Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.28, NO.6, Dec. 2018 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2018.28.6.1355

위협 모델링을 통한 스마트밴드 보안 요구사항 분석*

강 수 인,† 김 혜 민, 김 휘 강‡
고려대학교 정보보호대학원

Trustworthy Smart Band: Security Requirements Analysis with Threat Modeling*

Suin Kang,† Hye Min Kim, Huy Kang Kim‡
Graduate School of Information Security, Korea University

요 약

스마트밴드가 삶을 보다 편리하게 만들고 긍정적인 생활 습관을 들일 수 있도록 도와주게 되면서 많은 사람들이 현재 스마트밴드를 사용하고 있다. 또한, 스마트밴드는 사용자의 개인 정보를 다루기 때문에 스마트밴드 시스템에 대한 안전한 보안 설계 및 구현이 필요하게 되었다. 신뢰할 수 있는 스마트밴드를 만들기 위해서는 먼저 시스템의 보안 요구사항을 도출한 다음 이를 충족하도록 시스템을 설계해야 한다. 본 논문에서는 Data Flow Diagram, STRIDE, Attack Tree와 같은 위협 모델링 기술을 스마트밴드 시스템에 적용하여 위협을 식별하고 이에 따라 보안 요구사항을 도출하였다. 위협 모델링을 통해 식별한 위협을 이용하여 실제로 스마트폰을 스마트밴드에 연결하는 과정에서의 취약점을 발견하였고 이를 통해 스마트밴드의 제어 권한을 획득하는 공격을 수행할 수 있었다. 발견된 취약점과 위협에 대응하기 위해 보안 대책을 제안하였으며 보안 프로토콜 자동 검증 도구인 Scyther를 사용하여 제안한 보안 대책에 대한 안전성을 검증한다.

ABSTRACT

As smart bands make life more convenient and provide a positive lifestyle, many people are now using them. Since smart bands deal with private information, security design and implementation for smart band system become necessary. To make a trustworthy smart band, we must derive the security requirements of the system first, and then design the system satisfying the security requirements. In this paper, we apply threat modeling techniques such as Data Flow Diagram, STRIDE, and Attack Tree to the smart band system to identify threats and derive security requirements accordingly. Through threat modeling, we found the vulnerabilities of the smart band system and successfully exploited smart bands with them. To defend against these threats, we propose security measures and verify that they are secure by using Scyther which is a tool for automatic verification of security protocol.

Keywords: Security requirement, Smart band, Threat Modeling

I. 서 론

스마트밴드는 개인의 건강 상태를 유지 또는 향상시키기 위하여 자동으로 데이터를 측정하여 사용자에게 상태 정보를 제시하고 긍정적인 생활 습관을 들일 수 있도록 도와준다. 스마트밴드는 기본적으로 만

Received(09. 21. 2018), Modified(10. 23. 2018), Accepted(10. 23. 2018)
* 이 논문은 2018년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (과제번호 No.2018-0-00232, 클라우드 기반 IoT 위협 자율 분석 및 대응 기술 개발)
† 주저자, sikang@korea.ac.kr
‡ 교신저자, cenda@korea.ac.kr(Corresponding author)

보기 기능을 통해 사용자의 발걸음 수를 측정하거나 사용자의 심장박동 수, 수면 시간 등을 측정할 수 있다. 또한, 스마트폰에 연결되어 메시지 도착 알림, 현재 사용자 상태 모니터링, 타이머 알람 설정 등 다양한 기능을 수행한다. 이러한 기능들은 일상생활에 긍정적인 영향을 가져다주지만, 공격자가 공격에 성공하면 사용자의 이동 경로, 생활패턴 등의 개인 정보가 유출될 수 있기 때문에 스마트밴드 시스템은 높은 수준의 신뢰성(Trustworthiness)이 필요하다.

많은 연구에서 스마트밴드의 공격 방법과 취약점에 대해서 분석했다. Zhou 외 1인은 상업용 피트니스 트래커에 대해 위협 분석을 수행하고 로그인 정보 및 데이터가 암호화 과정 없이 일반 텍스트로 전송한다는 사실을 발견하였다[1]. Lee 외 4인은 웨어러블 서비스의 취약점을 발견하고 공격 시나리오를 생성하는 데 성공했다[2]. Goyal 외 2인은 웨어러블 피트니스 트래커의 사용자 개인 정보를 획득하고 DoS (Denial of Service) 공격을 성공적으로 수행하였다[3]. Seneviratne 외 6인은 웨어러블 장치에서 발생할 수 있는 보안 위협에 대해서 기밀성 (Confidentiality), 무결성 (Integrity), 가용성 (Availability) 3가지 카테고리에 따라 분석하였다[4]. 이러한 연구 결과에 따르면 많은 장치가 취약점을 가진 채로 출시된다는 것을 알 수 있다. 만약 스마트밴드가 발견되지 않은 취약점을 가진 채 출시가 된다면 사용자의 프라이버시와 중요한 개인 정보를 보호할 수 없다. 따라서 신뢰할 수 있는 스마트밴드 시스템을 구축하기 위해서는 알려진 취약점뿐만 아니라 알려지지 않은 취약점에도 대응할 수 있어야 하며 공격이 가능한 지점을 명확하게 식별할 수 있어야 한다.

신뢰할 수 있는 시스템은 Availability, Reliability, Security, Safety를 모두 고려하여 어떠한 상황에서도 동작하여 안전하게 목적을 달성할 수 있는 시스템을 뜻한다[5-6]. 신뢰할 수 있는 시스템을 개발 및 운영하기 위한 일련의 과정을 정보 보증(Information assurance)이라 하며 정보 보증을 달성하기 위해서 보안 요구사항 분석 및 설계 단계에서부터 시스템을 구현하고 운용하는 단계까지 단계별 보증을 달성해야 한다. 전체 시스템을 안전하게 설계하고 구현하기 위해서는 보안 요구사항을 정확하게 도출하고 이를 만족시키도록 설계해야 하므로 정확한 보안 요구사항 도출은 정보 보증을 달성하기 위해 가장 중요한 부분이다. 본 논문에서는 Data Flow Diagram [7-8], STRIDE[9-10], Attack Tree[11]와 같은 위협 모델링 기법을 스마트밴드 시스템에 적용하여 보안 위협을 식별하고 이에 따른 보안 요구사항을 도출한다. 이 프로세스를 통해 알려진 취약점과 더불어 알려지지 않은 취약점에도 대응할 수 있으며 공격 가능 지점을 미리 식별할 수 있다. 예시로 우리는 위협 모델링을 통해서 스마트밴드의 연결 설정 프로세스에서의 취약점을 발견하였고 해당 취약점을 이용하여 성공적으로 시스템 권한을 획득할 수 있었다. 연결 프로세스에서의 문제와 다른 취약점들에 대응하기 위해 필요한 보안 요구사항을 반영하여 안전한 연결 및 통신 프로토콜을 보안 대책으로 제안한다. 마지막으로 자동 검증 도구를 사용하여 제안한 보안 대책의 안전성을 객관적으로 검증하였다. 본 논문의 제안된 접근 방법에 대한 개요는 Fig. 1.에서 확인할 수 있다. 본 논문이 기여하는 바는 다음과 같다.

- 스마트밴드 시스템의 취약점과 보안 요구사항을 도출하였다. 이는 서비스 제공 업체가 정보 보증을 달성하는 신뢰할 수 있는 스마트밴드 시스템을 구축하는 데 유용하게 사용할 수 있다.
- 실제 공격 실험을 통해 위협 모델링 분석 방법이 스마트밴드 시스템의 위협과 공격 시나리오를 도출하는 데 효과적이라는 것을 보였다. 실제로 스마트밴드의 연결 프로세스에서 충분한 사용자 인증과정이 없다는 취약점을 발견하였으며 해당 취약점을 이용하여 공격자는 정상 사용자로 가장할 수 있었으며 스마트밴드의 통제권을 획득하여 정상적으로 사용할 수 있었다.
- 연결 프로세스에서의 문제점에 대응할 수 있고 스마트밴드 시스템의 보안 요구사항을 고려한 보안 대책을 제안하였다. 스마트밴드의 하드웨어 특성을 고려하여 보안 대책들은 간단하고 시스템에 적용하기 쉽도록 설계하였다. 또한, 보안 대책의 안전성을 검증하기 위해 자동 검증 도구인

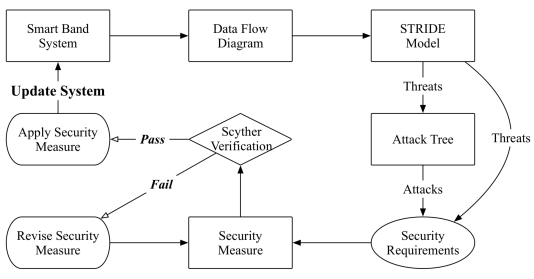


Fig. 1. Overview of the proposed approach

Scyther[12]를 사용하여 객관적으로 안전성을 검증하였다.

