KIPS Tr. Comp. and Comm. Sys.
Vol.5, No.12 pp.481~490 pISSN: 2287-5891

https://doi.org/10.3745/KTCCS.2016.5.12.481

MITM Attack on Bluetooth Pairing in Passkey Entry Mode and Its Countermeasure

Jearyong Lee† · Wonsuk Choi†† · DongHoon Lee†††

ABSTRACT

Bluetooth utilizes a symmetric key that is exchanged at the first pairing to establish a secure channel. There are four authentication modes which enables device authentication, Just work, Passkey Entry, Out of Band, and Numeric Comparison. Up to now, Just work has been considered as the authentication mode that is vulnerable to Man-In-The-Middle (MITM) Attack. In addition, it is possible to intentionally change any authentication mode to Just work mode, in order to succeed in MITM Attack under Just work mode. However, this kind of attacks have just worked under the assumption that users should not notice that authentication mode was changed. In this paper, We analyze the specification of Secure Simple Pairing, LE Legacy Pairing and LE Secure Connection Pairing. When using Passkey Entry mode on each approach, it seems the MITM attack is possible. Also it offers Passkey Entry MITM attack that does not require assumptions about the user's fault, because it isn't change verification process of the authentication mode unlike traditional attacks. We implement the proposed MITM attacks. Also we presents a scenario in which an attack can be exploited and a countermeasure.

Keywords : Bluetooth, Pairing, Passkey Entry, MITM, IO Capability Exchange

블루투스 Passkey Entry 인증 모드에 대한 MITM 공격과 대응방법

이 재 령† · 최 원 석†† · 이 동 훈†††

요 약

블루투스는 대칭키를 사용하여 암호화 채널을 형성하며 대칭키는 최초 연결 이전에 수행하는 페어링 단계에서 교환된다. 페어링 단계에서 기기인증을 위한 인증 모드는 크게 Just work, Passkey Entry, Out of Band, 그리고 Numeric Comparison 방식으로 구분된다. 지금까지 블루투스 페어링 방식의 중간자 공격(MITM)은 Just work 모드를 대상으로 하거나 인증 모드를 강제로 Just work 모드로 변환한 상태에서 MITM 공격을 수행하였고, 이때에는 인증모드가 변환되었다는 것을 인지하지 못하는 사용자의 부주의가 가정되어야 했다. 본 논문에서는 Secure Simple Pairing, Le Legacy Pairing, 그리고 LE Secure Connection Pairing 방식의 규격을 분석하였고, 각 방식에서 Passkey Entry 모드를 사용하였을 때, MITM 공격이 발생 가능함을 보인다. 또한 제안하는 Passkey Entry MITM 공격 방법은 기존 공격방법과 달리 인증 모드의 사용자 확인 절차를 바꾸지 않기 때문에 사용자의 부주의에 대한 가정 사항을 요구하지 않는다. 우리는 제안하는 Passkey Entry MITM 공격을 구현하고 이것이 악용될 수 있는 시나리오와 이를 방어할 수 있는 대응방법을 제시한다.

키워드 : 블루투스, 페어링, 패스키 입력, 중간자공격, 입출력 능력 교환

1. 서 론

블루투스(Bluetooth)는 각종 전자기기 간 통신에 2.4 GHz의 주파수를 이용하여 10m 이내(Class2의 경우) 거리에서 데이터 교환을 하는 무선 디지털 통신 규격이다[1]. 초기 블루투스 기술은 유선으로 연결되어 있는 주변기기들을 무선으로 대체하기 위해 디자인되었다. 블루투스의 대표적인 경쟁 기술은 ZigBee와 Z-wave, Wifi 통신 기술 등이 있지만, 적은 전력소모와 빠른 전송속도를 동시에 만족하는 블루투스 기술이 최근 다양한 기기에서 사용되고 있다[2]. 특히, 스마트폰의 경우 배터리로 동작하고 입·출력 인터페이스를 최소화해야 한다는 제약조건이 있다. 이러한 이유로 모든 스마트폰 제조사가 블루투스 기술을 채택하여 자신들의 스마트 폰에 탑재하고 있다. 2010년에는 IoT (Internet of Things) 기

※ 본 연구는 미래창조과학부 및 한국인터넷진흥원의 "2016년 고용계약형 정보보호 석사과정 지원사업"의 연구결과로 수행되었음.
† 준 회 원 : 고려대학교 정보보호대학원 금융보안학과 석사과정
†† 비 회 원 : 고려대학교 정보보호대학원 정보보호학과 박사과정
††† 정 회 원 : 고려대학교 정보보호대학원 교수
Manuscript Received : September 8, 2016
Accepted : October 9, 2016
* Corresponding Author : DongHoon Lee(donghlee@korea.ac.kr)

기를 겨냥한 v4.0 BLE (Bluetooth Low Energy) 규격이 새롭게 발표되었다. 이에 향후 IoT기기가 사용되는 수많은 애플리케이션에 블루투스 기술이 활용될 것으로 예상되고 있다[3].

일반적으로 여러 애플리케이션들은 블루투스 기술을 이용하여 채널을 형성한다. 이렇게 형성된 채널을 통하여 데이터를 교환하거나 기기를 제어하는 명령어(Command)를 전송한다. 블루투스 기술은 암호화 통신을 통하여 채널을 안전하게 보호한다. 암호화 통신에 사용되는 비밀키를 두 기기가 서로 교환하기 위한 블루투스 기술은 LE Legacy Pairing (LE LP), Secure Simple Pairing (SSP), 또는 LE Secure Connection Pairing (LE SCP) 3가지로 구분할 수 있다.[1] LE LP, SSP, LE SCP 들은 모두 중간자(Man In The Middle, MITM) 공격을 방어하기 위해 기기에 부착된 인터페이스(e.g., 키보드 또는 디스플레이)를 이용한 사용자 확인 과정이 존재 한다[4]. 두 기기의 인터페이스 종류에 따라 4가지 확인 방법이 가능하며, 이 중 Just work 방식을 제외한 나머지 3가지 방법(Numeric Comparison, Passkey Entry, Out of Band)은 모두 MITM 공격에 안전하다고 규격에 명시되어 있다[5]. 하지만, 4가지 모드 중 한 가지 모드를 선택하는 과정인 I/O Capability Exchange 단계에서 공격자가 I/O Capability에 대한 메시지 조작을 통해 실제 입·출력 능력과 상관없이 원하는 모드로 동작하게 만들 수 있다. 즉, MITM 공격에 안전하지 않은 Just work 모드로 동작하도록 공격자가 메시지를 조작할 수 있다. 이러한 사실을 이용하여, 사용자 입장에서 Just work 방식과 작동 방식의 차이를 구분할 수 없는 Out of Band 방식으로 동작하는 경우 MITM 공격이 가능하다는 연구결과가 있었다[6]. Numeric Comparison과 Passkey Entry 방식의 경우, Just work 방식과 사용자가 확인하는 절차가 명확히 다르기 때문에, MITM 공격을 수행하는데 제한이 있었다. 본 논문에서는 Keijo와 Toivanen[7] 연구를 확장하여 블루투스 각 버전별 규격을 연구하였으며, 해당 버전에 맞춰 Passkey Entry 모드에서도 MITM 공격이 가능함을 밝히고 구현을 통한 수행결과를 제시한다. 본 논문을 통해 도출되는 주요 결과는 다음과 같다.

