EE 418 - Assignment 3

Total Points: 100
Autumn Quarter, 2021

Prof. Radha Poovendran
Department of Electrical and Computer Engineering
University of Washington, Seattle, WA 98195

Due: 11:59 pm (PST) on Nov 18 (Thursday), 2021 via Canvas

Notes:

- This homework contains both computation questions (marked as [**Com**]) which are required to do by hand calculations and programming questions (marked as [**Pro**]) which are required to write Python /MATLAB codes. Zero points will be awarded if [**Com**] questions are solved via Python/MATLAB scripts and if [**Pro**] questions are solved by hand calculations.
- **Your answers to this homework must be submitted through canvas as a single zip file containing the following:** *i*) **hand written and scanned or word or pdf answers to all the computational and discussion questions as single pdf file.** *ii*) **Python/MATLB codes for programming questions as in filename.py or filename.m respectively.**
- **Name of your submission zip file should follow the following format. "#_$_*EE418_HW3.zip*", where "#" and "$" should be replaced with your first name and last name, respectively.**
- Show the computation steps and/or justify your answers in all the [**Com**] questions. Failure to show computation steps in [**Com**] questions will result zero points.
- You can use and modify the Python functions provided in the file section of the EE 418 canvas page when answering the [**Pro**] questions below.
- You can discuss with others but you need to write your own computation steps for [**Com**] questions and Python/MATLAB codes for [**Pro**] questions.

1. **[Com]**(Public Key Cryptosystem, 10 pts) Suppose that $m > 2$ users want to communicate securely and confidentially. Suppose further that each of the m users wants to be able to communicate with every other user without the remaining $m - 2$ users being able to listen on their conversation. How many distinct keys are needed if we are using:

   - A **symmetric key cryptosystem**, where two users use a shared secret key to communicate,
   - A **public key cryptosystem**, where every user i has a public key, PK_i and a private (secret) key, SK_i.

   How many keys are needed in each of the above cryptosystems if $m = 1000$?

2. **[Pro]**(RSA Decryption, 5pts $\times$ 3 = 15 pts) A sample of RSA ciphertext presented in Table 1 is generated using the following steps.

   (I) First alphabetic characters are "encoded" as the elements in $\mathbb{Z}_n$, where each element of $\mathbb{Z}_n$ represents three alphabetic characters as in the following examples:

   $$\begin{aligned} DOG &\rightarrow 3 \times 26^2 + 14 \times 26 + 6 &= 2398 \\ CAT &\rightarrow 2 \times 26^2 + 0 \times 26 + 19 &= 1371 \\ ZZZ &\rightarrow 25 \times 26^2 + 25 \times 26 + 25 &= 17575 \end{aligned}$$

   i.e., Each three letter plaintext block (m_i for $i = 1, 2, \ldots$) is "encoded" as in the above to get corresponding encoded-text block (e_i for $i = 1, 2, \ldots$).

   (II) Then each encoded-text block e_i is encrypted using RSA public key b to get ciphertext, $c_i = (e_i)^b \mod n$.

   Follow the steps given below to decrypt the ciphertext blocks c_i given in Table 1 assuming RSA Cryptosystem is using modulo base $n = 31313$ and public key $b = 4913$.

   (a) Write a Python/MATLAB code to factor the n and compute the RSA private key a from $\phi(n)$. (Hint: Since n is small you can use brute-force approach to factor n here. In such an approach you will need to check which prime number p in the range of $[2, \text{floor}(\sqrt{n})]$ will divide n)

   (b) Write a Python/MATLAB code to implement SQUARE-AND-MULTIPLY ALGORITHM in Algorithm 1. This algorithm implements exponentiation in modulo n in a computationally efficient way. It assumes that the exponent a is represented in binary notation, say $a = \sum_{i=0}^{l-1} a_i 2^i$, where $a_i = 0$ or 1, $0 \leq i \leq l - 1$ when computing $e = c^a \mod n$.

   (c) Write a Python/MATLAB code to decode any given e_i, encoded message blocks using the technique described in Step (I). and the use the Python/MATLAB functions you produced in part (a), part (b) and part(c) to decrypt the ciphertext given in Table 1.

