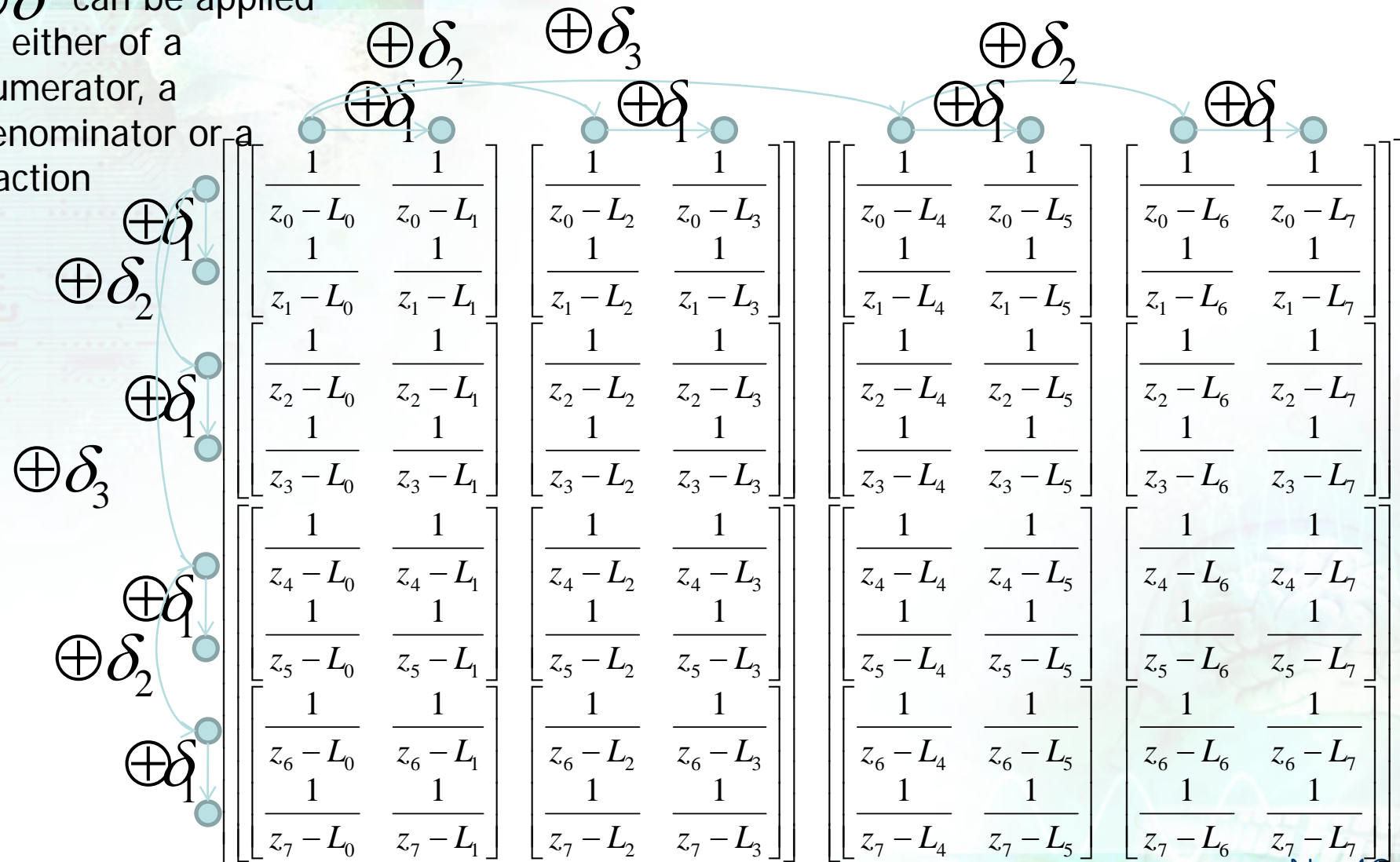
If $g(X) = \prod_{i=0}^{t-1} (z_i - x)$ where z_i and L_j are distinct
it can be written

$$(cH)^T = \begin{bmatrix} \frac{1}{z_0 - L_0} & \frac{1}{z_0 - L_1} & \cdots & \frac{1}{z_0 - L_{n-1}} \\ \frac{1}{z_1 - L_0} & \frac{1}{z_1 - L_1} & \cdots & \frac{1}{z_1 - L_{n-1}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{z_{t-1} - L_0} & \frac{1}{z_{t-1} - L_1} & \cdots & \frac{1}{z_{t-1} - L_{n-1}} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-1} \end{bmatrix} = 0$$

To Make It Dyadic

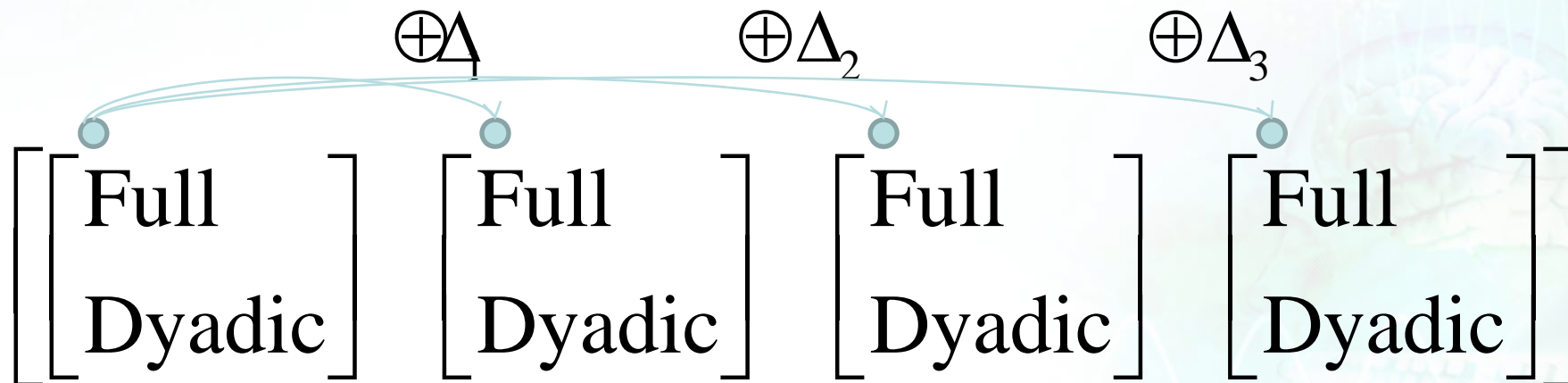
(more generic than [MB09])

