

Competición:

Selección nacional ECSC2023

Solucionario reto 8





SECRETARÍA DE ESTADO DE DIGITALIZACIÓN E INTELIGENCIA ARTIFICIAL



Plan de Recuperación, Transformación y Resiliencia







ÍNDICE

1. CARACTERÍSTICAS DE LA COMPETICIÓN	3
2. INFORMACIÓN DE RETO	4
3. SOLUCIÓN RETO 08	5





1. CARACTERÍSTICAS DE LA COMPETICIÓN

La competición **"Selección nacional para los ECSC2023"** tuvo lugar el 10 de junio de 2023 de forma online.







2. INFORMACIÓN DE RETO

Información general

- Identificador: Reto 08
- Categoría: Miscelánea
- Puntuación: 300
- Dificultad: Alta
- Tipo: Descargable

Conocimientos y habilidades

- MITRE: Discovery / Collection / Account Discovery
- NICE: Knowledge of Application Security Risks, Knowledge of hacking methodologies
- ENISA: Information and Knowledge Management, Information Security Management







3. SOLUCIÓN RETO 08

Enunciado

El equipo de seguridad del presidente del gobierno ha detectado un posible email malicioso, por suerte este dio error al ejecutarse, deberás investigarlo y sacar toda la información posible acerca de él.

Se nos piden unas coordenadas con 5 decimales.

Formato: flag{latitud, longitud}

Flag

 $flag{40.24521, -6.18136}$

Pistas

- 1. Investiga los commits de los repositorios.
- 2. Cualquier información es valiosa. Vuelve hacia atrás.
- 3. Mira que hay detrás de las webs





Solución

En el archivo data.eml, encontramos un correo electrónico. Dentro del mismo, hay un archivo ejecutable .exe. Observamos que no se puede ejecutar, de manera que procedemos a abrirlo en el programa Ghidra.

SG	servicio tecnico gobierno Actualización necesaria en su s	sistema			3 April 2070 at 14:06
Bueno	s dias senor presidente,				
Desde favor, I Mucha	el servicio técnico le solicita haga doble click en el siguier is gracias.	mos que haga una a nte ejecutable para p	ctualización de oder aplicar el p	seguridao barche.	d en su dispositivo. Por
Buend Desde favor, Mucha Un sal	el servicio técnico le solicita naga doble click en el siguier ls gracias. udo	mos que haga una a nte ejecutable para p	ctualización de oder aplicar el p	seguridao barche.	d en su dispositivo. Por
Bueno Desde favor, Mucha Un sal	el servicio técnico le solicita haga doble click en el siguier is gracias. udo	mos que haga una a nte ejecutable para p	ictualización de ioder aplicar el p	seguridad barche.	d en su dispositivo. Por

Ilustración 1 - Correo electrónico

Dentro de Ghidra, procedemos a realizar ingeniería inversa al binario para descubrir su funcionamiento y poder extraer la máxima información posible.

Debemos buscar la función "main" para empezar a entender el programa, podemos observar que se realizan varias llamadas al sistema.







Ilustración 2 - Función main descompilada en Ghidra

Procedemos a fijarnos en los datos declarados en la función. Encontramos un enlace hacia GitHub.

:	*	FUNCTION	*	
:			ki kiki kiki kiki kiki kiki kiki kiki	
	int <u>cdecl mai</u> assume GS_OF	n(int _Argc, char * * _Argv, ch FSET = 0xff00000000	ar * * _Env)	
int	EAX:4	<return></return>		
int	ECX:4	_Argc		
char * *	RDX:8	_Argv		
char * *	R8:8	_Env		
undefined1	Stack[-0x18]	1 local_18	XREF[1]:	14000165b(*)
undefined8	Stack[-0x20]	8 local_20	XREF[1]:	14000166c(W)
undefined8	Stack[-0x28]	8 local_28	XREF[3]:	140001677(W),
				140001695(R),
				1400016c7(R)
undefined8	Stack[-0x30]	8 local_30	XREF [4]:	140001682(W),
				1400016a4(R),
				1400016c3(R),
				1400016fc(R)
undefined8	Stack[-0x38]	8 local_38	XREF [4]:	1400016bf <mark>(W)</mark> ,
				1400016cb(R),
				1400016e4(R),
				1400016f0(R)
1	main		XREF[3]:tmai	nCRTStartup:1400013bc(c),
			14000a	1d8(*), 14000a1e0(*)
140001655 <mark>55</mark>	PUSH	RBP		
140001656 <mark>53</mark>	PUSH	RBX		
140001657 48 83 ec 48	SUB	RSP,0x48		
14000165b <mark>48 8d 6</mark> c	LEA	RBP=>local_18,[RSP + 0x40]		
24 40				
140001660 <mark>e8 7b 01</mark>	CALL	main	und	definedmain(void)
00 00				
140001665 48 8d 05 94 79 00 00	LEA	RAX,[s_https://github.com/shoka	amon/haha_140009 = '	'https://github.com/shokamon/h
14000166c 48 89 45 f8	MOV	qword ptr [RBP + local_20],RAX=	<pre>=>s_https://gith = '</pre>	'https://github.com/shokamon/h
140001670 48 8d 05	LEΔ	RAX [s hababa zin 14000049]		hahaha zin"







