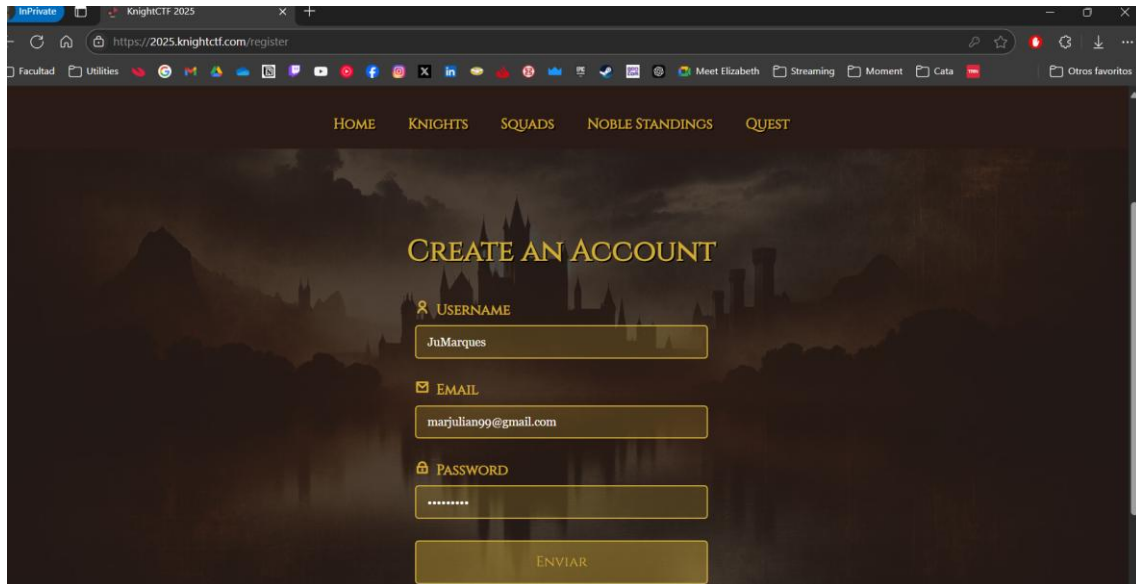


Informe trabajo final Introducción a la Ciberseguridad

Captura del registro del perfil:



HOME KNIGHTS SQUADS NOBLE STANDINGS QUEST

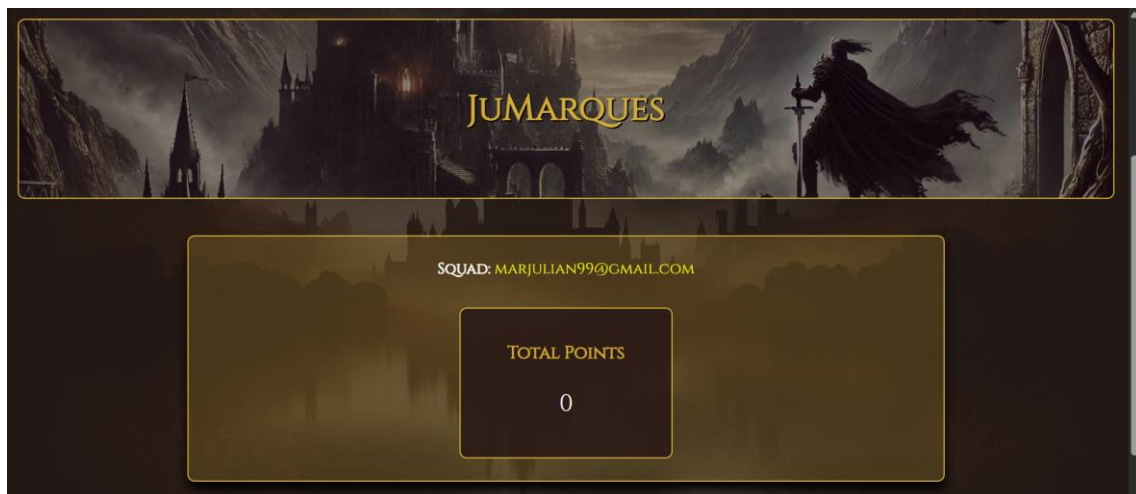
CREATE AN ACCOUNT

👤 USERNAME
JuMarques

✉ EMAIL
marjulian99@gmail.com

🔒 PASSWORD

ENVIAR



JUMARQUES

SQUAD: MARJULIAN99@GMAIL.COM

TOTAL POINTS
0

Ejercicio 1 PWN

Nombre del ejercicio: **Knight Bank**

Categoría: **Exploiting**

Descripción del ejercicio:

In the heart of the ancient kingdom lies the Knight Bank, a fortress of wealth guarded by its intricate magical arithmetic. Only the most cunning warriors who

understand the secret vulnerabilities of numbers can uncover the hidden treasure buried deep within its vaults.

