. Muhammad Fiqih Apriansyah Hakim
E1E1 20 016
Genap

- Algoritma : Key Scheduling Algorithm (KSA)

Kunci : "Saputra1" len (k) = 8
Array s : [0, 1, 2, 3, 4, 5, 6, 7, 8, ... 100, 101, 102, 103, ... 253, 254, 255]

Iterasi pertama → i = 0
j = 0
⇒ j = (j + s [i] + k [i mod len (k)]) mod 256
= (0 + 0 + k [0 % 8]) % 256
= (k [0]) %
= ("s") % 256 ⇒ nilai desimal dari "s" = 115
= 115 % 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 ...
199, 200, 201, 202, 203, 204, 205, ... 250, 251, 252, 253, 254, 255]

- Iterasi ke dua → i = 1 | j = 115
⇒ j = (j + s[i] + k[i % len (k)]) % 256
= (115 + s [i] + k[1 % 8]) % 256
= (115 + 1 + k [1]) % 256
= (116 + "a") % 256 ⇒ desimal dari "a" = 97
= (116 + 97) % 256
= (213 % 25)
j = 213
swap (s [i] . s [j])
swap (s [1] . s [213])
Array s = [115, 213, 2, 3, 4, 5, 6, 7, ... 112, 113, 114, 0, 116 ... 210, 211,
212, 1, 214, ... 250, 251, 253, 254, 255]

Iterasi ketiga → i = 2

j = 213

=> j = (j + s[i] + k[i % len (k)]) % 256
= (213 + s[2] + k[2 % 8] % 256
= (213 + 2 + k[2]) % 256
= (215 + "p") % 256 => desimal dari "p" + 12
= (215 + 112) % 256
= 327 % 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, ... 112, 113, 114, 0, 116 ...
210, 211, 212, 1, 214, ... 250, 251, 252, 253, 254, 255]

Iterasi keempat → i = 3

= 71

=> j = (j + s[i] + k[i % len (k)]) % 256
= (71 + s[3] + k[3 % 8]) % 256
= (71 + 3 + k[3]) % 256
= (74 + "u") % 256 => desimal dari "u" = 11)
= (74 + 117) % 256
= 191 % 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, ... 112, 113, 114, 0, 116, ...
189, 190, 3, 192, ... 210, 211, 212, 1, 214, ... 250, 251, 252, 253, 254, 255]

Iterasi ke lima → i = 4

j = 191

→ j = (j + s[i] + k[i % len (k)]) % 256

= (191 + s[4] + k[4 % 8]) % 256

= (191 + 4 + k[4]) % 256

= (195 + "t") % 256 => desimal "t" = 116

= (195 + 116) % 256

= 311 % 256

j = 55

swab (s[i], s[j])

swab (s[4], s[55])

Array s = [115, 213, 71, 191, 55, 5, 6, 7, 8, ... 53, 54, 4, 56, 57, ... 69, 70, 2, 72, 73, ...
113, 114, 0, 116, 117, ... 189, 190, 3, 192, ... 211, 212, 1, 214, ... 250, 251,
252, 253, 254, 252]

Iterasi ke ~~lima~~ ~~enam~~ → i = 5

j = 55

⇒ j = (j + s[i] + k[i % len (k)]) % 256

= (55 + s[5] + k[5 % 8]) % 256

= (55 + 5 + k[5]) % 256

= (60 + 114) % 256

= 174 % 256

= 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, ... 250, 251, 252, 253, 254, 255]

Iterasi ketujuh → i = 6

j = 174

⇒ j = (j + s[i] + k[i % len (k)]) % 256

= (174 + s[6] + k[6 % 8]) % 256

= (174 + 6 + k[6]) % 256

= (180 + "a") % 256 ⇒ desimal "a" = 97

= (180 + 97) % 256

= 277 % 256

j = 21

swab = (s[i] . s[j])

swab = (s[6] . s[174])

Array s = [115, 213, 71, 191, 55, 174, 21, 7, 8, ... 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, ... 250, 251, 252, 253, 254, 255]

Iterasi kedelapan → i = 7

j = 21

⇒ j = (j + s[i] + k[i % len (k)] % 256

= (21 + s[7] + k[7 % 8]) % 256

= (21 + 7 + k[7] % 256

= (28 + "1") % 256 ⇒ desimal "1" = 49

= (28 + 49) % 256

= 77 % 256

j = 77

swab = (s[i], s[j])

swab = (s[7], s[77])

Array s = [115, 213, 71, 191, 55, 21, 77, 8, ... 19, 20, 6, 22, 23, ... 53, 54, 4, 56, 57, ... 69, 70, 2, 72, 73, 74, 75, 76, 7, 78, ... 113, 114, 0, 116, 117, ... 172, 173, 5, 5, 175, 176, ... 189, 190, 3, 192, 193, ... 211, 212, 1, 214, 215, ... 250, 251, 252, 253, 254, 255].

