COMP90043 Cryptography and Security
Semester 2, 2022, Workshop Week 2 Solutions

Questions:

1. Modulo Arithmetic. Two integers p and q are said to be congruent modulo n, if (p mod n) = (q mod n). This is written as $p \equiv q$ (mod n). Solve the following pairs of numbers using modulo arithmetic:

   (a) $73 \bmod 23 = 4$

   (b) $-11 \bmod 7 = 3$

   (c) $(-13)^2 \bmod 9 = 7$

   (d) $32 \bmod 19 = 13$

   (e) $(-2)^3 \bmod 17 = 9$

   (f) $(-1) \bmod 19 = 18$

2. Greatest Common Division (GCD) A GCD is defined as the largest number m which divides two numbers p, and q. Find the GCD for the following pairs of numbers using the Euclid's algorithm: Make sure that you understand the process. You should be able to carry out the computations on a new set of numbers. Try creating your examples.

   (a) $GCD(60, 24) = 12$

   (b) $GCD(30, 105) = 15$

   (c) $GCD(1473, 1562) = 1$

3. When considering Data, stored digitally, how would you determine the satisfaction of the following criteria:

   (a) Confidentiality: Preserving authorized restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information.

   (b) Integrity: Guarding against improper information modification or destruction, including ensuring information nonrepudiation and authenticity.

   (c) Availability: Ensuring timely and reliable access to and use of information.

   (d) Authentication: Is the property of being genuine and being able to be verified and trusted.

   (e) Accountability: Is a security goal that requires all actions of an entity to be traced uniquely to that entity.

   (f) Which one of the three (Confidentiality, Integrity or Availability) do you think is the MOST important? Depends on the circumstances but some examples to discuss include (these examples are also relevant to the next question):

      i. students at University of Melbourne and their personal data and transcript information

      ii. wifi-enabled pacemaker or other critical medical device

      iii. online banking and/or buying/selling stock

iv. physical security at an airport (not data but useful to think about threats and attacks)

4. Security Attacks and Threats:

   (a) Define a Security Threat and a Security Attack: A Security Threat is a possible danger that might exploit a vulnerability A Security Attack is an intelligent act that is a deliberate attempt to evade security services and violate the security policy of a system.

   (b) Define the following attacks:

      i. Denial of Service: is an attack which prevents of inhibits the normal use or management of communications facilities.
      ii. Release of Message Contents: is an attack in which the contents of a message or transmission are either directly or indirectly released.
      iii. Message Modification: Is an attack which aims to alter a part or whole of a legitimate message as a means of delaying or reordering in order to produce an unauthorized effect.
      iv. Masquerade: A masquerade takes place when one entity pretends to be a different entity.
      v. Traffic Analysis: is an attack which aims to analyse data and information going across the network in order to infer the details of the message and/or communication.
      vi. Replay: Is an attack that passively captures a data unit and subsequently retransmits it to produce an unauthorized effect.

   (c) From the above, identify which constitute as active attacks and which constitute as passive attacks?

      An active attack is one that involves some modification of the data stream or the creation of a false stream. From the above, the following constitute as active attacks:

      i. Denial of Service
      ii. Message Modification
      iii. Masquerade
      iv. Replay

      A passive attack involves the eavesdropping or monitoring of transmissions with the goal of obtaining information that is being transmitted. From the above, the following constitute as passive attacks:

      i. Release of Message Contents
      ii. Traffic Analysis

COMP90043 Cryptography and Security

Semester 2, 2022, Workshop Week 3 Solutions

Part A

1. What is a cipher? What does it do? And, in general, how does it go about doing this?
2. What is a block cipher and a stream cipher?

   A block cipher is one in which a block of plaintext is treated as a whole and used to produce a ciphertext block of equal length.

   A stream cipher is one that encrypts a digital data stream one bit or one byte at a time.

3. What is a one-time pad? Discuss the practical applicability of the scheme in security?

   A one-time pad is a cryptographic scheme which uses a random key of equal length to the plain text to both encrypt and decrypt the message. While one-time padding offers complete security and is unbreakable, practical applicability is limited due to:

   - The requirement for a large quantity of random keys, in which the guarantee of true randomness of each character within the key is a significant task.
   - Key distribution is a major problem as each message has an associated key of equal length which needs to be known to the sender and receiver.

4. a. What is a symmetric cipher? What are the essential components of a symmetric cipher?

   A symmetric cipher is an enciphering primitive which uses the same cryptographic key to encrypt and decrypt the plaintext, known as the Secret Key, which must only share with the intended receiver. There are five essential components of a symmetric cipher:

   - Plaintext – The original intelligible message or data.
   - Encryption Algorithm – An algorithm which performs various substitutions and transformations on the plaintext.
   - Secret Key – Is an input required for the encryption algorithm and is independent of the plaintext and the algorithm
   - Ciphertext – This is the scrambled message of the plaintext produced by the encryption algorithm as an output.
   - Decryption Algorithm – An algorithm which allows for the receiver to obtain the plaintext back from the ciphertext. For a symmetric cipher, the decryption algorithm is normally the encryption algorithm run in reverse.

   b. What is an asymmetric cipher? How does it different from a symmetric cipher? Cite at least two differences.

   An asymmetric cipher is an enciphering primitive which uses two different cryptographic keys as part of the enciphering process. One key which is used for encryption known as the Public Key which is shared to all senders; and another which is used for decryption known as the Private key which is kept secret by the receiver. Some differences:

   - Symmetric ciphers use only one cryptographic key whereas asymmetric ciphers use two.
   - Symmetric ciphers normally use one algorithm which runs forward for encryption and backwards for decryption. Asymmetric ciphers normally have two different algorithms, one for encryption and another for decryption.

- Symmetric ciphers are normally much faster than asymmetric ciphers.
- Symmetric ciphers are prone to the key sharing problem, asymmetric ciphers were introduced to resolve the key sharing problem.
- Symmetric ciphers are secure with smaller key sizes as the keys are always kept secret. Asymmetric ciphers need much larger key sizes as the public key is always available and can be reverse engineered to obtain the private key.

5. Let's consider cryptographic keys.
   a. What is it and why do we need one?

      A cryptographic key is a unique input to an encryption and decryption algorithm which acts as a seed allowing the algorithm to scramble the plaintext or to obtain the plaintext from the ciphertext. A cryptographic key is needed so as to facilitate the working for the enciphering primitive. For asymmetric ciphers the two keys are derived as two parts of a whole thereby together allowing for the encryption and decryption of the message. For symmetric ciphers the secret key acts as an offset seed which allows the algorithm to make consistent substitutions and transformations on the same message.

   b. List some of the different types of cryptographic keys used in practice.

      Symmetric ciphers use a Secret Key

      Asymmetric ciphers use a Public Key and a Private Key

   c. What are some of the security requirements for storing keys? How is this different when considering both symmetric ciphers and asymmetric ciphers?

      Symmetric ciphers are more prone to key storage requirements in contrast to asymmetric ciphers. As the secret key is the same for both encryption and decryption, symmetric ciphers face the key sharing problem; re how to transfer the secret key from the sender to the receiver without anyone intercepting this transmission. This problem was especially difficult when the sender and receiver were not in the same place at the same time. A good analogy is to imagine a box which is locked by user A and the locked box is then sent to User B. But the key to the box is now sealed in an envelope and mailed to User B. How can user B guarantee that no one intercepted this mail, opened the envelope, copied the key, and then resealed the key in another envelope and mailed it to User B?

      Asymmetric ciphers require the receiver to only keep his private key secret. In order to allow multiple senders to send messages he can broadcast this public key for anyone to obtain. As both the keys form a pair, and are used by different algorithms, an adversary needs to either obtain both keys or spend significant time reverse engineering the public key in order to obtain the private key to compromise the cipher.

