SIM CT 2016 S2 CSCI361 V1 Suggested Answer

Database Management (Singapore Institute of Management)

UNIVERSITY OF WOLLONGONG AUSTRALIA

SIM GLOBAL EDUCATION

SCIT
School of Computing and Information Technology

Family Name .....

First Name .....

Student Number .....

CSCI361

Cryptography & Secure Applications

This paper is for students studying at the Singapore Institute of Management Pte Ltd.

SESSION 2 2016 – CLASS TEST – FULL TIME

(23 May 2016 – 9:30 am – 11:30 am)

Time Allowed: 2 hours

Number of Questions: 8 questions

DIRECTIONS TO CANDIDATES

1. Please attempt all questions as directed. Please answer in **consecutive order**.
2. Please write all answers neatly. Questions must be answered in the examination booklet provided. Please answer each question on *a new page* on the examination booklet. Clearly mark the number of the question attempted.
3. This test question must be submitted with your answer upon submission.
4. This paper is worth 15% of the total marks for the subject.

TEST MATERIALS/AIDS ALLOWED

Simple Non-Programmable Calculator Only

THIS TEST PAPER MUST NOT BE REMOVED FROM THE HALL

Version 1.0

Question 1 (1 mark)

In cryptography, what does a perfect secrecy mean?

Suggested answer:

A cryptosystem has perfect secrecy if the interception of a ciphertext gives the cryptanalyst no information about the underlying plaintext and no information about any future encrypted messages.

Question 2 (1 mark)

What is the relationship between MAC and one-way hash function?

Suggested answer:

A message authentication code (MAC) is an authentication tag known as a checksum that is generated by applying an authentication scheme, together with a secret key, to a message. The receiver of the message can then recompute the MAC and verify its correctness. MAC is very closely related to one-way hash function, which itself is a function that is efficient to calculate but difficult to invert. MAC is the result of application of one-way hash function. Important properties of MAC are:

- Knowing a message and MAC is infeasible or difficult to find another message with the same MAC.
- MACs should be uniformly distributed
- Mac should depend equally on all bits of the message.

These properties are achieved through the implementation of one-way hash function.

Question 3 (1 mark)

Given a system where each character of the English alphabets is encoded using binary, and the true rate of English is 1.5 bits/character, what is the redundancy of English in this system?

Suggested solution:

Let:
D = redundancy of English
R = absolute rate, that is, the maximum number of bits that can be encoded in each character
r = true rate, that is, the average number of bits required to represent the characters of English

$$D = R - r$$
$$= log_2 26 - 1.5$$
$$= \frac{log_{10} 26}{log_{10} 2} - 1.5$$
$$= 4.7 - 1.5 \approx 3.2\ bits$$

Hence the redundancy of English in the system is 3.2 bits.

Question 4 (2 marks)

Alice wishes to send the following message to Bob using the Affine cipher:

I LOVE UOW

Alice and Bob agreed to use a key a = 3 and b = 17. Encrypt the message.

You can assume the Affine cipher used in this encryption uses only 26 alphabetic characters, and you can ignore the spaces.

Suggested answer:

Convert the plaintext to its numerical equivalent using the following mapping:

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

Hence,

I	L	O	V	E	U	O	W
8	11	14	21	4	20	14	22

For each of the integers (8, 11, 14, 21, 4, 20, 14, 22), we compute their ciphertext in numerical form as follows:

