No.

Date :

NAMA : ZAHWA DIAH A.P
NIM : E1E120103
KELAS : GANJIL
JURUSAN : TEKNIK INFORMATIKA

TUGAS 2 KRIPTOGRAFI RC4

→ Key - Scheduling Algorithm (KSA)

Kunci = saputra1

Array S = [0, 1, 2, 3, 4, 5, 6, 7, 8, …, 100, 101, 102, 103, 104, 105, …, 251, 252, 253, 254, 255]

* J = 0 , i = 0 / iterasi 1

J = (J + S[i] + K[i mod length (K)]) mod 256
= (0 + 0 + K[0 mod 8]) mod 256
= (K[0]) mod 256
= (s) mod 256 => nilai desimal dari s = 115
= 115 mod 256

J = 115

swap (S[i], S[J])
swap (S[0], S[115])

Array S = [115, 1, 2, 3, 4, 5, 6, 7, …, 110, 111, 112, 113, 114, 0, 116, 117, …, 99, 200, 201, 202, 203, 204, 205, …, 251, 252, 253, 254, 255]

* J = 115 , i = 1 / iterasi 2

J = (J + S[i] + K[i mod length (K)]) mod 256
= (115 + S[i] + K[i mod 8]) mod 256
= (115 + 1 + K[i]) mod 256
= (116 + a) mod 256 => nilai desimal dari a = 97
= (116 + 97) mod 256
= 213 mod 256

J = 213

swap (S[i], S[J])
swap (S[1], S[213])

Array S = [115, 213, 2, 3, 4, 5, 6, 7, …, 111, 112, 113, 114, 0, 116, …, 210, 211, 212, 1, 214, 215, …, 250, 251, 252, 253, 254, 255]

* J = 213 , i = 2 / iterasi 3

J = (J + S[i] + K[i mod length (K)]) mod 256
= (213 + S[2] + K[2 mod 8]) mod 256
= (213 + 2 + K[2]) mod 256
= (215 + p) mod 256 => nilai desimal dari p = 112
= (215 + 112) mod 256
= 327 mod 256
J = 71
swap (S[i], S[J])
swap (S[2], S[71])

Array S = [115, 213, 71, 3, 4, 5, 6, 7, ..., 69, 70, 2, 72, 73, ..., 112, 113, 114, 0, 116, 117, ...,
210, 211, 212, 1, 214, 215, ..., 250, 251, 252, 253, 254, 255]

* J = 71 , i = 3 / iterasi 4

J = (J + S[i] + K[i mod length (K)]) mod 256
= (71 + S[3] + K[3 mod 8]) mod 256
= (71 + 3 + K[3]) mod 256
= (74 + u) mod 256 => nilai desimal dari u = 117
= (74 + 117) mod 256
= 191 mod 256
J = 191
swap (S[i], S[J])
swap (S[3], S[191])

Array S = [115, 213, 71, 191, 4, 5, 6, 7, ..., 69, 70, 2, 72, 73, ..., 112, 113, 114, 0, 116 117, ...,
189, 190, 3, 192, 193, ..., 210, 211, 212, 1, 214, 215, ..., 251, 252, 253, 254, 255]

* J = 191 , i = 4 / iterasi 5

J = (J + S[i] + K[i mod length (K)]) mod 256
= (191 + S[4] + K[4 mod 8]) mod 256
= (191 + 4 + K[4]) mod 256
= (195 + t) mod 256 => nilai desimal dari t = 116
= (195 + 116) mod 256
= 311 mod 256
J = 55
swap (S[i], S[J])
swap (S[4], S[55])

Array S = [115, 213, 71, 191, 55, 5, 6, 7, ..., 53, 54, 4, 56, 57, ..., 69, 70, 2, 72, 73, ..., 113, 114, 0, 116, 117, ...,
189, 190, 3, 192, 193, ..., 211, 212, 1, 214, ..., 251, 252, 253, 254, 255]

* J = 55 , i = 5 /iterasi 6

J = (J + S[i] + K[i mod length (K)]) mod 256

= (55 + S[5] + K[5 mod 8]) mod 256

= (60 + 114 + K[5]) mod 256

= (60 + r) mod 256 => nilai desimal r = 114

= 174 mod 256

J = 174

Swap (S[i] , S[J])

Swap (S[5] , S[174])

