Logotipo

Descrição gerada automaticamente com confiança baixa

Relatório de CTF

W1seguy – TryHackMe

|  |  |
| --- | --- |
| **Informações do documento** | |
| **Referência** | W1seguy – Guilherme Gonsales de Sá |
| **N° Revisão** | 1 |
| **Data de publicação** | 11/05/2025 |
| **Link** | <https://tryhackme.com/room/w1seguy> |

|  |  |  |
| --- | --- | --- |
| **Redação** | Guilherme Gonsales de Sá | Estudante |
| **Revisão** | Nome do revisor | Orientador |
| **Aprovação** | Nome do aprovador | Diretor |

|  |  |  |
| --- | --- | --- |
| **Histórico de revisões** | | |
| **N°** | **Entregas** | **Descrição** |
| **0** | 11/05/2025 | Produção |
| **1** | DD/MM/AAAA | Revisão |
| **2** | DD/MM/AAAA | Aprovação |

|  |  |
| --- | --- |
| **Informações do CTF** | |
| **Nível de Dificuldade** | Fácil |
| **Tipo de acesso** | Gratuito |
| **Conceitos envolvidos** | Criptografia |
| **Plataforma** | Tryhackme |
| **Área** | Red |

**Sumário**

[Contextualização 3](#_heading=h.gjdgxs)

[Desenvolvimento 3](#_heading=h.1fob9te)

[What is the first flag? 5](#_heading=h.3znysh7)

[What is the second flag? 7](#_heading=h.2et92p0)

[Conclusão 8](#_heading=h.1t3h5sf)

[Scripts 8](#_heading=h.1t3h5sf)

[Referências 9](#_heading=h.4d34og8)

**Contextualização**

O ctf w1seguy é introdutório na área de criptografia, e se relaciona com o conceito de criptografia de cifras XOR. Embora introdutório, ele demonstra as vulnerabilidades desse tipo de criptografia e a facilidade com que podemos quebra-la.

**Desenvolvimento**

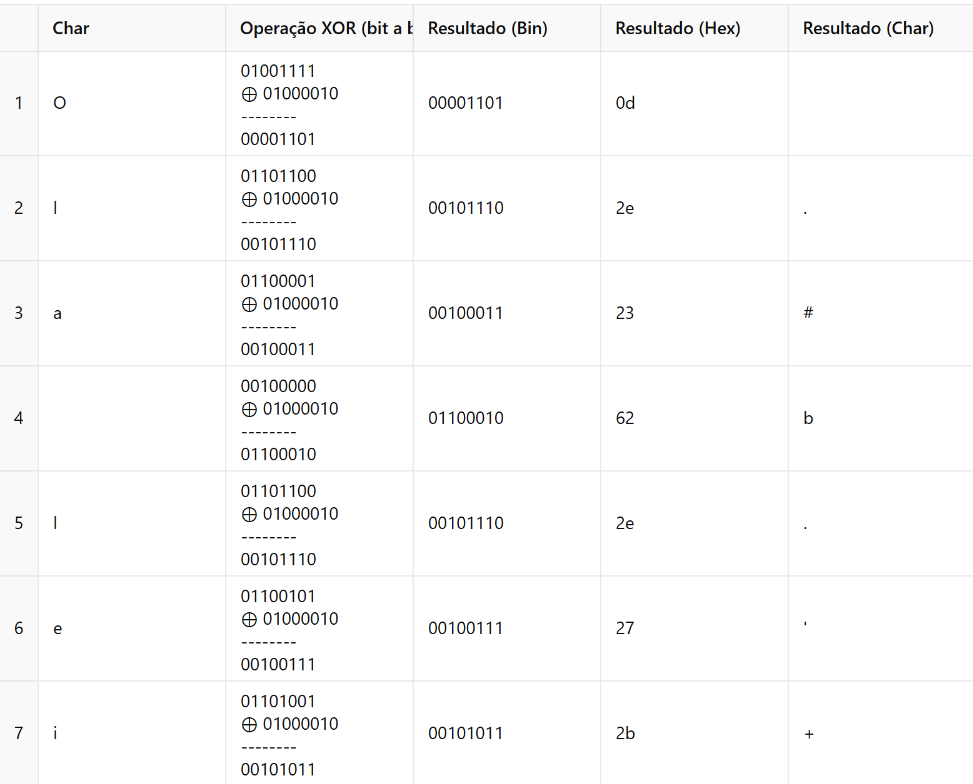
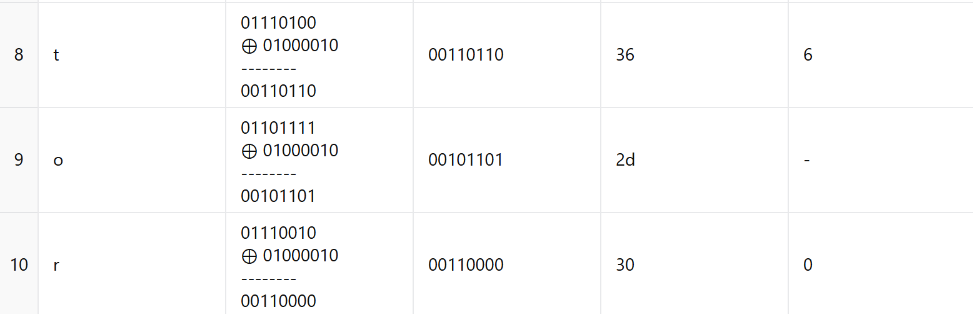
A criptografia utilizando cifra XOR é um tipo de criptografia simétrica, ou seja, usa-se a mesma chave/padrão tanto para encriptar como para decriptar uma mesma mensagem. Ela se baseia na operação lógica da porta XOR (Exclusive OR) e possui a seguinte tabela verdade (combinações de 0´s e 1´s com a porta e seus respectivos resultados): 

