

Reporte semanal - Semana 5

23/02/2024

—

Ismael Montoro Peñasco
Fundación Goodlob

Acciones llevadas a cabo.....	2
Avances realizados.....	2
Labor 1: Re-estructuración del proyecto.....	2
Labor 2: Algoritmos de Hashing.....	3
Labor 3: Codificaciones.....	4
Labor 4: Encriptaciones.....	5
Labor 5: Scapy.....	5
Labor 6: Ejercicios de la semana.....	6
Labor 7: Excel.....	8
Planes para la próxima semana.....	9

Acciones llevadas a cabo

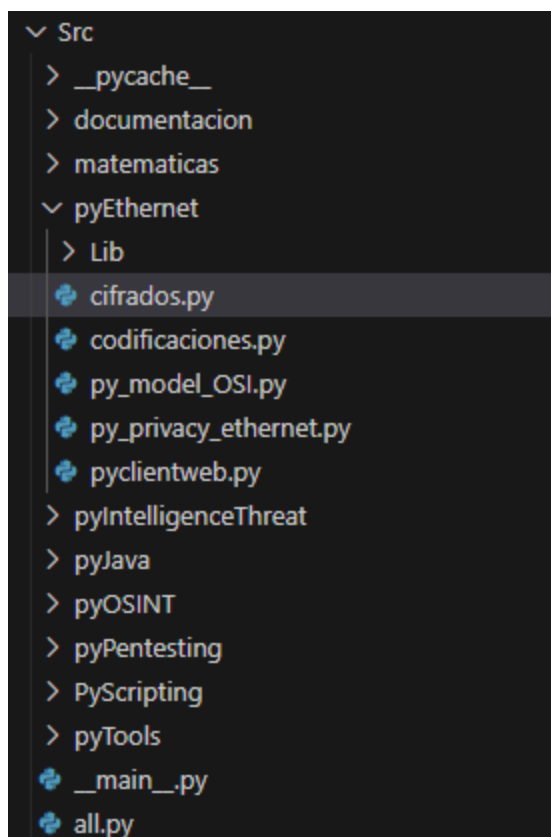
Empezaré a investigar librerías de cifrado, codificaciones, redes (Scapy) y hacking a través de mi proyecto “espadas”, además de realizar ejercicios de la semana. Mi prioridad para seguir aprendiendo será mi proyecto espadas, además a partir de ahora iré haciendo los ejercicios de la semana dentro del proyecto de espadas. También seguiré investigando sobre Excel.

Avances realizados

Labor 1: Re-estructuración del proyecto

He realizado una labor de estructuración en el proyecto. Clasificando las librerías como librerías globales y librerías locales.

Las librerías globales, son aquellas librerías que se van a utilizar en varias partes del proyecto, las librerías locales, son aquellas librerías que se van a utilizar en un script concreto del proyecto, todo con el fin de evitar importar algunas librerías base del proyecto de manera repetida y mejorar así la gestión de los recursos.



Labor 2: Algoritmos de Hashing

He trabajado con nuevos tipos de algoritmos de Hash:

- MD6: La nueva versión de los hashes MD (MD2, MD4, MD5).
- GOST: Algoritmo hash creado por la Federación Rusa.
- crc32b: Algoritmos hash de la serie CRC.

```
def md6(self, texto = "hello", size = 256, level = 64, key = "") -> dict:
    md6 = MD6(size=size, levels=level, key=key)
    md6_hash = md6(texto)
    return {"hash": str(md6_hash), "hash_raw": md6_hash.raw()}
```

```
def gost(self, texto = "hello", modo: int = 0) -> str:
    modos = ('streebog256', 'streebog512')
    conversion_bytes = texto.encode()
    gost_obj = gostcrypto.gosthash.new(modos[modo], data=conversion_bytes)
    gost_hash = gost_obj.hexdigest()
    return gost_hash
```

Labor 3: Codificaciones

He trabajado con varios tipos de Codificaciones:

- Binario
- Decimal
- Base16 o Hexadecimal
- Base32, Base45, Base58, Base62, Base64, Base85, Base92

```
1  import base64
2  import py3base92 as base92
3
4  class codificacion_datos():
5
6      def codificar_binario(self, texto = "hello") -> list:
7          paso_bytes = bytes(texto, "utf-8")
8          mapeo = map(bin, paso_bytes)
9          return list(mapeo)
10
11     def codificar_decimal(self, texto = "hello") -> list:
12         return [ord(caracter) for caracter in texto]
13
14     def codificar_base16(self, texto = "hello") -> bytes:
15         """
16         Con esta funcion codificas
17         el texto en base hexadecimal
18         """
19         codificacion = texto.encode()
20         paso_bytes = bytes(codificacion)
21         return base64.b16encode(paso_bytes)
22
23     def codificar_base32(self, texto = "hello") -> bytes:
24         codificacion = texto.encode()
25         paso_bytes = bytes(codificacion)
26         return base64.b32encode(paso_bytes)
```