      A widely used technique on the internet these days is to use a conjunction of both ciphers. Wherein a server sends a signed public key to a client. The client can then negotiate and generate a secret key and encrypt it using the server's public key and send it back to the server. The server then uses its private key to decrypt the message to obtain the shared key and then both parties can use the shared key to encrypt and decrypt further communications.

Part B

(1) Solve the following problems using Extended Euclid's algorithm using first principles. Make sure that you understand the process.

(a) $3^{-1}\ mod\ 7 =$ *5*

(b) $5^{-1}\ mod\ 13 =$ *8*

(c) $1473^{-1}\ mod\ 1562 =$ *351*

(d) $73^{-1}\ mod\ 127 =$ *87*

$7 = 3 \times 2 + 1$ 7 3 1 -2 3 1 0 1 1 0 1 0 $1 = 7 - 3 \times 2$ This shows the inverse is -2 (or 5 which is $-2\ mod\ 7$)	$13 = 5 \times 2 + 3$ $5 = 3 \times 1 + 2$ $3 = 2 \times 1 + 1$ $1 = 3 - 2 \times 1$ $1 = 3 - (5 - 3 \times 1) \times 1$ $= 3 \times 2 - 5 \times 1$ $1 = (13 - 5 \times 2) \times 2 - 5 \times 1$ $= 13 \times 2 - 5 \times 5$ This shows the inverse is -5 (or 8 which is $-5\ mod\ 13$)
$1562 = 1473 \times 1 + 89$ $1473 = 89 \times 16 + 49$ $89 = 49 \times 1 + 40$ $49 = 40 \times 1 + 9$ $40 = 9 \times 4 + 4$ $9 = 4 \times 2 + 1$ $1 = 9 - 4 \times 2$ $1 = 9 - (40 - 9 \times 4) \times 2$ $= 9 \times 9 - 40 \times 2$ $1 = (49 - 40 \times 1) \times 9 - 40 \times 2$ $= 49 \times 9 - 40 \times 11$ $1 = 49 \times 9 - (89 - 49 \times 1) \times 11$ $= 49 \times 20 - 89 \times 11$ $1 = (1473 - 89 \times 16) \times 20 - 89 \times 11$ $= 1473 \times 20 - 89 \times 331$ $1 = 1473 \times 20 - (1562 - 1473 \times 1) \times 331$ $= 1473 \times 351 - 1562 \times 331$ This shows the inverse is 351	$127 = 73 \times 1 + 54$ $73 = 54 \times 1 + 19$ $54 = 19 \times 2 + 16$ $19 = 16 \times 1 + 3$ $16 = 3 \times 5 + 1$ $1 = 16 - 3 \times 5$ $1 = 16 - (19 - 16 \times 1) \times 5$ $= 16 \times 6 - 19 \times 5$ $1 = (54 - 19 \times 2) \times 6 - 19 \times 5$ $= 54 \times 6 - 19 \times 17$ $1 = 54 \times 6 - (73 - 54 \times 1) \times 17$ $= 54 \times 23 - 73 \times 17$ $1 = (127 - 73 \times 1) \times 23 - 73 \times 17$ $= 127 \times 23 - 73 \times 40$ This shows the inverse is -40 (or 87 which is $-40\ mod\ 127$)

(2) Any number $a \geq 1$ has a unique factorization given by:

$$a = {p_1}^{a_1} {p_2}^{a_2} \cdots {p_n}^{a_n}$$

where $p_1, p_2, \cdots\ p_n$ are the first n primes.

Write an expression for the GCD of two numbers using the above representation of numbers.

$$a = {p_1}^{a_1} {p_2}^{a_2} \cdots {p_n}^{a_n}$$
$$b = {p_1}^{b_1} {p_2}^{b_2} \cdots {p_n}^{b_n}$$
$$GCD(a,b) = \prod_{i=1}^{n} {p_i}^{min(a_i,b_i)}$$

(3) Classical Ciphers

(a) What is a Caesar Cipher?

A Caesar Cipher is a substitution cipher and is also a stream cipher if the block size used is 1. The earliest version of the Caesar Cipher involved the substitution of an alphabet in the plaintext with another alphabet three places down. As an example, A would have the value of 1, and Z would have the value of 26. If the key k=3, then for the plaintext of ABC, the resulting ciphertext would be DEF.

A	B	C	=	A	B	C	=	D	E	F
1	2	3		1+3	2+3	3+3		4	5	6

(b) Explain differences between mono- and poly- alphabetic ciphers.

Mono-alphabetic cipher: Encoding using only one fixed alphabet (when the spaces between words are still there, these are fairly easy to break)

Poly-alphabetic cipher: Encoding using more than one alphabet, switching between them systematically.

(c) If you have a Caesar Cipher with key $k = 4$. Encrypt "MELBOURNE" using the key.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26

$$k = 4$$
$$A \to E, B \to F, C \to G, \cdots Y \to C, Z \to D$$
$$MELBOURNE \to QIPFSYVRI$$

(d) Consider the affine Caesar cipher defined as follows. The encryption function is defined as: $C = E_{[a,b]}(p) = (ap + b)\ mod\ 26$, where p is the plain text and the tuple $[a, b]$ is the key.

(i) How many different keys are possible with the system?

b can take any value from integers modulo 26. However, a should have an inverse, i.e. a should belong to {1,3,5,7,9,11,15,17,19,21,23,25}. So totally there are $12 \times 26 = 312$ keys.

(ii) Derive a decryption function and determine what values of a and b are allowed, if this function exists.

$$p = (c - b)a^{-1}\ mod\ 26$$

COMP90043 Cryptography and Security
Semester 2, 2022, Workshop Week 4 Solutions

Part A

1. Let C_1 and C_2 be two n-bit ciphertexts obtained by encrypting using one-time pad key K on plaintexts M_1 and M_2 respectively. Show that $M_1 \oplus M_2 = C_1 \oplus C_2$. What is the consequence of Known Plaintext attack on the one-time pad encryption?

$$\begin{aligned} C_1 &= M_1 \oplus K \\ C_2 &= M_2 \oplus K \\ C_1 \oplus C_2 &= (M_1 \oplus K) \oplus (M_2 \oplus K) \\ &= M_1 \oplus M_2 \oplus K \oplus K \\ &= M_1 \oplus M_2 \end{aligned}$$

   Ciphertext can be decrypted by using another pair of plaintext and ciphertext encrypted using the same key.

2. The Vernam cipher can be considered as a one-time pad where message and cipher space are English text treated as sequences of integers between 0 and 25 and the $\oplus$ operation is replaced by sum modulo 26. Let $M[i], K[i] \in \{0, 1, \cdots, 25\}, 0 \leq i < n$, then the encryption function can be implemented as:

```
for i = 0 to n-1 do
    C[i] = M[i] + K[i] mod 26
```

   (a) What's the decryption function?
   The decryption of C with the key K can be given by

```
for i = 0 to n-1 do
    M[i] = C[i] - K[i] mod 26
```

   (b) If the length of the key is n, how many different possible keys are there in Vernam cipher?
   26^n

   (c) Encrypt "unimelb" with the key "tuesday".
   u(20) + t(19) = n(13)
   n(13) + u(20) = h(7)
   i(8) + e(4) = m(12)
   m(12) + s(18) = e(4)
   e(4) + d(3) = h(7)
   l(11) + a(0) = l(11)
   b(1) + y(24) = z(25)

(d) What should be the key that decrypts the ciphertext in (c) to "rmituni"?
n(13) - r(17) = w(22)
h(7) - m(12) = v(21)
m(12) - i(8) = e(4)
e(4) - t(19) = l(11)
h(7) - u(20) = n(13)
l(11) - n(13) = y(24)
z(25) - i(8) = r(17)

3. State the condition for perfect secrecy.

$\mathbf{Pr}[\mathbf{M} = \mathbf{x}|\mathbf{C} = \mathbf{y}] = \mathbf{Pr}[\mathbf{M} = \mathbf{x}]$

Part B: Block Cipher Modes

(see next page)

(1) Only the plaintext unit corresponding to the ciphertext character is affected. In OFB method, the bit errors in transmission do not propagate. For example, if a bit error occurs in C_1, only the recovered value of P_1 is affected; subsequent plaintext units are not corrupted.

