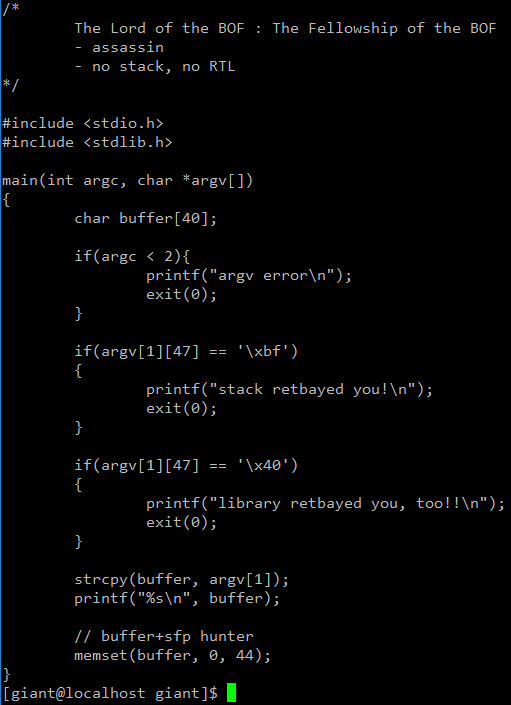
|  |
| --- |
| LOB assassin |
| 동아리마크  2018. 03. 27 (tue)  Dong-a .univ  COLONY  Young jin kim |

목차

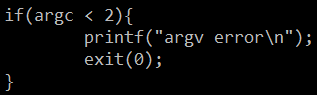
1. 소스코드 이해
2. 소스코드 분석
3. 공격
4. **소스코드 이해**



[ giant .c의 소스코드 ]

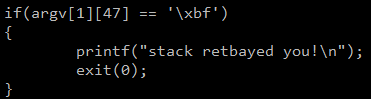
|  |  |
| --- | --- |
| 스택, 공유라이브러리 둘다 ret를 사용 못 한다.  인자가 1개이하면 프로그램 종료  Ret가 \bf로 시작하면 프로그램 종료  Ret가 \40로 시작하면 프로그램 종료 | * Ret Sled  1. Ret를 ret를 호출하는 주소로 수정 |

1. **소스코드 분석**



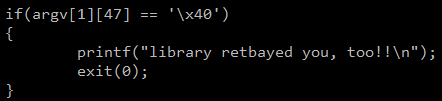
[ 조건 1 ]

첫번째, 인자는 2개 이상 들어와야한다.



[ 조건 2 ]

첫번째, 첫번째인자 48번째 데이터는 \xbf이면 안된다.



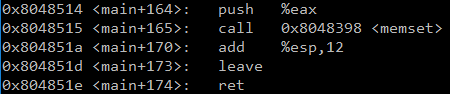
[ 조건 3 ]

첫번째, 첫번째 인자 48번째 데이터는 \x40이면 안된다.

**2.소스코드 분석**

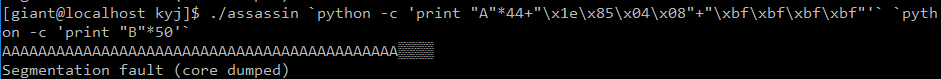
1. \xbf메모리 사용불가 -> no stack
2. \x40메모리 사용불가 -> 공유 라이브러리 사용 불가
3. 스택, 공유라이브러리가 아니면서 ret할 곳을 찾아야겠다는 생각을 가지고

웹 서핑을 하였다.



[ Ret 위치 ]

첫번째, Ret 위치는 0x0804851e 이다.



[ core파일 분석 1 ]

첫번째, 코어파일을 생성하였다.

두번째, 인자 2 의 메모리 주소값(쉘코드 삽입지점)도 파악 해야한다.



[ argv[2]의 주소값 ]

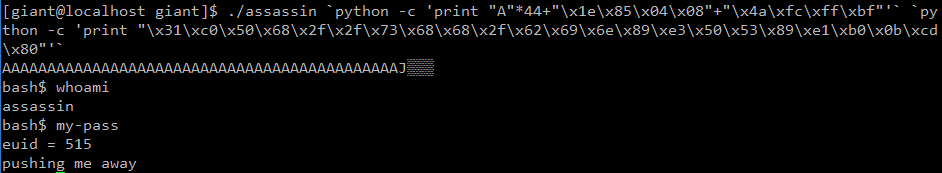
첫번째, argv[2]의 주소값은 0xbffffc4a 이다.

**3.공격**

공격을 시작해보자.

나의 페이로드는 다음과 같다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Buffer[40] | Sfp[4] | Ret[4] | Ret of Ret | Argv[2] |
| Dummy | Dummy | & Ret | & ShellCode | ShellCode |
| A\*40 | A\*4 | 0x0804851e | 0xbffffc4a | ShellCode |



[ 익스플로잇 ]

PW : pushing me away