Labor 4: Encriptaciones

He trabajado con varios tipos de Encriptaciones:

- DES
- Twofish
- ROT13
- Hill con matrices de 2x2
- Atbash

```
def cifrado_hill(self, texto: str, matriz: list):
    # https://es.wikipedia.org/wiki/Cifrado\_Hill
    # https://www.youtube.com/watch?v=Ch0hsL-zvBo
    abecedario = {'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6,
                  'H': 7, 'I': 8, 'J': 9, 'K': 10, 'L': 11, 'M': 12, 'N': 13,
                  'O': 14, 'P': 15, 'Q': 16, 'R': 17, 'S': 18, 'T': 19, 'U': 20,
                  'V': 21, 'W': 22, 'X': 23, 'Y': 24, 'Z': 25, 'None': 26}

    texto_mayusculas = texto.upper()

    # Pasamos las letras a numeros
    codificacion = list()
    for caracter in texto_mayusculas:
        codificacion.append(abecedario[caracter])

    # Nos aseguramos que el numero de longitud de la cadena sea multiplo del tamaño de la matriz
    while len(codificacion) % len(matriz[0]) != 0:
        codificacion.append(26)
```

Labor 5: Scapy

He mirado un poquito más la librería Scapy.

He aprendido a crear paquetes con la IP de origen ofuscada.

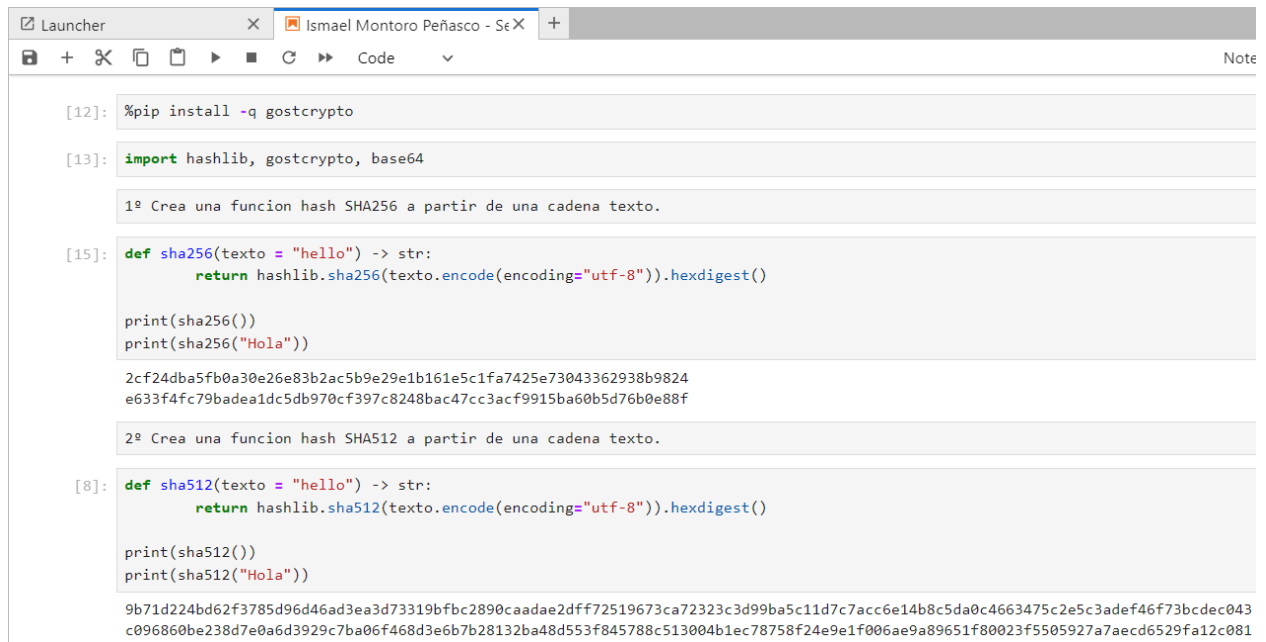
```
def ofuscar_origen(ip: str):
    # Crear un paquete ICMP con una dirección IP falsa
    paquete = IP(src="150.230.2.21", dst=ip)/ICMP()
    # Enviar el paquete
    send(paquete)
```