Ilustración 4 - Declaración de strings en Ghidra

Tras buscar las strings definidas, podemos ver el enlace completo, un enlace de descarga a un archivo .zip de un repositorio de GitHub.

Sign up	C) =
📮 shokamon / hahaha (Public)	○ Notifications 양 Fork 0 ☆ Star 0 ~
<> Code ⊙ Issues î1 Pull requests (🖸 Actions 🖽 Projects 🕕 Security 🗠 Insights
^{₽9} main ▾	Go to file Code - About
shokamon Add files via upload 📖	2 minutes ago 🕚 5 No description, website, or topics
DesktopGoose v0.31 Add files via uplo	ad 2 minutes ago
Create README.md Create README.	nd 2 weeks ago ① 1 watching
🕒 goose.zip Add files via uple	ad yesterday 양 0 forks
README.md	Report repository
	Releases
	No releases published

Ilustración 5 - Página del repositorio en GitHub

eration tG.



Sign up			Ç							≡
	III Ov	erview	🗍 Repo	ositories 3	⊞	Projects	🗘 Pack	kages 🖞	7 Stars	
	Popular repositories									
OF BE	shokamon	shokamon			Public ideal-barn. This is a new		rnacle ew reposito	cte repository		ublic
shokamon	hahaha			Public						
Follow	35 contribut	ions in t	he last y	'ear						
 nowhere and everywhere Joined 2 weeks ago Block or Report 	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr
	Learn ho	w we count	contributi					Less	Ma	

Ilustración 6 - Páginad de usuario en GitHub

Investigamos el perfil de github y encontramos 3 repositorios. En GitHub, el README es un archivo de texto breve que se encuentra en la raíz del repositorio y proporciona información sobre el proyecto.

El repositorio que tiene el mismo nombre que el usuario, actua como "página principal del perfil", normalmente encontraremos en ellos información sobre el usuario, RRSS, etc.

Sign up	() =
🛱 shokamon / shokamon (Public)	다 Notifications 🔮 Fork 0 🏠 Star 0 🚽
<> Code ⊙ Issues the Pull requests ⊙ Actions	🗄 Projects 🕕 Security 🗠 Insights
^{₽9} main →	Go to file Code - About
💮 shokamon Update README.md 📖	No description, website, or topics 2 weeks ago 26 provided.
README.md Update README.md	2 weeks ago ☐ Readme
README.md	ⓒ 1 watching 양 0 forks Report repository

Ilustración 7 - README.md del usuario





Procedemos a observar los distintos cambios del repositorio, en el commit acf58ab107cf11f678cafa29d428415de05b39f3 (anon), encontraremos que el usuario eliminó diversas cosas del readme.md, entre ellas un link a una cuenta de Twitter.

anon			Browse files
sho	okam	non committed 2 weeks ago (Verified) 1 parent 31d5cd9	commit acf58ab
Showing	1 ch	nanged file with 0 additions and 8 deletions.	Split Unified
	8	README.md (C	↔ 🗅 …
		@@ -30,13 +30,5 @@	
30 31 32	30 31 32		
33		<pre>- <div align="left"></div></pre>	
34		- <ing alt="li</mark>nkedin logo" height="40" src="https://raw.githubusercontent.com/maurodesouza/profile-readme-generator/master/src/assets/icons/social/linked</p></th><th>in</th></tr><tr><th></th><td></td><td>/default.sy<mark>o" width="52"></ing> <td></td>	
35		- 	
36		- <img alt="twitter logo" height="40" src="https://raw.githubusercontent.com/maurodesouza/profile-readme-generator/master/src/assets/icons/social/twit
/default.svg" width="52"/>	ter
37		-	
38		- <img 52"="" alt="discord logo" height="40" src="https://raw.githubusercontent.com/maurodesouza/profile-readme-generator/master/src/assets/icons/social/discor
width="/>	d/default.svg"
39		- <img 52"="" alt="youtube logo" height="40" src="https://raw.githubusercontent.com/maurodesouza/profile-readme-generator/master/src/assets/icons/social/youtub
width="/>	e/default.svg"
40		-	
41	33		
	34	888	

