

Write-Up: Byte-Circus

Player: Constantine

Kategori: Reverse Engineering



Phase 1: Decompilation & Static Analysis

Dapat file `script.cpython-313.pyc`, daripada pusing baca *opcode* manual pakai dis module, aku langsung pakai tools online **pychaos** untuk melakukan decompile .pyc ke .py.

Output Decompiler:

```
pychaos.io/decompiled?uuid=a99c4a77-98e1-4836-91e4-842a5b35a6dc

the_flag = [1423,1432,1403,1422,1404,1405,1410,1444,1436,1437,1450,1437,1458,1452,1416,1450,1463,1450,1461,1442,1436,1458,1436,1416,1458,1437,1416,1458,1436,1416,1450,1455,1372,1369,1452,1453,1446]
def enc(string):
    return [ord(x)^16+1337 for x in string]

def main():
    inp = input('Enter password: ').strip()
    inp_enc = enc(inp)
    if inp_enc == the_flag:
        print(f'Correct! Flag: {inp}')
        return None
    else:
        print('Wrong password')
        return None

if __name__ == '__main__':
    main()
```

Phase 3: Execution

Aku minta **Gemini** untuk generate *solver script* berdasarkan code di atas.

Solver Script (poc.py):

```
def solve():
    # Array dari soal
    the_flag = [
        1423, 1432, 1403, 1422, 1404, 1405, 1410, 1444, 1436, 1437,
        1450, 1437, 1458, 1452, 1416, 1450, 1463, 1450, 1461, 1442,
        1436, 1458, 1436, 1416, 1458, 1437, 1416, 1458, 1436, 1416,
        1450, 1455, 1372, 1369, 1452, 1453, 1446
    ]

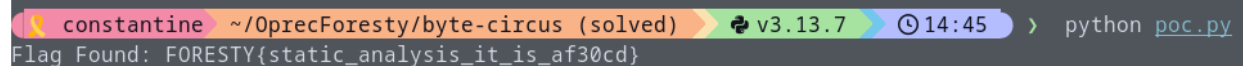
    decoded_flag = ""

    for value in the_flag:
        # 1. Kurangi offset (inverse dari + 1337)
        temp = value - 1337
        # 2. XOR dengan key (inverse dari ^ 16)
        original_ord = temp ^ 16
        # 3. Konversi ke karakter
        decoded_flag += chr(original_ord)

    print(f"Flag Found: {decoded_flag}")

if __name__ == '__main__':
    solve()
```

Execute:



```
constantine ~/.0precForesty/byte-circus (solved) v3.13.7 14:45 > python poc.py
Flag Found: FORESTY{static_analysis_it_is_af30cd}
```

Flag: FORESTY{static_analysis_it_is_af30cd}