II. 관련 연구

2.1 스마트밴드 보안

Zhou 외 1인은 피트니스 트래커 중 하나인 Fitbit에서 발생할 수 있는 위협에 대해 분석하였다[1]. Fitbit은 웹 서버와 통신할 때 암호화 없이 일반 텍스트로 로그인 정보와 HTTP 데이터를 전송한다. 이를 통해 공격자가 인증 절차 없이 데이터에 쉽게 접근할 수 있다. 이 취약점을 해결하기 위해 Rahman 외 2인은 FitLock을 제안하였다[13]. 그러나 FitLock이 시스템에 적용되더라도 여전히 위험한 공격이 가능하였다. 따라서 온라인 네트워크에 연결된 피트니스 트래커의 경우 취약점에 대해 자세히 분석해야 하며 이에 대응할 수 있는 보안 대책을 설계해야 한다. Lee 외 4인은 웨어러블 서비스의 취약점을 찾기 위해 디바이스, 게이트웨이, 서버의 관점에서 시스템을 분석하였다[2]. 그들은 허가되지 않은 장치와의 연결 설정, 위조된 게이트웨이를 통한 정보 유출, 악의적인 코드 주입과 같이 3가지로 공격 시나리오를 설정하고 이를 통해 성공적으로 스마트밴드를 공격하였다. Goyal 외 2인은 보안과 프라이버시 이슈에 초점을 맞추어 웨어러블 트래커를 분석하였다[3]. 그들은 웨어러블 트래커가 취약할 수 있는 공격 유형에 대하여 보안 분석을 수행하고 보안 평가 테이블을 공식화하였다. 그들은 사용자의 개인 정보에 접근하기 위해 Gatt-Tool를 사용했으며, DoS 공격을 수행하였다. 또한, 그들은 모바일 어플리케이션의 소스 코드를 수정하여 사용자 데이터를 획득하였고 서버 인증서 유효성 검사에서 HTTP와 SSL 취약성을 발견하였다. Fereidooni 외 4인은 17개의 웨어러블 디바이스를 분석하고 쉽게 악용될 수 있는 취약점들을 밝혀냈다[14]. 그들은 인증된 사용자로 위장하여 사용자의 개인 정보를 얻거나 디바이스의 정보를 수정할 수 있음을 입증하며 피트니스 트래커의 통신 프로토콜에는 End-to-End 암호화가 구현되어야 하고 전자 서명이 추가되어야 한다고 주장하였다.

이러한 연구들에 따르면, 많은 스마트밴드들이 취약점을 인지하지 못한 채로 출시가 되고 있다는 것을 알 수 있다. 알려지지 않은 취약점이 있는 스마트밴드가 상용화되면, 장치를 구매한 사용자들은 위협으로부터 보호받을 수 없다. 그러므로 위협 모델링을 통해 위협 분석을 진행하여 알려진 취약점과 더불어 알려지지 않은 취약점에 대해서도 미리 발견해야 한다. 스마트밴드 제공 업체는 스마트밴드 시스템을 보다 안전하고 신뢰할 수 있도록 만들기 위해 발견된 위협에 대한 보안 대책을 시스템에 적용해야 한다.

2.2 Data Flow Diagram (DFD)

공격이 가능한 부분을 명확히 식별하기 위해서 우리는 전체 스마트밴드 시스템에 대하여 이해해야 한다. Data Flow Diagram (DFD) 를 사용하면 시스템 구조를 추상화하여 시스템의 데이터 흐름을 한눈에 볼 수 있다. DFD는 시스템 내 구성 요소들의 데이터 흐름을 보여주기 때문에 위협 모델링에 이상적이다[7]. 추상화 과정을 통해 분석에 필요하지 않은 부분은 제외하고 시스템 전체 구조와 데이터 흐름을 쉽게 이해할 수 있다[8]. 일반적으로 공격은 데이터가 신뢰할 수 있는 영역을 벗어나 송수신될 때 발생하기 때문에 데이터가 신뢰할 수 있는 영역을 벗어나는 시점에 무슨 일이 발생하는지 명확하게 파악할 필요가 있다. 그러므로 보안 전문가는 취약점을 미리 식별하기 위해 데이터 흐름을 명확하게 보여주는 DFD를 작성해야 한다. 또한, DFD는 시스템 내 요소 단위로 작성되므로 각 요소의 취약점을 쉽게 발견할 수 있다. DFD의 구성 요소는 다음과 같다.

- **External entity (Rectangle).** External entity는 내부 요소와 상호 작용하는 시스템 외부의 요소를 뜻한다.
- **Process (Circle).** Process는 응용 프로그램 내에서 데이터 처리 기능을 수행하는 작업을 말한다. 입력 데이터를 받아 기능을 수행하고 결과값을 출력한다.
- **Data flow (Directed arrow).** Data flow는 두 요소 사이 데이터 전송을 나타낸다. 데이터 흐름의 방향은 화살표로 표시한다.
- **Datastore (Two parallel lines).** Datastore는 시스템 내 데이터를 저장하는 데 사용된다. 이는 데이터를 저장하고 전송하지만, 데이터와 관련된 기능을 수행할 수 없다.
- **Trust boundary (Red dashed lines).** Trust boundary는 신뢰 수준이 변하는 경계선을 말한다. 이를 통해 권한이 변경되는 위치와 신뢰할 수

없는 출처에서 오는 데이터가 송신되는 위치를 쉽게 식별할 수 있다.

2.3 STRIDE

STRIDE는 위협들을 카테고리로 분류하기 위해 Microsoft에서 개발한 분류 체계이다 [9], [10]. STRIDE는 Spoofing, Tampering, Repudiation, Infor - mation disclosure, Denial of service, Elevation of privilege의 6가지 카테고리들의 머리글자에서 파생되었다. STRIDE에 대한 자세한 설명은 다음과 같다.

- **Spoofing.** 공격자는 정상 사용자로 가장하여 시스템 내 목표물에 접근한다.
- **Tampering.** 공격자는 정상 사용자를 속이기 위해 데이터를 수정한다. 공격 대상은 주로 데이터 저장소에 있는 데이터와 요소들 사이에 송수신되는 데이터들이다.
- **Repudiation.** 공격자는 특정 행위가 발생한 일을 부인한다. 이는 악의적인 행위를 수행한 사람을 식별하기 어렵게 만든다.
- **Information disclosure.** 특정 정보가 권한이 없는 공격자에게 노출된다. 이에 대응하기 위해서는 권한을 임의로 변경할 수 없도록 하며 데이터가 노출될 수 있는 지점을 주의해야 한다.
- **Denial of service.** 공격자는 정상 사용자가 서비스를 정상적으로 이용할 수 없도록 방해한다. 이에 대응하기 위해서 시스템 가용성과 신뢰성이 향상되어야 한다.
- **Elevation of privilege.** 공격자는 적절한 권한 부여 프로세스 없이 높은 권한을 획득한다. 공격을 성공적으로 수행하면, 공격자는 시스템에 접근할 수 있는 권한을 얻는다.

STRIDE는 분석 시스템에 대한 제한이 없으며 표준화된 분석 프로세스로 구성되므로 다양한 분야에 적용할 수 있다. Dev 외 1인은 STRIDE를 사용하여 Google Chrome API의 취약점을 분석하고 다양한 위협의 가능성을 제시하였다[15]. Karahasanovic 외 2인은 AUTOSAR 자동차 표준에 따라 STRIDE와 TARA 기술을 이용하여 IoT 기술이 적용된 자동차의 취약점에 대하여 분석하였다[16]. Cagnazzo 외 3인은 STRIDE 위협 모델을 사용하여 IoT 디바이스와 연결된 모바일 헬스 시스템의 위협을 분석하였다[17]. 그들은 BAN, WLAN과 같은 네트워크에 특히 초점을 맞추었다. DFD로 모바일 헬스 시스템을 추상화하고 위협 모델링을 수행한 후 시스템에 암호화 기술과 인증 기술을 적용하는 완화 전략을 제안하였다. 강태운 외 1인은 STRIDE를 사용하여 가상현실 기기와 게임 시스템의 정보보증을 위한 위협 분석을 수행하였다[18]. 김혜민 외 1인은 STRIDE를 사용한 위협 모델링을 통해 PS4와 PC 간의 Remote Play 환경에서 발생할 수 있는 위협을 도출하였다[19].

STRIDE는 시스템이 도달해야 할 보안 목표 6가지(인증, 무결성, 부인봉쇄, 기밀성, 가용성, 접근제어)를 공격자의 입장에서 고려하여 개발하였기 때문에 위협 식별에 용이하다. 또한 DFD를 함께 사용하면 시스템의 데이터 흐름을 파악할 수 있고 신뢰 수준이 변하는 지점을 알 수 있기 때문에 대상 시스템의 공격 가능 지점과 취약점을 쉽게 식별할 수 있다. 그러므로 본 논문에서는 스마트밴드 시스템의 DFD를 작성하여 시스템 구조를 파악하고 STRIDE를 적용하여 스마트밴드의 취약점과 위협을 식별한다.

