Lazenca ROP(x86)

ROP (Return Oriented Programming)

- 공격자가 실행 공간 보호 (NXbit) 및 코드 서명 (Code signing)과 같은 보안 방어가 있는 상태에서 코드를 실행할 수 있게 해주는 기술
- Stack Overflow 취약점, Gadgets (해당 프로그램이 사용하는 메모리에 이미 있는 기계 명령어) 가 필요
- 기본적으로는 RTL 기법을 이용하는 것과 같음, 공격자는 RTL + Gadgets를 이용하여 공격에 필요한 코드를 프로그래밍 하는 것
- 시나리오
  - Stage 1 : 공격을 위해 필요한 요소들을 구하는 과정
    - 메모리 보호 기법이나 여러 상황에 따라 필요한 요소들을 구함
  - Stage 0 : Exploit 진행
    - **함수 호출 → 그 함수의 PLT 참조 → 그 함수의 GOT 참조 → GOT에 들어 있는 실제 주소에 접근해 함수 호출**

    1. read 함수를 이용해 "/bin/sh" 명령을 쓰기 가능한 메모리 영역에 저장 → 거의 bss 영역
    2. write 함수를 이용해 read 함수의 .got 영역에 저장된 값을 출력
    3. read 함수를 이용해 read 함수의 .got 영역에 system 함수의 주소로 덮어씀 (got overwriting)
    4. read 함수를 호출 → read .got 영역에 system 함수의 주소가 저장되어 system 함수가 호출됨

    - PLT : 바이너리 영역에 바이너리에서 사용되는 함수들의 정보를 따로 모아두고 사용
    - GOT : 런타임에 라이브러리 주소를 불러와 저장하고, 그것이 실제로 저장되는 공간