5월 Dreamhack CTF Writeup Dreamhack CTF Season 5 Round #10

32기 송지윤

이번 Dreamhack CTF는 리버싱 문제에 도전해 보았다. flag의 형식은 DH{...} 이라고 한다. 문제 파일을 다운로드 받으니 다음과 같은 폴더와 파일들이 있었다.

1a3d40f4-d365-4002-8aea-b52519c88a19	
deploy	
Dockerfile	
prob prob	
deploy 폴더 안에는 다음과 같은 파일들이 들어 있었다.	
[] flag	
prob	

먼저 Dockerfile을 메모장으로 열어보았다.

우선 Dockerfile이 어떤 걸 나타내는 파일인지 몰라서 찾아봤다. Dockerfile은 Docker 이미지를 만드는 데 사용되는 파일이다. Docker 이미지는 애플리케이션과 그 애플리케이션을 실행하는 데 필요한 모든 것을 포함하는 일종의 템플릿이다. ChatGPT의 예시에 따르면 Docker 이미지는 요리의 레시피와 같다고 할 수 있다.

아까 열어 본 Dockerfile을 다시 한 번 보면, 첫 부분에 'FROM ubuntu:22.04'라고 쓰여있다. 뭔가 '이 문제를 ubuntu 운영체제에서 풀어야 하나'라는 생각이 들었다.

다음으로

@sha256:aabed3296a3d45cede1dc866a24476c4d7e093aa806263c27ddaadbdce3c1054 이런 내용이 적혀 있어 찾아보았다.

이것은 Docker 이미지를 사용할 때 이미지를 정확하게 식별하기 위한 고유한 해시값이라고 한다. @sha256은 SHA-256 해시 알고리즘을 사용해 생성된 해시값임을 나타낸다고 한다.

FROM ubuntu:22.04 @sha256:aabed3296a3d45cede1dc866a24476c4d7e093aa806263c27ddaadbdce3c1054

ENV user infant ENV chall_port 31337

RUN apt-get update RUN apt-get -y install socat

RUN adduser Suser

ADD ./deploy/flag /home/\$user/flag ADD ./deploy/prob /home/\$user/prob

RUN chown rootSuser /home/Suser/flag RUN chown rootSuser /home/Suser/prob

RUN chmod 755 /home/\$user/prob RUN chmod 440 /home/\$user/flag

WORKDIR /home/\$user
USER \$user
EXPOSE \$chall_port
CMD socat -T 25 TCP-LISTEN:\$chall_port.reuseaddr.fork EXEC:/home/\$user/prob

그 밑에 적힌 내용들은 Docker 이미지를 만들기 위한 명령어와 설정들인 것 같다.

다음으로 'prob'라고 적힌 파일을 메모장을 이용해 열어보았다. 그런데 다음과 같이 처음 보는 이상한 형태로 보인다.

```
DELEGGG
           0 > 0 7
                     (20)
                          H1
                                  m 8
@000 0 @
                                   0 0 00
                                                  00
                                                        00
    0 0 0
                                                   п
                                       □ X-
0 0
                            0
                                 X=
                                                     X=
                                                              D
0 0 h-
                                 D 8D
                                                80
                                                     80
                                                          0
                                                               Ó
    0 0 60
               hū
                    hD
                          D
                               D
                                        S?dD 8D
    P?dE ?
                                    Q?dD
L
                               tī.
                                                               D
R?dD X-
         X=
              X=
                           /lib64/ld-linux-x86-64.so.2
                                                     П
                                                           GNU
         00 7 0
                   ㅁ ㅁ ㅁ GNU G??<뛰F텍? 미리??ㅁ ㅁ ㅁ GNU
0 ? 0
                         (D???fUaD970
0 0 0 0 0?00000 0
                                                  d D
_ []
                                                      0
                                                    1
D
             0
                           D
                                       0 0
D
                        p
                          .
                                      1
                                        v
                           X 00 @
                                      getline _cxa_finalize read alarm _libc_start_main signal fopen
    0 0000
fclose memset set/buf stdout puts free stdin stderr exit _stack_chk_fail printf libc.so.6 GLIBC_2.4
GLIBC_2.34 GLIBC_2.2.5 _ITM_deregisterTMCloneTable __gmon_start__ _ITM_registerTMCloneTable
0 ? 0 27 0 ? 0 u0i
                       D 7
                                  п
                                                     00

    □ 600

                                                                0
                             X=
0@
         0 0
                       0 0
         0 0
                       0 0
                                 (0)
                                      0 0
                                                      0 0
                                                                00
0 0
                n
                              п
                                             0
         p?
                        x?
                                     177
0
                  7
                        0
                              .7
```

메모장을 이용해서 여는 파일이 아닌가 싶어서 vs code를 이용해 열어보려고 했는데



파일이 이잔미거나 지원되지 않는 텍스트 인코딩을 사용하므로 텍스트 편집기에서 표시되지 않습니다.

계속 열기

이런 경고 문구가 뜨면서 열리지 않는 것 같다. 이 파일은 어떤 걸 사용해서 여는 파일인지 잘 모르겠다.

다음으로 deploy 폴더의 flag 파일을 메모장으로 열어보았다.

testflag

'testflag' 라고만 쓰여있고 별다른 내용은 없다.

그리고 deploy 폴더에도 prob 파일이 있어 일단 메모장으로 열어봤는데 앞서 열어 본 prob 파일과 동일하게 보였다.

일단 문제에 들어있던 모든 파일들을 열어봤는데 이제 어떻게 해야 하는지 잘 모르겠다...

사실 본격적인 문제 풀이는 전혀 하지 못한 것 같다. CTF는 사실 아직 많이 어렵다. 어떤 식으로 접근해야 하는지도 모르겠고 뭔가 분석해 보고 싶어도 문제 파일조차 어떻게 여는지 모르는 경우가 많다. 또 CTF 문제들을 풀기 위해서는 다양한 도구를 사용해야 하는 것으로 알고있는데 어떤 경우에 어떤 도구를 사용해야 하는지, 그 도구는 어떻게 사용하는 건지에 대해서도 더 많이 공부해야 할 것 같다.