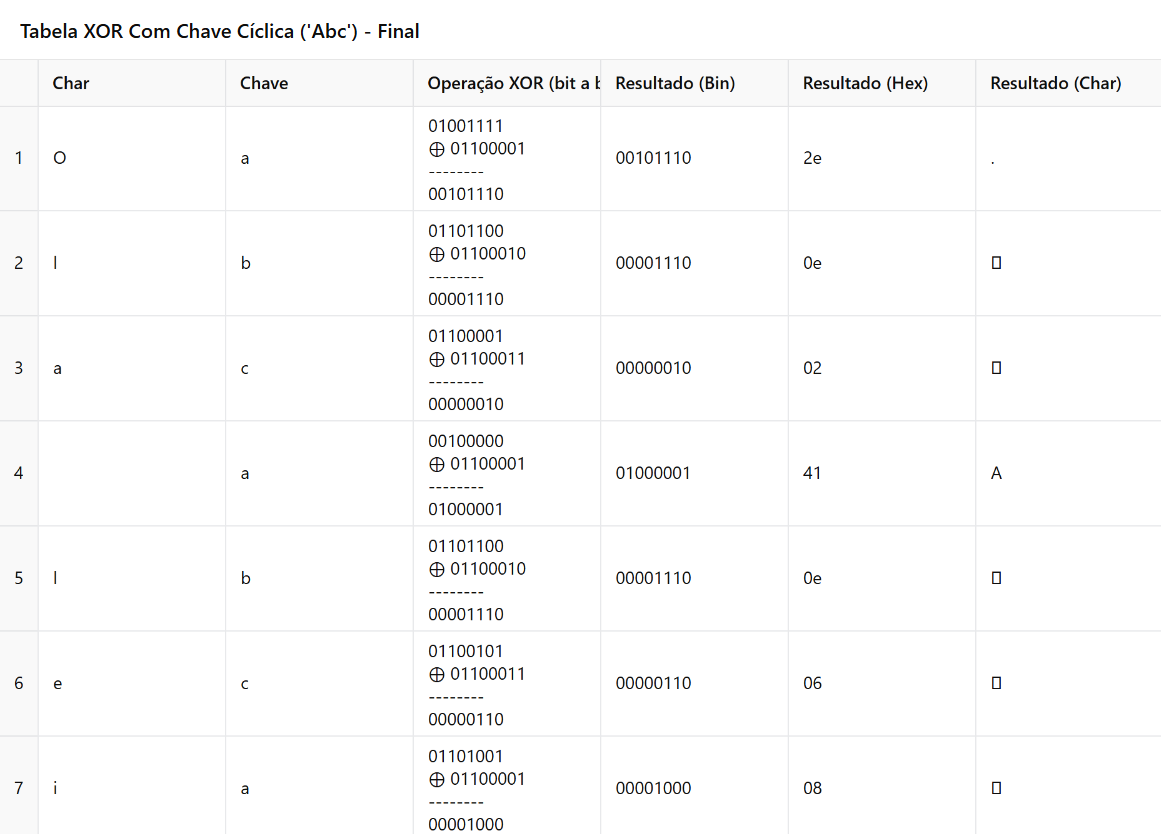
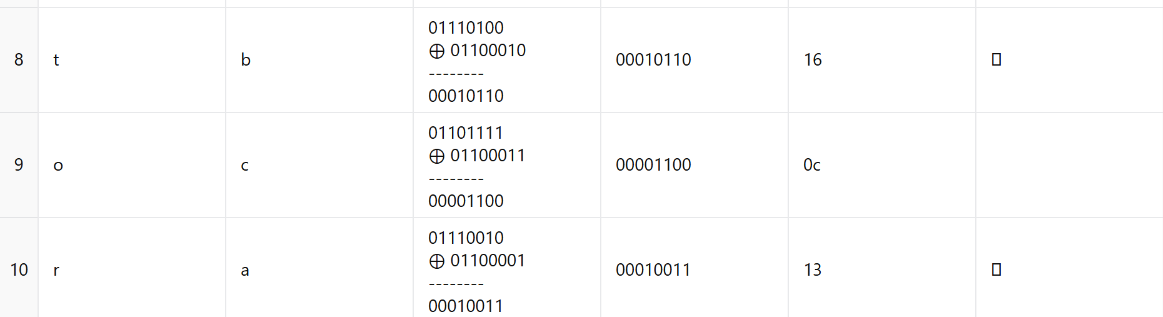
Isso funciona como método de criptografia devido ao fato de todos os caracteres presentes em um sistema possuírem um equivalente binário (0´s e 1´s), possibilitando assim a aplicação dessa e de outras portas lógicas para criptografia.