Algoritma : Pseudo-random Generation Algorithm (PRGA)

Array s = [115, 213, 71, 191, 55, 174, 21, 77, 8, ... 19, 20, 6, 22, 23 ... 53, 54, 4, 56, 57, ...
69, 70, 2, 72, 73, 74, 75, 76, 7, 78 ... 113, 114, 0, 116, 172, ... 211, 212, 1, 214, 215 ...
250, 251, 252, 253, 254, 255]

Plaintoks = "2016"

Iterasi pertama → idx = 0

i = 0

j = 0

⇒ i = (i + 1) % 256

= (0 + 1) % 256

= 1 % 256

= 1

⇒ j = (j + s[i]) % 256

= (0 + s[1]) % 256

= (0 + 213) % 256

= 213

swab (s[i], s[j])

swab (s[1], s[213])

Array s = [115, 1, 71, 191, 55, 174, 21, 77, 8, ... 19, 20, 6, 22, 23, ... 53, 54, 4, 56, 57 ... 69,
70, 2, 72, 73, 74, 75, 76, 7, 78, ... 113, 114, 0, 116, 117, --- 172, 173, 5, 175, 176 ...
189, 190, 3, 192, 193, ... 212, 213, 214, ... 250, 251, 252, 253, 254, 255]

⇒ t = (s[i] + s[j]) % 256

= (s[1] + s[213]) % 256

= (1 + 213) % 256

= 214

⇒ u = s[t]

s[214] = 214 ⇒ biner 214 = 11010110

⇒ c = u ⊕ p[idx]

= u ⊕ p[0]

= u ⊕ "2" ⇒ biner "2" = 110010

= 11010110

$\underline{00110010}$ ⊕

11100100

c = "a" didesimalkan menjadi 228

Iterasi kedua → idx = 1

i = 1
j = 213

⇒ i = (i + 1) % 256
= (1 + 1) % 256
= 2

→ j = (j + s [i]) % 256
= (213 + s [2]) % 256
= (213 + 71) % 256
= 284 % 256
= 28

swab = (s [i] , s [j])
swab = (s [2] , s [28])

Array s = [115, 1, 28, 191, 55, 174, 21, 77, 8, ... 19, 20, 6, 22, 23, ... 26, 27, 71, 29, 30,
53, 54, 4, 56, 57, ... 69, 70, 2, 73, 74, 75, 76, 7, 78, ... 113, 114, 0, 116, 117, ...
172, 173, 5, 175, 176, ... 189, 190, 3, 192, 193, ... 212, 213, 214, 215, ... 250,
251, 252, 253, 254, 255]

⇒ t = (s [i] + s [j]) % 256
= (s [2] + s [28]) % 256
= (28 + 71) % 256
= 99 % 256
= 99

⇒ u = s [t]
= s [99]
= 99 ⇒ biner 99 = 1100011

→ c = u ⊕ p (idx)
= u ⊕ p [1]
= u ⊕ "0" ⇒ biner "0" = 110000
= 1100011
  110000 ⊕
  ———————
  1010011

c = "S" desimal = 83

Iterasi ketiga → idx = 2

i = 2, J = 28

⇒ i = (i + 1) % 256

= (2 + 1) % 256

= 3

swab = (s [i], s [j])

swab = (s [3], s [21, 9])

Array s = [115, 1, 28, 219, 55, 174, 21, 77, 8, … 19, 20, 6, 22, 23, … 26, 27, 71, 29, 30,
53, 54, 4, 56, 57, … 69, 70, 2, 73, 74, 75, 76, 7, 78, 79, … 113, 114, 0, 116,
117, … 172, 173, 5, 175, 176, … 189, 190, 3, 192, 193, … 212, 213, 214, 215, 216,
217, 218, 191, 220, … 253, 254, 255]

⇒ t = (s [i] + s [j]) % 256

= (s [3] + s [219] % 256

= 410 % 256

= 154

⇒ u = s [t]

= s [154]

= 154 ⇒ biner 154 = 10011010

⇒ C = u ⊕ p (idx)

= u ⊕ p [2]

= u ⊕ "1" biner "1" = 110001

= 10011010

$$\begin{array}{r} 10011010 \\ \underline{00110001} \oplus \\ 10101011 \end{array}$$

c = "«" . desimal = 171

Iterasi keempat → idx = 3

i = 3, j = 219

=> i = (i + 1) % 256
= (3 + 1) % 256
= 4

j = (j + s[i]) % 256
= (219 + s[47]) % 256
= (219 + 55) % 256
= 274 % 256
= 18

swab = (s[i], s[j])
swab = (s[4], s[18])

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

t = (s[i] + s[j]) % 256
= (s[4] + s[18]) % 256
= (18 + 55) % 256
= 73

u = s[t]
= s[73]
= 73 => biner 73 = 1001001

c = u ⊕ p[idx]
= u ⊕ p[3]
= u ⊕ "6" => biner "6" = 110110

```
= 1001001
  0110110 ⊕
  -------
  1111111
```

c = "▯" desimal = 127