Algorithm 1 Computationally efficient exponentiation in modulo n

function SQUARE-AND-MULTIPLY(c, a, n)
    e $\leftarrow 1$
    for $i \leftarrow (l - 1)$ downto 0 **do**
        $e \leftarrow e^2 \mod n$
        if $a_i = 1$ **then**
            $e = (e \times c) \mod n$
        end if
    end for
    return e
end function

6340	8309	14010	8936	27358	25023	16481	25809
23614	7135	24996	30590	27570	26486	30388	9395
27584	14999	4517	12146	29421	26439	1606	17881
25774	7647	23901	7372	25774	18436	12056	13547
7908	8635	2149	1908	22076	7372	8686	1304
4082	11803	5314	107	7359	22470	7372	22827
15698	30317	4685	14696	30388	8671	29956	15705
1417	26905	25809	28347	26277	7897	20240	21519
12437	1108	27106	18743	24144	10685	25234	30155
23005	8267	9917	7994	9694	2149	10042	27705
15930	29748	8635	23645	11738	24591	20240	27212
27486	9741	2149	29329	2149	5501	14015	30155
18154	22319	27705	20321	23254	13624	3249	5443
2149	16975	16087	14600	27705	19386	7325	26277
19554	23614	7553	4734	8091	23973	14015	107
3183	17347	25234	4595	21498	6360	19837	8463
6000	31280	29413	2066	369	23204	8425	7792
25973	4477	30989					

Table 1: RSA cipher text for Question 2

3. **[Com]** (Chinese Remainder Theorem, 5 pts $\times$ 2 = 10 pts) Solve the following system of congruences.

   (a)

$$\begin{aligned} x &\equiv 12 \mod 25 \\ x &\equiv 9 \mod 26 \\ x &\equiv 23 \mod 27 \end{aligned}$$

   (b)

$$\begin{aligned} 13x &\equiv 4 \mod 99 \\ 15x &\equiv 56 \mod 101 \end{aligned}$$

   (HINT: For part (b) use the Extended Euclidean Algorithm and then apply the Chinese remainder theorem)

4. (RSA protocol failure, 5 pts $\times$ 2 = 10 pts) This exercise exhibits what is called a *protocol failure*. It provides an example where ciphertext can be decrypted by an opponent, without determining the key, if a cryptosystem is used in a careless way. (Since the opponent does not determine the key, it is not accurate to call it cryptanalysis.) The moral is that it is not sufficient to use a "secure" cryptosystem in order to guarantee "secure" communication.

   Suppose "B" has an RSA Cryptosystem with a large modulus n for which the factorization cannot be found in a reasonable amount of time. Suppose "A" sends a message to Bob by representing each alphabetic character as an integer between 0 and 25 (i.e., $A \leftrightarrow 0$, $B \leftrightarrow 1$, etc.), and then encrypting each residue modulo 26 as a separate plaintext character.

   (a) Describe how an eavesdropper "E" can easily decrypt a message which is encrypted in this way.

   (b) **[Pro]** Write a Python/MATLAB code to illustrate this attack by decrypting the following ciphertext (which was encrypted using an RSA Cryptosystem with $n = 18721$ and public key $b = 25$) without factoring the modulus:

$$\mathbf{365,\ 0,\ 4845,\ 14930,\ 2608,\ 2608, 0}$$

Note: For the Questions 5, 6, 7, and 8 use the following fact. If $x^2 \equiv y^2 \mod n$ and $x \not\equiv \pm y \mod n$, then $\gcd(x-y, n)$ is a nontrivial factor of n.

5. **[Com]** (Factorization, 10 pts) Let $n = 642401$. Suppose you discover that,

$$516107^2 \equiv 7 \mod n$$

and that

$$187722^2 \equiv 2^2 \cdot 7 \mod n$$

Use this information to factor n.

6. (Factorization, 5 pts $\times$ 2 = 10 pts) Suppose you discover that

$$880525^2 \equiv 2 \mod 2288233, \quad 2057202^2 \equiv 3 \mod 2288233, \quad 648581^2 \equiv 6 \mod 2288233$$

$$668676^2 \equiv 77 \mod 2288233$$

   (a) How would you use this information to factor 2288233 ? Clearly, explain what are the steps you would do, but do not perform the hand calculations in this part.

   (b) **[Pro]** Write a Python/MATLAB script to find the factors of 2288233 using the steps you provided for part (a).

7. **[Pro]** (Factorization, 5 pts $\times$ 2 = 10 pts) Write Python/MATLAB scripts to perform the following tasks.

   (a) Let $n = 537069139875071$. Suppose you know that

$$85975324443166^2 \equiv 462436106261^2 \mod n$$

   Use this information to factor n.

   (b) Let $n = 985739879 \times 1388749507$. Note that numbers 985739879 and 1388749507 are prime numbers. Can you Find x and y with $x^2 \equiv y^2 \mod n$ but $x \not\equiv \pm y \mod n$.

   (Hint: Note such x and y will satisfiy one or both of the following properties:

   - $\gcd(x-y, n)$ is a nontrivial factor of n
   - $\gcd(x+y, n)$ is a nonttivial factor of n

   You may have to try different x and y values.)