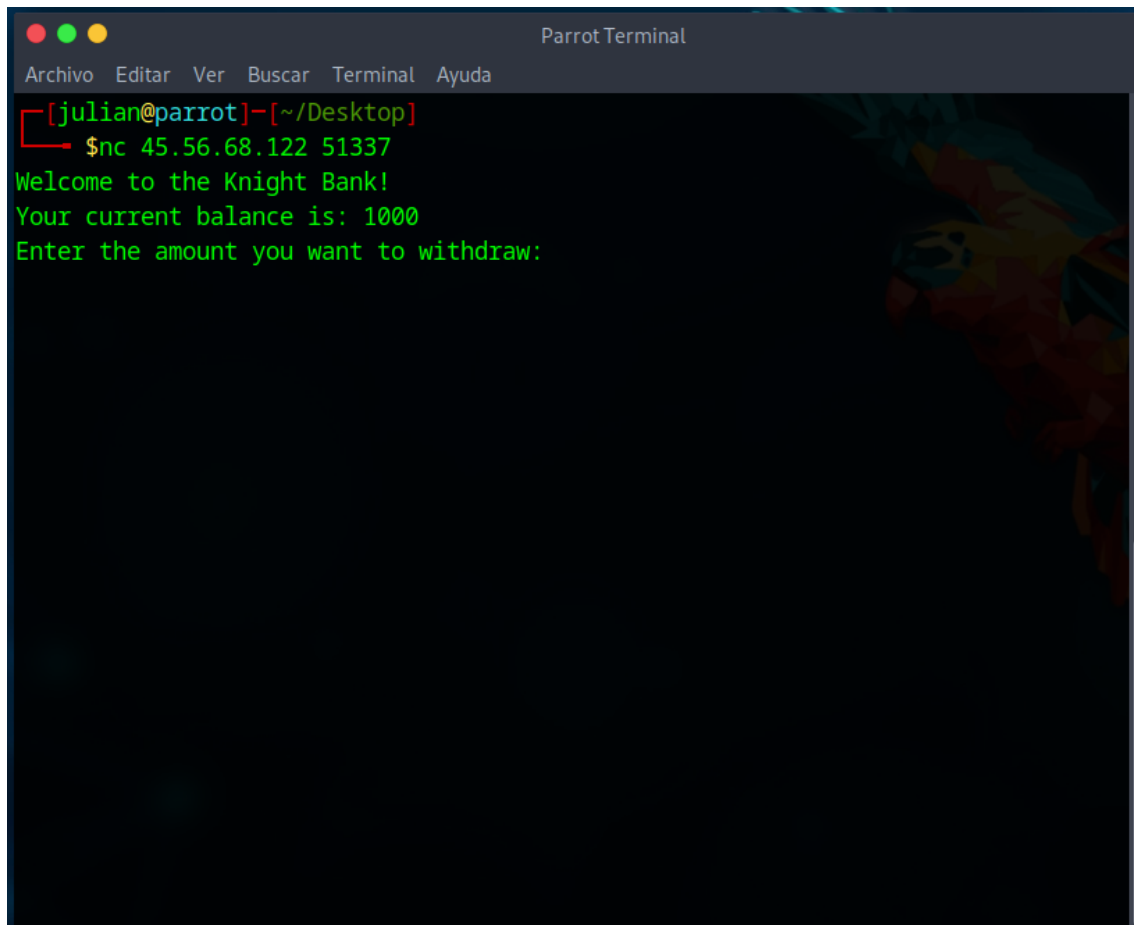


knight_bank

Archivo adjunto:

Conexion: nc 45.56.68.122 51337

Lo primero que hice para resolver este ejercicio fue ejecutar el comando para realizar la conexión con la aplicación.

