

**UNIVERSIDAD PRIVADA FRANZ TAMAYO**

**FACULTAD DE INGENIERIA**

**CARRERA DE INGENIERIA DE SISTEMAS**



**ALGORITMO DE ENCRIPCIÓN DE ARCHIVOS CON PYTHON**

**Estudiante:**

Cardenas Fernandez Daniela Jhoselyn

**Materia:**

Seguridad Informática

**Docente:**

Ing. Henry Raul Vargas Grillo

**LA PAZ – BOLIVIA**

**2021**

## Contenido

<b>1.</b>	<b>Introducción .....</b>	<b>5</b>
<b>1.1.</b>	<b>Presentación del problema .....</b>	<b>5</b>
<b>1.2.</b>	<b>Argumentación de la importancia de objeto del estudio ....</b>	<b>6</b>
<b>CAPÍTULO II    DESARROLLO .....</b>		<b>7</b>
<b>2.</b>	<b>Marco teórico .....</b>	<b>7</b>
<b>2.1.</b>	<b>Criptografía.....</b>	<b>7</b>
<b>2.2.</b>	<b>Criptografía simétrica .....</b>	<b>7</b>
<b>2.3.</b>	<b>Criptografía asimétrica .....</b>	<b>7</b>
<b>2.4.</b>	<b>Criptografía como elemento de Seguridad .....</b>	<b>8</b>
<b>CAPÍTULO III    JUSTIFICACIONES.....</b>		<b>10</b>
<b>3.1.</b>	<b>Justificación técnica .....</b>	<b>10</b>
<b>3.2.</b>	<b>Justificación económica.....</b>	<b>10</b>
<b>3.3.</b>	<b>Justificación social .....</b>	<b>10</b>
<b>CAPÍTULO IV    DISEÑO TEÓRICO DE LA INVESTIGACIÓN.....</b>		<b>11</b>
<b>4.1.</b>	<b>Planteamiento del problema.....</b>	<b>11</b>
<b>4.2.</b>	<b>Formulación del problema.....</b>	<b>11</b>
<b>4.3.</b>	<b>Delimitación del contenido .....</b>	<b>11</b>
<b>4.3.1.</b>	<b>Alcances .....</b>	<b>11</b>
<b>4.3.2.</b>	<b>Límites.....</b>	<b>11</b>
<b>4.4.</b>	<b>Delimitación temporal .....</b>	<b>11</b>

4.5. Delimitación espacial .....	12
4.6. Objetivos .....	12
4.6.1. Objetivo General .....	12
4.6.2. Objetivos Específicos .....	12
<b>CAPÍTULO 5 MARCO PRÁCTICO .....</b>	<b>13</b>
5.1. Librerías utilizadas .....	13
5.2. Código Fuente .....	13
5.2.1. Código fuente Librerías.py .....	13
5.2.2. Código fuente encriptación.py .....	15
5.3. Ejecución .....	18
6. Referencias .....	20
7. Anexos .....	21

Ilustración 1 "Ejecución de leer archivo" .....	18
Ilustración 2 Ejecución "Agregar texto" .....	18
Ilustración 3 Ejecución "leer archivo, después de agregar texto" .....	18
Ilustración 4 Ejecución "Encriptar un solo archivo" .....	19
Ilustración 5 Ejecución "Desencriptar un solo archivo" .....	19
Ilustración 6 Ejecución "Encriptar todos los archivos" .....	19
Ilustración 7 Ejecución "Desencriptar todos los archivos" .....	20
Ilustración 8 Página para descargar Python .....	21
Ilustración 9 Instalación de la librería Fernet .....	21
Ilustración 10 Ejecución de código .....	22
Ilustración 11 Selección de acciones.....	22
Ilustración 12 Agregar texto desde el algoritmo.....	23
Ilustración 13 Agregar texto fuera del algoritmo .....	23

## **1. Introducción**

Gracias a la evolución de la tecnología, hoy en día contamos con diferentes tipos de dispositivos, a los que se les fue asignando más y más cosas, entre sensores, cámaras, micrófonos, etc. Gracias a esto, se fue vulnerando la seguridad de los usuarios de estos dispositivos, gracias a los bajos niveles de seguridad que existen en algunos dispositivos, es sencillo poder acceder a la información y archivos de los distintos equipos.

Existen medidas de seguridad para poder proteger los equipos de estos ataques, pero debido a que las personas no están conscientes de lo que se puede realizar con un dispositivo, no tienen las precauciones necesarias para poder manejarlo.

En este caso, existen softwares legales que ayudan a hackear equipos, los cuales una vez instalados en el ordenador son imposibles de eliminar, con esta se pueden monitorear todas las actividades que son realizadas en el ordenador, de esta forma se puede robar la información como archivos y fotos sin que el usuario lo sepa, violando la privacidad de los usuarios.

### **1.1. Presentación del problema**

La privacidad es un ámbito de la vida profesional de un individuo, la cual es desarrollada en un espacio reservado, es algo que debe mantenerse de forma confidencial, actualmente existen muchas herramientas que vulneran los derechos y la privacidad de los seres humanos, dándoles así la susceptibilidad de ser hackeados o de que vulneren su privacidad.

