코드엔진 Basic RCE L14

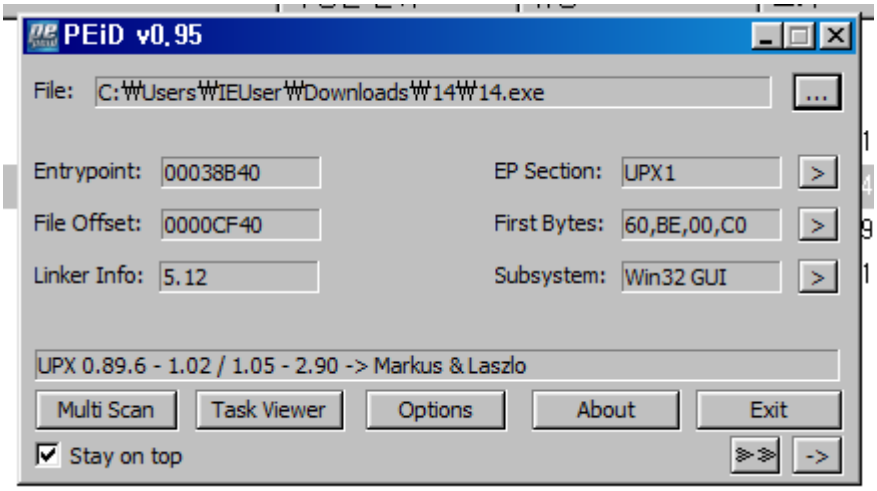
문제

> CodeEngn.com [코드엔진]
>
> 코드엔진은 국내 리버스엔지니어링 정보공유를 위해 2007년 부터 리버스엔지니어링 컨퍼런스 및 세미나, 워크숍을 현업 실무자들과 함께 운영하고 있는 비영리 커뮤니티입니다.
>
> https://ch.codeengn.com/

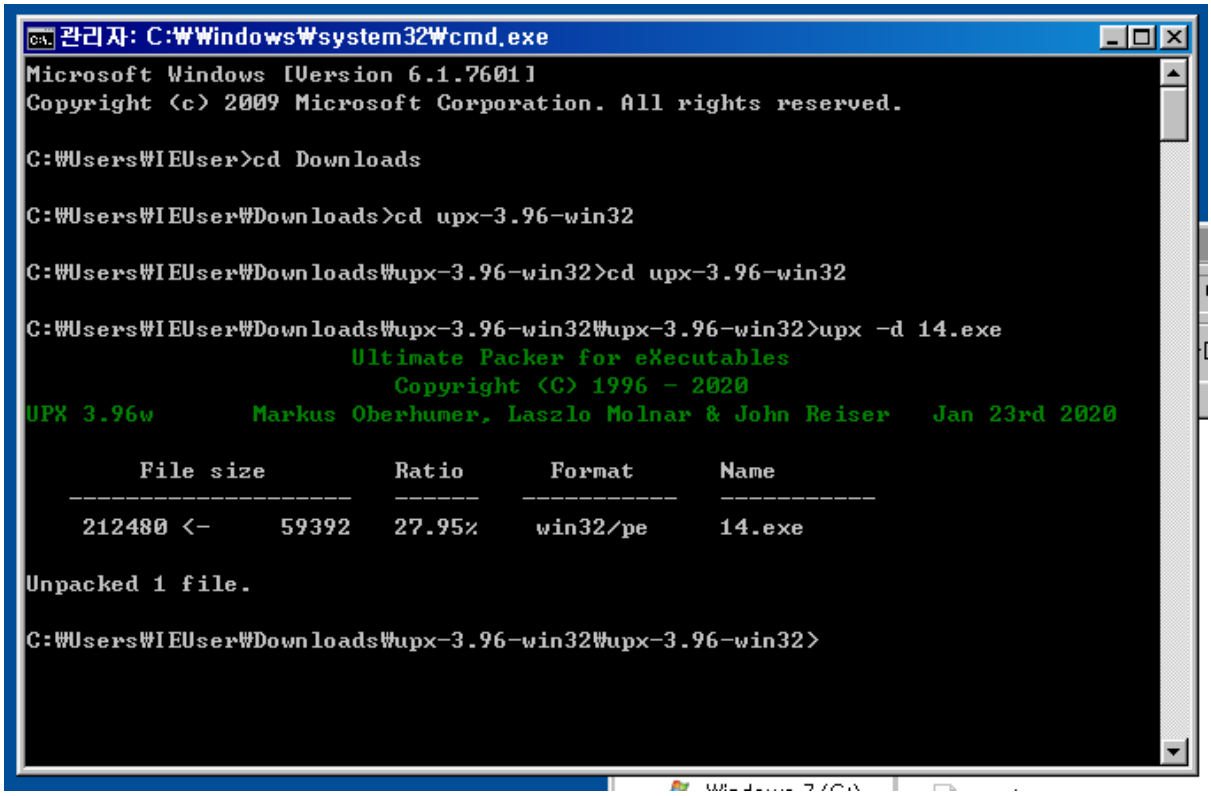
의 Basic RCE L14

해결 과정

- PEiD를 사용하여 실행파일 정보파악(압축여부, 작성시 사용한 언어)