Existem dois tipos de cifras XOR: com byte fixo e com chave. Com o byte fixo, escolhe-se um byte para fazer todas as operações XOR. Já com a chave, os caracteres a serem encriptados e os caracteres da chave são XORed (sofrem a operação XOR) de forma cíclica.

Para exemplificar, digamos que temos a frase ‘Ola leitor’. Segue um diagrama de como é sua encriptação com cifra XOR, com um byte fixo e com uma chave.

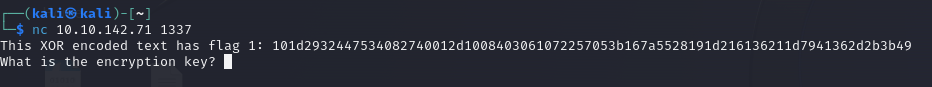


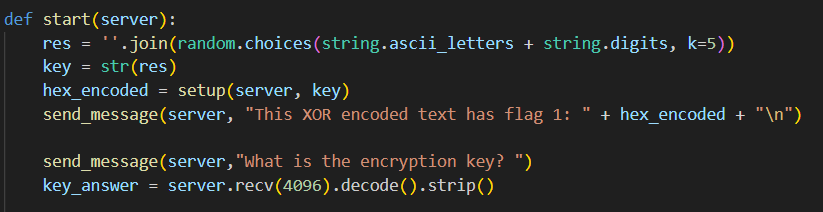
 

**What is the first flag?**

Ao nos conectarmos ao servidor via *netcat,* recebemos uma mensagem criptografada via cifra XOR com chave.