$\oplus \delta$ can be applied
 to either of a
 numerator, a
 denominator or a
 fraction



[MB09] generates a large full Dyadic block then picks up

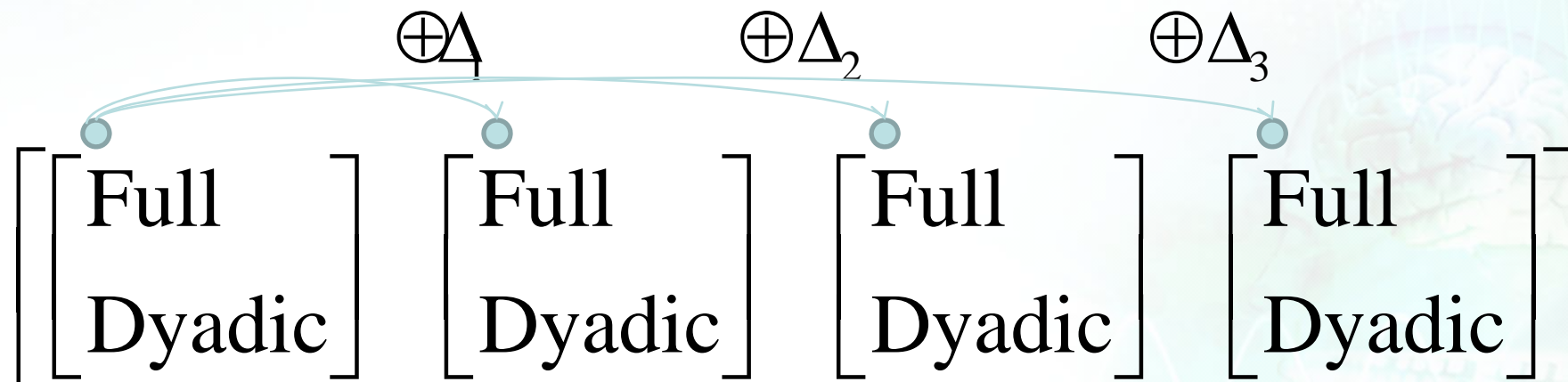
- We propose to generate a small full Dyadic block and then generate other full Dyadic blocks by introducing $\oplus\Delta$ (we call it outer delta)
 - Whereas we call $\oplus\delta$ inner delta (since it defines the inner structure of a full Dyadic block)



- Can generate the txn Dyadic matrix more flexibly,
 - $N=2^m$ can be closer to $n+t$
- Can remove the block-wise permutation and removal in key generation
 - Since they are already included by the parameter choice of $\oplus \Delta$

Remark

- Block-Wise Permutation is the same as changing Δ_j 's appropriately.
- Removal of one block is the same as reducing the number of blocks by one and then changing Δ_j 's appropriately



By applying $\oplus \delta$ and $\oplus \Delta$
to the denominators

■ txn Dyadic matrix can be obtained with

$$z_1 = z_0 \oplus \delta_1 \quad z_2 = z_0 \oplus \delta_2 \quad z_3 = z_2 \oplus \delta_1 = z_0 \oplus \delta_1 \oplus \delta_2$$

$$L_1 = L_0 \oplus \delta_1 \quad L_2 = L_0 \oplus \delta_2 \quad L_3 = L_2 \oplus \delta_2 = L_0 \oplus \delta_1 \oplus \delta_2$$

$$L_t = L_0 \oplus \Delta_1 \quad L_{j \times t} = L_0 \oplus \Delta_j$$

■ where

$$z_0, L_0, \delta_1, \delta_2 \cdots \delta_{\log_2 t}, \Delta_1, \Delta_2 \cdots \Delta_{(n/t)-1} \in GF(2^m)$$

are chosen at random while making all
the z_i for $0 \leq i < t$ and L_j for $0 \leq j < n$
distinct.

Shuffle While Keeping Dyadic Structure

■ With Column-Block-Wise Scalar Multiplication

$$\begin{bmatrix} \text{Full} \\ \text{Dyadic} \end{bmatrix} \begin{bmatrix} \text{Full} \\ \text{Dyadic} \end{bmatrix} \cdots \begin{bmatrix} \text{Full} \\ \text{Dyadic} \end{bmatrix} \begin{bmatrix} [1] & & & 0 \\ & [p_1] & & \\ & & \ddots & \\ 0 & & & [p_{b-1}] \end{bmatrix}$$

where $p_i \in GF(q^m) \setminus \{0\}$ and $b = \frac{n}{t}$

$GF(q^m)$ to $GF(q)$

$A, \gamma_i \in GF(q^m)$ $\{\gamma_1, \gamma_2 \cdots \gamma_{m-1}\}$ is the
 $a_i \in GF(q)$ dual basis of $GF(q^m)$

