BIL470 KRİPTOGRAFİ VE BİLGİSAYAR GÜVENLİĞİ PROJESİ BÖLÜM 2 RAPORU

Fatih Selim YAKAR - 161044054

GİRİŞ

A ve B şıkkı:

Projede belirtilen A ve B şıklarını uygulamak amacıyla piton programlama dili kullanarak kodlar yazıldı. Genel olarak kod 2 adet sınıf içeriyordu. Sınıflardan biri AES algoritmasının uygulanmasını gerçekliyordu ve sınıflardan diğeri ise AES algoritmasının çalışma modlarını gerçekliyordu. Bu bağlamda çalışma görselleri ve gerçeklenmesi hususunda detayları raporun devamında görebilirsiniz.

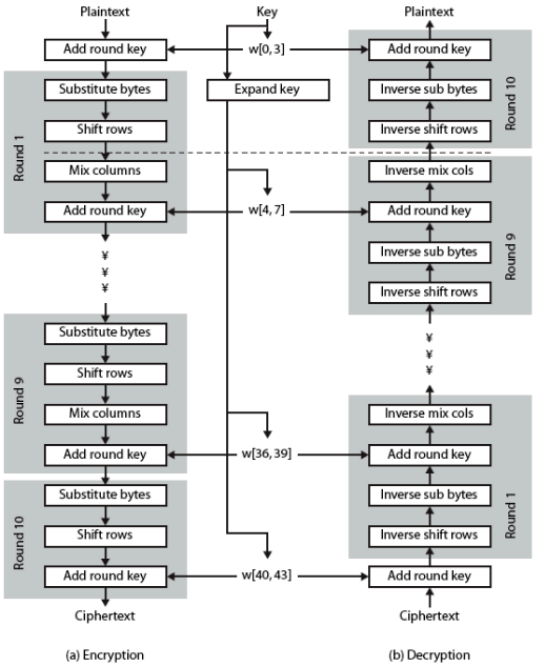
C ve D şıkkı:

Projede belirtilen özet alma işlemini kurgularken önce dosyanın özetini alma ve ardından özeti alınan metnin şifrelemesi adına çeşitli kodlar yazıldı. Özeti alınıp şifrelenen metnin dosyanın sonuna yazılmasından sonra aynı anahtar dahilinde dosyanın bütünlüğünün kontrolünü sağlamak için ise dosya sonuna yazılan metnin deşifrelenip, ardından dosyanın ana metninin özetini alıp ardından bu iki metnin karşılaştırılması yöntemi kullanıldı. Bu bağlamda çalışma görselleri ve gerçeklenmesi hususunda detayları raporun devamında görebilirsiniz.

KAYNAK KODLARI VE AÇIKLAMALARI

AES algoritması:

AES algoritmasını gerçeklemek amacıyla aşağıdaki hedef iskelet kullanıldı:



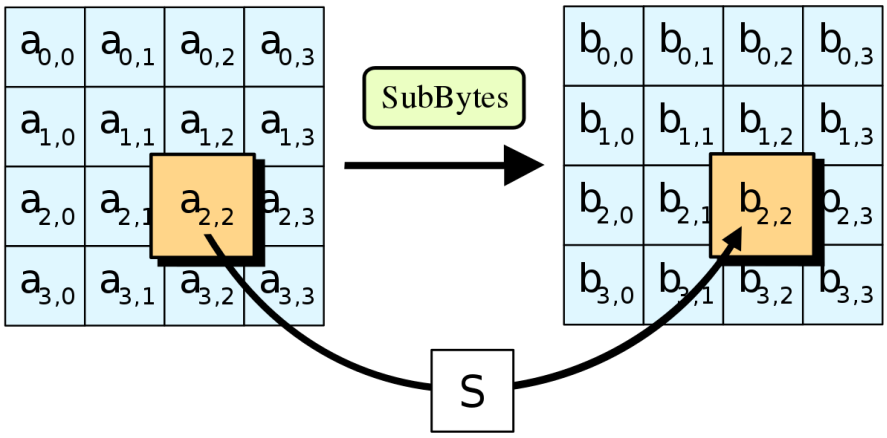
AES algoritmasını gerçeklemek için ise bir adet sınıf yazıldı. Bu sınıf genel olarak AES algoritmasının gereği olan bazı operasyonları gerçekliyordu. Bu operasyonlar ise şunlardı:

- Baytları yerine koyma
- Satırları kaydırma
- Sütunları karıştırma
- Tur anahtarı ekleme
- Anahtarı genişletme

Operasyonlar detaylıca açıklayacak ve kaynak kodları üzerinden gösterilmesi şu şekildedir:

1. **Baytları yerine koyma**

Önceden belirlenmiş bir tablo ile parametre olarak gelen tablo arası endeks bazlı yerine koyma yapılır. Bu tablo 16x16'lık 256 8-bit değerlerin tüm permütasyonlarını içeren bir kaynak olarak kullanılır ve $Gallois\ Field\ 2^8$ ($GF(2^8)$ olarakta ifade edilebilir.

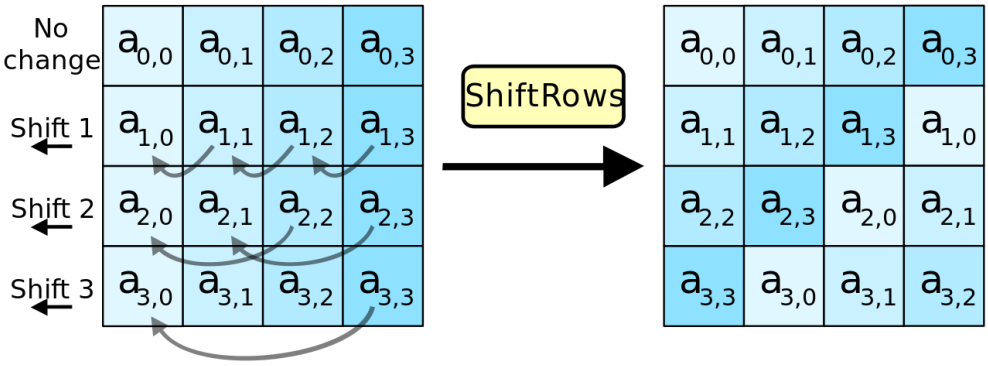


Şifrelemede bu şekilde kullanılırken deşifreleme de ise bu matrisin tersi için aynı işlemler yapılır. Gerçeklenmesi ise şu şekildedir:

```
# Substitutes all the values from the table with the value in the SBox table
def substitudeBytes(self, table, isInv):
    if isInv:
        for i in range(16):
            table[i] = self.getInvertedSBoxValue(table[i])
    else:
        for i in range(16):
            table[i] = self.getSBoxValue(table[i])

    return table
```

2. Satırları kaydırma

Dairesel şekilde bayt kaydırma yapar. Şifrelemede sola kaydırma yaparken, deşifreleme de ise sağa kaydırma yapar. Durum/tablo sütunlar tarafından işlendiğinden, bu adım sütunlar arasındaki permütasyona izin verir.