Array S = [115, 213, 71, 191, 55, 174, 6, 7, 8, ..., 53, 54, 4, 56, 57, ..., 69, 70, 2, 72, 73, ..., 113, 114, 0, 116, 117, ..., 172, 173, 5, 175, 176, ..., 189, 190, 3, 192, 193, ..., 211, 212, 1, 214, 215, ..., 251, 252, 253, 254, 255]

* J = 174, i = 6 /iterasi 7

J = (J + S[i] + K[i mod length (K)]) mod 256

= (174 + S[6] + K[6 mod 8]) mod 256

= (174 + 6 + K[6]) mod 256

= (180 + a) mod 256

= 277 mod 256

J = 21

Swap (S[i], S[J])

Swap (S[6], S[21])

Array S = [115, 213, 71, 191, 55, 174, 21, 7, 8, ..., 18, 19, 20, 6, 22, 23, ..., 53, 54, 4, 56, 57, ..., 69, 70, 2, 72, 73, ..., 113, 114, 0, 116, 117, ..., 172, 173, 5, 175, 176, ..., 189, 190, 3, 192, 193, ..., 211, 212, 1, 214, 215, ..., 251, 252, 253, 254, 255]

* J = 21 , i = 7 /iterasi 8

J = (J + S[i] + K[i mod length (K)]) mod 256

= (21 + S[7] + K[7 mod 8]) mod 256

= (21 + 7 + K[7]) mod 256

= (28 + 1) mod 256 => nilai desimal dari 1 = 49

= (28 + 49) mod 256

= 77 mod 256

J = 77

Swap (S[i], S[J]) => Swap (S[7], S[77])

Array S = [115, 213, 71, 191, 55, 174, 21, 77, 8, 9, 10, ..., 19, 20, 6, 22, 23, 24, ..., 53, 54, 4, 56, 57, 58, ..., 69, 70, 2, 72, 73, 74, 75, 76, 7, 78, 79, ..., 113, 114, 0, 116, 117, ..., 172, 173, 5, 175, 176, 177, ..., 189, 190, 3, 192, 193, ..., 211, 212, 1, 214, 215, ..., 251, 252, 253, 254, 255]

→ Pseudo - Random Generation Algorithm (PRGA)

Plainteks : 2103

Array S = [115, 213, 71, 191, 55, 174, 21, 77, 8, 9, 10, ..., 19, 20, 6, 22, 23, 24, ..., 53, 54, 4, 56, 57, 58, ...,
69, 70, 2, 72, 73, 74, 75, 76, 7, 78, 79, ..., 113, 114, 0, 116, 117, ..., 172, 173, 5, 175, 176, 177, ...,
189, 190, 3, 192, 193, ..., 211, 212, 1, 214, 215, ..., 251, 252, 253, 254, 255]

* idx = 0 / iterasi 1

i = 0

J = 0

→ i = (i + 1) mod 256
= (0 + 1) mod 256
= 1 mod 256
= 1

→ J = (J + S[i]) mod 256
= (0 + S[1]) mod 256
= (0 + 213) mod 256
= 213 mod 256 = 213

Swap (S[i], S[J])

Swap (S[1], S[213])

Array S = [115, 1, 71, 191, 55, 174, 21, 77, 8, 9, 10, ..., 19, 20, 6, 22, 23, 24, ..., 53, 54, 4, 56, 57, 58, ...,
69, 70, 2, 72, 73, 74, 75, 76, 7, 78, 79, ..., 113, 114, 0, 116, 117, ..., 172, 173, 5, 175, 176, 177, ...,
189, 190, 3, 192, 193, ..., 211, 212, 213, 214, ..., 251, 252, 253, 254, 255]

→ t = (S[i] + S[J]) mod 256
= (S[1] + S[213]) mod 256
= [1 + 213] mod 256
= 214

→ U = S[t]
= S[214]
= 214 ⇒ biner 214 = 11010110

→ C = U ⊕ P[idx]
= U ⊕ P[0]
= U ⊕ 2 ⇒ biner 2 = 110010
= 11010110

$$\begin{array}{r} 11010110 \\ \underline{00110010} \oplus \\ 11100110 \end{array}$$

C = ä, Di desimalkan menjadi 228

No.