$$A = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{m-1} \end{bmatrix} = \begin{bmatrix} tr(\gamma_0 A) \\ tr(\gamma_1 A) \\ \vdots \\ tr(\gamma_{m-1} A) \end{bmatrix}$$

$$tr(A) = A + A^q + A^{q^2} + \cdots + A^{q^{m-1}}$$

$GF(q^m)$ to $GF(q)$

$$\begin{bmatrix} A & B \\ B & A \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} a_0 \\ \vdots \\ a_{m-1} \end{bmatrix} & \begin{bmatrix} b_0 \\ \vdots \\ b_{m-1} \end{bmatrix} \\ \begin{bmatrix} b_0 \\ \vdots \\ b_{m-1} \end{bmatrix} & \begin{bmatrix} a_0 \\ \vdots \\ a_{m-1} \end{bmatrix} \end{bmatrix} \xrightarrow{\quad} \begin{bmatrix} \begin{bmatrix} a_0 & b_0 \\ b_0 & a_0 \end{bmatrix} \\ \vdots \\ \begin{bmatrix} a_{m-1} & b_{m-1} \\ b_{m-1} & a_{m-1} \end{bmatrix} \end{bmatrix}$$

Shuffle While Keeping Dyadic Structure

■ Multiplication of a Random Dyadic Matrix

$$\begin{bmatrix} \begin{bmatrix} \text{Random} \\ \text{Dyadic} \end{bmatrix} & \begin{bmatrix} \text{Random} \\ \text{Dyadic} \end{bmatrix} \\ \begin{bmatrix} \text{Random} \\ \text{Dyadic} \end{bmatrix} & \begin{bmatrix} \text{Random} \\ \text{Dyadic} \end{bmatrix} \end{bmatrix} \begin{bmatrix} \begin{bmatrix} \text{Full} \\ \text{Dyadic} \end{bmatrix} & \begin{bmatrix} \text{Full} \\ \text{Dyadic} \end{bmatrix} \\ \begin{bmatrix} \text{Full} \\ \text{Dyadic} \end{bmatrix} & \begin{bmatrix} \text{Full} \\ \text{Dyadic} \end{bmatrix} \end{bmatrix} \\ \begin{bmatrix} \begin{bmatrix} \text{Full} \\ \text{Dyadic} \end{bmatrix} & \begin{bmatrix} \text{Full} \\ \text{Dyadic} \end{bmatrix} \\ \begin{bmatrix} \text{Full} \\ \text{Dyadic} \end{bmatrix} & \begin{bmatrix} \text{Full} \\ \text{Dyadic} \end{bmatrix} \end{bmatrix} \\ \begin{bmatrix} \begin{bmatrix} \text{Full} \\ \text{Dyadic} \end{bmatrix} & \begin{bmatrix} \text{Full} \\ \text{Dyadic} \end{bmatrix} \\ \begin{bmatrix} \text{Full} \\ \text{Dyadic} \end{bmatrix} & \begin{bmatrix} \text{Full} \\ \text{Dyadic} \end{bmatrix} \end{bmatrix} \\ = (HS)^T$$

Note: Dyadic X Dyadic = Dyadic

$$\begin{bmatrix} A & B \\ B & A \end{bmatrix} \begin{bmatrix} C & D \\ D & C \end{bmatrix} = \begin{bmatrix} AC + BD & AD + BC \\ BC + AD & BD + AC \end{bmatrix}$$

Attack on Quasi Dyadic [UL09]

- Exploits the fact that each column of each low of H'^T is $\text{tr}(f(L_i))$ where $f()$ and L_i are unknown
- If one can make $f(x)=\text{const}$, $\text{tr}(f(L_i))$ becomes independent of L_i and all the coordinates in the same block of the low take the same value.
- On the contrary, by choosing r_i s.t.

$$r_i H'^T = ([h_{i,0}, \dots, h_{i,0}], [h_{i,1}, \dots, h_{i,1}], \dots, [h_{i,b-1}, \dots, h_{i,b-1}])$$

$f(x)$ can be const with high probability

Attack on Quasi Dyadic [UL09]

$$r_i H^T = ([h_{i,0}, \dots, h_{i,0}], [h_{i,1}, \dots, h_{i,1}], \dots, [h_{i,b-1}, \dots, h_{i,b-1}])$$

These constants are linear combinations
of $\text{tr}(\text{constant})$

$$\begin{bmatrix} r_0 \\ r_1 \\ \vdots \\ r_{m-1} \end{bmatrix} H^T = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} =$$

$\begin{bmatrix} h_{0,0} \\ h_{1,0} \\ \vdots \\ h_{m-1,0} \end{bmatrix}$

$\begin{bmatrix} \beta_{0,1} & \dots & \beta_{0,m-1} \\ \beta_{1,1} & \dots & \beta_{1,m-1} \\ \vdots & \ddots & \vdots \\ \beta_{m-1,1} & \dots & \beta_{m-1,m-1} \end{bmatrix}$

$\begin{bmatrix} [\text{tr}(\gamma_0 1), \dots] \\ [\text{tr}(\gamma_1 1), \dots] \\ \vdots \\ [\text{tr}(\gamma_{m-1} 1), \dots] \end{bmatrix}$

$\begin{bmatrix} [\text{tr}(\gamma_0 p_1), \dots] & \dots \\ [\text{tr}(\gamma_1 p_1), \dots] & \dots \\ \vdots & \dots \\ [\text{tr}(\gamma_{m-1} p_1), \dots] & \dots \end{bmatrix}$

$m(m-1)$ unknown in $\text{GF}(q)$
so the uncertainty of $(p_1 \text{ to } p_{m-1})$ is $m(m-1)\log_2 q$

No. 56

Countermeasure Against The Attack on Quasi Dyadic

- $m(m-1)$ unknown in $GF(q)$ and hence the uncertainty of $(p_1 \text{ to } p_{m-1})$ is $m(m-1)\log_2 q$ bits

- E.g.)

