5.2.1 AES Encryption

AES consists of r rounds (numbered $1, \dots, r$), and needs $r+1$ round keys $K_0, K_1, \dots, K_r$ each of length 128 bits.
$K_0, \dots, K_r$ derived from the original key K, as described later.
The number of rounds depends on the key size

key size	no. of rounds r
128	10
192	12
256	14

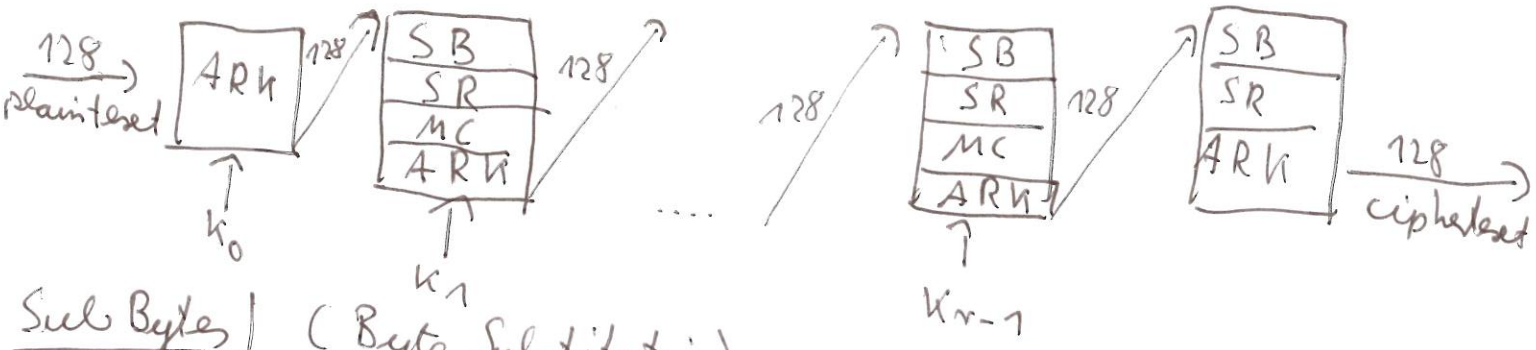
Plaintext of 128 bits arranged as a 4×4 byte matrix

$$\begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix}$$

The round keys are also organized as 4×4 byte matrices
Encryption uses the following operations (or layers)

- Add Round Key (ARK)
- Round $1, \dots, r-1$
  - Sub Bytes (SB)
  - Shift Rows (SR)
  - Mix Columns (MC)
  - Add Round Key (ARK)
- Round r: SB, SR, ARK

Graphically



Sub Bytes (Byte Substitution)

Each byte $f = (b_7, \dots, b_0)$ is viewed as an element in $\mathbb{F}_{28}$

$\sum_{i=0}^{7} b_i x^i \in \mathbb{F}_{28}$

1. Compute f^{-1} in $\mathbb{F}_{28}$, (Set $0^{-1} := 0$), let $f^{-1} = (y_7, \dots, y_0)$
2. Affine transformation

$$\begin{pmatrix} z_0 \\ \vdots \\ z_7 \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 & \dots & 1 \\ \vdots & & & \vdots \\ 0 & 0 & \dots & 1 \end{pmatrix}}_{= F \in \mathbb{F}_2^{8\times 8}} \begin{pmatrix} y_0 \\ \vdots \\ y_7 \end{pmatrix} + \underbrace{\begin{pmatrix} 1 \\ \vdots \\ 0 \end{pmatrix}}_{\in \mathbb{F}_2^8}$$

Input: $(b_7, \dots, b_0)$

Output: bin $(S(b_7, \dots, b_4), (b_3, \dots, b_0))$

SubBytes															
99	124	119	123	242	107	111	197	48	1	103	43	254	215	171	118
202	130	201	125	250	89	71	240	173	212	162	175	156	164	114	192
183	253	147	38	54	63	247	204	52	165	229	241	113	216	49	21
4	199	35	195	24	150	5	154	7	18	128	226	235	39	178	117
9	131	44	26	27	110	90	160	82	59	214	179	41	227	47	132
83	209	0	237	32	252	177	91	106	203	190	57	74	76	88	207
208	239	170	251	67	77	51	133	69	249	2	127	80	60	159	168
81	163	64	143	146	157	56	245	188	182	218	33	16	255	243	210
205	12	19	236	95	151	68	23	196	167	126	61	100	93	25	115
96	129	79	220	34	42	144	136	70	238	184	20	222	94	11	219
224	50	58	10	73	6	36	92	194	211	172	98	145	149	228	121
231	200	55	109	141	213	78	169	108	86	244	234	101	122	174	8
186	120	37	46	28	166	180	198	232	221	116	31	75	189	139	138
112	62	181	102	72	3	246	14	97	53	87	185	134	193	29	158
225	248	152	17	105	217	142	148	155	30	135	233	206	85	40	223
140	161	137	13	191	230	66	104	65	153	45	15	176	84	187	22

Table 5.8.: Lookup-table for the SubBytes Operation of AES, column=(b_3, b_2, b_1, b_0) and row=(b_7, b_6, b_5, b_4)

