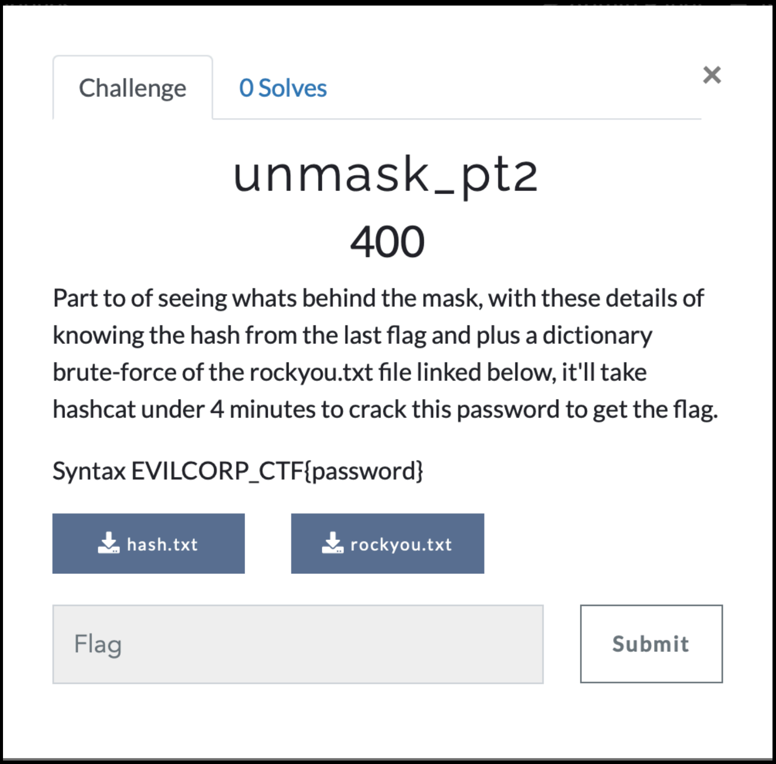
MR. ROBOT

1. This flag is a a part 2 of the last flag, and it looks like that SHA512 is a password that’s hashed. The rockyou wordlist that is needed to break the hash is provided. Rockyou wordlist is a very known wordlist when it comes to brute forcing passwords. So download the rockyou.txt wordlist and you can use the same hash in the last challenge because its the same txt file.



unmask_pt2

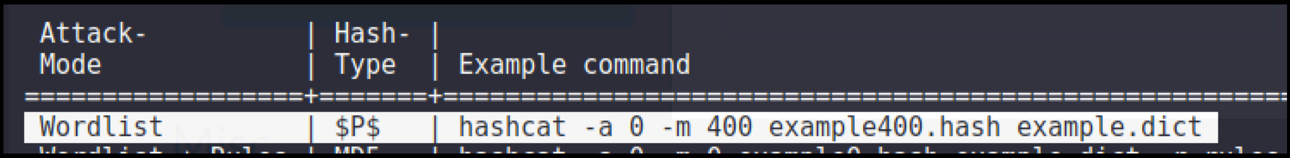
400

Part to of seeing whats behind the mask, with these details of knowing the hash from the last flag and plus a dictionary brute-force of the rockyou.txt file linked below, it'll take hashcat under 4 minutes to crack this password to get the flag.

Syntax EVILCORP_CTF{password}

*** Brute Forcing Tool is required for this challenge so downloading hashcat(Tool used to do this challenge on your local machine or using a Kali machine that has it installed is needed****

2. Hashcat will be the tool of choice of this due to how powerful hash cat can be when brute forcing passwords especially when you have the right wordlist and the right hash algorithm that was used to hash a password. Lets take a look how to run the brute force against this hash.txt file.

