Category:

forensics

Name:

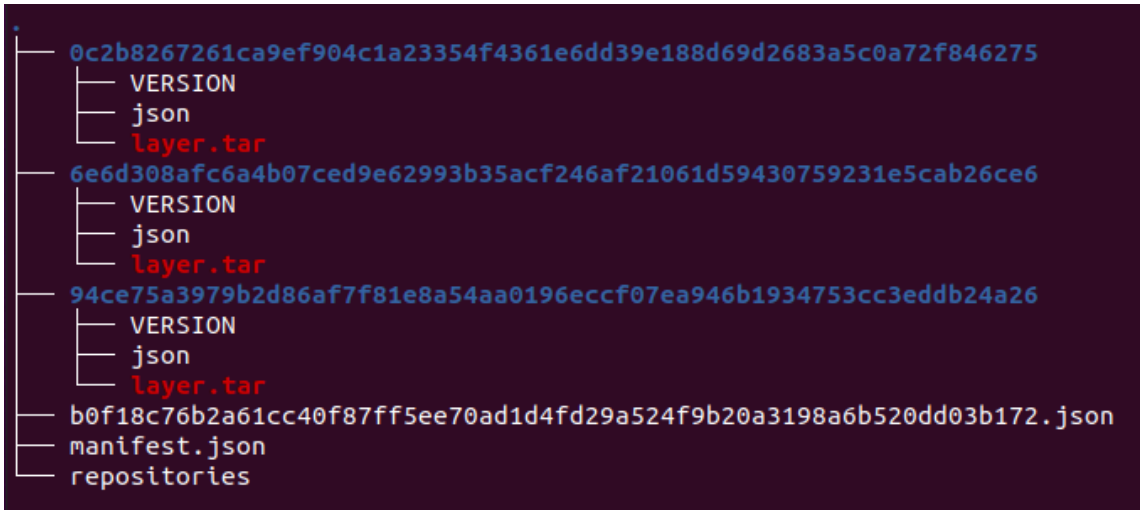
recovery

Message:

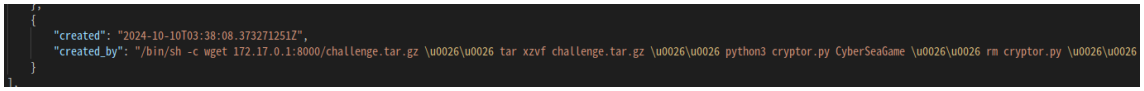
recover flag.txt.

Instructions:

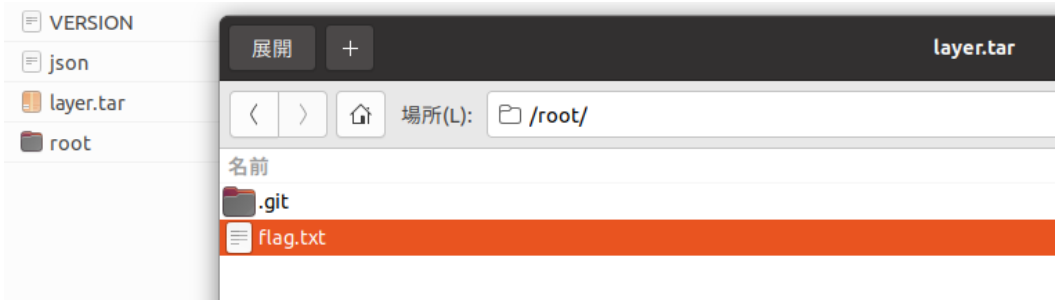
Check the challenge.tar file. Several folders and json files are present. This is the backup file of the docker container.



Check the docker log file b01f(snip)b172.json.” cryptor.py” has been executed and deleted.



Next, as instructed in this challenge, look for flag.txt. flag.txt is easy to found, but it is encrypted.



```
$ xxd flag.txt
00000000: e5f4 1443 e3b2 0a52 46e3 7807 c446 be95  ...C...RF.x..F..
00000010: 547c 2bfa c8f8 2774 c0a1 3970 5ecb 7c07  T|+...'t..9p^.|.
00000020: d74b c082 31                             .K..1
$
```

The ".git" exists in the same directory as flag.txt. Check "git status".

```
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    cryptor.py
```

cryptor.py has been deleted from the commit. use the restore command to restore it.

```
$ git restore -s 32e3c1ce7c75ec8449111e8f09a74b76ad24aa42 cryptor.py
$ cat cryptor.py
import sys
import hashlib
from datetime import datetime,timezone

FILE = "./flag.txt"


key1 = datetime.now(timezone.utc).strftime("%Y%m%d%H%M%S").encode()
key2 = sys.argv[1].encode()
key = key1 + key2
key = hashlib.sha1(key).digest()

with open(FILE,"rb") as f:
    flag = f.read()

while True:
    if len(flag) > len(key) :
        key += key
    else:
        break

enc = bytearray([])
for (f,k) in zip(flag,key):
    enc.append(f^k)

with open(FILE,"wb") as wf:
    wf.write(enc)$
```

Cryptor.py is coded to create a key based on the time it was executed and the string received as the argument and xor the flag.txt. From the Docker log file, the argument and execution time can be determined.

```
"created": "2024-10-10T03:38:08.373271251Z",
"created_by": "/bin/sh -c wget 172.17.0.1:8000/challenge.tar.gz \u0026\u0026 tar xzvf challenge.tar.gz \u0026\u0026 python3 cryptor.py CyberSeaGame \u0026\u0026 rm cry
```

Load the Docker image and check the timestamp just to be sure.

```
$ docker load < challenge.tar
63ca1fbb43ae: Loading layer [==================================================>]  8.082MB/8.082MB
9365b1dee728: Loading layer [==================================================>]  43.87MB/43.87MB
6d0c915867f2: Loading layer [==================================================>]  52.22kB/52.22kB
Loaded image: challenge:latest
$ docker run -itd challenge
e7872c779b293b0b95f01ff530e0e2ebc6b01a6e1e8bd0b68ca81d19cdfe7a25
$ docker ps
CONTAINER ID   IMAGE       COMMAND     CREATED          STATUS          PORTS     NAMES
e7872c779b29   challenge   "/bin/sh"   13 seconds ago   Up 12 seconds             elated_driscoll
$ docker exec -it e7872c779b29 sh
~ # ls --full-time flag.txt
-rw-rw-r--    1 1000     1000            37 2024-10-10 03:38:08 +0000 flag.txt
~ # 1
```

The timestamp is “202410100338” and the argument is “CyberSeaGame”.

The next step is to create the decryption script. Since a simple xor is used for the encryption logic, only the key of cryptor.py is rewritten to create decryptor.py.

```python
# key1 = datetime.now(timezone.utc).strftime("%Y%m%d%H%M%S").encode()
key1 = "20241010033808".encode()
# key2 = sys.argv[1].encode()
key2 = "CyberSeaGame".encode()
```

If you run this script, you will succeed in recovering the flag.txt file

```
$ python3 decryptor.py
$ cat flag.txt
CSG_FLAG{Great_job_on_the_recovery!}
```