Input: $0xCB \hat{=} (12|11) \hat{=} (1100|1011) \hat{=} x^7+x^6+x^3+x+1$

$\Rightarrow f^{-1} = (0000|0100) \hat{=} x^2 \hat{=} y$

Just take 3rd column of F as result of $F \cdot y$

$$\begin{pmatrix}0\\0\\1\\1\\1\\1\\1\\0\end{pmatrix} + \begin{pmatrix}1\\1\\0\\0\\0\\1\\1\\0\end{pmatrix} = \begin{pmatrix}1\\1\\1\\1\\1\\0\\0\\0\end{pmatrix}$$

$z = (0001|1111) \hat{=} (1|F) \hat{=} 31$

ShiftRows (SR)

Rows are cyclically (left-) shifted as

$$\begin{pmatrix} b_{00} & \dots & b_{03} \\ \vdots & & \vdots \\ b_{30} & \dots & b_{33} \end{pmatrix} \to \begin{pmatrix} b_{00} & b_{01} & b_{02} & b_{03} \\ b_{11} & b_{12} & b_{13} & b_{10} \\ b_{22} & b_{23} & b_{20} & b_{21} \\ b_{33} & b_{30} & b_{31} & b_{32} \end{pmatrix} = \begin{pmatrix} c_{00} & \dots & c_{03} \\ \vdots & & \vdots \\ c_{30} & \dots & c_{33} \end{pmatrix}$$

MixColumns (MC)

Regard each byte c_{ij} $0 \le i,j \le 3$, as an element in $\mathbb{F}_{2^8}$

Apply a linear transformation by a fixed Matrix A in $\mathbb{F}_{2^8}^{4\times 4}$

$$\underbrace{\begin{pmatrix} 00000010 & \dots & 00000001 \\ \vdots & & \\ 00000011 & \dots & 00000010 \end{pmatrix}}_{A} \begin{pmatrix} c_{00} & \dots & c_{03} \\ \vdots & & \\ c_{30} & \dots & c_{33} \end{pmatrix} = \begin{pmatrix} d_{00} & \dots & d_{03} \\ \vdots & & \vdots \\ d_{30} & \dots & d_{33} \end{pmatrix}$$

A may be written as

$$\begin{bmatrix} x & x+1 & 1 & 1 \\ 1 & x & x+1 & 1 \\ 1 & 1 & x & x+1 \\ x+1 & 1 & 1 & x \end{bmatrix} = T \in \mathbb{F}_{2^8}^{4\times 4}$$

Thereby the calculation of D might be done more efficiently $\to E_+$

AddRoundKey (ARK)

Bitwise addition modulo 2: $d_{ij} \oplus k_{ij} = e_{ij}$

5.2.2 AES Key expansion

Only for key length 128 bits, though similar for 192, 256 bits.

Master key K, $K_0 = K$, 128 bits, 4×4 byte matrix with columns $w(0), w(1), w(2), w(3)$

Expanded by 40 more columns, $i = 4, \dots, 43$

$$w(i) = \begin{cases} w(i-4) \oplus T(w(i-1)) & \text{if } i \equiv 0 \pmod 4 \\ w(i-4) \oplus w(i-1) & \text{if } i \not\equiv 0 \pmod 4 \end{cases}$$

Transformation $T(w(i-1))$ $\quad w(i-1) = \begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \end{pmatrix}$

1. Cyclic Shift (RotByte): $(w_0, w_1, w_2, w_3) \to (w_1, w_2, w_3, w_0)$
$= (u_0, u_1, u_2, u_3)$
2. Apply SubBytes, i.e., $v_i = SB(u_i)$: $(u_0, u_1, u_2, u_3) \to (v_0, v_1, v_2, v_3)$
3. Compute $p(i) = \underbrace{(0000\,0010)}_{RC(i)}^{i/4-1}$ in $\mathbb{F}_{2^8}$
4. $T(w(i-1)) = (v_0 \oplus p(i), v_1, v_2, v_3)$

$p(i)$ or the vector $(p(i), 0, 0, 0)$ are referred to as $RCON(i)$

Round key for round k:

$$K_k = (w(4k), w(4k+1), w(4k+2), w(4k+3)), \quad k = 0, \dots, 10$$

Remark: In the lecture notes this procedure is given in algorithmic form with RotByte, Rcon(i) (RC(i))

5.2.3 / AES Decryption /

Each of the steps SubBytes, ShiftRow, MixColumns, AddRoundKey are invertible, giving the transformation

- Inv SubBytes (ISB)
- Inv ShiftRows (ISR)
- Inv MixColumns (IMC)
- AddRoundKey (ARK)

These operations are applied in reverse order with round keys in in inverse order.

Some of the procedures are interchangeable the decryption can be made to look more like the encryption; if in addition the procedures ARK and IMC are substituted by the equivalent procedures IMC and IARK. Doing so you will have the same structure as in the encryption:

Encryption

ARK

SB SR MC ARK

⋮

SB SR MC ARK

SB SR ~~MC~~ ARK

Decryption

ARK ~~IMC~~

ISR ISB ARK IMC

⋮

ARK IMC

ISR ISB ARK

Equivalent to

ARK

ISB ISR IMC IARK

⋮

IMC IARK

ISB ISR ARK