① 블루투스 규격의 3가지 페어링 방식인 LE LP, SSP, 그리고 LE SCP에서 모두 사용되는 Passkey Entry 모드에 대하여 적용 가능한 MITM 공격 방법 제안
② 스마트기기와 블루투스 키보드 간에 MITM 공격방법 제안 및 이에 따른 구현
③ 제안하는 공격방법에 대한 대응방법 제시

2. 배경 지식

보안관점에서, 블루투스 규격은 크게 4가지(Legacy Pairing, Secure Simple Pairing, LE Legacy Pairing, LE Secure Connection Pairing)로 구분할 수 있다. 블루투스 v1.0부터 v2.0까지 사용되었던 Legacy Pairing의 경우, Link key라 불리는 비밀 키가 4자리 PIN 코드로부터 생성되었기 때문에, 전수조사 공격에 매우 취약하였다. 이러한 이유로 Legacy Pairing 방법은 현재 거의 사용되지 않고 있으며, 본 논문에서는 Legacy Pairing 방법을 제외한 나머지 3가지 방법에 대하여 설명한다. SSP, LE SCP 방법은 타원곡선 디피헬만(Elliptic Curve Diffie-Hellman) 키 교환 방식을 이용하여 안전한 채널을 형성한다. 하지만 디피헬만 키 교환 방식은 일반적으로 MITM 공격에 안전하지 않기 때문에 각 페어링 방법은 기기의 입·출력 능력에 따라 추가 인증단계를 갖는다[8]. LE Legacy Pairing의 경우, MITM 공격에 안전하기 위하여 키 교환 시 비밀 키의 시드(Seed) 값을 사용자 인터페이스를 통해 교환한다. 하지만, 이와 같이 MITM 공격에 안전하기 위한 설계에도 불구하고, 우리는 I/O Capability Exchange 단계의 취약점을 이용하여 각 페어링 방법에서 MITM 공격이 가능함을 보인다. 이번 장에서는 우리가 제안하는 공격방법에 대한 이해를 위해, 각 페어링 방법에서 사용하고 있는 I/O Capability Exchange 절차에 대해 설명한다.

2.1 I/O Capability Exchange

1999년에 블루투스 v1.0 기술이 처음 소개된 이후 현재 v4.2 BLE (Bluetooth Low Energy)까지, 블루투스 SIG (Special Interest Group)은 계속해서 새로운 규격을 발표하고 있다. 블루투스에서 사용하는 페어링 방식은 MITM 공격으로부터 안전하기 위해 기기 인터페이스를 통해 사용자 확인을 요구하는 절차를 포함하고 있다. 이 추가절차는 기기들이 가지고 있는 입·출력 능력에 따라 결정된다. 일반적으로 기기의 입·출력 능력은 크게 4가지(DisplayOnly, DisplayYesNo, KeyboardOnly, NoInputNoOutput)로 구분하며, v4.2 BLE에서 사용하고 있는 LE SCP의 경우에는 KeyboardDisplay 능력이 추가되었다. Table 1은 I/O Capability Exchange 단계

Table 1. I/O Capability

	Description
Display YesNo	Device has the ability to display or communicate a 6 digit decimal number. Device has at least two buttons that can be easily mapped to 'yes' and 'no'.
Display Only	Device has the ability to display or communicate a 6 digit decimal number.
Keyboard Only	Device has a numeric keyboard that can input the numbers '0' through '9' and a confirmation.
NoInput NoOutput	Device does not have any ability.
KeyboardDisplay	Device has the ability to display or communicate a 6 digit decimal number. Device has a numeric keyboard that can input the numbers '0' through '9' and a confirmation. (It exists only LE Secure Connection Pairing.)

1) 본 논문에서는 초기 버전의 블루투스에서 사용한 Legacy Pairing 방식은 고려하지 않는다.

에서 정의하고 있는 각 입·출력 능력에 대해 설명하고 있다. 기기들의 입·출력 능력에 따라 MITM 공격에 안전하기 위한 추가 절차가 결정된다. 각 페어링 방법에 따라 해당 추가 절차는 상이하며 이에 대한 자세한 설명은 다음과 같다.

1) Secure Simple Pairing

Secure Simple Pairing 절차는 Public Key Exchange 단계에서 두 기기가 192-bit 또는 256-bit 크기의 대칭키인 DHKey를 서로 교환한다. 그리고 Authentication 1 stage 단계에서 MITM 공격이 발생하였는지 여부를 확인한다. 이를 확인하기 위하여 Secure Simple Pairing 방법에서는 전체 3가지 종류(Just Work, Passkey Entry, Numeric Comparison)의 확인 절차가 사용되고 있으며 각 절차에 대한 설명은 Table 2와 같다.

Table 2. Association Models

	Description
Just Work	The user is never shown a number and the application may simply ask the userto accept the connection.
Passkey Entry	The user is shown a six digit number (from "000000" to "999999") on thedevice with a display, and is then asked to enter the number on the otherdevice. If the value entered on the second device is correct, the pairing issuccessful.
Numeric Comparison	The user is shown a six digit number (from "000000" to "999999") on bothdisplays and then asked whether the numbers are the same on both devices. If "yes" is entered on both devices, the pairing is successful.

Table 3. Authentication Method Selection Rules

		Device A			
		Display Only	Display YesNo	Keyboard Only	NoInput NoOutput
Device B	Display Only	JW	JW	PE	JW
	Display YesNo	JW	NC	PE	JW
	Keyboard Only	PE	PE	PE	JW
	NoInput NoOutput	JW	JW	JW	JW

JW: Just Work
PE : Passkey Entry
NC : Numeric Comparison

그리고 3가지 확인 방법 중 Just work 방식의 경우 MITM 공격을 방어할 수 없다는 점이 블루투스 규격에 명시되어 있다. 또한 Passkey Entry, Numeric Comparison 방식은 도청과 MITM 공격에 대한 방어가 가능함이 규격에 명시되어 있다[5].

Secure Simple Pairing에서 MITM 공격을 막기 위한 3가지 타입의 추가 절차는 I/O Capability Exchange 단계에서 교환된 기기의 입·출력 능력에 따라 Table 3과 같이 결정된다.

Authentication 1 stage 단계 이후 DHKey 확인과 Link Key 생성 절차를 거쳐 암호화 채널을 생성한다. Fig. 1은 Secure Simple Pairing의 전제 과정을 보여주고 있다.

