Challenge: SPG

Challenge Description :

After successfully joining the academy, there is a process where you have to log in to eclass in order to access notes in each class and get the current updates for the ongoing prank labs. When you attempt to log in, though, your browser crashes, and all your files get encrypted. This is yet another prank for the newcomers. The only thing provided is the password generator script. Can you crack it, unlock your files, and log in to the spooky platform?

Context :

- Analyze this python script and try to figure out the master key to the encryption, The note.txt file contains the Flag and the generated password. Find out a way to decrypt it and get the Flag.

Flag :

- First, install the necessary file and check the provided Python script.
- The process of the main function is as follows:
    - A password is generated using the MASTER_KEY.
      An encryption key is then created by hashing the MASTER_KEY with SHA256. An AES cipher object is initialized using this encryption key in ECB mode.
    - The flag is encrypted with the AES cipher, and the output is written to a file.
    - To get the flag, you will need to modify the script to decrypt the ciphertext using the AES key, thereby revealing the flag.
- The function to generate the password converts the MASTER_KEY into a little-endian integer.
- It then iterates over the bits of this integer, choosing characters from an ALPHABET list based on the bit values. If the bit is 1, a character from the first half of ALPHABET is chosen; if 0, a character from the second half is chosen.
- To decrypt the ciphertext, we need to determine the original MASTER_KEY. This can be achieved by analyzing the generated password.

```python
ALPHABET = string.ascii_letters + string.digits + '~!@#$%^&*'

def generate_password():
    master_key = int.from_bytes(MASTER_KEY, 'little')
    password = ''

    while master_key:
        bit = master_key & 1
        if bit:
            password += random.choice(ALPHABET[:len(ALPHABET)//2])
        else:
            password += random.choice(ALPHABET[len(ALPHABET)//2:])
        master_key >>= 1

    return password

def main():
    password = generate_password()
    encryption_key = sha256(MASTER_KEY).digest()
    cipher = AES.new(encryption_key, AES.MODE_ECB)
    ciphertext = cipher.encrypt(pad(FLAG, 16))

    with open('output.txt', 'w') as f:
        f.write(f'Your Password : {password}\nEncrypted Flag : {b64encode(ciphertext).decode()}')
```

- Modified Version is the one below, the link for the script is at the bottom given with the flag.

```python
def generate_password():
    master_key = int.from_bytes(MASTER_KEY, 'little')
    password = ''

    while master_key:
        bit = master_key & 1
        if bit:
            password += random.choice(ALPHABET[:len(ALPHABET)//2])
        else:
            password += random.choice(ALPHABET[len(ALPHABET)//2:])
        master_key >>= 1

    return password

def crack_password(password):
    master_key = 0

    for i, p in enumerate(password):
        if p in ALPHABET[:len(ALPHABET) // 2]:
            master_key |= 1 << i

    return master_key.to_bytes((7 + len(password)) // 8, 'little')

def main():
    password = 't*!zGnf#LKO~drVQc@n%oFFZyvhvGZq8zbfXKvE1#*R%uh*$M6c$zrxWedrAENFJB7xz0ps4zh94EwZOnVT9&h'
    ciphertext = 'GKLlVVw9uz/QzqKiBPAvdLA+QyRqyctsPJ/tx8Ac2hIUl8/kJaEvHthHUuwFDRCs'
    MASTER_KEY = crack_password(password)
    encryption_key = sha256(MASTER_KEY).digest()
    cipher = AES.new(encryption_key, AES.MODE_ECB)

    decrypted_flag = unpad(cipher.decrypt(b64decode(ciphertext)), AES.block_size).decode()
    print("Decrypted Flag:", decrypted_flag)

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

Script {Link}

- The Flag for this challenge is : HTB{m4ll34bl3_p4ssw0rd_g3n3r4t0r!}