ШИНЖЛЭХ УХААН ТЕХНОЛОГИЙН ИХ СУРГУУЛЬ

Мэдээлэл, холбооны технологийн сургууль

БИЕ ДААЛТЫН АЖЛЫН ТАЙЛАН

Өгөгдөл ба сүлжээний нууцлалт хамгаалалт (F.NS311) 2023-2024 оны хичээлийн жилийн хавар

Хичээл заасан багш: **Г.Баяр**

Бие даалтын ажил гүйцэтгэсэн: **Г.Цэрэнням(B210930847)**

Вариант: **14**

Вариант:14	2.34	3.14	7.2	Prommaming 7.1

Бодлого1.

2.34 The example used by Sun-Tsu to illustrate the CRT was x K 2 (mod 3); x K 3 (mod 5); x K 2 (mod 7)`

x=2(mod3) m1=3, a1=2

x=3(mod5) m2=5,a2=3

x=2(mod7) m3=7,a3=2

1). M=m1*m2*m3=3*5*7=105

2). M1=M/m1=105/3=35

M2=M/m2=105/5=21

M3=M/m3=105/7=15

3). $M1^{-1}$ =35^{-1}(mod3)=2

$M2^{-1}$ =21^{-1}(mod5)=1

$M3^{-1}$ =15^{-1}(mod7)=1

4). x=((a1*M1*$M1^{-1}$)+ (a2*M2*$M2^{-1}$)+ (a3*M3*$M3^{-1}$))modM

=((2*35*2)+(3*21*1)+(2*15*1)mod105

=233mod105=23

x=23

Онол : Хятадын үлдэгдэл теорем (CRT) нь конгрюцын системийг шийдвэрлэх хэрэгсэл юм. Энэ нь янз бүрийн модулиудаар (эерэг хуваагч) хуваахад үлдэгдлийг тодорхойлсон тэгшитгэлийн багцыг хангасан өвөрмөц бүхэл тоог олох боломжийг олгоно. Гол нөхцөл нь эдгээр модулиуд нь хос хосолсон байх ёстой бөгөөд энэ нь 1-ээс өөр нийтлэг хүчин зүйлийг хуваалцахгүй гэсэн үг юм.

CRT нь янз бүрийн салбарт олон тооны хэрэглээтэй бөгөөд үүнд:

Компьютерийн шинжлэх ухаан: Энэ нь том бүхэл тоонуудын тооцооллыг модулиар бага тоогоор тооцоолоход хялбар болгодог.

Криптограф: Теорем нь RSA гэх мэт нийтийн түлхүүрийн криптографийн системд үүрэг гүйцэтгэдэг.

Бодлого2.

3.14 a. Encrypt the message “meet me at the usual place at ten rather than eight o clock’’ using the with the key ¢7 3 2 5≤. Show your calculations and the result. b. Show the calculations for the corresponding decryption of the ciphertext to recover the original plaintext.

Hill cipher

m	e	e	t	m	e	a	t	t	h	e	u	s	u	a	L
12	4	4	19	12	4	0	19	19	7	4	20	18	20	0	12
p	l	a	c	e	a	t	t	e	n	r	a	t	h	e	R
15	12	0	2	4	0	19	19	4	13	17	0	19	7	4	17
t	h	a	n	e	i	g	h	t	o	c	l	o	c	k	
19	8	0	13	4	8	6	7	19	14	2	11	14	2	10	

$C = K \cdot P \bmod 26 \longrightarrow$ encryption

$P = K^{-1} \cdot C \bmod 26 \longrightarrow$ decryption

pt = meet me at the usual place rather than eight o clock

$$\text{Key } (K) = \begin{bmatrix} 7 & 3 \\ 2 & 5 \end{bmatrix}$$

$$K^{-1} = \frac{\text{Adj}(K)}{|K|}$$

$$= \frac{1}{29}\begin{bmatrix} 5 & -3 \\ -2 & 7 \end{bmatrix}$$

$P = K^{-1} \cdot C \bmod 26$

12 4	4 19	12 4	0 19	19 7	4 20	18 20	0 11	15 11	0 2	4 17
ME	ET	ME	AT	TH	EU	SU	AL	PL	AC	ER

AT	HE	RT	AA	NE	TH	HT	OC	LO	CK
0 19	7 4	17 19	7 0	13 4	8 6	7 19	14 2	11 14	2 10

$$P_1 = \begin{bmatrix} \frac{5}{29} & \frac{-3}{29} \\ \frac{-2}{29} & \frac{7}{29} \end{bmatrix} \times \begin{vmatrix} 12 \\ 4 \end{vmatrix} \bmod 26$$

$$= \begin{vmatrix} 1.65 \\ 0.13 \end{vmatrix} \bmod 26 \Rightarrow \begin{vmatrix} 2 \\ 0 \end{vmatrix}$$