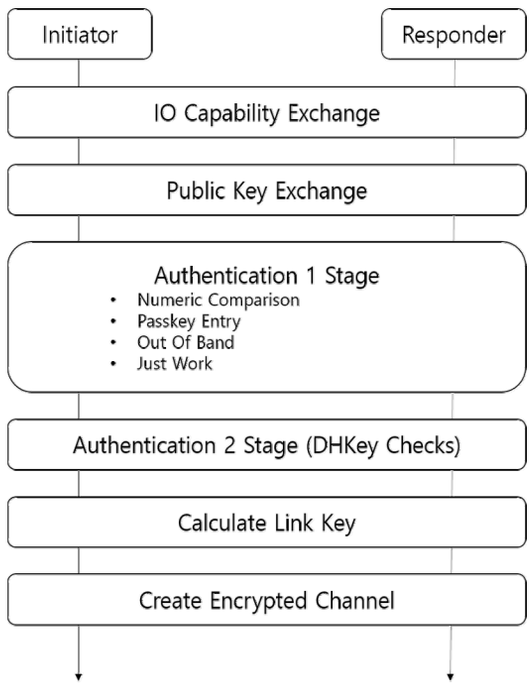


Fig. 1. Secure Simple Pairing Process

2) LE Secure Connection Pairing

Fig. 2는 LE SCP의 전체 페어링 절차를 보여주고 있으며, 이는 이후에 설명할 LE LP도 동일한 절차를 수행한다. LE SCP에서는 위에서 설명한 SSP의 I/O Capability Exchange 단계에서 수행하던 작업이 phase1의 Pairing_Request, Pairing_Response 패킷에 포함되어 동작한다. phase2에서는 각 페어링 방식과 입출력 능력에 따라 각자 다른 방식으로 MITM 공격 여부 검증과 비밀 키 생성 과정을 진행한다.

LE Secure Connection Pairing의 Phase2는 Secure Simple Pairing과 유사하다. LE SCP는 타원곡선 상에서의 디피헬만 키 교환을 통하여 256-bit DHKey를 공유한다. 이후 Authentication 1 stage 단계에서 MITM 공격 발생여부를 확인한다. 이때 사용되는 방법은 Secure Simple Pairing와 동일하게 Table 2의 3가지 방법 중 1가지 인증 방법을 선택하며, 이 때 사용되는 인증 방법은 두 기기의 입·출력 능력 조합에 의해서 결정된다. 그리고 KeyboarDisplay라는 입·출력 능력이 추가됨에 따라 두 기기의 입·출력 능력 조합에 따른 인증모드 결정 방식이 Table 4와 같이 추가되었다.

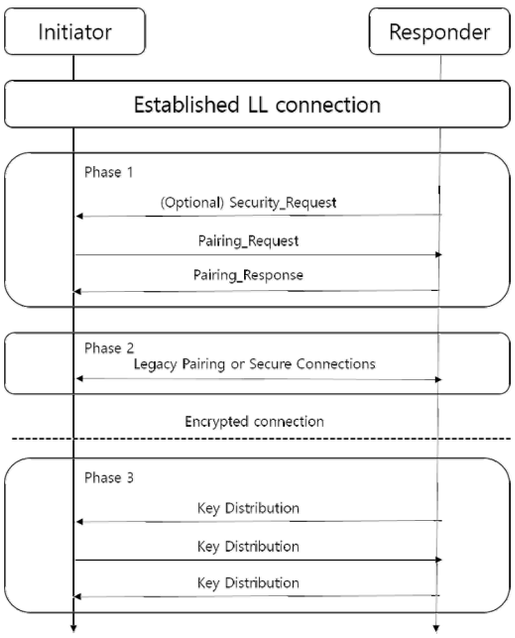


Fig. 2. LE Pairing Process

Table 4. Additional Authentication Method Selection Rules for the Keyboard Display

	Display Only	Display YesNo	Keyboard Only	NoInput NoOutput	keyboard Display
keyboard Display	PE	NC	PE	JW	NC

JW: Just Work
PE : Passkey Entry
NC : Numeric Comparison

3) LE Legacy Pairing

앞서 언급한대로 LE Legacy Pairing은 Fig. 2의 LE SCP와 동일한 절차로 수행된다. 하지만 Phase2 단계에서 디피헬만 키 교환 방식을 사용한 LE SCP와 다르게 LE LP는 독자적인 키 교환 알고리즘을 사용한다. 그리고 MITM 공격을 막기 위한 인증 방법 중 하나인 NC 방법이 제외되고 나머지 3가지(Just Work, Passkey Entry, OOB) 방식만을 제공한다. Fig. 3은 LE LP의 비밀키인 Short-term Key (STK)를 교환하는 Phase2 단계를 보여주고 있다.

LE LP의 Just work, Passkey Entry, OOB의 3가지 인증 방식에는 동일한 알고리즘이 사용되며, 비밀 키의 시드 값인 TK를 생성하는 방법으로 구분된다. Just Work 일 경우 TK는 0을 사용하고 Passkey Entry에서는 사용자가 입력하는 6자리 정수를 20-bit로 표현하여 사용한다. OOB의 경우 128-bit 난수를 별도의 채널을 통해 전달한다. 여기서 Just Work는 도청과 MITM 공격에 대하여 안전하지 않으며, Passkey Entry의 경우 20-bit 엔트로피를 가지기 때문에 제한적인 안전성을 제공한다. OOB는 도청에 대한 방어와 MITM공격에 대한 방어를 모두 제공한다.

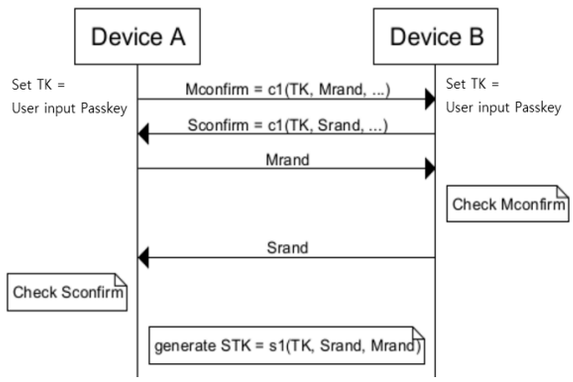


Fig. 3. Phase2 of LE Legacy Pairing

3. 공격 모델

본 장에서는 SSP, LE SCP, 그리고 LE LP 3가지 페어링 방식에 모두 사용되고 있는 인증 방식인 PE 모드에 대한 새로운 MITM 공격 방법을 제안한다. Fig. 4는 본 논문에서 제안하는 PE 모드에 적용되는 새로운 MITM 공격에 대한 전체 흐름을 보여주고 있다. 일반적으로 PE 모드는 키보드와 같이 0부터 9까지의 숫자 입력이 가능한 기기가 사용할 수 있는 인증 방식이기 때문에, 우리는 블루투스 키보드를 예로 들어 제안하는 공격 방법을 설명한다.

