Khoa Khoa học & Kỹ thuật máy tính
Trường ĐH Bách Khoa TP.HCM

Cryptography and Network Security
Lab 3
RSA Algorithm

INTRODUCTION

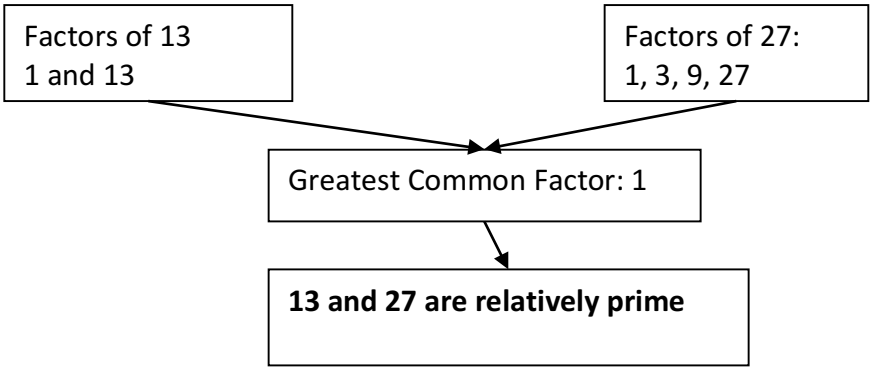
In 1978, Ron Rivest, Adi Shamir, and Leonard Adleman introduced a cryptographic algorithm. RSA is motivated by the published works of Diffie and Hellman from several years before, who described the idea of such an algorithm, but never truly developed it. Introduced at the time when the era of electronic email was expected to soon arise, RSA implemented two important ideas: Public-key encryption and Digital signatures.

EXPERIENCE

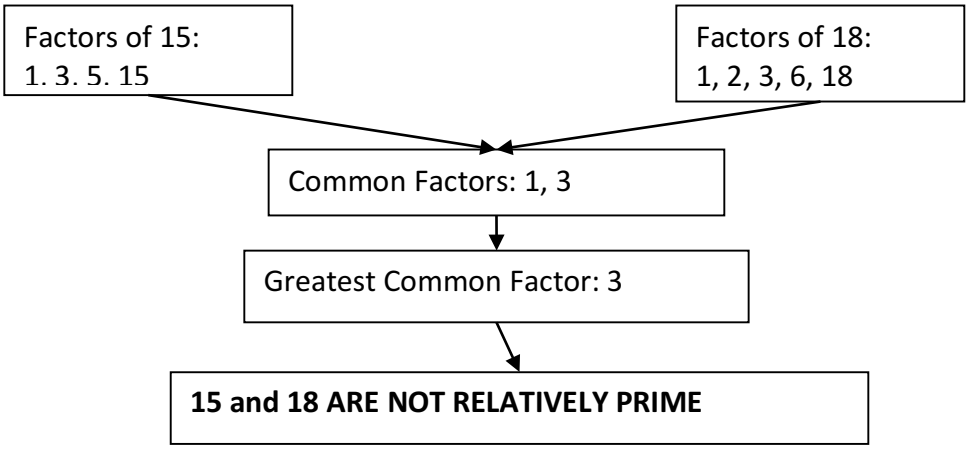
1. Relatively Prime Numbers:

a and b are relatively prime
if the greatest common factor of a and b is 1

Example 1: 13 and 27 are relatively prime:



Example 2: 15 and 18 are NOT relative prime



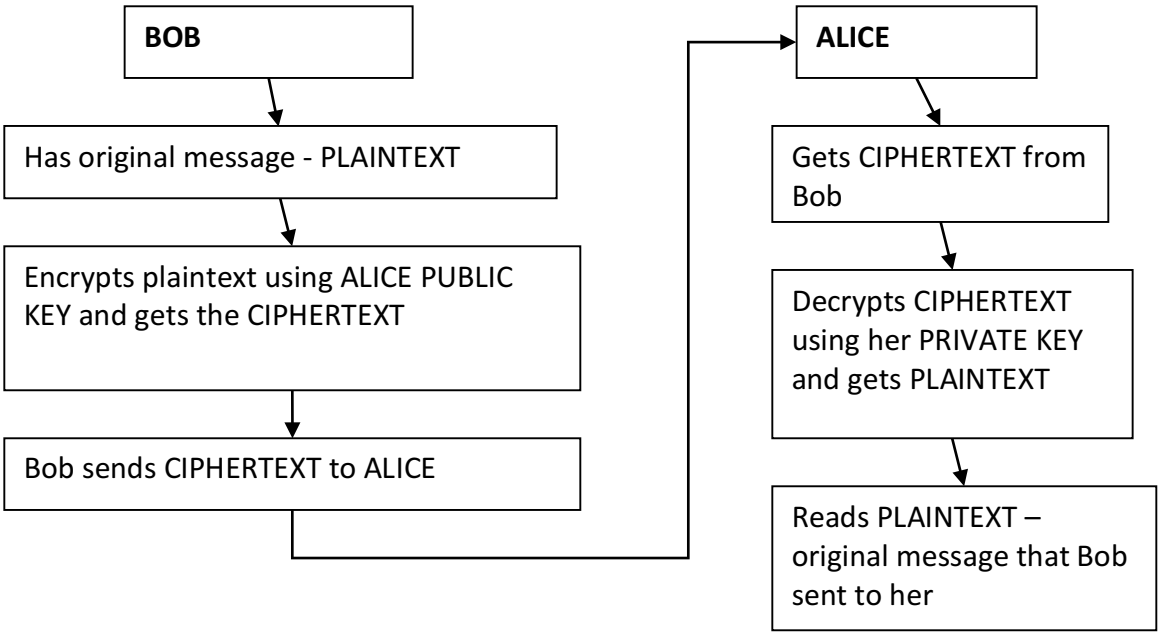
2. RSA

Alice and Bob would like to communicate secure. They decided to use the public key cryptology algorithm RSA.
In our examples:

- Bob would be able to ENCRYPT the original message (PLAINTEXT) and to SEND ENCRYPTED MESSAGE (CIPHERTEXT) to Alice.
- BOB will use ALICE PUBLIC KEY to encrypt

- Alice would be able to DECRYPT the CIPHERTEXT that she got from Bob and to read an original message (PLAINTEXT).
- ALICE will use her PRIVATE KEY to decrypt

In general, both, Alice and Bob are able to encrypt and decrypt. We will learn this later.



Example 1:

In order for Bob to encrypt and to send encrypted messages (ciphertexts) to Alice, he needs ALICE PUBLIC KEY.
This is the reason that our example will start from Alice, since without her Public Key communication would not be possible.

ALICE will follow the steps below to create her public key and her private key:

Step 1: Alice chooses two prime numbers p and q.
In our example, Alice chooses p = 3 and q = 5

Step 2: Alice computes m = p*q = 3* 5 = 15

Step 3: Alice computes n = (p-1)*(q-1) = 2 * 4 = 8

Step 4: Alice chooses the number e which is relatively prime with n.
In this example, Alice chooses e = 7 (note, that 7 and 8 are relatively prime)

Step 5: Alice will calculate d – multiplicative inverse of e mod n, means d satisfies the following condition: **(d*e) MOD n = 1**

In our example d = 7:
Note, in our case:
(d*e) MOD n = (7 * 7) MOD 8 = 49 MOD 8 = 1

IN GENERAL:

PUBLICK KEY is m and e
PRIVATE KEY is p, q and d

In our example:
PUBLIC KEY is m = 15 and e = 7
PRIVATE KEY is p = 3, q = 5 and d = 7
ALICE PUBLISHES HER PUBLIC KEY (m = 15 and e = 7) on her website and now BOB will be able to send her encrypted messages.

BOB:
Would like to send the following plaintext message:
plaintext = 3

(Note: in this communication the original message is the NUMBER. We will learn how to convert the TEXT into the NUMBER. But at that moment we will just assume that we have already the number that corresponds to our text).
Also, there is a limitation on the size of plaintext the Bob can encrypt.
The plaintext MUST satisfy the following condition: **plaintext < m**
(in our case plaintext = 3 and m = 15 and 3 < 15)

TO ENCRYPT, Bob would use the following ENCRYPTION formula:

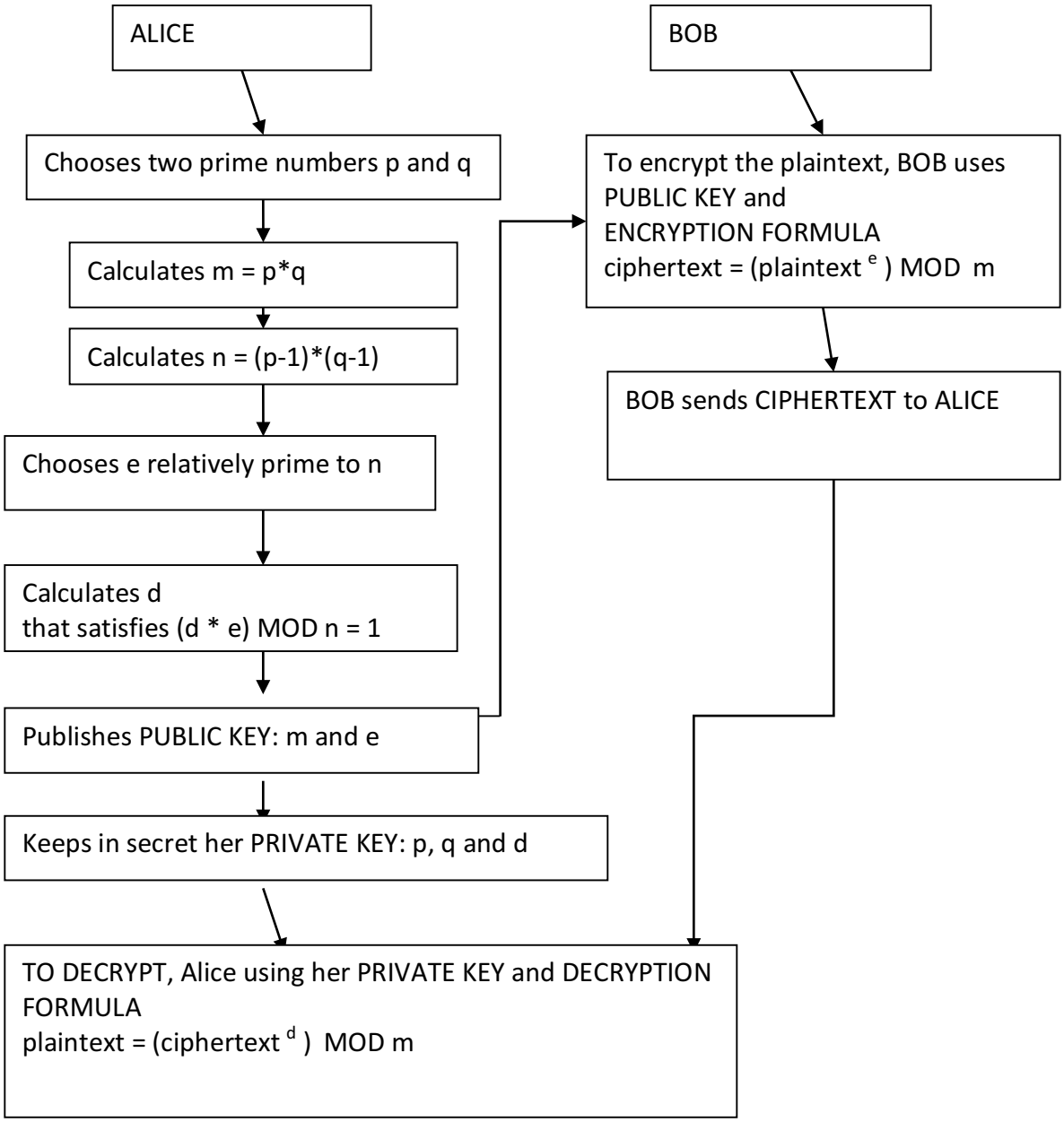
ciphertext = (plaintext^{e}) MOD m

In our example (check my calculations!):
ciphertext = (3^7)MOD 15 = 12
BOB will send 12 to ALICE.
ALICE:
Gets ciphertext = 12 from Bob, and she needs to DECRYPT.
Alice will use the following DECRYPTION formula:

plaintext = (ciphertext ^{d}) MOD m

In our example (check my calculations!):
plaintext = (12 ^{7})MOD 15 = 3
and Alice recovered the original message, plaintext = 3, that Bob sent.

SUMMARY



Example 2 (read and understand all steps):

Alice:
Step 1: p = 5 and q = 11
Step 2: m = p*q = 5*11 = 55
Step 3: n = (p-1)*(q-1) = 4*10 = 40
Step 4: e = 3 (40 and 3 are relatively prime)
Step 5: d = 27 (verify, that (27 * 3) MOD 40 = 1)

Alice Public Key: m = 55 and e = 3
Alice Private Key: p = 5, q = 11 and d = 27

Encryption:
Bob would like to send a message: plaintext = 12
ciphertext = 12^3 MOD 55 = 23
Bob sends 23 to Alice

Decryption:
Alice gets 23. To decrypt and recover the original message she does the following:
plaintext = 23^{27} MOD 55 = 12

Alice recovered the original message which is 12

Example 3 (in this example you would need to complete some calculations)

