버퍼 오버런 종합보수세트

Tae Eun Kim

2022.03.24

Abstract

버퍼 오버런은 보안과 직접 연결되는 심각한 결함이다. 악성 사용자의 경우 버퍼 오버런을 통해 중요한 데이터를 훼손하거나 탈취할 수도 있기 때문이다. 다행히 버퍼 오버런만 목표로 한다면, 그 특성을 활용하여 효과적인 대응시스템을 고안할 수 있다. 특히 메모리 접근 분석 도구, 정적 분석기, 지향성 퍼저, 자동 프로그램 수정기와 같은 도구를 종합하여 적용한다면 버퍼 오버런으로부터 우리 프로그램을 지킬 수 있을 것이다.

버퍼 오버런(buffer overrun)은 빈번히 발생하면서도 위험성은 높기에 심각한 종류의 결함이다. 그 유명한 하트블리드 사태의 경우도 사실은 버퍼 오버런으로 인해 발생하였다. 버퍼 오버런이 발생하면 권한이 없는 메모리에 값을 쓰거나 읽어올 수 있게 되는데, 악성 사용자는 이를 의도적으로 발생시켜 중요한 데이터를 오염시키거나 탈취할 수도 있다. 따라서 버퍼 오버런의 위험을 확실하게 줄일 수 있는 방법이 필요하다.

이 때, 버퍼 오버런에 특화된 접근을 구상한다면 이에 더욱 효과적으로 대응할 수 있을 것이다. 일반적인 기능 오류와 달리, 버퍼 오버런의 발생은 메모리 접근 분석 도구를 활용해 잡아낼 수 있다. 따라서 이 점을 적극 활용하여 대응책을 구상해야 한다. 다음과 같은 도구 사슬을 생각해보자: 정적 분석기, 지향성 퍼저, 자동 프로그램 수정기.

먼저 정적 분석을 통해 버퍼 오버런이 발생할 가능성이 있는 지점을 찾는다. 그 후, 지향성 퍼저를 통해 의심되는 지점들을 차례대로 테스트하면 실제로 결함이 발생하는지에 대한 정보와, 결함을 발생시키는 입력을 찾을 수 있다. 이 때, 입력만 있으면 메모리 접근 분석 도구를 통해 버퍼 오버런 결함이 발생하는지 확인할 수 있다. 마지막으로 자동 프로그램 수정기로 결함이 발생한 위치를 수정하되, 그 전 단계인 퍼징 시에 생성된 입력들로 생성된 패치를 더욱 확실하게 검증할 수 있다.

위의 도구 사슬이 버퍼 오버런에 효과적으로 대응 할 수 있는 것은 세 도구의 시너지가 발생하기 때문이다. 각각의 도구는 한계가 존재하지만, 함께 사용하는 것이 이를 보완한다. 이렇게 각 도구의 한계를 파악하고, 서로 보완할 수 있는 도구를 조합할 수 있다면 버퍼 오버런과 같은 중요한 문제도 해결할 수 있을 것이다.