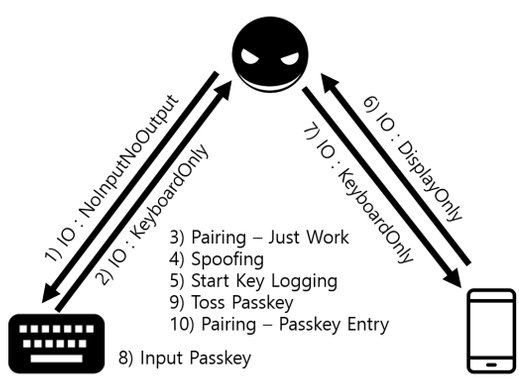


Fig. 4. Scenario of the MITM Attack

3.1 공격 가정 사항

I/O capability exchange 단계의 취약점을 이용하여 블루투스 페어링에 대한 기존 MITM 공격은 사용자의 부주의를 가정하고 수행 된다[6]. 예를 들어, NC 모드로 동작해야 하는 페어링 과정에서 공격자는 I/O capability exchange 단계의 메시지 조작을 통하여 JW 모드 변경을 통한 MITM 공격이 있을 수 있다. 이 때, 사용자는 NC 모드와 JW 모드의 차이점을 인지하지 못한다는 가정 사항을 말한다. 하지만, 블루투스 페어링의 4가지 인증 모드에 익숙한 사용자라면 이를 쉽게 구분할 수 있다. 우리는 보다 현실적인 MITM 공격을 위하여, 4가지 인증 모드에 익숙한 희생자를 가정한

다. 즉, 인증 방식이 비정상적으로 변경된 경우 MITM 공격으로 간주하고 페어링을 바로 해제 할 수 있다.

또 다른 가정 사항으로 희생자는 Passkey 입력 이전에 키보드와 MITM 공격자가 JW 모드로 페어링이 완료된 것을 인지하지 못한다. 블루투스 규격에는 페어링이 완료된 이후 주변 기기가 사용자에게 페어링 종료를 나타내는 방식에 대하여 명시하지 않았다. 게다가 블루투스 키보드의 경우 페어링 완료를 나타낼 수 있는 출력장치가 제한되어 있다. 따라서 실제로 희생자가 페어링이 완료되었다는 것을 인지하기는 매우 어렵다. 예를 들어 LED가 탑재되어 있는 키보드의 경우, 페어링 완료 시와 정상 연결 상태를 나타내는 LED 패턴은 제품마다 다양하다. 때문에 이를 구분하는 것은 어렵고 이를 숙지하고 있는 사용자는 매우 드물다. 그러므로 사용자가 페어링을 수행할 때 자신이 아직 Passkey를 입력하기도 전에 키보드가 공격자와 페어링을 완료했다는 사실을 인지하기 어렵다.

마지막으로, 정상적인 페어링을 수행한 사용자라 할지라도 Jamming과 같은 DoS (Denial of Service) 공격에 의해 해당 채널이 단절될 수 있다. 이 때 사용자는 아무런 의심 없이 페어링을 다시 수행한다고 가정한다.

3.2 공격자 모델

본 절에서는 제안하는 공격 방법에 대한 이해를 돕기 위해 공격자 모델에 대하여 설명한다. 먼저, 우리는 공격자의 목적을 크게 2가지로 구분한다. 첫 번째 공격자의 목적은 PE 모드를 이용하여 페어링 되는 두 기기의 통신 채널에서 전송되는 메시지를 도청하는 것이다. 두 번째 목적은 사용자의 입력 장치로 위장 공격을 수행하여 사용자 기기를 제어하는 것이다. 블루투스 키보드와 스마트 기기는 대표적인 PE모드를 이용하여 페어링되는 예 이다. 이 때, 공격자는 사용자가 블루투스 키보드를 이용하여 입력하는 내용을 모두 확인할 수 있다. 심지어 블루투스 키보드로 위장하여 스마트 기기를 임의로 조작할 수도 있다.

제안하는 공격방법을 수행하기 위해서 공격자는 공격대상이 되는 블루투스 기기의 이름을 사전에 알고 있어야 한다. 블루투스 페어링은 자주 일어나지 않기 때문에, 공격대상 기기 하나만이 discoverable 모드인 경우가 일반적이다. 만약, 다수의 기기가 discoverable 모드인 경우라도 사회공학적 기법을 통하여 블루투스 기기의 이름을 쉽게 유추할 수 있다.

블루투스 기기의 통신 가능 거리는 10m 이내이기 때문에, 제안하는 공격 방법 또한 10m 이내에 수행되어야 한다. 하지만, Bluesnifer rifle과 같은 확장 안테나를 이용하여 블루투스 통신 가능 거리를 최대 1km 까지 확장할 수 있다 [9]. 이는 우리가 제안하는 공격 방법도 1km 밖에서 수행 가능함을 의미한다.

3.3 제안하는 공격 방법

본 절에서는 제안하는 공격 방법에 대하여 설명한다. 제안하는 공격 방법은 두 기기가 PE 모드를 이용하여 페어링 되는 경우에 적용 가능한 MITM 공격 방법이다. 일반적으로 두 기기 중 하나는 키보드와 같은 사용자 입력장치가 되고 나머지는 다양한 형태의 기기가 될 수 있다. 그 중 스마트기기가 가장 대표적인 예가 된다. 제안하는 공격 방법을 설명하기 위하여, 우리는 두 기기를 사용자 입력장치와 사용자 기기로 구분한다. 이를 통하여, 공격 방법을 다음과 같은 단계로 나누어 설명한다.

단계 1 : 사용자 입력 장치와 페어링
단계 2 : 사용자 입력 장치의 위장
단계 3 : 사용자 입력의 키로깅
단계 4 : 사용자 기기와 페어링 단계

단계 1 : 사용자 입력 장치와 페어링

공격 대상이 되는 사용자의 입력 장치는 discoverable 모드로 설정되어 있다. 공격대상의 주소는 일반적인 블루투스 동글을 통해 inquery를 수행하여 주소를 얻어올 수 있다. 블루투스 주소를 알고 있다면 언제든지 추가 명령을 통해 이름과 CoD (class of device)를 얻어올 수 있다. 사용자 기기가 inquery 요청을 브로드캐스트 하면 이를 수신하고 사용자 기기에게 기기 이름을 전송한다. 공격자는 이러한 사실을 이용하여 사용자 기기보다 먼저 입력 장치에게 inquery 요청을 보내고 응답을 받는다. 그리고 공격자는 사용자 입력 장치와 페어링 절차를 수행한다. 이 때, 사용자 입력 장치에는 페어링 진행여부를 나타내는 출력장치가 불충분하기 때문에 사용자가 이를 인지하기 어렵다. 공격자는 사용자 입력 장치와 페어링 과정을 진행할 때 자신의 I/O capability 메시지를 NoInputNoOutput으로 조작하여 전송한다. 이 때, 사용자 입력 장치는 KeyboardOnly 능력을 가지고 있다. 하지만 2장에서 설명한 인증 모드 결정 규칙에 따라 사용자 입력 장치와 공격자의 기기는 JW 모드를 이용하여 페어링 된다.

단계 2 : 사용자 입력 장치로 위장

앞 단계에서 사용자 입력 장치와 공격자 기기가 JW 모드를 이용하여 페어링이 완료되었기 때문에, 사용자 입력 장치는 discoverable 모드를 종료하게 된다. 이후 공격자는 앞서 얻어온 이름과 주소, CoD를 사용하여 사용자 입력 장치로 위장하고 사용자 기기의 inquery 요청에 대신 응답한다. 즉, 공격자의 기기가 사용자 입력 장치로 위장하여 discoverable 모드로 대기한다. 사용자가 스마트 기기에 블루투스를 활성화 하면 자동으로 inquery 요청을 브로드캐스트 하고 응답을 페어링 가능 리스트에 출력한다. 출력된 페어링 가능 기기의 정보는 공격자의 장치이지만 자신의 장치와 정보가 완전히 동일하기 때문에 사용자는 공격자의 기기에 페어링을 요청한다.