Algunos de los usuarios, no tienen mucho conocimiento tecnológico, por lo cual no se encuentran conscientes de la facilidad con la que pueden acceder a sus datos personales, como archivos, imágenes además de diferentes tipos de accesos a sus equipos y brindan de una forma “poco consciente” el acceso a estos dentro de sus dispositivos.

Debido a lo anteriormente expuesto, nos lleva a la siguiente pregunta:

¿Cómo demostrar y concientizar sobre la necesidad de mantener los archivos encriptados de forma simétrica?

## **1.2. Argumentación de la importancia de objeto del estudio**

La importancia del objeto de estudio del proyecto es poder lograr la concientización de las personas, para que puedan tener más niveles de seguridad y protección en sus dispositivos y que así estos tengan dispositivos mucho más seguros y no acepten cualquier tipo de permisos que se requiera en aplicaciones, juegos y otras cosas que sean instaladas dentro de un dispositivo móvil o una computadora.

Con este proyecto, podremos enseñar a las personas a cuidar un poco la información confidencial de sus archivos, gracias a una encriptación, la cual les ayudará a mantener una información segura, debido a que el cifrado que se utilizará no podrá ser descifrado sin la respectiva llave o código.

## **CAPÍTULO II**

## **DESARROLLO**

### **2. Marco teórico**

#### **2.1. Criptografía**

Los códigos secretos han sido utilizados desde la antigüedad para enviar mensajes seguros, por ejemplo, en tiempo de guerra o de tensiones diplomáticas. Muchos códigos están basados en teoría de números. Estos códigos son fáciles de romper pues se pueden probar todos los posibles valores hasta obtener un mensaje comprensible, o podemos comparar las letras más frecuentes en el mensaje con las que se saben que son más frecuentes en la lengua original. Históricamente primero han aparecido los códigos basados en una única clave conocidos como sistemas de clave privada.

Estos tienen el inconveniente de que el remitente y el receptor deben estar de acuerdo, con anterioridad, en los valores de la clave. Teniendo en cuenta que por seguridad deberán cambiar la clave de vez en cuando, ¿de qué modo podrían hacerlo de una manera segura? (Lomparte, 2011)

#### **2.2. Criptografía simétrica**

La criptografía simétrica utiliza la misma clave para cifrar y descifrar el mensaje de datos, es decir se basa en un secreto compartido. Es por esta razón que la seguridad de este proceso depende de la posibilidad de que una persona no autorizada consiga la clave de sesión o clave secreta. (Medoza, 2008)

Lo cual nos quiere decir que este tipo de criptografía es una de las más seguras, debido a que no se puede descifrar la información de este encriptado, sin la llave o clave generada.

#### **2.3. Criptografía asimétrica**

El cifrado asimétrico también es conocido como cifrado de clave pública o cifrado de dos claves. En este tipo de cifrado se utiliza una clave para el cifrado y otra para el descifrado. Ambas claves están relacionadas, pero de una forma demasiado compleja como para encontrar una a partir de la otra si no se conoce

concretamente la relación. El cifrado se realiza con lo que se llama clave pública y el descifrado con la clave privada. (Monreal, 2019)

Gracias a la anterior definición, podemos deducir que el cifrado simétrico y asimétrico son dos formas seguras de poder realizar el descifrado de archivos, claves, palabras y contraseñas, a diferencia del cifrado simétrico, el cifrado asimétrico utiliza dos claves distintas, una para poder cifrar los datos y otra para poder descifrarlos.

#### **2.4. Criptografía como elemento de Seguridad**

Entre las propiedades más importantes de los documentos electrónicos se encuentran la confidencialidad y la autenticidad.

La primera se refiere a la posibilidad de mantener un documento electrónico inaccesible a todos, excepto a una lista de individuos autorizados. La autenticidad, por su parte, es la capacidad de determinar si uno o varios individuos han reconocido como suyo y se han comprometido con el contenido del documento electrónico. El problema de la autenticidad en un documento tradicional se soluciona mediante la firma autógrafa. Mediante su firma autógrafa, un individuo, o varios, manifiestan su voluntad de reconocer el contenido de un documento, y en su caso, a cumplir con los compromisos que el documento establezca para con el individuo.

Los problemas relacionados con la confidencialidad, integridad y autenticidad en un documento electrónico se resuelven mediante la tecnología llamada Criptografía.

Debido a su importancia entre los aspectos tratados actualmente en materia de seguridad informática, los autores de este trabajo se propusieron realizar una aproximación teórico - práctica al tema con el objetivo de



comprender sus principales aristas y determinar algunas de las tendencias fundamentales que caracterizan el desarrollo de esta disciplina. (Travieso, 2003)

Como se pudo apreciar en el párrafo anterior, la criptografía ayuda a que se puedan mantener seguros archivos, imágenes y documentos, gracias a esto podemos tener presentes los principales fundamentos de la seguridad informática, la confidencialidad y la integridad de los datos, además de contar también con el acceso a estos cuando es necesario.

## **CAPÍTULO III**

## **JUSTIFICACIONES**

### **3.1. Justificación técnica**

Para la elaboración de dicho proyecto, se utilizará como base principal el lenguaje de programación Python, el cual nos ayudará a crear el algoritmo para poder encriptar y desencriptar los diferentes archivos.