For $q=2$, by making $m \geq 10$, the uncertainty becomes ≥ 90 bits and this attack can be avoided

(though this attack might be improved in future)

Parameters for PKE

m	t	Binary work factor	PK size (KB)	Plaintext /Ciphertext size	
				McEliece	Niederreiter
11	32	$2^{86.8}$	72.9KB	1696/2048 bits	233/352bits
12	41	$2^{128.5}$	216.5KB	3604/4096 bits	327/492bits

↓ QD (Parameters are not optimized)

m	t	n	p	l	Binary work factor	PK size (KB)	Plaintext/Cip hertext Size
							QD
16	64	2560	1	12	$2^{91.3}$	3.0KB	427/1024 bits
13	64	2048	2	16	$2^{90.2}$	1.9KB	406/832 bits
12	128	2400	1	8	$2^{90.7}$	1.3KB	716/1536 bits

Parameters for Digital Signature

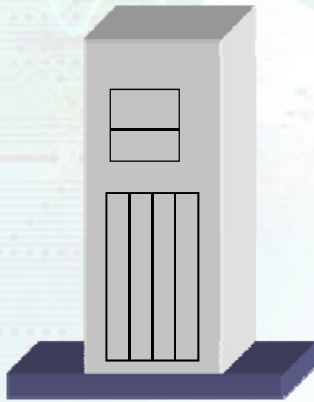
m	t	BWF	PK size (KB)	Iteration *1	Signature Size *2,*3
14	13	$2^{80.7}$	360.0KB	$2^{32.5}$	182~214.5 bits
13	14	$2^{80.0}$	178.0KB	$2^{36.4}$	182~218.4 bits
15	12	$2^{88.2}$	86.0KB	$2^{40.3}$	180~220.3 bits

↓ QD (Parameters are not optimized)

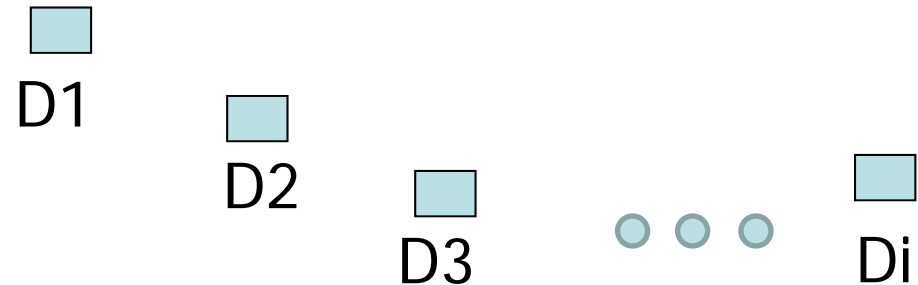
m	t	n	Binary work factor	PK size (KB)	Iteration *1	Signature Size *2,*3
15	12	26528	$2^{84.0}$	48.2KB	$2^{32.5}$	180~212.5 bits
14	13	13316	$2^{83.4}$	22.4KB	$2^{36.4}$	182~218.4 bits
13	14	6736	$2^{82.9}$	10.4KB	$2^{40.3}$	182~222.3 bits

Suitable Applications for Code- Based PKCs

Code-Based PKCs Fit With Heterogeneous Network/Applications



One side has high
computational power



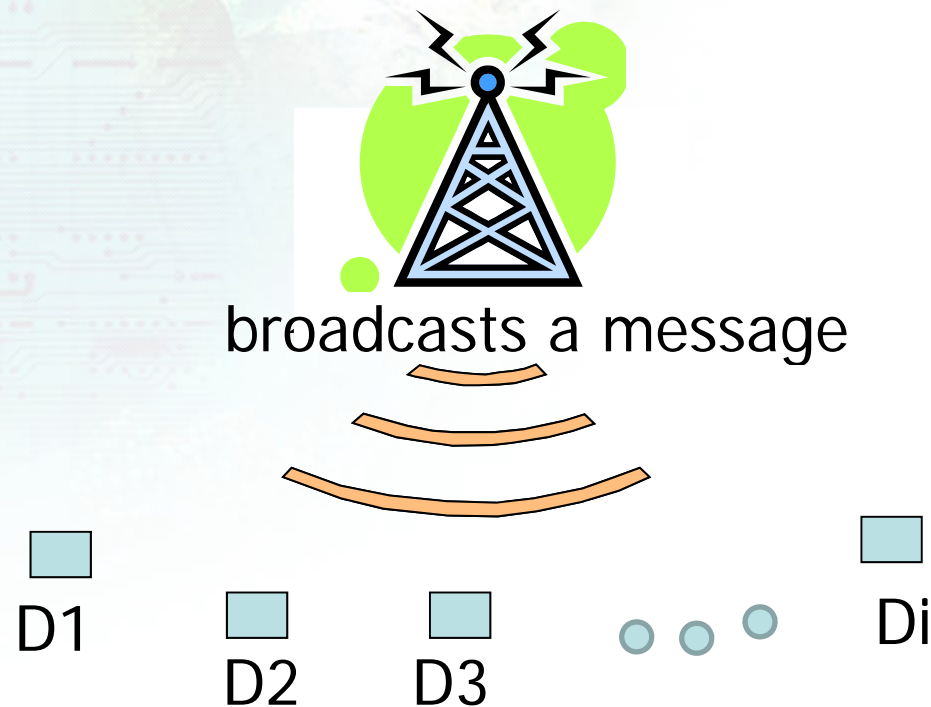
A lot of low cost and low
computational power devices,
such as RFIDs, sensors and
SCADA devices

Since

- Encryption and signature verification consist mostly of **xors** and they are **highly parallelizable**.
- They do not require **heavy multi precision modular exponentiations**

Broadcast Authentication

One (powerful) sender



Applications include:

- Transmission of **mission critical commands** and data to sensors and SCADA devices.
- Issue of **disaster warning**

a lot of (lightweight) receivers receive it
and check its **authenticity and data integrity**.