단계 3 : 사용자 입력의 키로깅

앞 단계에서 사용자 입력 장치와 공격자 기기는 이미 JW 모드로 페어링 되었고, 앞서 설명한 이유로 인해 사용자가

이를 인지하는 것은 매우 어렵다. 공격자는 페어링이 완료된 입력 장치로부터 제공되는 HID 프로파일에 접속하여 모든 입력을 키로깅하기 시작한다. 사용자는 아직 입력 장치가 자신의 스마트 기기와 페어링 되지 않았으므로 아무런 입력을 넣지 않는다. 따라서 공격자는 최초로 입력되는 숫자 6자리와 엔터키를 사용자가 입력한 Passkey임을 유추할 수 있다. 만약 실수로 다른 숫자가 먼저 입력되었더라도 엔터키를 기준으로 숫자 6자를 골라낼 수 있으며 엔터가 입력되었더라도 앞선 6자리가 숫자가 아니라면 무시할 수 있다.

단계 4 : 사용자 기기와 페어링

사용자는 공격자의 기기를 페어링 가능 리스트에서 선택하여 페어링을 시도한다. 공격자는 사용자의 입력 장치를 위장하고 있으므로 자신의 I/O capability 메시지를 KeyboardOnly로 변조하여 전달한다. 앞서 설명한 인증 모드 결정 규칙에 따라 PE모드로 페어링이 진행되고 키보드 입력을 넣는 단계에서 대기한다. 사용자의 시각에서는 예정대로 PE모드로 페어링이 동작하기 때문에 공격을 인지하지 못한다. 사용자의 스마트 기기에는 Passkey가 출력되고 사용자는 그것을 보고 자신의 입력기기를 통해 입력한다. 공격자은 해당 입력을 통해 Passkey를 알 수 있다. 따라서 이를 사용하여 검증단계를 통과하고 PE모드로 페어링을 완수 할 수 있다.

이와 같이 사용자 입력 장치와 사용자 기기에 MITM 공격을 성공하게 되면, 키보드로부터 들어오는 모든 입력을 키로깅 할 수 있다. 또한 스마트 기기와 같은 사용자 기기에 대하여 자신이 생성한 임의의 키보드 입력을 넣을 수 있다. 이를 악용하여 공격자는 희생자가 잠시 한눈을 팔고 있는 사이에 키보드 입력만으로 악성 앱 배포 서버로부터 악성 앱을 다운로드 받아 설치 및 실행 할 수 있다. 필요하다면 사용자 기기의 보안 설정을 변경하는 것도 가능하다.

4. 구 현

이번 장에서는 앞서 설명한 공격 방법에 대한 상세 구현에 대한 설명한다. 먼저 제안하는 공격방법을 구현한 개발환경에 대하여 설명한다. 이후 공격 방법의 핵심이 되는 I/O capability exchange 메시지 조작, 사용자 입력 장치로 위장, 키 로깅, 사용자 기기에 악성 앱 설치에 대한 구현 방법을 설명한다.

4.1 개발 환경

가장 먼저 공격 대상이 되는 사용자 입력 기기와 사용자 기기는 블루투스 키보드와 스마트 폰을 각각 사용하였다. 블루투스 키보드를 구현하기 위하여, Raspberry Pi 2 model B, NEXT-204BT Bluetooth dongle, 그리고 유선 키보들 함께 조합하였다. 스마트 폰의 경우에는 Andoroid OS 6.0.1 버전이 탑재되어 있는 Nexus 5x 모델을 사용하였다. 상용 블루투스 키보드를 사용하지 않은 이유는 다음과 같다. 일반적으로 블루투스 키보드 제조회사는 자신들의 키보드가 어떠한 종류의 페어링을 수행하는지 명시하지 않기 때문에 적합한 블루투스 키보드를 선택하는데 어려움이 있었다. 게다가 제안하는 공격 방법은 SSP와 LE LP 페어링 2가지 방식에 적용 가능하다. 2가지 페어링 방식을 모두 제공하는 블루투스 기기는 존재하지 않기 때문에 블루투스 키보드 에뮬레이터를 각 페어링 버전에 맞게 직접 구현하였다. 공격자가 사용하는 공격 기기의 경우 Raspberry Pi 2 model B와 NEXT-204BT Bluetooth dongle를 사용하였다.

Raspberry Pi의 OS는 라즈비안 리눅스를 설치하였으며, 블루투스 스택 bluez5 [10]에서 제공하는 API를 이용하여 블루투스 표준에 맞게 개발 환경을 구축하였다. Fig. 5는 제안하는 공격 방법 구현을 위한 개발환경을 보여주고 있다.

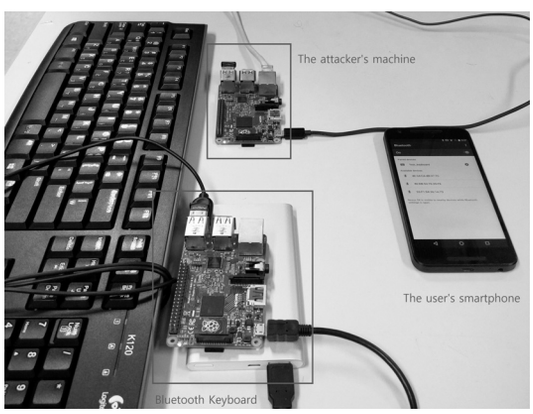


Fig. 5. Setup for MITM Attack on Bluetooth

4.2 I/O Capability Exchange 메시지 조작

2장에서 설명한 인증 모드 선택 규칙은 페어링되는 두 기기의 I/O capability에 의해 결정된다. 제안하는 공격방법의 기본 개념은 PE 인증 모드를 이용하여 페어링되는 블루투스 키보드와 스마트 폰 사이의 MITM 공격 방법이다. MITM 공격을 위해 블루투스 키보드와 공격기기와는 JW 모드를 이용하여 페어링되고 스마트폰과 공격기기와는 PE 모드를 이용하여 페어링되는 것이다. 이를 위하여, 공격자는 블루투스 키보드에게는 자신의 I/O capability를 NoInputNoOuput으로 전송하고 스마트폰에게는 KeyboardOnly로 전송하여야 한다.