### **3.2. Justificación económica**

Los recursos para utilizarse para dicho proyecto no son elevados, debido a que solo necesitamos tener instalado Python, un editor de código como Visual Studio code y una computadora para poder realizar la programación del algoritmo, en este caso, debido a que las herramientas de software pueden ser encontradas de forma gratuita en la web, el costo es mínimo, además de ayudar a las personas a ocultar información secreta de sus archivos, como ser estados de cuentas, socios y proveedores.

### **3.3. Justificación social**

El presente proyecto, ayudará a las personas y empresarios a poder tener seguros sus archivos, gracias a que sus archivos se encontrarán cifrados con un cifrado simétrico, el cual no puede ser descifrado si no se tiene la llave generada para poder descifrar dicho archivo.

Gracias a esto, se podrá mantener la confidencialidad de los archivos, además de ayudar a que las personas tomen en cuenta la importancia del resguardo correcto de su información.

## **CAPÍTULO IV DISEÑO TEÓRICO DE LA INVESTIGACIÓN**

### **4.1. Planteamiento del problema**

La seguridad es vital para las personas de todas partes del mundo, además de la seguridad, es necesario concientizar a las personas sobre los permisos que otorgan sobre el acceso a sus dispositivos, dado este caso, gracias al gran incremento de cibercriminales, es necesario mantener niveles de seguridad dentro de los equipos, debido a que hoy en día existen herramientas que logran acceder al dispositivo, logrando robar información.

Hoy en día es necesario el tener un método para poder asegurar la información dentro de los archivos, debido a que muchas veces existen archivos que deben mantenerse de forma privada, como ser: Estados de cuenta, socios y principales proveedores dentro de la empresa, ya que la competencia hoy en día, puede buscar el robo de la información para perjudicar a las empresas gracias al fácil acceso y poca seguridad de este tipo de archivos.

### **4.2. Formulación del problema**

¿Cómo mantener seguro el contenido de los archivos importantes de un equipo?

### **4.3. Delimitación del contenido**

#### **4.3.1. Alcances**

El siguiente proyecto, llevará a cabo la encriptación y desencriptación de un solo archivo y de varios archivos al mismo tiempo, utilizando un cifrado simétrico con la librería Fernet de Python.

#### **4.3.2. Límites**

El presente proyecto no tendrá una aplicación de escritorio, debiendo ser ejecutada desde la terminal de Python, además de que no se podrán agregar más archivos externos que no sean de tipo .txt

### **4.4. Delimitación temporal**

El presente proyecto se encuentra trabajando en la gestión 2/2021, por lo cual se trabajará en un lapso de 3 meses, teniendo como fecha de finalización el 30 de noviembre del 2021.

#### **4.5. Delimitación espacial**

El algoritmo de encriptación y desencriptación con Python, será realizado en el departamento de La Paz, con el fin de cubrir un sector demográfico más delimitado y hacer las pruebas respectivas antes de ponerlo en marcha.

#### **4.6. Objetivos**

##### **4.6.1. Objetivo General**

Realizar un algoritmo que permita encriptar y desencriptar archivos como documentos con una llave única, utilizando un cifrado simétrico.

##### **4.6.2. Objetivos Específicos**

- Aprender a utilizar Python para la encriptación y desencriptación de archivos con la librería Fernet.
- Demostrar a los usuarios lo sencillo que es tener la información de sus documentos seguros gracias a la encriptación.
- Realizar el proyecto, desde una parte analítica, dando a conocer al usuario la importancia de la encriptación de archivos.

## CAPÍTULO 5

## MARCO PRÁCTICO

### 5.1. Librerías utilizadas

#### 5.1.1. Librería fernet

El módulo fernet del paquete de criptografía tiene funciones incorporadas para la generación de la clave, el cifrado de texto plano en texto cifrado y el descifrado de texto cifrado en texto plano utilizando los métodos de cifrado y descifrado respectivamente. El módulo fernet garantiza que los datos cifrados con él no se puedan manipular ni leer sin la clave. (Chackraborti, 2021)

#### 5.1.2. Librería Path

Este módulo ofrece clases que representan rutas del sistema de archivos con semántica apropiada para diferentes sistemas operativos.

### 5.2. Código Fuente

#### 5.2.1. Código fuente Librerias.py

```
from cryptography import fernet
from cryptography.fernet import Fernet
from pathlib import Path #archivos y carpetas

def leerArchivo(archivos):
    stream = open(archivos, "rt", encoding="utf-8")
    print(stream.read())
def leerArchivo(hola):
    stream = open(hola, "rt",encoding="utf-8")
    print(stream.read() )
def leerArchivo(pruebas):
    stream = open(pruebas, "rt",encoding="utf-8")
    print(stream.read())
def escribirArchivo(linea, archivos):
    with open(archivos, 'a') as file:
        file.write("\n"+linea)
def escribirArchivo(linea, hola):
    with open(hola, 'a') as file:
        file.write("\n"+linea)
def escribirArchivo(linea, pruebas):
    with open(pruebas, 'a') as file:
        file.write("\n"+linea)
def generar_clave():
    archivos = r'llave.llave'
    objetoArchivo= Path(archivos)
    if not objetoArchivo.is_file():
```

```

        clave = Fernet.generate_key() #genero la clave

        with open("llave.llave","wb") as key_file:
            key_file.write(clave)

def cargar_clave():
    return open("llave.llave","rb").read()

def encriptar(archivos,clave):
    f = Fernet(clave)
    with open(archivos, "rb") as file:
        file_data = file.read()

    datos_encriptados= f.encrypt(file_data)

    with open(archivos, "wb") as file:
        file.write(datos_encriptados)
def encriptar(hola,clave):
    f = Fernet(clave)
    with open(hola, "rb") as file:
        file_data = file.read()

    datos_encriptados= f.encrypt(file_data)

    with open(hola, "wb") as file:
        file.write(datos_encriptados)
def encriptar(pruebas,clave):
    f = Fernet(clave)
    with open(pruebas, "rb") as file:
        file_data = file.read()

    datos_encriptados= f.encrypt(file_data)

    with open(pruebas, "wb") as file:
        file.write(datos_encriptados)

def desencriptar(archivos,clave):
    f= Fernet(clave)
    with open(archivos, "rb") as file:
        datos_encriptados = file.read()

    datos = f.decrypt(datos_encriptados)

    with open(archivos,"wb") as file:
        file.write(datos)
def desencriptar(hola,clave):
    f= Fernet(clave)
    with open(hola, "rb") as file:
        datos_encriptados = file.read()