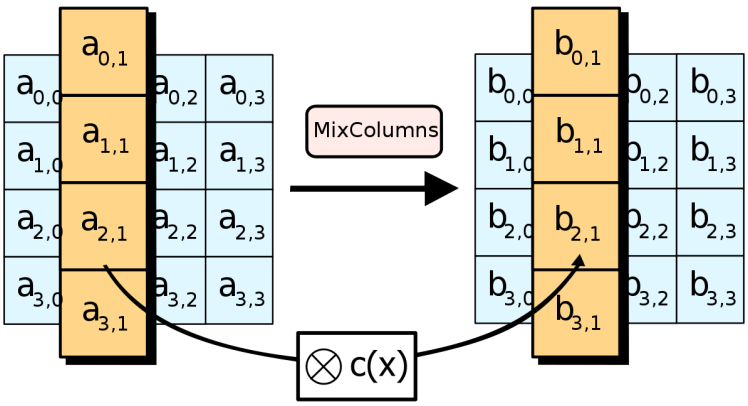


Gerçeklenmesi ise şu şekildedir:

```
# Iterates over the 4 rows and  with indexed row
def shiftRows(self, table, isInv):
    for i in range(4):
        table = self.shiftRow(table, i*4, i, isInv)
    return table

# Each iteration shifts the row to the left
def shiftRow(self, table, tablePointer, size, isInv):
    for i in range(size):
        if isInv:
            table[tablePointer:tablePointer+4] = \
                    table[tablePointer+3:tablePointer+4] + \
                    table[tablePointer:tablePointer+3]
        else:
            table[tablePointer:tablePointer+4] = \
                    table[tablePointer+1:tablePointer+4] + \
                    table[tablePointer:tablePointer+1]
    return table
```

3. Sutünları karıştırma

Her sütün farklı şekilde ele alınır. Her bayt, sütundaki 4 baytın tümüne bağlı bir değerle değiştirilir. Efektif olarak $GF(2^8)$ kullanarak bir matris çarpımı yapar.

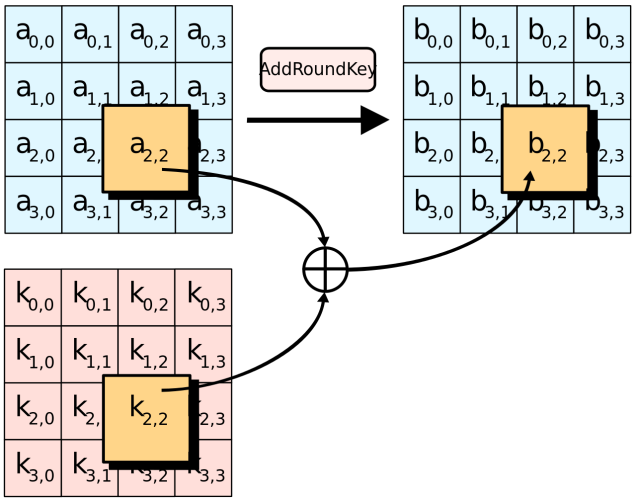


Deşifreleme yaparken de çarpımı yaptığı matrisin tersi ile yapar.Gerçeklenmesi ise şu şekildedir:

```python
# Galois multiplication of the matrix
def mixColumns(self, table, isInv):
    # For all columns
    for i in range(4):
        column = table[i:i+16:4]
        column = self.mixColumnProcess(column, isInv)
        table[i:i+16:4] = column

    return table

# Galois multiplication of 1 column of the matrix
def mixColumnProcess(self, column, isInv):
    columnList = list(column)
    if isInv:
        mult = [14, 9, 13, 11]
    else:
        mult = [2, 1, 1, 3]

    column[0] = self.galoisMultiplication(columnList[0], mult[0]) ^ self.galoisMultiplication(columnList[3], mult[1]) ^ \
                self.galoisMultiplication(columnList[2], mult[2]) ^ self.galoisMultiplication(columnList[1], mult[3])
    column[1] = self.galoisMultiplication(columnList[1], mult[0]) ^ self.galoisMultiplication(columnList[0], mult[1]) ^ \
                self.galoisMultiplication(columnList[3], mult[2]) ^ self.galoisMultiplication(columnList[2], mult[3])
    column[2] = self.galoisMultiplication(columnList[2], mult[0]) ^ self.galoisMultiplication(columnList[1], mult[1]) ^ \
                self.galoisMultiplication(columnList[0], mult[2]) ^ self.galoisMultiplication(columnList[3], mult[3])
    column[3] = self.galoisMultiplication(columnList[3], mult[0]) ^ self.galoisMultiplication(columnList[2], mult[1]) ^ \
                self.galoisMultiplication(columnList[1], mult[2]) ^ self.galoisMultiplication(columnList[0], mult[3])
    return column
```

4. Tür Anahtarı Ekleme

Durum veya tablo ile 128-bitlik tur anahtarını XOR işlemine tabi tutar. Bu durum sonucu yine sutünlar arası bir işlem yapılmış olur. En basit şekilde tasarlanmıştır.

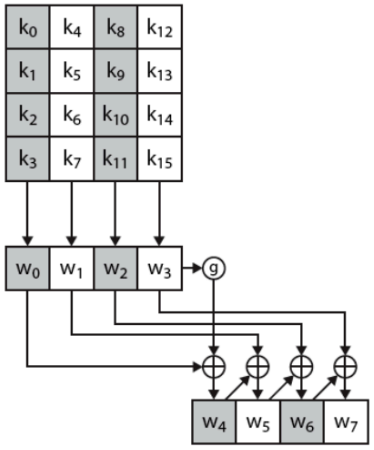


Deşifreleme yaparken de XOR'u anahtarın kendi tersi ile yapar.Gerçeklenmesi ise şu şekildedir:

```
# Applies XOR the round key to the table.
def addRoundKey(self, table, roundKey):
    for i in range(16):
        table[i] ^= roundKey[i]
    return table
```

5. Anahtar genişletme

128,192,256 anahtarlarını 176,208,240 bayt anahtarına genişletir. Rijndael türü anahtar genişletmesini uygular. Öncelikle anahtarda 4 tane 32-bit kopyalar. Ardından dörtten ilk 32-bit'i sırayla döndürme,s-box,xor işlemlerine tabi tutulur. Ardından gelenlerde ise 2.için 1.ile xor, 3.için 2.ile xor işlemlerine tabi tutularak devamı bulunur.