(2) In some modes, the plaintext does not pass through the encryption function, but is XORed with the output of the encryption function. The math works out that for decryption in these cases, the encryption function must also be used.

(3)(a) If the IVs are kept secret, the 3-loop case has more bits to be determined and is therefore more secure than 1-loop for brute force attacks.

(3)(b) For software implementations, the performance is equivalent for most measurements. One-loop has two fewer XORs per block. Three-loop might benefit from the ability to do a large set of blocks with a single key before switching. The performance difference from choice of mode can be expected to be smaller than the differences induced by normal variation in programming style.
For hardware implementations, three-loop is three times faster than one-loop, because of pipelining. That is: Let P_i be the stream of input plaintext blocks, X_i the output of the first DES, Y_i the output of the second DES and C_i the output of the final DES and therefore the whole system's ciphertext.
In the 1-loop case, we have:

$$X_i = DES(XOR(P_i, C_{i-1}))$$
$$Y_i = DES(X_i)$$
$$C_i = DES(Y_i)$$

where C_0 is the single IV.

If P_1 is presented at $t = 0$ (where time is measured in units of DES operations), X_1 will be available at $t = 1$, Y_1 at $t = 2$ and C_1 at $t = 3$. At $t = 1$, the first DES is free to do more work, but that work will be: $X_2 = DES(XOR(P_2, C_1))$ but C_1 is not available until $t = 3$, therefore X_2 can not be available until $t = 4$, Y_2 at $t = 5$ and C_2 at $t = 6$.

In the 3-loop case, we have:

$$X_i = DES(XOR(P_i, X_{i-1}))$$
$$Y_i = DES(XOR(X_i, Y_{i-1}))$$
$$C_i = DES(XOR(Y_i, C_{i-1}))$$

where X_0, Y_0 and C_0 are three independent IVs.

If P_1 is presented at $t = 0$, X_1 is available at $t = 1$. Both X_2 and Y_1 are available at $t = 4$. X_3, Y_2 and C_1 are available at $t = 3$. X_4, Y_3 and C_2 are available at $t = 4$. Therefore, a new ciphertext block is produced every 1 tick, as opposed to every 3 ticks in the single-loop case. This gives the three-loop construct a throughput three times greater than one-loop construct.

COMP90043 Cryptography and Security
Semester 2, 2022, Workshop Week 5 Solutions

Part B: RSA Exercises

1. Given the parameters below, fill in the blanks accordingly for the relevant RSA

   parameter: p =13 q = 7 n = p.q = 91

   a) Using Euler's Totient Function, calculate

   $\phi(n) = \underline{\phi(91) = \phi(7 \cdot 13) = \phi(7) \cdot \phi(13) = (7-1).(13-1) = 6.12 = 72}$

2. For the RSA algorithm to work, it requires two coefficients – e and d. Where e represents the encryption component (generally the public key) and d represents the decryption component (generally the private key)

   In order to calculate d, we can use Extended Euclidean Algorithm which can be summarized as follows for any a and b such that (a > b).

GCD(a,b) $a = q_1 b + r_1$ $b = q_2 r_1 + r_2$ $r_1 = q_3 r_2 + r_3$ $r_2 = q_4 r_3 + r_4$ … (1) $r_{n-2} = q_n r_{n-1} + r_n$ (2) $r_{n-1} = q_{n+1} r_n + r_{n+1}$, **where $r_{n+1} = 1$ (GCD exists)** (3) $r_n = q_{n+2} r_{n+1} + r_{n+2}$, **where $r_{n+2} = 0$**	Now we can perform a back substitution to get d as follows: From (2) we get $r_{n+1} = 1 = r_{n-1} - q_{n+1} r_n$ We know r_n from (1), so we can substitute $= r_{n-2} - q_{n+1}(r_{n-2} - q_n r_{n-1})$ We continue this for each r while simplifying each step until we can represent the r_{n+1} in terms of b.

a) Suppose $\phi(n) = 72$. For each of the following given values of e, calculate the value of d such that

$$d.e = 1 \bmod \phi(n)$$

e = 5	e = 7
GCD(ϕ(n), e) = GCD(72, 5)	GCD(ϕ(n), e) = GCD(72, 7)
$\phi(n) = 72 = q_1e + r_1 = 14 * 5 + 2$	$\phi(n) = 72 = q_1e + r_1 = 10 * 7 + 2$
$e = 5 = q_2r_1 + r_2 = 2 * 2 + 1$	$e = 7 = q_2r_1 + r_2 = 3 * 2 + 1$
$r_1 = 2 = q_3r_2 + r_3 = 2 * 1 + 0$	$r_1 = 2 = q_3r_2 + r_3 = 2 * 1 + 0$
Back Substitution we get	Back Substitution we get
$1 = [e - q_2r_1]\ \phi(n) = [5 – (2*2)] \bmod \phi(n)$	$1 = [e - q_2r_1]\ \phi(n) = [7 – (3*2)] \bmod \phi(n)$
$1 = [e – q_2\ (\phi(n)- q_1e)]\ \phi(n)$	$1 = [e – q_2\ (\phi(n)- q_1e)]\ \phi(n)$
$= [5 – (2*(72-(14*5)))]\ \phi(n)$	$= [7 – (3*(72-(10*7)))]\ \phi(n)$
$= [5 + (-2*(72-(14*5)))]\ \phi(n)$	$= [7 + (-3*(72-(10*7)))]\ \phi(n)$
$= [5 +(-2*72 + 2*(14*5))]\ \phi(n)$	$= [7 +(-3*72 + 3*(10*7))]\ \phi(n)$
$= [5 + (-2*72 + 28*5)]\ \phi(n)$	$= [7 + (-3*72 + 30*7)]\ \phi(n)$
$= [5 + 28*5 – 2*72]\ \phi(n)$	$= [7 + 30*7 – 3*72]\ \phi(n)$
$= [29*5 – 2*72]\ \phi(n)$	$= [31*7 – 3*72]\ \phi(n)$
From the above if we want to determine	From the above if we want to determine
$d.e = 1 \bmod \phi(n)$	$d.e = 1 \bmod \phi(n)$
where e = 5, then **d = 29**	where e = 7, then **d = 31**

b) Suppose we have two primes p=23 and q=37. For the following e, calculate the value of d such that

