AFCS / Spring 2014 / *Applications of Field Theory to Cryptography*

Prof.Dr. Ferucio Laurențiu Țiplea

“Al. I. Cuza” University of Iași
Department of Computer Science
Iasi 740083, Romania
E-mail: fltiplea@mail.dntis.ro
URL: http://www.infoiasi.ro/~fltiplea

5. Applications to cryptography (AES)

- On January 2, 1997, the American National Institute for Standards and Technology (NIST) invited cryptographers from all over the world to develop candidates for a new standard for the protection of sensitive electronic information;

- Twenty-one teams of cryptographers from 11 countries submitted candidates;

- On October 2, 2000, the winner was announced: the algorithm Rijndael (pronounced “Rhine-dahl”), designed by two Flemish researchers, Joan Daemen and Vincent Rijmen.

5. Applications to cryptography (AES)

The strong points of Rijndael are:

- a simple and elegant design;
- efficient and fast on modern processors, but also compact in hardware and on smartcards.

These features make Rijndael suitable for a broad range of applications.

On November 26, 2001, Rijndael was officially published as the Advanced Encryption Standard (AES).

5. Applications to cryptography (AES)

AES processes data blocks of $4 \times m \times 8$ bits using a key of $4 \times k \times 8$ bits, where $m, k \in \{4, 6, 8\}$, as follows:

- first, the data block is divided into groups of 8 bits each (called bytes), obtaining in this way an array of bytes which is then organized as a $4 \times m$ matrix

$$\begin{pmatrix} b_0 & b_4 & \cdots & b_{4m-4} \\ b_1 & b_5 & \cdots & b_{4m-3} \\ b_2 & b_6 & \cdots & b_{4m-2} \\ b_3 & b_7 & \cdots & b_{4m-1} \end{pmatrix}$$

- the matrix obtained as above is considered as a plaintext symbol of the cryptosystem. It is encrypted by performing a set of transformations on it. The result is a $4 \times m$ matrix of bytes as well.

In Rijndael, bytes are represented in various ways:

- as sequences of 8 bits, or
- as 8-dimensional (row) vectors over $\mathbf{Z}_2$, or
- as sequences of two hexadecimal digits.

For example, the following notations refer to the same byte:

$$00111101, \qquad (0,0,1,1,1,1,0,1), \qquad (3d)_h$$

("$(\cdot)_h$" stands for the hexadecimal notation).

5. Applications to cryptography (AES)

- $\mathcal{P} = \mathcal{C} = \mathcal{M}_{4\times m}(\mathbf{Z}_2^8)$, where $m \in \{4, 6, 8\}$;
- $\mathcal{K} = \mathcal{M}_{4\times k}(\mathbf{Z}_2^8)$, where $k \in \{4, 6, 8\}$;
- For any $K \in \mathcal{K}$,

$$e_K = T^f_{K_n} \circ T_{K_{n-1}} \circ \cdots \circ T_{K_1} \circ T^i_{K_0}$$

and

$$d_K = T^{-f}_{K_0} \circ T^{-1}_{K_1} \circ \cdots \circ T^{-1}_{K_{n-1}} \circ T^i_{K_n}$$

- n denotes the number of *rounds* to be performed during the execution of the algorithm. It is dependent on the key and block length

n	$m=4$	$m=6$	$m=8$
$k=4$	10	12	14
$k=6$	12	12	14
$k=8$	14	14	14

5. Applications to cryptography (AES)

- T_Z^i, T_Z, and T_Z^f are transformations given by:
  - $T_Z^i = A_Z$,
  - $T_Z = A_Z \circ Mc \circ Sh \circ S$,
  - $T_Z^f = A_Z \circ Sh \circ S$,

  for any $Z \in \mathcal{M}_{4\times m}(\mathbf{Z}_2^8)$.

- A_Z, called the AddRoundKey transformation, is just a simple bitwise XOR operation extended to matrices. That is,

$$A_Z(X)(i,j) = X(i,j) \oplus Z(i,j),$$

for any $X \in \mathcal{M}_{4\times m}(\mathbf{Z}_2^8)$, $0 \leq i \leq 3$ and $0 \leq j \leq m-1$. We simply write $A_Z(X) = X \oplus Z$;

