Good authentication

Description: After your first strike, the development team has increased the power of there login function. Are you strong enough to break it again?
Level: Level 1
Type: Coding
Link: *https://ctf.securityvalley.org/dashboard*

So we got an authentification js code,a bit harder thant the 1st one:

```
const readline = require('readline').createInterface({

input: process.stdin,

output: process.stdout

});

readline.question('Please enter password \n', password => {

console.log(`Gonna check if ${password} is correct`);

readline.close();

validate(password)

});

function validate(password) {

if (password.length != 12) {

throw new Error("pass violation. wrong password length");
```

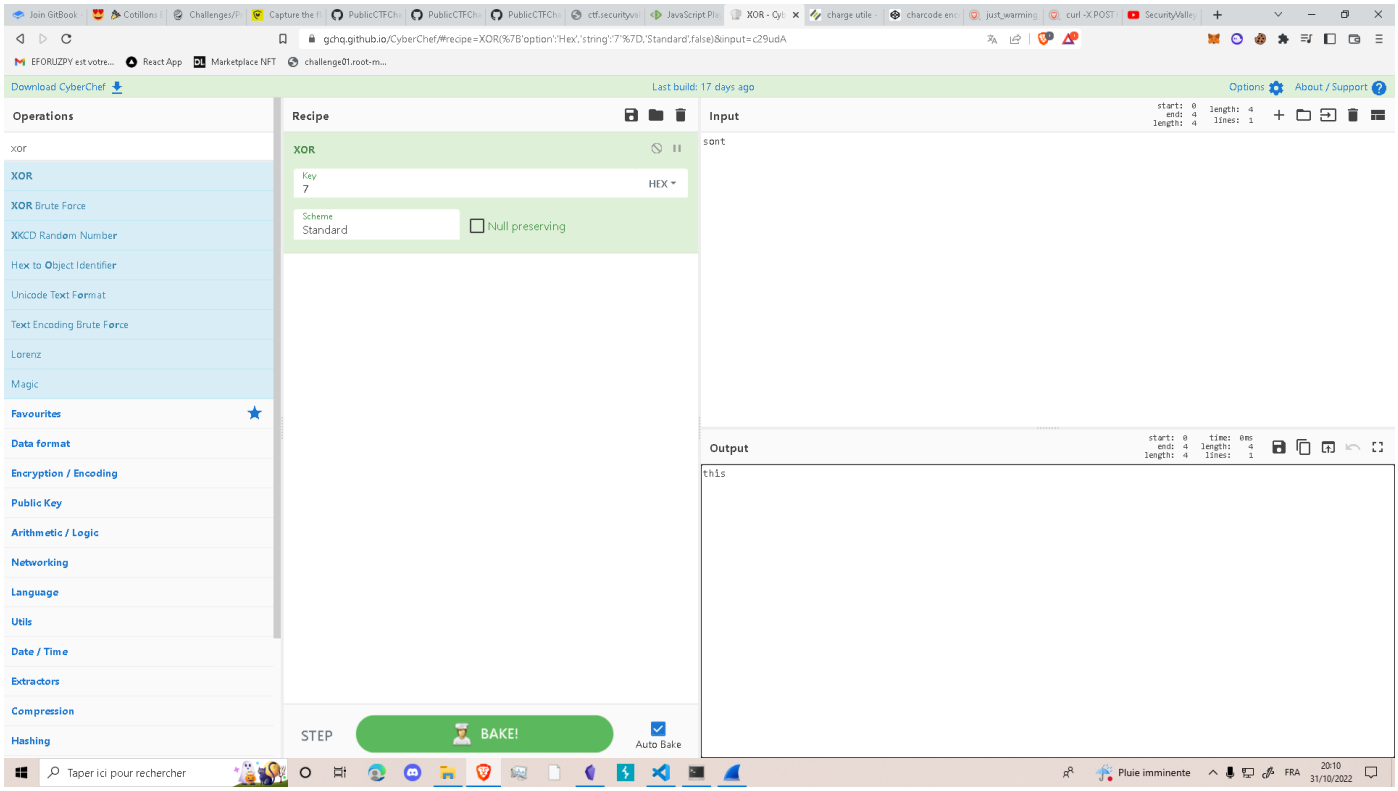
```
}

const block1 = Array.from(password).slice(0, 4)

const block2 = Array.from(password).slice(4, 8)

const block3 = Array.from(password).slice(8, 12)

const block = [

block1,

block2,

block3

]

let crafted = "";

for (let i = 0; i < block.length; i++) {

for (let a = 0; a < block[i].length; a++) {

if (i == 0) {

crafted += String.fromCharCode(String(block[i][a]).charCodeAt(0) ^ 7)

} else if (i == 1) {

crafted += String.fromCharCode(String(block[i][a]).charCodeAt(0) ^ 11)

} else {

crafted += String.fromCharCode(String(block[i][a]).charCodeAt(0) ^ 9)

}

}

}

if(crafted !== "sontTbxTjffe") {
```