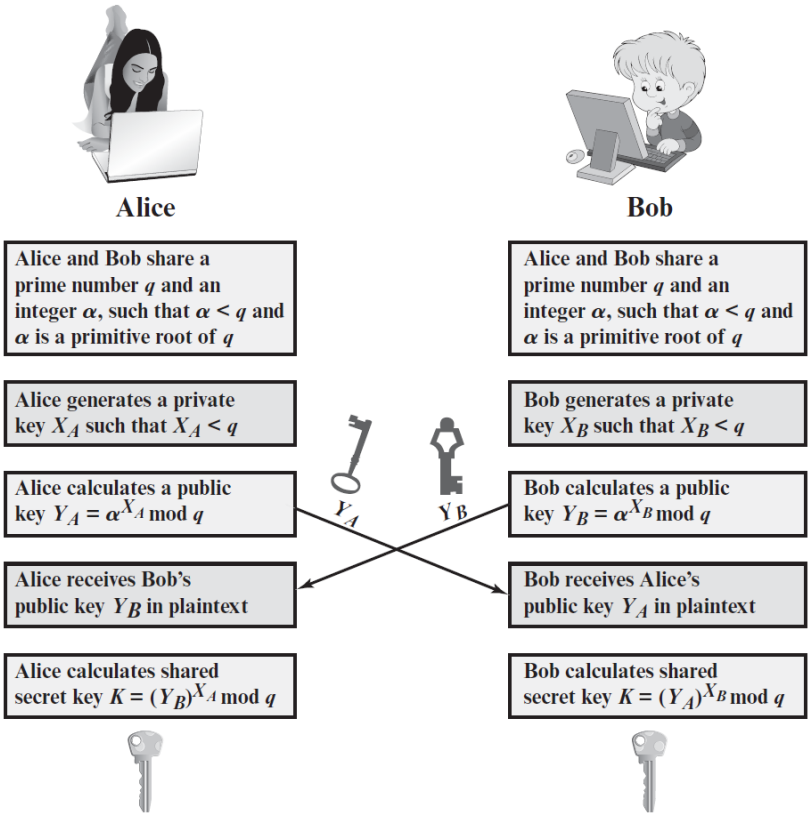
$$d.e = 1 \bmod \phi(n)$$

$n = p.q = 851$ $\phi(n) = 792$

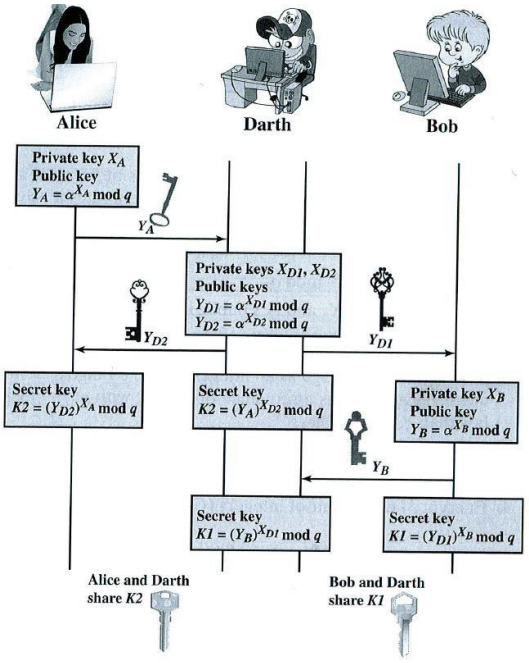
e = 5	e = 61
GCD(ϕ(n), e) = GCD(792, 5)	GCD(ϕ(n), e) = GCD(792, 61)
$\phi(n) = 792 = q_1e + r_1 = 158 * 5 + 2$	$\phi(n) = 792 = q_1e + r_1 = 12 * 61 + 60$
$e = 5 = q_2r_1 + r_2 = 2 * 2 + 1$	$e = 61 = q_2r_1 + r_2 = 1 * 60 + 1$

$r_1 = 2 = q_3 r_2 + r_3 = 2 * 1 + 0$ Back Substitution we get $1 = [e - q_2 r_1]\ \phi(n) = [5 - (2*2)] \bmod \phi(n)$ $1 = [e - q_2 (\phi(n) - q_1 e)]\ \phi(n)$ $= [5 - (2*(792-(158*5)))]\ \phi(n)$ $= [5 + (-2*(792-(158*5)))]\ \phi(n)$ $= [5 + (-2*792 + 2*(158*5))]\ \phi(n)$ $= [5 + (-2*792 + 316*5)]\ \phi(n)$ $= [5 + 316*5 - 2*792]\ \phi(n)$ $= [317*5 - 2*792]\ \phi(n)$ From the above if we want to determine $d.e = 1 \bmod \phi(n)$ where e = 5, then **d = 317**	$r_1 = 60 = q_3 r_2 + r_3 = 60 * 1 + 0$ Back Substitution we get $1 = [e - q_2 r_1]\ \phi(n) = [61 - (1*60)] \bmod \phi(n)$ $1 = [e - q_2 (\phi(n) - q_1 e)]\ \phi(n)$ $= [61 - (1*(792-(12*61)))]\ \phi(n)$ $= [61 + (-1*(792-(12*61)))]\ \phi(n)$ $= [61 + (-1*792 + 1*(12*61))]\ \phi(n)$ $= [61 + (-1*792 + 12*61)]\ \phi(n)$ $= [61 + 12*61 - 1*792]\ \phi(n)$ $= [13*61 - 1*72]\ \phi(n)$ From the above if we want to determine $d.e = 1 \bmod \phi(n)$ where e = 7, then **d = 13**

3. The Diffie-Hellman key exchange algorithm can be defined as follows, show that Diffie-Hellman is subject to a man-in-the-middle attack.



A man in the middle attack is possible as shown in the below figure, where an attacker generated two separate keys and then intercepts the communication between Alice and Bob. The communication is compromised as the attacker uses the generated key to convince Alice or Bob that it belong to the other person. When this key is used to establish the connection, what Alice or Bob are actually doing is establishing a connection with the attacker who then is establishing another simultaneous connection to the other person after reading everything sent on the first connection.



((Image borrowed from Cryptography and Network Security, Stallings, 6th Edition)

4. Given the encryption and decryption formulas for RSA as

   follow:

   $C = M^e \ mod \ n$

   $M = C^d \ mod \ n = (M^e)^d \ mod \ n = M^{ed} \ mod \ n$

   Perform encryption and decryption for the given values of p, q, e and M

$p = 3;\ q = 13;\ e = 5;\ M = 10;$ $n = 39;\ \varphi(n) = 24; d = 5;$ $C = M^e \ mod \ n = 10^5 \ mod \ 39 = 4;$ $M = C^d \ mod \ n = 4^5 \ mod \ 39 = 10;$	$p = 5;\ q = 7;\ e = 7;\ M = 12;$ $n = 35;\ \varphi(n) = 24; d = 7;$ $C = M^e \ mod \ n = 12^7 \ mod \ 35 = 33;$ $M = C^d \ mod \ n = 33^7 \ mod \ 35 = 12;$
$p = 11;\ q = 7;\ e = 11;\ M = 7;$ $n = 77;\ \varphi(n) = 60; d = 11;$ $C = M^e \ mod \ n = 7^{11} \ mod \ 77 = 7;$ $M = C^d \ mod \ n = 7^{11} mod \ 77 = 7;$	

5. In a public-key system using RSA, you intercepted the cipher text $C = 8$ sent to a user whose public key is e = 13; n = 33. What is the plaintext M?

   To show this, note that we know that $n = 33$, which has only two prime dividers. Therefore, $p = 3$ and $q = 11$. $\varphi(n) = 2 \times 10 = 20$. Using the Extended Euclidean Algorithm, d, the multiplicative inverse of $e \ mod \ \varphi(n) = 11 \ mod \ 20$, is found to be 17. Therefore, we can determine M to be $M = C^d \ mod \ n = 8^{17} \ mod \ 33 = 2$.

