- Konstantinos Filippou
- ics23044

Exercise 1

Έστω ότι **p = 41** και **q = 17** αποτελούν τους πρώτους αριθμούς που επιλέγονται στη φάση δημιουργίας κλειδιών στου RSA.

- Ποιος από τους δύο υποψήφιους ακεραίους **e1 = 5** και **e2 = 19** αποτελούν έγκυρο δημόσιο εκθέτη για τον RSA? Αιτιολογήστε την απάντησή σας.
- Υπολογίστε το ιδιωτικό κλειδί **kpr = d** εφαρμόζοντας τον Διευρυμένο Ευκλείδιο Αλγόριθμο στο χαρτί.

Solution

- Για να βρούμε αν αποτελούν έγκυρο δημόσιο εκθέτη αρκεί να δούμε αν είναι πρώτοι ως προς φ(n):

  n = pxq = 41x17 = 697

  φ(n) = (p-1)x(q-1) = 40x16 = 640

  Για e1 = 5 ελέγχουμε gcd(5,640):

  640 = 5x128 + 0. Άρα gcd(5,640) δεν είναι 1, επομένως το e1 = 5 δεν έχει αντίστροφο και δεν μπορεί να χρησιμοποιηθεί ως δημόσιος εκθέτης για τον RSA.

  Για e2 = 19 ελέγχουμε gcd(19,640):

  640 = 19x33 + 13, 19 = 13x1 + 6, 13 = 6x2 + 1, 6 = 1x6 + 0.Άρα gcd(15,640) είναι 1, επομένως το e2 = 19 έχει αντίστροφο και μπορεί να χρησιμοποιηθεί ως δημόσιος εκθέτης για τον RSA.
- Εφόσον e2 = 19 είναι έγκυρος δημόσιος εκθέτης, βρίσκουμε d τέτοιο ώστε de = 1 mod (φ(n))

  Άρα για να βρούμε το d, βρίσκουμε τον αντίστροφο του 19 mod 640 (e mod φ(n))

  Από 640 = 19x33 + 13 προκύπτει 13 = 640 - 19x33, από 19 = 13x1 + 6 προκύπτει 6 = 19 - 13x1, Αντικαθιστώντας την έκφραση για 13 (από 1):

  6 = 19 − (640−19x33) = 19 − 640 + 19x33 = 19x34 − 640. Άρα 6 = -1x640 + 34x19.

  Από 13 = 6x2 + 1 προκύπτει 1 = 13 − 6x2.Αντικαθιστούμε τις εκφράσεις για 13 και 6:

  1 = (640 − 19x33) − 2x(−1x640 + 34x19).

  Αναπτύσσουμε:

  1 = 640 − 19x33 −2(−640) +−2(34x19)

  1 = 640 − 33x19 + 2x640 − 68x19

  1 = (1+2)x640 + (−33−68)x19

  1 = 3x640 + (−101)x19.

  Επομένως προκύπτει: 1 = 3x640 + (-101)x19

  Ο συντελεστής του 19 είναι -101 επομένως ο αντίστροφος του 19 mod 640 είναι 640-101 = 539

  Επομένως d = 539.

Exercise 2

Ο αποδοτικός υπολογισμός της υπολοιπικής εκθετοποίησης είναι κεφαλαιώδους σημασίας για την πρακτική εφαρμογή του RSA. Υπολογίστε τις ακόλουθες υπολοιπικές εκθετοποιήσεις **m^e mod n** εφαρμόζοντας στο χαρτί τον αλγόριθμο "τετραγώνισε και πολλαπλασίασε":

- **m = 2, e = 31, n = 101**
- **m = 3, e = 97, n = 101**

Solution

- 2 ^ 31 mod 101 =

  31 = 11111 in binary

i	**4**	**3**	**2**	**1**	**0**
bi	1	1	1	1	1
	2 mod 101	2^2 * 2 mod 101	8^2 *2 mod 101	27^2 *2 mod 101	44^2 * 2 mod 101
	=2	=8	=27	=44	=34