```

```
datos = f.decrypt(datos_encryptados)

with open(hola,"wb") as file:
    file.write(datos)
def desencriptar(pruebas,clave):
    f= Fernet(clave)
    with open(pruebas, "rb") as file:
        datos_encryptados = file.read()

    datos = f.decrypt(datos_encryptados)

    with open(pruebas,"wb") as file:
        file.write(datos)
```

### 5.2.2. Código fuente encriptación.py

```
import librerias

opcion = 0

librerias.generar_clave()
clave=librerias.cargar_clave()

archivos = "archivo.txt"
hola = "hola.txt"
pruebas = "prueba.txt"

while opcion !=5:
    print("\nSelecciona una opción")
    print("1.Leer Archivo\n2.Agregar Texto al Archivo\n3.Encriptar\n4.Desencriptar\n5.Salir")
    opcion = int(input ("Ingresa una opción: "))
    if opcion ==1:
        while opcion !=4:
            print("\n¿Qué archivo quieres ver?")
            print("1.Archivo\n2.Hola\n3.Pruebas\n4.Atrás")
            opcion = int(input ("Ingresa una opción: "))
            if opcion ==1:
                librerias.leerArchivo(archivos)
            elif opcion ==2:
                librerias.leerArchivo(hola)
            elif opcion == 3:
                librerias.leerArchivo(pruebas)
            else:
                print("\n Atrás")
        elif opcion == 2:
            while opcion != 4:
                print("\n¿Qué archivo deseas?")
```

```

print("1.Archivo\n2.Hola\n3.Pruebas\n4.Atrás")
opcion = int(input ("Ingresa una opción: "))
if opcion ==1:
    linea = input("Texto a agregar:")
    librerias.escribirArchivo(linea, archivos)
elif opcion ==2:
    linea = input("Texto a agregar:")
    librerias.escribirArchivo(linea, hola)
elif opcion == 3:
    linea = input("Texto a agregar:")
    librerias.escribirArchivo(linea, pruebas)
else:
    print("\n Atrás")

elif opcion == 3:
    while opcion != 4:
        print("\n¿Qué quieres hacer?")
        print("1.Encriptar un archivo\n2.Encriptar todos los
archivos\n3.Atrás")
        opcion = int(input ("Ingresa una opción: "))
        if opcion ==1:
            while opcion != 4:
                print("\n¿Qué archivo deseas?")
                print("1.Archivo\n2.Hola\n3.Pruebas\n4.Atrás")
                opcion = int(input ("Ingresa una opción: "))
                if opcion ==1:
                    librerias.encryptar(archivos,clave)
                    print("Archivo encriptado, el texto es: ")
                    librerias.leerArchivo(archivos)
                elif opcion ==2:
                    librerias.encryptar(hola,clave)
                    print("Archivo encriptado, el texto es: ")
                    librerias.leerArchivo(hola)
                elif opcion == 3:
                    librerias.encryptar(pruebas,clave)
                    print("Archivo encriptado, el texto es: ")
                    librerias.leerArchivo(pruebas)
                else:
                    print("\n Atrás")
            elif opcion ==2:
                librerias.encryptar(archivos,clave)
                librerias.encryptar(hola,clave)
                librerias.encryptar(pruebas,clave)
                print("Archivos encriptados")
                print("\nNombre del archivo: archivo")
                librerias.leerArchivo(archivos)
                print("\nNombre del archivo: hola")
                librerias.leerArchivo(hola)
                print("\nNombre del archivo: prueba")