In emergency applications, low latency is crucial



Especially ➡

Against

- Earthquake
- Tsunami
- Flood
- Tornado
- Thunderbolt
- Fire

...



Disaster warning



D1



D2



D3

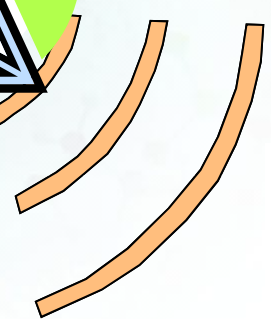
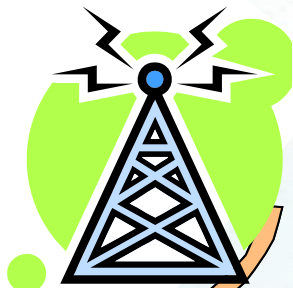


Di



Such warning receivers may be deployed

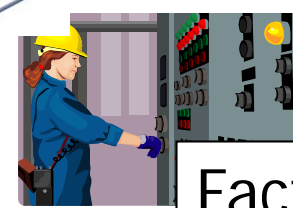
at home, critical infrastructures, and then
take appropriate actions if a warning is
verified.



hospitals



(nuclear) power plants

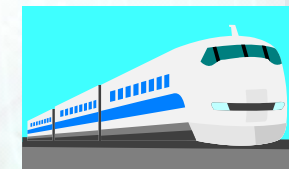


Factories



homes

- Turn off gases
- Unlock and open doors



- Stop the lines
- Slow down the trains

In some cases, a few seconds are enough to mitigate serious damages, and hence delay is should be minimized while maintaining adequate security level.

Comparison Among Solutions

	MAC with one master key	MAC with pair- wise keys	TESLA (hash- chain and delayed auth)	Digital Signature	
				Conventio nal (RSA, DSA, ECDSA)	
Authenticity and Data Integrity	X^{*1}	O	O	O	
Computational Cost	O	O	O	X	
Delay	O	X^{*2}	X^{*3}	X	

*1: Crack of one device breaks it.

*2: Sender must broadcast a lot of MACs and each device must wait until his MAC is received.

*3: Verification key is released in the next time slot.

TESLA: Timed Efficient Stream Loss-tolerant Authentication

Comparison Among Solutions

	MAC with one master key	MAC with pair- wise keys	TESLA (hash- chain and delayed auth)	Digital Signature	
				Conventio nal (RSA, DSA, ECDSA)	Code- Based
Authenticity and Data Integrity	X^{*1}	O	O	O	O
Computational Cost	O	O	O	X	O
Delay	O	X^{*2}	X^{*3}	X	O

*1: Crack of one device breaks it.

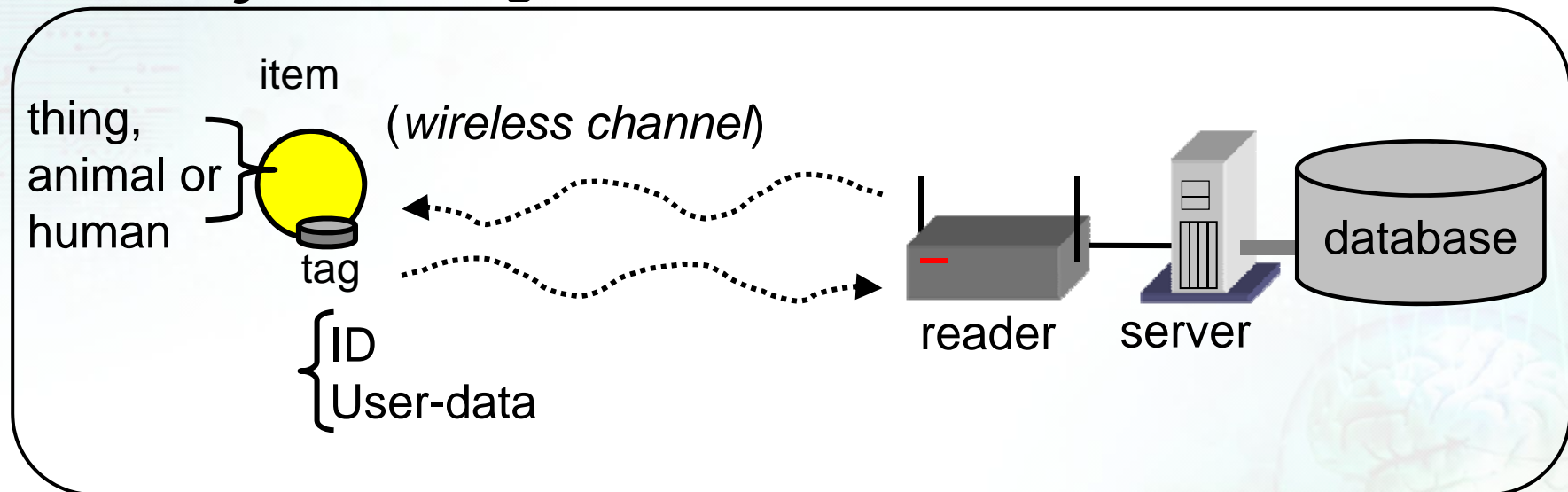
*2: Sender must broadcast a lot of MACs and each device must wait until his MAC is received.

