

CS427.

HW1

Zongyan Lu.

1.

1.4 No. Bitwise-AND does not a good encryption method.

Compare with Bitwise-XOR.

The AND operator has 25% create 0
75% create 1.

Bitwise-AND

$0 \& 0 = 0$
 $0 \& 1 = 0$
 $1 \& 0 = 0$
 $1 \& 1 = 1$

But Bitwise-XOR has equal possibility to generate either 0 or 1 uniformly.

Bitwise-XOR.

$0 \oplus 0 = 0$

$0 \oplus 1 = 1$

$1 \oplus 0 = 1$

$1 \oplus 1 = 0$

So, we say otp using XOR operation is special.

And AND-operation does not a good choice for encryption.

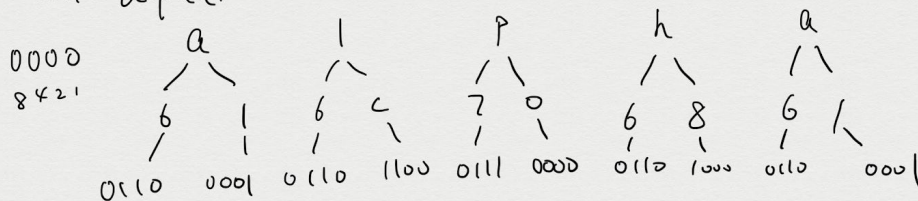
1.7. Based on equation. $\begin{cases} m_1 \oplus k = C_1 \\ m_2 \oplus k = C_2 \end{cases}$ so. $m_1 \oplus k \oplus m_2 \oplus k = C_1 \oplus C_2$
 $\Rightarrow m_1 \oplus m_2 = C_1 \oplus C_2$

$C_1 = 1111\ 1001\ 0111\ 1001\ 1100\ 1100\ 0001\ 0111\ 1000\ 0110$
 $\oplus C_2 = 1111\ 1010\ 0110\ 0111\ 1101\ 1101\ 0000\ 1001\ 1000\ 1000$

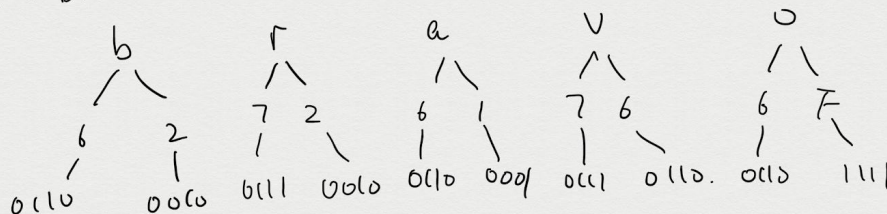
$C_1 \oplus C_2 = 0000\ 0011\ 0001\ 1110\ 0001\ 0001\ 0001\ 1110\ 0000\ 1110$

Case 1.

$m_1 = \text{alpha.}$



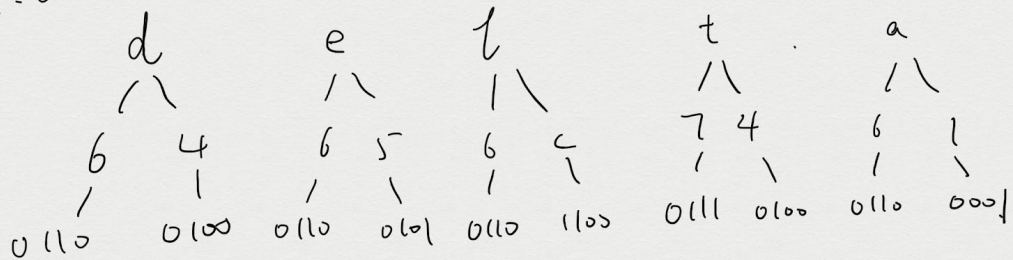
$m_2 = \text{bravo.}$



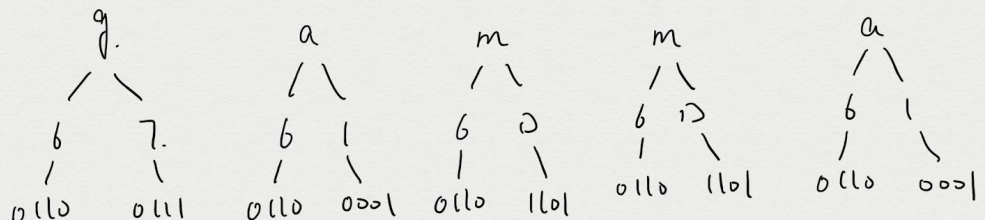
$$\begin{array}{r}
 \text{So. } m_1 \oplus m_2 = \begin{array}{cccccccc}
 0110 & 0001 & 0110 & 1100 & 0111 & 0000 & 0110 & 1000 & 0110 & 0001 \\
 \oplus & 0110 & 0010 & 0111 & 0010 & 0110 & 0001 & 0111 & 0110 & 0110 & 1111 \\
 \hline
 0000 & 0011 & 0001 & 1110 & 0001 & 0001 & 0001 & 1110 & 0000 & 1110
 \end{array} \\
 \therefore m_1 \oplus m_2 = c_1 \oplus c_2 \text{ when } \begin{cases} m_1 = \text{alpha} \\ m_2 = \text{bravo} \end{cases}
 \end{array}$$

Case 2.

