Advanced Encryption Standard (AES) for the US

Bill Bauldry
Appalachian State University

Abstract

Rijndael, a symmetric key cypher, designed by Joan Daemen and Vincent Rijmen, both of Belgium, has been selected by NIST as the proposed *Advanced Encryption Standard* (AES) replacing DES. Rijndael is based on arithmetic in the Galois field of 2^8 elements, $GF(2^8)$. We will present an overview of the algorithm with notes on computability in $GF(2^8)$.

AES/Rijndael

Designed by:

Joan Daemen, Proton World International

Vincent Rijmen, Katholique Universiteit Lueven

- Block cypher
- Symmetric key
  - Arithmetic based in the Galois field $GF(2^8)$
- Fast and scalable
- Resistant to all known cryptanalysis attacks

Background: Finite Fields, I

- **Theorem.** For every prime p and positive integer n, there exists a finite field having p^n elements.

- **Theorem.** Any two fields with p^n elements are isomorphic.

$< \mathbf{Z}_p, \oplus, \odot >$ is the finite field of order p

Background: Finite Fields, II

- **Theorem.** Any finite field is isomorphic to a simple field extension of $\mathbf{Z}_p$ for some prime p.

- **Theorem.** Let F be a field, let $m(x)$ be irreducible in $F[x]$. Then $F[x] / m(x)$ is a field.

$GF(2^8)$

- The Galois field with 2^8 elements is the finite field

$$GF(2^8) = \mathbf{Z}_2[x] \,/\, m(x)$$

where m is irreducible in $\mathbf{Z}_2[x]$ and has degree 8.

- Rijndael chooses $m(x) = 1 + x + x^3 + x^4 + x^8$

Computing in $GF(2^8)$

- Addition is *xor* (subtraction is addition)
- Multiplication by x is *shift left* and, if overflow, subtract $1 + x + x^3 + x^4 + x^8$ i.e.,

$$[b_7, b_6, \ldots, b_0] \times x = \begin{cases} [b_6, \ldots, b_0, 0] & \text{if } b_7 = 0 \\ [b_6, \ldots, b_0, 0] \otimes 00011011 & \text{if } b_7 = 1 \end{cases}$$

Block Size, Key Size, Number of Rounds

- Text Blocks are:
  128 bits in a 4 × 4 byte array
  (originally 128, 192, or 256 bits in 4 × *Nb* byte arrays)
- Key lengths are:
  128, 192, or 256 bits in 4 × *Nk* byte array
- Number of Rounds:
  10, 12, or 14 matching key length

Block & Key Structure

- Block and Key — both 128 bit / 16 bytes:

$$[b_0, b_1, b_2, \ldots, b_{15}] \Rightarrow \begin{bmatrix} b_0 & b_4 & b_8 & b_{12} \\ b_1 & b_5 & b_9 & b_{13} \\ b_2 & b_6 & b_{10} & b_{14} \\ b_3 & b_7 & b_{11} & b_{15} \end{bmatrix}$$

- Data matrix is *State*;
- Key matrix is *RoundKey*

Rijndael Round

1. SubBytes (S-Box substitution)
2. ShiftRows (rotations)
3. MixColumn (linear comb. in $GF(2^8)$)
   (skipped for last round)
4. AddRoundKey (*State* xor *RoundKey*)

