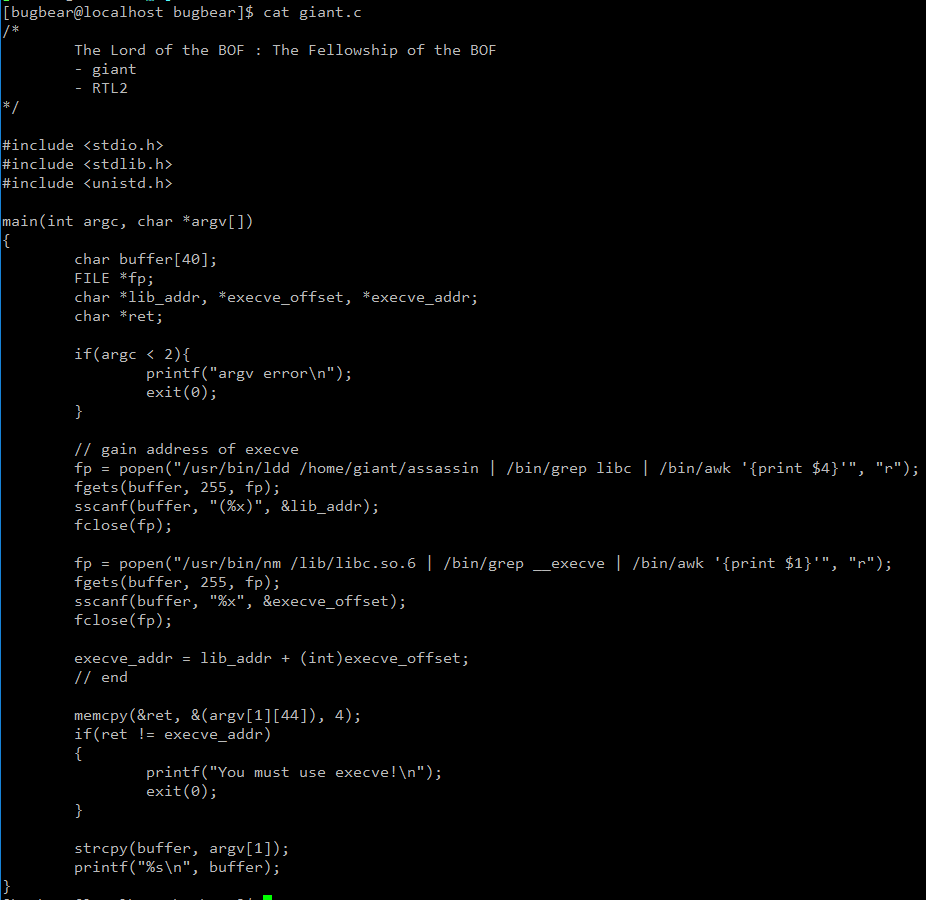
|  |
| --- |
| LOB giant |
| 동아리마크  2018. 03. 11 (SUN)  Dong-a .univ  COLONY  Young jin kim |

목차

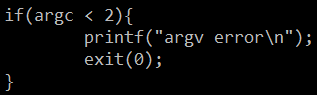
1. 소스코드 이해
2. 소스코드 분석
3. 공격
4. **소스코드 이해**



[ bugbear.c의 소스코드 ]

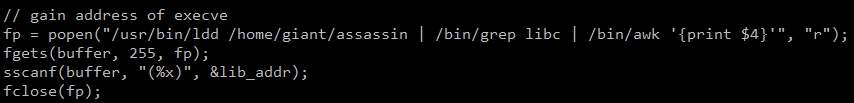
|  |  |
| --- | --- |
| **Ldd** : 프로그램이 사용하는 공유 라이브러리를 알 수 있다.  **Awk** : 표준입력이나 파일로부터 입력을 받고 문자를 파싱해서 원하는 결과를 얻어내는데 사용  **파싱** : 문자열을 구분자로 쪼개는 것  **Nm** : 오브젝트 파일,라이브러리,프로그램에 쓰인 함수가 어떤 것인지 확인할 때 사용 | 1 |

1. **소스코드 분석**



[ 조건 1 ]

첫번째, 인자는 2개 이상 들어와야한다.



[ 소스 분석 ]

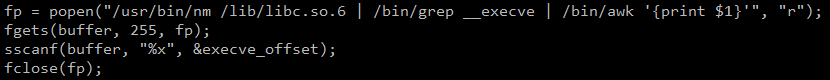
첫번째, fp파일포인터 변수로 스트림 연결을 한다.

두번째, /home/giant/assassin이 사용하는 공유 라이브러리 주소값을 읽는다.

세번째, fgets로 fp에 있는 공유라이브러리 주소값을 버퍼에 넣는다.

네번째, buffer에 있는 공유라이브러리 주소값을 lib\_addr변수에 16진수로 넣는다.

다섯번째, fp파일포인터 스트림 종료.



