Nama : Febriansyah
NIM : E1E1 20 068
Kelas : Genap

Tugas Kriptografi
Key Scheduling Algorithm (KSA)

K = Saputra1 => k_0 = s, k_1 = a, k_2 = p, k_3 = u, k_4 = t, k_5 = r, k_6 = a, k_7 = 1

Array S = [0, 1, 2, 3, 4, 5, 6, ..., 245, 246, 247, 248, 249, 250, 251, 252, 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] % 256
= ("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, ..., 119, 200,
201, 202, 203, 204, 205, ..., 250, 251, 252, 253, 254, 255]

Iterasi kedua → i = 1
j = 115
=> = j = (j + S (i) + K [i % len(K)]) % 256
= (115 + S (1) K [1 % 8]) % 256
= (115 + 1 + K [1]) % 256
= (116 + "a") % 256 => desimal dari "a" = 97
= (116 + 97) % 256
= 213 % 256
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, 252, 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" = 112
= (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, 214, ..., 250, 251, 252, 253, 254, 255]

Iterasi keempat -> i=3

j= 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" = 117
= (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, 1, 214, ..., 250, 251,
252, 253, 254, 255]

Iterasi kelima 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
= 55

Swap (s[i], s[j])
Swap (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, 255]

•) Iterasi keenam -> 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 + "r") % 256 => desimal "r" = 114

= (60 + 114) % 256

= 174 % 251

= 174 ~~256~~

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]) % 25

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

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

= [(180 + "97")%256

= (277 % 256

j = 21

swap (s [i] . s [7])

swap (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 + "i") % 256 → desimal "i" = 49
=(28 + 49) % 256
= 77 % 256
J = 77

Swap (S[i] , S[j])
swap (S[7] , S[77])
Array S = [115, 213, 71, 191, 55, 21, 77, 8, ...19, 20, 6, 22, 23, ...53, 54, 56,
56, 57, ...) 69, 70, 2, 72, 73, 74, 55, 76, 7, 78..., 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]

Pseudo-Random Generation Algorithm (PRGA)

Array S = [115, 213, 71, 191, 55, 174, 21, 77, 8, 9, 10, ... 20, 6, 22, ..., 54, 4, 56, ..., 70, 2, 72, 73, 74,
75, 76, 7, 78, ..., 114, 0, 116, ..., 173, ..., 190, 3, 192, ... 212, 1, 214, ..., 254, 255]

Iterasi pertama
i = 0   P = 2047
j = 0

For index = 0 to length (P)-1
For index = 0 to (4)-1 = 0 to (3)
i = (0+1) mod 256
i = 1
j = (j + S[i]) mod 256
j = (0 + 213) mod 256
j = 213
S[i] , S[j]   S[i] = 1   S[j] = 213
= S[1], S[213]   S[213] + S[1] = isi index
= 1 + 213
t = (S[1] + S[213]) mod 256
= 1 + 213 mod 256 = 214
u = S[214]

C = u ⊕ P[0]
= 214 ⊕ 2
= 11010110
00110010 = 228 "ä"
11100100

Iterasi kedua

i = 1   j = 213

For index = 0 to (3)

i = (i + 1) mod 256

i = (1 + 1) mod 256

i = 2

j = (j + S [i]) mod 256

j = (213 + S [2]) mod 256

j = (213 + 71) mod 256 = 284 mod 256

= 28

swap (S [i] , S [j]) = (S [2] , S [28])

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

t = (28 + 71) mod 256 = 99 mod 256

t = 99

u = S [99]

C = u $\oplus$ P [1]

= 99 $\oplus$ 0

= 01100011

$\underline{00110000 \oplus}$ = 83 = S (capital s)

01010011

Iterasi ketiga

I = 2

j = 28

For index = 0 to 3

i = (i + 1) mod 256

= (2 + 1) mod 256

= 3 mod 256

= 3

j = (j + S [i]) mod 256

j = (28 + S [3]) mod 256

j = 219

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

(S [3] , S [219])

t = (S [i] + S [219] mod 256

t = (219 + 191) mod 256 = 410 mod 256

t = 154

y = S [154]

c = y ⊕ P[2]

= 154 ⊕ 6

= 10011010

00110110 ⊕

10101100

= 172 ¬

Iterasi Ke empat

i = 3   j = 219

For index = 0 to (3)

i = (i+1) mod 256

i = 4

j = (j+S [i]) mod 256

j = (219 + S[3]) mod 256

j = (219 + 55) mod 256 = 274 mod 256

j = 18

Swap (S [i] , S [j]) = (S[4] , S [18])

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

t = (18+55) mod 256 = 73 mod 256

t = 73

y = S [73]

c = y ⊕ P [3]

= 73 ⊕ 8

= ~~01001110~~   01001001

~~00111000 ⊕~~   00111000 ⊕

~~00101110~~   01110001

= ~~46~~ .   = 113 q

Hasilnya: 'a' S ¬ q

Kemudian hasil arraynya.