ᛀ **main** ⌄   **ctf-writeup** / **2023-ARA** / **Cryptography-One-Time-Password** /   ···

👤 **masnurrm** feat: OTP ···   1 hour ago   🕘 **History**

..

📄 readme.md   1 hour ago

---

readme.md   ✏️

Cryptography - One Time Password (?)

In this challenge, we were given `txt` file with some line of string hexadecimal written in it.

```
A: 161a1812647a765b37207a1c3b1a7b54773c2b660c46643a1a50662b3b3e42
B: 151d616075737f322e2d130b381666547d3d4470054660287f33663d2a2e32

XOR: 415241323032337b7468335f705f3574346e64355f6630725f7034647a7a7d
```

We try to approach it by applying the XOR operation between the given hexadecimal A and B. But, first we need to convert A and B into binary. Thanks to binaryhackconverter.com.

```
A:
000101100000110001100000100010110010001110100110101110110010110110011011100100000011101000110000
B:
000101010001110101100001011100000111001011010101011111110011001010111101011111010000110110001000
```

Then, try to apply the XOR operation between A and B so it will give us the result here.

```
00000011000001011110010001100110001010000111110110110110110110111111101000110100010100111001
```

Convert the XOR result into binary.

```
XOR:
010000010101001000110000001100100001101110011001000110011011101110011110011110001110100001101000
```

Then, apply the XOR operation between A and B result and the XOR string. It will give us this result.

415241323032337b7468335f705f3574346e64355f6630725f7034647a7a7d

Lastly, convert it into ASCII characters.

So, this is the flag.

ARA2023{th3_p_5t4nd5_f0r_p4dzz}