$P_1 \longrightarrow C_1 \Longrightarrow$ $C = K \cdot P \bmod 26$

$C_1 = K \cdot P_1 \bmod 26$

$$= \begin{bmatrix} 7 & 3 \\ 2 & 5 \end{bmatrix} \begin{vmatrix} 1.65 \\ 0.13 \end{vmatrix} \bmod 26$$

$$= \begin{vmatrix} 11.94 \\ 3.98 \end{vmatrix} = \begin{vmatrix} M \\ E \end{vmatrix}$$

$$P_2 = \begin{vmatrix} \frac{5}{29} & \frac{-3}{29} \\ \frac{-2}{29} & \frac{7}{29} \end{vmatrix} \times \begin{vmatrix} 4 \\ 19 \end{vmatrix} \bmod 26$$

$$= \begin{vmatrix} -1.27 \\ 4.31 \end{vmatrix} \bmod 26 =$$

$$C_2 = K \cdot P_2 \bmod 26$$

$$= \begin{vmatrix} 7 & 3 \\ 2 & 5 \end{vmatrix} \begin{vmatrix} -1.27 \\ 4.31 \end{vmatrix} \bmod 26$$

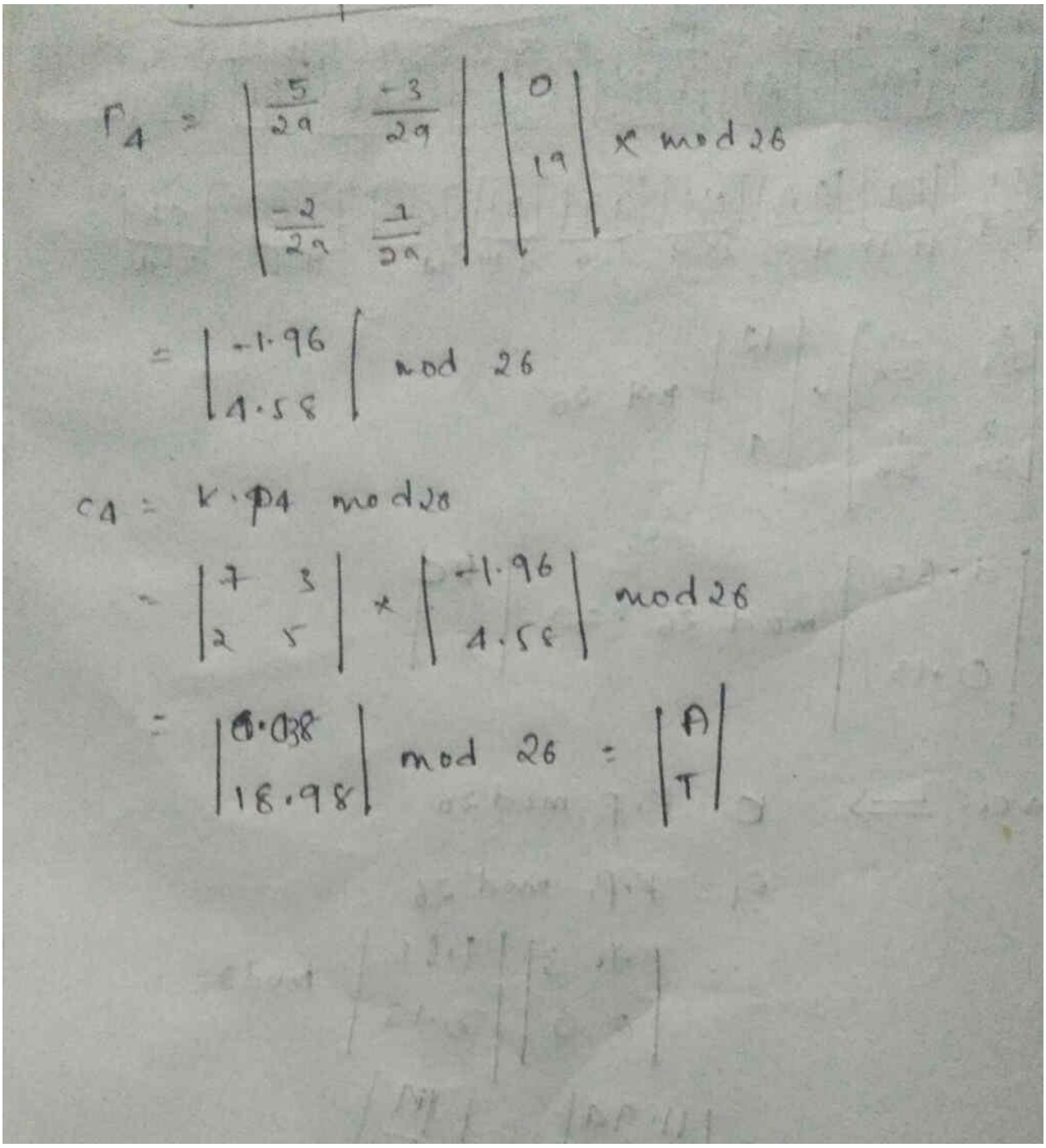
$$= \begin{vmatrix} 4.04 \\ 19.01 \end{vmatrix} \bmod 26 = \begin{vmatrix} E \\ T \end{vmatrix}$$