Ilustración 8 - Commit en el que aparece un enlace a Twitter



Ilustración 9 - Perfil de Twitter







llustración 7 - Tweet con enlace a Mastodon

Buscando en tweets antiguos, encontramos un enlace a la red social Mastodon. https://mastodon.social/@kashoma



Ilustración 8 - Perfil de Mastodon

Analizando todas las publicaciones y las respuestas a ellas, encontramos que el usuario @lyrazenith, interactua en diversas ocasiones con @kashoma.







Ilustración 10 - Interacción con otra cuenta



Ilustración 9 - Cuenta 2





Ilustración 11 - Publicaciones y respuestas

Nos desplazamos a la sección de "Posts and replies" para analizar todas las interacciones de esta cuenta con otras.



Ilustración 12 - Enlace a Bot de Telegram

En esta respuesta a @kashoma encontramos un enlace a un bot de telegram, t.me/Neuronetix_bot.



Ilustración 13 - Bot de Telegram





Tras iniciar el Bot, debemos intentar sacar la máxima información posible. Para ello ejecutamos el comando /help, que suele estar presente en la mayoría de Bots, nos pregunta que cual es nuestro nombre.

SECRETÁRIÁ DE ESTADO

Ejecutamos el comando Lyra, ya que es la creadora del Bot y nos devuelve una string cifrada.

Neuronetix bot	Q		0 0 0
tiyaf.Onfvney p ctlto.omj pflkod.ried kzl	21	:34	
14 de abril			
	/lyra	20:20	<i>*</i>
Jml mfqk vtqzt efzs dgke Nsrrplka j qvid.Azeyrf uym.Fcscet tiyaf.Onfvney p ctlto.omj pflkod.ried kzl	l w 20:	:20	
24 de abril			
	/lyra	20:26	<i>*</i>
Jml mfqk vtqzt efzs dgke Nsrrplka j qvid.Azeyrf uym.Fcscet tiyaf.Onfvney p ctlto.omj pflkod.ried kzl	l w 20:	:26	
30 de abril			
	/lyra	10:53	~
Jml mfqk vtqzt efzs dgke Nsrrplka j qvid.Azeyrf uym.Fcscet tiyaf.Onfvney p ctlto.omj pflkod.ried kzl	lw 10:	:53	
Escribe un mensaje		::	Ŷ

\$incibe





Ilustración 14 - Obtención de mensaje encriptado en Telegram

Para conseguir descifrarlo, debemos volver hacia atrás y analizar otra vez el perfil de GitHub. Además de todos los repositorios publicados en el perfil, encontramos los Gist, son pequeñas publicaciones de código. En este caso encontramos un programa escrito en Python que cifra strings.

\leftarrow \rightarrow \mathbf{C} $\mathbf{\widehat{G}}$ $\mathbf{\widehat{O}}$ $\mathbf{\widehat{A}}$ https://gist.github.com/shol	mon			☆
GitHubGist Search All	ists Back to GitHub			Sign ir
	Instantly share code, notes, and snippets.			
	• All gists 1		Sort: Recently of	created -
	Created 51 minutes ago crypher and decypher	1 file 양 0 forks	s 💭 0 comments	ත් 0 stars
	<pre>1 import string 2 3 def cifrado_vigenere(texto, clave): 4 alfabeto = string.ascii_letters + string.digits + string.punctuation + "</pre>			
abataman	<pre>5 resultado = "" 6 clave_extendida = clave * (len(texto) // len(clave) + 1) 7 for i, caracter in enumerate(texto): 8 if caracter in alfabeto:</pre>			
on nowhere and everywhere	9 codigo = alfabeto.index(caracter) + alfabeto.index(clave_extendid 10 resultado += alfabeto[codigo % len(alfabeto)]	(1)		
A Joined 2 weeks ago				
View GitHub Profile				