Gerçeklenmesi ise şu şekildedir:

```
# Key schedule base algorithm.
def keySchedule(self, word, iteration):
    # rotate
    word = self.keyScheduleRotate(word)
    # substitution
    for i in range(4):
        word[i] = self.getSBoxValue(word[i])
    # XOR the output
    word[0] = word[0] ^ self.getRconValue(iteration)
    return word

# Expands an 128,192,256 key into an 176,208,240 bytes key. Applies Rijndael type key expansion.
def expandKey(self, key, size, expandedKeySize):
    currentSize = 0
    rconIteration = 1
    expandedKey = [0] * expandedKeySize

    # set the bytes of the expanded key to the input key
    for j in range(size):
        expandedKey[j] = key[j]
    currentSize += size

    while currentSize < expandedKeySize:
        temp = expandedKey[currentSize-4:currentSize]
        # keySchedule schedule to temp
        if currentSize % size == 0:
            temp = self.keySchedule(temp, rconIteration)
            rconIteration += 1
        # If key is 256 bit then add extra sbox
        if size == self.keySize["KEY_SIZE_256"] and ((currentSize % size) == 16):
            for l in range(4):
                temp[l] = self.getSBoxValue(temp[l])
        # XOR temp with the block 16,24,32 bytes
        for m in range(4):
            expandedKey[currentSize] = expandedKey[currentSize - size] ^ \
                    temp[m]
            currentSize += 1

    return expandedKey
```

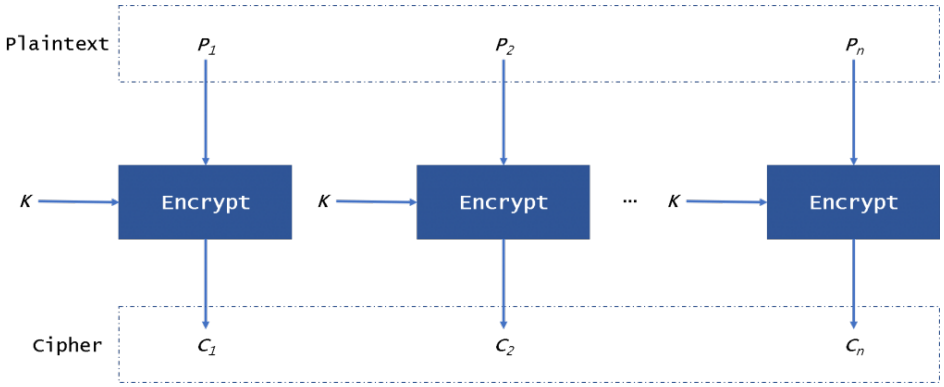
AES modları:

AES algoritmasına ek olarak ECB(varsayılan),OFB, CBC ve CFB modları gerçeklendi. Bu modlar ise basitçe şu şekildeydi:

1. ECB modu

Mesajı şifreli olan birbirinden bağımsız bloklara ayırıyordu. Ve her blok birbirinden bağımsız şekilde şifreleniyordu ve her blok kod kitabı olarak isimlendiriliyordu. Formül olarak basitçe şu şekilde işlem yapıyordu:

$$C_i = AES_{K1}(P_i)$$



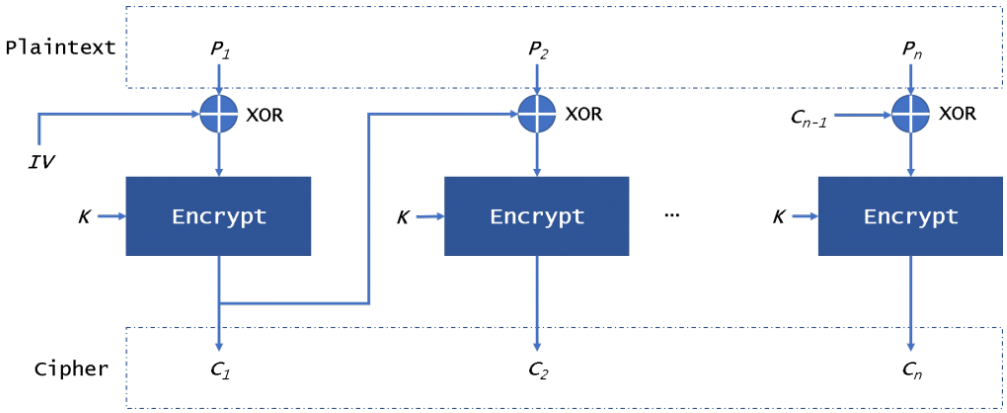
Gerçeklemesi ise şu şekildeydi:

```
# Mode ECB
elif mode == self.modeOfOperation["ECB"]:
    cipheredText=[]
    for i in range(16):
        if i>=len(ciphertext):
            cipheredText.append(0)
        else:
            cipheredText.append(ciphertext[i])
    plaintext = self.aes.decrypt(cipheredText, key, size)
    firstRound = False
    for k in range(16):
        charOutput.append(chr(plaintext[k]))
    inputText = cipheredText
```