$$P_3 = \begin{vmatrix} \frac{5}{29} & \frac{-3}{29} \\ \frac{-2}{29} & \frac{7}{29} \end{vmatrix} \times \begin{vmatrix} 12 \\ 4 \end{vmatrix} \bmod 26$$

$$= \begin{vmatrix} -0.55 \\ 0.137 \end{vmatrix} \bmod 26$$

$$C_3 = K \cdot P_3 \bmod 26 = \begin{vmatrix} 7 & 3 \\ 2 & 5 \end{vmatrix} \times \begin{vmatrix} -0.55 \\ 0.13* \end{vmatrix} \bmod 26$$

$$= \begin{vmatrix} -3.46 \\ -0.45 \end{vmatrix} \bmod 26 = \begin{vmatrix} M \\ E \end{vmatrix}$$



Hill шифр нь матриц болон модульчлагдсан арифметик ашиглан блокууд дахь мессежийг шифрлэдэг. Энэ нь полиграфик орлуулалтын шифр бөгөөд шифрлэхэд зориулж үсгүүдийг бүлэглэдэг гэсэн үг юм. Хэдийгээр ухаалаг боловч Hill-ийн үндсэн шифр нь халдлагад өртөмтгий байдаг.

Бодлого3.

10.1 Alice and Bob use the Diffie–Hellman key exchange technique with a common prime q = 157 and a primitive root a = 5.

a. If Alice has a private key XA = 15, find her public key YA.

b. If Bob has a private key XB = 27, find his public key YB.

c. What is the shared secret key between Alice and Bob?

Өгөгдсөн өгөгдлүүд: q=157, a=5, XA=15, XB=27

a). YA= $(a)^{XA}\ mod(q)$=

5^{15}mod(157)=79

YA=79

b). YB=$a^{XB}\ mod$ (q)=

$5^{27} mod(157) = 65$

YB=65

c).Alice-ийн хувьд Shared key

$(YB)^{XA} mod(157)$=

$(65)^{15} mod(157) = 78$

Bob-ийн хувьд Shared key

$(YA)^{XA} mod(157)$=

$(79)^{27} mod(157)$=78

Shared key =78

Нийтийн түлхүүр: Хоёр тал олон нийтэд мэдэгдэж буй анхны тоо (p) ба анхдагч язгуур (g) модуль p дээр санал нийлдэг. Эдгээр утгууд нь бэлэн бөгөөд нууцлал шаарддаггүй.

Хувийн түлхүүрүүд: Тал бүр санамсаргүй хувийн түлхүүрийг бие даан сонгоно (Алис, Боб нарын хувьд a, b). Эдгээр хувийн түлхүүрүүд нь маш чухал бөгөөд нууц байх ёстой.

Түлхүүр солилцоо:

Алис g^a (модуль p) тооцоолж, энэ утгыг Боб руу олон нийтэд илгээдэг.

Боб g^b (модуль p) тооцоолж, Алис руу олон нийтэд илгээдэг.

Хуваалцсан нууц: Алис, Боб хоёр одоо хүлээн авсан мэдээлэл болон хувийн түлхүүрээ ашиглан хуваалцсан нууц түлхүүрийг тооцоолох боломжтой.

Алис: (g^b)^a (модуль p)

Боб: (g^a)^b (модуль p)

Модуль экспонентацийн математик шинж чанаруудын улмаас хоёр тал хэзээ ч шууд солилцдоггүй байсан ч ижил хуваалцсан нууц түлхүүрт хүрэх болно. Энэ түлхүүрийг дараа нь тэгш хэмт түлхүүрийн алгоритмуудыг ашиглан аюулгүй холболтод ашиглаж болно.