ALICE:
Step 1: p = 7 and q = 13
Step 2: calculate and write below the value of m.
m =
Step 3: calculate and write below the value of n.
n =
Step 4: e = 5
Step 5: Use my program to calculate the value of d (call me to show you how to use the program). Write the value of d below.
d =

Write below ALICE PUBLIC KEY and PRIVATE KEY
PUBLIC KEY: _____

PRIVATE KEY: _____

Suppose BOB would like to send a message: plaintext = 18
Find a ciphertext and write it below.
cipherterxt =

Perform the step that Alice would perform to decrypt the message. Write all calculations below.
plaintext =

Example 4: In real life both people create public and private keys and both people can send and receive the messages. In this example you will follow the steps that Alice and Bob will perform in order to establish the secure communication and also the steps that Alice and Bob will perform to encrypt and decrypt the messages.

1. Alice and Bob create their public and private keys respectively:

Alice (will follow her 5 steps as before):

1. p = 3 and q = 11
2. m = p*q = 3*11 = 33
3. n = (p-1)*(q-1)= 2*10 = 20
4. e = 7 (7 and 20 are relatively prime)
5. d = 3 (7*3 MOD 20 = 1)

Alice Public Key: m = 33 and e = 7
Alice Private Key: p = 3, q = 11 and d = 3

Bob (will follow the same 5 steps to create his public and private keys)

1. p = 5 and q = 13
2. m = p*q = 65
3. n = (p-1)*(q-1)= 4*12= 48
4. e = 11 (11 and 48 are relatively prime)

5. d = 35 (35*11 MOD 48 = 1)

Bob Public Key: m = 65 and e = 11
Bob Private Key: p = 5, q= 13 and d = 35

Alice would like to send message to Bob. She is doing the following:

1. Goes to the Bob website to get his public key: m = 65 and e = 11
2. Alice would like to send the plaintext = 17. Alice checks that 17 < 65
3. Alice encrypts the plaintext using Bob Public key: 17^{11} MOD 65 = 23
4. Alice sends 23 to Bob. 23 is a ciphertext that Bob will receive.

Bob receives ciphertext 23 from Alice. To decrypt, Bob will use his PRIVATE KEY and decryption formula:
23^{35} MOD 65 = 17 (Bob is using his private key d = 35)
Bob recovers the plaintext which is 17.
17 was the original message that Alice sent to Bob.
Now, Bob would like to send Alice a message. He is doing the following:
1. Bob goes to Alice website to get Alice Public Key: m = 33 and e =7
2. Bob would like to send a plaintext = 14 (14 < 33)
3. Bob encrypts the plaintext using Alice Public Key: 14^7 MOD 33 = 20
4. Bob sends 20 to Alice. 20 is a ciphertext.

Alice gets ciphertext = 20 and she needs to decrypt the ciphertext using her Private Key and decryption formula:
20^3 MOD 33 = 14 (Alice is using her private key d = 3)
Alice recovers the plaintext which is 14.
14 was the original message that Bob sent to Alice.

In this example you observe two – way secure communication using RSA algorithm.

- Alice is using Bob Public Key to send messages to Bob.
- Bob is using Alice Public Key to send messages to Alice.
- To decrypt, each of them is using their Private Keys.

HOMEWORK

Exercise 1. What are the roles of the public and private key?

Exercise 2. What is a one-way function?

Exercise 3. What is a trap-door one-way function?

Exercise 4. Perform encryption and decryption using the RSA algorithm, for the following:

a. p=3;q=11,e=7;M=5

b. p=5;q=11,e=3;M=9

c. p=7;q=11,e=17;M=8

d. p=11;q=13,e=11;M=7

e. p=17;q=31,e=7;M=2

Exercise 5. In a public-key system using RSA, you intercept the ciphertext C = 10 sent to a user whose public key is e = 5, n = 35.

What is the plaintext M?

Exercise 6. In an RSA system, the public key of a given user is e = 31, n = 3599.What is the private key of this user?

Hint: First use trial-and-error to determine p and q; then use the Extended Euclidean algorithm to find the multiplicative inverse of 31 modulo $\phi(n)$.

Extended Euclidean algorithm

```
Procedure Euclid_Extended (a,m)
int,  y0=0,y1:=1;
While a>0 do {
     r:= m mod a
     if r=0 then Break
     q:= m div a
     y:= y0-y1*q
     m:=a
     a:=r
     y0:=y1
     y1:=y
  }
 If a>1 Then Return null
 else Return y
```