llustración 15 - Gist de GitHub





🛯 Para	lyra	Raw
	import string	
	def cifrado_vigenere(texto, clave):	
	alfabeto = string.ascii_letters + string.digits + string.punctuation + " "	
	resultado = ""	
	clave_extendida = clave * (len(texto) // len(clave) + 1)	
	for i, caracter in enumerate(texto):	
	if caracter in alfabeto:	
	codigo = alfabeto.index(caracter) + alfabeto.index(clave_extendida[i])	
	resultado += alfabeto[codigo % len(alfabeto)]	
11	else:	
12	resultado += caracter	
13	return resultado	
14		
	def descifrado_vigenere(texto, clave):	
17	alfabeto = string.ascii_letters + string.digits + string.punctuation + " "	
	resultado = ""	
	clave_extendida = clave * (len(texto) // len(clave) + 1)	
	for i, caracter in enumerate(texto):	
21	if caracter in alfabeto:	
22	codigo = alfabeto.index(caracter) - alfabeto.index(clave_extendida[i])	
23	resultado += alfabeto[codigo % len(alfabeto)]	
	else:	
	resultado += caracter	
	return resultado	
28		
	texto_original = input("Ingrese el texto que desea encriptar: ")	
30	clave = input("Ingrese la clave de encriptación: ")	
32	texto_encriptado = cifrado_vigenere(texto_original, clave)	
	print("Texto encriptado:", texto_encriptado)	
	texto_desencriptado = descifrado_vigenere(texto_encriptado, clave)	
	print("Texto desencriptado:", texto_desencriptado)	
37		

Ilustración 16 - Cifrador publicado por Shoka

Probando el programa nos damos cuenta de que solicita un texto, una contraseña y devuelve el texto encriptado.



Ilustración 17 - Muestra del funcionamiento del programa





Recuperación, Transformación Resiliencia Resiliencia



```
import string
def cifrado_vigenere(texto, clave):
    alfabeto = string.ascii_letters + string.digits + string.punctuation + " "
    resultado = ""
    clave_extendida = clave * (len(texto) // len(clave) + 1)
    for i, caracter in enumerate(texto):
        if caracter in alfabeto:
            codigo
                                          alfabeto.index(caracter)
                             =
                                                                              +
alfabeto.index(clave_extendida[i])
           resultado += alfabeto[codigo % len(alfabeto)]
        else:
           resultado += caracter
    return resultado
def descifrado_vigenere(texto, clave):
    alfabeto = string.ascii_letters + string.digits + string.punctuation + " "
    resultado = ""
    clave_extendida = clave * (len(texto) // len(clave) + 1)
    for i, caracter in enumerate(texto):
        if caracter in alfabeto:
            codigo
                                          alfabeto.index(caracter)
alfabeto.index(clave_extendida[i])
           resultado += alfabeto[codigo % len(alfabeto)]
        else:
           resultado += caracter
    return resultado
texto_encriptado = input("Ingrese el texto que desea desencriptar: ")
clave = input("Ingrese la clave de encriptación: ")
texto_desencriptado = descifrado_vigenere(texto_encriptado, clave)
print("Texto desencriptado:", texto_desencriptado)
texto_encriptado = cifrado_vigenere(texto_desencriptado, clave)
print("Texto encriptado:", texto_encriptado)
```





La cadena encontrada en el bot de Telegram, es un cifrado "Vigenère" y su clave es "lyra", al descifrarlo obtenemos el siguiente mensaje.

Recipe	2 • 1	Input	+ (
Vigenère Decode	⊘ 11	Jml mfqk vtqzt efzs dgke Nsrrplka j qvid.Azeyrf uym.Fcscetl w tiyaf.Onfvney p ctlto.omj pflkod.ried k	z1	
Key lyra				
		ane 103 📻 1 🛕 103		Ŧ
		Output	1	3
		You must visit this site Cuarenta y seis.Ciento uno.Ochenta y cinco.Ochenta y cinco.dos puntos.tres m	il	

You must visit this site Cuarenta y seis.Ciento uno.Ochenta y cinco.Ochenta y cinco.dos puntos.tres mil

Lo que es lo mismo, esta URL:

```
http://46.101.85.85:3000/
```

Siguiendo este enlace, nos lleva a un formulario de inicio de sesión con una sección de forgot password.



Ilustración 18 - Inicio de sesión del enlace recibido

Nos pide el nombre de nuestra mascota para recuperar la contraseña.





Ilustración 19 - Página de contraseña olvidada

Volviendo a los perfiles de mastodon, encontramos un post en el cual Lyra, publica el nombre de su gato.



Ilustración 20 - Nombre de la mascota de Lyra





Tras introducir este nombre la página no realiza ningún cambio, analizando el monitor de red de las herramientas para desarrolladores del navegador, observamos una ruta hacia "secret".