- S, called the SubBytes transformation, is a non-linear byte substitution that operates independently on each byte of the input matrix. It uses a substitution table that can be computed by

$$S(X)(i,j)^t = M_1 \cdot X(i,j)' \oplus C,$$

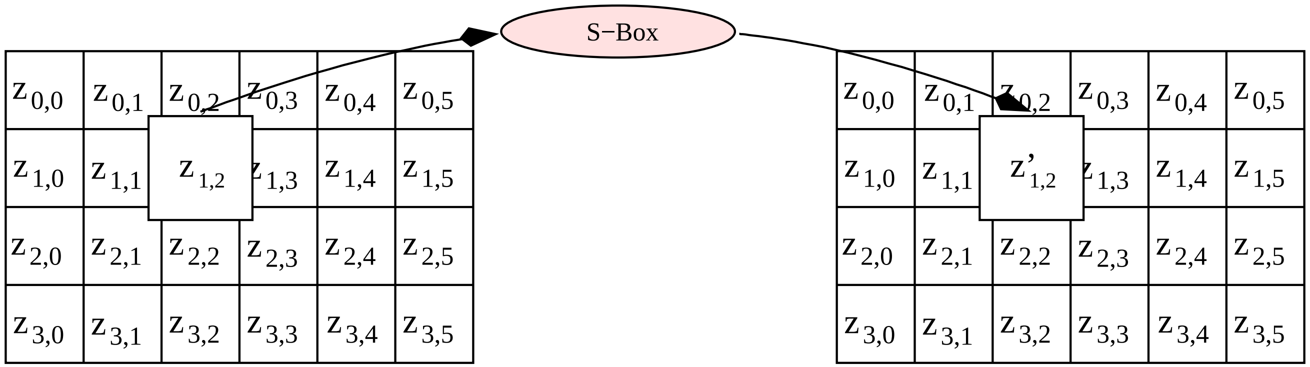
where

$$M_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}, \quad C = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix},$$

and

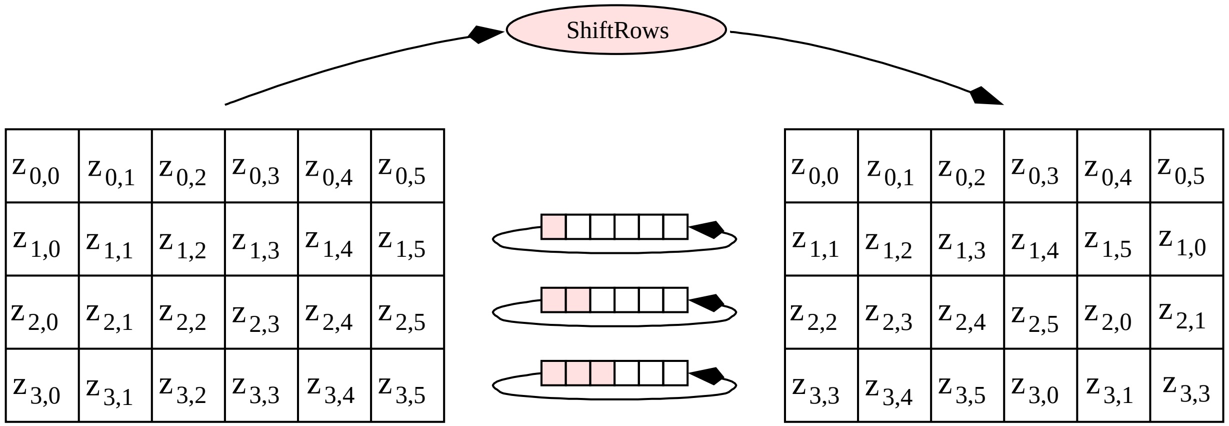
$$X(i,j)' = \begin{cases} (0,0,0,0,0,0,0,0)^t, & \text{if } X(i,j) = (00)_h \\ (X(i,j)^{-1})^t, & \text{otherwise} \end{cases}$$

(the inverse is in the finite field $GF(2^8)$ with the irreducible polynomial $x^8 + x^4 + x^3 + x + 1 \in \mathbf{Z}_2[x]$).



5. Applications to cryptography (AES)

- Sh, called the ShiftRows transformation, cyclically shifts the rows of the input matrix over different numbers of positions (offsets). The ith row is shifted over $C_i = i$ positions (i=0,1,2,3)



Formally, we may write

$$Sh(X)(i,j) = X(i,(j + C_i)\ mod\ m),$$

for any $X \in \mathcal{M}_{4\times m}(\mathbf{Z}_2^8)$, $0 \leq i \leq 3$ and $0 \leq j \leq m-1$;

- Mc, called the MixColumns transformation, treats each column as a polynomial over $GF(2^8)$ and multiplies it modulo x^4+1 with a fixed polynomial $a(x)$ given by:

$$a(x) = (03)_h x^3 + (01)_h x^2 + (01)_h x + (02)_h.$$

This transformation can be written as a matrix multiplication in $GF(2^8)[x]$,

$$Mc(X) = M_2 \bullet X,$$

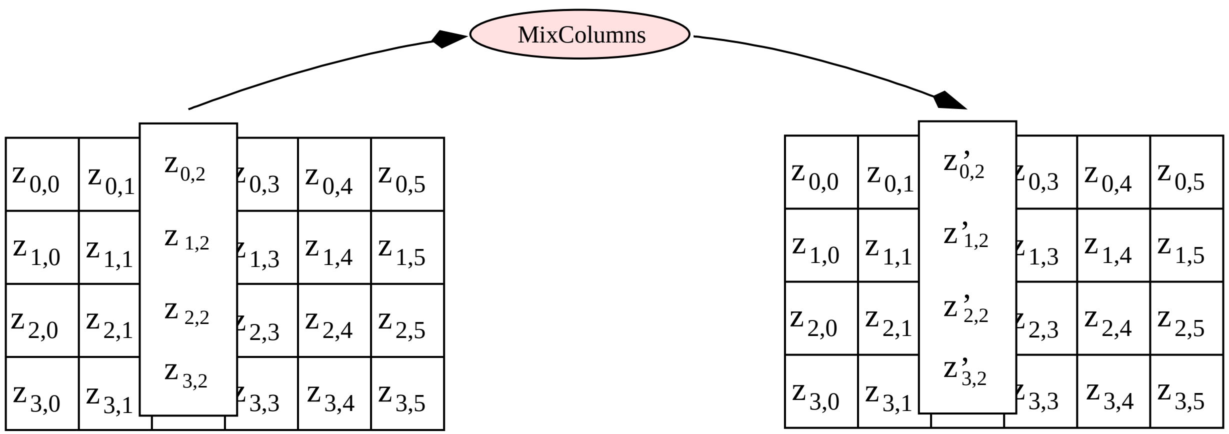
where

$$M_2 = \begin{pmatrix} (02)_h & (03)_h & (01)_h & (01)_h \\ (01)_h & (02)_h & (03)_h & (01)_h \\ (01)_h & (01)_h & (02)_h & (03)_h \\ (03)_h & (01)_h & (01)_h & (02)_h \end{pmatrix}$$

The matrix M_2 is invertible and its inverse is

$$M_2^{-1} = \begin{pmatrix} (0e)_h & (0b)_h & (0d)_h & (09)_h \\ (09)_h & (0e)_h & (0b)_h & (0d)_h \\ (0d)_h & (09)_h & (0e)_h & (0b)_h \\ (0b)_h & (0d)_h & (09)_h & (0e)_h \end{pmatrix}$$

The transformation is pictorially represented by



5. Applications to cryptography (AES)

- T_Z^{-1} and T_Z^{-f} are transformations given by:
  - $T_Z^{-1} = A_{Mc^{-1}(Z)} \circ Mc^{-1} \circ Sh^{-1} \circ S^{-1}$
  - $T_Z^{-f} = A_Z \circ Sh^{-1} \circ S^{-1}$,

  for any $Z \in \mathcal{M}_{4\times m}(\mathbf{Z}_2^8)$.

  The transformations S, Sh, and Mc are invertible;

5. Applications to cryptography (AES)

- $K_0, \ldots, K_n$, called the round keys, and obtained as follows:
  - define first $W_0, W_1, \ldots, W_{m(n+1)-1}$ by
    * $W_i = K(-, i)$, for any $0 \leq i \leq k-1$;
    * $W_i = W_{i-k} \oplus T(W_{i-1})$, where:
      · $T(W) = \begin{cases} SB(RB(W)) \oplus Rcon(i/k), & \text{if } i \bmod k = 0 \\ SB(W), & \text{if } k > 6 \text{ and } i \bmod k = 4 \\ W, & \text{otherwise} \end{cases}$

      · $RB((z_0, z_1, z_2, z_3)^t) = (z_1, z_2, z_3, z_0)^t$;

      · $SB((z_0, z_1, z_0, z_3)^t) = (S(z_0), S(z_1), S(z_2), S(z_3))^t$;

      · $Rcon(i) = (RC(i), (00)_h, (00)_h, (00)_h)$, where $RC(1) = (01)_h$ and $RC(i) = x \bullet RC(i-1)$, for all $k \leq i \leq m(n+1)-1$;
  - $K_i = (W_{im}, W_{im+1}, \ldots, W_{(i+1)m-1})$, for any $i \geq 0$.

5. Applications to cryptography (AES)

AES Security (very hot topic):

- In a recent paper (Asiacrypt 2002), Nicolas Courtois and Josef Pieprzyk showed that Rijndael can be written as an over-defined system of multivariate quadratic equations (MQ). For example authors showed that for 128-bit Rijndael, the problem of recovering the secret key from one single plaintext can be written as a system of 8000 quadratic equations with 1600 binary unknowns;

- If Shamir's XL algorithm would work for efficiently solving large systems of equations, then attacking Rijndael by such a method would require only a few known plaintexts to succeed.

AES Security:

- A new sub-class of XL attacks: XSL attacks;

- XL and XSL attacks do work in many interesting cases. Unfortunately they are heuristic, and their behavior is not well understood. There are examples where these or similar attacks do behave in practice as it is predicted, and there are examples where they do not.

For the time being, no efficient attack against AES is known.