Diffie-Hellman протокол нь хэд хэдэн давуу талыг санал болгодог.

Түлхүүрийг найдвартай солилцох: Түлхүүрийг хэзээ ч нээлттэй дамжуулахгүйгээр тогтоодог бөгөөд энэ нь харилцаа холбооны сувгийг чагнахад тэсвэртэй болгодог.

Урьдчилан итгэлцэл байхгүй: Талууд түлхүүр солилцохын тулд өмнө нь итгэмжлэгдсэн дэд бүтэцтэй байх шаардлагагүй.

Төгс дамжуулах нууцлал: Хувийн түлхүүр дараа нь алдагдсан ч өмнө нь суулгасан түлхүүрүүд аюулгүй хэвээр байна.

Гэсэн хэдий ч, Diffie-Hellman өөрөө өгөгдөл шифрлэлт эсвэл баталгаажуулалт хийдэггүй гэдгийг анхаарах нь чухал юм. Энэ нь хуваалцсан нууцыг тогтоодог бөгөөд үүнийг дараа нь бусад криптограф алгоритмуудтай аюулгүй харилцаа холбоонд ашиглаж болно.

Програмын хэсэг:

7.1) Create software that can encrypt and decrypt in cipher block chaining mode using one of the following ciphers: affine modulo 256, Hill modulo 256, S-DES, DES. Test data for S-DES using a binary initialization vector of 1010 1010. A binary plaintext of 0000 0001 0010 0011 encrypted with a binary key of 01111 11101 should give a binary plaintext of 1111 0100 0000 1011. Decryption should work correspondingly

```
p10 = [2, 4, 1, 6, 3, 9, 0, 8, 7, 5]
p8 = [5, 2, 6, 3, 7, 4, 9, 8]
ip8 = [1, 5, 2, 0, 3, 7, 4, 6]
ep = [3, 0, 1, 2, 1, 2, 3, 0]
p4 = [1, 3, 2, 0]
p1 = [3, 0, 2, 4, 6, 1, 7, 5]
s0 = [['01', '00', '11', '10'], ['11', '10', '01', '00'],
      ['00', '10', '01', '11'], ['11', '01', '11', '10']]
s1 = [['00', '01', '10', '11'], ['10', '00', '01', '11'],
```

```
     ['11', '00', '01', '00'], ['10', '01', '00', '11']]

import lab6 as SDES

initVector = "10101010"

def encrypt(plainText):
    binArray = plainText
    vec = initVector
    l = SDES.XORstr(vec, binArray[0])
    cbc = []
    cbc.append(SDES.EncryptChar(l, key))
    for i in range(len(binArray) - 1):
        l = SDES.XORstr(cbc[i], binArray[i + 1])
        cbc.append(SDES.EncryptChar(l, key))
    return cbc
def decrypt( cipherText):
    vec = initVector
    cbc = []
    l = SDES.XORstr(SDES.DecryptChar(cipherText[0], key), vec)
    cbc.append(l)
    for i in range(1, len(cipherText)):
        lo = SDES.DecryptChar(cipherText[i], key)
        l = SDES.XORstr(lo, cipherText[i - 1])
        cbc.append(l)
    return cbc

key = SDES.GenerateKey("0111111101")
plainText = ["00000001", "00100011"]
cipherText = encrypt(plainText)
print("cipher", cipherText)
plainText = decrypt(cipherText)
print("plain", plainText)
```

Лаборатори 6 болон feistel ашиглав

Үр дүнд:

```
PS C:\Users\warlo\OneDrive\Desktop\hicheel\nuttslal> & C:/Users/warlo/AppData/Local/Programs/Pyth
p/hicheel/nuttslal/bd/7_1.py
['0110', '1000', '0110', '0101', '0110', '1100', '0110', '1100', '0110', '1111', '0110', '1111']
b3bfb5b5b7b7
['0110', '1000', '0110', '0101', '0110', '1100', '0110', '1100', '0110', '1111', '0110', '1111']
helloo
cipher ['11110100', '00001011']
plain ['00000001', '00100011']
PS C:\Users\warlo\OneDrive\Desktop\hicheel\nuttslal>
```

Дүгнэлт: Энэхүү бие даалтын хүрээнд Chinese Remainder Theorom , Hill cipher , Diffie–Hellman key exchange technique , SDES ,DES гэх шифэрлэлтийн аргуудыг судалж . Судалсан онол түүний аргуудыг ашиглан энэхүү Бие Даалтын даалгавруудыг хийж гүйцэтгэв.