

Simple XOR & 46esab CTF Writeup

This document is a walkthrough on one way to solve the **Simple XOR & 46esab** CTFs on **TheBlackSide**. The goal is, not only to give the solution, but also to emphasise the importance of thorough reading of the challenge's title and description.

General Information

- *Difficulty:* **Easy**
- *Category:* **Cryptography**
- *Link:* [Simple XOR - TheBlackSide](#) and [46esab - TheBlackSide](#)

Solution

Simple XOR

Une donnée secrète a été xorée avec une clef de taille 1 octet. Retrouvez la donnée en clair !
Fj96PDY7PXo/KS56YHoOGAkhEj82NjV3AjUodxgjPyc=

☆ 15

comores11

03/08/2021

✓ 305

Cryptographie

Express

Flag

Valider

"A secret piece of data has been XOR'd with a 1-byte key. Find the data in plain text!"

We have a 1-byte key (ex: 00100010), which we don't know, but we know that there are 256 possibilities, and it looks like the text *"Fj96PDY7PXo/KS56YHoOGAkhEj82NjV3AjUodxgjPyc="*, has been XOR'd. Now, the XOR operator is reversible, meaning that with the same key we can get the secret message.

However, brute-forcing the XOR operation on all 256 keys on the text will not give us the flag. What we have to do is carefully read the challenge's description; *"a **SECRET** piece of data has been XOR'd"* This suggests that the text underwent a second encryption process, likely using a Base64 algorithm.

Recipe	Input
<p>From Base64</p> <p>Alphabet: A-Za-z0-9+/= <input checked="" type="checkbox"/> Remove non-alphabet chars</p> <p><input type="checkbox"/> Strict mode</p> <p>XOR Brute Force</p> <p>Key length: 1 Sample length: 100 Sample offset: 0</p> <p>Scheme: Standard <input type="checkbox"/> Null preserving <input checked="" type="checkbox"/> Print key</p> <p><input type="checkbox"/> Output as hex Crib (known plaintext string)</p> <p>STEP BAKE! <input checked="" type="checkbox"/> Auto Bake</p>	<p>Fj96PDY7PXo/K5S6YHoOGAkH Ej82NjV3A jUodxgjPyc=</p> <p>Output</p> <pre> Key = 54: BK.hbo1.kjz.4.ZLjuKbba#Va Lwks Key = 55: Cj/i cnh/j {/5/[M\tGjcc"~M"}~Mvjrn Key = 56: @i,j"mk,i*x,6,XN_w0i"~c!Tc~!Nuiq Key = 57: Ah~ka1j~h~y~7~Y0^vEhaab Ub•Othp Key = 58: Ng"dnce"gv"8"~V@QyJgnnm/Zmp/@[g• Key = 59: Of#eobd#fpw#9#WAPxKfool.[1q.Azf~ Key = 5a: Le flag est : TBS{Hello-Xor-Bye} Key = 5b: Md!gm"fldru!;!UCRzIdmm,Yns,Cxd Key = 5c: Jc&"jga&cur&&RDU Ncjji+^It+D+c{ Key = 5d: Kb"akf"bts"~SET 0bkkh*_hu"E~bz Key = 5e: Ha\$bhec\$awp\$>\$PFW•Lahhk)\kv)F)ay Key = 5f: I"%cidb%"vq%?%QGV~M"iij()jw(G `x Key = 60: v_sue\V[]sue_INsuevuesfxiAr_VVUerebUHetsXC_G Key = 61: w^esc]WZ\esc^HOescowescOyh@s^MWTaveCTIenvB^F Key = 62: +1...ATV...1K1...1kCa1TtM...M7...A1E </pre>

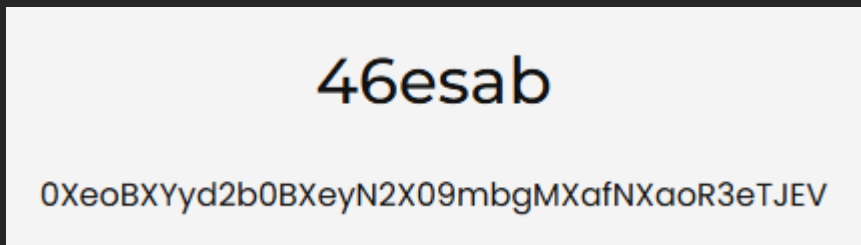
Using Cyberchef, an online cryptographic tool, we get the following flag: **Hello-Xor-Bye**

Friendly Reminder

READ THE CHALLENGE TITLE & DESCRIPTION CAREFULLY !!!

Although this challenge was pretty simple, it serves as a prime example of how **hidden details** in the title/description can be pivotal to finding the solution.

Let's give another example. Here's another challenge on **TheBlackSide**:



Looking closely, **"46esab"** is **"base64"** in reverse

This suggests that the flag was most likely reversed after being encrypted using the base64 algorithm. That being said, the title tells us all we need to know to complete the challenge.

Recipe	Input
<p>Reverse</p> <p>By: Character</p> <p>From Base64</p> <p>Alphabet: A-Za-z0-9+/= <input checked="" type="checkbox"/> Remove non-alphabet chars</p> <p><input type="checkbox"/> Strict mode</p>	<p>0XeobXYYd2b0BXeyN2X09mbgMXafNXaor3eTJEV</p> <p>Output</p> <p>TBS{this_is_not_cryptography}</p>

We get the following flag: **TBS{this_is_not_cryptography}**