CS 463/563: Cryptography for Cybersecurity
Spring 2025
Homework #10
Points: 20

Question 1: [Points 7] RSA Signature Scheme (*page 265 in the textbook*): Given the following table describing the procedure for Alice to send a signed message with RSA signature to Bob, calculate the unknown entities and verify that Bob has received the correct message sent by Alice.

Alice	Bob
Chooses **p = 23, q = 43**	
Compute **n = p*q =**	
Compute **φ(n)** =	
Choose **e = 17**	
Compute $d = e^{-1} \bmod \varphi(n) =$	
Compute Public key (**e, n**) = Private key (**d, n**) =	
Send Public key (**e, n**) to Bob:	Receives Alice's public key (**e, n**):
Message to send is **m = 9**	
Computes signatures **s** for m: $m^d \bmod n =$	
Send (**m, s**) to Bob:	Receives (**m, s**):
	Compute **m'**: $s^e \bmod n =$
	Verifies if **m = m'**

Question 2: [Points 7] **Elgamal Signature Scheme** (*page 270-272*): Given the following table describing the procedure for Alice to send a signed message with Elgamal signature to Bob, calculate the unknown entities and verify that Bob has received the correct message sent by Alice.

Alice	Bob
Chooses **p = 17**	
Chooses a primitive element **α = 11**	
Choose a random integer **d = 7**	
Compute $\beta = \alpha^d \bmod p =$	
Public key is $k_{pub} = (p, \alpha, \beta) =$ Private key is $k_{pr} = d =$	
Send Public key $k_{pub} = (p, \alpha, \beta)$ to Bob:	Receives Alice's public key $k_{pub} = (p, \alpha, \beta) =$
Choose an ephemeral key **KE = 5**	
Message to send is **m = 9**	
Computes signatures **(s, r)** for **m** $r = \alpha^{KE} \bmod p =$ Compute $KE^{-1} \bmod (p - 1) =$ $s = (m - d*r) * KE^{-1} \bmod (p - 1) =$	
Send **(m, (r, s))** to Bob:	Receives **(m, (r, s))** =
	Compute $t = \beta^r * r^s \bmod p =$
	Verifies if $t = \alpha^m \bmod p =$

Question 3: [Points 6] Compute **CBC-MAC** (*pages 325-526 in textbook*) for a message of 24 bits, **"A1A2A3"** (in Hexa).

Assume a block size of 8 bits with an **IV = D3 (hexa)** and **key** = **E4 (hexa)**.

Assume the encryption (and decryption) to be as follows:

If plaintext is LT||RT
Key is LK||RK, where LC, RC, LT, and RT are each 4 bits, then
Ciphertext = LC||RC
LC = LK XOR RT
RC = RK XOR LT
Plaintext and ciphertext are each 8 bits. Similarly, to decrypt ciphertext, we perform exactly the reverse operation
LT = RC XOR RK
RT = LC XOR LK.

Hint: *Divide the message into blocks of 8 bits each; XOR each block with the previous cipher output; then encrypt this with the key. For the first block, XOR it with IV. Details in pages 325-326 Ch-12 of the textbook.*

What to submit? Submit a pdf file with your answers via Canvas. Show your work