

masnurrm feat: sh432 7 minutes ago History

- assets 7 minutes ago
- readme.md 7 minutes ago
- sha256.py 7 minutes ago

readme.md

Cryptography - SH4-32

In this challenge, we were given a hash value as follows.

9be9f4182c157b8d77f97d3b20f68ed6b8533175831837c761e759c44f6feeb8

Using the hash analyzer from tunnelsup.com, we know that this is a SHA2-256 type of hash value.

Hash:	9be9f4182c157b8d77f97d3b20f68ed6b8533175831837c761e759c44f6feeb8
Salt:	Not Found
Hash type:	SHA2-256
Bit length:	256
Character length:	64
Character type:	hexidecimal

Also, we were given an attachment named Dictionary.txt that contains many lines of random strings.

123456
123456789
111111
password
qwerty

```
abc123
12345678
password1
...
...
...
```

As we know, the SHA2-256 type of hash can't be decrypted. So we try to encrypt the all of strings in the `Dictionary.txt` to find which string that have the encrypted string of SHA2-256 that same as the hash provided in the challenge. We use this script.

```
import hashlib

with open('Dictionary.txt', 'r') as f:

    for line in f:
        line = line.strip()

        sha256_hash = hashlib.sha256(line.encode()).hexdigest()
        expected = "9be9f4182c157b8d77f97d3b20f68ed6b8533175831837c761e759c44f6feeb8"

        if sha256_hash == expected:
            print(f"{line} : {sha256_hash}")
```

We found this string.

```
415241323032337b6834736833645f30525f6e4f545f6834736833647d
```

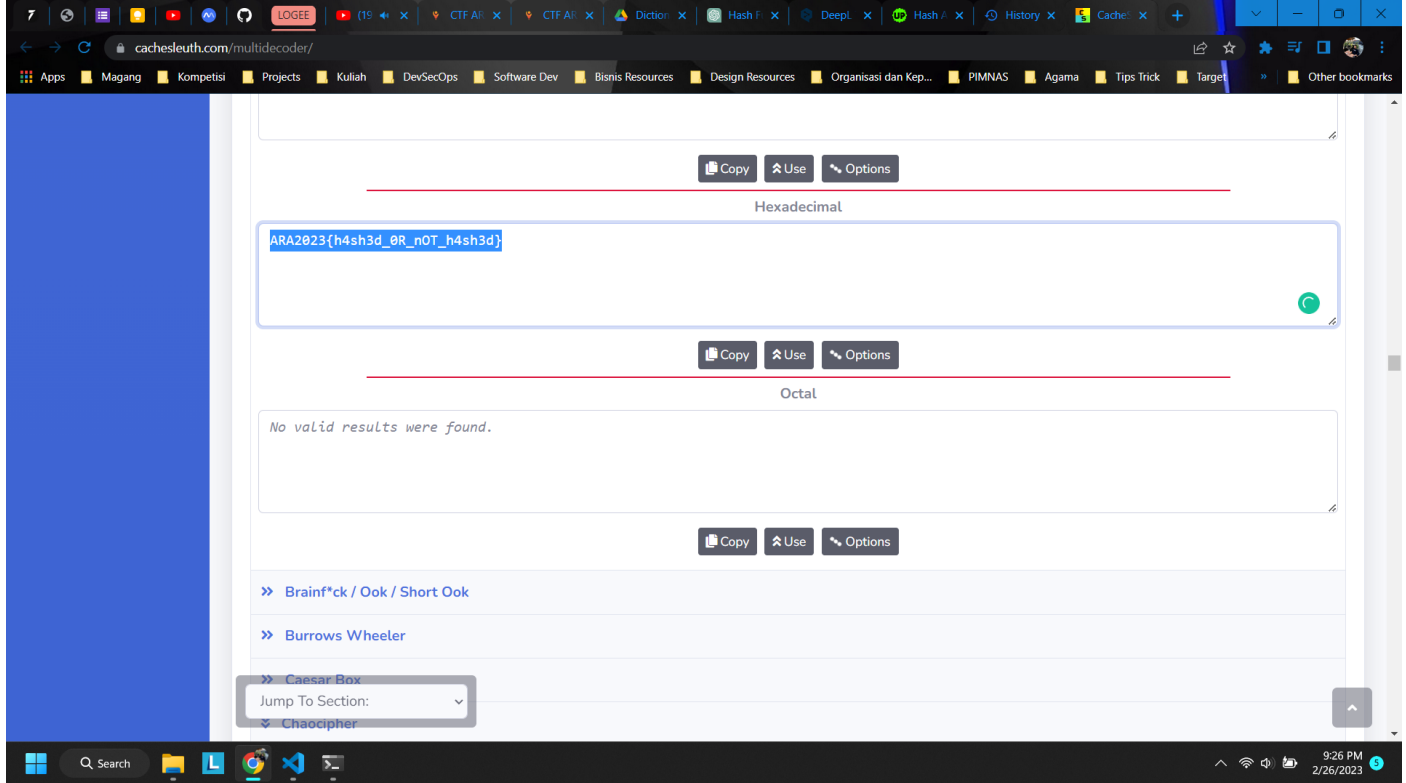
The image shows a screenshot of a code editor with a dark theme. At the top, there's a tab labeled 'sha256.py'. Below it, the script content is visible, matching the code block above. The script opens 'Dictionary.txt', iterates through its lines, calculates the SHA256 hash for each, and compares it to a hardcoded 'expected' hash. A match is found, and the corresponding string and its hash are printed. At the bottom, a terminal window shows the command to run the script: 'python.exe "e:/Kompetisi/ARA CTF 2023/sha256.py"'. The output of the script is displayed in the terminal: '415241323032337b6834736833645f30525f6e4f545f6834736833647d : 9be9f4182c157b8d77f97d3b20f68ed6b8533175831837c761e759c44f6feeb8'.

```
sha256.py X
sha256.py > ...
2
3   with open('Dictionary.txt', 'r') as f:
4
5       for line in f:
6           line = line.strip()
7
8           sha256_hash = hashlib.sha256(line.encode()).hexdigest()
9           expected = "9be9f4182c157b8d77f97d3b20f68ed6b8533175831837c761e759c44f6feeb8"
10
11       if sha256_hash == expected:
12           print(f"{line} : {sha256_hash}")

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  GITLENS  SQL CONSOLE  COMMENTS

PS E:\Kompetisi\ARA CTF 2023> & C:/Users/sangm/AppData/Local/Programs/Python/Python310/python.exe "e:/Kompetisi/ARA CTF 2023/sha256.py"
415241323032337b6834736833645f30525f6e4f545f6834736833647d : 9be9f4182c157b8d77f97d3b20f68ed6b8533175831837c761e759c44f6feeb8
PS E:\Kompetisi\ARA CTF 2023> █
```

Try to convert it into hexadecimal, we got the flag. Thanks to [cachesleuth](#).



So, this is the flag.

```
ARA2023{h4sh3d_0R_n0T_h4sh3d}
```

Give feedback