2. CBC modu

ECB modunda da olduğu gibi CBC moduda mesajı bloklara ayırıyordu. Fakat ECB modundan farkı aslında bir Başlangıç Vektörü ile başlaması ve blok şifrelemesinin birbirine bağlı olmasından kaynaklanıyordu. Formülü ve diagramı şu şekildeydi:

$$C_i = AES_{K1}(P_i \text{ XOR } C_{i-1})$$
$$C_{-1} = IV$$



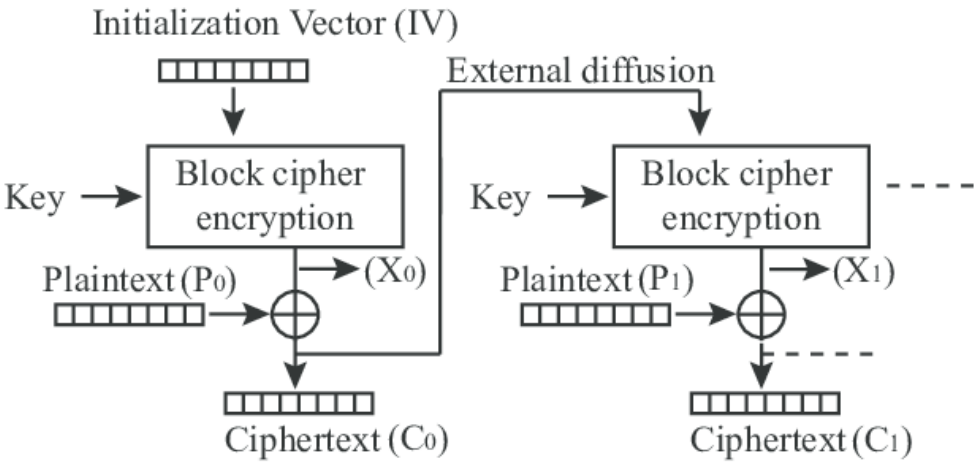
Gerçeklemesi ise şu şekildeydi:

```
# Mode CBC
elif mode == self.modeOfOperation["CBC"]:
    output = self.aes.decrypt(ciphertext, key, size)
    for i in range(16):
        if firstRound:
            plaintext[i] = initialVector[i] ^ output[i]
        else:
            plaintext[i] = inputText[i] ^ output[i]
    firstRound = False
    if originalsize is not None and originalsize < end:
        for k in range(originalsize-start):
            charOutput.append(chr(plaintext[k]))
    else:
        for k in range(end-start):
            charOutput.append(chr(plaintext[k]))
    inputText = ciphertext
```

3. CFB modu

CFB modu önceki diğer modlara nazaran blok tipinde değil akış şeklinde ele alıyordu. Ve yine bu modda da bir Başlangıç Vektörü ve diğer bloklara bağlılık mevcuttu.

$$C_i = P_i \text{ XOR } AES_{K1}(C_{i-1})$$
$$C_{-1} = IV$$



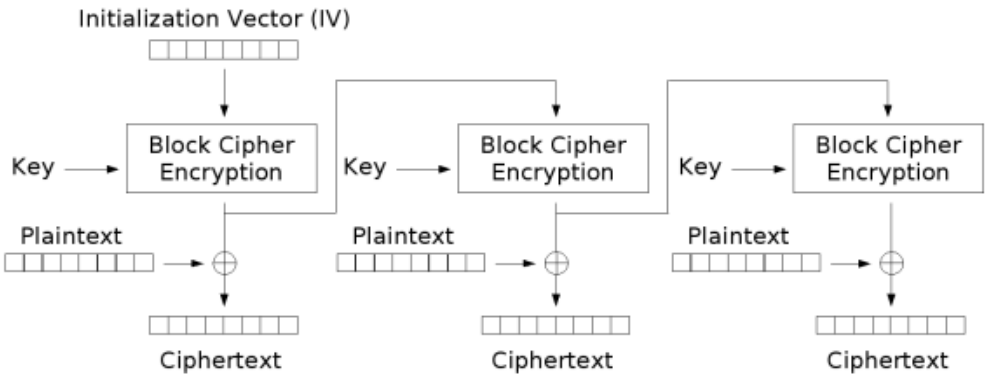
Gerçeklemesi ise şu şekildeydi:

```
# Mode CFB
if mode == self.modeOfOperation["CFB"]:
    if firstRound:
        output = self.aes.encrypt(initialVector, key, size)
        firstRound = False
    else:
        output = self.aes.encrypt(inputText, key, size)
    for i in range(16):
        if len(output)-1 < i:
            plaintext[i] = 0 ^ ciphertext[i]
        elif len(ciphertext)-1 < i:
            plaintext[i] = output[i] ^ 0
        elif len(output)-1 < i and len(ciphertext) < i:
            plaintext[i] = 0 ^ 0
        else:
            plaintext[i] = output[i] ^ ciphertext[i]
    for k in range(end-start):
        charOutput.append(chr(plaintext[k]))
    inputText = ciphertext
```

4. OFB modu

Bu modun diğerlerinden farkı geri beslemeli bir çıktısı olup mesajı bununla XOR işlemine tabi tutmasıydı. Ve bu geri beslemenin aslında mesajdan bağımsız olmasıydı.

$$C_i = P_i \text{ XOR } O_i$$
$$O_i = AES_{K1}(O_{i-1})$$
$$O_{-1} = IV$$



Output Feedback (OFB) mode encryption

Gerçeklemesi ise şu şekildeydi:

```
# Mode OFB
elif mode == self.modeOfOperation["OFB"]:
    if firstRound:
        output = self.aes.encrypt(initialVector, key, size)
        firstRound = False
    else:
        output = self.aes.encrypt(inputText, key, size)
    for i in range(16):
        if len(output)-1 < i:
            plaintext[i] = 0 ^ ciphertext[i]
        elif len(ciphertext)-1 < i:
            plaintext[i] = output[i] ^ 0
        elif len(output)-1 < i and len(ciphertext) < i:
            plaintext[i] = 0 ^ 0
        else:
            plaintext[i] = output[i] ^ ciphertext[i]
    for k in range(end-start):
        charOutput.append(chr(plaintext[k]))
    inputText = output
```

Özet alma,şifreleme ve kontrol etme:

Özet alma işlemi olarak basitçe önce ikinin katı sayısınca olcak şekilde girdi genişletildi ardından bu girdi 16 karakter olana kadar ilk yarısı ve ikinci yarısı xor işlemine tabi tutuldu. Böylece girdi olarak gelen mesajın uzunluğu sabit bir metne indirgendi. Ardından şifrelenip dosyanın sonuna eklendi. Kontrol için ise özet alınıp şifrelenen metin deşifrelendikten sonra asıl metnin tekrardan özeti alınarak karşılaştırıldı. Bu karşılaştırma sonucu aynı çıkarsa bütünlük korunmuş oldu. Sonuç