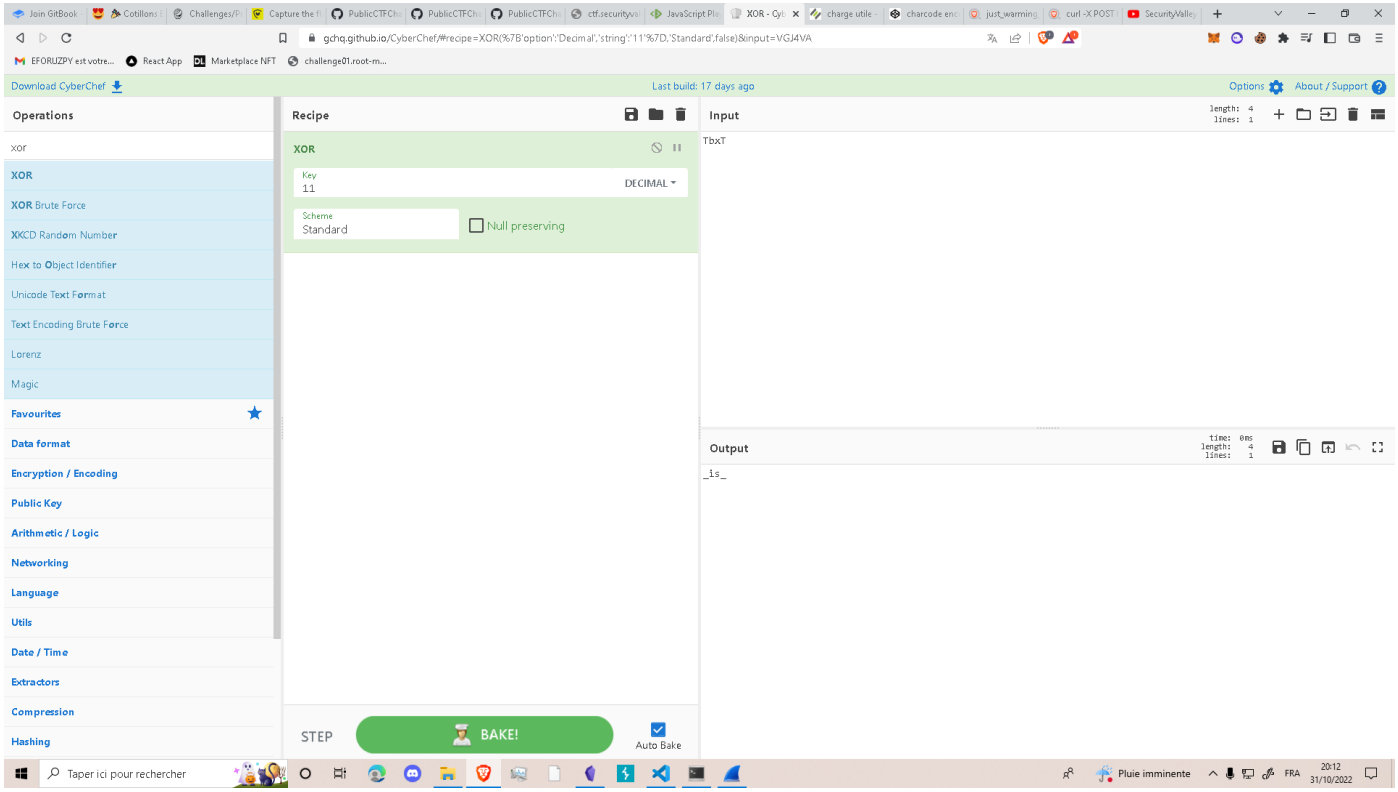
```
throw new Error("pass violation. wrong credentials");

}

banner(password);

}

function banner(payload) {

console.info("that was great !!!");

console.info("run the following command to get the flag.")

console.info(`curl -X POST http://ctf.securityvalley.org:7777/api/v1/validate -H 'Content-Type: application/json' -d '{"pass": "${payload}"}'`)

}
```