COMP90043 Cryptography and Security
Semester 2, 2022, Workshop Week 6 Solutions

Revision:

1. Perform encryption and decryption using the RSA algorithm, as in Figure 9.5 (of the textbook), for the following:

   (a) p = 3; q = 11, e = 7; M = 5
   $n = 33; \phi(n) = 20; d = 3$
   $C = 5^7 \bmod 33 = 14$
   $M = 14^3 \bmod 33 = 5$

   (b) p = 5; q = 11, e = 3; M = 9
   $n = 55; \phi(n) = 40; d = 27$
   $C = 9^3 \bmod 55 = 14$
   $M = 14^{27} \bmod 55 = 9$

   (c) p = 7; q = 11, e = 17; M = 8
   $n = 77; \phi(n) = 60; d = 53$
   $C = 8^{17} \bmod 77 = 57$
   $M = 57^{53} \bmod 77 = 8$

   (d) p = 11; q = 13, e = 11; M = 7
   $n = 143; \phi(n) = 120; d = 11$
   $C = 7^{11} \bmod 143 = 106$
   $M = 106^{11} \bmod 143 = 7$

   (e) p = 17; q = 31, e = 7; M = 2
   $n = 527; \phi(n) = 480; d = 343$
   $C = 2^7 \bmod 527 = 128$
   $M = 128^{343} \bmod 527 = 2$

Questions:

1. State Fermat's and Euler's theorems. Using these two theorems simplify the following equations.

   Fermat's: if p is prime, then for any integer a,

   $$a^p = a \pmod{p}$$

   Euler's: if a and n are coprime, then

   $$a^{\phi(n)} = 1 \pmod{n}$$

   (a) $4^{12} \pmod{21} = 1 \pmod{21}$
   (b) $2^{22} \pmod{23} = 1 \pmod{23}$
   (c) $3^{17} \pmod{17} = 3 \pmod{17}$
   (d) $5^{35} \pmod{17} = 5^3 \pmod{17} = 6 \pmod{17}$
   (e) $73^{10001} \pmod{101} = 73 \pmod{101}$

2. Solve for x satisfying the following simultaneous congruences:

   $$x \equiv 7 \pmod{11},$$
   $$x \equiv 9 \pmod{13}.$$

   7 mod 11 is congruent to $18, 29, 40, 51, 62, 73, 84, 95, 106, 117, 128, 139$. 9 mod 13 is congruent to $22, 35, 48, 61, 74, 87, 100, 113, 126, 139$. Hence, $x = 139$.

   Let $a_1 = 7$ and $a_2 = 9$; $n_1 = 11$ and $n_2 = 13$. There exists m_1 and m_2 such that $m_1 n_1 + m_2 n_2 = 1$. Using Extended Euclidean algorithm, we find $m_1 = 6$ and $m_2 = -5$. Finally, $x = a_1 m_2 n_2 + a_2 m_1 n_1 = 7 \times -5 \times 13 + 9 \times 6 \times 11 = 139$.

3. Solve for x satisfying the following simultaneous congruences:

$$x \equiv 2 \pmod 3,$$
$$x \equiv 3 \pmod 5,$$
$$x \equiv 2 \pmod 7.$$

$x = -82 \pmod{105} = 23 \pmod{105}$

4. Assume that Alice chooses two primes 43 and 47 to construct her RSA key prime factors. Help her to set up public and private keys and demonstrate encryption and decryption with an example. Choose the smallest possible exponent for the public key.

   $n = 2021; \phi(n) = 1932.$
   Smallest (non-trivial) e is 5.
   Therefore, $5 \times d = 1 \bmod 1932; d = 773.$

   Suppose we have $M = 313$.
   For encryption, $C = 313^5 \bmod 2021 = 464$.
   For decryption, $M = 464^{773} \bmod 2021 = 313$.

6. $GCD(m, n)$ gives you either p or q.

7. Explain how you can use RSA encryption function to construct a digital signature scheme.

   - Public Key: $< n, e >$
   - Private Key: $< n, d >$
   - Hash Function: $H(m)$
   - Compute Signature $s = H(M)^d \bmod n$; $[M, s]$ form message signature pair.
   - Verification Algorithm: If $H(M) == s^e \bmod n$ then accept the signature else declare verification failure.

8. With RSA, discuss how the concept of Blinding can be implemented?

   RSA allows the incorporation of Blinding in two ways:

   - By allowing the multiplication of the plaintext message with an arbitrary number before performing exponentiations which allows the operations to be performed not on the real input or the real output. Not all functions can use this approach. One example is when Alice wants Bob to return to her the result of a function F which only Bob has access to. But Alice doesn't want Bob to know the input M. By using blinding, Alice blinds, encrypts, and sends M' to Bob. Bob then calculates and returns the value of $F(M')$. Alice then decrypts $F(M')$ and reverses the blinding to get $F(M)$.
   - RSA supports Blind Signatures, wherein similar to the above concept; the content of the passed message is oblivious to the person signing it. This is applicable when the authenticity of the message needs validation from an entity who did not write the message.

COMP90043 Cryptography and Security

Semester 2, 2022, Workshop Week 7 Solutions

1. **What are the advantages of using Hash functions in digital signatures?**

   - You can sign arbitrary long messages.
   - Can be used to build authentication mechanisms (see textbook / slides).
   - Assuring integrity of messages.

2. **Explain how you can use RSA encryption function to construct a digital signature scheme.**

   Public Key Parameters: (n, e)

   Private Key parameter: (n, d)

   Hash Function: (H)

   Signature is generated as $S = (H(M)^d) \bmod n$; [M, s] form message signature pair.

   Verification logic: If $H(M) == ((s^d) \bmod n)$ then accept the signature otherwise declare verification failure.

3. **What characteristics are needed in a secure hash function?**

   1. H can be applied to a block of data of any size.
   2. H produces a fixed-length output.
   3. H(x) is relatively easy to compute for any given x, making both hardware and software implementations practical.
   4. For any given value h, it is computationally infeasible to find x such that H(x) = h. This is sometimes referred to in the literature as the one-way property.
   5. For any given block x, it is computationally infeasible to find $y \neq x$ with H(y) = H(x).
   6. It is computationally infeasible to find any pair (x, y) such that H(x) = H(y).

4. **What is the difference between weak and strong collision resistance?**

   Weak collision resistance: Property 5 of answer in previous question;

   Strong collision resistance: Property 6 of answer in previous question.

5. **Is it possible to use a hash function to construct a DES like block cipher?**

   If you examine the structure of a single round of DES, you see that the round includes a one-way function, f, and an XOR: $R_i = L_{i-1}\ f(R_{i-1}, K_i)$

   For DES, the function f maps a 32-bit R and a 48-bit K into a 32-bit output (see the slides/text for details). That is, it maps an 80-bit input into a 32-bit output. This is clearly a one-way function. Any hash function that produces a 32-bit output could be used for f. The demonstration in the text that decryption works is still valid for any one-way function f.

6. **Explain the birthday paradox. What is the main implication of this for hash function?**

   The problem for adversary in collision resistant attack is to produce two messages x and y which give same hash value, i.e. H(x) = H(y).

   Birthday paradox is not a paradox! If we choose random variables from a uniform distribution in the range 0 to N-1, then the probability that a repeated element is encountered exceeds 0.5 after about $\sqrt{n}$ choices have been made.

   You can see Appendix 1A of Chapter 11 of the textbook for precise derivation of this result.

   Hence for m-bit hash, if we pick messages at random, we can expect to find two messages with the same hash value with about $\sqrt{2^m}$ = $2^{m/2}$ attempts. Hence to break the collision resistant property of an m bit hash function, the effort for adversary is upper bounded by a quantity approximately $2^{m/2}$.

   This implies that m should be chosen reasonably high.

7. **Name three important hash functions used in practice.**

   MD4, MD5, SHA-1/2/3.

8. **Discuss how the security of the hash functions depends on the length of the hash.**

   (Discuss Brute force attack and birthday attacks and show how they influence the decision on length of the hash.) 32-bit hash is trivial to break. The length of the hash should be at least 160 bits or more.

9. **Why CRC checksum cannot be used as a secure hash function?**

   Collisions are easy to find and does not satisfy one-way property requirement of hash functions.

10. **What is a Timing Attack? How can Timing Attacks be prevented?**

    This is a very dangerous attack side channel attack, as it factors in the time complexity involved in deciphering a message by the intended receiver in order to be able to recover the private key.

