AGH University of Science
and Technology

Selected Topics in Cryptography
Advanced Encryption Standard

Mateusz Dyrdol

Department of Telecommunications

23.10.2017

AGH

Origins of Advanced Encryption Standard Requirements for AES

Advanced Encryption Standard

Origins

Due to limitations of DES (small key and block sizes), National Institute of Standards and Technology (NIST) started a open process to select a new block cipher. In the 1998, fifteen proposals were submitted.

AES is a subset of the Rijndael cipher developed by two Belgian cryptographers, Vincent Rijmen and Joan Daemen, who submitted a proposal to NIST during the AES selection process. It supersedes the Data Encryption Standard (DES).

AES is available in many different encryption packages, and is the first (and only) publicly accessible cipher approved by the National Security Agency (NSA) for top secret information.

Advanced Encryption Standard

Requirements for AES

In the selection process, NIST asked for:

- A block cipher, block length: 128 bit
- Key length: 128, 192 and 256 bit
- Suitability for hardware and software

NIST platform used to test candidate cipher algorithms:

- PC IBM-compatible,Pentium Pro 200 MHz, 64 MB RAM, WINDOWS 95
- Borland C++ 5.0 compiler and Java Development Kit (JDK) 1.1

NIST selection of the winning algorithm based on:

- Security
- Efficient implementation **both** in hardware and software
- Code length and memory utilization

Advanced Encryption Standard

Overwiev of AES

- AES is **not** Feistel Network
- AES is a type of substitution-permutation network
- AES has 128 bits block size
- AES has three allowable key sizes K: 128,192,256 bit
- AES has variable number of rounds:
  - If K= 128, then Nr= 10
  - If K= 192, then Nr= 12
  - If K= 256, then Nr= 14

Advanced Encryption Standard

Key and Block

Key with variable length (128, 192, 256 bits), represented with a matrix (array) of bytes with 4 *rows* and N_k *columns*, N_k = key length/32:

- Key of 128 bits = 16 bytes, $N_k = 4$
- Key of 192 bits = 24 bytes, $N_k = 6$
- Key of 256 bits = 32 bytes, $N_k = 8$

Block length 128 bits, represented with a matrix (array) of bytes with 4 *rows* and N_b *columns*, N_b = block length/32:

- Block of 128 bits = 16 bytes, $N_b = 4$

Advanced Encryption Standard

State

Internally, the AES algorithm's operations are performed on a two-dimensional array of bytes called the **State**

- Four rows, each containing Nb bytes
- Nb columns, constituted by 32-bit words
- Sr,c denotes the byte in row r and column c
- The array of bytes in input is copied in the State matrix
- At the end, the State matrix is copied in the output matrix