farklı çıktığı durumda ise bütünlük korunmamış oldu. Bu bağlamda kullanılan fonksiyonların gerçeklenmesi ise şu şekildeydi:

```
class hashAndControl(object):
    # Controls if n is a power of 2
    def isPowerOf2(self,n):
        return (n & (n - 1) == 0) and n != 0

    # Expands state string to power of 2
    def expand16xbit(self,state):
        counter=0
        if(type(state)== str):
            while((self.isPowerOf2(len(state))==False) or len(state)<16):
                state+=chr(counter)
                counter+=1
                counter%=256
        else:
            while((self.isPowerOf2(len(state))==False) or len(state)<16):
                state.append(counter)
                counter+=1
                counter%=256

        return state
```

```
# It expands the string that comes as a parameter, divides it into half and converts it to 16 digits by xor.
def hashFunction(self,state):
    state = self.expand16xbit(state)
    #print state
    if(len(state)==16):
        return state

    size=len(state)/2
    newState=[-1]*size

    counter=0
    while(size>=16):
        if(counter==0):
            for i in range(size):
                if(type(state)==str):
                    newState[i]=ord(state[i])^ord(state[i+size])
                else:
                    newState[i]=state[i]^state[i+size]
        else:
            for i in range(size):
                newState[i]=newState[i]^newState[i+size]
        counter+=1
        size/=2
    return newState[0:16]
```

```
# It hashes,encrypts then writes the end of the file
def hashAndEncrypt(self,fileName,key):
    f=open(fileName,"rb")
    willHash = f.read()
    hashedList = self.hashFunction(willHash)
    print "HashedList",hashedList
    f.close()

    with open(fileName,"a") as fh:
        fh.seek(0,2)
        hashed = ' '.join([str(elem) for elem in hashedList])
        cipher=AES.runAesEncrypt(hashed,key,"ECB")
        cipherReadable=[ord(x) for x in cipher]
        toWrite=' '.join([str(elem) for elem in cipherReadable])
        print "ENCRYPT",toWrite
        fh.write(toWrite)

    fh.close()

    return hashedList,len(toWrite)
```

```
def controlChange(self,fileName,key,size):
    f=open(fileName,"rb")
    f.seek(-size,2)
    fileString = f.read()
    #print "CONTROL",fileString
    lst = list(fileString.split(" "))
    cipherText=""
    for i in range(0,len(lst)):
        lst[i]=int(lst[i])
        cipherText+=chr(lst[i])

    plaintext=AES.runAesDecrypt(cipherText,key,"ECB")

    f.seek(0,0)
    fileString=f.read()
    willHash=""
    for i in range(len(fileString)-(size-1)):
        willHash+=fileString[i]

    hashedList = self.hashFunction(willHash[0:len(willHash)-1])
    hashedString = ' '.join([str(elem) for elem in hashedList])

    f.close()

    print hashedString
    print plaintext

    for i in range(len(hashedString)):
        if hashedString[i]!=plaintext[i]:
            return True

    return False
```

Testler ve çalışma görüntüleri:

A ve B şıkkı için:

```
Mod: ECB
Anahtar boyutu: 16
Sifrelenmemis metin: Merhaba, bu sifrelencek metin
Rastgele Anahtar: [151, 232, 138, 143, 107, 50, 12, 244, 203, 209, 41, 105, 28, 148, 101, 52]
Sifrelenmis Metin: [55, 21, 255, 137, 23, 55, 80, 178, 119, 136, 131, 228, 37, 188, 33, 22, 90,
 122, 231, 168, 190, 133, 32, 76, 125, 68, 245, 170, 9, 90, 241, 116, 53, 241, 12, 44, 125, 219
, 43, 109, 98, 185, 110, 162, 115, 65, 200, 121]
Desifrelenmis Metin: Merhaba, bu sifrelencek metin

Mod: CBC
Anahtar boyutu: 16
Sifrelenmemis metin: Merhaba, bu sifrelencek metin
Rastgele Anahtar: [118, 84, 145, 23, 229, 175, 139, 231, 140, 46, 161, 247, 173, 183, 164, 131]
Sifrelenmis Metin: [86, 214, 197, 217, 95, 160, 105, 68, 25, 179, 24, 118, 91, 187, 127, 115, 3
2, 246, 142, 164, 3, 6, 92, 42, 23, 47, 129, 31, 92, 145, 227, 166, 146, 26, 165, 170, 23, 150,
 110, 255, 223, 57, 181, 72, 32, 24, 48, 252]
Desifrelenmis Metin: Merhaba, bu sifrelencek metin

Mod: OFB
Anahtar boyutu: 16
Sifrelenmemis metin: Merhaba, bu sifrelencek metin
Rastgele Anahtar: [79, 6, 98, 245, 147, 15, 144, 63, 101, 40, 37, 104, 115, 42, 210, 90]
Sifrelenmis Metin: [155, 238, 90, 91, 220, 194, 249, 211, 6, 40, 202, 161, 96, 89, 8, 10, 138,
85, 42, 46, 77, 136, 66, 136, 220, 43, 55, 170, 83, 13, 122, 6, 212, 135, 246, 61, 207, 151, 17
0, 30, 204, 132, 173, 200, 221]
Desifrelenmis Metin: Merhaba, bu sifrelencek metin

Mod: CFB
Anahtar boyutu: 16
Sifrelenmemis metin: Merhaba, bu sifrelencek metin
Rastgele Anahtar: [95, 94, 164, 27, 36, 54, 135, 240, 42, 176, 94, 28, 103, 175, 245, 208]
Sifrelenmis Metin: [206, 175, 3, 141, 38, 36, 251, 247, 5, 76, 223, 219, 11, 82, 190, 98, 79, 1
3, 213, 150, 235, 253, 58, 35, 183, 57, 130, 148, 61, 166, 75, 238, 79, 157, 138, 70, 137, 34,
39, 233, 122, 55, 32, 250, 227]
Desifrelenmis Metin: Merhaba, bu sifrelencek metin
```

```
Mod: ECB
Anahtar boyutu: 24
Sifrelenmemis metin: Merhaba, bu sifrelencek metin
Rastgele Anahtar: [37, 203, 45, 188, 183, 33, 254, 65, 96, 100, 234, 222, 44, 115, 40, 158, 4,
172, 169, 106, 156, 178, 2, 201]
Sifrelenmis Metin: [67, 187, 205, 96, 10, 157, 144, 164, 125, 42, 54, 84, 179, 175, 92, 85, 48,
 226, 7, 222, 152, 199, 21, 26, 79, 217, 201, 219, 168, 74, 112, 129, 41, 208, 189, 245, 188, 1
34, 65, 1, 18, 15, 55, 72, 167, 129, 16, 175]
Desifrelenmis Metin: Merhaba, bu sifrelencek metin

Mod: CBC
Anahtar boyutu: 24
Sifrelenmemis metin: Merhaba, bu sifrelencek metin
Rastgele Anahtar: [152, 42, 17, 72, 243, 88, 183, 164, 247, 97, 90, 164, 46, 133, 179, 148, 218
, 86, 215, 26, 56, 232, 219, 246]
Sifrelenmis Metin: [229, 14, 79, 30, 112, 24, 135, 58, 216, 39, 56, 149, 120, 66, 141, 199, 179
, 176, 207, 128, 203, 160, 150, 250, 165, 199, 166, 235, 6, 215, 130, 123, 208, 162, 162, 156,
102, 166, 157, 224, 58, 80, 155, 220, 123, 173, 191, 109]
Desifrelenmis Metin: Merhaba, bu sifrelencek metin

Mod: OFB
Anahtar boyutu: 24
Sifrelenmemis metin: Merhaba, bu sifrelencek metin
Rastgele Anahtar: [224, 83, 218, 184, 97, 62, 175, 13, 162, 137, 230, 236, 121, 183, 105, 68, 1
16, 150, 24, 232, 205, 58, 164, 16]
Sifrelenmis Metin: [128, 71, 209, 172, 246, 50, 224, 123, 83, 255, 74, 167, 187, 162, 239, 134,
 94, 171, 160, 225, 39, 229, 40, 73, 158, 235, 248, 122, 237, 135, 50, 184, 183, 25, 115, 89, 4
2, 243, 12, 70, 59, 85, 126, 70, 86]
Desifrelenmis Metin: Merhaba, bu sifrelencek metin

Mod: CFB
Anahtar boyutu: 24
Sifrelenmemis metin: Merhaba, bu sifrelencek metin
Rastgele Anahtar: [33, 9, 69, 9, 88, 186, 130, 47, 18, 185, 43, 124, 225, 105, 245, 252, 184, 9
8, 120, 186, 236, 53, 149, 96]
Sifrelenmis Metin: [224, 161, 243, 176, 85, 200, 156, 158, 250, 49, 220, 25, 72, 164, 69, 121,
68, 64, 73, 239, 73, 234, 24, 132, 220, 24, 135, 220, 91, 250, 193, 137, 111, 200, 234, 121, 16
0, 167, 20, 53, 131, 72, 133, 34, 236]
Desifrelenmis Metin: Merhaba, bu sifrelencek metin
```

```
Mod: ECB
Anahtar boyutu: 32
Sifrelenmemis metin: Merhaba, bu sifrelencek metin
Rastgele Anahtar: [65, 226, 169, 140, 51, 60, 21, 243, 183, 0, 244, 144, 251, 72, 235, 134, 65,
 111, 11, 35, 253, 65, 247, 158, 228, 39, 149, 97, 231, 38, 251, 12]
Sifrelenmis Metin: [124, 163, 255, 122, 46, 130, 33, 188, 118, 108, 133, 234, 0, 10, 57, 53, 23
9, 252, 51, 109, 94, 59, 121, 21, 99, 89, 90, 83, 159, 229, 233, 175, 115, 21, 49, 233, 112, 19
6, 137, 99, 29, 242, 166, 189, 250, 181, 32, 43]
Desifrelenmis Metin: Merhaba, bu sifrelencek metin

Mod: CBC
Anahtar boyutu: 32
Sifrelenmemis metin: Merhaba, bu sifrelencek metin
Rastgele Anahtar: [40, 204, 122, 152, 145, 168, 230, 112, 162, 139, 223, 201, 230, 5, 128, 131,
 171, 1, 85, 253, 153, 50, 34, 211, 158, 25, 210, 55, 12, 51, 115, 232]
Sifrelenmis Metin: [105, 144, 96, 115, 155, 215, 78, 24, 24, 55, 3, 7, 203, 241, 236, 87, 126,
153, 214, 141, 186, 130, 225, 204, 16, 226, 38, 163, 93, 129, 231, 216, 154, 59, 143, 154, 65,
45, 235, 104, 230, 140, 125, 38, 60, 95, 203, 155]
Desifrelenmis Metin: Merhaba, bu sifrelencek metin

Mod: OFB
Anahtar boyutu: 32
Sifrelenmemis metin: Merhaba, bu sifrelencek metin
Rastgele Anahtar: [9, 120, 41, 184, 64, 211, 200, 220, 219, 15, 96, 96, 154, 43, 85, 44, 58, 2,
 74, 94, 49, 196, 177, 69, 122, 154, 59, 21, 209, 19, 78, 50]
Sifrelenmis Metin: [61, 189, 231, 142, 30, 187, 124, 122, 200, 132, 187, 117, 127, 69, 11, 219,
 100, 244, 179, 50, 197, 162, 143, 56, 223, 218, 255, 56, 187, 24, 202, 98, 103, 140, 102, 251,
 251, 178, 121, 120, 61, 243, 86, 132, 187]
Desifrelenmis Metin: Merhaba, bu sifrelencek metin

Mod: CFB
Anahtar boyutu: 32
Sifrelenmemis metin: Merhaba, bu sifrelencek metin
Rastgele Anahtar: [111, 117, 157, 107, 150, 95, 21, 57, 0, 63, 52, 147, 75, 141, 32, 174, 134,
189, 144, 158, 29, 51, 139, 173, 198, 210, 253, 0, 1, 220, 131, 193]
Sifrelenmis Metin: [241, 252, 212, 173, 154, 70, 125, 95, 83, 176, 222, 44, 229, 176, 184, 211,
 191, 67, 99, 30, 123, 130, 198, 175, 115, 121, 57, 78, 195, 126, 49, 104, 135, 189, 201, 250,
37, 100, 69, 36, 177, 178, 219, 202, 96]
Desifrelenmis Metin: Merhaba, bu sifrelencek metin
```

C ve D şıkkı için:

- TXT dosyası için bütünlük testleri

```
Dosya ismi: sample.txt
Anahtar: [145, 102, 114, 244, 150, 49, 122, 3, 191, 182, 4, 131, 132, 27, 238, 47, 207, 3, 187, 123, 9, 5,
121, 39, 22, 184, 100, 43, 169, 125, 175, 210]
**Ozetleme,sifreleme ve dosyaya yazma islemi:
Ozetlenmis metin: [83, 139, 137, 32, 73, 68, 44, 248, 133, 164, 193, 68, 240, 102, 134, 208]
Sifrelenmis metin: 122 75 247 44 55 81 159 165 108 251 56 64 133 252 89 6 67 179 40 94 255 57 97 30 21 183
34 63 64 242 191 2 118 150 65 115 57 81 94 117 36 242 5 104 23 231 177 234 158 88 249 83 41 105 212 19 146
121 176 45 254 213 170 187 155 194 232 166 82 164 32 34 130 162 241 202 84 210 241 231
**Dosya degistirilmedi.**
**Kontrol islemi**:
Ana metinden olusturulmus ozet metin: 83 139 137 32 73 68 44 248 133 164 193 68 240 102 134 208
Desifre edilmis ozet metin: 83 139 137 32 73 68 44 248 133 164 193 68 240 102 134 208
Degistirildi mi?: False
```

```
Dosya ismi: sample.txt
Anahtar: [66, 120, 126, 57, 214, 59, 81, 252, 239, 249, 98, 225, 145, 126, 153, 112, 186, 106, 32, 192, 175
, 155, 175, 25, 18, 174, 194, 50, 199, 93, 241, 143]
**Ozetleme,sifreleme ve dosyaya yazma islemi:
Ozetlenmis metin: [83, 139, 137, 32, 73, 68, 44, 248, 133, 164, 193, 68, 240, 102, 134, 208]
Sifrelenmis metin: 88 227 217 59 91 220 129 64 96 234 76 160 64 206 255 218 162 227 36 164 4 194 18 17 45 1
0 22 176 61 163 48 112 250 11 166 32 243 144 223 244 45 3 158 15 87 246 228 144 154 143 236 179 118 29 217
3 4 113 105 168 158 136 167 2 227 204 201 249 205 213 139 224 222 68 109 189 41 192 78 73
**Dosya degistirildi.
**Kontrol islemi:
Ana metinden olusturulmus ozet metin: 70 252 104 147 208 219 65 7 167 200 207 174 109 84 47 83
Desifre edilmis ozet metin: 83 139 137 32 73 68 44 248 133 164 193 68 240 102 134 208
Degistirildi mi?: True
fatihselimyakar@Fatihs-MacBook-Pro proje %
```

- DOCX dosyası için bütünlük testleri

```
Dosya ismi: sample3.docx
Anahtar: [79, 87, 88, 90, 72, 61, 124, 54, 130, 141, 124, 186, 56, 69, 223, 15, 41, 120, 34, 14, 89, 158, 1
77, 150, 160, 139, 104, 28, 231, 159, 99, 62]
**Ozetleme,sifreleme ve dosyaya yazma islemi:
Ozetlenmis metin: [81, 212, 16, 1, 66, 53, 226, 27, 62, 222, 163, 91, 147, 83, 71, 175]
Sifrelenmis metin: 194 222 225 142 173 215 193 8 140 237 99 159 87 30 211 211 124 203 228 179 201 236 11 25
0 103 78 62 224 106 249 14 9 59 12 11 195 34 238 0 135 24 146 242 226 132 66 33 120 124 23 96 190 225 116 5
5 126 207 98 152 167 101 252 9 218 238 91 181 156 37 76 242 16 26 92 122 90 43 58 44 209
**Dosya degistirilmedi.**
**Kontrol islemi**:
Ana metinden olusturulmus ozet metin: 81 212 16 1 66 53 226 27 62 222 163 91 147 83 71 175
Desifre edilmis ozet metin: 81 212 16 1 66 53 226 27 62 222 163 91 147 83 71 175
Degistirildi mi?: False
```

```
Dosya ismi: sample3.docx
Anahtar: [193, 182, 180, 3, 153, 130, 165, 26, 85, 215, 130, 82, 183, 19, 12, 132, 146, 226, 52, 201, 215,
234, 48, 83, 164, 169, 27, 135, 101, 249, 238, 203]
**Ozetleme,sifreleme ve dosyaya yazma islemi:
Ozetlenmis metin: [99, 240, 67, 74, 191, 194, 40, 254, 100, 43, 206, 142, 3, 210, 215, 114]
Sifrelenmis metin: 237 106 234 48 85 214 179 64 21 70 153 52 10 252 0 157 38 254 143 1 238 246 6 236 134 10
8 119 175 114 170 241 217 135 80 94 78 183 15 190 67 76 18 33 211 203 103 196 194 2 139 11 181 209 244 179
113 48 40 128 96 61 103 124 44 240 215 191 81 12 153 222 42 205 102 188 64 29 143 177 198
**Dosya degistirildi.
**Kontrol islemi:
Ana metinden olusturulmus ozet metin: 108 239 60 34 146 255 46 217 33 210 165 70 159 65 84 168
Desifre edilmis ozet metin: 99 240 67 74 191 194 40 254 100 43 206 142 3 210 215 114
Degistirildi mi?: True
```

- PDF dosyası için bütünlük testleri

```
Dosya ismi: sample2.pdf
Anahtar: [14, 0, 123, 216, 97, 117, 178, 200, 230, 42, 189, 222, 73, 117, 74, 128, 130, 243, 95, 34, 79, 11
9, 230, 19, 150, 16, 74, 19, 140, 209, 205, 122]
**Ozetleme,sifreleme ve dosyaya yazma islemi:
Ozetlenmis metin: [232, 164, 4, 61, 88, 146, 22, 76, 31, 215, 153, 159, 226, 10, 94, 251]
Sifrelenmis metin: 141 26 102 30 24 139 31 101 182 144 76 98 212 126 17 72 246 150 162 175 29 161 95 230 14
6 164 28 45 211 77 251 3 141 30 139 222 90 82 254 17 81 155 141 55 190 50 102 218 208 70 217 88 205 140 145
 89 34 64 34 160 17 156 81 107 147 196 142 176 111 131 17 57 152 107 101 152 43 230 96 141
**Dosya degistirilmedi.**
**Kontrol islemi**:
Ana metinden olusturulmus ozet metin: 232 164 4 61 88 146 22 76 31 215 153 159 226 10 94 251
Desifre edilmis ozet metin: 232 164 4 61 88 146 22 76 31 215 153 159 226 10 94 251
Degistirildi mi?: False
```

```
Dosya ismi: sample2.pdf
Anahtar: [78, 164, 11, 220, 140, 189, 81, 169, 22, 159, 121, 230, 198, 133, 169, 231, 99, 107, 48, 132, 15,
 68, 111, 252, 84, 144, 16, 236, 69, 83, 233, 116]
**Ozetleme,sifreleme ve dosyaya yazma islemi:
Ozetlenmis metin: [226, 160, 228, 202, 188, 175, 50, 127, 46, 238, 183, 184, 214, 58, 99, 209]
Sifrelenmis metin: 230 194 78 45 199 39 178 143 31 208 41 212 52 79 31 212 38 180 45 250 3 118 138 224 162
228 147 68 15 141 215 214 78 144 96 82 136 205 25 171 169 174 12 232 101 187 46 40 254 187 181 58 164 113 2
47 15 196 190 35 21 59 237 19 12 81 226 168 117 234 100 219 64 243 33 185 137 36 57 95 76
**Dosya degistirildi.
**Kontrol islemi:
Ana metinden olusturulmus ozet metin: 218 186 84 22 177 61 63 65 96 18 9 155 223 200 80 30
Desifre edilmis ozet metin: 226 160 228 202 188 175 50 127 46 238 183 184 214 58 99 209
Degistirildi mi?: True
```

