Network Security
Project 2

0556562 陳鴻君

my port = 140.113.194.78:20025
all hints are in robots.txt

Step 1: get temporary file from website

We know Bob is a vim enthusiast, so the temporary file is “.memorandum.txt.swp”.
Download the file as expect.

Step 2: base64 decode

In “decode.py”, use python base64 library to decode “memorandum.txt.swp”.
The output is piped into “base64_decode_result”.

Step 3: XOR Cracker

(https://wiremask.eu/tools/xor-cracker)

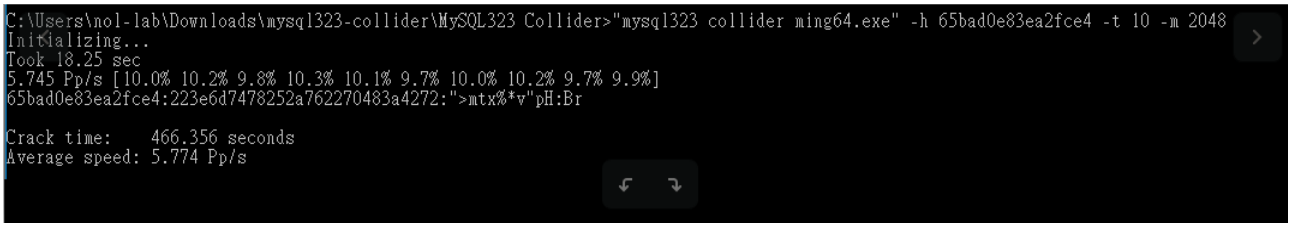
Upload “base64_decode_result” to this website. It will output probability and guess keys.
Download possible keys as “xor_result”. There’s phpMyAdmin’s account and password inside.

2016.01.13
phpMyAdmin Account & Password
Account: BobIsGod
Password: heptagonsapprenticelimply

Step 4: MySQL323 Cracker

(https://tobtu.com/mysql323.php)

According to “functions.php” in “backup.tar.gz”, the encryption method of password of “My Lovely Girlfriend” in Bob’s blog is hash method.
The hash password can be found in SQL. By the result of using MySQL323 Cracker to decode this hash. I got the result below:



“>mtx%*v”pH:Br is the password of article in Bob’s blog.

Result:

What have I learned?

1. Web backend access.
2. Temporary file is unsafe. Make sure to delete it when we finish jobs.
3. “robots.txt” can’t content important information of system, or it will show the leakages of your system. (robots.txt is for web crawler.)
4. How XOR encryption works, and how to decode it.
5. How hash encryption works, and how to decode it.

How to prevent or patch these vulnerabilities?

1. Clean all temporary files on Internet.
2. Do not put significant informations in robots.txt.
3. Encryption method can’t be known easily, or it may be decode.
4. Make all scripts can not be take easily. (hide or encryption or so on.)
5. Check the security of system again and again.