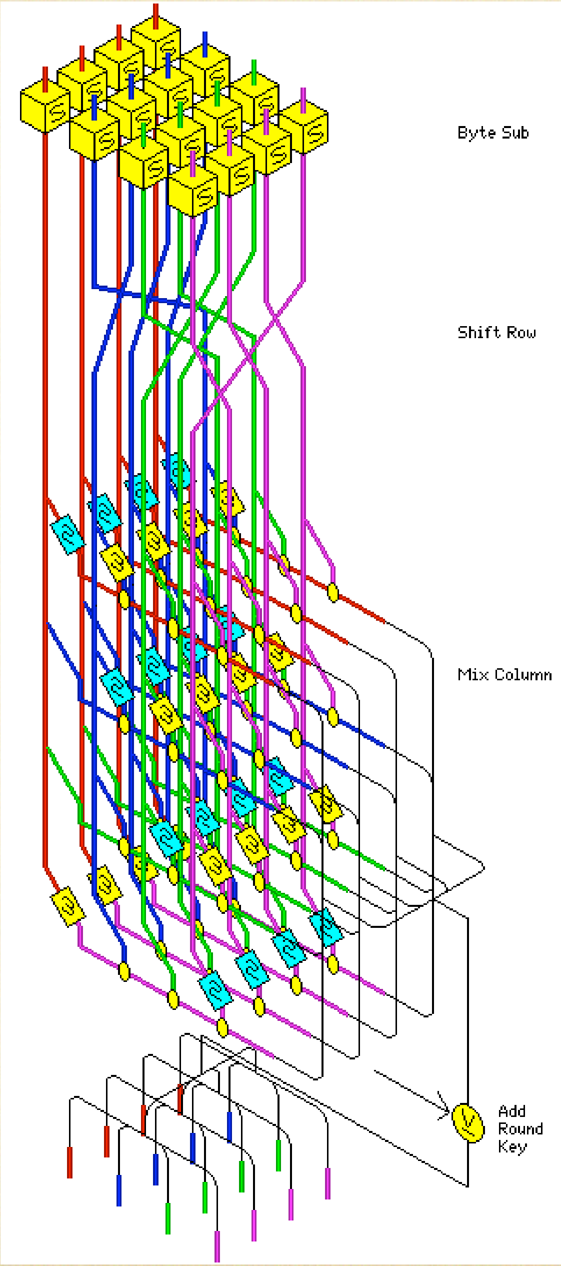


Diagram by John Savard Edmunton, CA

Rijndael Cypher

AES(*data_block, key*)

{in *State, RoundKeys*

State ← *State* xor $RoundKey_0$

for *Round* = 1 **to** *Nr*

SubBytes(*State*)

ShiftRow (*State*)

If not(last Round) **then** *MixColumn*(*State*)

State ← *State* xor $RoundKey_{Round}$

out *State* }

S-Box Arithmetic

- Elements in $G := GF(2^8, 1+\alpha+\alpha^3+\alpha^4+\alpha^8)$
  $n_{hex} \Rightarrow n_{bin} \Rightarrow$ (polynomial with n's bits for coeffs)
  Arithmetic in Z_2 (+/*), then mod by $1+\alpha+\alpha^3+\alpha^4+\alpha^8$
  polynomial $\Rightarrow n_{bin} \Rightarrow n_{hex}$

- ByteSub(x) = $\mathbf{A} \times x^{-1} + 63_{hex}$

- Precompute and use look-up table

ShiftRow

- Left rotate row n of *State* by n positions for $n = 0..3$

$$\begin{bmatrix} b_0 & b_4 & b_8 & b_{12} \\ b_1 & b_5 & b_9 & b_{13} \\ b_2 & b_6 & b_{10} & b_{14} \\ b_3 & b_7 & b_{11} & b_{15} \end{bmatrix} \Rightarrow \begin{bmatrix} b_0 & b_4 & b_8 & b_{12} \\ b_5 & b_9 & b_{13} & b_1 \\ b_{10} & b_{14} & b_2 & b_6 \\ b_{15} & b_3 & b_7 & b_{11} \end{bmatrix}$$

MixColumn Arithmetic

- Elements in

$$G[x] \Big/ x^4 + 1$$

$$\begin{bmatrix} b_i \\ b_{i+1} \\ b_{i+2} \\ b_{i+3} \end{bmatrix} \Rightarrow b(x) = b_i + b_{i+1}x + b_{i+2}x^2 + b_{i+3}x^3 \in G[x]$$

$$\Rightarrow p(x) = b(x) \times q(x)$$

where $q(x) = 2 + x + x^2 + 3x^3$

$$p(x) \bmod \left(1 + x^4\right) = B_0 + B_1 x + B_2 x^2 + B_3 x^3 \Rightarrow \begin{bmatrix} B_i \\ B_{i+1} \\ B_{i+2} \\ B_{i+3} \end{bmatrix}$$

MixColumn Arithmetic

- MixColumn is equivalent to

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \times \begin{bmatrix} b_0 & b_4 & b_8 & b_{12} \\ b_1 & b_5 & b_9 & b_{13} \\ b_2 & b_6 & b_{10} & b_{14} \\ b_3 & b_7 & b_{11} & b_{15} \end{bmatrix}$$