2.4 Scyther

Scyther는 인터넷 및 개방형 네트워크에서 사용되는 프로토콜의 보안성을 검증해주는 오픈소스 자동 검증 도구이다. 추상화 기술을 사용하지 않고 분석할 수 있으며 검증 속도가 다른 분석 도구에 비해 빠르다. 이러한 장점들로 인해 많은 연구 논문에서 Scyther를 사용하여 프로토콜의 안전성을 확인하였다[12]. Basin 외 2인은 Scyther를 사용하여 ISO/IEC 9798 표준 프로토콜 내 취약점을 분석하고 해당 취약점에 대응할 수 있는 보안 대책을 제안하였다[20]. 그 후, ISO는 표준 프로토콜을 업데이트하고 배포하였다. Cremers는 인터넷 키 교환 프로토콜을 공식화하고 Scyther를 사용하여 프로토콜의 알려지지 않은 취약점을 발견할 수 있도록 단계별로 분석을 수행하였다[21].

III. 스마트밴드 위협 모델링

이 절에서는 스마트밴드 시스템의 취약점과 위협에 대해 분석한다. 위협 분석은 메시지 알람, 타이머 알람, 데이터 송수신 등을 포함하는 스마트밴드의 기본적인 서비스에 대해서 수행하였으며 자세한 분석 절차는 다음과 같다.

1) 시스템의 경계와 범위를 정의한 후, 중요한 자산을 식별한다. 이때, DFD를 작성하여 스마트밴드 시스템의 주요 자산을 명확하게 식별하고 데이터의 흐름과 구조를 파악한다.
2) 작성한 DFD를 기반으로 하여 STRIDE를 시스템에 적용하고 발생 가능한 위협들을 도출한다.
3) STRIDE로부터 도출한 위협들이 스마트밴드 시스템을 공격하는데 어떻게 사용될 수 있는지 파악하기 위하여 Attack Tree를 설계하고 가능한 공격 시나리오를 도출한다.

3.1 스마트밴드 시스템 DFD 작성

스마트밴드 시스템의 위협을 분석하고 데이터 흐름을 확인하기 위하여 DFD를 작성한다. 공격자는 주로 목표 데이터가 신뢰 영역을 벗어날 때 공격을 수행하기 때문에 DFD를 정확하게 작성하면 보안 위협을 식별하기 수월하다. Fig. 2.에서 스마트밴드 시스템의 DFD를 확인할 수 있으며 스마트폰, 스마트밴드, 웹 서버가 사용자 정보를 주고받는다. 스마트밴드 시스템은 3가지 부분으로 나눌 수 있으며 각 부분에 대한 설명은 다음과 같다.

- Fig. 2.의 상단 부분은 웹 서버와 사용자의 스마트폰 사이의 데이터 교환 프로세스를 보여준다. 스마트폰(E1)에 입력된 ID와 Password는 웹 서버로 보내지고 사용자 인증 절차(P1)을 거친 후 웹 서버 데이터베이스(D2)로 전송된다. 웹 서버는 해당하는 ID의 사용자 정보와 어플리케이션 데이터를 스마트밴드 어플리케이션 저장소(D1)에 전송한다(P2). 어플리케이션은 스마트밴드에서 측정한 걸음 수 및 심박 수와 같은 데이터를 웹 서버 데이터베이스로 전달한다(P2).
- Fig. 2.의 중앙 부분은 스마트폰과 스마트밴드(E3) 사이 연결 설정 및 데이터 교환 프로세스를 보여준다. 스마트폰이 메시지를 받았을 때, 메시지 종류를 확인하고 어플리케이션에 알림 정보를 전달한다(P3). 기기 간 연결을 위해 스마트밴드 어플리케이션은 스마트밴드에 연결을 요청하며 스마트폰 정보를 전달한다(P4). 스마트밴드는 이에 대한 응답과 함께 스마트밴드 정보를 전달하여 연결을 설정한다. 연결이 완료되면 스마트밴드

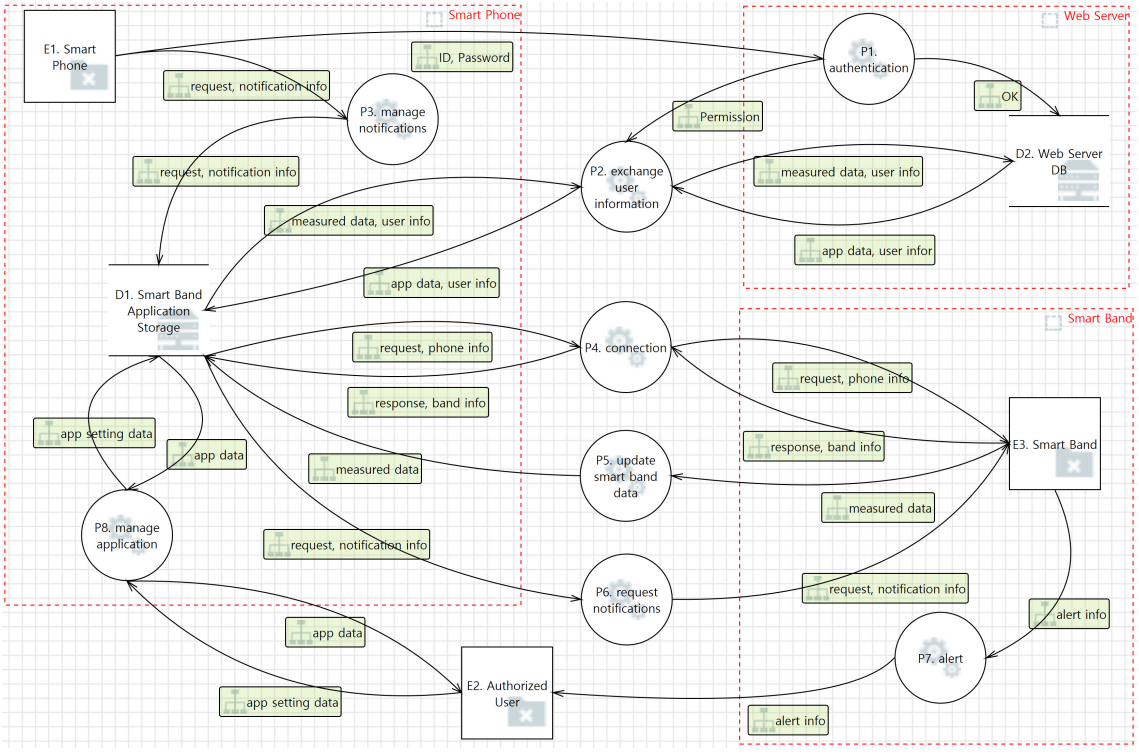


Fig. 2. Data Flow Diagram of smart band system

는 동기화 프로세스를 수행하고 어플리케이션에 측정된 데이터를 업데이트한다(P5). 이후 어플리케이션은 알림 요청과 함께 알림 정보를 전달한다(P6).

- Fig. 2.의 하단 부분은 인가받은 사용자(E2)와 어플리케이션 사이의 데이터 교환 및 알람 프로세스를 보여준다. 스마트밴드는 진동과 함께 화면에 아이콘이나 숫자를 출력하여 사용자에게 알림 정보를 전달한다(P7). 인가받은 사용자는 타이머 알람과 같은 어플리케이션 설정 정보를 입력하거나 어플리케이션을 통해 스마트밴드가 측정한 정보를 얻는다(P8).

스마트밴드 시스템의 DFD를 살펴보면 스마트 밴드의 주요 기능은 사용자의 개인 정보를 활용하여 편의성을 높이는 것임을 알 수 있다. 따라서 스마트밴드 시스템의 주요 자산을 사용자의 개인 정보와 스마트밴드 서비스 그 자체로 정의한다. 또한, 공격자의 목표는 시스템의 주요 자산을 획득하는 것이기 때문에 공격자의 최종 목표를 사용자의 정보 획득과 사용자가 정상적으로 스마트밴드의 기능을 사용할 수 없도록 서비스 거부 공격을 수행하는 것으로 정의한다.