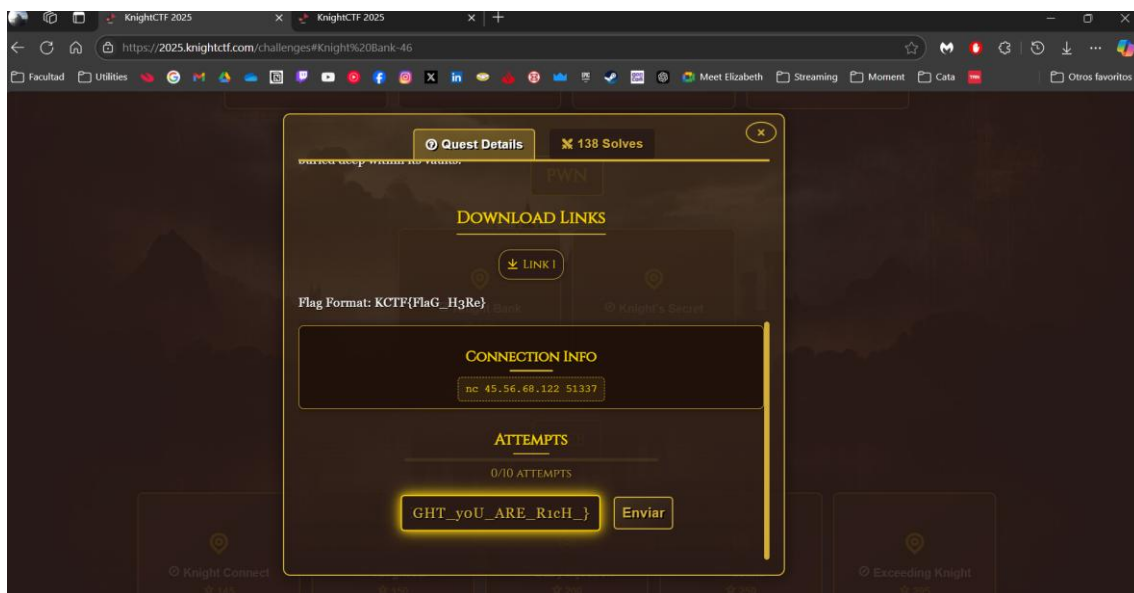


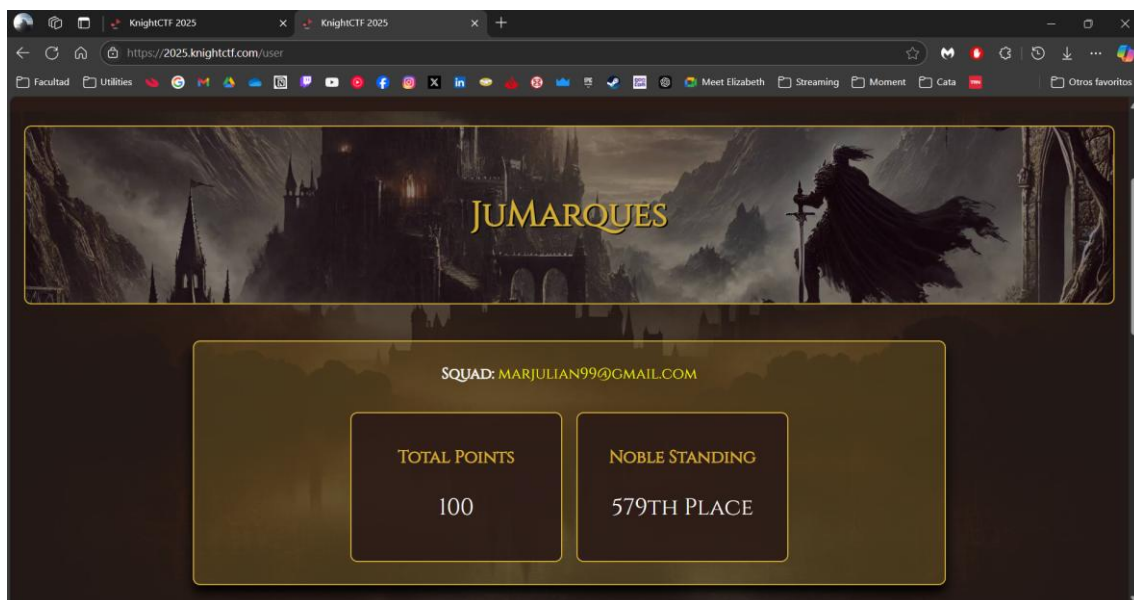
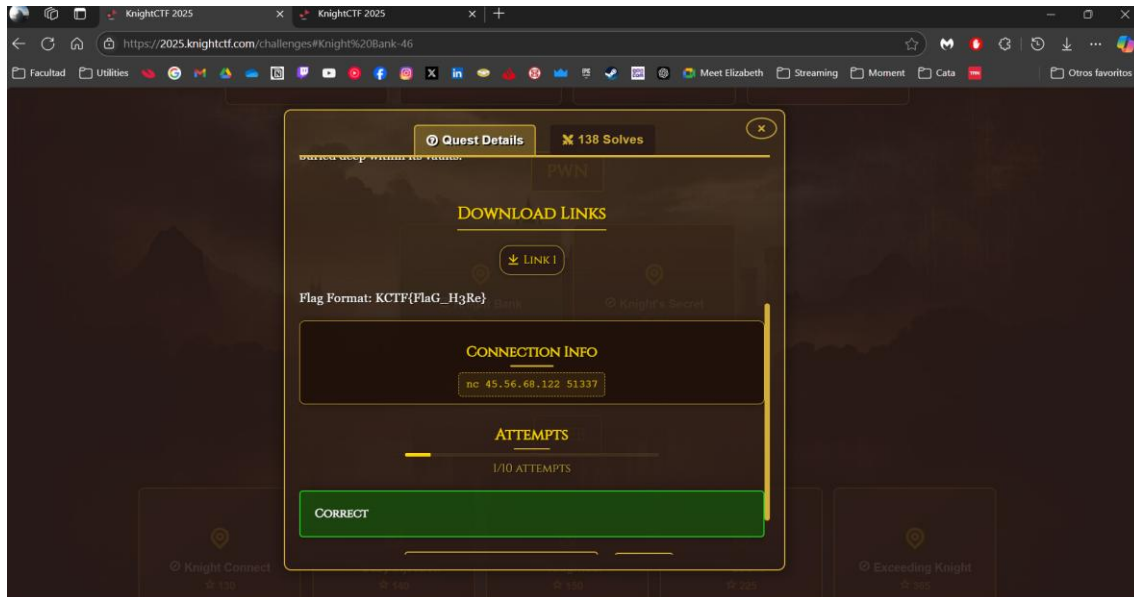
```
Parrot Terminal
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
[julian@parrot]--[~/Desktop]
$nc 45.56.68.122 51337
Welcome to the Knight Bank!
Your current balance is: 1000
Enter the amount you want to withdraw:
```

Acto seguido probe intentando sacar más dinero que el que había depositado en la cuenta bancaria. Como resultado obtuve la flag para completar el desafio

```
Parrot Terminal
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
[julian@parrot]-[~/Desktop]
$nc 45.56.68.122 51337
Welcome to the Knight Bank!
Your current balance is: 1000
Enter the amount you want to withdraw: 2000
You withdrew 2000. Your new balance is 4294966296.
Congratulations! You win the prize!
KCTF{W0W_KNIGHT_y0U_ARE_R1cH_}
```

Capturas evidencia de resolución:





Ejercicio 2 PWN

Nombre del ejercicio: **Knight's Secret**

Categoría: **PWN**

Descripción del ejercicio: Exploit & get the flag.

