

노용환 멘토님 | 윈도우 강의 | 07.18

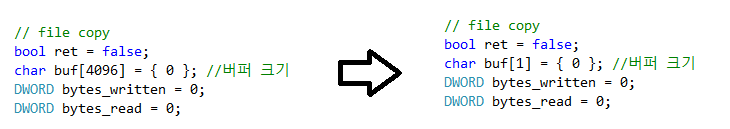
운영체제 Hw2

취약점 분석 트랙 \_ 안보경

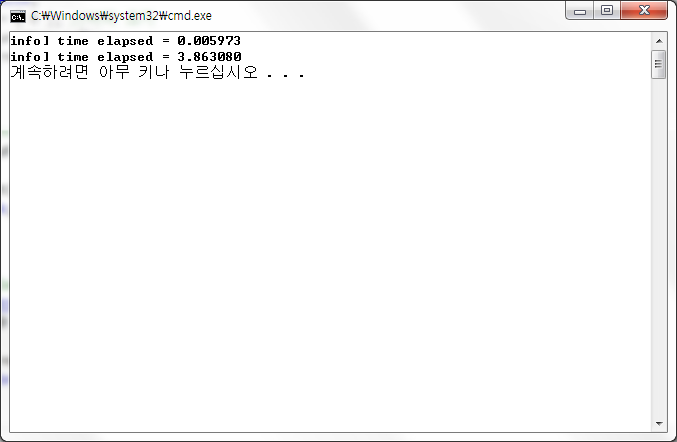
# 

처음 프로그램을 실행 시키게 되면, 의도와 다르게 프로그램의 실행 시간의 차이가 file\_copy\_using\_read\_write 를 사용한 input/ourpur 이 더 빠르게 나오는 것을 확인할 수 있습니다.

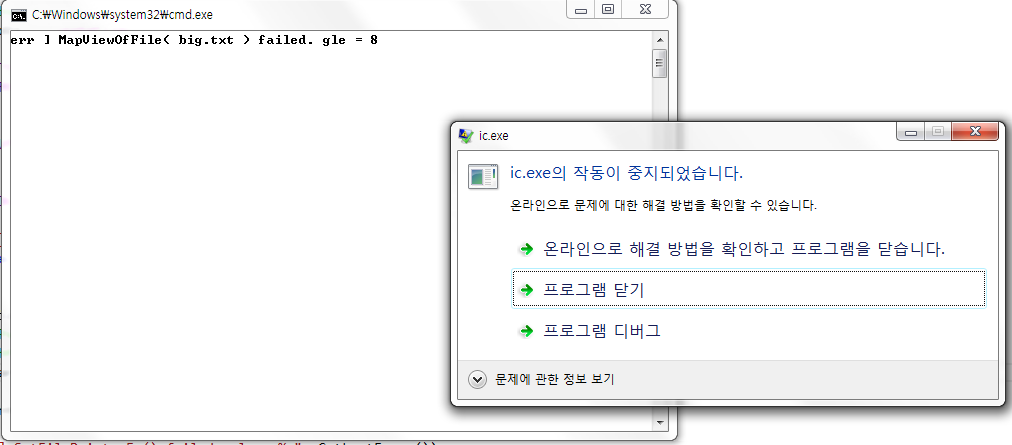
이유는 파일을 복사하는 부분에서 버퍼 크기를 초기에 4096으로 주었기 때문에 차이가 나게 되는데, 이 문제를 해결하기 위해서는 버퍼의 크기를 1로 수정해 주어 풀이하면 될 것 같습니다.



다음과 같이 복사하는 부분의 코드를 수정한 뒤에

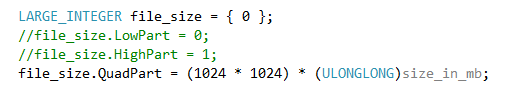


다음과 같이 의도대로 프로그램의 실행시간 속도 차이가 상당히 나는 것을 확인할 수 있었습니다. 또한 파일의 크기를 5G 이상으로 잡게 되면,

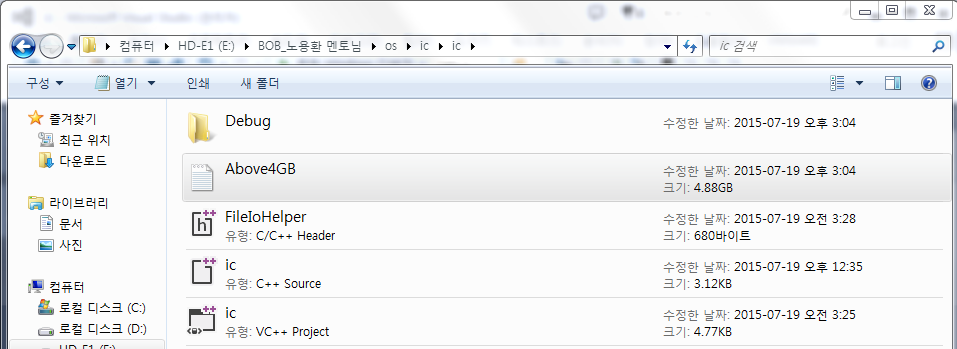


위와 같이 한번에 읽을 수 없기 때문에 파일을 분할해서 처리해야 할 것 같습니다.