$$8: \quad c = aX + b \ (mod\ 26) = (3) \times (8) + 17 \ (mod\ 26) = \ 41 \ mod\ 26 = 15$$
$$11: \ c = aX + b \ (mod\ 26) = (3) \times (11) + 17 \ (mod\ 26) = 50 \ mod\ 26 = 24$$
$$14: \ c = aX + b \ (mod\ 26) = (3) \times (14) + 17 \ (mod\ 26) = \ 59 \ mod\ 26 = 7$$
$$21: \ c = aX + b \ (mod\ 26) = (3) \times (21) + 17 \ (mod\ 26) = 80 \ mod\ 26 = 2$$
$$4: \quad c = aX + b \ (mod\ 26) = (3) \times (4) + 17 \ (mod\ 26) = 29 \ mod\ 26 = 3$$
$$20: \ c = aX + b \ (mod\ 26) = (3) \times (20) + 17 \ (mod\ 26) = 77 \ mod\ 26 = 25$$
$$14: \ c = aX + b \ (mod\ 26) = (3) \times (14) + \ 17 \ (mod\ 26) = 59 \ mod\ 26 = 7$$
$$22: \ c = aX + b \ (mod\ 26) = (3) \times (22) + 17 \ (mod\ 26) = 83 \ mod\ 26 = 5$$

Next, convert the integers (15, 24, 7, 2, 3, 25, 7, 5) back to letters, and we have:

15	24	7	2	3	25	7	5
P	Y	H	C	D	Z	H	F

Hence, the ciphertext is PYHCDZHF.

Question 5 (4 marks)

Compute the following by demonstrating the step-by-step calculation correctly.

(a) Compute gcd(12345, 67890) and find integers x and y such that $12345x + 67890y = \gcd(12345,67890)$.

(b) Compute $3221^{-1} \bmod 4019$

(c) Use the fast exponentiation algorithm described in lecture to determine 25^{166} mod 123.

(d) Given Z_{11}^*, and a primitive element 2, how many primitive elements (generators) are there and what are they?

Suggested answer:

(a) gcd(12345, 67890) = 15 = (11) (12345) + (-2) (67890)

The student is expected to use Extended Euclidean Algorithm to compute as follow:

n1	n2	r	q	a1	b1	a2	b2
67890	12345	6165	5	1	0	0	1
12345	6165	15	2	0	1	1	-5
6165	15	0	411	1	-5	-2	11

(11) (12345) + (-2) (67890) = 15

Hence x = 11, and y = -2.

(b) $3221^{-1} \bmod 4019 = -277 \bmod 4019 = 3742$

The student is expected to use Extended Euclidean Algorithm to compute as follow:

n1	n2	r	q	a1	b1	a2	b2
4019	3221	798	1	1	0	0	1
3221	798	29	4	0	1	1	-1
798	29	15	27	1	-1	-4	5
29	15	14	1	-4	5	109	-136
15	14	1	1	109	-136	-113	141
14	1	0	14	-113	141	222	-277

(c) $25^{166} \bmod 123 = 16 \bmod 123$.

166 = 10100110 (in binary)

	2^7=128	2^6=64	2^5=32	2^4=16	2^3=8	2^2=4	2^1=2	2^0=1
166 =	1	0	1	0	0	1	1	0
$25^{166} \bmod 123$	$25^{128} \bmod 123$	$25^{64} \bmod 123$	$25^{32} \bmod 123$	$25^{16} \bmod 123$	$25^8 \bmod 123$	$25^4 \bmod 123$	$25^2 \bmod 123$	$25^1 \bmod 123$
	37 mod 123	100 mod 123	**10 mod 123**	16 mod 123	37 mod 123	**100 mod 123**	**10 mod 123**	25 mod 123
	$75^{140} \bmod 123 = 37 \times 10 \times 100 \times 10 \bmod 123 =$ **16 mod 123**							

(d) In Z_{11}^*, we can find primitive elements by checking if gcd(i, p-1) = 1 for all i = 1 to 10. If gcd(i, p-1) = 1 and g is a generator (primitive element), then $g^i \bmod p$ is a generator.

gcd(1, 10) = 1
gcd(2, 10) = 2
gcd(3, 10) = 1
gcd(4, 10) = 2
gcd(5, 10) = 5
gcd(6, 10) = 2

gcd(7, 10) = 1
gcd(8, 10) = 2
gcd(9, 10) = 1
gcd(10, 10) = 10
Thus, there are 4 generators, and they are

$2^1 \bmod 11 = 2$,
$2^3 \bmod 11 = 8$,
$2^7 \bmod 11 = 7$, and
$2^9 \bmod 11 = 6$.