우리는 먼저 bluez5에서 제공하는 클래스중 하나인 “AgentManager1”의 “RegisterAgent” 메소드를 사용하였다. 해당 메소드의 파라미터에 capability를 전달할 수 있으며 우리가 원하는 I/O capability을 가진 Agent객체를 생성하여 시스템에 등록할 수 있다. 이 때, 블루투스 키보드를 위한 Agent와 스마트 폰을 위한 Agent로 구분할 수 있다. 블루투스 키보드를 위한 Agent 생성 이후, “Device1” 클래스의 “Pair”메소드를 통하여 블루투스 키보드에 페어링을 요청할 수 있다. 그리고 스마트 폰을 위한 Agent의 경우에는 페어링이 완료된 블루투스 키보드로부터 Passkey를 입력받아 스마트 폰에 전달해주어야 한다. 실제 사용자는 이러한 사실을 인지하지 못하고 있으며, 블루투스 키보드를 이용한 Passkey에 대한 입력이 바로 스마트 폰으로 전달된다고 생각

한다. 이는 Agent 클래스의 “RequestPasskey” 메소드를 재정의 하여 구현하였다. 페어링 대기상태에서 요청받은 페어링을 수행하는 것은 “AgentManager1” 클래스의 “RequestDefaultAgent” 메소드를 이용하였다. 등록한 KeyboardOnly 능력의 Agent를 페어링 요청을 받았을 시 동작하는 디폴트 Agent로써 설정해야 한다. 결론적으로 블루투스 키보드와 공격기기는 JW모드를 이용하여 페어링을 완료하게 된다. 페어링 요청을 하는 스마트 폰의 경우에는 PE모드를 사용하여 페어링이 진행되며 사용자 입력을 요구하는 단계에서 사용자 입력을 기다린다. 사용자로부터 유효 값(즉, Passkey)이 입력된 경우 페어링이 완료된다.

4.3 사용자 입력 장치로 위장

키보드와 JW모드로 페어링을 완료한 이후 스마트 폰과 페어링을 진행하기 위해서는 스마트 폰에서 공격자 기기로 페어링을 요청 해줘야한다. 공격자는 사용자를 속여 페어링 요청을 받기위해 미리 얻어온 사용자의 키보드 정보를 사용하여 공격자 기기를 위장해야 한다. 또한 이러한 일련의 작업들은 사용자의 스마트 폰 보다 빠르게 진행되어야 한다. inquiry 스캔 방식은 동기식과 비동기식이 있는데 단순 정보 획득에는 비동기식이 훨씬 빠르다. 안드로이드 폰의 블루투스 기기 검색 기능은 동기식 스캔을 사용한다. 안드로이드 기기에서 정상 기기를 찾는 것 보다 먼저 공격기기가 비동기식 스캔을 사용하여 키보드에 페어링을 먼저 맺을 수 있다. 또한 사용자는 스마트 폰의 블루투스 설정을 사용하여 사용자의 기기를 선택 해주어야 하기 때문에 공격기기가 자동으로 페어링을 맺고 위장하는 속도보다 빠를 수 없다.

블루투스 파이썬 확장 모듈에는 이러한 목적을 달성하기위한 충분한 기능이 존재한다. 확장모듈에서 제공하는 “DeviceDiscoverer” 클래스를 상속받아 “device_discovered” 콜백 메소드를 재정의 해야 한다. 우리는 원하는 이름을 가진 키보드 장치에 대하여 비동기적인 inquery를 통해 스마트 폰보다 먼저 페어링 작업을 완료할 수 있다. “DeviceDiscoverer” 클래스 내의 “device_discovered” 콜백 메소드는 장치를 발견하였을 때 호출되며 주소, 이름, 신호세기, 디바이스 종류를 인자로 받는다. 키보드의 CoD는 10진수로 9536이고 이것과 장치 이름을 조건으로 추가하여 원하는 장치만 특정할 수 있다. 이후 얻어온 정보를 사용하여 사용자 블루투스 키보드와 페어링을 완료하고 사용자 스마트 폰의 inquiry에 공격자가 위장하여 응답한다. 우리는 이것을 위해 “hciconfig”라고 하는 리눅스의 기본적인 블루투스 관리 프로그램을 사용하였다. 해당 프로그램을 사용하여 공격자의 블루투스 모듈의 이름과 주소, CoD를 바꾸고 discoverable 모드로 대기할 수 있다.

4.4 키로깅

공격자의 기기가 정상적으로 사용자의 블루투스 키보드와 페어링을 마치면 키보드는 공격자 기기의 “/dev/input/event0”에 등록된다. 우리는 키보드에서 들어오는 입력을 확인할 수 있으며, 공격자가 원하는 정보로 변환하여 스마트 폰에 전달 할 수 있다.

블루투스 키보드는 사용자 입력을 스마트 폰으로 전달하기 위해 블루투스 버전에 따라 HID(Human Interface Device) 또는 HOG(HID Over GATT) 프로파일을 제공한다. LE에서는 HOG 프로파일을 사용하고 아니라면 HID프로파일을 사용한다. BR/EDR과 LE 버전 둘 다 API 차이만 있을 뿐 기본적으로 수행하는 동작은 동일하다. 공격자는 HID, HOG 서비스를 스마트 폰에 제공할 수 있다. 안드로이드 폰은 기기에서 제공되는 프로파일을 외부에 알리고 그것을 확인할 수 있는 SDP (Service Discovery Protocol)를 지원한다. 이를 통해 상대 기기에서 제공하는 서비스를 검색하고 HID, HOG, A2DP 등의 프로파일에 어떠한 사용자 확인도 없이 먼저 접속을 시도한다. 따라서 공격자는 합법적으로 스마트 폰에 키보드 입력을 넣을 수 있는 능력을 갖는다. 우리는 “/dev/input/event0”로부터 들어오는 입력을 HID또는 HOG 프로파일 서비스에 맞게 변환하여 전달하는 프로그램을 제작하였다. 또한 SDP 를 이용하여 스마트 폰이 스스로 HID 프로파일 서비스에 접속하도록 하였다. 최초의 Passkey 입력를 키로깅하여 알아내고 페어링을 통과하면 공격자 기기는 사용자의 키보드 입력을 읽고 사용자의 스마트 폰에 입력할 수 있는 완전한 권한을 가지게 된다.

4.5 사용자 기기에 악성 앱 설치

앞에서 PE페어링을 우회 할 수 있는 방안을 보였다. 이번에는 페어링을 우회한 이후 공격자가 어떤 공격을 할 수 있는가에 대하여 보인다.

HID Keyboard Attack은 키보드로 인식되는 악성 usb를 사용하여 정해진 입력을 PC에 입력하는 공격이다. 주로 cmd, powershell 등을 사용하여 백도어 스크립트를 입력하여 실행 시킨다. 키보드가 처음 연결 되었을 시 사용자 확인없이 PC에서 바로 동작하는 점을 악용한다. 이는 스마트 폰에서도 마찬가지로 수행될 수 있다. 우리는 사용자 스마트 폰을 HID 프로파일에 접속시키고 HID Keyboard Attack을 블루투스 채널을 통하여 수행할 것이다. 이를 이용한 가장 심각한 공격은 키보드 입력만을 통해 스마트 폰에 임의로 제작한 악성 앱을 설치하는 것이다. 안드로이드 롤리팝 이상부터 안드로이드 레퍼런스 장치의 기본 런처로 “구글나우런처”가 설치된다. “구글나우런처”는 보다 편리한 검색을 돕는 많은 기능을 가지고 있다. 다르게 말하면 공격자는 이것을 사용하여 보다 쉽게 스마트 폰을 공격할 수 있다. “구글나우런처”는 홈 화면에서 키보드로 앱의 이름을 입력하면 해당 앱을 실행할 수 있고 url을 입력하면 해당 url에 바로 접속할 수 있다. 공격자는 단축키를 이용하여 홈 화면으로 이동하고 “setting”을 입력하여 설정 창으로 진입할 수 있다. 설정창의 기본 순서는 OS버전별로 정해져 있으므로 공격자는 방향키 입력을 눌러 “보안” 탭 내부의 “알수 없는 소스 설치” 항목을 설정할 수 있다. 그 후 다시 홈 화면에서 url을 입력하여 악성 앱을 다운로드 받는다. 이 때 브라우저에서