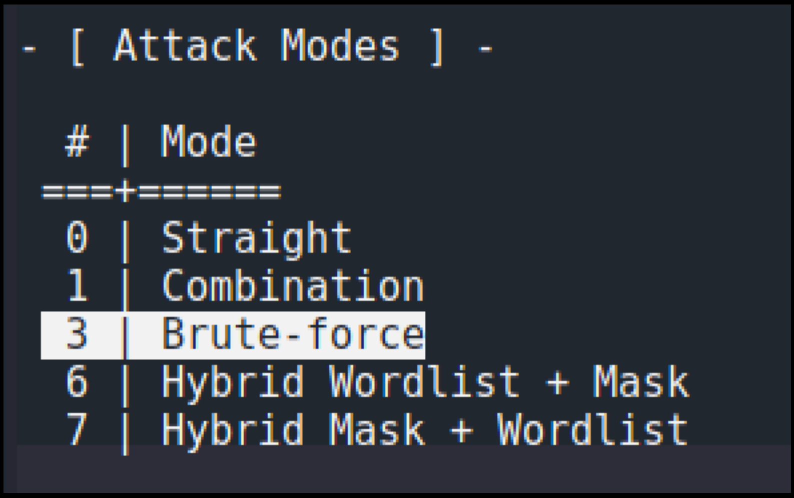


```
 Attack-            | Hash- |
 Mode               | Type  | Example command
====================+=======+==========================================================
 Wordlist           | $P$   | hashcat -a 0 -m 400 example400.hash example.dict
 Wordlist + Rules   | MD5   | hashcat -a 0 -m 0 example0.hash example.dict -r rules
```

3. Above shows hashtags example of how to run a brute force attack using a wordlist. Lets break it down:
    1. hashcat - this is to run the hashcat tool
    2. -a - this is the attack mode
    3. -m this is the hash mode
    4. Then the hash txt file
    5. Then the wordlist used to brute force

   To look for these options run hashcat -h

3. So we've broke down the command above now we just need to put our own modes to suit the hash file and the attack mode we want to use so first were going to take a look at the attack modes.

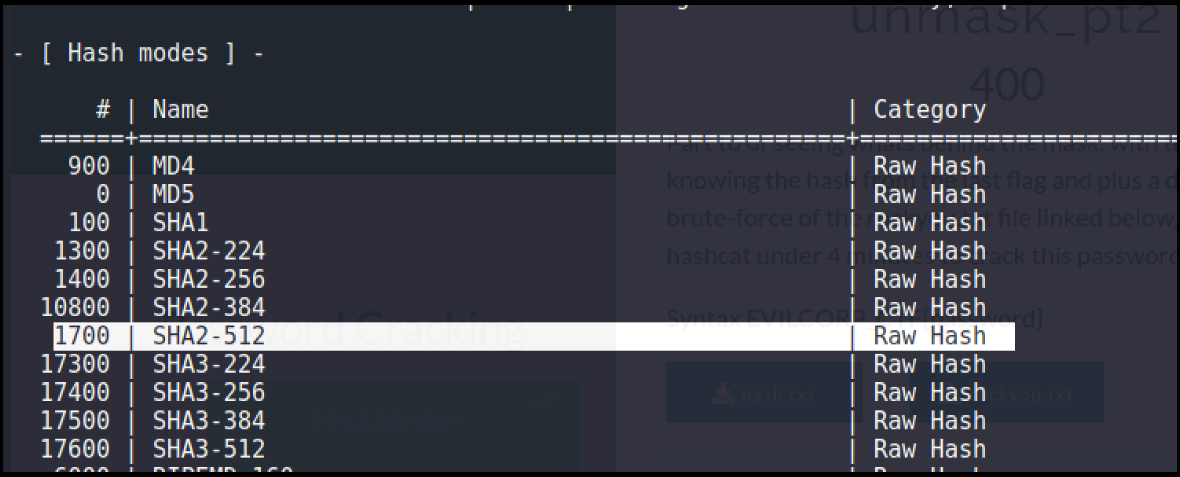


```
- [ Attack Modes ] -

  # | Mode
 ===+======
  0 | Straight
  1 | Combination
  3 | Brute-force
  6 | Hybrid Wordlist + Mask
  7 | Hybrid Mask + Wordlist
```

We are brute forcing the password so we will be using -a 3 to brute force.