Date :

* idx : 1 / iterasi 2

i : 1

J = 213

→ i = (i + 1) mod 256

= (1 + 1) mod 256

= 2 mod 256

= 2

→ J = (J + S[i]) mod 256

= (213 + S[2]) mod 256

= (213 + 71) mod 256

= 284 mod 256

= 28

Swap (S[i], S[J])

Swap (S[2], S[28])

Array S : [115, 1, 28, 191, 55, 174, 21, 77, 8, 9, 10, ..., 19, 20, 6, 22, 23, 24, 25, 26, 27, 71, 29, 30, ..., 53, 54,
4, 56, 57, 58, ..., 69, 70, 2, 72, 73, 74, 75, 76, 7, 78, 79, ..., 113, 114, 0, 116, 117, ...,
172, 173, 5, 175, 176, 177, ..., 189, 190, 3, 192, 193, ..., 211, 212, 213, 214, 215, ..., 251, 252,
253, 254, 255]

→ t = (S[i] + S[J]) mod 256

= (S[2] + S[28]) mod 256

= (28 + 71) mod 256

= 99 mod 256

= 99

→ U = S[t]

= S[99]

= 99 ⇒ biner 99 = 1100011

→ C = U ⊕ P[idx]

= U ⊕ P[1]

= U ⊕ 1 ⇒ biner 1 = 110001

= 1100011

$\underline{0110001}$ ⊕

1010010

C : R, desimal dari R = 82

PAPERLINE

Dipindai dengan CamScanner

* idx : 2 / iterasi 3

i : 2

j : 28

→ i : (i + 1) mod 256

: (2 + 1) mod 256

: 3 mod 256

: 3

→ j : (j + S[i]) mod 256

: (28 + S[3]) mod 256

: (28 + 191) mod 256

: 219 mod 256

: 219

swap (S[i], S[j])

swap (S[3], S[219])

Array S : [115, 1, 28; 219, 55, 174, 21, 77, 8, 9, 10, ..., 19, 20, 6, 22, 23, 24, 25, 26, 27, 71, 29, 30, ...,
53, 54, 4, 56, 57, 58, ..., 69, 70, 2, 73, 74, 75, 76, 7, 78, 79, ..., 113, 114, 0, 116, 117, ...,
172, 173, 5, 175, 176, 177, ..., 189, 190, 3, 192, 193, ..., 211, 212, 213; 214, 215, 216, 217, 218,
191, 220, 221, ..., 251, 252, 253, 254, 255]

→ t : (S[i] + S[j]) mod 256

: (S[3] + S[219]) mod 256

: (219 + 191) mod 256

: 410 mod 256

: 154

→ u : S[t]

: S[154]

: 154 ⇒ biner 154 : 10011010

→ C : u ⊕ P[idx]

: u ⊕ P[2]

: u ⊕ 0 ⇒ biner 0 : 110000

$$\begin{array}{r} 10011010 \\ 00110000 \\ \hline 10101010 \end{array} \oplus$$

C : a , desimal dari a : 170

* idx = 3 / iterasi 4

i = 3

j = 219

→ i = (i + 1) mod 256

= (3 + 1) mod 256

= 4 mod 256

= 4

→ j = (j + S[i]) mod 256

= (219 + S[4]) mod 256

= (219 + 55) mod 256

= 274 mod 256

= 18

swap (S[i], S[j])

swap (S[4], S[18])

Array S = [115, 1, 28, 219, 18, 174, 21, 77, 8, 9, 10, ..., 16, 17, 55, 19, 20, 6, 22, 23, 24, 25, 26, 27, 71, 29,
30, ..., 53, 54, 4, 56, 57, 58, ..., 69, 70, 2, 73, 74, 75, 76, 7, 78, 79, ..., 113, 114, 0, 116, 117,
..., 172, 173, 5, 175, 176, 177, ..., 189, 190, 3, 192, 193, ..., 211, 212, 213, 214, 215, 216,
217, 218, 191, 220, 221, ..., 251, 252, 253, 254, 255]

→ t = (S[i] + S[j]) mod 256

= (S[4] + S[18]) mod 256

= (18 + 55) mod 256

= 73 mod 256

= 73

→ u = S[t]

= S[73]

= 73 ⇒ biner 73 = 1001001

→ c = u ⊕ p[idx]

= u ⊕ p[3]

= u ⊕ 3 ⇒ biner 3 = 110011

$$
\begin{array}{r}
1001001 \\
0110011 \; \oplus \\
\hline
1111010
\end{array}
$$

C = z , desimal z = 122