Άρα 2^31 mod 101 = 34

- 3^97 mod 101 =

  97 = 1100001 in binary

i	**6**	**5**	**4**	**3**	**2**	**1**	**0**
bi	1	1	0	0	0	0	1
	3 mod 101	3^2 * 3 mod 101	27^2 mod 101	22^2 mod 101	80^2 mod 101	37^2 mod 101	56^2 * 3 mod 101
	3	=27	=27	=80	=37	=56	= 15

  Άρα 3^97 mod 101 = 15

Exercise 3

Χρησιμοποιώντας τον κρυπταλγόριθμο RSA κρυπτογραφήστε και αποκρυπτογραφήστε έχοντας ως παραμέτρους:

- **p = 3, q = 11, d = 7, m = 4**
- **p = 5, q = 11, e = 3, m = 20**

Solution

- n = pxq = 3x11 = 33

  Πρέπει να βρούμε το e για την υλοποίηση του RSA. Από d = 7 βρίσκουμε e από $e*d \equiv 1 \pmod{\varphi(n)}$.

Εφαρμόζουμε Διευρυμένο Ευκλείδειο

20=7x2+6, 7=6·1+1

Επιστρέφοντας: 1 = 7−6 = 7−(20 − 7x2) = 7x3 − 20, οπότε 3*7 ≡ 1(mod20). Άρα e = 3.

Κρυπτογράφηση:

c = m^e (mod n) = 4^3 (mod 33) = 64 mod 33 = 31

Αποκρυπτογράφηση:

Η αποκρυπτογράφηση μας δίνεται και είναι 4

- n = pxq = 5x11 = 55

  φ(n) = (p-1)x(q-1) = 4x10 = 40

  Κρυπτογράφηση:

  c ≡ m^e (mod n) = 20^3 mod 55 = 8000 mod 55. 8000 / 55 = 7975, επομένως υπόλοιπο = 8000-7975 = 25.

  Αποκρυπτογράφηση:

  Η αποκρυπτογράφηση μας δίνεται και είναι 20

Exercise 4

Ένα σχήμα κρυπτογράφησης RSA έχει ως παραμέτρους **p = 31** , **q = 37** και δημόσιο εκθέτη **e = 17**

- Αποκρυπτογραφήστε το κρυπτοκείμενο **c = 2** με χρήση του Κινέζικου Θεωρήματος Υπολοίπων (CRT)
- Επαληθεύστε το αποτέλεσμα κρυπτογραφώντας το απλό κείμενο (χωρίς χρήση του CRT)

Solution

- cp = c mod p = 2 mod 31 = 2

cq = c mod q = 2 mod 37 = 2

n = pxq = 31x37 = 1147

φ(n) = (p-1)x(q-1) = 30x36 = 1080

Θέλουμε d τέτοιο ώστε e*d ≡ 1 (mod 1080)

1080 = 17x63 + 9

17 = 9x1 + 8

9 = 8x1 + 1

8 = 8x1 + 0

Άρα υπάρχει αντίστροφος αφού υπόλοιπο 1

1 = 9 - 8x1

Αντικαθιστούμε το 8 από την εξίσωση:

8 = 17 − 9x1

Άρα 1 = 9 − (17−9) = 9x2−17

Αντικαθιστούμε το 9 από την πρώτη εξίσωση:

Άρα 1 = (1080 − 17x63)x2 − 17

1 = 2x1080 − 17x126 − 17

1 = 2x1080 − 17x127

Άρα d = -127 , 1080-127 = 953.

dp = d mod (p-1) = 953 mod (30) = 23

dq = d mod (q-1) = 953 mod (36) = 17

mp = 2^23 mod 31 =

23 = 010111 in binary

i	**4**	**3**	**2**	**1**	**0**
bi	1	0	1	1	1
	2 mod 31	2^2 mod 31	4^2 *2 mod 31	1^2 * 2 mod 31	2^2 * 2 mod 31
	=2	=4	=1	=2	=8

