CS60065: Cryptography and Network Security Total: 50 *points*

Assignment 3

Name: Anit Mangal Due: *11.59 pm, Oct 6, 2024*

Note: The basic policies are stated in the course page. Using GPT (or similar tools) to solve problems from the assignment is **strictly prohibited**. Use of any other (possibly online) source(s) **must** be clearly stated in the solution. **Any dishonesty, if caught, will yield zero credits for the entire assignment.**

A. [**Block Ciphers :** $\mathbf{8 + 10 + (3 \times 4) = 30}$ **points**.]

Answers

A.1 Bit representation for the S-box:

$\mathbf{X_1}$	$\mathbf{X_2}$	$\mathbf{X_3}$	$\mathbf{X_4}$	$\mathbf{Y_1}$	$\mathbf{Y_2}$	$\mathbf{Y_3}$	$\mathbf{Y_4}$
0	0	0	0	1	0	0	0
0	0	0	1	0	1	0	0
0	0	1	0	0	0	1	0
0	0	1	1	0	0	0	1
0	1	0	0	1	1	0	0
0	1	0	1	0	1	1	0
0	1	1	0	0	0	1	1
0	1	1	1	1	1	0	1
1	0	0	0	1	0	1	0
1	0	0	1	0	1	0	1
1	0	1	0	1	1	1	0
1	0	1	1	0	1	1	1
1	1	0	0	1	1	1	1
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	0	0	0	0

For an expression $\mathbf{E} = (\oplus_{i=1}^{4}\ a_i\mathbf{X_i}) \oplus (\oplus_{j=1}^{4}\ b_j\mathbf{Y_j})$

Writing count of X, Y assignments which satisfy $\mathbf{E} = 0$ in LAT

	b															
a	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	16	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
1	8	10	6	8	10	8	8	6	4	6	6	8	10	8	4	10
2	8	10	8	10	6	8	6	8	6	8	10	4	4	6	8	10
3	8	8	10	10	8	12	10	6	6	6	8	8	10	6	12	8
4	8	10	8	6	8	10	8	6	10	4	10	8	6	8	6	4
5	8	12	6	6	10	10	8	12	6	12	8	8	8	8	10	6
6	8	8	12	8	10	10	6	10	8	8	8	12	6	6	6	10
7	8	6	6	8	12	6	10	8	8	6	6	8	4	6	10	8
8	8	10	10	8	8	6	6	8	10	8	4	6	10	4	8	6
9	8	8	8	12	10	10	6	10	10	6	6	6	8	12	8	8
A	8	12	10	10	6	6	12	8	8	8	6	10	6	10	8	8
B	8	6	12	6	8	6	8	10	4	6	8	6	8	10	8	6
C	8	8	10	10	12	8	10	6	8	12	10	6	8	8	6	6
D	8	6	8	6	6	12	10	8	8	10	4	6	6	8	6	8
E	8	6	6	12	6	8	8	10	6	8	8	10	8	6	6	4
F	8	8	8	8	8	8	12	12	10	6	10	6	10	6	6	10

A.2 A brute-force attack against single DES takes 2^{56} DES encryption calls in the worst case, since the key is 56 bits long.
Consider DESA,

$$y = \text{DESA}_{K,K'}(x) = \text{DES}_K(x) \oplus K'$$

If we have 2 valid message-ciphertext pairs (x_1, y_1) and (x_2, y_2) under DESA using key (K, K'),

$$\begin{aligned} y_1 \oplus y_2 &= (\text{DES}_K(x_1) \oplus K') \oplus (\text{DES}_K(x_2) \oplus K') \\ &= \text{DES}_K(x_1) \oplus \text{DES}_K(x_2) \end{aligned}$$

We can thus brute-force on K and verify by plugging it into the above equation. This requires 2^{57} DES encryption calls in the worst case.
When we find K, we can find K':

$$\begin{aligned} y_1 &= \text{DES}_K(x_1) \oplus K' \\ \Rightarrow K' &= \text{DES}_K(x_1) \oplus y_1 \end{aligned}$$

This requires one additional call to DES encryption.
Hence, we would need roughly $2^{57} + 1$ DES encryption calls to break DESA encryption using a brute-force attack as compared to roughly 2^{56} encryption calls to break DES, making it around as difficult to break as DES.

A.3 Encrypting $x = 01101\ 11011\ 11010\ 00110$ using $\text{E}(m_1m_2m_3m_4m_5) = (m_2m_5m_4m_1m_3)$

1. Mode ECB
   E(01101) = 11001
   E(11011) = 11110
   E(11010) = 10110
   E(00110) = 00101
   Hence, E(01101 11011 11010 00110) = 11001 11110 10110 00101 with ECB mode

2. Mode CBC
IV $= 11001$, $x_1 = 01101 \Rightarrow y_1 = \text{E}(\text{IV}\oplus x_1) = \text{E}(11001 \oplus 01101) = \text{E}(10100) = 00011$
$y_1 = 00011$, $x_2 = 11011 \Rightarrow y_2 = \text{E}(y_1 \oplus x_2) = \text{E}(11000) = 10010$
$y_2 = 10010$, $x_3 = 11010 \Rightarrow y_3 = \text{E}(y_2 \oplus x_3) = \text{E}(01000) = 10000$
$y_3 = 10000$, $x_4 = 00110 \Rightarrow y_4 = \text{E}(y_3 \oplus x_4) = \text{E}(10110) = 00111$
Hence, E(01101 11011 11010 00110) = 00011 10010 10000 00111 with CBC mode