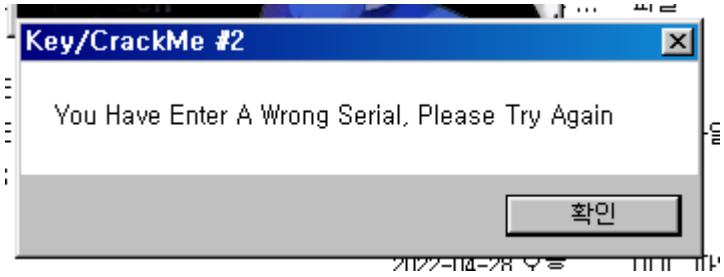


- 실행파일이 UPX로 압축되어 있으므로 압축을 풀어야 함



- UPX를 github에서 다운로드 받은 후, UPX가 실행파일이 있는 폴더로 가서 upx -d “압축해제하고 싶은 실행파일명” 명령을 실행하면 실행파일의 UPX 압축을 해제할 수 있음

- 실행파일을 실행해 본 결과 값을 입력해야 할 두 칸이 보임



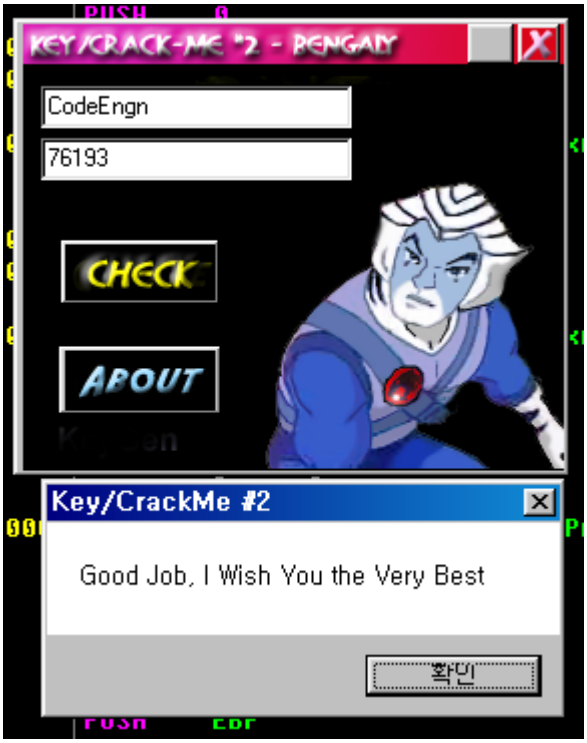
- 앞 칸에 'CodeEngn' ,뒷 칸에 '11111'이라는 똑같은 값을 입력한 결과 잘못된 시리얼을 입력했다는 메시지창을 출력함

```
(Initial CPU selection)
ASCII "Bengaly"
ASCII "MainWindow"
ASCII " Key/CrackmMe - #2"
ASCII "Key/CrackMe #2    "
ASCII "    +=================================+    |  Key/Crac
ASCII "Key/CrackMe #2    "
ASCII "     Please Fill in 1 more Char!!"
ASCII "Key/CrackMe #2    "
ASCII " Good Job, I Wish You the Very Best"
ASCII "Key/CrackMe #2    "
ASCII " You Have Enter A Wrong Serial, Please Try Again "
```

- ollyDbg에 실행파일을 올리고 레퍼런스된 문자열을 검색함
- 문자열 검색의 결과 올바른 시리얼 값을 입력했을 때 나올것으로 추정되는 문자열이 보임

```
00401334   .  E8 4A000000   CALL   00401383
00401339   .  5E            POP    ESI
0040133A   .  3BC6          CMP    EAX, ESI
0040133C   .  75 15         JNZ    SHORT 00401353
```

- 추정되는 문자열을 클릭하여 문자열을 사용하는 코드를 찾은 후 그 위에 있는 무언가를 비교하는 코드에 브레이크 포인트를 검
- 브레이크 포인트를 건 이유는 비교 코드 이후에 점프 코드가 있는데 이 점프가 작동하면 성공 문자열을 출력하는 메시지박스 함수를 실행하지 않기 때문임
- 여러 값을 각 입력칸에 넣어보면서 비교 코드에서 사용되는 2개의 레지스터 EAX와 ESI 값의 변화를 보면 EAX는 입력한 Serial값에 따라 값이 달라지고 ESI는 입력한 Name 값에 따라 값이 달라지는 것을 확인할 수 있음
- Serial 값에 '2222'를 넣은 후 '2223'을 넣어보면 '2223'에서의 EAX값은 '2222'에서의 EAX값에 1을 더한 값임을 알 수 있음
- 따라서 알맞은 Serial을 구하려면 ESI값에서 EAX값을 빼고 뺀 값을 10진수로 변환한 후 그 변환한 값과 Serial에 넣었던 값을 더하면 됨



- 위의 방식으로 구한 Serial을 입력값으로 넣어보면 정답 문자열을 보여주는 메시지 박스를 출력하는 것을 확인할 수 있음