    The following are examples of approaches which can be used to prevent timing attacks:

    - **Constant Exponentiation Time:** Requires the modifications of algorithms so as to ensure that all exponentiation operations take a fixed amount of time to execute before producing a result. This can be detrimental on performance however.
    - **Random Delay:** Furthers on the exponentiation time problem by introducing random delay periods during each exponentiation operation to throw off an attacker from knowing how long the operation actually took. Offers better performance than constant exponentiation time operations.
    - **Blinding:** Involves the multiplication of the plaintext by a random integer prior to performing any exponentiation operations on it.

COMP90043 Cryptography and security

Semester 2, 2022, Workshop Week 8 Solutions

1. What is a message authentication code?
   It is an authenticator that is a cryptographic function of both the data to be authenticated and a secret key.

2. What types of attacks are addressed by message authentication?
   Masquerade: Insertion of messages into the network from a fraudulent source. This includes the creation of messages by an opponent that are purported to come from an authorized entity. Also included are fraudulent acknowledgments of message receipt or non-receipt by someone other than the message recipient.

   Content modification: Changes to the contents of a message, including insertion, deletion, transposition, and modification.

   Sequence modification: Any modification to a sequence of messages between parties, including insertion, deletion, and reordering.

   Timing modification: Delay or replay of messages. In a connection-oriented application, an entire session or sequence of messages could be a replay of some previous valid session, or individual messages in the sequence could be delayed or replayed. In a connectionless application, an individual message (e.g., datagram) could be delayed or replayed.

3. What is the main difference between hash functions and Message Authentication codes?
   A hash function, by itself, does not provide message authentication. A secret key must be used in some fashion with the hash function to produce authentication. A MAC, by definition, uses a secret key to calculate a code used for authentication.

4. Discuss the following scenarios for using MACs for implementing authentication and confidentiality discussed in lectures.

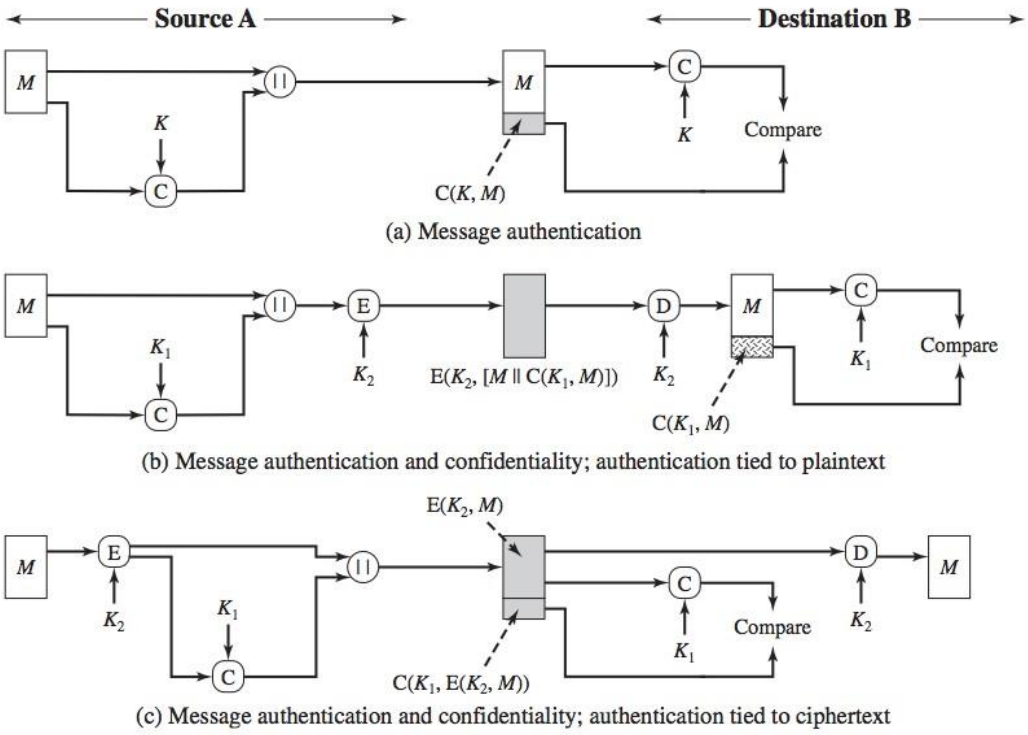


Figure 12.4 Basic Uses of Message Authentication code (MAC)

5. List two disputes that can arise in the context of message authentication.
   Suppose that John sends an authenticated message to Mary. The following disputes that could arise:

   - Mary may forge a different message and claim that it came from John. Mary would simply have to create a message and append an authentication code using the key that John and Mary share.
   - John can deny sending the message. Because it is possible for Mary to forge a message, there is no way to prove that John did in fact send the message.

6. What are the properties a digital signature should have?

   - It must be able to verify the author and the date and time of the signature.
   - It must be able to authenticate the contents at the time of the signature
   - The signature must be verifiable by third parties, to resolve disputes.

7. What are some threats associated with a direct digital signature scheme?
   - The validity of the scheme depends on the security of the sender's private key. If a sender later wishes to deny sending a particular message, the sender can claim that the private key was lost or stolen and that someone else forged his or her signature.

   - Another threat is that some private key might actually be stolen from X at time T. The opponent can then send a message signed with X's signature and stamped with a time before or equal to T.

8. List ways in which secret keys can be distributed to two communicating parties.

   For two parties A and B, key distribution can be achieved in a number of ways, as follows:

   1. A can select a key and physically deliver it to B.

   2. A third party can select the key and physically deliver it to A and B.

   3. If A and B have previously and recently used a key, one party can transmit the new key to the other, encrypted using the old key.

   4. If A and B each has an encrypted connection to a third party C, C can deliver a key on the encrypted links to A and B.

9. What is the difference between a session key and a master key?

   A session key is a temporary encryption key used between two principals.
   A master key is a long-lasting key that is used between a key distribution center and a principal for the purpose of encoding the transmission of session keys. Typically, the master keys are distributed by noncryptographic means.

10. What is a nonce?

    A nonce is a value that is used only once, such as a timestamp, a counter, or a random number; the minimum requirement is that it differs with each transaction.

11. Explain the problems with key management and how it affects symmetric cryptography?

    The primary weakness of symmetric encryption algorithms is keeping the single key secure. Known as key management, it poses a number of significant challenges. If a user wants to send an encrypted message to another using symmetric encryption, he must be sure that she has the key to decrypt the message. How should the first user get the key to the second user? He would not want to send it electronically through the Internet, because that would make it vulnerable to eavesdroppers. Nor can he encrypt the key and send it, because the recipient would need some way to decrypt the key. And if he can even get the get securely to the user, how can be he certain that an attacker has not seen

the key on that person's computer? Key management is a significant impediment to using symmetric encryption.

COMP90043 Cryptography and Security
Semester 2, 2022, Workshop Week 9 Solutions

Symmetric Key Distribution Protocol

1. Consider a variation of the symmetric key distribution protocol discussed in the lecture involving n users and a KDC. Here every user decides to generate random number themselves for the communication they seek to start. All users share a master key with the KDC, all communications can be observed by all users.
   The steps are as follows:

   (a) A generates a random session key K_s and sends to the KDC his identity ID_A, destination ID_B, and $E(K_A, K_s)$.

   (b) KDC responds by sending $E(K_B, K_s)$ to A.

   (c) A sends $E(K_s, M)$ together with $E(K_B, K_s)$ to B.

   (d) B knows K_B, thus decrypts $E(K_B, K_s)$, to get K_s and will subsequently use K_s to decrypt $E(K_s, M)$ to get M.

   Is this secure?

   It's not secure. Consider the following steps:
   An attacker Z could send to the server the source identity ID_A, the destination ID_Z (his own), and $E(K_A, K_s)$, as if A wanted to send Z a message encrypted under the same key K_s as A did with B.
   The server will respond by sending $E(K_Z, K_s)$ to A which could be intercepted by Z.
   Because Z knows his own key K_Z, he can decrypt $E(K_Z, K_s)$, thus getting his hands on K_s that can be used to decrypt $E(K_s, M)$ and obtain M.