3.2 STRIDE 위협 분석

3.1에서 작성한 DFD를 바탕으로 External entity, Process, Data flow, Datastore와 같은 DFD 요소에 존재하는 보안 위협을 식별하기 위해 STRIDE를 적용한다. DFD 각 요소에 적용할 수 있는 STRIDE는 Table 1.에서 확인할 수 있다[10]. 예를 들어, External entity는 시스템 외부의 요인이기 때문에 정상 사용자로 가장하거나 사용자가 특정 행위를 시도한 것을 부인하여 스마트밴드 시스템에 위협을 가할 수 있지만, 사용자를 조작하거나 높은 권한을 획득하게 하는 방식으로 시스템에 위협을 가할 수 없으므로 STRIDE 중 Spoofing과 Repudiation만 적용할 수 있다. STRIDE 위협

Table 1. STRIDE threats per DFD element

Element	S	T	R	I	D	E
External entity	×		×			
Process	×	×	×	×	×	×
Data flow		×		×	×	
Datastore		×		×	×	

Table 2. symbol settings

Elements	Name	Symbol
External entity	Smartphone	E1
	Authorized user	E2
	Smart band	E3
Process	Authentication	P1
	Exchange user information	P2
	Manage notifications	P3
	Connection	P4
	Update smart band data	P5
	Request notifications	P6
	Alert	P7
	Manage application	P8
Datastore	Smart band application storage	D1
	Web server database	D2

분석에 앞서 스마트밴드 DFD 내 각 요소의 동작 원리에 따라서 Table 2.와 같이 기호로 정의한다[7]. 스마트밴드 DFD에는 3개의 External entity, 8개의 Process, 2개의 Data - store, 그리고 23개의 Data flow가 있다. 따라서 STRIDE 결과로 나올 수 있는 전체 위협의 개수는 총 129개이다. 보다 객관적으로 STRIDE 모델을 스마트밴드 시스템에 적용하기 위해 Microsoft Threat Modeling tool[22-23]을 사용하였다. Microsoft Threat Modeling tool에서 생성한 위협 모델링 보고서를 참고하여 보안 위협을 도출하였다. 스마트밴드의 취약점에 관한 사전 연구[1-4], [13], [14], [17], [24]를 고려하여 공격자의 목표를 달성하기 위해 적용될 수 있는 위협들을 선별하고 이를 Table 3.에 나열하였다.

3.3 Attack Tree 작성

Attack Tree는 시스템에 가능한 공격을 계층적으로 정리하는 도구로 시스템의 공격 시나리오를 찾는데 유용하다[11]. 우리는 STRIDE 모델을 통해 얻은 위협들과 공격자의 목표 사이의 연관성을 분석하여 Attack Tree를 설계하였다. Attack Tree를

Table 3. STRIDE threat analysis of smart band system

Element	Type	Threat Description
E1	S	T1. Attacker spoofs smartphone to get user information.
	R	T2. Attacker sends data and denies this later.
E2	S	T3. Attacker spoofs an authorized user to manipulate application data.
	R	T4. Attacker sets application data and denies this later.
P1	S	T5. Attacker spoofs the smartphone or web server to obtain ID/ PW of user or access permission for information exchange.
	T	T6. Attacker modifies ID/ PW of user to disturb authentication.
	I	T7. Attacker obtains ID/ PW.
	D	T8. Attacker makes it impossible to perform the normal authentication through excessive authentication requests.
	E	T9. Attacker gives the higher privilege to someone who does not have it.
P2	S	T10. Attacker spoofs the authorized user's smartphone to transmit wrong data or gain access to the transmitted data.
	T	T11. Attacker modifies application data or user information and transmits it.
	I	T12. Attacker obtains user information or application data.
	D	T13. Attacker makes it impossible to exchange data through excessive data transmission.
	E	T14. Attacker gives the privilege to someone who could not access to application information and user information.
P4	S	T15. Attacker pretends to be an authorized user to control smart band.
	T	T16. Attacker modifies the smartphone/ smart band information and transmits them.
	I	T17. Attacker obtains smartphone/ smart band information and connection information.
	D	T18. Attacker makes it impossible to connect through an excessive connection request.
	E	T19. Attacker gives the privilege to someone who could not access to connection information.
P6	S	T20. Attacker spoofs the authorized user's smartphone to transmit wrong notification information.
	T	T21. Attacker modifies notification information and transmits it.
	I	T22. Attacker obtains notification information.
	D	T23. Attacker makes it impossible to transmit the normal notification request through excessive notification requests.
	E	T24. Attacker gives the privilege to someone who could not access notification request.
D1	T	T25. Attacker modifies the data in the storage and inserts wrong data.
	I	T26. Attacker accesses the datastore and obtains application information and user information in the storage.
	D	T27. Attacker inserts data too much so that the storage does not work properly.
D2	T	T28. Attacker modifies the data in the web server database and inserts wrong data.
	I	T29. Attacker accesses the database and obtains application information and user information in the web server database.
	D	T30. Attacker transmits data excessively to disturb normal data exchange.
P1 → D2	T	T31. Attacker modifies authentication result so that the authorized user could not access web server database, or an unauthorized user could access web server database.
	I	T32. Attacker obtains authentication result.
	D	T33. Attacker transmits authentication data excessively to disturb the normal authentication.

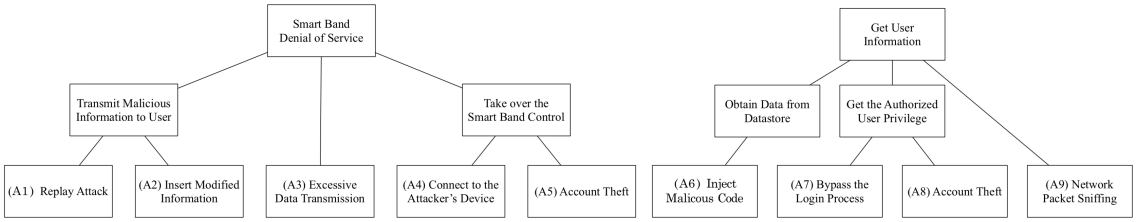


Fig. 3. Attack tree for smart band system

설계할 때, 부채널 공격과 사회 공학 공격 그리고 스마트밴드 시스템의 경계 외부는 고려하지 않는다. Attack Tree의 설계에 사용된 공격들은 스마트밴드와 웨어러블 디바이스에 관한 사전 연구 [1-4], [13], [14], [17], [24]와 STRIDE 위협 분석의 결과를 참고하여 도출하였다. Attack Tree의 최상위 노드들은 공격자의 최종 목표를 말하며 하위 노드는 하위 공격 목표들을 말한다. 공격자는 하위 목표들을 달성하여 최종 공격 목표를 달성할 수 있다. 또한, Attack Tree를 따라 내려가면 하나의 공격 시나리오를 얻을 수 있다. 사전에 정의한 공격자의 최종 목표인 사용자 정보 획득과 사용자 서비스 사용 거부를 각각 최상위 노드로 하여 두 개의 Attack Tree를 생성하였으며, 각 Attack Tree는 Fig. 3.에서 확인할 수 있다. Attack Tree에 대한 자세한 설명은 다음과 같다.

1) Smart band denial of service

스마트밴드의 서비스 거부 공격은 사용자에게 악의적인 정보 전달, 과도한 데이터 전송, 스마트밴드 제어권 획득 중 하나를 달성하여 수행할 수 있다.

- 악의적인 정보를 전달하기 위해서 공격자는 사전에 전송된 정보를 저장하고 나중에 재전송하는 방법(A1)이나 조작된 데이터를 함께 전송하는 방법(A2)을 사용할 수 있다. 이러한 공격은 정보에 접근하여 수정할 수 있는 권한을 가진 사용자를 가장하거나 공격자의 권한을 더 높게 변경하여 수행할 수 있다. 재전송 공격에 사용할 패킷은 정상적인 데이터 전송이 일어날 때, 패킷 스니핑을 통해서 획득할 수 있다.
- 과도한 데이터 전송 공격(A3)은 데이터를 과도하게 전송하여 스마트밴드가 계속 진동하게 하거나 계속 알람이 울리도록 하여 스마트밴드를 정상적으로 사용할 수 없도록 한다. 또한, 웹 서버에 과도한 데이터 전송 및 인증 요청 등을 시도하여 공격을 수행할 수 있다. 이 공격들을 효과적으로 수행하기 위하여 공격자는 사용자 인증 프로세스 또는 사용자 정보 교환 프로세스를 대상으로 할 수 있다.
- 스마트밴드의 제어권을 획득하여 스마트밴드를 사용할 수 없도록 하는 공격은 스마트밴드를 공격자의 디바이스에 연결하는 방법(A4)과 정상 사용자의 계정을 도용하는 방법을 통해 수행할 수 있다. 공격자는 연결 프로세스 동안에 정상적인 사용자인 것처럼 가장하여 스마트밴드에 연결을 시도하거나 계정 도용 공격(A5)을 위해서 인증 프로세스에서 사용자 정보를 탈취할 수 있다.

2) Get user information

사용자 정보를 얻는 공격은 데이터 저장소에서 데이터 획득, 정상 사용자 권한 획득, 네트워크 트래픽 스니핑 중 하나를 달성하여 수행할 수 있다.