```



```
        librerias.leerArchivo(pruebas)
    else:
        print("\nAtrás")

elif opcion == 4:
    print("\n¿Qué quieres hacer?")
    print("1.Desencriptar un archivo\n2.Desencriptar todos los
archivos\n3.Atrás")
    opcion = int(input ("Ingresa una opción: "))
    if opcion ==1:
        while opcion != 4:
            print("\n¿Qué archivo deseas?")
            print("1.Archivo\n2.Hola\n3.Pruebas\n4.Atrás")
            opcion = int(input ("Ingresa una opción: "))
            if opcion ==1:
                librerias.desencriptar(archivos,clave)
                print("archivo desencriptado, el texto es: ")
                librerias.leerArchivo(archivos)
            elif opcion ==2:
                librerias.desencriptar(hola,clave)
                print("archivo desencriptado, el texto es: ")
                librerias.leerArchivo(hola)
            elif opcion == 3:
                librerias.desencriptar(pruebas,clave)
                print("archivo desencriptado, el texto es: ")
                librerias.leerArchivo(pruebas)
            else:
                print("\n Atrás")
        elif opcion ==2:
            librerias.desencriptar(archivos,clave)
            librerias.desencriptar(hola,clave)
            librerias.desencriptar(pruebas,clave)
            print("Archivos desencriptados")
            print("\nNombre del archivo: archivo, el texto es:")
            librerias.leerArchivo(archivos)
            print("\nNombre del archivo: hola, el texto es: ")
            librerias.leerArchivo(hola)
            print("\nNombre del archivo: prueba, el texto es: ")
            librerias.leerArchivo(pruebas)
        else:
            print("\nAtrás")

    else:
        print("\n¿Opción seleccionada Incorrecta!")
```

### 5.3. Ejecución

```
PS C:\Users\Jhossy\Documents\8vo Semestre\SIN\proyecto> & C:/Users/Jhossy/AppData/Local/Programs/Python/Python310/python.exe "c:/Users/Jhossy/Documents/8vo Semestre/SIN/proyecto/encryptacion.py"

Selecciona una opción
1.Leer Archivo
2.Agregar Texto al Archivo
3.Encriptar
4.Desencriptar
5.Salir
Ingresa una opción: 1

¿Qué archivo quieres ver?
1.Archivo
2.Hola
3.Pruebas
4.Atrás
Ingresa una opción: 1
Hola
hola hola
Esto es una prueba

¿Qué archivo quieres ver?
1.Archivo
2.Hola
3.Pruebas
4.Atrás
Ingresa una opción: █
```

*Ilustración 1 “Ejecución de leer archivo”*

```
Selecciona una opción
1.Leer Archivo
2.Agregar Texto al Archivo
3.Encriptar
4.Desencriptar
5.Salir
Ingresa una opción: 2

¿Qué archivo deseas?
1.Archivo
2.Hola
3.Pruebas
4.Atrás
Ingresa una opción: 1
Texto a agregar:Es una prueba antes de la presentación final
```

*Ilustración 2 Ejecución "Agregar texto"*

```
Selecciona una opción
1.Leer Archivo
2.Agregar Texto al Archivo
3.Encriptar
4.Desencriptar
5.Salir
Ingresa una opción: 1

¿Qué archivo quieres ver?
1.Archivo
2.Hola
3.Pruebas
4.Atrás
Ingresa una opción: 1
Hola
hola hola
Esto es una prueba
Es una prueba antes de la presentacion final
```

*Ilustración 3 Ejecución "leer archivo, después de agregar texto"*

```
10/python.exe "c:/Users/3hossy/Documents/8vo Semestre/SIN/proyecto/criptacion.py"

Selecciona una opción
1.Leer Archivo
2.Agregar Texto al Archivo
3.Encriptar
4.Desencriptar
5.Salir
Ingresa una opción: 3

¿Qué quieres hacer?
1.Encriptar un archivo
2.Encriptar todos los archivos
3.Atrás
Ingresa una opción: 1

¿Qué archivo deseas?
1.Archivo
2.Hola
3.Pruebas
4.Atrás
Ingresa una opción: 2
Archivo encriptado, el texto es:
gAAAAABhqb704013auB8gjUww08KCebbGrVSWxE9NGq6uFkAZEg53FuscABqIoJ_xvsh1VE8ea5gx_r7JrXe7iZjc8qVEeAkHdZ9gb6EFsW_o-4
LmcDuAVA=
```

*Ilustración 4 Ejecución "Encriptar un solo archivo"*

```
Selecciona una opción
1.Leer Archivo
2.Agregar Texto al Archivo
3.Encriptar
4.Desencriptar
5.Salir
Ingresa una opción: 4

¿Qué quieres hacer?
1.Desencriptar un archivo
2.Desencriptar todos los archivos
3.Atrás
Ingresa una opción: 1

¿Qué archivo deseas?
1.Archivo
2.Hola
3.Pruebas
4.Atrás
Ingresa una opción: 2
archivo desencriptado, el texto es:
holaaaaa
Hola como estas
```

*Ilustración 5 Ejecución "Desencriptar un solo archivo"*

```
Selecciona una opción
1.Leer Archivo
2.Agregar Texto al Archivo
3.Encriptar
4.Desencriptar
5.Salir
Ingresa una opción: 3

¿Qué quieres hacer?
1.Encriptar un archivo
2.Encriptar todos los archivos
3.Atrás
Ingresa una opción: 2
Archivos encriptados

Nombre del archivo: archivo
gAAAAABhqb8Vved6duFF5YpFSFiKXwP3nvUZ6m6oa-5POXceAV9qxqZBy-Y1UXJFpfZp3bVDF81xn0aTC0z-wwE7PITtuFH8xqTfOR22MU_30nT
akZuzz3yCfuU8asqFquNMbxQ_13E3WvGkxstJRY_2uvxue3aEiCiGnm7UAbzXaSKERY964h1MZr9zKrIrFqN6_u2LKmcI

Nombre del archivo: hola
gAAAAABhqb8VzVKT84Tec9sLs9X3eF4jeRZxG-yUQJTpZvOpd4kHZsdfa--CilFCRBoru0X2rhctzN5odKaMv9Acj1uXMzq_NHx117yR02HaU6o
x3y71tYo=

Nombre del archivo: prueba
gAAAAABhqb8VwKmxT-yVV2A0W_8r6P1G1Daz1n81L8qeN5kwDw8HMYTbmILcVL79kx4P10BKallZnxpv3-IRvPSxv7bmt1jucQ==
```