Conexión info: nc 45.56.68.122 1337

Resolución:

Hice la conexión con NC a la ip otorgada para el ejercicio obteniendo la siguiente respuesta

```
Parrot Terminal
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
[julian@parrot]--[~/Desktop]
$nc 45.56.68.122 1337
=====
Welcome to the Knight's Secret!
The castle's vault holds a secret key, protected within the CONFIG dictionary.
You are a knight tasked with proving the strength of the vault's defenses.
To succeed, you must craft an input to reveal the hidden key within the system.
You will be provided with a user object representing a knight, with attributes '
name' and 'role'.
Once you discover the key, input it again to receive the banner of victory.

Example of a safe template: 'Greetings, {person_obj.name}, the {person_obj.role}
.'
Type 'hint' if you need guidance or 'exit' to withdraw from the quest.
=====

Enter your secret: █
```

Acto seguido probe utilizar el template que otorgaba la conexión

```
Parrot Terminal
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
[julian@parrot]~[~/Desktop]
$nc 45.56.68.122 1337
=====
Welcome to the Knight's Secret!
The castle's vault holds a secret key, protected within the CONFIG dictionary.
You are a knight tasked with proving the strength of the vault's defenses.
To succeed, you must craft an input to reveal the hidden key within the system.
You will be provided with a user object representing a knight, with attributes '
name' and 'role'.
Once you discover the key, input it again to receive the banner of victory.

Example of a safe template: 'Greetings, {person_obj.name}, the {person_obj.role}
.'
Type 'hint' if you need guidance or 'exit' to withdraw from the quest.
=====

Enter your secret: Greetings, {person_obj.name}, the {person_obj.role}
Output: Greetings, Brave Knight, the Defender of the Realm

Enter your secret: █
```

Por lo tanto decidí que la forma para acceder a config iba a ser buscando opciones en el objeto del caballero. Comencé a probar diferentes comandos tratando de encontrar como acceder al CONFIG que contenía la key. Probe acceder a las keys de la clase persona, al método init de la clase y probe también con el comando class.

```
Parrot Terminal
Archivo  Editar  Ver  Buscar  Terminal  Ayuda

To succeed, you must craft an input to reveal the hidden key within the system.
You will be provided with a user object representing a knight, with attributes '
name' and 'role'.
Once you discover the key, input it again to receive the banner of victory.

Example of a safe template: 'Greetings, {person_obj.name}, the {person_obj.role}
.'
Type 'hint' if you need guidance or 'exit' to withdraw from the quest.
=====

Enter your secret: Greetings, {person_obj.name}, the {person_obj.role}
Output: Greetings, Brave Knight, the Defender of the Realm

Enter your secret: {person_obj.__dict__}
Output: {'name': 'Brave Knight', 'role': 'Defender of the Realm'}

Enter your secret: {person_obj.__init__}
Output: <bound method Person.__init__ of <__main__.Person object at 0x70d221c464
80>>

Enter your secret: {person_obj.__class__}
Output: <class '__main__.Person'>

Enter your secret: █
```

Seguía sin encontrar la solución y decidí buscar en la documentación de Python (Ya que las formas me hacían acordar a ese lenguaje porque maneja los objetos como si fueran diccionarios) y ejecute la siguiente línea

```
{person_obj.__class__.__init__.__globals__}
```

```
Parrot Terminal
Archivo  Editar  Ver  Buscar  Terminal  Ayuda

The castle's vault holds a secret key, protected within the CONFIG dictionary.
You are a knight tasked with proving the strength of the vault's defenses.
To succeed, you must craft an input to reveal the hidden key within the system.
You will be provided with a user object representing a knight, with attributes 'name' and 'role'.
Once you discover the key, input it again to receive the banner of victory.

Example of a safe template: 'Greetings, {person_obj.name}, the {person_obj.role}.'
Type 'hint' if you need guidance or 'exit' to withdraw from the quest.
=====

Enter your secret: Greetings, {person_obj.name}, the {person_obj.role}
Output: Greetings, Brave Knight, the Defender of the Realm

Enter your secret: {person_obj.__dict__}
Output: {'name': 'Brave Knight', 'role': 'Defender of the Realm'}

Enter your secret: {person_obj.__init__}
Output: <bound method Person.__init__ of <__main__.Person object at 0x70d221c46480>>

Enter your secret: {person_obj.__class__}
Output: <class '__main__.Person'>

Enter your secret: {person_obj.__class__.__init__.__globals__}
Output: {'__name__': '__main__', '__doc__': None, '__package__': None, '__loader__': <_frozen_importlib_external.SourceFileLoader object at 0x70d221e9f8f0>, '__spec__': None, '__annotations__': {}, '__builtins__': <module 'builtins' (built-in)>, '__file__': '/challenge/challenge.py', '__cached__': None, 'CONFIG': {'KEY': '_KNIGHTSECRET2025_', 'Person': <class '__main__.Person'>, 'fun': <function fun at 0x70d221e86340>, 'main': <function main at 0x70d221c64d60>}}

Enter your secret:
```

Explicación:

`person_obj.__class__.__init__.__globals__`:

`person_obj` es una instancia de la clase `Person`.