- 데이터 저장소로부터 데이터를 얻기 위해 공격자는 파일 복구 툴을 사용하거나 악성 코드를 주입하여 사용자의 스마트폰이나 스마트밴드에서 데이터를 가져올 수 있다(A6). 이러한 공격들은 공격자가 데이터 저장소에 접근할 수 있는 경우 발생할 수 있으며 이 외에 정상 사용자로 가장하거나 공격자의 권한을 변경하여 수행할 수 있다.
- 공격자는 정상 사용자 권한을 획득하여 사용자 정보를 얻을 수 있다. 이는 로그인 절차를 우회(A7)하거나 정상 사용자의 계정을 도용(A8)하여

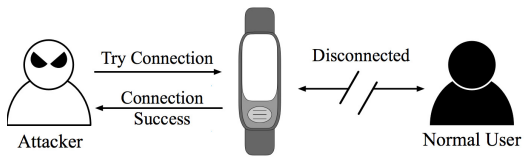


Fig. 4. Attack scenarios using vulnerabilities on connection process

수행 가능하며, 이러한 공격들을 위해 공격자는 인증 프로세스나 사용자 정보 교환 프로세스를 공격 대상으로 설정할 수 있다.

- 공격자는 데이터가 전송될 때, 네트워크 트래픽 스니핑(A9)을 통해 정보를 획득할 수 있다[24]. 해당 공격은 스마트폰과 웹 서버 사이의 데이터 송수신 과정이나 스마트폰과 스마트밴드 사이의 데이터 송수신 과정에서 발생할 수 있다.

위에 언급된 위협 외에도 공격자는 STRIDE 위협 분석에서 파생된 다양한 위협을 결합하여 공격을 시도할 수 있다. 우리는 위협 모델링을 통해 얻은 위협과 공격 시나리오를 통해 스마트밴드의 연결 프로세스의 취약점을 발견하였고 이를 이용해 스마트밴드 시스템의 권한을 획득하였다. 공격 영상은 다음 웹 사이트(https://youtu.be/QFb1AV7yUas[25])에서 확인할 수 있다. 정상 사용자가 사용자의 스마트폰과 스마트밴드를 연결한 상태에서는 공격자가 자신의 스마트폰을 스마트밴드에 쉽게 연결할 수 없지만, 이미 연결된 스마트폰이 없는 경우에 공격자는 쉽게 스마트밴드에 접근할 수 있다 (Fig. 4. 참조). 이 공격은 스마트밴드에서 연결을 요청하는 스마트폰에 대한 사용자 인증 절차를 거치지 않았기 때문에 발생한다. 공격자는 이 취약점을 이용하여 정상 사용자로 위장하고 스마트밴드 시스템의 제어권을 획득하여 스마트밴드를 사용할 수 있다. 그러므로 안전한 연결 프로세스를 위하여 사용자 인증 절차가 반드시 필요하며 이를 4.2.2에서 제안한다. 이 공격실험을 통해서 본 논문의 위협 모델링 기법이 시스템 취약점을 탐지하는데 효과적이라는 사실을 입증할 수 있다.

IV. 보안 요구사항 및 보안 대책

본 절에서는 스마트밴드 시스템의 보안 요구사항을 도출하고 보안 요구사항을 만족하는 보안 대책을 제시한다. 이후 제시한 보안 대책의 안전성을 객관적으로 검증하기 위하여 자동 검증 도구인 Scyther를 사용하여 보안 대책을 검증한다.

4.1 보안 요구사항

보안 요구사항은 위협 모델링을 통해 얻은 위협과 공격을 고려하여 공격자가 최종 목표인 '스마트밴드 서비스 거부' 와 '사용자 정보 획득' 을 달성하지 못하도록 작성하였으며 자세한 내용은 다음과 같다.

- **타임스탬프 추가.** 재전송 공격(A1)에 대응하기 위해서는 모든 데이터 송수신 프로세스 및 연결 프로세스의 패킷에 타임스탬프를 추가해야 한다.
- **사용자 인증.** 공격자가 메시지를 수정하더라도 메시지 인증 프로세스를 통해서 이를 탐지할 수 있으며 조작된 정보가 유입되는 것을 막을 수 있다. 또한, 사용자 인증 프로세스는 공격자의 장치로 연결하는 공격(A4), 계정 도용(A5, A8), 로그인 절차 우회(A7) 등을 막을 수 있다.
- **메시지 암호화.** 메시지를 암호화하여 보내면 공격자가 본래 메시지를 읽을 수 없게 하며 중간에 가로채서 메시지를 조작하기 어렵게 만들 수 있다. 그러므로 메시지 조작 공격(A2), 네트워크 패킷 스니핑(A9) 등을 막기 위해 메시지를 암호화해야 한다. 또한, 데이터베이스 내 모든 중요한 데이터는 암호화하여 저장함으로써 공격자가 저장소를 물리적으로 획득하더라도 내부 데이터를 읽을 수 없도록 해야 한다.
- **트래픽 분석 및 침입 탐지.** 과도한 데이터 전송 공격(A3)은 완전히 차단하기 어렵다. 따라서 서버 측면에서 트래픽 분석과 침입 탐지를 수행하여 해당 공격에 대한 위협을 완화해야 한다.
- **안전한 계정 관리.** 모든 사용자는 계정 도용(A5, A8)에 대응하기 위해 계정을 안전하게 관리해야 하며 서비스 제공 업체는 사용자에게 계정 도용의 위험에 대하여 꾸준히 인지시켜야 한다. 또한, 데이터 저장소를 안전하게 관리하여 사용자의 계정 정보가 노출되지 않도록 해야 한다.

4.2 보안 대책

스마트밴드 시스템의 DFD 및 보안 요구사항을 살펴보면 두 부분에서 인증과 암호화가 필요하다는 것을 알 수 있다. 첫 번째 부분은 스마트폰과 웹 서버 사이에 통신하는 부분이다. 스마트폰과 웹 서버 사이 통신 패킷은 사용자에 대한 인증 정보를 갖고 있어야 하고 암호화하여 전송해야 한다. 두 번째 부분은 스마트폰과 스마트밴드 사이에 통신하는 부분이다. 이 경우, 사전에 언급한 연결 프로세스에서의 문제점을 해결하기 위해서 안전한 연결 설정이 필수적이다. 안전한 연결을 위해서는 스마트밴드에서 사용자 인증 과정을 통해 정상 사용자가 아니면 해당 메시지를 차

단하는 기능이 필요하다. 따라서 안전한 메시지 통신 프로토콜과 사용자 인증을 통한 안전한 연결 프로세스를 보안 대책으로 제시한다. 이후 제시한 보안 대책은 정형화 과정을 통해 수학적으로 안전성을 검증할 수 있도록 하였다.

4.2.1 스마트폰과 웹 서버 사이 통신

스마트폰과 웹 서버 사이 통신은 두 단계로 구분된다. 첫 번째 단계는 로그인 단계로 웹 서버와 통신하기 위해 사용자는 먼저 로그인을 하고 데이터를 웹 서버로 보낸다. 이 프로세스에서 재전송 공격, 로그인 우회, 계정 도용과 같은 공격이 발생할 수 있기 때문에 사용자 인증과 데이터 암호화 과정을 거쳐야 한다. 본인 인증을 위해 사용자의 ID, 타임스탬프, 그리고 PW의 해쉬값을 암호화하여 전송한다. 암호화는 웹 서버의 공개키를 사용하여 암호화하고 웹 서버에서는 이를 개인키로 복호화하여 해당 ID에 대한 PW의 해쉬값을 얻는다. 이때, 스마트폰과 웹 서버의 인증서는 안전한 채널을 통해 서로 교환되었다는 것을 전제로 한다. 웹 서버는 주어진 ID와 PW의 해쉬값을 검증하여 검증에 통과하면 해당 스마트폰의 로그인 과정을 마친다. 두 번째는 스마트폰과 웹 서버가 실제 데이터를 주고받는 단계이다. 이 경우에 데이터의 사이즈가 커질 수 있기 때문에 대칭키 암호 알고리즘을 사용해야 한다. 대칭키 암호에 사용할 키를 교환하기 위해 국-대-국 프로토콜 (station - to - station protocol)[26]을 이용할 수 있으며 이 외의 다른 안전한 키 교환 프로토콜을 사용할 수도 있다. 키를 교환한 후에는 데이터에 타임스탬프를 붙여 함께 암호화하여 전송한다. 스마트폰과 웹 서버 사이 통신의 보안 대책을 정형화시킨 내용은 Fig. 5.에서 살펴볼 수 있으며 절차는 다음과 같다.

1) 스마트폰은 웹 서버의 공개키를 사용하여 타임스탬프, 사용자 ID, PW의 해쉬값을 암호화한다.
2) 웹 서버는 개인키를 사용하여 메시지를 복호화하고 타임스탬프, 사용자 ID, PW의 해쉬값을 얻는다.
3) 웹 서버는 타임스탬프의 유효성을 확인하고 해당 ID 및 PW의 해쉬값을 검사한다. 정상적으로 검사를 통과하면, 웹 서버는 로그인을 승인한다.
4) 국-대-국 프로토콜을 사용하여 스마트폰과 웹 서버 사이 통신 메시지를 암호화하는 데 필요한 대칭키를 교환한다.
5) 스마트폰과 웹 서버는 교환한 대칭키를 사용하여 데이터와 타임스탬프를 암호화한 뒤 보낸다.