He realizado una extracción de la información de un paquete y la he metido en un diccionario de diccionarios para poder manejar dicha información más fácil.

```
68 def descomponer_paquete(paquete) -> dict:
69     partes = dict()
70
71     archivo_salida = "salida_paquete.txt"
72     posicionamiento.ubicarme_directorio_actual(__file__)
73     with open(archivo_salida, "w") as archivo:
74         with redirect_stdout(archivo):
75             paquete.show()
76
77     with open(archivo_salida, "r") as archivo:
78         componetes = archivo.read()
79         componetes = componetes.split("\n")
80         titulo_campo = ""
81         for campo in componetes:
82             if "\options" in campo:
```

Labor 6: Ejercicios de la semana

Como es costumbre he realizado los diez ejercicios semanales de python, esta vez todo orientada a cifrados, hashes y codificaciones



The screenshot shows a Jupyter Notebook window with the title 'Ismael Montoro Peñasco - Set'. The interface includes a 'Launcher' button, a file explorer, and a 'Code' dropdown menu. The notebook contains the following code and output:

```
[12]: %pip install -q gostcrypto
```

```
[13]: import hashlib, gostcrypto, base64
```

1º Crea una funcion hash SHA256 a partir de una cadena texto.

```
[15]: def sha256(texto = "hello") -> str:
      return hashlib.sha256(texto.encode(encoding="utf-8")).hexdigest()

print(sha256())
print(sha256("Hola"))
```

2cf24dba5fb0a30e26e83b2ac5b9e29e1b161e5c1fa7425e73043362938b9824
e633f4fc79badea1dc5db970cf397c8248bac47cc3acf9915ba60b5d76b0e88f

2º Crea una funcion hash SHA512 a partir de una cadena texto.

```
[8]: def sha512(texto = "hello") -> str:
      return hashlib.sha512(texto.encode(encoding="utf-8")).hexdigest()

print(sha512())
print(sha512("Hola"))
```

9b71d224bd62f3785d96d46ad3ea3d73319bfb2890caadae2dff72519673ca72323c3d99ba5c11d7c7acc6e14b8c5da0c4663475c2e5c3adef46f73bcdec043
c096860be238d7e0a6d3929c7ba06f468d3e6b7b28132ba48d553f845788c513004b1ec78758f24e9e1f006ae9a89651f80023f5505927a7aec6529fa12c081

He aprendido a Rastrear precedentes, solo hay que seleccionar una celda que tenga referenciada a otra celda, y presionar Rastrear precedentes

formulas		D	E	F
	A			
				7
	Ejemplo			
	A			
	DEVUELVE LA CADENA INTRODUCIDA A MAYUSCULA			
	devuelve la cadena introducida a minuscula			
	VERDADERO			
Completa	Nos da un carácter dependiendo del decimal de la tabla Ascii; Devuelve la cadena introducida a mayuscula; Devuelve la cadena introducida a minuscula; Devuelve Verdadero si la cadena es igual a la			
	la			2 a
	Ho			1 das
	HolaMundo!			3 dsa
	Holmkdndo!			
	01/03/2024			
	01/03/2024 1:22			
de dinero	20,00 €			
	25/02/2021			
	2021			
	2			
	25			

He aprendido a rastrear aquellas celdas dependientes de una celda en concreto

Archivo Inicio Insertar Disposición de página Fórmulas Datos Revisar Vista Ayuda				
<div> <div>Insertar función</div> <div>Autosuma</div> <div>Usado recientemente</div> <div>Financieras</div> <div>Lógicas</div> <div>Texto</div> <div>Fecha y hora</div> <div>Búsqueda y referencia</div> <div>Matemáticas y trigonométricas</div> <div>Más funciones</div> </div> <div>Biblioteca de funciones</div> <div> <div>Administrador de nombres</div> <div>Asignar nombre</div> <div>Utilizar en la fórmula</div> <div>Crear desde la selección</div> <div>Nombres definidos</div> </div> <div> <div>Rastrear precedentes</div> <div>Rastrear dependientes</div> <div>Quitar flechas</div> </div>				
D21				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				
21				
22				
23				
24				

He aprendido a manejar las funciones FECHANUMERO, DIAS360, COLUMNAS y COINCIDIR.

BUSCARH	Busca en HORIZONTAL.	IZQUIERDA
FECHANUMERO	Devuelve un fecha en formato numerico desde una cadena de texto	22/02/2024 0:00
DIAS360	Devuelve el numero de dias que hay entre dos fechas	254
COLUMNAS	Devuelve el numero de columnas de una matriz	3
COINCIDIR	Devuelve una coincidencia de una columna que tu selecciones	15

Planes para la próxima semana

Terminar con los ejercicios de hashes y encriptación, y seguir con Scapy y con Excel.