`person_obj.__class__` te da la clase de la instancia (en este caso, `Person`).

`person_obj.__class__.__init__` es la referencia al método `__init__` de la clase `Person`.

Los métodos en Python (como `__init__`) tienen un atributo especial llamado `__globals__`, que contiene las variables globales del entorno donde se definió el método.

Al acceder a `person_obj.__class__.__init__.__globals__`, estás directamente accediendo a la variable global en el contexto del método `__init__`, donde `CONFIG` es probablemente definido.

Al ver que `CONFIG` era efectivamente definido allí, introduje la key en el input de la conexión y logre obtener la flag

```
Parrot Terminal
Archivo Editar Ver Buscar Terminal Ayuda
Once you discover the key, input it again to receive the banner of victory.

Example of a safe template: 'Greetings, {person_obj.name}, the {person_obj.role}.'
Type 'hint' if you need guidance or 'exit' to withdraw from the quest.
=====

Enter your secret: Greetings, {person_obj.name}, the {person_obj.role}
Output: Greetings, Brave Knight, the Defender of the Realm

Enter your secret: {person_obj.__dict__}
Output: {'name': 'Brave Knight', 'role': 'Defender of the Realm'}

Enter your secret: {person_obj.__init__}
Output: <bound method Person.__init__ of <__main__.Person object at 0x70d221c46480>

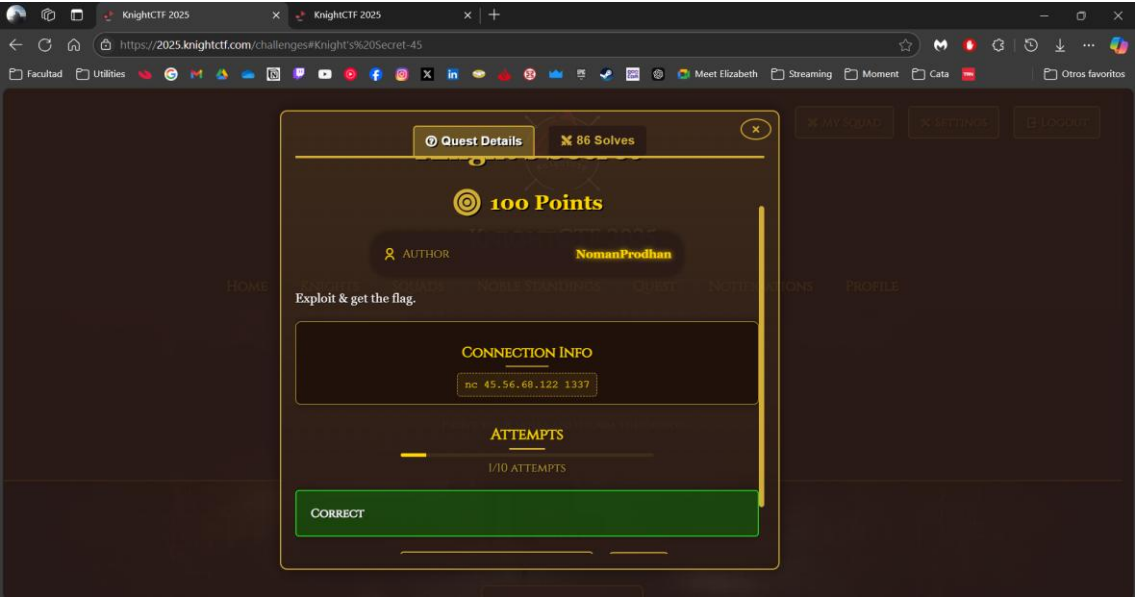
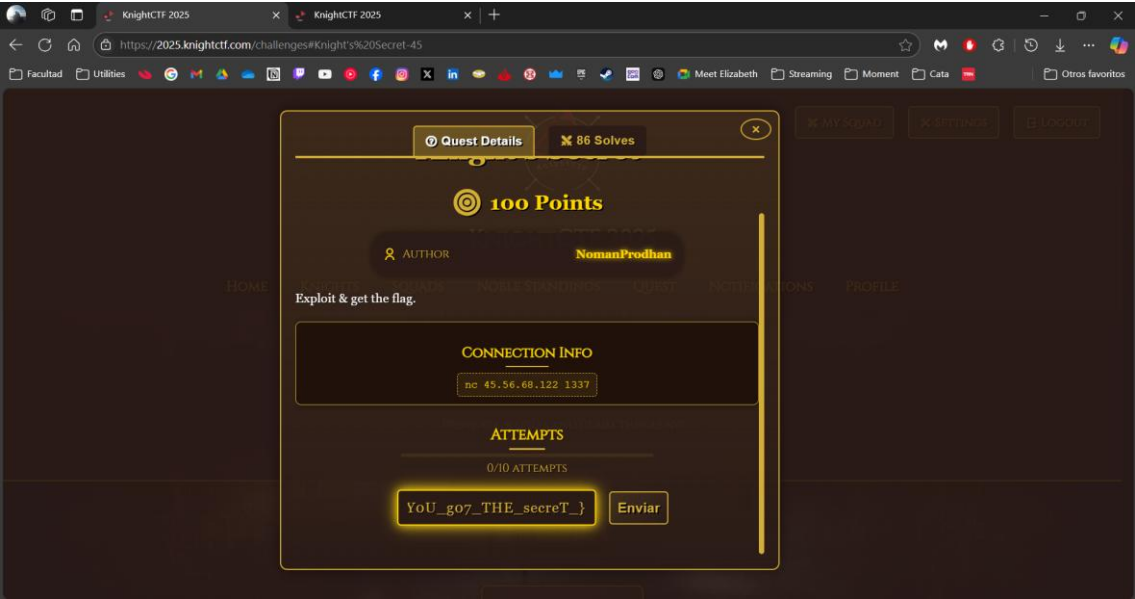
Enter your secret: {person_obj.__class__}