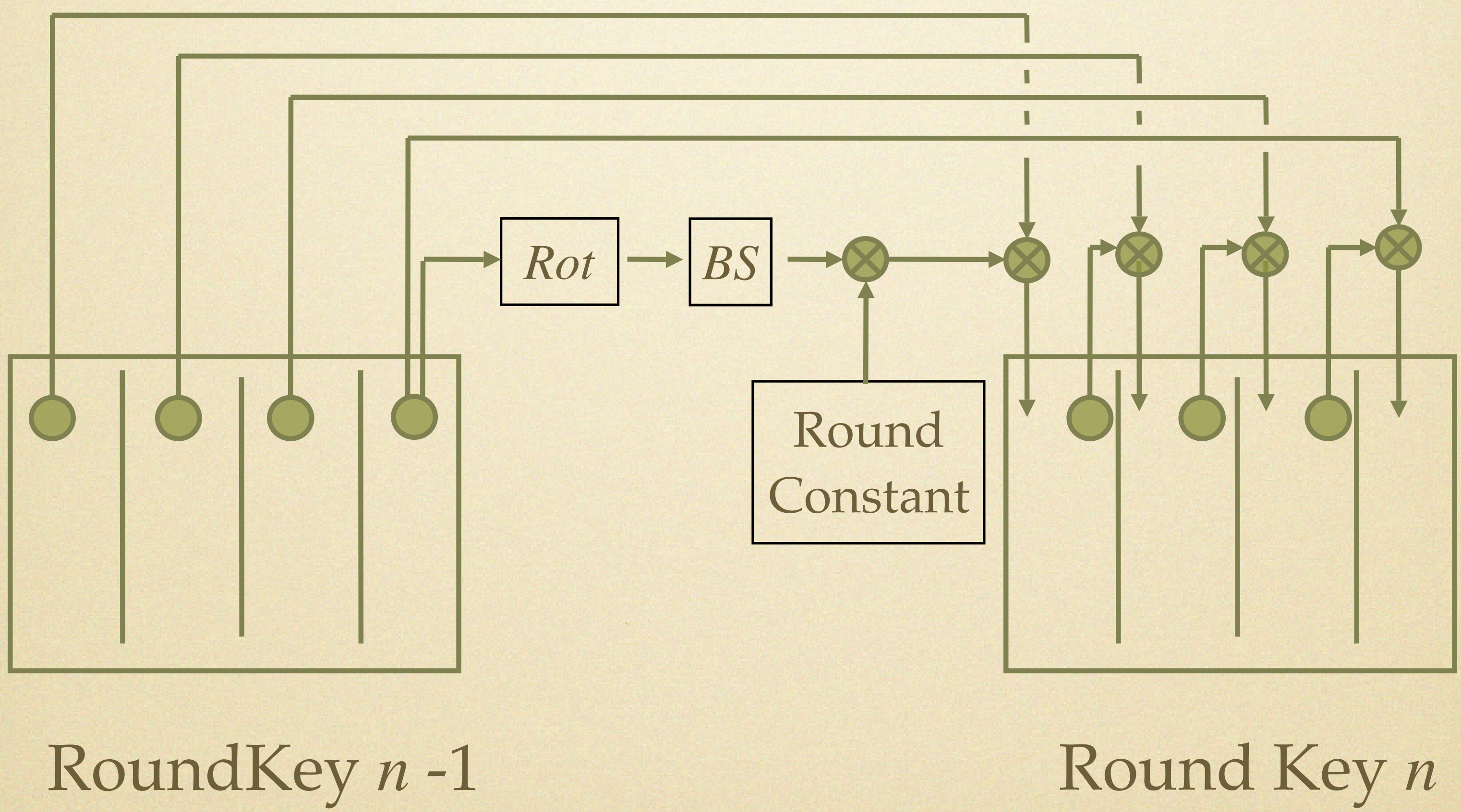
with calculations in $GF(2^8)$.

AddRoundKey

- Addition in $\mathbf{Z}_2$ is equivalent to *xor* of bits.

$$\textit{State} \leftarrow \textit{State} \text{ xor } \textit{RoundKey}$$

Key Expansion



Inverse Cypher

- Reverse Steps
- Use Expanded Keys in Reverse Order
- ByteSub and ShiftRow Commute
- MixColumn Matrix is Invertible

References

- NIST

  csrc.nist.gov/encryption/aes/rijndael/

- Vincent Rijmen's *Rijndael Home Page*

  www.esat.kuleuven.ac.be/~rijmen/rijndael/

- Maple Cryptology Package with Rijndael

  www.mathsci.appstate.edu/~wmcb/CryptologyInClass/