무선 통신 프로토콜 구현 결함의 위험성: 키스트로크 주입 공격 분석과 대응 방안

하소희[1], 김상엽[2], 박보경[3], 한성수[4]
[1,2,3]강원대학교 AI소프트웨어학과 학부생
[4]강원대학교 자유전공학부 교수

ihyraxi@gmail.com, x1111101101@gmail.com, b.gyung17@gmail.com, sshan1@kangwon.ac.kr

Risk of Wireless Protocol Implementation Defects: Analysis and Response to Keystroke Injection Attack

So-hee Ha[1], Sang-yeob Kim[2], Bo-gyung Park[3], Seong-soo Han[4]
[1,2,3]Dept. of Artificial Intelligence & Software, Kangwon National University
[4]Dept. of Liberal Studies, Kangwon National University

요 약

BLE 기술은 저전력과 효율적인 데이터 전송을 제공하지만, 통신 프로토콜의 결함을 악용한 키스트로크 주입 공격과 같은 보안 위협이 존재한다. 본 논문에서는 리눅스 커널의 BlueZ 패키지에서 발생한 CVE-2023-45866 취약점을 중심으로, 해당 공격의 원리와 심각성, 그리고 이를 해결하기 위한 패치를 분석한다. BLE 네트워크의 보안을 강화하기 위해서는 지속적인 연구와 대응이 필요하며, 보안 위협에 대한 대응 방안을 개발하고, 사용자의 보안 의식을 높이는 교육이 필수적임을 강조한다.

1. 서론

BLE(Bluetooth Low Energy) 기술은 저전력 소비와 효율적인 데이터 전송을 통해 다양한 사물인터넷(IoT) 기기 간의 원활한 연결을 지원하는 핵심 기술로 자리 잡고 있다[1]. 그러나 BLE의 저전력 통신 방식은 공격자가 취약한 연결을 노려 악용할 수 있는 다양한 보안 위협을 동반한다. 그 중 키스트로크 주입 공격(Keystroke Injection Attack)은 특히 위험한 유형으로, 공격자가 BLE 기기를 위장하여 피해자의 디바이스에 키 입력을 강제로 전송할 수 있는 치명적인 취약점이다. 이러한 공격은 주로 BLE 통신 프로토콜 구현의 결함을 이용해 발생하며, 적절한 보안 대책이 마련되지 않으면 시스템의 전체적인 안전성을 위협할 수 있다. 본 논문에서는 무선 통신 프로토콜에서 발생할 수 있는 이러한 공격의 위험성을 분석하고, 키스트로크 주입 공격에 대한 효과적인 대응 방안을 제시하고자 한다.

2. CVE-2023-45866 분석

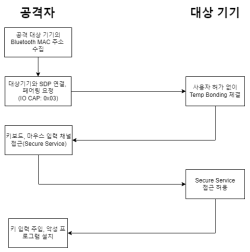
CVE-2023-45866은 프로토콜 스택 구현에서 발생한 결함으로, 리눅스 커널에서 블루투스 통신 스택을 구현하는 BlueZ 패키지에서 발견된 취약점이다. 이 결함은 공격자가 BLE 네트워크 상의 기기로 위장해 권한을 탈취하고, 사용자의 허가 없이 keystroke injection과 같은 공격을 수행할 수 있도록 한다. 특히, 이 취약점은 안드로이드, iOS, 리눅스(Ubuntu 등) 환경에서 모두 적용 가능하다. 본 연구에서는 안드로이드 플랫폼에서 CVE-2023-45866을 악용한 공격 과정과 그 심각성을 집중적으로 분석한다.

2.1. 공격원리

이 취약점은 안드로이드 블루투스 스택의 android/platform/bluetooth btm_sec.cc 파일 내의 결함으로 인해 발생한다. 안드로이드 블루투스 프로토콜에서는 일반적인 페어링 과정과 사용자의 허가 없이 이루어지는 temp bonding이라는 두 가지 페어링 방식이 존재한다.

일반 페어링의 경우, 사용자는 별도의 다이얼로그 창을 통해 연결 허가 절차를 밟게 된다. 그러나 temp bonding의 경우에는 사용자의 개입 없이 자동으로 연결이 이루어질 수 있다. 특히 페어링 과정에서 IO Capabilities(Input/Output Capabilities) 필드가 "NoInput/NoOutput"으로 설정될 경우, 안드로이드는 별도의 사용자 허가 절차 없이 temp bonding을 수행한다. 키보드와 마우스 입력 같은 서비스는 보통 보안 서비스(secure service)에 속하며, 이러한 서비스는 interrupt와 control 채널을 통해 통신한다. 그러나 btm_sec.cc 파일 내 구현상 결함으로 인해, temp bonding 상태의 기기를 포함한 페어링된 모든 기기의 경우 안드로이드는 이러한 보안 서비스를 무조건적으로 허용하는 구조를 가지고 있다. 공격자는 이 취약점을

악용하여, Pairing Feature Exchange 단계에서 자신의 기기의 IO Capabilities 필드를 "NoInput/NoOutput"으로 설정한 후, interrupt와 control 채널에 연결을 시도한다. 안드로이드는 이를 사용자 허가 없이 허용하며, 공격자는 기기와 연결된 후 키스트로크 주입 공격을 통해 악성 소프트웨어를 설치하고, 최종적으로 기기를 장악할 수 있게 된다.