Output: <class '__main__.Person'>

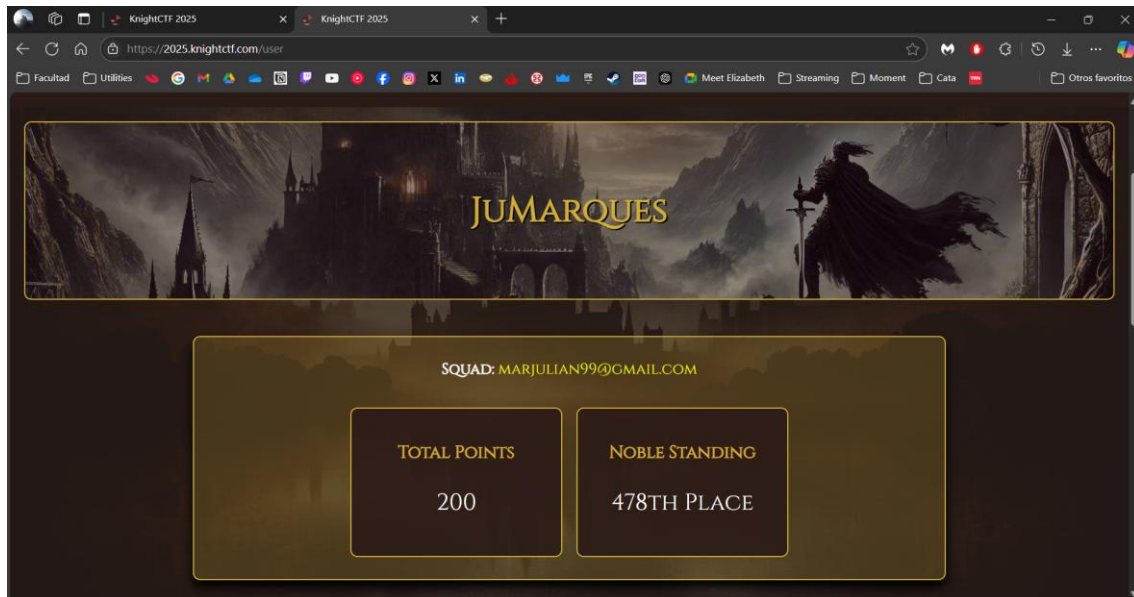
Enter your secret: {person_obj.__class__.__init__.__globals__}
Output: {'__name__': '__main__', '__doc__': None, '__package__': None, '__loader__': <_frozen_importlib_external.SourceFileLoader object at 0x70d221e9f8f0>, '__spec__': None, '__annotations__': {}, '__builtins__': <module 'builtins' (built-in)>, '__file__': '/challenge/challenge.py', '__cached__': None, 'CONFIG': {'KEY': '_KNIGHTSECRET2025_', 'Person': <class '__main__.Person'>, 'fun': <function fun at 0x70d221e86340>, 'main': <function main at 0x70d221c64d60>}}

Enter your secret: _KNIGHTSECRET2025_
Congratulations, noble knight! You have unveiled the vault's secret.
Here is your banner of victory: KCTF{c0ngRaT5_Kn1GHt_Y0U_g07_THE_secrE1_}

[julian@parrot]~$
```


Evidencia de resolución:





Ejercicio 3 Criptografia

Nombre del ejercicio: **Reflections in the Random**

Descripcion: We've uncovered a single string that's saturated with possibilities. At first glance, it might resemble standard Base64 output—but every attempt to decode it directly results in chaotic gibberish.

Some agents suspect an unconventional passphrase or a stray cosmic phenomenon that shifted the bits; others whisper about symmetrical illusions that hide the real message. We even tried old-fashioned classical ciphers—simple shifts, sub-harmonic permutations, you name it—but the truth remains elusive.