Άρα mp = 8

mq = 2^17 mod 37 =

17 = 10001 in binary

i	**4**	**3**	**2**	**1**	**0**
bi	1	0	0	0	1
	2 mod 37	2^2 mod 37	4^2 mod 37	16^2 mod 37	34^2 * 2 mod 37
	=2	=4	=16	=34	=18

Άρα mq = 18

βρίσκουμε tp = q^-1 mod p = 37^-1 mod 31 = 37-31 = 6, βρίσκουμε το αντίστροφο του αριθμού 6 mod 31. 31 = 5x6 + 1, 6 = 1x6 + 0 (αντιστρέφεται)

λύνουμε ως προς 1, 1 = 31-5x6, επομένως το αντίστροφο είναι -5, άρα 31-5 = 26. Άρα tp=26.

Βρίσκουμε tq = p^-1 mod q = 31^-1 mod 37, 37 = 1x31 +6, 31 = 6x5 + 1, 6 = 6x1 + 0.

Λύνουμε ως προς 1, 1 = 31-6x5, όμως 6 = 37-1x31 , άρα 1=31-5x(37 - 1x31), 1 = 31 - 5x37 + 5x31, 1 = 6x31 - 5x37, 6x31 ≡ 1 (mod 37)

Άρα tq = 6

m = qtpmp + ptqmq mod n = (37x26x8 + 31x6x18) (mod 1147) = (7696 + 3348) (mod 1147) = 11044 mod 1147 = 721

- Για την επαλύθευση αρκεί να εφαρμόσουμε τον τύπο c = m^e (mod n) , c = 721^17 (mod 1147)

i	4	3	2	1	0
bi	1	0	0	0	1
	721 mod 1147	721^2 mod 1147	250^2 mod 1147	562^2 mod 1147	419^2 * 721 mod 1147
	= 721	= 250	= 562	= 419	= 2

Exercise 5

Υλοποιήστε τη Δραστηριότητα RSA με κατάλληλο ορισμό και κλήση των συναρτήσεων όπως αυτές περιγράφονται παραπάνω.

Solution

[74]:
```
# Key generation
def keygen(bits):
    a=next_prime(ZZ.random_element(2^(bits//2 +1)))
    b=next_prime(ZZ.random_element(2^(bits//2 +1)))
    n=a*b
    phi_n = (a-1)*(b-1)
    while True:
        e = ZZ.random_element(1,phi_n)
        if gcd(e,phi_n) == 1:
            break
    d = inverse_mod(e,phi_n)
    return n,e,d
```

[75]:
```
# Code and Decode
def str2num(s):
    x = map(ord,s)
    return ZZ(list(x), 128)
def num2str(n):
    dgs = n.digits(128)
    return ''.join(map(chr,dgs))
```

[76]:
```
# Encryption and Decryption
def rsa_enc(m,e,n):
    messageCoded = str2num(m)
    ciphertextCoded = lift(Mod(messageCoded,n)^e)
    return num2str(ciphertextCoded)
def rsa_dec(c,d,n):
    cipherCoded = str2num(c)
    decryptedMsgCoded = lift(Mod(cipherCoded,n)^d)
    return num2str(decryptedMsgCoded)
```

[77]:
```
# Call for key generation
n, e, d = keygen(1024)
```

[78]:
```
# Call for encryption
message='Meeting at dawn behind the school'
ciphertext=rsa_enc(message,e,n)
print (ciphertext)
```

```
Q$t□@)□"/aC□(□□e$CI'□y□GlMvOUD_m□*%□_]kL
|□>vwG□K\6:□O□],xWE3<□76□ga□P□4\        □v.U□g&Z!q`□3+F□^□{uJ&  U#w5T#Bi□'P□□?]-nK]□(□%ao□,□□
```