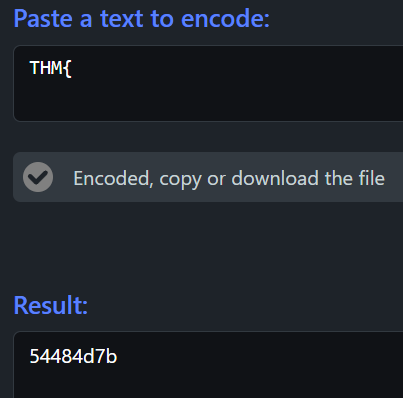
****

Embora saibamos que temos uma chave, não sabemos seu comprimento e, embora seja possível realizar um ataque de força bruta (*bruteforce*) para tal, precisaríamos ter algum indício de como é o formato dessa chave. Para isso, analisemos o código fonte fornecido. Nele, temos:

****

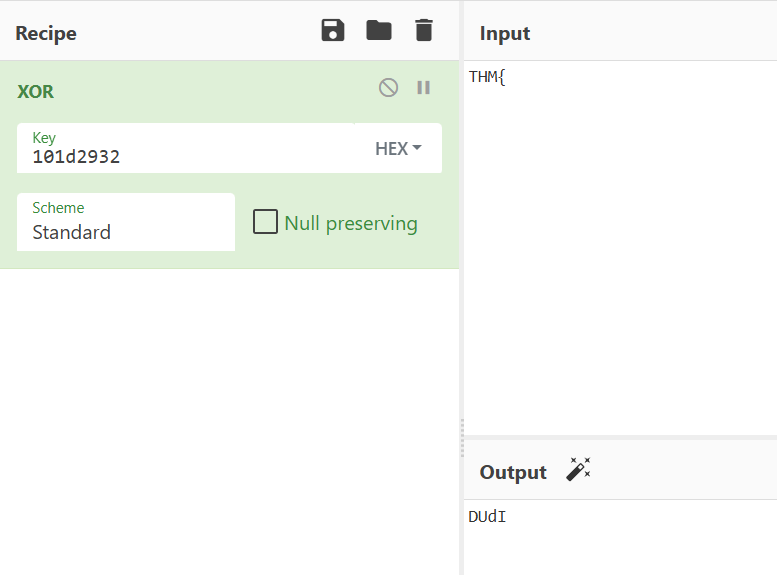
Descobrimos o tamanho da chave: 5 caracteres. Sabemos que a mesma está limitada a apenas letras e dígitos. Além disso, é sabido que todas as flags do site *TryHackMe* começam com a mesma estrutura: *THM{* .

Assim, convertamos esse início de flag para hexadecimal e observemos o resultado.