4. Next is the hash mode, from doing the first part of the challenge we know the hash of this password is a SHA-512 so we have to look for the hash mode of that.

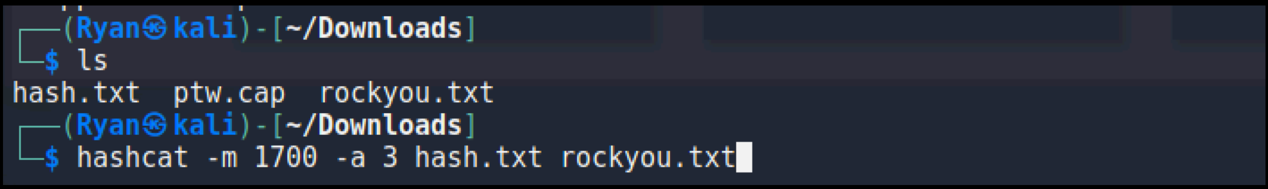


```
- [ Hash modes ] -

      # | Name                                                 | Category
  ======+======================================================+==========
    900 | MD4                                                  | Raw Hash
      0 | MD5                                                  | Raw Hash
    100 | SHA1                                                 | Raw Hash
   1300 | SHA2-224                                             | Raw Hash
   1400 | SHA2-256                                             | Raw Hash
  10800 | SHA2-384                                             | Raw Hash
   1700 | SHA2-512                                             | Raw Hash
  17300 | SHA3-224                                             | Raw Hash
  17400 | SHA3-256                                             | Raw Hash
  17500 | SHA3-384                                             | Raw Hash
  17600 | SHA3-512                                             | Raw Hash
```

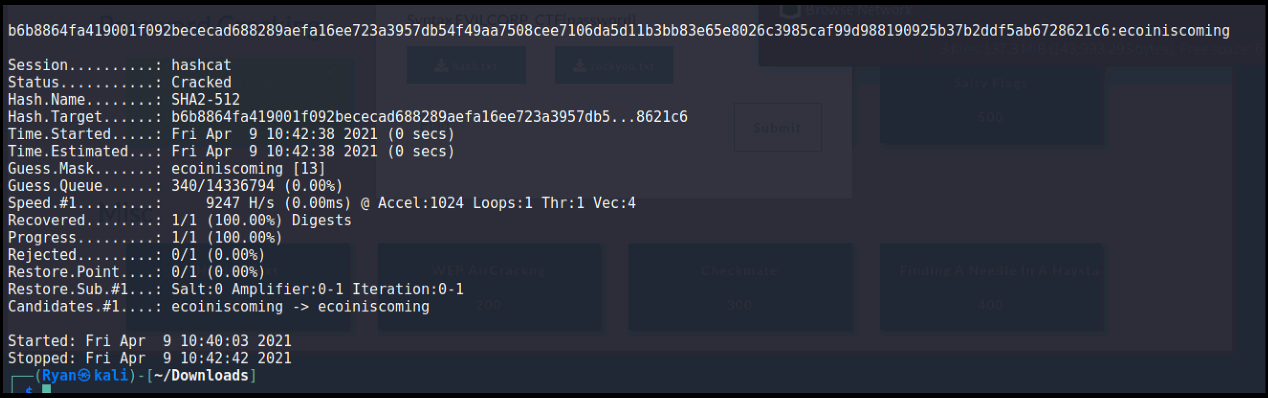
Looking at the above photo hash mode 1700 matches with SHA-512 so in this our command we will be running -m 1700

5. Okay so now all we need is the hash.txt and the wordlist, looking at the below picture change into the directory that the files are downloaded into and run ls to make sure they are there. Using the mode numbers that are suitable to our attack shown in the above steps run the command with the numbers that are suitable to our attack. Hashcat may take a few minutes to run so be patient, if you ran the command correctly with the correct hash.txt and the wordlist provided in the CTF the password will be cracked.



```
┌──(Ryan㉿kali)-[~/Downloads]
└─$ ls
hash.txt  ptw.cap  rockyou.txt
┌──(Ryan㉿kali)-[~/Downloads]
└─$ hashcat -m 1700 -a 3 hash.txt rockyou.txt
```