다운로드 여부를 확인하는 창을 띄우는데 Tap키와 방향키를 이용하여 “yes”을 눌러 통과한다. 이후 악성 서버에서 다운로드가 완료된 것을 확인하면 홈 화면에서 “downloads”를 입력하여 다운로드 목록으로 들어간다. 해당 리스트에서 가장 위에 있는 항목이 악성 앱 이므로 해당 앱을 눌러 설치한 후 실행하면 스마트 폰에 백도어가 설치된다. 이후에는 백도어를 통해 스마트 폰의 모든 기능에 접근 가능하다. 킷캣 이하의 버전에서는 구글 나우 런처는 없지만 대신 별도의 단축키를 제공하여 설정과 브라우저에 키보드만으로 접근할 수 있다. 이하 다른 공격과정은 동일하다. 공격자는 MITM공격을 수행하고 나서 의심 없이 정상적으로 키보드를 사용하는 희생자를 관찰하다가 희생자가 잠깐 한눈파는 사이 공격을 수행할 수 있다. 이러한 공격이 가능한 이유는 공격기기에서 HID 프로파일을 제공하면 스마트 폰이 스스로 접속한다는 점과 구글 나우 런처가 키보드만으로 악성 앱 설치를 위한 모든 기능을 제공하는 점 때문이다.

5. 대응 방법

Passkey Entry MITM Attack은 기기의 I/O 능력에 따라 인증 모드를 선택하는 성질로 인해 야기된다. 본 장에서는 우리가 제안하는 공격방법을 부분적으로 방어할 수 있는 대응 방법에 대하여 설명한다.

- **기기 상태 표기 통일** : 블루투스 페어링에 대하여 규격에 명시된 내용으로는 해당 공격을 막을 수 없다. 이에 대응할 수 있는 방법은 블루투스 기기의 페어링 종료 및 정상 연결 상태를 사용자에게 확실하게 인식시키는 일이다. 이를 위해 먼저 모든 제조사들이 준수 할 수 있는 상태 표기 방법이 표준화 되어야 한다. 각 기기마다 제각각인 LED의 개수와 위치, 색, 패턴은 사용자를 혼란스럽게 한다. 블루투스 기기에 익숙한 사용자라 할지라도 기기가 바뀔 때마다 매번 매뉴얼을 확인해야 하기 때문이다. 이는 공격 성공 확률을 상승 시키는 요인이 된다. 사실상 현재 Passkey Entry 모드를 사용하는 기기는 블루투스 키보드가 전부이므로 블루투스 키보드 제조사들 간의 표준화 노력이 필요하다.
- **충분한 출력장치 부착** : 위에서 언급한 표준화 노력과 더불어 사용자에게 키보드 상태를 인식시키기 위해 더욱 풍부한 자원이 필요하다. 심지어 고급 블루투스 키보드라 할지라도 크기와 미관을 위해 상태 LED를 하나만 부착하거나 측면 또는 후면에 부착하는 경우가 있다. 이런 경우 부저가 좋은 대안이 될 수 있다. 부저는 미관을 해치지 않으면서도 저렴한 비용으로 추가될 수 있다. 또한 LED는 꺼짐, 켜짐, 점멸 등 매우 한정적인 경우의 수만 가지고 있는데 반해 부저는 소리로서 매우 다양한 패턴을 출력할 수 있으며 사용자의 주의를 끌기도 좋다. 상태 LED만을 사용할 경우 키보드의 모든 상태를 상세하게 표현할 수 있도록 LED 수를 충분히 늘리고 시야에 잘 보일 수 있는 위치에 배치하여야 한다. 하지만 이 방식의 보안성은 결국 사용자의 판단에 의존한다. 이것만으로는 근본적인 해결책이 될 수 없고 앞서 제안한 표준화와 함께 이루어 져야 효과가 있다.
- Numeric Comparison 방식 사용 : NC 방식의 페어링을 수행하여 안전한 키 교환이 가능하다. 몇몇 고급 키보드들은 화려한 미관을 위해 키보드 자판에 백라이트를 장착하는 경우가 있다. 이렇게 풍부한 출력능력을 갖춘 경우 별도의 하드웨어 추가 없이 소프트웨어 변경만으로 NC 방식의 페어링을 수행할 수 있다. 처음과 끝을 알리는 문자와 함께 6자리 숫자의 백라이트를 순차적으로 점등하여 사용자에게 6자리 정수를 알린다. 사용자가 Yes/No를 결정하기 전까지 순차 점등을 반복한다. 사용자가 성공적으로 숫자를 읽을 수 있다면 MITM 공격에 대한 완전한 보호가 가능하다.

6. 논의 및 향후 연구

본 논문에서 우리는 블루투스 페어링 방법 중 SSP와 LE LP를 타겟으로 한 공격방법을 제안하고 이를 구현하였다. 현재 블루투스 페어링 방법 중 가장 최신의 방법인 LE SCP는 본 논문에서 구현하지 않았다. 그 이유로는 LE SCP 페어링 방법은 현재 최신 스마트 폰에만 적용되고 있다. 아직 블루투스 동글 형태로는 판매되고 있지 않기 때문에, 실험 환경을 구성하는데 어려움이 있었다. 하지만, LE LP와 SSP에 대한 실험을 통해 LE SCP에서도 우리의 공격이 동작할 것임을 유추할 수 있다. LE SCP의 페어링 요청 단계와 암호화 키 생성 단계는 LE LP 페어링과 동일하고 phase 2 내부의 사용자 확인 단계가 SSP와 같은 절차를 쓰기 때문이다. 단 4.2버전의 LE SCP에서 하나의 추가적인 제약사항이 존재한다. Table 4를 보면 DisplayOnly와 KeyboardDisplay가 조합되었을 경우 PE모드로 페어링을 수행하는 새로운 조합이 생기게 된다. 이러한 상황의 대표적인 예는 스마트 폰에 블루투스 키보드 에뮬레이터 앱을 깔고 PC와 페어링을 수행하는 것이다. 스마트 폰의 소프트웨어 키보드로 키보드 역할과 디스플레이 역할을 동시에 수행 가능하다. 이 경우 스마트 폰이 출력능력이 제한되지 않기 때문에 충분히 사용자에게 페어링 종료 및 정상 동작 상태를 알릴 수 있다. 이런 상황에서는 본 논문에서 제안한 공격이 사용자에게 발각될 것으로 생각된다.