*Ilustración 6 Ejecución "Encriptar todos los archivos"*

```
3.Encriptar
4.Desencriptar
5.Salir
Ingresa una opción: 4

¿Qué quieres hacer?
1.Desencriptar un archivo
2.Desencriptar todos los archivos
3.Atrás
Ingresa una opción: 2
Archivos desencriptados

Nombre del archivo: archivo, el texto es:
Hola
hola hola
Esto es una prueba
Es una prueba antes de la presentacion final

Nombre del archivo: hola, el texto es:
holaaaaa
Hola como estas

Nombre del archivo: prueba, el texto es:
hola
```

*Ilustración 7 Ejecución "Desencriptar todos los archivos"*

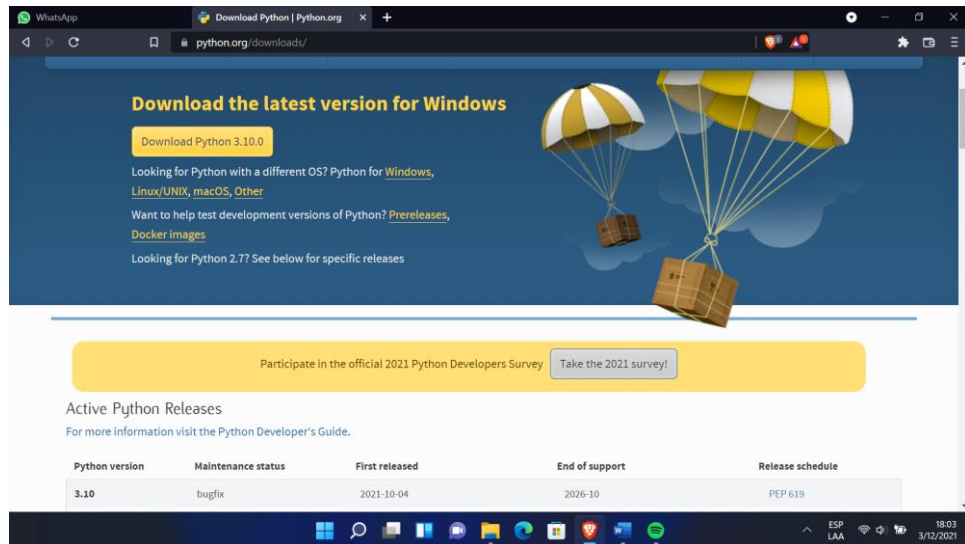
## 6. Referencias

- Chackraborti, S. (Febrero de 2021). *Acervo Lima*. Obtenido de Fernet (cifrado simetrico) usando el módulo de criptografia Python: <https://es.acervolima.com/2021/02/09/fernetcifrado-simetrico-usando-el-modulo-de-criptografia-en-python/>
- Lomparte, K. (2011). *Encriptación RSA de archivos de texto*. Perú: Pontificia Universidad Católica de Perú.
- Medoza, J. (2008). Demostración de Cifrado simetrico y asimetrico. *Ingenius: Revista de ciencia y tecnología*, 46-53.
- Monreal, C. B. (2019). *Criptografía de clave simétrica y de clave asimétrica*. Universitat Jaume I.
- Travieso, Y. M. (2003). *La Criptografía como elemento de la seguridad informática*. Acimed.

## 7. Anexos

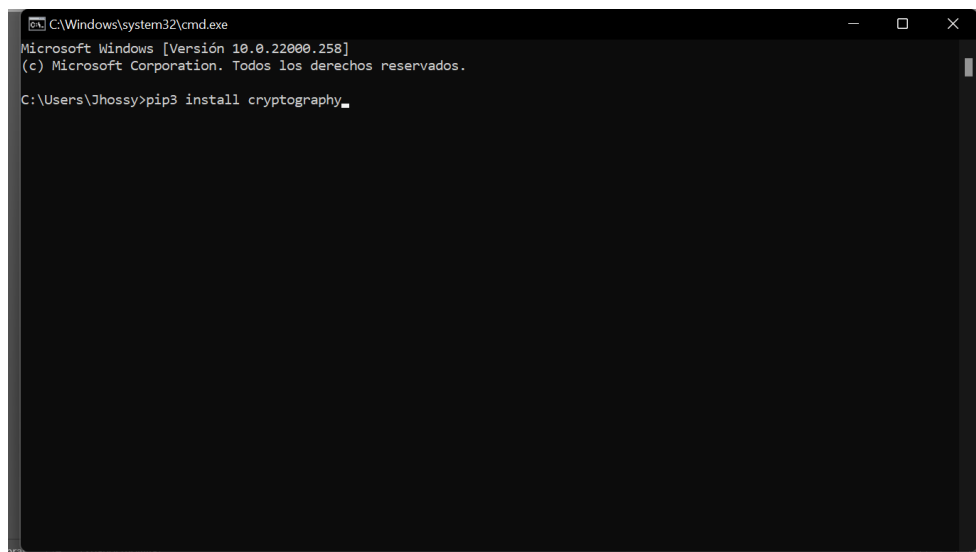
### MANUAL DE USUARIO

**Paso 1.** Como principal tener instalado Python para poder correr el archivo, este puede ser descargado desde su página oficial.