$m_1 = \text{delta}$



$m_2 = \text{gamma}$.



$$\begin{array}{r}
 \text{So. } m_1 \oplus m_2 = \begin{array}{cccccccc}
 0110 & 0100 & 0110 & 0101 & 0110 & 1100 & 0111 & 0100 & 0110 & 0001 \\
 \oplus & 0110 & 0111 & 0110 & 0001 & 0110 & 1101 & 0110 & 1101 & 0110 & 0001 \\
 \hline
 0000 & 0011 & 0000 & 0100 & 0000 & 0001 & 0001 & 1001 & 0000 & 0000
 \end{array}
 \end{array}$$

$$C_1 \oplus C_2 = 0000\ 0011\ 0001\ 1110\ 0001\ 0001\ 0001\ 1110\ 0000\ 1110$$

$\therefore m_1 \oplus m_2 \neq C_1 \oplus C_2$ when $\begin{cases} m_1 = \text{delta} \\ m_2 = \text{gamma} \end{cases}$

Hence. The plaintext alpha and bravo. take the same value by xor gate calculation.

And plaintext delta and gamma doesn't.

So The correct encryption messages are.
 $m_1 = \text{alpha}, m_2 = \text{bravo}.$

Based on this, the encryption key.

$$k = m_1 \oplus C_1 = 0110\ 0001\ 0110\ 1100\ 0111\ 0000\ 0110\ 1000\ 0110\ 0001$$

$$\oplus 1111\ 1001\ 0111\ 1001\ 1100\ 1100\ 0001\ 0111\ 1000\ 0110$$

key $\rightarrow 1001\ 1000\ 0001\ 0101\ 1011\ 1100\ 0111\ 1111\ 1110\ 0111$

2.8. $K = \{1, \dots, 9\}$. KeyGen: Enc(k, m):
 $M = \{1, \dots, 9\}$. $k \leftarrow \{1, \dots, 9\}$. return $(k \times m) \% 10$.
 return k
 $C = \mathbb{Z}_{10}$

Based on two One-time Secrecy libraries:

we say.

$$\begin{array}{l} \mathbb{Z} \\ \text{ots-L} \\ \hline \text{EAVESDROP}(m_L, m_R \in \mathbb{Z}.M). \\ K \leftarrow \{1, \dots, 9\}. \\ C \leftarrow (k \times m_L) \% 10 \\ \text{return } C. \end{array}$$

$$\begin{array}{l} \mathbb{Z} \\ \text{ots-R} \\ \hline \text{EAVESDROP}(m_L, m_R \in \mathbb{Z}.M). \\ K \leftarrow \{1, \dots, 9\} \\ C \leftarrow (k \times m_R) \% 10 \\ \text{return } C. \end{array}$$

And we construct a program A.

$$\begin{array}{l} A. \\ \hline C \leftarrow \text{EAVESDROP}(1, 2 \in \mathbb{Z}.M). \\ \text{return } C \stackrel{?}{=} 1 \end{array}$$

Based on the key and message. we get possibility table:

$m \backslash k$	1	2	3	4	5	6	7	8	9
1	1 ₁	2 ₂	3 ₃	4 ₄	5 ₅	6 ₆	7 ₇	8 ₈	9 ₉
2	2 ₂	4 ₄	6 ₆	8 ₈	10 ₀	12 ₂	14 ₄	16 ₆	18 ₈
3	3 ₃	6 ₆	9 ₉	12 ₂	15 ₅	18 ₈	21 ₁	24 ₄	27 ₇
4	4 ₄	8 ₈	12 ₂	16 ₆	20 ₀	24 ₄	28 ₈	32 ₂	36 ₆
5	5 ₅	10 ₀	15 ₅	20 ₀	25 ₅	30 ₀	35 ₅	40 ₀	45 ₅
6	6 ₆	12 ₂	18 ₈	24 ₄	30 ₀	36 ₆	42 ₂	48 ₈	54 ₄
7	7 ₇	14 ₄	21 ₁	28 ₈	35 ₅	42 ₂	49 ₉	56 ₆	63 ₃
8	8 ₈	16 ₆	24 ₄	32 ₂	40 ₀	48 ₈	56 ₆	64 ₄	72 ₂
9	9 ₉	18 ₈	27 ₇	36 ₆	45 ₅	54 ₄	63 ₃	72 ₂	81 ₁

$$\text{So, } \Pr[A \diamond \mathcal{L}_{\text{ots-L}}^{\Sigma} \Rightarrow 1] = 1/9.$$

$$\Pr[A \diamond \mathcal{L}_{\text{ots-R}}^{\Sigma} \Rightarrow 1] = 0/9.$$

$$\text{Which } \Pr[A \diamond \mathcal{L}_{\text{ots-L}}^{\Sigma} \Rightarrow 1] \neq$$

$$\Pr[A \diamond \mathcal{L}_{\text{ots-R}}^{\Sigma} \Rightarrow 1].$$

So, for pick message $m_L=1, m_R=2$

Which comes from secure scheme.

They demonstrate different possibilities

by constructing program A when

linked to the two libraries from

ots.

This is contradicted with the definition of

ots. which the ciphertext has same distribution

possibilities for any input from secure scheme.

So. we conclude this encryption algorithm does

not satisfy the definition of one-time secrecy.