****

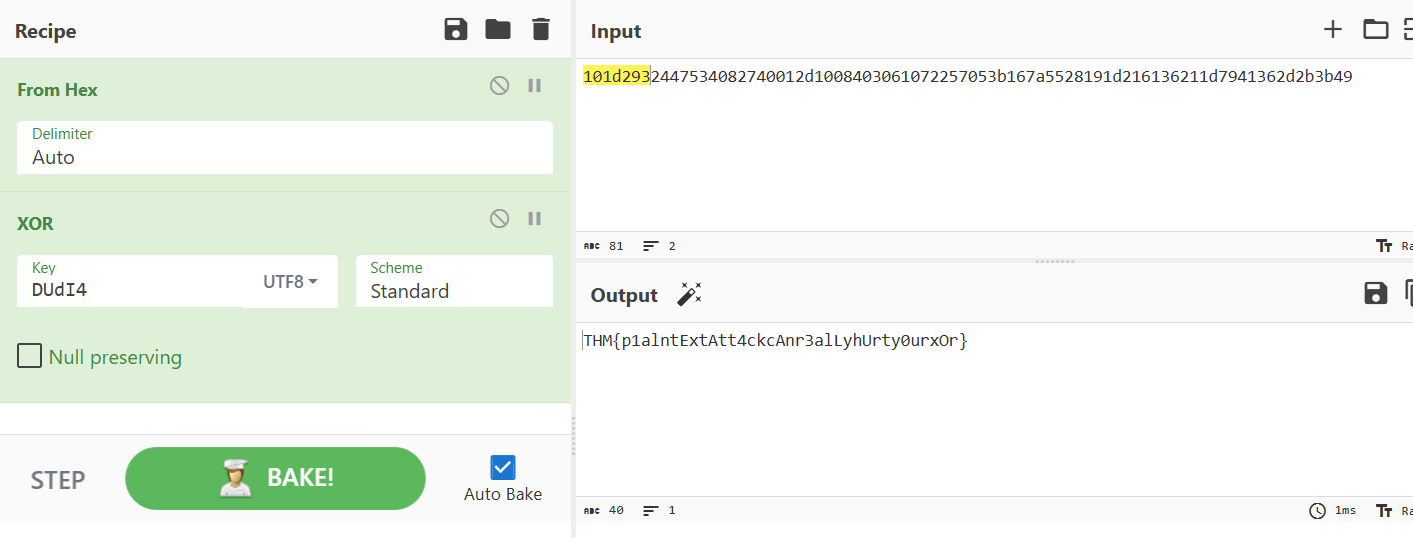
Agora que sabemos a representação dos caracteres em hexadecimal (cada um com 8 bit de comprimento/dois caracteres em hexadecimal) e, sabendo que a encriptação com cifra XOR não altera o tamanho da mensagem, podemos pegar os primeiros 8 caracteres da mensagem encriptada e derivar os 4 primeiros caracteres da chave. Para tal, pode-se fazer um script de *bruteforce.* No entanto, utilizamos do site *CybreChef,* ferramenta que faz esse processo para nós.

Sabendo que estamos trabalhando com uma criptografia simétrica, se pegarmos os caracteres encriptados e coloca-los como chave para a operação XOR, conseguiremos a chave original da encriptação. Dessa forma, tem-se:

****

Agora, possui-se duas opções: Testar manualmente (visto que são apenas 62 possibilidade no último caractere) ou automatizar o processo com um script.

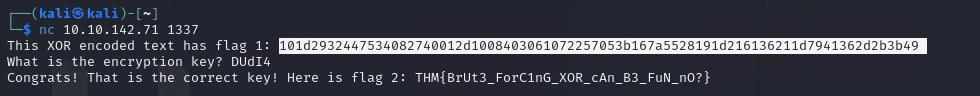
Variando de forma manual, obtivemos o seguinte resultado:

****

Com isso, descobrimos nossa primeira flag: *THM{p1aInTExtAtt4ckcAnr3a11yHurty0urXOr}*