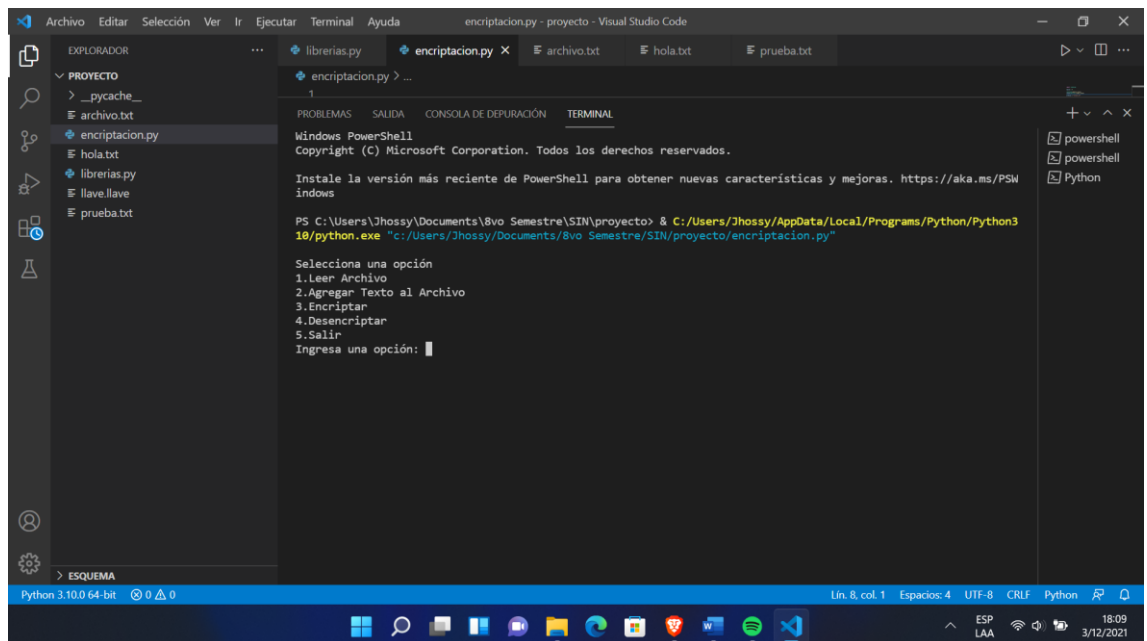
*Ilustración 8 Página para descargar Python*

**Paso 2.** Tener instalado la librería Fernet con el comando `pip3 install cryptography` desde el cmd Windows y `sudo pip3 install cryptography` desde la terminal Linux.



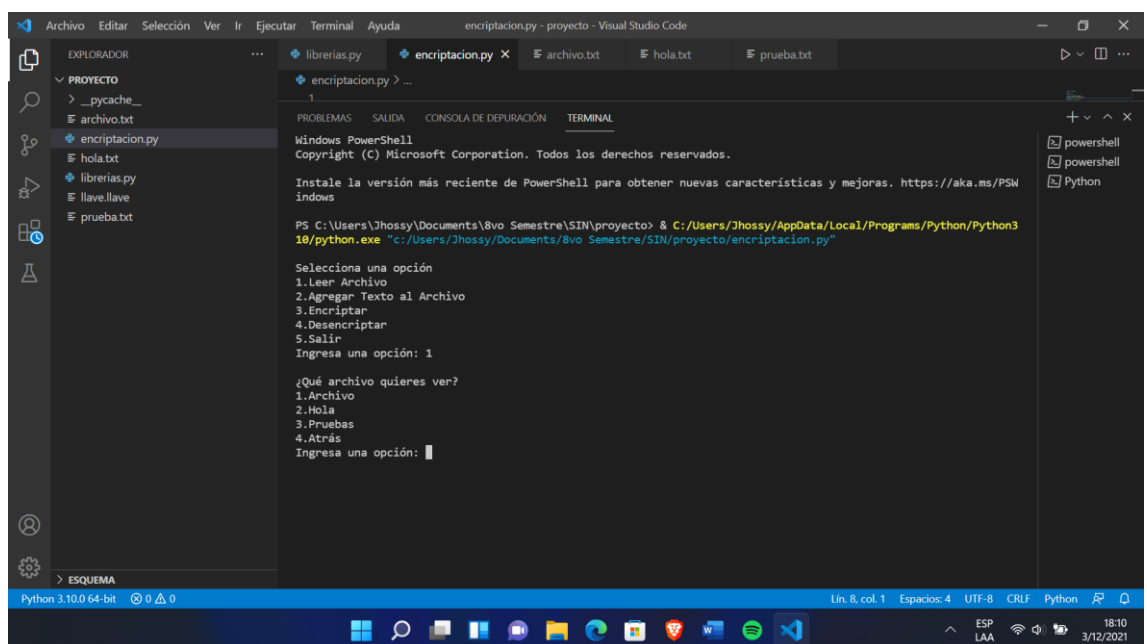
*Ilustración 9 Instalación de la librería Fernet*