Advanced Encryption Standard

Steps for encryption

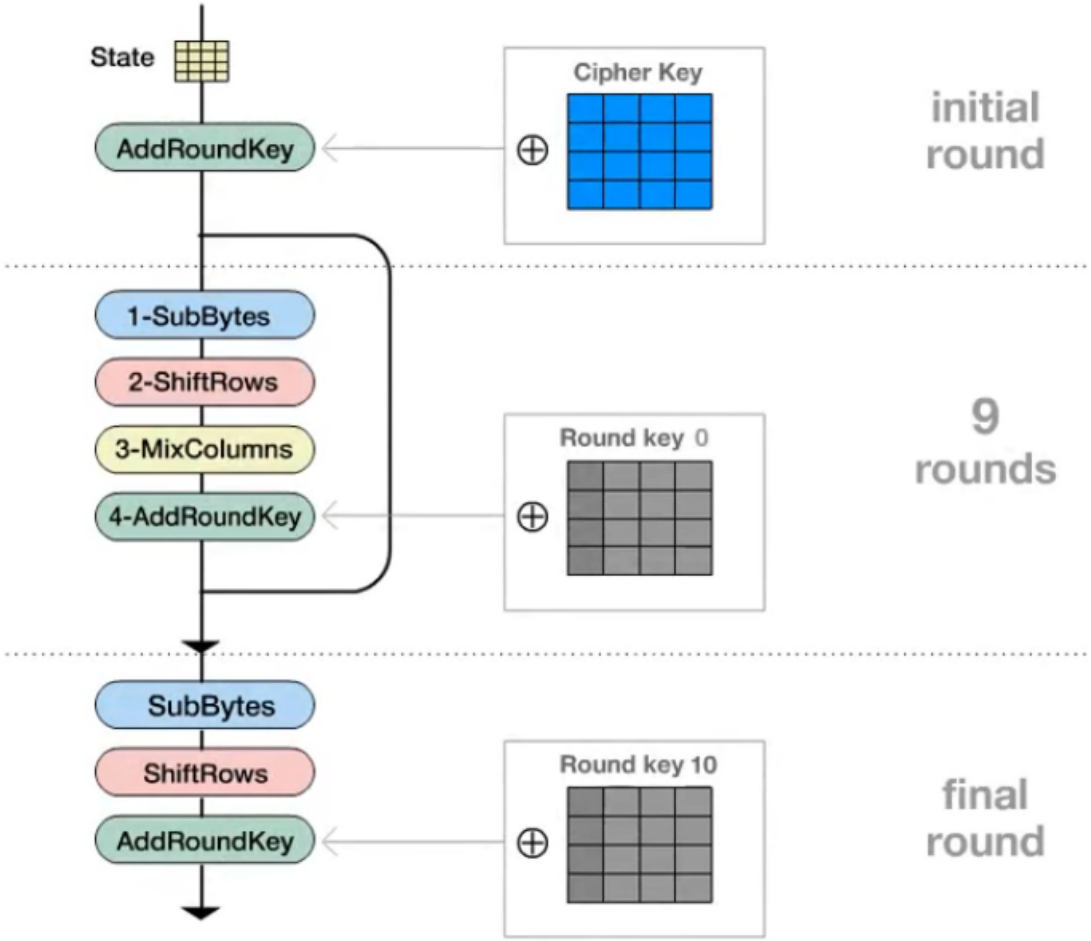


Figure: AES algorithm description

Advanced Encryption Standard

1.SubBytes

• Byte substitution using a non-linear (but *invertible*) S-Box (independently on each byte)
• S-box is represented as a 16x16 array, rows and columns indexed by hexadecimal bits
• 8 bytes replaced as follows: 8 bytes define a hexadecimal number **rc**, then $S_{r,c}$ = binary(S-box(r,c))

• How is AES S-box different from DES S-boxes?
  - Only **one** S-box
  - S-boxes based on modular arithmetic with polynomials, can be defined algebraically
  - Easy to analyze, prove attacks fail

Advanced Encryption Standard

1.SubBytes using Rijndael S-Box Table

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	3	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Example: hexa 53 is replaced with hexa ED

Advanced Encryption Standard

2.ShiftRows

The first row is left unchanged. Each byte of the second row is shifted one to the left. Similarly, the third and fourth rows are shifted by offsets of two and three respectively.