3. Mode CFB
IV $= 11001$, $x_1 = 01101 \Rightarrow z_1 = \text{E}(\text{IV}) = \text{E}(11001) = 11010$
$\Rightarrow y_1 = z_1 \oplus x_1 = 11010 \oplus 01101 = 10111$
$x_2 = 11011 \Rightarrow z_2 = \text{E}(y_1) = \text{E}(10111) = 01111$
$\Rightarrow y_2 = z_2 \oplus x_2 = 01111 \oplus 11011 = 10100$
$x_3 = 11010 \Rightarrow z_3 = \text{E}(y_2) = \text{E}(10100) = 00011$
$\Rightarrow y_3 = z_3 \oplus x_3 = 00011 \oplus 11010 = 11001$
$x_4 = 00110 \Rightarrow z_4 = \text{E}(y_3) = \text{E}(11001) = 11010$
$\Rightarrow y_4 = z_4 \oplus x_4 = 11010 \oplus 00110 = 11100$
Hence, E(01101 11011 11010 00110) = 10111 10100 11001 11100 with CFB mode

4. Mode OFB
IV $= 11001$, $x_1 = 01101 \Rightarrow z_1 = \text{E}(\text{IV}) = \text{E}(11001) = 11010$
$\Rightarrow y_1 = z_1 \oplus x_1 = 11010 \oplus 01101 = 10111$
$x_2 = 11011 \Rightarrow z_2 = \text{E}(z_1) = \text{E}(11010) = 10110$
$\Rightarrow y_2 = z_2 \oplus x_2 = 10110 \oplus 11011 = 01101$
$x_3 = 11010 \Rightarrow z_3 = \text{E}(z_2) = \text{E}(10110) = 00111$
$\Rightarrow y_3 = z_3 \oplus x_3 = 00111 \oplus 11010 = 11101$
$x_4 = 00110 \Rightarrow z_4 = \text{E}(z_3) = \text{E}(00111) = 01101$
$\Rightarrow y_4 = z_4 \oplus x_4 = 01101 \oplus 00110 = 01011$
Hence, E(01101 11011 11010 00110) = 10111 01101 11101 01011 with OFB mode

B. [Cryptographic Hash Functions & MACs: $(3+4)+(4+3)+(3+3) = 20$ points.]

Answers

B.1 a) Failure probability $= \dfrac{\substack{\text{Number of ways to choose } \mathcal{X}_0 \\ \mathcal{X}_0 \bigcap h^{-1}(y) = \Phi}}{\text{Number of ways to choose } \mathcal{X}_0} = \dfrac{\binom{N - s_y}{q}}{\binom{N}{q}}$

$\Rightarrow$ Success probability $(\epsilon) = 1$ - Failure probability $= 1$ - $\dfrac{\binom{N - s_y}{q}}{\binom{N}{q}}$

b) $\epsilon_y = 1 - \dfrac{\binom{N - s_y}{1}}{\binom{N}{1}} = 1 - \dfrac{N - s_y}{N} = \dfrac{s_y}{N}$

$$
\begin{aligned}
\text{Average success probability} &= \frac{1}{M} \cdot \sum_{y \in \mathcal{Y}} \epsilon_y \\
&= \frac{1}{M} \cdot \sum_{y \in \mathcal{Y}} \frac{s_y}{N} \\
&= \frac{1}{M \cdot N} \sum_{y \in \mathcal{Y}} s_y \\
&= \frac{1}{M \cdot N} \cdot N \\
&= \frac{1}{M}
\end{aligned}
$$

Hence, Average success probability $= \frac{1}{M}$

B.2 a) Assume $x \neq x'$, let $h(x) = h(x')$

Case 1. $|x| = |x'| = n$

$$
\begin{aligned}
&h(x) = h(x') \\
\Rightarrow\ &0||x = 0||x' \\
\Rightarrow\ &x = x'
\end{aligned}
$$

But we had assumed that $x \neq x'$. This is a contradiction. Hence, $h(x) \neq h(x')$

Case 2. $|x| \neq n, |x'| \neq n$

$$
\begin{aligned}
&h(x) = h(x') \\
\Rightarrow\ &1||g(x) = 1||g(x') \\
\Rightarrow\ &g(x) = g(x') \\
\Rightarrow\ &x = x' && (g \text{ is collision resistant})
\end{aligned}
$$

But we had assumed that $x \neq x'$. This is a contradiction. Hence, $h(x) \neq h(x')$

Case 3. $|x| = n, |x'| \neq n$ (Or $|x| \neq n, |x'| = n$)

$$
\begin{aligned}
&h(x) = h(x') \\
\Rightarrow\ &0||x = 1||g(x')
\end{aligned}
$$

This is not possible since the first bit does not match. Hence, $h(x) \neq h(x')$

Hence, h is collision-resistant if g is collision-resistant.

b) Let y be a message digest from $h(x)$.
So, $|y| = n + 1$.
If $y_0 = 0$, we know that $y = 0||x$, from definition of h
$\Rightarrow\ x = y_1 y_2 \ldots y_{n+1}$
For all possible $n + 1$-bit digests, half of the digests are such that $y_0 = 0$.
Hence, h is not preimage-resistant.

B.3 a) The MAC tag $x||h_K(x)$ is secure. This is because it does not allow forgery, and does not give information about K.

b) MAC-and-Encrypt should be avoided. This is because:
Suppose the secure block cipher scheme, CTR, is used. Mallory wants to forge a

message m. It intercepts a cryptogram $E_K(m_0)||H_K(m_0)$, where $E_K(m_0)$ is of the form $IV||C_0$, C_0 is the size of m_0. Here, m_0 is known and its size assumed to be at least that of m.
Mallory computes K as $C_0 \oplus m_0$, which can be truncated to size of m.
Mallory computes $C = m \oplus K$.
Mallory replaces the cryptogram by $IV||C$, causing a successful forgery.