**What is the second flag?**

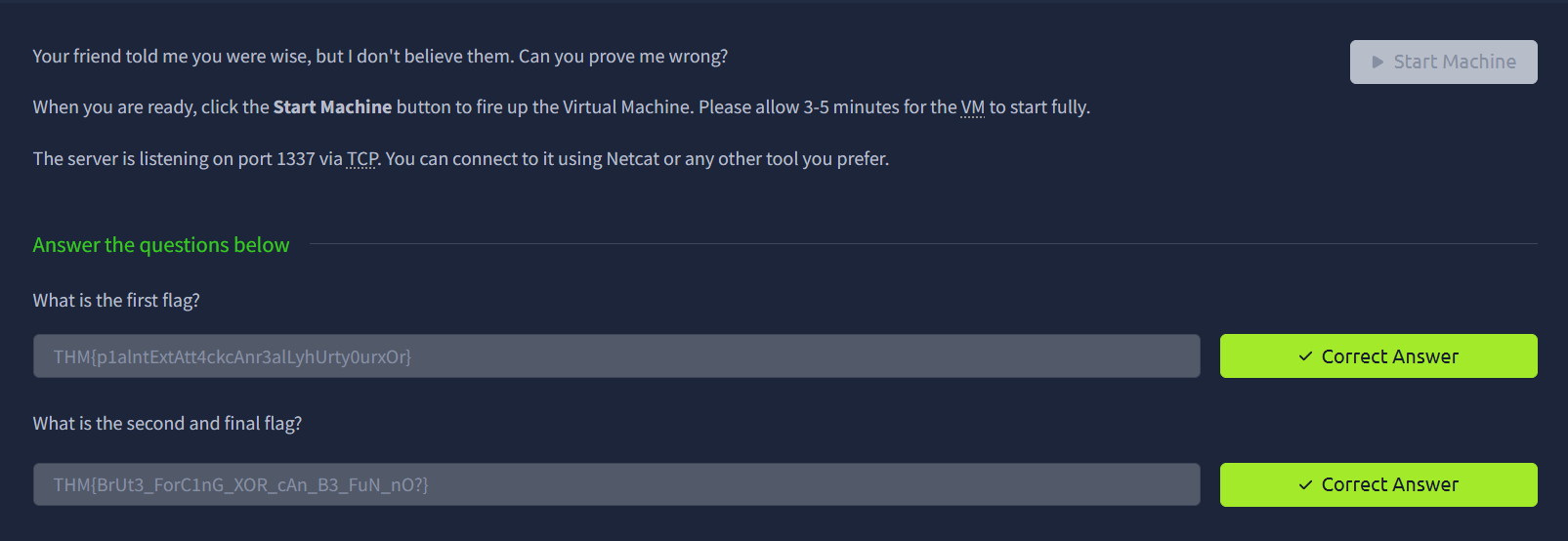
Com a primeira flag em mãos e a chave utilizada para criptografa-la, coloquemos a flag como resposta da pergunta que o servidor nos fez. No caso, a chave utilizada foi *DUdI4.* Atente-se ao fato que a cada vez que nos conectarmos ao servidor, a chave e, portanto, a mensagem encriptada, será diferente.

****

****

Dessa forma, encontramos a segunda flag: *THM{BrUt3\_ForC1nG\_XOR\_cAn\_B3\_FuN\_nO?}*

Com isso, concluímos as duas tasks postas e, consequentemente, o exercício.

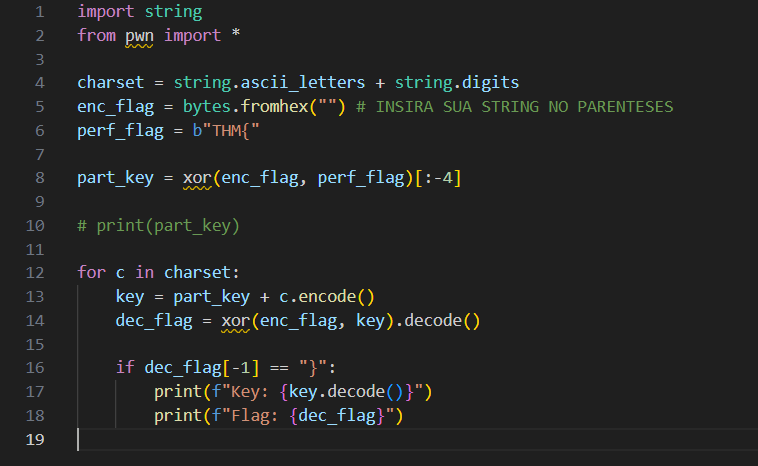
****

**Conclusão**

Embora a parte prática do exercício não seja complicada, o conceito de uma cifra XOR é essencial a todos que buscam conhecer mais sobre a área de criptografia, visto que a mesma é parte integral de cifras mais complexas. Sendo assim, é um exercício excelente para solidificar os conceitos importantes que serão usados posteriormente, como na análise de cifras AES, que usam desse conceito como parte da encriptação.

**Scripts**

Decriptação de operação XOR com chave de 5 dígitos:

****

**Referências**

<https://cyberchef.org/#input=JDYkemRrMC5qVW0keXZhMjRjR00xZHVKa3dNNWIxN1EyMDV4REo0N0xPQWcvT3BadkoxZ0tiTEY4UEpCRGtKQTRhNk0uSllQVVRBYVd1NGluZkRqSTg4VTl5VVhFVmdMLgo>

<https://en.wikipedia.org/wiki/XOR_cipher>

Tabelas – feitas com o auxílio da IA ChatGPT. <https://chatgpt.com/>