D8	56	A3	23
29	E2	FE	48
5A	C6	92	B4
1D	37	52	61

D8	56	A3	23
E2	FE	48	29
92	B4	5A	C6
61	1D	37	52

Advanced Encryption Standard

3.MixColumns

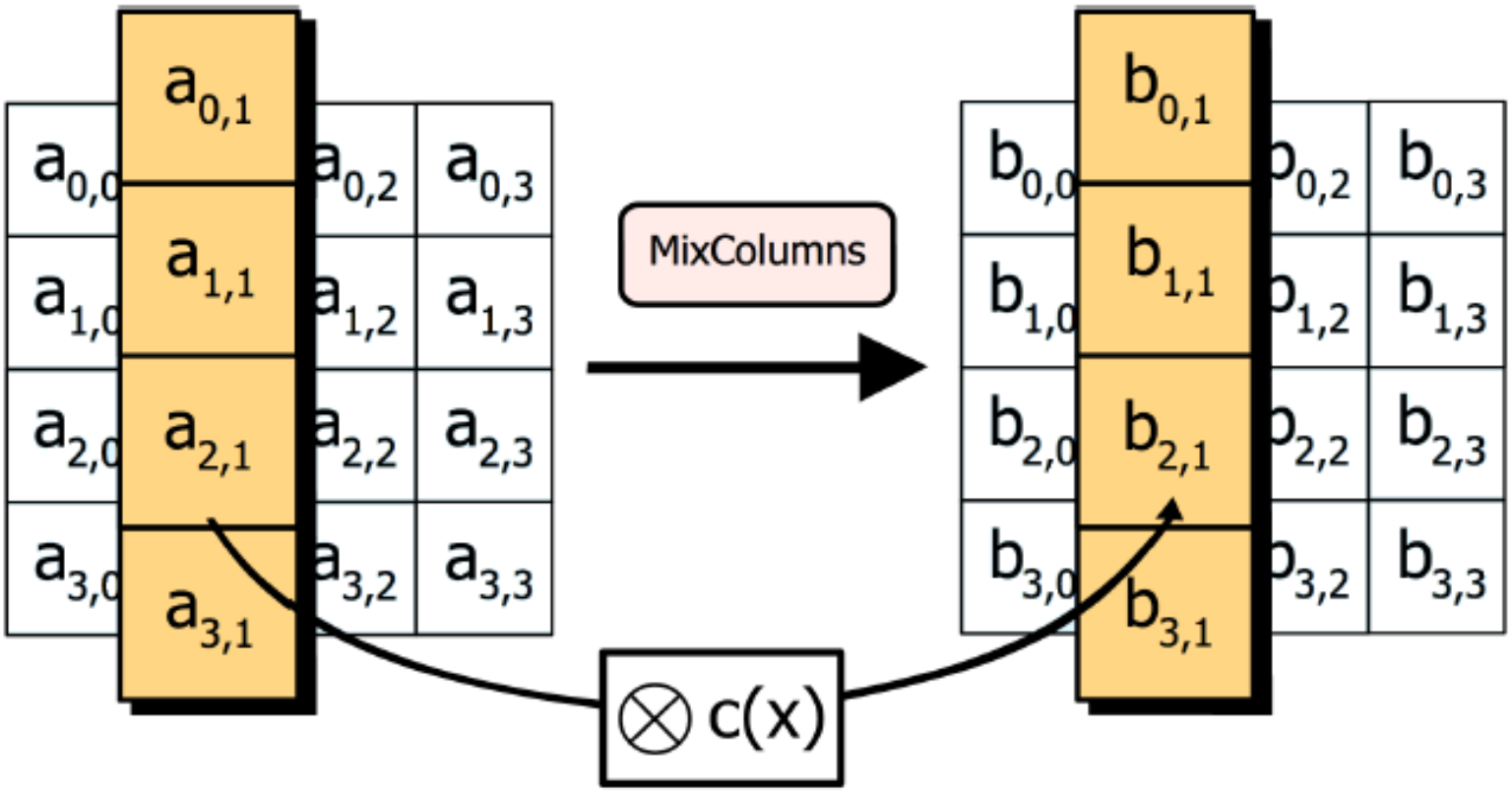
Each column is represented as a 4 byte vector.
Each column of State is replaced by a new column which is formed by multiplying that column by a certain matrix of elements of the field.
We can also see this operation as polynomial multiplication where each column is represented with polynomial a(x):

$$a(x) = c(x).a(x) mod x^4 + 1 =$$
$$(03x^3 + 01x^2 + 01x + 02).(a_3x^3 + a_2x^2 + a_1x^1 + a_0) mod x^4 + 1$$

$$c(x) = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$$

Advanced Encryption Standard

3.MixColumns



Advanced Encryption Standard

4.AddRoundKey

The subkey is added by combining each byte of the state with the corresponding byte of the subkey using bitwise XOR.