[79]:
```
# Call for decryption
decryptedMsg=rsa_dec(ciphertext,d,n)
print (decryptedMsg)
```

```
Meeting at dawn behind the school
```

Exercise 6

Μια από τις εφαρμογές ενός κρυπτοσυστήματος δημοσίου κλειδιού είναι η εγκαθίδρυση με ασφάλεια ενός κλειδιού συνόδου που αποτελεί το μυστικό κλειδί ενός συμμετρικού αλγορίθμου, όπως ο AES, πάνω από ένα μη ασφαλές κανάλι. Υποθέστε ότι ο Bob διαθέτει ένα ζεύγος ιδιωτικού και δημοσίου κλειδιού για το κρυπτοσύστημα RSA. Σχεδιάστε ένα απλό πρωτόκολλο το οποίο χρησιμοποιώντας τον RSA θα επιτρέψει στην Alice και τον Bob να συμφωνήσουν σε ένα κοινό μυστικό κλειδί. Ποιο μέλος της επικοινωνίας καθορίζει το μυστικό κλειδί, η Alice, ο Bob ή και οι δύο?

Solution

Η Alice παράγει ένα τυχαίο μυστικό κλειδί, έστω K το οποίο έχει μήκος μικρότερου του n λόγω RSA.

Η Alice κρυπτογραφεί το K με το δημόσιο κλειδί του Bob:

C = K^e mod n, και στέλνει το c στον Bob.

Ο Bob αποκρυπτογραφεί με το ιδιωτικό του κλειδί:
K = C^d mod n. Τώρα γνωρίζουν και οι δύο το K.

Στο παραπάνω πρωτόκολλο η Alice είναι αυτή που επιλέγει το K και ο Bob το δέχεται χωρίς να μπορεί να επηρεάσει την επιλογή. Αυτό σημαίνει ότι μόνο η Alice καθορίζει το μυστικό κλειδί.

Exercise 7

Ας υποθέσουμε ότι στο κρυπτοσύστημα RSA ο αντίπαλος ανακαλύπτει ένα μήνυμα **m** , το οποίο είναι ένας μη μηδενικός ακέραιος που δεν είναι σχετικά πρώτος με το modulus **n = p * q** .

- Αποδείξτε ότι ο αντίπαλος μπορεί να παραγοντοποιήσει το και, κατά συνέπεια, να παραβιάσει το σύστημα.
- Αν η επιλογή του **m** είναι τυχαία, ποια είναι η πιθανότητα να μην είναι το μήνυμα σχετικά πρώτος με το **n**;

Solution

- Έστω m γνωστό, gcd(m,n)=g, 1<g<n.
  Τότε g διαιρεί το n, και n = g * (n/g)
  Και τα δύο g,n/g, είναι ακέραιοι > 1, άρα n παραγοντοποιήθηκε.

  Έχοντας p,q, υπολογίζει φ(n)=(p−1)(q−1),

μπορεί να υπολογίσει το ιδιωτικό κλειδί d από το δημόσιο e (λύνοντας $e*d \equiv 1 \pmod{\varphi(n)}$),
άρα **παραβιάζει πλήρως το RSA** (μπορεί να αποκρυπτογραφεί οποιοδήποτε κρυπτογράφημα)

- Αν η επιλογή του m είναι τυχαία,

  Οι αριθμοί στο {1,2,...,n−1} που **δεν** είναι σχετικά πρώτοι με το n είναι αυτοί που διαιρούνται είτε με p είτε με q.

  Πλήθος πολλαπλάσιων του p: n-1/p = q-1

  Πλήθος πολλαπλάσιων του q: n-1/q = p-1

  Άρα συνολικά αριθμοί που έχουν κοινό διαιρέτη με n:

  (q-1) + (p-1) = p+q-2

  Πιθανότητα (p+q-2) / n-1 , που για μεγάλους πρώτους αριθμούς αυτό είναι εξαιρετικά μικρό.