*3: Verification key is released in the next time slot.

TESLA: Timed Efficient Stream Loss-tolerant Authentication

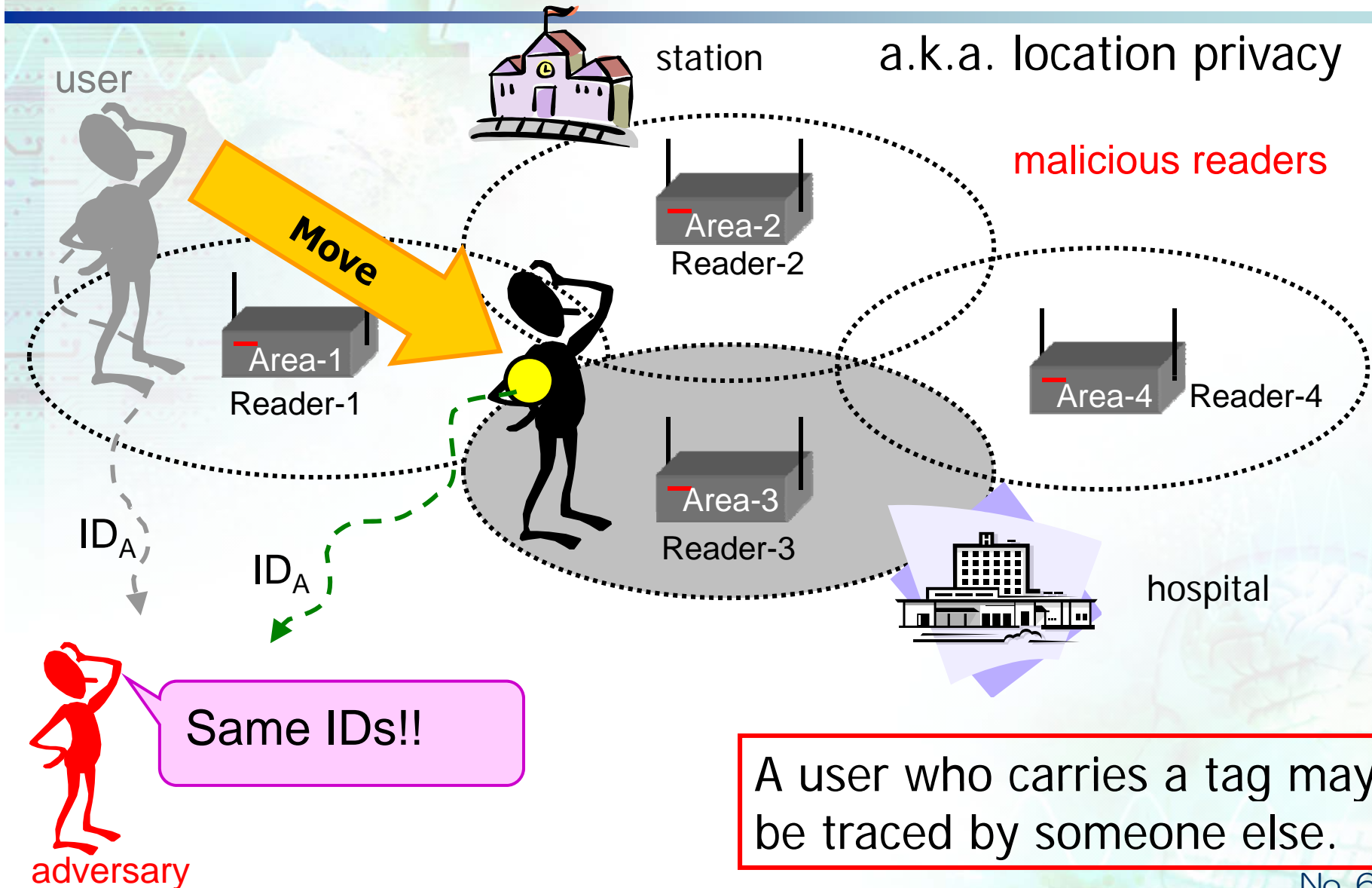
RFID: a tag used for **reading/writing** ID (and information) via **wireless communication**.

RFID system in general:



Applications: management of items, e.g. in **supply chain**

RFID (for long distance) requires privacy protection mechanism



Solutions Can Be Divided Into

■ Tag disabling solutions

- Manually removal or destruction
- Kill command

■ Temporally tag- disabling solutions

- Faraday cage
- Access password
- Hash lock
- Blocker tag
- Mode switch

■ Tag enabling solutions

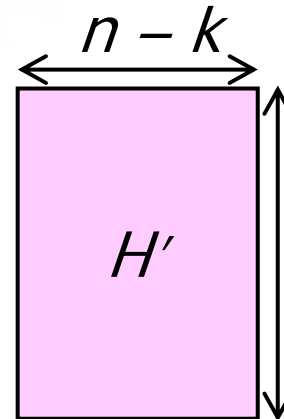
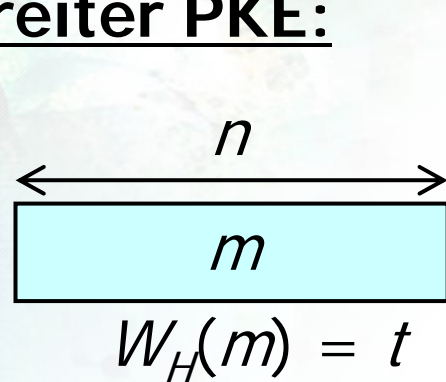
- Randomized hash lock [WSRE03]
- HB+ [JW05] and its variants
- **Code-Based Unlinkable-ID** [SKI06] [CKM+07]