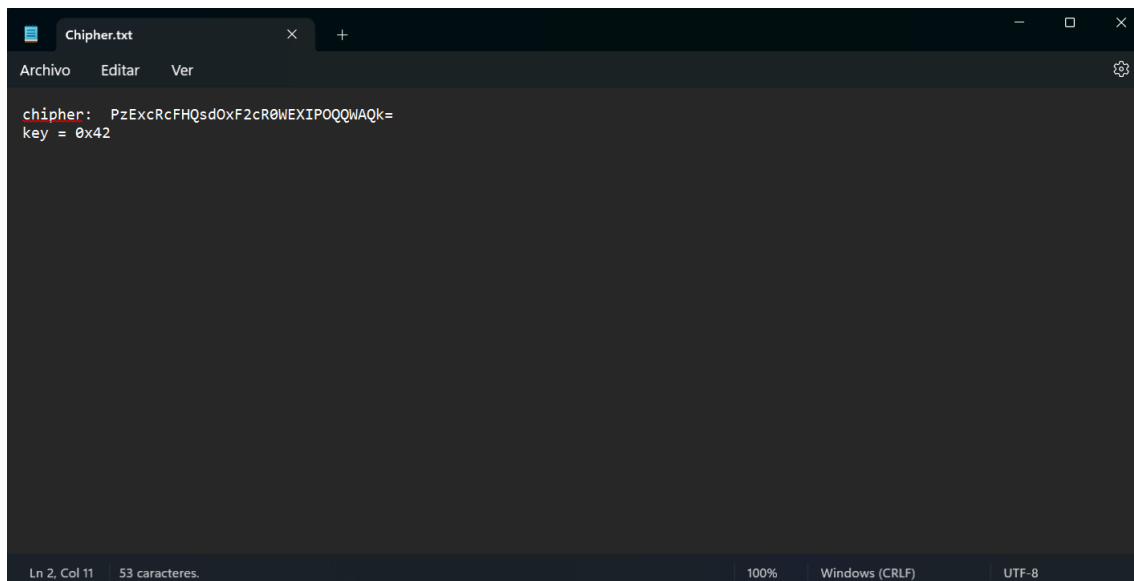
All we know is that the message is said to be “spun backward from a single pivot,” though no one agrees what that means. Could it mean time is reversed? Maybe it's an obscure numeric transformation. Rumor has it that if you find “the key,” everything falls into place. Or maybe it's simpler than we think—just cleverly disguised.

Good luck dissecting this anomaly. Remember: “When the obvious leads nowhere, perhaps the solution sits right in front of you—only viewed from the wrong angle.



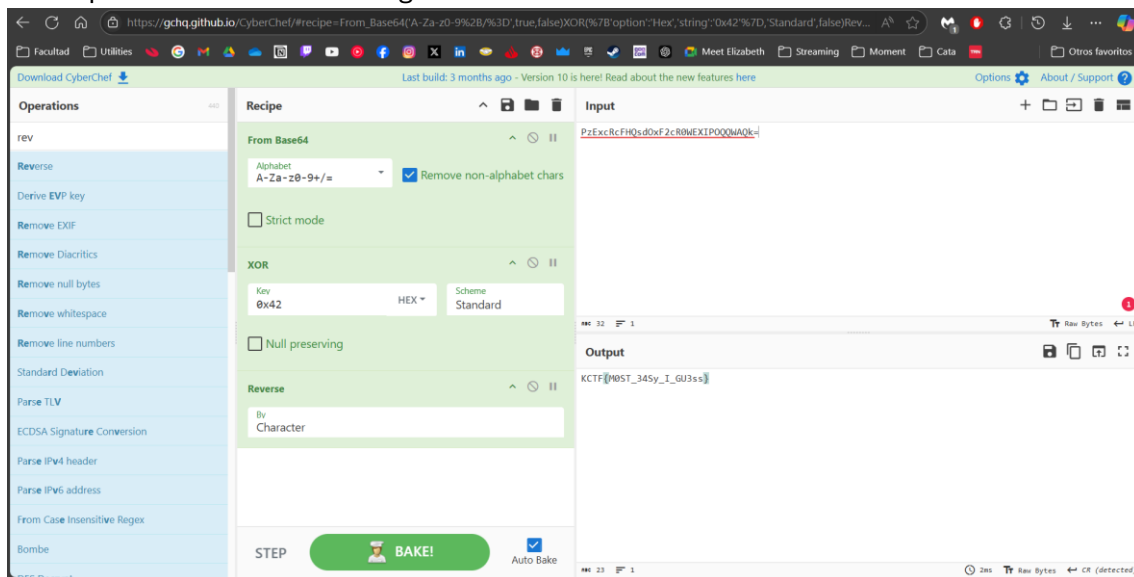
Archivo adjunto:

Para resolver el ejercicio lo primero que hice fue abrir el archive adjunto. En el mismo se encontraban la falg cifrada y la key.