2. Consider the following protocol, designed to let A and B decide on a fresh, shared session key K_s. We assume that they already share a long-term key K_{AB}.

$$
\begin{aligned}
A \rightarrow B &: ID_A, N_A \\
B \rightarrow A &: E(K_{AB}, [N_A, K_s]) \\
A \rightarrow B &: E(K_s, N_A)
\end{aligned}
$$

   (a) Why would A and B believe after the protocol ran that they share K_s with each other?

   A believes that she shares K_s with B since her nonce came back in message 2 encrypted with a key known only to B (and A).
   B believes that he shares K_s with A since N_A was encrypted with K_s, which could only be retrieved from message 2 by someone who knows K_{AB} (and this is known only by A and B).

(b) Why would they believe that this shared key K_s is fresh?

A believes that K_s is fresh since it is included in message 2 together with N_A (and hence message 2 must have been constructed after message 1 was sent).
B believes (indeed, knows) that K_s is fresh since He chose it himself.

(c) Assume now that A starts a run of this protocol with B. However, the connection is intercepted by the adversary C. Show how C can start a new run of the protocol using reflection, causing A to believe that she has agreed on a fresh key with B (in spite of the fact that he has only been communicating with C). Thus, in particular, the belief in (a) is false.

Consider the following interleaved runs of the protocol:

$$\begin{aligned} A \to C :& ID_A, N_A \\ & C \to A : ID_B, N_A \\ & A \to C : E(K_{AB}, [N_A, K_s]) \\ C \to A :& E(K_{AB}, [N_A, K_s]) \\ A \to C :& E(K_s, N_A) \end{aligned}$$

C cannot encrypt A's nonce, so he needs to get help with message 2. He therefore starts a new run with A, letting A do the encryption and reflecting the reply back. Note that C cannot decrypt any further message from A, nor sending any message to A, but A will accept the unprimed protocol run and believe that B is present.

(d) Propose a modification of the protocol that prevents this attack.

To prevent the attack, we need to be more explicit in the messages. For example, by changing message 2 to include both the sender and receiver: $E(K_{AB}, [ID_A, ID_B, N_A, K_s])$.

Key Management and Distribution

1. Discuss four methods which are used in distributing public keys.

   Public announcement
   Publicly available directory
   Public-key authority
   Public-key certificates

2. What are the essential ingredients of a public-key directory?

   (a) The authority maintains a directory with a name, public key entry for each participant.

   (b) Each participant registers a public key with the directory authority. Registration would have to be in person or by some form of secure authenticated communication.

(c) A participant may replace the existing key with a new one at any time, either because of the desire to replace a public key that has already been used for a large amount of data, or because the corresponding private key has been compromised in some way.

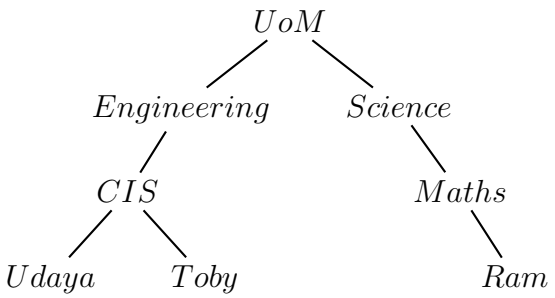
(d) Periodically, the authority publishes the entire directory or updates to the directory. For example, a hard-copy version much like a telephone book could be published, or updates could be listed in a widely circulated newspaper.

(e) Participants could also access the directory electronically. For this purpose, secure, authenticated communication from the authority to the participant is mandatory.

3. What is a chain of certificates? What are forward and reverse certificates?

   A chain of certificates consists of a sequence of certificates created by different certification authorities (CAs) in which each successive certificate is a certificate by one CA that certifies the public key of the next CA in the chain.
   Forward Certificates: Certificates of X generated by other CAs.
   Reverse Certificates: Certificates generated by X that are the certificates of other CAs.

4. For the following hierarchy, what is the chain of certificates that user "Udaya" needs to obtain in order to establish a certificate path to "Ram"? You can use X.509 conventions for the certificate chain, for example the certificate for "Udaya" by CA "CIS" is represented as CIS«Udaya».



   CIS«Engineering» Engineering«UoM» UoM«Science» Science«Maths» Maths«Ram»

COMP90043 Cryptography and Security
Semester 2, 2022, Workshop Week 11 Solutions

1. What are differences between $\mathbf{GF}(8)$ and $\mathbf{Z}_8$?

   $\mathbf{GF}(8)$ is a finite field, represented as polynomials over $\mathbf{GF}(2)$ (binary field) and of characteristic 2. Whereas Z_8 is a is a finite ring. All non-zero elements of $\mathbf{GF}(8)$ have inverses. Some non-zero elements in $\mathbf{Z}_8$ does not have inverse.

2. Describe the conditions under which $\mathbf{GF}(m)$ and $\mathbf{Z}_m$ are identical.

   They are identical when m is a prime number.

3. For any finite field of size p^k, p is a prime number, k is an integer ≥ 1, $a \in \mathbf{GF}(p^k)$ and $a \neq 0$, we have
   $$a^{p^k-1} = 1.$$
   Use this result to derive a function for determining inverse of an element in $\mathbf{GF}(p^k)$.

   As $a^{p^m-1} = 1$, a^{p^m-2} is inverse of a, because $aa^{p^m-2} = a^{p^m-1} = 1$.

4. Use the irreducible polynomial $x^4 + x + 1$ to create a table for the finite field $GF(16)$.

i	Elements:x^i	As Polynomials	As Vectors	Multiplicative Order
$-\infty$	0	0	$[0,0,0,0]$	–
0	1	1	$[0,0,0,1]$	1
1	x	x	$[0,0,1,0]$	15
2	x^2	x^2	$[0,1,0,0]$	15
3	x^3	x^3	$[1,0,0,0]$	5
4	x^4	$x+1$	$[0,0,1,1]$	15
5	x^5	x^2+x	$[0,1,1,0]$	3
6	x^6	x^3+x^2	$[1,1,0,0]$	5
7	x^7	x^3+x+1	$[1,0,1,1]$	15
8	x^8	x^2+1	$[0,1,0,1]$	15
9	x^9	x^3+x	$[1,0,1,0]$	5
10	x^{10}	x^2+x+1	$[0,1,1,1]$	3
11	x^{11}	x^3+x^2+x	$[1,1,1,0]$	15
12	x^{12}	x^3+x^2+x+1	$[1,1,1,1]$	5
13	x^{13}	x^3+x^2+1	$[1,1,0,1]$	15
14	x^{14}	x^3+1	$[1,0,0,1]$	15
15	x^{15}	1	$[0,0,0,1]$	1

   (a) Complete the missing entries in the table. (see table above)

   (b) Determine multiplicative order of elements. (see table above)

   (c) What's the multiplicative inverse of $x^3 + x^2$?
   $x^3 + x$

5. Derive the verification equations of the ElGamal signature using the defining equations of signing.

   Read the slides and first consider the signing equation. Then consider taking a^{th} power on both sides of the signing equation and simplifying the equation using public parameters.

6. Discuss Elgamal digital signature scheme with an example. Say, for $q = 19$ and $\alpha = 13$, $m = 7$, calculate the signature and verify it.