Tag enabling solutions:
Enable RFID functionalities
while providing
unlinkability of IDs

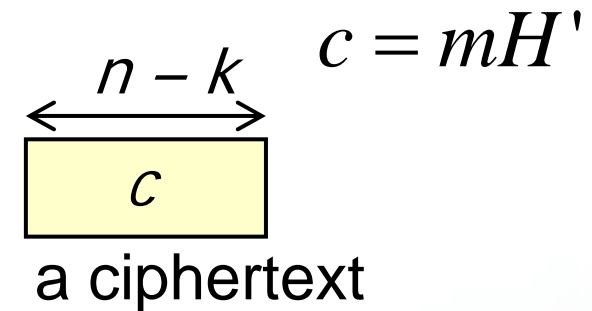
Comparison

	Exhaustive search of IDs at the server	Unlinkability
Randomized hash lock, HB+ and its variants	Necessary (Tag identification cost depends on # of tags)	○
Code-Based Unlinkable-ID	Unnecessary (Tag identification cost is independent of # of tags)	○

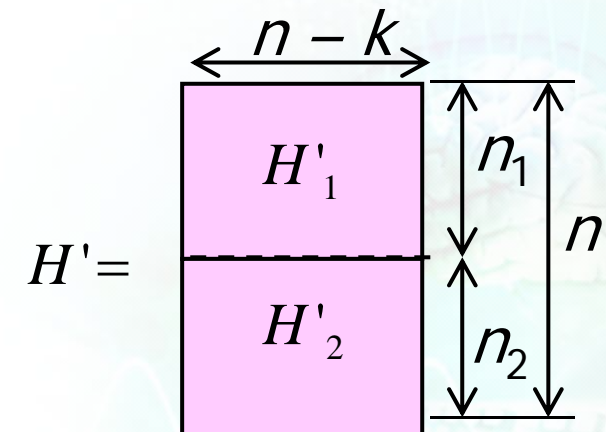
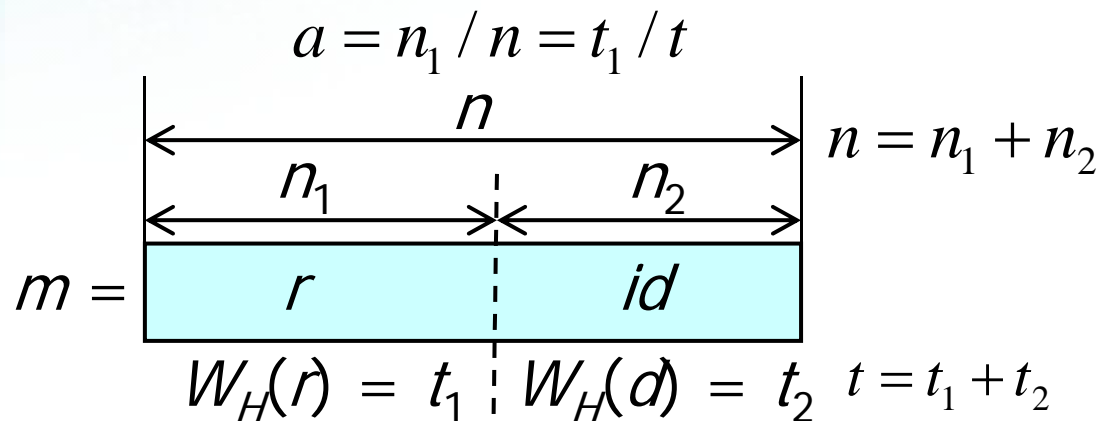
Niederreiter PKE:



($W_H()$: Hamming weight)



Code-Based Unlinkable-ID:



(random num.) (ID)

Code-Based Unlinkable-ID

(2/3)

$$\boxed{c} = \underbrace{\boxed{r} \times \boxed{H'_1}}_{\boxed{c_1}} \oplus \underbrace{\boxed{id} \times \boxed{H'_2}}_{\boxed{c_2}}$$

Let each tag calculate this

Pre compute this at the server
and assigns it to each tag

Query & Reply Phase

Reader (S, H, P)

Tag (H'_1, c_2)

Query

RNG()

$$c_1 = r \times H'_1$$

$$c = c_1 \oplus c_2$$

Unlinkable-ID = C

ID Resolve Phase

$r || id := \text{Decrypt}(c)$

This scheme provides unlinkability of IDs against passive attack. But by make this challenge-response type, this can also be secure against adaptive attack [CKM+07]

- Code-based PKCs are suitable for
 - **Heterogeneous** Network/Applications, such as
 - Broadcast Authentication and Unlinkable-ID
 - for **light weight** devices such as
 - RFID, sensors, SCADA devices
- Since they
 - do not require heavy multi-precision modular exponentiation
 - can be executed mostly using only xors highly in parallel

- Research themes left in this area include
 - Further reduction of PK sizes
 - New attacks (especially on QD)
 - New primitives/applications
 - Implementation and side-channel attacks
 - Provable security
 - etc.

Thank you very much for your
kind attention!!