다음으로 우리가 제안한 공격방법을 하위 레벨에서 더욱 효과적으로 구현하는 방법에 대하여 설명한다. 공격자는 Link Layer에 대하여 미리 MITM 공격을 완료해야 한다. 이후 페어링의 I/O capability exchange단계를 조작하여 원하는 인증 모드로 동작하도록 하고 이에 대한 MITM 공격을 수행하는 것이 가능하다. PE 모드의 사용자 확인을 먼저 연결된 키보드에서 키로깅 하여 통과하는 것은 앞서 설명한

방법과 같다. 둘 다 키보드 쪽 장치와의 페어링을 완전히 완료하고 프로파일 서비스 접속까지 수행해야 하므로 페어링 완료 시점만 다를 뿐 같은 원리이다. 단 이 방법은 공격자의 기기가 양쪽 기기의 페어링 사이에 끼어들어 진행한다. 사용자 입력 단계에서 스마트 폰 쪽은 대기 상태에 들어가고 키보드 쪽은 JW인증 모드로 계속 페어링이 진행된다. 즉 키보드의 페어링이 완료되어 키보드의 상태가 바뀌는 시점이 사용자 입력 직전이 된다. 이 방법의 장점은 보다 정밀한 속임수가 가능하다는 점이다. 또한 사용자 입력 장치로 위장하는 과정이 생략된다. 두 가지 방법 모두 규격에 나온 확인방법으로는 공격여부 확인이 불가하다. 단, 키보드의 상태 확인 LED를 부착하는 것은 제조사 재량에 달려있다. 페어링 대기 상태, 페어링 진행 상태, 페어링 종료, 서비스 정상 연결과 같은 기기 동작 상태를 확실하게 사용자에게 인지 시키도록 제작된 제품은 여전히 안전하다. 사용자는 페어링 수행 이전에 키보드가 서비스 정상 접속이 된 것을 인지하고 공격을 의심할 수 있다. 지금 소개하는 방법은 키보드가 서비스 정상 연결 상태로 변하는 시점이 숫자 키를 누르기 직전이기 때문에 실제 공격에서 사용자를 속일 수 있을 확률이 보다 높아질 것이다. 단점으로는 Link Layer상에서의 MITM 공격을 구현하기 위해서 비싼 하드웨어가 필요하고 소프트웨어 구현이 어렵다[11]. 반면 앞서 우리가 구현한 방법은 일반 블루투스 동글이 수행할 수 있는 정상기능들을 사용하여 구현하기 때문에 적은 비용으로 쉽게 구현할 수 있다. 하지만 사용자가 스마트 폰과 페어링을 수행할 키보드를 리스트에서 선택하기 이전에 키보드가 연결 완료상태가 된다. 따라서 LED의 패턴 변경 시점이 빨라지고 공격이 발각될 확률이 좀 더 높다. 본 논문의 Passkey Entry MITM Attack은 키보드와의 페어링에서만 IO Capability를 NoInputNoOutput으로 조작하여 JW 인증 모드를 수행한다. 스마트 폰과의 페어링에서는 PE 인증 모드를 수행한다. 이를 통해 사용자 확인 절차를 정상 상황과 똑같이 유지시키는 것이 공격의 핵심이므로 어떠한 방법으로 구현해도 공격을 검증하는데 아무 문제가 없다.

7. 결 론

본 논문에서는 블루투스 페어링 방식 중 SSP, LE LP의 PE 인증 모드에서 동작 가능한 MITM 공격 방법을 제안하였다. 해당 공격 방법은 매번 생성되는 난수를 Passkey로 사용하는 PE모드에 대한 최초의 공격 방법이며, 현재 판매되고 있는 모든 블루투스 키보드에 적용가능하다. 또한 제안하는 공격방법을 구현하고 이를 통하여 실제 사용자의 블루투스 키보드와 스마트폰 사이의 MITM 공격이 간단하게 수행될 수 있음을 보였다. 그리고 제안하는 공격방법을 부분적으로 방어할 수 있는 대응방법을 제안하였다.

마지막으로, 본 논문을 통하여 우리는 다음과 같은 결론을 내린다. PE 모드는 인증 알고리즘 중 가장 불편한 사용자 확인 과정을 가지고 있다. 그럼에도 불구하고 안전성은 JW모드와 동일하다. 편의성을 위해서라면 JW를 사용하는 편이 좋고 안전성을 위해서라면 OOB나 NC을 사용해야한다. 심지어 편의성 또한 OOB와 NC가 더 좋다. 이러한 사실이 규격에서 언급되어야 하며 블루투스 기기 제조사들은 이에 대한 대책을 세워야한다.

References

[1] Wikipedia, Bluetooth [Internet], https://en.wikipedia.org/wiki/Bluetooth.

[2] Tae-ho Kim, iot-wi-fi-bluetooth-z-wave [Internet], http://goodnirvana.blogspot.kr/2015/09/iot-wi-fi-bluetooth-z-wave.html.

[3] Nextdaily, Bluetooth 4.0 [Internet], http://www.nextdaily.co.kr/news/article.html?id=20100616800003.

[4] Bluetooth SIG Proprietary, BLUETOOTH SPECIFICATION Version 4.2, Vol.1, Part A, p.85, 2014.

[5] Bluetooth SIG Proprietary, BLUETOOTH SPECIFICATION Version 4.2, Vol.1, Part A, pp.88-89, 2014.

[6] Haataja, Keijo, and Pekka Toivanen, “Two practical man-in-the-middle attacks on bluetooth secure simple pairing and countermeasures,” *IEEE Transactions on Wireless Communications*, Vol.9, Iss.1, pp.384-392, 2010.

[7] Haataja Keijo, and Pekka Toivanen, “Practical man-in-the-middle attacks against bluetooth secure simple pairing,” *2008 4th International Conference on Wireless Communications, Networking and Mobile Computing*, IEEE, 2008.

[8] Dong-Hoon Lee, “A First Course in Modern Cryptography,” seoul: Irun Inc., pp.450-452, 2012.

[9] John Hering, blue snifer rifle [Internet], http://www.smallnetbuilder.com/wireless/wireless-howto/24256-howtobluesniperpt1.

[10] Bluetooth Special Interest Group (SIG), Official Linux Bluetooth protocol stack [Internet], http://www.bluez.org/.

[11] Barnickel, Johannes, Jian Wang, and Ulrike Meyer. “Implementing an attack on bluetooth 2.1+ secure simple pairing in passkey entry mode,” *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*, IEEE, 2012.

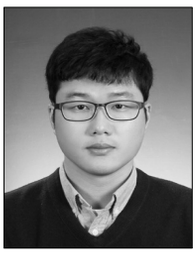
이 재 령

e-mail : arche2013@gmail.com

2015년 연세대학교 컴퓨터공학과(학사)

2015년~현 재 고려대학교 정보보호대학원 금융보안학과 석사과정

관심분야 : Information Security & embedded system



최 원 석

e-mail : beb0396@naver.com
2008년 서울시립대학교 수학과(학사)
2013년 고려대학교 정보보호대학원(석사)
2013년～현 재 고려대학교 정보보호대학원 정보보호학과 박사과정
관심분야 : Information Security & Medical Device Security

이 동 훈

e-mail : donghlee@korea.ac.kr
1983년 고려대학교 경제학과(학사)
1987년 Oklahoma University 전산학과 (석사)
1992년 Oklahoma University 전산학과 (박사)
1993년～1997년 고려대학교 전산학과(조교수)
1997년～2001년 고려대학교 전산학과(부교수)
2001년～현 재 고려대학교 정보보호대학원 교수
관심분야 : Cryptographic protocols & Cryptographic theory & USN theory & Key Exchange & Anonymity research & PET technology