Advanced Encryption Standard

Key Schedule

Since the key schedule for 128-bit, 192-bit, and 256-bit encryption are very similar, with only some constants changed, the following keysize constants are defined here:

- n has a value of 16 for 128-bit keys, 24 for 192-bit keys, and 32 for 256-bit keys
- b has a value of 176 for 128-bit keys, 208 for 192-bit keys, and 240 for 256-bit keys (with 128-bit blocks as in AES, it is correspondingly larger for variants of Rijndael with larger block sizes).

Advanced Encryption Standard

Key Schedule: Rcon

Rcon is what the Rijndael documentation calls the exponentiation of 2 to a user-specified value. Note that this operation is not performed with regular integers, but in Rijndael's finite field. In polynomial form, 2 is:

$$2_{10} = 00000010_2 = 0x^7 + 0x^6 + 0x^5 + 0x^4 + 0x^3 + 0x^2 + 1x + 0x = b,$$

and we compute:

$$rcon(i) = 2^{i-1} = 2 * 2^{i-2} = 2 * rcon(i-1)$$

Advanced Encryption Standard

Expanding 128-bit key

1. The first 16 bytes of the expanded key are simply the encryption key
2. The rcon iteration value i is set to 1
3. Until we have 176 bytes of expanded key, we do the following to generate 16 more bytes of expanded key:
    1. We do the following to create the first four bytes of expanded key:
        1. We create a 4-byte temporary variable, t
        2. We assign the value of the previous four bytes in the temporary key to t
        3. We rotate one byte to te left
        4. We substitute bytes on t with Rijndael S-Box
        5. We XOR t with i as the rcon iteration value.
        6. We increment i by one.
        7. We XOR t with the first four-bytes of the old key.

        This becomes the next four bytes in the expanded key.
    2. We then do the following three times to create the next twelve bytes of expanded key:
        1. We assign the value of the previous four bytes in the temporary key to t
        2. We XOR t with the four-byte block 16 bytes before the new expanded key.

        This becomes the next four bytes in the expanded key.
4. Done. We now have 176 bytes of key generated.

Advanced Encryption Standard

Attacks on AES

Algebraic attacks

People have shown Rijndael can be written as an over defined system of multivariate quadratic equations.
Paper published at Eurocrypt 2000 Shamir describe an algorithm called XL able to solve efficiently many such systems of equations.

However this fails miserably!

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.

Algebraic attacks(cont.)

Nicolas Courtois and Josef Pieprzyk investigate how to improve XL and adapt it to such special systems. They propose a new class of attacks, attack, called XSL attacks.

Ciphers like Rijndael were referred to as XSL ciphers, because their rounds are composed of the XOR of key material, a nonlinear substitution provided by an S-box, and a linear diffusion stage.

Advanced Encryption Standard

Attacks on AES

Theory that breaks AES-128

In 2011, Andriej Bogdanow, Dymitr Kowratowicz and Christian Rechberger devise a theoretical way to broke any cipher that is using AES-128

Despite the fact, it was impractical. It would take billions of years. This is an example:

If you assume:

- Every person on the planet owns 10 computers.
- There are 7 billion people on the planet.
- Each of these computers can test 1 billion key combinations per second.
- On average, you can crack the key after testing half of the possibilities.

Then the earth's population can crack one encryption key in 77,000,000,000,000,000,000,000,000 years!

Is the AES secure?

AES-256 keys sniffed in seconds

Researchers at Fox-IT have managed to wirelessly extract secret AES-256 encryption keys from a distance of one metre, using $200 worth of parts obtained from a standard electronics store

The research team used a simple loop antenna, attached it to an external amplifier and bandpass filters bought online, and then plugged it into a software defined radio USB stick they bought for $20.

By running a different encryption run on a test rig, the researchers mapped out how the power consumption related to individual bytes of information. That allowed them to take guesses at the 256 possible values of a single byte and the correct choice showed the highest power spike.

There are, of course, some caveats. The tests took place under laboratory conditions, rather than in a busy office or server room where other signals might interfere with the data collection. But it's an interesting example of how an attack previously thought of as unfeasible due to cost and distance has been made easier by smarter and cheaper technology.

AGH

Thank you for attention!