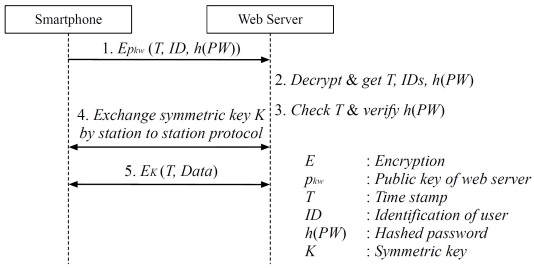


Fig. 5. Smartphone & web server protocol

4.2.2 스마트폰과 스마트밴드 사이 통신

공격자가 자신의 스마트폰을 사용자의 스마트밴드에 연결할 수 없도록 스마트폰과 스마트밴드 사이에는 안전한 연결 프로세스가 필요하다. 안전한 연결 설정을 위해서 스마트밴드는 사용자 인증 프로세스를 수행해야 하며 인증을 통과하지 못하면 연결 요청을 거부하거나 메시지를 수신하지 않도록 해야 한다. 스마트폰과 스마트밴드 사이에 대칭키를 안전하게 교환하면 해당 키를 사용하여 메시지를 암호화하여 송수신할 수 있으며 대칭키는 정상 사용자의 스마트폰과 스마트밴드만 갖고 있기 때문에 자체적으로 사용자 인증 기능을 수행할 수 있다. 본 논문에서 제안하는 대칭키 교환 프로세스는 Fig. 6.에서 살펴볼 수 있다. U1, U2, B1, B2, B3, P1, P2, P3는 각각 사용자(U), 스마트밴드(B), 스마트폰(P)의 동작 순서를 나타내며 이때 사용된 난수 생성기의 안전성 검증은 본 논문의 범위에서 벗어나므로 안전하다고 가정하였다. 키 교환 프로세스의 자세한 과정은 다음과 같다.

1) 스마트폰은 스마트밴드에 연결을 요청한다(P1).
2) 연결 요청을 받은 스마트밴드는 응답 메시지를 전송하며 특정 숫자를 화면에 출력한다(B1).
3) 스마트폰은 응답 메시지를 수신하고 사용자가 숫자를 입력할 수 있는 화면을 출력한다(P2).
4) 사용자는 스마트밴드의 숫자를 읽고 버튼을 클릭하고(U1) 스마트밴드는 숫자를 난수 생성기의 입력 값으로 사용하여 대칭키를 생성한다(B2).
5) 사용자는 숫자를 스마트폰에 입력하고(U2) 스마트 폰은 입력받은 숫자를 난수 생성기의 입력 값

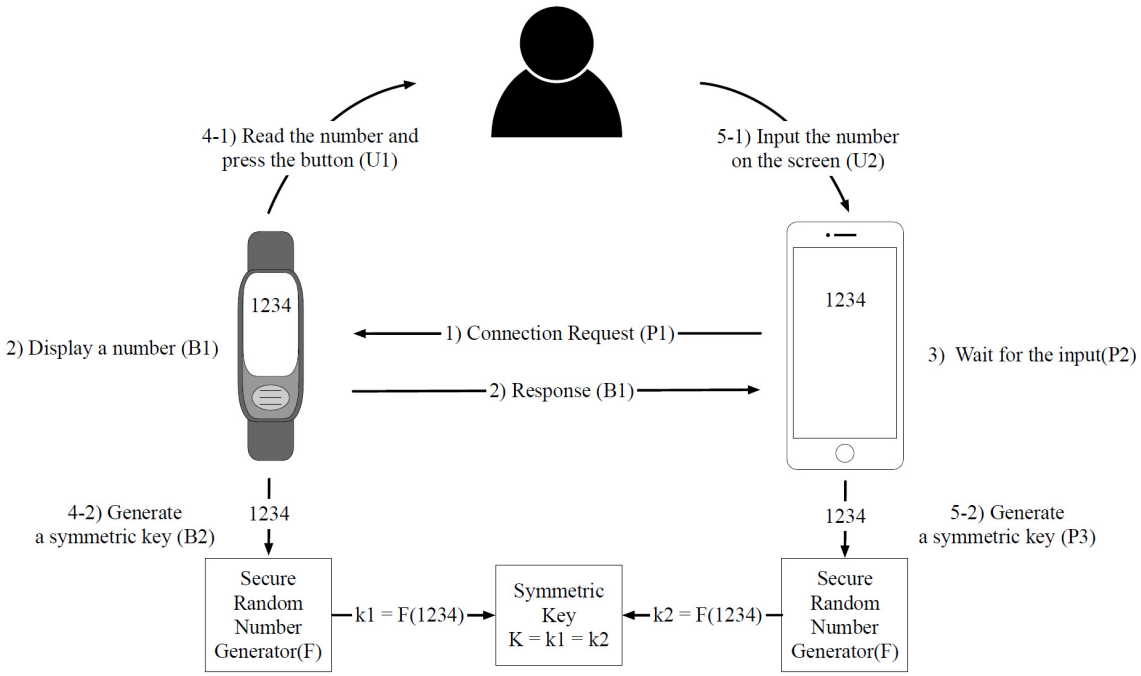


Fig. 6. Symmetric key generation process for secure connection

으로 사용하여 대칭키를 생성한다(P3).

본 대칭키 교환 프로세스는 초기 연결 설정에 사용되며 사용자가 숫자를 읽고 스마트폰에 직접 입력하는 것이기 때문에, 대칭키 교환 프로세스가 안전하다고 가정할 수 있다. 이후 연결에서는 연결 요청 메시지에 타임스탬프를 추가하여 동일한 대칭키로 암호화하여 보내야 하며 스마트밴드는 해당 메시지를 복호화했을 때, 정상적인 포맷의 데이터를 얻을 수 없으면 해당 메시지 수신을 거부한다. 연결 설정이 완료되면 추가적인 메시지를 보낼 때, 타임스탬프와 함께 교환한 대칭키로 암호화하여 송수신한다. 사용자가 스마트밴드를 새로운 스마트폰에 연결하려는 경우 이전 스마트폰에서 연결 해제 요청 메시지를 전송하여 스마트폰과 스마트밴드 사이 연결을 해제한다. 스마트폰과 스마트밴드 사이 통신의 보안 대책을 정형화시킨 내용은 Fig. 7.에서 살펴볼 수 있으며 절차는 다음과 같다.

1) 스마트폰 사용자는 사전에 교환한 대칭키를 사용하여 타임스탬프와 연결 요청 메시지를 암호화하여 전송한다.
2) 스마트밴드는 전달받은 패킷을 복호화하여 메시지를 확인하고 정상적인 포맷의 데이터를 얻을 수 없으면 해당 메시지 수신을 거부하고 정상적인 경우 연결 설정을 완료하고 타임스탬프와 응답 메시지를 암호화하여 전송한다.
3) 연결 이후 스마트폰과 스마트 밴드는 공유하고 있는 대칭키를 이용하여 타임스탬프와 데이터를 암호화한 뒤 보낸다.

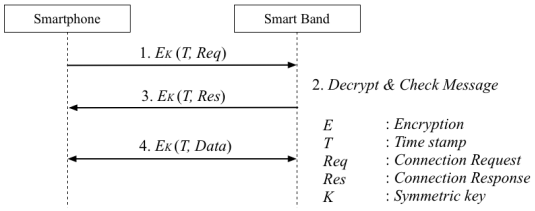


Fig. 7. Smartphone & smart band protocol

4.3 보안 대책 검증

4.2에서 보안 대책으로 제시한 프로토콜의 안전성을 검증하기 위하여 Scyther를 사용하였으며 검증한 결과, 코드, Scyther에 대한 추가 정보는 Github [27]에서 확인할 수 있다. 먼저 스마트폰과 웹 서버 사이 통신 프로토콜을 검증한다. 이때, 국-대-국 프