8. **[Com]** (ElGamal Public Key Cryptosystems, 10 pts) Consider the ElGamal encryption scheme. "A" chooses a prime p and a primitive element α of $\mathbb{Z}_p$. "A" also chooses a private key a and computes $\beta = \alpha^a$. Then public key is $PK = (p, \alpha, \beta)$ and the private key is $SK = a$.

   If "B" wants to send a message m to "A", "B" uses public key PK and encrypts a message m as follows. Choose a secret random number $k, 1 \leq k \leq p-2$. The encryption of m is $E_{PK}(m, k) = c = (y_1, y_2)$ where $y_1 = \alpha^k \mod p$ and $y_2 = m\beta^k \mod p$. "A" performs the decryption of $c = (y_1, y_2)$ as $D_{SK}(c) = y_2(y_1^a)^{-1} = m \mod p$.

   "B" chooses two messages m_1 and m_2 and secret random numbers k_1 and k_2." B" encrypts m_1 using k_1 and obtains $E_{PK}(m_1, k_1) = (y_1, y_2)$ and encrypts m_2 using k_2 and obtains $E_{PK}(m_2, k_2) = (y_3, y_4)$. "B" then transmits $c = (y_1 y_3 \bmod p,\ y_2 y_4 \bmod p)$ to "A". What is the plaintext that "A" obtains after decrypting c? Show your steps.

   Awards under this announcement will be made only to U.S. institutions of higher education which award degrees in science, engineering or mathematics. U.S. non-profit organizations operating primarily for scientific and educational services may also submit proposals. The principal investigator of a proposal must be a U.S. citizen, national or permanent resident (on the date proposals are due), holding a first or second full-time tenure-track or tenure-track-equivalent faculty position at that university, and has received his/her doctorate or equivalent degree within the past seven years. See solicitation for eligibility dates. The term "national" of the United States includes a native resident of a possession of the United States, such as American Samoa.

9. **[Pro]**(ElGamal Decryption, 15 pts) Write a Python/MATLAB code to decrypt the ElGamal ciphertext presented in Table 2 assuming groups of three alphabetic characters are encoded using the technique described in Question 2, Step (I) before encrypting using the ElGamal public key (α, β, p). The parameters of ElGamal public key cryptosystem is given as $\alpha = 5$, $\beta = 18074$, $p = 31847$, and private key $a = 7899$.

(3781, 14409)	(31552, 3930)	(27214, 15442)	(5809, 30274)
(54000, 31486)	(19936, 721)	(27765, 29284)	(29820, 7710)
(31590, 26470)	(3781, 14409)	(15898, 30844)	(19048, 12914)
(16160, 3129)	(301, 17252)	(24689, 7776)	(28856, 15720)
(30555, 24611)	(20501, 2922)	(13659, 5015)	(5740, 31233)
(1616, 14170)	(4294, 2307)	(2320, 29174)	(3036, 20132)
(14130, 22010)	(25910, 19663)	(19557, 10145)	(18899, 27609)
(26004, 25056)	(5400, 31486)	(9526, 3019)	(12962, 15189)
(29538, 5408)	(3149, 7400)	(9396, 3058)	(27149, 20535)
(1777, 8737)	(26117, 14251)	(7129, 18195)	(25302, 10248)
(23258, 3468)	(26052, 20545)	(21958, 5713)	(346, 31194)
(8836, 25898)	(8794, 17358)	(1777, 8737)	(25038, 12483)
(10422, 5552)	(1777, 8737)	(3780, 16360)	(11685, 133)
(25115, 10840)	(14130, 22010)	(16081, 16414)	(28580, 20845)
(23418, 22058)	(24139, 9580)	(173, 17075)	(2016, 18131)
(198886, 22344)	(21600, 25505)	(27119, 19921)	(23312, 16906)
(21563, 7891)	(28250, 21321)	(28327, 19237)	(15313, 28649)
(24271, 8480)	(26592, 25457)	(9660, 7939)	(10267, 20623)
(30499, 14423)	(5839, 24179)	(12846, 6598)	(9284, 27858)
(24875, 17641)	(1777, 8737)	(18825, 19671)	(31306, 11929)
(3576, 4630)	(26664, 27572)	(27011, 29164)	(22763, 8992)
(3149, 7400)	(8951, 29435)	(2059, 3977)	(16258, 30341)
(21541, 19004)	(5865, 29526)	(10536, 6941)	(1777, 8737)
(17561, 11884)	(2209, 6107)	(10422, 5552)	(19371, 21005)
(26521, 5803)	(14884, 14280)	(4328, 8635)	(28250, 21321)
(28327, 19237)	(15313, 28649)		

Table 2: ElGamal cipher text for Question 9