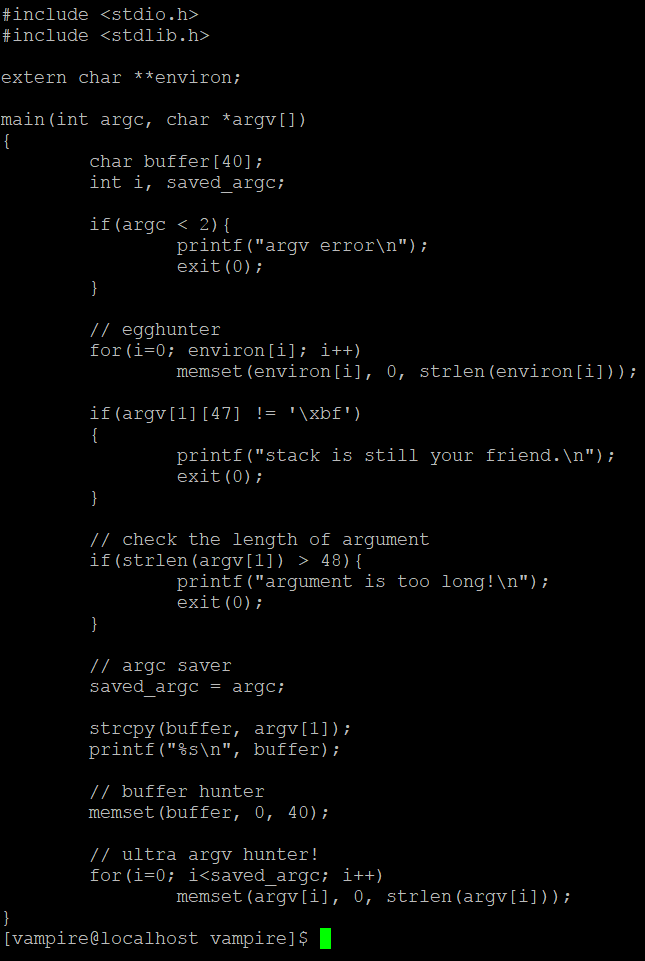
|  |
| --- |
| LOB skeleton |
| 동아리마크  2018. 02. 22 (thu)  Dong-a .univ  COLONY  Young jin kim |

목차

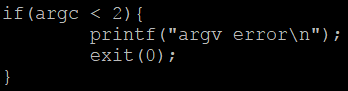
1. 소스코드 이해
2. 소스코드 분석
3. 공격
4. **소스코드 이해**



[ skeleton.c의 소스코드 ]

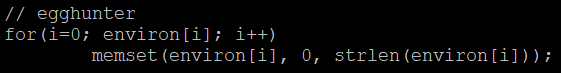
|  |  |
| --- | --- |
| Environ 포인터 선언  Buffer,I,argv 변수 선언  인자가 두개 이하이면 프로그램 종료  환경변수 초기화  Argv[1][47]의 값이 \xbf가 아니면 종료  첫번째 인자의 길이가 48이상이면 종료  argc값을 save\_argc에 저장  argv[1]을 buffer에 복사 | buffer문자열 출력  buffer변수 초기화  argv[i]를 argv[i]의 길이만큼 0으로 초기화 |

1. **소스코드 분석**



[ 조건 1 ]

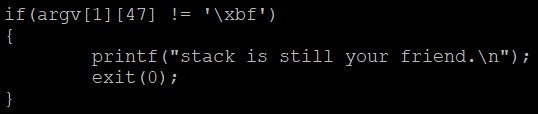
첫번째, 인자는 2개 이상 사용한다



[ 조건 2 ]

첫번쩨, 환경변수를 for구문으로 초기화 함

두번재, 환경변수에 쉘코드 삽입은 불가하다.

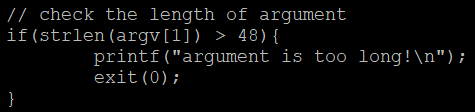


[ 조건 3 ]

첫번째, 첫번재 인자의 48번째 데이터는 \xbf가 들어가야한다.

두번째, 우리가 ret하는 지점의 메모리값이 강제 된다.

1. **소스코드 분석**



[ 조건 4 ]

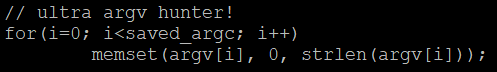
첫번째, argv[1]의 길이가 48보다 크면 프로그램 종료



[ 조건 5 ]

첫번째, buffer변수 초기화

두번째, buffer에도 쉘코드를 넣는 것은 불가능하다.



[ 조건 6 ]

첫번째, 인자를 순차적으로 초기화 한다.

두번째, 이름에 쉘코드 삽입도 불가능해졌다.

1. **소스코드 분석**

정리하면

1. 인자 2개이상
2. 환경변수사용 금지
3. \xbf으로 시작하는 메모리
4. 첫번째 인자의 크기는 48이하
5. buffer변수 사용불가
6. 이름에 쉘코드 삽입 불가.



[ 힌트 1 ]

첫번째, 다지워진줄 알았던 메모리에 우리가 실행한 파일 경로가 남아있는 것을 확인

두번째, 경로에 쉘코드 심볼릭 링크를 걸어 준다

세번재, 이 경로가 남아있는 메모리 주소로 ret한다.

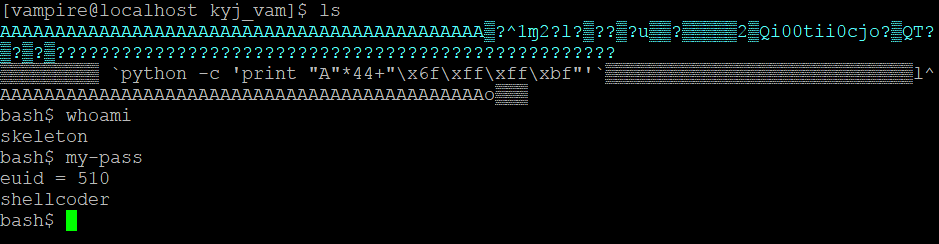


[ 힌트 2 ]

첫번째, 우리가 사용했던 경로가 \xbfffff6b에 들어가 있다.

두번째, 여유있게 \xbfffff6f를 ret로 사용한다.

1. **공격**



[익스플로잇]

Pw : Shellcoder