[ 소스분석 2 ]

첫번째, libc.so.6에 쓰인 함수 \_\_execve의 주소값을 buffer로 읽는다.

두번째, execve가 얼마나 떨어져있는지 알려주는 주소값을 execve\_offset에 넣는다.

세번째, fp파일포인터 스트림 종료.

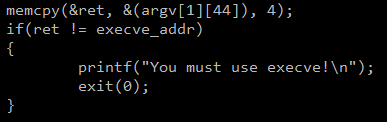
**2.소스코드 분석**



[ 소스분석 3 ]

첫번째, 공유라이브러리 주소값에 execve함수가 있는만큼의 거리를 더해준다.

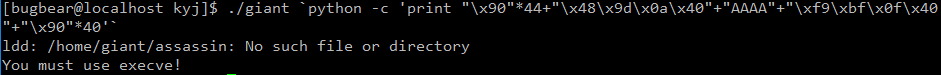
두번째, execve함수의 메모리값을 execve\_addr에 저장한다.



[ 조건 2 ]

첫번째, ret하는 주소와 공유 라이브러리의 execve함수의 메모리값과 다르면 프로그램이 종료

두번째, ret하는 주소는 execve주소로 가야한다.



[ execve 에러 발생 ]

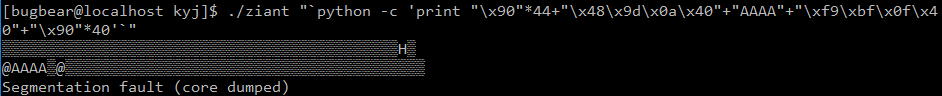
첫번째, 사본파일을 이용해서 익스플로잇을 하려 했으나 execve의 주소가 틀렸다.

두번째, /home/giant/assassin을 찾을 수 없다고 나온다.

세번째, 내가 찾은 execve의 주소는 assassin의 공유 라이브러리 주소인 것 같다.

네번째, /home/giant/assassin을 /home/bugbear/giant로 수정하였다.

**2.소스코드 분석**

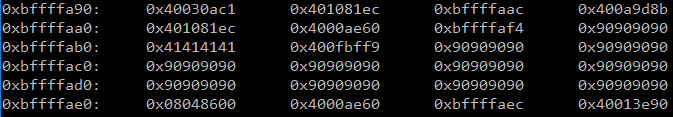


[ 세그먼트 폴트 ]]

첫번째, 위의 짐작과는 상관없이 파이썬에서 0a를 개행문자로 인식한다고 한다.

두번째, 따라서 파이썬 코드를 “ “(더블 쿼테이션)으로 감싸주어야 한다.

세번째, 세폴이 떳으니 다시 디버깅을 시작해보자.



[ 예상치 못한 오류 ]

첫번째, ret를 수정했는데 코어파일을 확인해 보면 제대로 버퍼오버플로우가 발생하지 않았다.

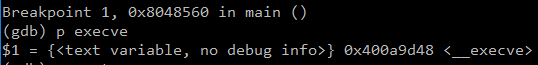
두번째, 페이로드를 조금 수정해서 system함수를 콜한다.



[ system함수 주소 ]

첫번째, system함수의 주소는 0x40058ae0

**2.소스코드 분석**



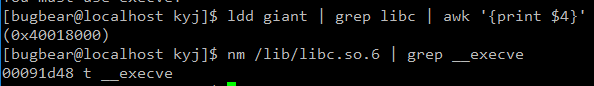
[ execve의 주소 ]

첫번째, execve함수의 주소는 0x400a9d48이다.



[ “/bin/sh”의 주소 ]

첫번째, /bin/sh의 주소는 0x400fbff9이다.



400a9d48



[ system함수의 주소 ]

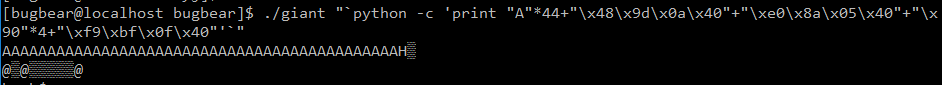
첫번째, system 함수의 주소는 0x40058ae0

**3.공격**

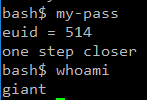
공격을 시작해보자.

나의 페이로드는 다음과 같다.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Buffer[40] | Sfp[4] | Ret[4] | Ret[4](execve) | Argc[4] | Argv[1] |
| A\*40 | A\*4 | &execve() | & system | AAAA | & /bin/sh |



[ 익스플로잇 ]



[ 성공 ]

PW : one step closer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Buffer[40] | Sfp[4] | Ret[4] | argc | Argv[1] |
| A\*40 | A\*4 | &execve() | AAAA | & /bin/sh |

[오류 1]

첫번째, RTL2 에서 ret만 수정하는게 아니라 ret한 함수에서 ret되는 그 뒤에 메모리값

Argc도 수정을 해 줘야 공격이 가능하다.