**Paso 4.** Abrir el editor de tu preferencia o el editor predeterminado de Python y ejecuta el código.



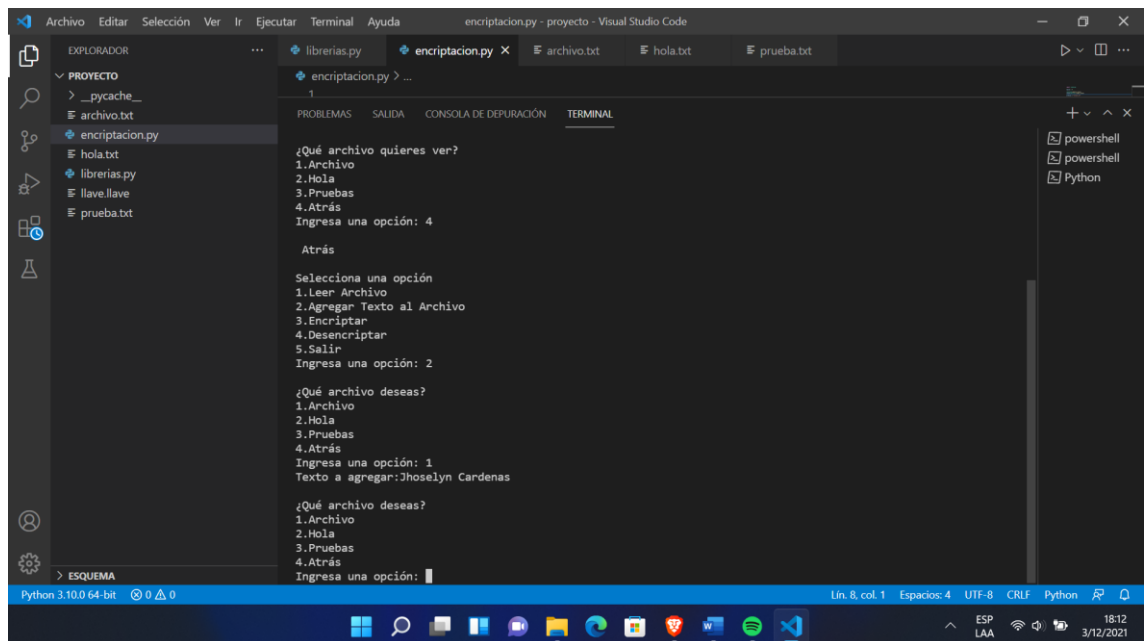
*Ilustración 10 Ejecución de código*

**Paso 5.** Selecciona cualquiera de las opciones que desees, dependiendo a lo que se quiere realizar.

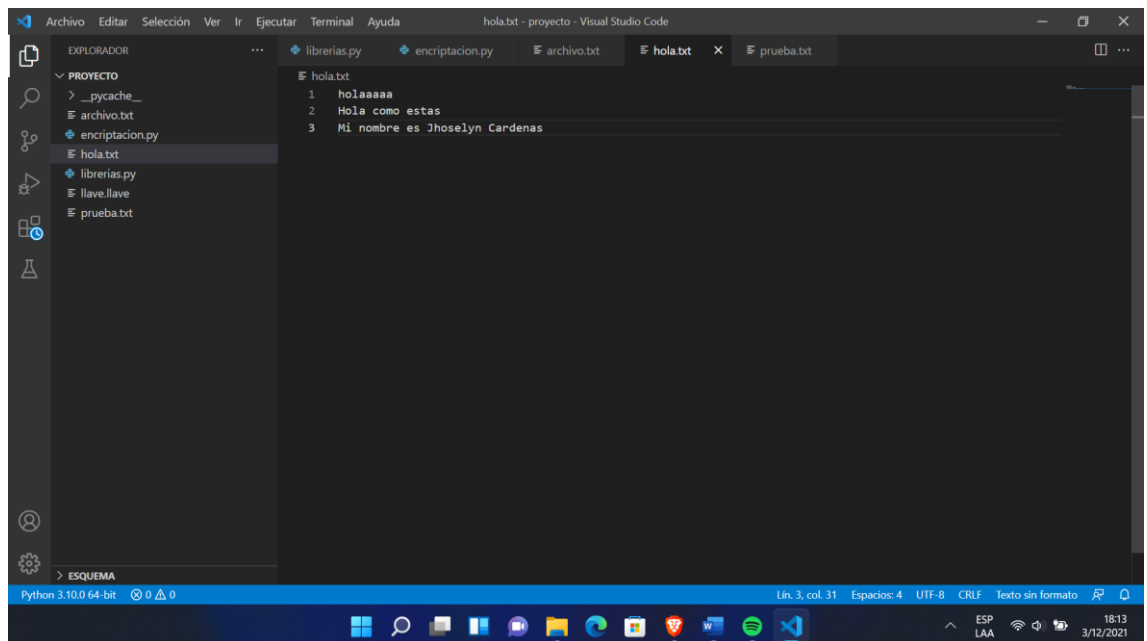


*Ilustración 11 Selección de acciones*

**NOTA:** El nuevo texto de los archivos puede ser agregado desde el algoritmo en ejecución o ser agregado a parte desde el editor de su preferencia.



*Ilustración 12 Agregar texto desde el algoritmo*



*Ilustración 13 Agregar texto fuera del algoritmo*