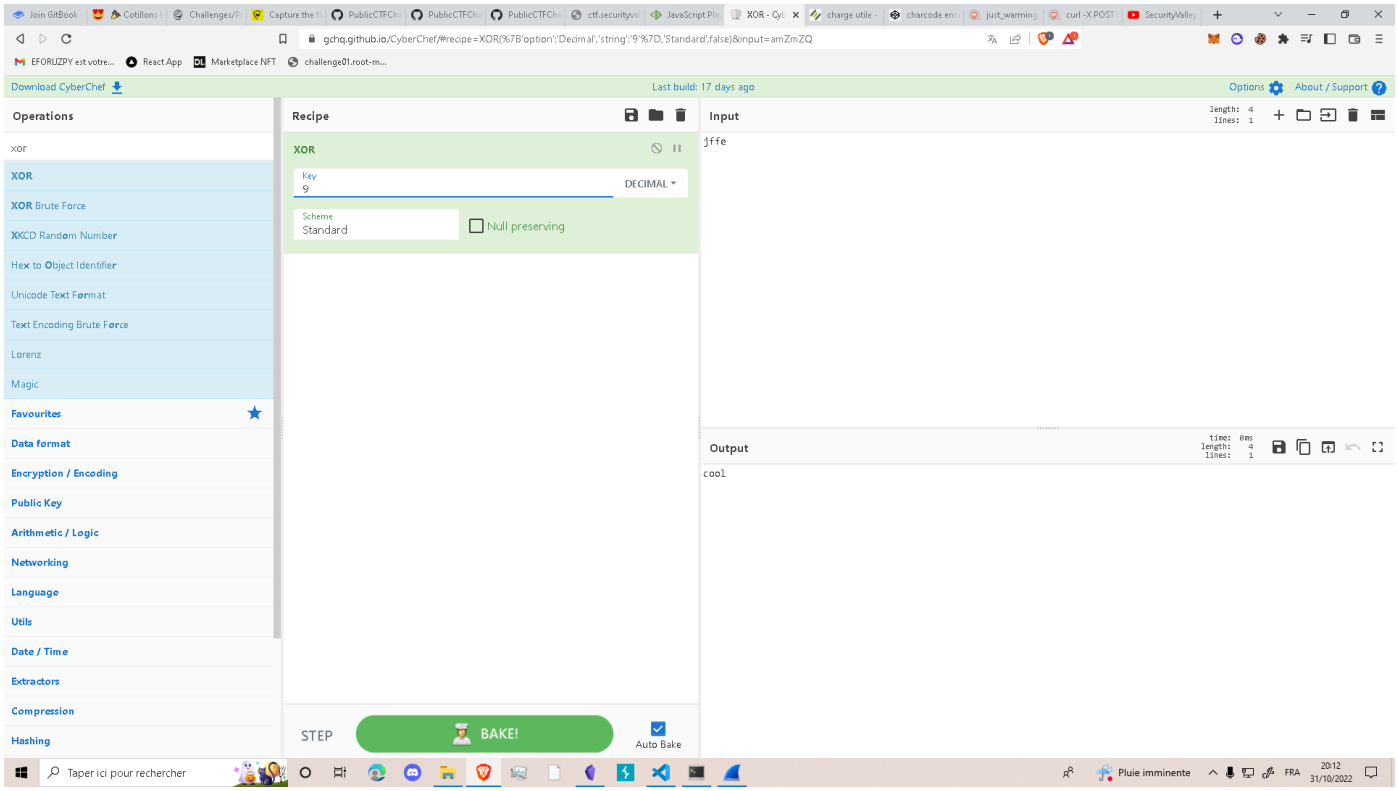
At the end,we can see that the "final pass" need to be `sontTbxTjffe`.The pass is in 3 part:

```
sont ^ 7
TbxT ^ 11
jffe ^ 9
```

This is **Xor** (algo) so lets decode with cyberchef:

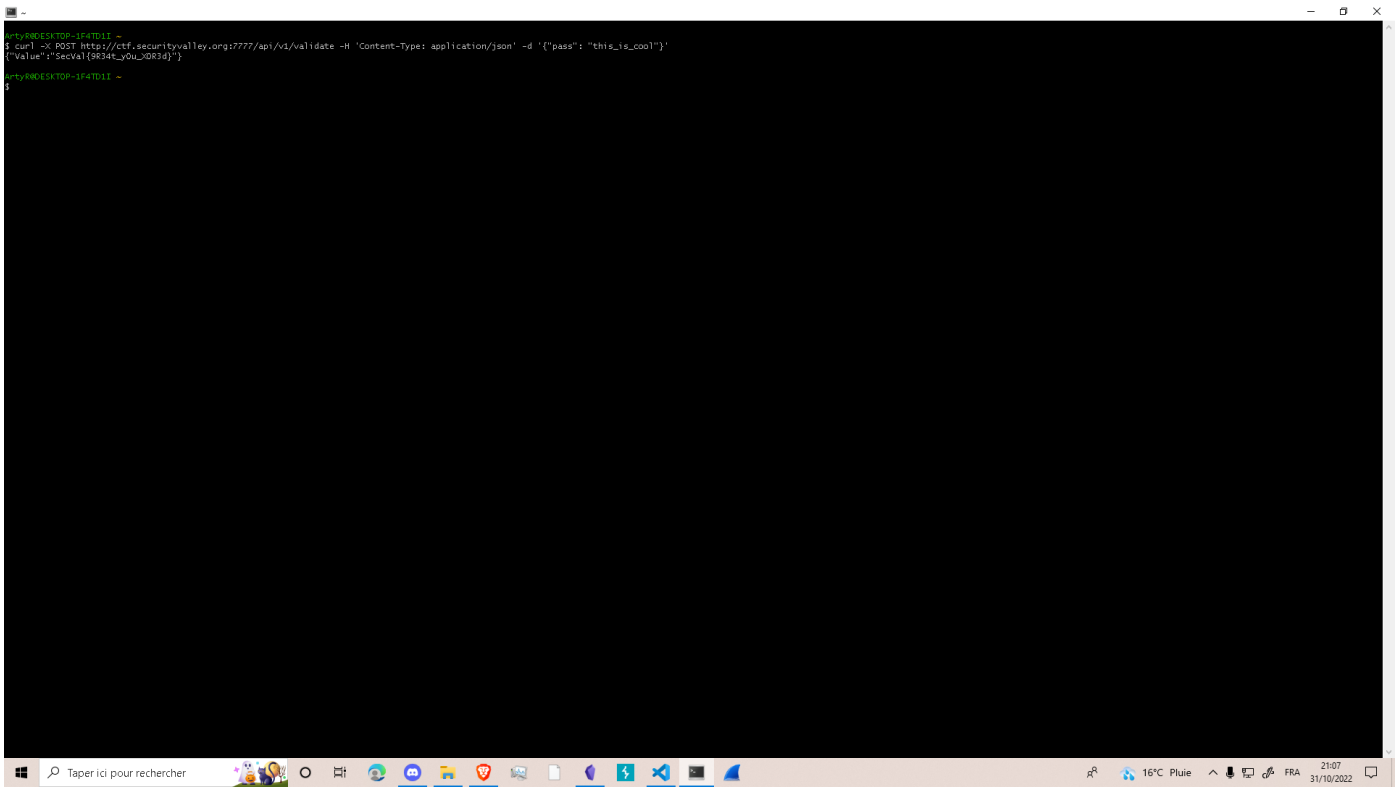

gchq.github.io/CyberChef/#recipe=XOR(%7B'option':'Hex','string':'7'%7D,'Standard',false)&input=c29udA
Download CyberChef
Last build: 17 days ago
Options
About / Support
Operations
xor
XOR
XOR Brute Force
XKCD Random Number
Hex to Object Identifier
Unicode Text Format
Text Encoding Brute Force
Lorenz
Magic
Favourites
Data format
Encryption / Encoding
Public Key
Arithmetic / Logic
Networking
Language
Utils
Date / Time
Extractors
Compression
Hashing
Recipe
XOR
Key
7
HEX
Scheme
Standard
Null preserving
STEP
BAKE!
Auto Bake
Input
sont
Output
this
Taper ici pour rechercher
Pluie imminente
FRA
20:10
31/10/2022


gchq.github.io/CyberChef/#recipe=XOR(%7B'option':'Decimal','string':'11'%7D,'Standard',false)&input=VGJ4VA
Download CyberChef
Last build: 17 days ago
Options
About / Support
Operations
xor
XOR
XOR Brute Force
XKCD Random Number
Hex to Object Identifier
Unicode Text Format
Text Encoding Brute Force
Lorenz
Magic
Favourites
Data format
Encryption / Encoding
Public Key
Arithmetic / Logic
Networking
Language
Utils
Date / Time
Extractors
Compression
Hashing
Recipe
XOR
Key
11
DECIMAL
Scheme
Standard
Null preserving
STEP
BAKE!
Auto Bake
Input
TbxT
Output
_is_
Taper ici pour rechercher
Pluie imminente
FRA
20:12
31/10/2022




```
sont = this
TbxT = _is_
jffe = cool
```

So the password is `this_is_cool`,so we need to replace the `${payload}` with `this_is_cool`, so we enter: `curl -X POST http://ctf.securityvalley.org:7777/api/v1/validate -H 'Content-Type: application/json' -d '{"pass": "this_is_cool"}'`



The flag appear !

The flag is: **SecVal{9R34t_y0u_X0R3d}**