Question 6 (2 marks)

What is blind signature scheme? Provide an example how this is achieved using RSA.

Suggested answer:

A signature scheme introduced by David Chaum. This signature scheme allows a person to get a message signed by another party without revealing any information about the message to the other party. In general, this is what happens:

- The requester wants to obtain the signer's signature of message *m*.
- The requester doesn't want to reveal *m* to anyone, including the signer.
- The signer signs *m* blindly, not knowing what they are signing.
- The requester can retrieve the signature.

Example:

Setup:

(i) Bob has (d, e), a (private, public) key pair, and $N=pq$, where p, q are large primes, associated with Bob.
(ii) For a message X, that Alice wants Bob to sign, Alice constructs $\mu = Xr^e mod\ N$, where $r \in_R Z_N^*$ and e is $Bob's$ $public$ key, and sends μ to Bob.
μ is known as the blinded message.
(iii) Bob signs μ using his private key d, and sends his signature $s' = \mu^d\ mod\ N$ back to Alice. The signature s' that Alice receives is Bob's signature on the blinded message.
(iv) Alice retrieves Bob's signature S of X by computing:

$$s = \frac{s}{r} = \frac{\mu^d}{r} = \frac{m^d r^{ed}}{r} = \frac{m^d r}{r} = m^d\ mod\ N$$

Question 7 (2 marks)

We consider a Cipher-Block Chaining Mode (CBC mode) for a block cipher which implements the encryption as $C_i = E(k, M_i \oplus C_{i-1})$ for i > 0 where $M_1, M_2, M_3 \ldots$ are the messages and C_0 is a randomly chosen initial vector.

(i) Explain how decryption is done, and give the mathematical expression for the decryption.
(ii) How does a bit error in the ciphertext influence decryption? (Assume that C_i is obtained corrupted because of a bit error. How does it affect the next decryption steps?)

Suggested answer:

(i) To decrypt, each cipher block is passed through the decryption algorithm. The result of the decryption algorithm is then XORed with the preceding ciphertext block to produce the plaintext block. The plaintext can be recovered from just two adjacent blocks of the ciphertext, thus, decryption can be parallelized.

The decryption expression for the CBC mode described above is as follow:

$$P_i = D(K, C_i) \oplus C_{i-1} \; where \; C_0 = IV \; (Initial \; vector).$$

Since the plaintext can be recovered from just two adjacent blocks of the ciphertext, if a bit error occurs in a ciphertext, this error will affect only two blocks of the plaintext, the plaintext of the current block as well as the next block. This is because the ciphertext of the current block is used to XORed with the preceding ciphertext block to produce the plaintext block. In other words, the error is propagated only to the next block, and not further.

Question 8 (2 marks)

Explain Diffie-Hellman key exchange. On what hard problem does its security depend? Describe the Diffie-Hellman key agreement protocol. Why is Diffie-Hellman susceptible to man-in-the-middle attacks? Name one way to prevent such attacks.

Suggested answer:

Two parties each create a public-key, private-key pair and communicate the public key to the other party. The keys are designed in such a way that both sides can calculate the same unique secret key based on each side's private key and the other side's public key. The key agreement protocol is explained using example as follow:

- Adam and Barbie choose a large random prime *p* and a generator $g \in Z_p$.
- The prime *p* and *g* are publicly known.
- Privately, Adam chooses integer x, a secret known to Adam, and Barbie chooses y, a secret known to Barbie.
- Next Adam sends $g^x \ mod \ p$ to Barbie, and Barbie sends $g^y \ mod \ p$ to Adam.
- Adam and Barbie can now compute the joint key $(g^x)^y = (g^y)^x \ mod \ p$.

Diffie-Hellman key exchange protocol susceptible to man-in-the-middle attacks because any party could generate a Diffie-Hellman key exchange message as there is no identifying secret in the protocol. One-way to prevent man-in-the-middle attack, both parties can add authentication by signing the join key.

END OF TEST