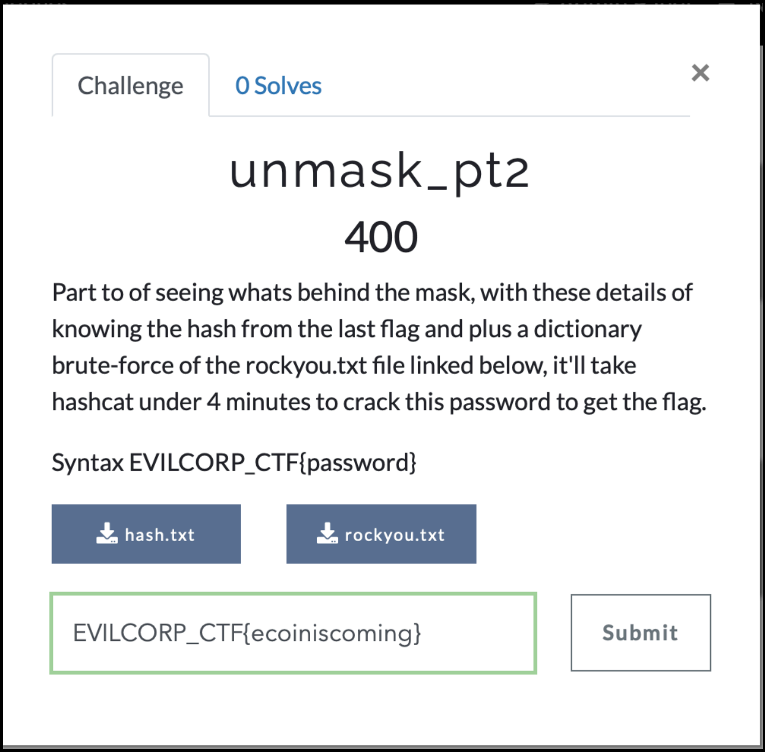
6. It shouldn’t take more than 5 minutes to crack so if it does check your command and make sure your using the wordlist and hash.txt file provided in the challenge. Below is the result you should see and the answer to the flag. ecoiniscoming is the password.



```
b6b8864fa419001f092bececad688289aefa16ee723a3957db54f49aa7508cee7106da5d11b3bb83e65e8026c3985caf99d988190925b37b2ddf5ab6728621c6:ecoiniscoming

Session..........: hashcat
Status...........: Cracked
Hash.Name........: SHA2-512
Hash.Target......: b6b8864fa419001f092bececad688289aefa16ee723a3957db5...8621c6
Time.Started.....: Fri Apr  9 10:42:38 2021 (0 secs)
Time.Estimated...: Fri Apr  9 10:42:38 2021 (0 secs)
Guess.Mask.......: ecoiniscoming [13]
Guess.Queue......: 340/14336794 (0.00%)
Speed.#1.........:     9247 H/s (0.00ms) @ Accel:1024 Loops:1 Thr:1 Vec:4
Recovered........: 1/1 (100.00%) Digests
Progress.........: 1/1 (100.00%)
Rejected.........: 0/1 (0.00%)
Restore.Point....: 0/1 (0.00%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidates.#1....: ecoiniscoming -> ecoiniscoming

Started: Fri Apr  9 10:40:03 2021
Stopped: Fri Apr  9 10:42:42 2021
┌──(Ryan㉿kali)-[~/Downloads]
```



unmask_pt2

400

Part to of seeing whats behind the mask, with these details of knowing the hash from the last flag and plus a dictionary brute-force of the rockyou.txt file linked below, it'll take hashcat under 4 minutes to crack this password to get the flag.

Syntax EVILCORP_CTF{password}

EVILCORP_CTF{ecoiniscoming}