   $q = 19$, $\alpha = 13$, $m = 7$
   Lets choose $X_A = 12$
   Then $Y_A = \alpha^{X_A} \bmod q = 13^{12} \bmod 19 = 7$
   So Private key = {12}, Public key = {19,13,7}
   Lets choose $K = 5$, which is relative prime to $q-1$ that is 18. Using extended gcd algorithm, we can calculate K^{-1} to be 11.
   Then, $S_1 = \alpha^K \bmod q = 13^5 \bmod 19 = 14$, and
   $S_2 = K^{-1}(m - X_A S_1) \bmod (q-1) = 11$ (7 - 12 * 14) mod 18 = 11
   So the signature for this message is {14,11}
   Let's very this now at the receivers end
   $V_1 = \alpha^m \bmod q = 13^7 \bmod 19 = 10$
   and $V_2 = (Y_A)^{S_1}(S_1)^{S_2} \bmod q = 7^{14}14^{11} \bmod 19 = 10$

13.2 ELGAMAL DIGITAL SIGNATURE SCHEME

Before examining the NIST Digital Signature Algorithm, it will be helpful to understand the Elgamal and Schnorr signature schemes. Recall from Chapter 10, that the Elgamal encryption scheme is designed to enable encryption by a user's public key with decryption by the user's private key. The Elgamal signature scheme involves the use of the private key for digital signature generation and the public key for digital signature verification [ELGA84, ELGA85].

Before proceeding, we need a result from number theory. Recall from Chapter 2 that for a prime number q, if α is a primitive root of q, then

$$\alpha, \alpha^2, \ \ldots \ , \alpha^{q-1}$$

are distinct (mod q). It can be shown that, if α is a primitive root of q, then

1. For any integer m, $\alpha^m \equiv 1 \pmod{q}$ if and only if $m \equiv 0 \pmod{q-1}$.
2. For any integers, i, j, $\alpha^i \equiv \alpha^j \pmod{q}$ if and only if $i \equiv j \pmod{q-1}$.

As with Elgamal encryption, the global elements of **Elgamal digital signature** are a prime number q and α, which is a primitive root of q. User A generates a private/public key pair as follows.

1. Generate a random integer X_A, such that $1 < X_A < q - 1$.
2. Compute $Y_A = \alpha^{X_A} \bmod q$.
3. A's private key is X_A; A's pubic key is $\{q, \alpha, Y_A\}$.

To sign a message M, user A first computes the hash $m = \mathrm{H}(M)$, such that m is an integer in the range $0 \leq m \leq q - 1$. A then forms a digital signature as follows.

1. Choose a random integer K such that $1 \leq K \leq q - 1$ and $\gcd(K, q - 1) = 1$. That is, K is relatively prime to $q - 1$.
2. Compute $S_1 = \alpha^K \bmod q$. Note that this is the same as the computation of C_1 for Elgamal encryption.
3. Compute $K^{-1} \bmod (q - 1)$. That is, compute the inverse of K modulo $q - 1$.
4. Compute $S_2 = K^{-1}(m - X_A S_1) \bmod (q - 1)$.
5. The signature consists of the pair (S_1, S_2).

Any user B can verify the signature as follows.

1. Compute $V_1 = \alpha^m \bmod q$.
2. Compute $V_2 = (Y_A)^{S_1}(S_1)^{S_2} \bmod q$.

The signature is valid if $V_1 = V_2$. Let us demonstrate that this is so. Assume that the equality is true. Then we have

$\alpha^m \bmod q = (Y_A)^{S_1}(S_1)^{S_2} \bmod q$	assume $V_1 = V_2$
$\alpha^m \bmod q = \alpha^{X_A S_1}\alpha^{KS_2} \bmod q$	substituting for Y_A and S_1
$\alpha^{m - X_A S_1} \bmod q = \alpha^{KS_2} \bmod q$	rearranging terms
$m - X_A S_1 \equiv KS_2 \bmod (q - 1)$	property of primitive roots
$m - X_A S_1 \equiv KK^{-1}(m - X_A S_1) \bmod (q - 1)$	substituting for S_2

COMP90043 Cryptography and Security
Semester 2, 2022, Workshop Week 12 Solutions

Authentication

1. What are the steps involved in an authentication process?
   Two steps are involved in an authentication process.
   Identification step: Presenting an identifier to the security system.
   Verification step: Presenting or generating authentication information that corroborates the binding between the entity and the identifier.

2. List three general approaches to deal with replay attacks.

   (a) Attach a sequence number to each message used in an authentication exchange. A new message is accepted only if its sequence number is in the proper order.

   (b) Party A accepts a message as fresh only if the message contains a timestamp that, in A's judgement, is close enough to A's knowledge of current time. This approach requires that clocks among the various participants be synchronized.

   (c) Party A, expecting a fresh message from B, first sends B a nonce (challenge) and requires that the subsequent message (response) received from B contain the correct nonce value.

3. What is a suppress-replay attack?
   Give an example of attack when a party's clock is ahead of that of the KDC.
   Give an example of attack when a party's clock is ahead of that of another party.

   When a sender's clock is ahead of the intended recipient's clock, an opponent can intercept a message from the sender and replay it later when the timestamp in the message becomes current at the recipient's site. This replay could cause unexpected results.

   An unintentionally post-dated message (message with a clock time that is in the future with respect to the recipient's clock) that requests a key is sent by a client. An adversary blocks this request message from reaching the KDC. The client gets no response and thinks that an omission or performance failure has occurred. Later, when the client is off-line, the adversary replays the suppressed message from the same workstation (with the same network address) and establishes a secure connection in the client's name.

   An unintentionally post-dated message that requests a stock purchase could be suppressed and replayed later, resulting in a stock purchase when the stock price had already changed significantly.

4. Consider Mutual Authentication proposed by Woo and Lam.

   (a) $A \rightarrow KDC : ID_A||ID_B$

(b) $KDC \rightarrow A : E(PR_{auth}, [ID_B||PU_b])$

(c) $A \rightarrow B : E(PU_b, [N_a||ID_A])$

(d) $B \rightarrow KDC : ID_A||ID_B||E(PU_{auth}, N_a)$

(e) $KDC \rightarrow B : E(PR_{auth}, [ID_A||PU_a])||E(PU_b, E(PR_{auth}, [N_a||K_s||ID_A||ID_B]))$

(f) $B \rightarrow A : E(PU_a, [N_b||E(PR_{auth}, [N_a||K_s||ID_A||ID_B])])$

(g) $A \rightarrow B : E(K_s, N_b)$

The protocol can be reduced from 7 steps to 5. Show the message transmitted at each step. Hint: the final message in this protocol is the same as the final message in the original protocol.

(a) $A \rightarrow B : ID_A||N_a$

(b) $B \rightarrow KDC : ID_A||ID_B||N_a||N_b$

(c) $KDC \rightarrow B : E(PR_{auth}, [ID_A||PU_a])||E(PU_b, E(PR_{auth}, [N_a||N_b||K_s||ID_A||ID_B]))$

(d) $B \rightarrow A : E(PU_a, E(PR_{auth}, [N_a||N_b||K_s||ID_A||ID_B]))$

(e) $A \rightarrow B : E(K_s, N_b)$

5. List three typical ways to use nonce as challenge.
Suppose N_a is a nonce generated by A, A and B share key K, and $f()$ is a function (such as an increment).

(a) $A \rightarrow B : N_a$
$B \rightarrow A : E(K, N_a)$

(b) $A \rightarrow B : E(K, N_a)$
$B \rightarrow A : N_a$

(c) $A \rightarrow B : E(K, N_a)$
$B \rightarrow A : E(K, f(N_a))$

6. From a web API security perspective what are the drawbacks using HMAC? Present an alternative to HMAC and discuss its benefit in web API authentication.
HMAC requires shared key for verification, use of HMAC by API gateways or edge gateways which perform the verification results in the need to store the MAC key (at least in memory).
Alternate approach given the structure of JWT is to use digital signatures, whereby the signing authority manages the private key but the gateways perform token verification using public keys. But the use of digital signature introduces the computational cost in comparison to HMAC.