Lazenca Technote ROP Summary

윤준혁

ROP(Return-oriented programming)는 공격자가 실행 공간 보호(Nxbit) 및 코드 서명(Code signing)과 같은 보안 방어가 있는 상태에서 코드를 실행할 수 있게 해주는 기술이다. 이 기법은 기본적으로 RTL기법을 이용하며, 공격자는 가젯(Gadgets)이라고 하는 해당 프로그램이 사용하는 메모리에 이미 있는 기계 명령어와 RTL을 이용해 공격에 필요한 코드를 프로그래밍 하는 것이다. 각 가젯은 일반적으로 반환 명령어(ret)로 끝이나고 여러 개의 함수를 호출하기 위해 사용된다. (ex – "pop; pop; pop; ret") 해당 Gadgets들의 역할은 ESP 레지스터의 값을 증가시키는 것이며 RTL에서 호출할 함수의 다음 영역에 Gadgets의 주소를 저장함으로써 연속해서 다음 함수가 호출될 수 있다.

PLT(Procedure linkage table, 프로시저 링키지 테이블)에는 동적 링커가 공유 라이브러리의 함수를 호출하기 위한 코드가 저장되어 있고 GOT(Global offset table, 전역 오프셋 테이블)에는 동적 링커에 의해 공유 라이브러리에서 호출할 함수의 주소가 저장된다.
Debug 시 볼 수 있는 read@plt 영역에는 libc 에서 read() 함수를 호출하기 위한 코드가 저장되어 있는데 그 진행과정은 다음과 같다.

1. read@got(0x804a00c) 영역에 저장된 주소로 이동합니다. 이때 read@got(0x804a00c) 영역에는 <read@plt+6>(0x8048306)영역의 주소가 저장되어 있는데 이는 해당 프로그램에서 read() 함수가 한번도 호출되지 않았기 때문이다.
2. <read@plt+11> 의 "jmp 0x80482f0" 코드에 의해 _dl_runtime_resolve() 함수를 호출합니다.
3. 해당 함수는 libc 에서 찾고자 하는 함수(read)의 주소를 .got.plt 영역에 저장합니다.
4. read() 함수가 호출된 후 read@got(0x804a00c)영역에는 libc 의 read() 함수 주소가 저장되어 있습니다.

전체적인 ROP 기법의 Exploit 방법은 다음과 같다.

1. read함수를 이용해 "/bin/sh" 명령을 쓰기 가능한 메모리 영역에 저장
2. write 함수를 이용해 read 함수의 .got 영역에 저장된 값을 출력
3. read 함수를 이용해 read 함수의 .got 영역에 system 함수의 주소로 덮어씀
4. read 함수 호출 – read .got 영역에 system 함수의 주소가 저장되어 있기 때문에 system 함수가 호출됨