로토콜을 사용하여 대칭키를 교환하는 부분은 안전하다고 가정하여 검증에서 제외한다. 사용자 ID(*ID*), 사용자 암호(*PW*), 스마트폰과 웹 서버 사이 전송된 데이터(*PhoneData, ServerData*), 타임스탬프(*T*), 대칭키(*kir*)에 대하여 안전성을 검사하였고 프로토콜 검증 결과 모든 요소가 안전하며 공격자가 발견되지 않았다. 스마트폰과 웹 서버 사이 프로토콜의 검증 코드와 결과는 각각 Fig. 8.과 Fig. 9.에서 확인할 수 있다. 두 번째로 스마트폰과 스마트밴드 사이 통신 프로토콜을 검증한다. 이때, 대칭키를 생성하는 프로세스는 처음 연결 설정할 때에만 사용되며 사용자가 직접 읽은 숫자를 입력하기 때문에 안전하다고 가정하여 검증에서 제외한다. 스마트폰의 연결 요청 메시지(*connectionReq*), 스마트밴드의 연결 응답 메시지(*connectionRes*), 스마트폰과 스마트밴드 사이 전송된 데이터(*Phonedata, BandData*), 타임스탬프(*T*), 대칭키(*kir*)에 대하여 안전성을 검사하였고 프로토콜 검증 결과 모든 요소가 안전하며 공격자가 발견되지 않았다. 스마트폰과 스마트밴드 사

```
1 /* Communication protocol between smartphone and web server*/
2
3 hashfunction hash;
4
5 usertype Key;
6 usertype Timestamp;
7 usertype SmartPhoneData;
8 usertype WebServerData;
9 usertype Password;
10 usertype Identification;
11
12 protocol pw(P,W)
13 {
14     role P
15     {
16         const PW: Password;
17         const ID: Identification;
18         const kir: Key;
19         fresh T1: Timestamp;
20         fresh T2: Timestamp;
21         fresh PhoneData: SmartPhoneData;
22         var T3: Timestamp;
23         var ServerData: WebServerData;
24
25         send_1 (P, W, { T1, ID, hash(PW)}pk(W) );
26         send_2 (P, W, { T2, P, PhoneData}kir );
27         recv_3 (W, P, {T3, W, ServerData}kir);
28
29         claim_p1 (P, Secret, PW);
30         claim_p2 (P, Secret, ID);
31         claim_p3 (P, Secret, PhoneData);
32         claim_p4 (P, Secret, ServerData);
33         claim_p5 (P, Secret, kir);
34         claim_p6 (P, Secret, T1);
35         claim_p7 (P, Secret, T2);
36     }
37
38     role W
39     {
40         const PW: Password;
41         const ID: Identification;
42         const kir: Key;
43         var T1: Timestamp;
44         var T2: Timestamp;
45         var PhoneData: SmartPhoneData;
46         fresh T3: Timestamp;
47         fresh ServerData: WebServerData;
48
49         recv_1( P, W, { T1, ID, hash(PW)}pk(W) );
50         recv_2 (P, W, { T2, P, PhoneData}kir );
51         send_3 (W, P, {T3, W, ServerData}kir);
52
53         claim_w1 (W, Secret, PW);
54         claim_w2 (W, Secret, ID);
55         claim_w3 (W, Secret, PhoneData);
56         claim_w4 (W, Secret, ServerData);
57         claim_w5 (W, Secret, kir);
58         claim_w6 (W, Secret, T1);
59         claim_w7 (W, Secret, T2);
60     }
61 }
```

Fig. 8. Scyther verification code of smartphone & web server protocol

Scyther results : verify

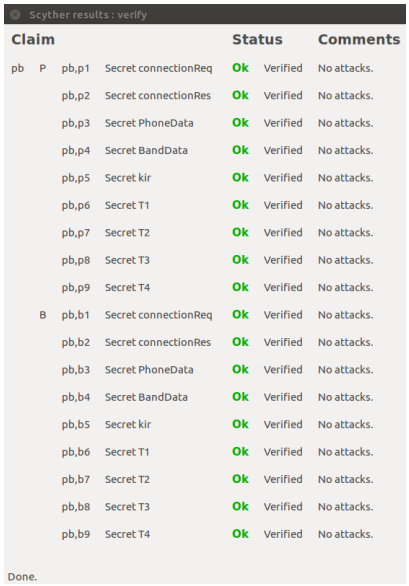
Claim				Status		Comments
pw	P	pw,p1	Secret PW	Ok	Verified	No attacks.
		pw,p2	Secret ID	Ok	Verified	No attacks.
		pw,p3	Secret PhoneData	Ok	Verified	No attacks.
		pw,p4	Secret ServerData	Ok	Verified	No attacks.
		pw,p5	Secret kir	Ok	Verified	No attacks.
		pw,p6	Secret T1	Ok	Verified	No attacks.
		pw,p7	Secret T2	Ok	Verified	No attacks.
	W	pw,w1	Secret PW	Ok	Verified	No attacks.
		pw,w2	Secret ID	Ok	Verified	No attacks.
		pw,w3	Secret PhoneData	Ok	Verified	No attacks.
		pw,w4	Secret ServerData	Ok	Verified	No attacks.
		pw,w5	Secret kir	Ok	Verified	No attacks.
		pw,w6	Secret T1	Ok	Verified	No attacks.
		pw,w7	Secret T2	Ok	Verified	No attacks.

Done.

Fig. 9. Scyther verification result of smartphone & web server protocol

```
1 /* Connection and Communication protocol between smartphone and smart band*/
2
3 hashfunction hash;
4
5 usertype Key;
6 usertype Timestamp;
7 usertype ConnectionRequest;
8 usertype ConnectionResponse;
9 usertype SmartPhoneData;
10 usertype SmartBandData;
11
12 protocol pb(P,B)
13 {
14     role P
15     {
16         const kir: Key;
17         fresh T1: Timestamp;
18         fresh T3: Timestamp;
19         fresh connectionReq: ConnectionRequest;
20         fresh PhoneData: SmartPhoneData;
21         var T2: Timestamp;
22         var T4: Timestamp;
23         var connectionRes: ConnectionResponse;
24         var BandData: SmartBandData;
25
26         send_1 (P, B, { T1, connectionReq}kir );
27         recv_2 (B, P, { T2, connectionRes}kir );
28         send_3 (P, B, { T3, P, PhoneData}kir);
29         recv_4 (B, P, { T4, B, BandData}kir );
30
31         claim_p1 (P, Secret, connectionReq);
32         claim_p2 (P, Secret, connectionRes);
33         claim_p3 (P, Secret, PhoneData);
34         claim_p4 (P, Secret, BandData);
35         claim_p5 (P, Secret, kir);
36         claim_p6 (P, Secret, T1);
37         claim_p7 (P, Secret, T2);
38         claim_p8 (P, Secret, T3);
39         claim_p9 (P, Secret, T4);
40     }
41
42     role B
43     {
44         const kir: Key;
45         var T1: Timestamp;
46         var T3: Timestamp;
47         var connectionReq: ConnectionRequest;
48         var PhoneData: SmartPhoneData;
49         fresh T2: Timestamp;
50         fresh T4: Timestamp;
51         fresh connectionRes: ConnectionResponse;
52         fresh BandData: SmartBandData;
53
54         recv_1( P, B, { T1, connectionReq}kir );
55         send_2 (B, P, { T2, connectionRes}kir );
56         recv_3 (P, B, { T3, P, PhoneData}kir);
57         send_4 (B, P, { T4, B, BandData}kir );
58
59         claim_b1 (B, Secret, connectionReq);
60         claim_b2 (B, Secret, connectionRes);
61         claim_b3 (B, Secret, PhoneData);
62         claim_b4 (B, Secret, BandData);
63         claim_b5 (B, Secret, kir);
64         claim_b6 (B, Secret, T1);
65         claim_b7 (B, Secret, T2);
66         claim_b8 (B, Secret, T3);
67         claim_b9 (B, Secret, T4);
68     }
69 }
```

Fig. 10. Scyther verification code of smartphone & smart band protocol


Scyther results : verify

Claim				Status		Comments
pb	P	pb,p1	Secret connectionReq	Ok	Verified	No attacks.
		pb,p2	Secret connectionRes	Ok	Verified	No attacks.
		pb,p3	Secret PhoneData	Ok	Verified	No attacks.
		pb,p4	Secret BandData	Ok	Verified	No attacks.
		pb,p5	Secret kir	Ok	Verified	No attacks.
		pb,p6	Secret T1	Ok	Verified	No attacks.
		pb,p7	Secret T2	Ok	Verified	No attacks.
		pb,p8	Secret T3	Ok	Verified	No attacks.
		pb,p9	Secret T4	Ok	Verified	No attacks.
	B	pb,b1	Secret connectionReq	Ok	Verified	No attacks.
		pb,b2	Secret connectionRes	Ok	Verified	No attacks.
		pb,b3	Secret PhoneData	Ok	Verified	No attacks.
		pb,b4	Secret BandData	Ok	Verified	No attacks.
		pb,b5	Secret kir	Ok	Verified	No attacks.
		pb,b6	Secret T1	Ok	Verified	No attacks.
		pb,b7	Secret T2	Ok	Verified	No attacks.
		pb,b8	Secret T3	Ok	Verified	No attacks.
		pb,b9	Secret T4	Ok	Verified	No attacks.

Done.

Fig. 11. Scyther verification result of smartphone & smart band protocol

이 프로토콜의 검증 코드와 결과는 각각 Fig. 10.과 Fig. 11.에서 확인할 수 있다.