- XLSX dosyası için bütünlük testleri

```
Dosya ismi: sample4.xlsx
Anahtar: [152, 253, 102, 89, 254, 157, 234, 243, 124, 140, 189, 244, 241, 161, 155, 130, 132, 127, 62, 218,
 131, 212, 36, 57, 36, 4, 81, 196, 206, 138, 241, 87]
**Ozetleme,sifreleme ve dosyaya yazma islemi:
Ozetlenmis metin: [164, 50, 100, 165, 212, 241, 157, 146, 7, 186, 46, 31, 9, 61, 22, 27]
Sifrelenmis metin: 149 236 215 73 110 150 27 54 87 23 117 46 248 223 150 131 88 142 87 225 231 18 30 150 38
 174 13 62 240 175 160 226 94 250 184 55 128 106 231 243 183 162 166 81 34 161 38 173 77 213 105 225 107 21
5 96 56 117 23 223 118 91 206 158 136 78 78 139 74 16 109 99 203 241 217 139 99 142 71 137 62
**Dosya degistirildi.
**Kontrol islemi:
Ana metinden olusturulmus ozet metin: 29 5 51 3 27 163 117 105 167 210 255 156 154 73 172 63
Desifre edilmis ozet metin: 164 50 100 165 212 241 157 146 7 186 46 31 9 61 22 27
Degistirildi mi?: True
fatihselimyakar@Fatihs-MacBook-Pro proje %
```

```
Dosya ismi: sample4.xlsx
Anahtar: [76, 173, 232, 233, 246, 153, 62, 176, 24, 62, 10, 167, 158, 56, 125, 203, 208, 255, 124, 154, 5,
231, 186, 215, 26, 67, 233, 227, 233, 120, 157, 95]
**Ozetleme,sifreleme ve dosyaya yazma islemi:
Ozetlenmis metin: [19, 5, 56, 0, 19, 162, 104, 120, 161, 206, 229, 131, 133, 74, 173, 2]
Sifrelenmis metin: 40 54 146 137 173 82 10 236 25 228 223 166 44 13 153 227 194 80 89 194 220 32 254 61 151
 126 227 107 13 107 108 161 186 147 30 110 161 7 98 152 111 147 127 33 5 161 102 59 250 210 135 135 62 69 1
17 232 62 113 33 168 22 147 26 229 123 222 245 160 140 33 127 237 81 255 63 163 211 201 253 209
**Dosya degistirilmedi.**
**Kontrol islemi**:
Ana metinden olusturulmus ozet metin: 19 5 56 0 19 162 104 120 161 206 229 131 133 74 173 2
Desifre edilmis ozet metin: 19 5 56 0 19 162 104 120 161 206 229 131 133 74 173 2
Degistirildi mi?: False
fatihselimyakar@Fatihs-MacBook-Pro proje %
```

- PPTX dosyası için bütünlük testleri

```
Dosya ismi: sample5.pptx
Anahtar: [175, 222, 90, 56, 215, 155, 253, 247, 13, 178, 195, 119, 112, 4, 122, 68, 50, 112, 19, 122, 25, 1
50, 28, 143, 76, 129, 97, 110, 132, 237, 187, 121]
**Ozetleme,sifreleme ve dosyaya yazma islemi:
Ozetlenmis metin: [98, 228, 138, 54, 58, 17, 16, 14, 2, 227, 225, 82, 137, 151, 230, 107]
Sifrelenmis metin: 53 1 73 175 118 49 85 5 242 80 174 245 106 0 219 34 249 95 156 208 155 215 201 76 116 16
0 113 84 77 91 40 197 249 224 144 100 15 148 117 123 18 16 156 20 115 185 82 192 219 53 128 63 62 229 36 20
7 30 17 97 6 17 154 79 152 253 37 188 79 173 137 147 149 33 204 35 234 66 145 136 195
**Dosya degistirilmedi.**
**Kontrol islemi**:
Ana metinden olusturulmus ozet metin: 98 228 138 54 58 17 16 14 2 227 225 82 137 151 230 107
Desifre edilmis ozet metin: 98 228 138 54 58 17 16 14 2 227 225 82 137 151 230 107
Degistirildi mi?: False
fatihselimyakar@Fatihs-MacBook-Pro proje %
```

```
Dosya ismi: sample5.pptx
Anahtar: [212, 125, 207, 117, 215, 49, 103, 120, 2, 191, 53, 25, 0, 124, 222, 43, 56, 221, 177, 245, 112, 9
4, 39, 185, 238, 223, 217, 175, 152, 179, 91, 187]
**Ozetleme,sifreleme ve dosyaya yazma islemi:
Ozetlenmis metin: [93, 204, 188, 29, 53, 5, 19, 24, 29, 247, 242, 95, 185, 164, 197, 64]
Sifrelenmis metin: 240 110 234 154 197 231 143 99 124 20 189 51 167 112 240 53 2 138 54 5 131 143 100 178 1
26 71 22 196 88 196 166 162 115 208 9 186 249 166 73 20 22 70 131 115 163 21 4 238 109 193 212 109 11 126 1
55 144 142 101 188 235 25 148 244 85 89 11 157 243 62 88 123 87 112 186 3 0 148 16 22 105
**Dosya degistirildi.
**Kontrol islemi:
Ana metinden olusturulmus ozet metin: 93 85 74 48 160 200 25 47 129 161 153 132 134 193 115 109
Desifre edilmis ozet metin: 93 204 188 29 53 5 19 24 29 247 242 95 185 164 197 64
Degistirildi mi?: True
fatihselimyakar@Fatihs-MacBook-Pro proje %
```