References

- [GPT91] E.M. Gabidulin, A.V. Paramonov, O.V. Tretjakov, "Ideals over a Non-commutative Ring and Their Application in Cryptology," Proc. of Eurocrypt'91, LNCS 547, pp.482-489, 1991
- [SS92] V. Sidelnikov and S. Shestakov, "On insecurity of cryptosystems based on generalized Reed-Solomon codes." Discrete Math. Appl. 2(4), pp. 439–444, 1992
- [MRS00] C. Monico, J. Rosenthal, and A. Shokrollahi, "Using low density parity check codes in the McEliece cryptosystem," Proc. of IEEE ISIT'00, p. 215, 2000
- [Loi00] P. Loidreau "Strengthening McEliece cryptosystem," Proc. of ASIACRYPT'00, pp. 585–598, 2000
- [PCT+02] A. Perrig, R. Canetti, J.D. Tyger and D. Song, "The TESLA Broadcast Authentication Protocol," CryptoBytes, Vol.5, No.2, Summer/Fall, pp. 2-13
- [KI02] K. Kobara and H. Imai "Semantically secure McEliece public-key cryptosystem". IEICE Trans., E85-A(1), pp. 74–83, 2002

References

- [KI03] K. Kobara and H. Imai “On the one-wayness against chosen-plaintext attacks on the Loidreau’s modified McEliece PKC”. IEEE Trans. on IT, 49(12), pp. 3136–3168, 2003
- [Ove05] R. Overbeck, “A new structural attack for GPT and variants,” Proc. of Mycrypt’05, LNCS 3517, pp. 50-63, 2005
- [AFS05] D. Augot, M. Finiasz, and N. Sendrier, “A family of fast syndrome based cryptographic hash function,” Proc. of Mycrypt’05, LNCS 3715, pp. 64-83, 2005
- [Gab05] P. Gaborit, “Shorter keys for code based cryptography,” Proc of WCC’05, pp. 81–91, 2005
- [SKI06] M. Suzuki, K. Kobara, and H. Imai. “Privacy enhanced and light weight RFID system without tag synchronization and exhaustive search,” Proc. of IEEE SMC’06, pp. 1250-1255, 2006
- [Wie06] C. Wieschebrink, “An attack on a modified Niederreiter encryption scheme,” Proc. of PKC’06, LNCS 3958, pp.14–26, 2006

References

- [Ble06] D. Bleichenbacher,
<http://www.derkeiler.com/Newsgroups/sci.crypt/2006-12/msg00480.html> , 2006
- [BC07] M. Baldi and F. Chiaraluce “Cryptanalysis of a new instance of McEliece cryptosystem based on QC-LDPC Codes,” IEEE ISIT, pp. 2591-2595, 2007
- [CKM+07] Y. Cui, K. Kobara, K. Matsuura, and H. Imai “Lightweight Asymmetric Privacy-Preserving Authentication Protocols Secure against Active Attack” Proc. of PerSec’07, pp. 223–228, 2007
- [NIK+07] R. Nojima, H. Imai, K. Kobara and K. Morozov, “Semantic Security for the McEliece Cryptosystem without Random Oracles,” Proc. Of WCC’07, pp. 257-268, 2007
- [DQN08] R. Dowsley, J. Müller-Quade and A. C. A. Nascimento, “A CCA2 Secure Public Key Encryption Scheme Based on the McEliece Assumptions in the Standard Model,” <http://eprint.iacr.org/2008/468> ,2008

References

- [AFG+08] D. Augot, M. Finiasz, Ph. Gaborit, S. Manuel, and N. Sendrier, "SHA-3 proposal: FSB," SHA-3 NIST competition, 2008
- [Ber08] D. J. Bernstein "List decoding for binary Goppa codes," <http://cr.yp.to/codes/goppalist-20081107.pdf> , 2008
- [DGQ+08] R. Dowsley, J. van de Graaf, J. M.-Quade, and A. Nascimento. "Oblivious transfer based on the McEliece assumptions." Proc. of ICITS'08, LNCS 5155, pp. 107-117, 2008
- [KMO08] K. Kobara, K. Morozov and R. Overbeck "Coding-Based Oblivious Transfer" Proc. of Mathematical Methods in Computer Science, LNCS 5393, pp. 142-156, 2008
- [BLN+09] D. J. Bernstein, T. Lange, R. Niederhagen, C. Peters, P. Schwabe. "FSBday: implementing Wagner's generalized birthday attack against the SHA-3 round-1 candidate FSB." URL: <http://cr.yp.to/papers.html#fsbday> , 2009

References

- [OTD08] A. Otmani, J.P. Tillich, L. Dallot, "Cryptanalysis of Two McEliece Cryptosystems Based on Quasi-Cyclic Codes," <http://arxiv.org/abs/0804.0409> 2008
- [DQN08] R. Dowsley and J. M. Quade and A. C. A. Nascimento, "A CCA2 Secure Public Key Encryption Scheme Based on the McEliece Assumptions in the Standard Model," <http://eprint.iacr.org/2008/468>, 2008
- [FS09] M. Finiasz and N. Sendrier, "Security Bounds for the Design of Code-based Cryptosystems " Proc of Asiacrypt'09, <http://eprint.iacr.org/2009/414> , 2009

References

- [Wie09] C. Wieschebrink, "Cryptanalysis of the Niederreiter Public Key Scheme Based on GRS Subcodes," <http://eprint.iacr.org/2009/452>, 2009
- [MB09] R. Misoczki and P. Barreto, "Compact McEliece Keys from Goppa Codes," Proc. of SAC'09, pp. 2009
- [MB09a] R. Misoczki and P. Barreto, personal communication, 2009
- [BCG+09] T. Berger, P.-L. Cayrel, P. Gaborit, and A. Otmani, "Reducing Key Length of the McEliece Cryptosystem," Proc. of AFRICACRYPT'09, LNCS5580, pp. 77-97, 2009
- [UL09] V. G. Umana and G. Leander "Practical Key Recovery Attacks On Two McEliece Variants," <http://eprint.iacr.org/2009/509>, 2009