			FORG	OT PA	SSWORD?	•		
				jac	cu			
				Sub	mit			
Lik 📋 Elements Cor	isole Sour	rces Pe	erformance ir	nsights	Netwo	ork Perfo	rmance »	>
● ◎ ¥ ♀ □ Pre	eserve log	Disabl	e cache No	throttli	ng 🔻 😪	1 ± ±		
Filter	Invert (Hide d	ata URLs All	Fetcł	h/XHR JS CS	S Img Me	dia Font D	>
Has blocked cookies B	locked Keque	ests 📋 31	rd-party requ	ests				Joc
20 ms 4	0 ms	60 ms	8	0 ms	100 ms		120 ms	Joc
20 ms 4	0 ms	60 ms	8	0 ms	100 ms		120 ms	Joc
20 ms 4	0 ms	60 ms	8 Type	0 ms	100 ms	Size	120 ms	W
20 ms 4	0 ms	60 ms atus	8 Type fetch / R	0 ms Initiato <u>word.js</u>	100 ms or s:5	Size 223 B	120 ms Time 48 ms	W
20 ms 4	0 ms St 30 30	atus	8 Type fetch / R fetch / R	0 ms Initiato word.js 46.101	100 ms or <u>s:5</u> .85.85/	Size 223 B (disk cac	120 ms Time 48 ms 1 ms	W

Ilustración 21: Vista de las herramientas de desarrollador con la petición creada.



🔁 231s11	🤁 231s110	逼 231s12
📴 231s13	iii 231s14	📔 231s15
📴 231s16	Cal 231s17	逼 231s18
📴 231s19	🤁 aas	📴 ha
a 2578951.jpg	files.html	ho
readme.txt	s	

Ilustración 22: Vista de los archivos.



Ilustración 23: Imagen del servidor.





Para analizar la imagen utilizaremos la herramienta Exiftool para extraer los metadatos.

exiftool 2578951.ipg	
ExifTool Version Number	: 12.48
File Name	: 2578951.jpg
Directory	:.
File Size	: 87 kB
File Modification Date/Time	: 2023:04:10 09:35:02+02:00
File Access Date/Time	: 2023:04:10 09:35:05+02:00
File Inode Change Date/Time	: 2023:04:10 09:35:04+02:00
File Permissions	: -rw-rr
File Type	: JPEG
File Type Extension	: ipg
MIME Type	: image/ipeg
JFIF Version	: 1.01
Exif Byte Order	: Big-endian (Motorola, MM)
Make	: Apple
Camera Model Name	: iPhone 13
X Resolution	:1
Y Resolution	:1 / / / /
Resolution Unit	: None
Y Cb Cr Positioning	: Centered
GPS Version ID	: 2.3.0.0
GPS Latitude Ref	: North
GPS Longitude Ref	: West
Image Width	: 664
Image Height	: 498
Encoding Process	: Progressive DCT, Huffman coding
Bits Per Sample	: 8
Color Components	: 3
Y Cb Cr Sub Sampling	: YCbCr4:2:0 (2 2)
Image Size	: 664×498
Megapixels	: 0.331
GPS Latitude	: 40 deg 14' 42.76" N
GPS Longitude	: 6 deg 10' 52.90" W
GPS Position	: 40 deg 14' 42.76" N, 6 deg 10' 52.90" W

Ilustración 24: Coordenadas GPS en metadatos.

Obtenemos las coordenadas en las que se tomo esta imagen. Debemos convertirlo a dotación "Decimal Degrees" en lugar de DMS (Degrees, minutes, seconds)

Ilustración 25 - Metadatos de la imagen





Degrees Minutes Seconds to Decimal Degrees

Please enter the degrees, minutes, seconds (DMS) coordinates values to

Degrees for La	titude	Minutes		Seconds	
40	0	14		42.76	"
Degrees for Lo	ngitude	Minutes		Seconds	
6	•	10		52.90	"
Convert to Decimal Degrees					
			Desident De		
Decimal Degre	es Lat		Decimal De	grees Long	
Decimal Degree	ees Lat	0	6.181361	grees Long 11	0
Decimal Degre	es Lat 1	o	6.181361	grees Long 11	o

Ilustración 26: Convertidor de coordenadas a dotación decimal de la web gps-coordinates.org

DD (decimal degrees)	
Latitude 40.2452111111111	
Longitude -6.18136111111111	
Get Address	
DMS (degrees, minutes, seconds)	
Latitude 0 40 14 42.759	
Longitude ^O ^O 6 [°] 10 ['] 52.899 ^{''}	
Get Address	

Ilustración 27: Muestra de la localización.

Flag: flag{40.24521, -6.18136}