파일을 저장하거나 읽어올 때 32 bit로는 한계가 있기 때문에

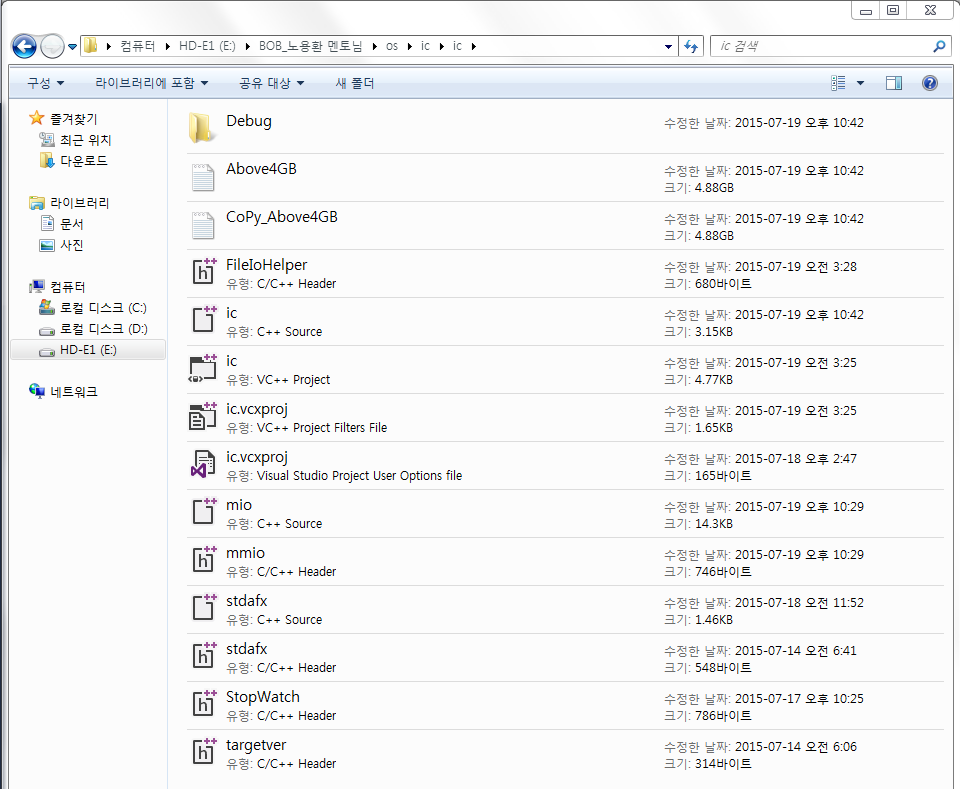


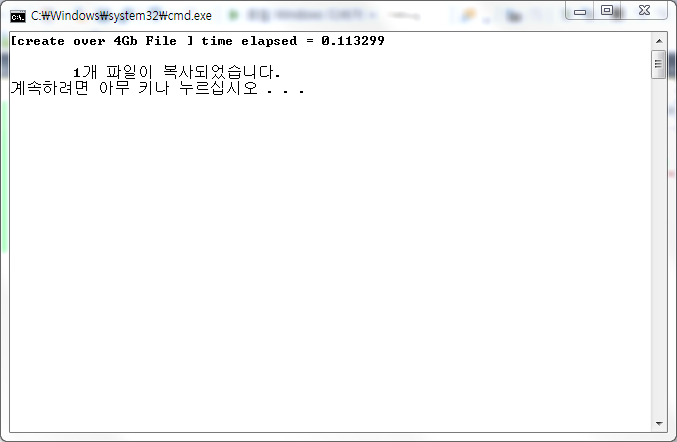
다음과 같이 코드를 수정해서 큰 파일을 저장하는 부분을 구현했습니다. 또한



또한 파일 복사에 대한 부분은 system Call 을 사용해서 두 번째 파일도 구현할 수 있었습니다.

물론 정상적인 경로는 아니지만, 해킹과 취약점 분석을 배우는 학생으로서 이렇게 풀이해보았습니다.





------------------------------------------------------- ----------------------------------------------------------------

다음과 같은 풀이를 통해 이번 과제를 완료할 수 있었습니다. 파일 입출력을 그 동안 여러 번 수행 해봤지만, 이렇게 큰 크기의 파일을 만들 일이 없었는데 이번 기회를 통해 큰 파일을 생성할 때 어떻게 만들어야 하는지 알 수 있었습니다.