(그림 1)

2.2. 심각성

CVE-2023-45866의 심각성은 안드로이드 기기의 기본적인 보안 체계를 무너뜨릴 수 있다는 점에서 크다. 공격자는 사용자의 허가 없이 temp bonding을 통해 안드로이드 기기에 접속할 수 있으며, 이를 통해 악성 명령을 실행하거나 중요한 데이터를 탈취할 수 있다. 특히, 키스트로크 주입 공격을 통해 공격자는 피해자의 디바이스에 원격으로 악성 소프트웨어를 설치하고, 시스템을 완전히 장악할 수 있는 심각한 보안 위협을 초래한다.

2.3. 패치 분석

이 취약점을 해결하기 위해, 안드로이드 블루투스 스택에 대한 패치가 적용되었다. 패치는 temp bonding을 통한 연결이 보안(security) 서비스에 접근할 수 없도록 하는 방식으로 이루어졌다. 구체적으로는 btm_sec_l2cap_access_req_by_requirement와 btm_sec_execute_procedure 함수에서 호출되어, temp bonding된 기기가 interrupt와 control 채널을 통한 보안 서비스에 접근하지 못하도록 차단하는 방식으로 수정되었다. 이를 통해 사용자의 허가 없이 이루어질 수 있는 temp bonding 연결이 제한되었고, 이를 악용한 keystroke injection 공격도 차단될 수 있게 되었다. 이 패치는 안드로이드 기기의 BLE 통신 프로토콜 구현에서 발생한 심각한 보안 결함을 수정함으로써, 기기의 안전성을 향상시키는 데 기여했다.

3. 결론

본 논문에서는 BLE(Bluetooth Low Energy) 기술의 보안 취약점을 악용한 키스트로크 주입 공격(Keystroke Injection Attack)의 위험성과, 이를 구체적으로 악용한 CVE-2023-45866 취약점을 분석하였다. BLE는 저전력 소비와 효율적인 통신을 제공하는 기술이지만, 그 특성상 temp bonding과 같은 자동 연결 절차가 보안의 취약점을 노출시킬 수 있다. 특히 CVE-2023-45866는 안드로이드 기기의 BLE 통신 프로토콜 구현에서 발생한 결함으로, 공격자가 악용할 경우 사용자의 허가 없이 기기에 접근해 악성 소프트웨어를 설치하고 시스템을 장악할 수 있는 심각한 보안 위협을 초래한다.

본 연구에서 분석한 바와 같이, 이 취약점은 안드로이드의 BLE 통신 스택에서 temp bonding 과정에서 발생하며, 공격자가 IO Capabilities 필드를 조작하여 사용자 개입 없이 키스트로크 주입 공격을 수행할 수 있음을 확인했다. 다행히도, 이에 대한 보안 패치가 이루어졌으며, temp bonding을 통한 보안 서비스 접근이 차단되었다. 이러한 패치는 BLE 통신에서 발생할 수 있는 유사한 보안 위협을 예방하는 데 중요한 역할을 한다.

따라서 본 연구는 BLE 기반 기기의 보안성을 향상시키기 위해 지속적인 취약점 분석과 프로토콜 개선이 필수적임을 강조하며, 특히 향후 IoT 기기에서 BLE 통신을 사용할 경우 보안 취약점을 사전에 방지하는 노력이 요구됨을 시사한다. BLE 기술의 지속적인 발전과 더불어 이에 상응하는 보안 강화 대책이 필요하며, 연구 결과는 관련 분야에서의 보안성을 높이는 기초 자료로 활용될 수 있을 것이다.

참고문헌

[1] Embedded Staff, Bluetooth low energy (BLE) fundamentals, embedded, 2016.10.18.

[2] NEGI, Arun; RATHORE, Santosh Singh; SADHYA, Debanjan. USB Keypress Injection Attack Detection via Free-Text Keystroke Dynamics. In: 2021 8th International Conference on Signal Processing and Integrated Networks (SPIN). IEEE, 2021. p. 681-685.

[3] SCHWARZ, Michael, et al. Keydrown: Eliminating software-based keystroke timing side-channel attacks. In: Network and Distributed System Security Symposium 2018. 2018. p. 15.