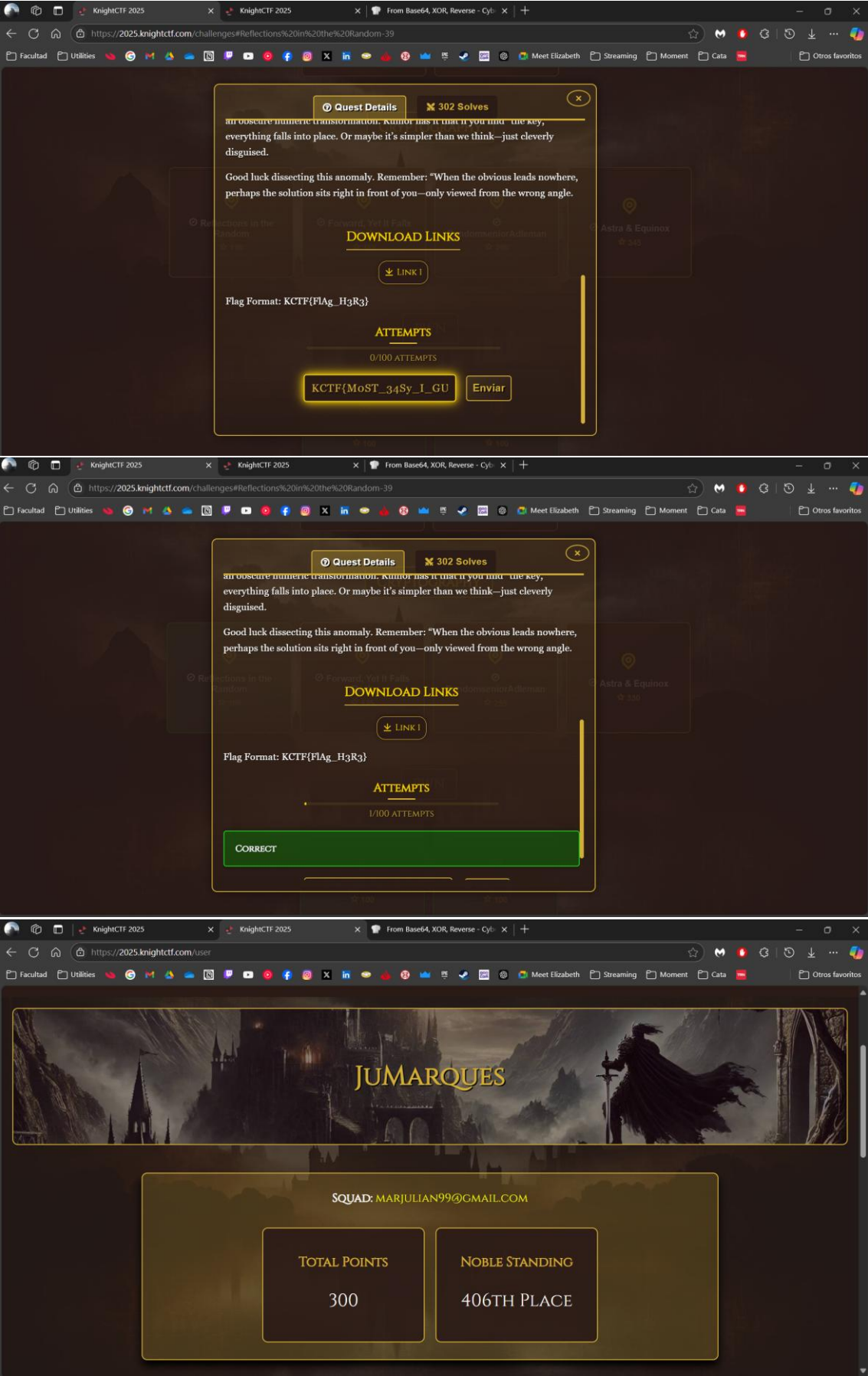


A screenshot of a text editor window titled 'Cipher.txt'. The editor shows two lines of text: 'cipher: PzExcRcFHQsd0xF2cR0WEXIPOQQWQAQk=' and 'key = 0x42'. The status bar at the bottom indicates 'Ln 2, Col 11', '53 caracteres.', '100%', 'Windows (CRLF)', and 'UTF-8'.

A continuacion, desglosando lo que la descripcion del ejercicio decia, y utilizando cyberchef para hacerlo, decidi decodificar el archive en base64. Luego aplique un XOR siguiendo la pista del enunciado de la clave y por ultimo, “spun backward from a single pivot,” esto me indico que había que dar vuelta el resultado final. Asi logre obtener la flag correspondiente a este challenge



Capturas evidencia de resolucion:



Conclusiones

Al realizar estos desafíos, pude ver como diversos conceptos de la materia como el de PWN y criptografía, podían aplicarse de maneras diferentes a las que vimos en la materia. Si bien solo subo estos ejercicios, también intente hacer algunos de reversing de los que presentaba el CTF, la mayoría sin éxito, pero encontrando varios conceptos interesantes como el uso de programación paralela para hacer el procesado de cifrados y ver también que se podía aplicar el concepto de ingeniería inversa en otros tipos de archivos diferentes de los ejecutables de Linux y Windows. También pude apreciar lo que es participar de un CTF, aunque estaría bueno participar también en el futuro en algún CTF con un equipo para poder vivir la experiencia. Dato aparte para destacar también es que estos CTF resultan divertidos cuando uno se deja llevar por la imaginación al participar de escenarios de fantasía como es el caso puntual de este CTF de KnightsCTF2025