V. 결 론

본 논문에서는 DFD, STRIDE 및 Attack Tree와 같은 위협 모델링 기법을 사용하여 스마트밴드 시스템에 대한 취약점과 보안 요구사항을 도출하였다. 위협 모델링을 통해 얻은 위협과 공격을 분석하여 연결 프로세스에서의 취약점을 발견했고 이 취약점을 이용하여 스마트밴드 시스템 권한을 획득할 수 있었다. 이는 스마트밴드 시스템에 적용한 위협 모델링 기법이 취약점을 발견하는 데 효과적이었음을 입증하며 보안 요구사항 분석이 스마트밴드 시스템을 안전하게 재구축하는 데 유용하게 사용될 수 있음을 보여준다. 이후 연결 프로세스에서의 문제점과 다른 가능한 공격들에 대응할 수 있는 보안 대책을 제안하였다. 제안한 보안 대책들을 수학적으로 검증할 수 있도록 정형화하였으며 Scyther를 사용하여 우리가 제시한 보안 대책이 안전하며 공격자를 발견할 수 없음을 보였다. 위협 모델링을 통해 발견한 위협과 발생 가능한 공격에 대응하기 위한 보안 대책을 설립한 후에는 반드시 보안 대책을 고려하여 시스템을 재구축해야 한다. 하지만 보안 대책을 시스템에 적용하는 과정에서 키 관리와 같은 새로운 문제점이 발생할 수 있다. 그러므로 더욱 안전한 스마트밴드 시스템을 구축하기 위해, 시스템을 재구축한 다음 본 논문에서 제안한 방법들을 반복적으로 적용하여 시스템의 안전성을 검증해야 한다.

References

[1] W. Zhou and S. Piramuthu, "Security /privacy of wearable fitness tracking IoT devices," in *Proc. 9th Iberian Conf. Inf. Syst. Technol.*, pp. 1-5, Jun. 2014

[2] M. LEE, K. Lee, J. Shim, S. J. Cho, and J. Choi, "Security threat on wearable services: empirical study using a commercial smartband," in *Consumer Electronics-Asia (ICCE-Asia), IEEE International Conference on*. IEEE, pp. 1-5, Oct. 2016

[3] R. Goyal, N. Dragoni, and A. Spognardi, "Mind the tracker you wear: A security analysis of wearable health trackers," in *Proc. ACM Symp. Appl. Comput.*, pp. 131-136, Apr. 2016

[4] S. Seneviratne, Y. Hu, T. Nguyen, G. Lan, S. Khalifa, K. Thilakarathna, M. Hassan, and A. Seneviratne, "A survey of wearable devices and challenges," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2573-2620, Jul. 2017

[5] NIST, Trustworthy information system [Online]. Available: https://www.nist.gov/itl /trustworthy-information-systems. Accessed on: Oct 23, 2018

[6] H.F. Tipton and M. Krause, Information Security Management Handbook, CRC Press, May. 2007

[7] A. Shostack, *Threat Modeling: Designing for Security*, 1st ed. John Wiley & Sons, 2014

[8] A. Shostack, "Experiences Threat

Modeling at Microsoft", Modeling Security Workshop, Toulouse, Sep. 2008

[9] STRIDE threat model of Microsoft. [Online]. Available: https://msdn.microsoft.com/en-us /library/ee823878(v=cs.20).aspx. Accessed on: Oct 23, 2018

[10] I. Williams, X. Yuan, "Evaluating Effectiveness of Microsoft Treat Modeling Tool", *ISCD Conference*, 2015, Oct. 2015

[11] B. Schneier, "Attack trees: Modeling security threats," *Dr. Dobb's Journal*, Dec. 1999

[12] C. Cremers, "The Scyther Tool: Verification, falsification, and analysis of security protocols," in *Computer Aided Verification (CAV)*, ser. LNCS, vol. 5123. Springer, pp. 414-418, Jul. 2008

[13] M. Rahman, B. Carbunar, and M. Banik, "Fit and vulnerable: Attacks and defenses for a health monitoring device," *arXiv 1304.5672*, Apr. 2013

[14] H. Fereidooni, T. Frassetto, M. Miettinen, A.-R. Sadeghi, and M. Conti. "Fitness Trackers: Fit for Health but Unfit for Security and Privacy," in *Connected Health: Applications, Systems and Engineering Technologies (CHASE), 2017 IEEE/ACM International Conference on*. IEEE, pp. 19-24, Jul. 2017

[15] P.K. Akshay Dev and K.P. Jevitha, "STRIDE Based Analysis of the Chrome Browser Extensions API," in *Proceedings of the 5th International Conference on Frontiers in Intelligent Computing: Theory and Applica- tions : FICTA 2016, Volume 2*, S. C. Satapathy, V. Bhateja, S. K. Udgata, and P. K. Pattnaik, Eds., ed Singapore: Springer Singapore, pp. 169-178, Mar. 2017

[16] A. Karahasanovic, P. Kleberger, M. Almgren, "Adapting Threat Modeling Methods for the Automotive Industry," [Online]. Available: http://publications.lib.chalmers.se/records/fulltext/252083/local_252083.pdf. Accessed on: Oct 23, 2018

[17] M. Cagnazzo, M. Hertlein, T. Holz, and N. Pohlmann "Threat Modeling for Mobile Health Systems," in *Wireless Communications and Networking Conference Workshops (WCNCW), 2018 IEEE*. IEEE, pp. 314-319, Apr. 2018

[18] Tae Un Kang and Huy Kang Kim, "VR Threat Analysis for Information Assurance of VR Device and Game System," *Jonornal of The Korea Institute of information Security & Cryptology*, 28(2), pp. 437-447, Apr. 2018

[19] Hye Min Kim and Huy Kang Kim, "Threat Modeling and Risk Analysis: PS4 Remote Play with PC," *Jonornal of The Korea Institute of information Security & Cryptology*, 28(1), pp. 135-143, Feb. 2018

[20] D. Basin, C. Cremers, and S. Meier, "Provably Repairing the ISO/IEC 9798 Standard for Entity Authentication," *Proc. 1st Int'l Conf. Principles of Security and Trust* (POST 12), LNCS 7215, P. Degano and J.D. Guttman, eds., pp. 129-148, Dec. 2012

[21] C. Cremers, "Key exchange in IPsec revisited: formal analysis of IKEv1 and IKEv2," in *European conference on research in computer security (ESORICS)*, Leuven, Belgium, Sep. 2011

[22] Microsoft, SDL Threat Modeling Tool. [Online]. Available: https://www.microsoft .com/en-us/sdl/adopt/threatmode

ling.aspx. Accessed on: Sep 21, 2018.
[23] B. Potter, "Microsoft SDL threat modelling tool," *Network Security*, vol. 2009, no. 1, pp. 15-18, Jan. 2009
[24] A.K. Das, P.H. Pathak, C.-N. Chuah, and P. Mohapatra, "Uncovering privacy leakage in BLE network traffic of wearable fitness trackers," in *Proc. 17th Int. Workshop Mobile Comput. Syst. Appl. (HotMobile)*, pp. 99-104, Feb. 2016
[25] Smart band attack demo video, [Online]. Available: https://youtu.be/QFb1AV7yUas. Accessed on: Oct 23, 2018.
[26] W. Diffie, P. C. Van Oorschot, and M. J. Wiener, "Authentication and authenticated key exchanges," *Designs, Codes, Cryptography.*, vol. 2, no. 2, pp. 107-125, Jun. 1992
[27] Scyther verification code for connection and communication protocol, [Online]. Available: https://github.com/hausdorfff/Protocol-Verification. Accessed on: Oct 23, 2018.

〈저자소개〉

강 수 인 (Suin Kang) 학생회원
2017년 2월: 서울시립대학교 수학과 졸업
2017년 3월~현재: 고려대학교 정보보호학과 석사과정
〈관심분야〉 정보보호, 데이터 분석, IoT 보안, 자동차 보안, Anomaly Detection

김 혜 민 (Hye Min Kim) 학생회원
2017년 2월: 고려대학교 컴퓨터학과 졸업
2017년 3월~현재: 고려대학교 정보보호학과 석사과정
〈관심분야〉 데이터마이닝, 데이터 분석, 온라인게임 보안

김 휘 강 (Huy Kang Kim) 종신회원
1998년 2월: KAIST 산업경영학과 학사
2000년 2월: KAIST 산업공학과 석사
2009년 2월: KAIST 산업및시스템공학과 박사
2004년 5월~2010년 2월: 엔씨소프트 정보보안실장, Technical Director
2010년 3월~2014년 12월: 고려대학교 정보보호대학원 조교수
2015년 1월~현재: 고려대학교 정보보호대학원 부교수
〈관심분야〉 온라인게임 보안, 네트워크 보안, 네트워크 포렌식