KIPS Tr. Comp. and Comm. Sys.
Vol.6, No.3 pp.143~158 pISSN: 2287-5891

https://doi.org/10.3745/KTCCS.2017.6.3.143

Threat-Based Security Analysis for the Domestic Smart Home Appliance

Paul Hong[†] · Sangmin Lee[††] · Minsu Park[†††] · Seungjoo Kim[††††]

ABSTRACT

Smart Home Appliance which makes people to operate machines in the home by remote control is service or technology to provide convenience. It is close to home, so it has privacy problem and security problem. If Smart Home Applications is attacked, Scale of damage is anticipated. In case of products from overseas country, various vulnerability has been announced every year. Therefore, It is necessary to identify and to analysis threats of Smart Home Appliance using systematically method for using safe Smart home appliance service. In this paper, we present check list for identifying and analyzing threats using Threat Modeling and then we analyzed the Domestic Smart Home Appliance using check list which we present.

Keywords : Threat Modeling, Smart Home Appliance, Threat Analysis

위협 모델링을 이용한 국내 스마트 홈 보안 분석에 대한 연구

홍 바 울[†] · 이 상 민[††] · 박 민 수[†††] · 김 승 주[††††]

요 약

스마트 홈은 집에서 사용하는 가전기기를 집 밖에 있는 사람이 원격으로 제어할 수 있는 기술 및 서비스로서 사용자에게 편의성을 제공한다. 사람과 가장 밀접한 장소인 집 내부에 서비스를 제공하기 때문에 공격자에 의한 악의적인 기기제어, 사생활 침해와 같은 보안 사고가 있는 경우 피해 규모가 클 것으로 예상된다. 현재 해외 제품의 경우 가전기기의 취약점들이 매년 지속적으로 발표되고 있으며 이러한 취약점들은 사용자의 안전과 개인 자산에 심각한 문제를 발생할 수 있다. 따라서 국내의 스마트 홈 서비스를 안전하게 이용하기 위해서는 스마트 홈 환경에서 발생할 수 있는 전체적인 위협을 체계적으로 식별하고 분석하는 것이 필요하다. 본 논문에서는 위협 모델링을 사용하여 스마트 홈 환경에서 발생할 수 있는 위협을 체계적으로 식별하고 스마트 홈 환경의 보안을 분석하기 위한 체크리스트를 제안한다. 제안한 체크리스트의 실효성을 위해 실제 서비스에 적용하여 스마트 홈을 분석한다.

키워드 : 위협모델링, 스마트 홈, 위협 분석

1. 서 론

사물인터넷(이하 IoT: Internet of Things)이라는 용어는 1999년 MIT 공과대학의 Kevin Ashton이 “향후 RFID와 기타 센서를 일상생활에 사용하는 사물에 탑재한 사물인터넷이 구축될 것”이라고 전망하면서 처음으로 사용되었다. 2005년 ITU(International Telecommunication Union)는 IoT의 개념을 언제, 어디서나, 어떤 것이든 연결해주는 기술이라고 정의하였으며 현재에 이르러서는 각 기관마다 IoT의 개념을 조금씩 다르게 정의하고 있다[1, 2]. 그러나 공통적으로 IoT는 사람과 사람, 사람과 사물, 사물과 사물간의 정보를 상호 소통하는 기술 및 서비스로 보고 있다. 미래창조과학부에 따르면 IoT의 서비스는 크게 개인 IoT, 산업 IoT, 공공 IoT로 분류할 수 있다[3]. 스마트 홈은 사람과 사물이 통신하는 개인 IoT로 분류되며 국내의 경우 2015년 A사에서 서비스 시작을 필두로, B사 C사가 서비스를 제공하고 있다. 시장조사기관 STRATEGY ANALYTICS는 전 세계 스마트 홈 시장은 2014년 480억 달러에서 2019년 1150억 달러 수준으로 급성장할 것으로 전망한다[4]. 스마트 홈은 시장이 성장함에 따라 사용자에게 편의성은 확대될 것으로 보이나 현재 스마트 홈 서비스의 보안에 대한 반영은 미흡하다. 해외 컨퍼런

※ 이 논문은 2016년 대한민국 교육부와 한국연구재단의 지원을 받아 수행된 연구임(NRF- 2016S1A3A2924760).
† 준 회 원 : 고려대학교 정보보호대학원 정보보호학과 석사과정
†† 비 회 원 : 고려대학교 정보보호대학원 정보보호학과 석사과정
††† 준 회 원 : 고려대학교 정보보호대학원 정보보호학과 박사과정
†††† 종신회원 : 고려대학교 사이버국방학과/정보보호대학원 정교수
Manuscript Received : November 11, 2016
First Revision : November 25, 2016
Accepted : December 10, 2016
* Corresponding Author : Seungjoo Kim(skim71@korea.ac.kr)

스 Black Hat 2013에서 Behrang Fouladi는 Z-Wave 프로토콜상의 취약점을 발표하였으며[5], Black Hat USA 2015에서 Tobias Zillner는 Zigbee 프로토콜을 사용하는 스마트 홈 Hub와 스마트 전구에 대한 취약점을 발표하였다[6]. 2016년 SchmooCon에서 Joseph Hall은 Z-wave 프로토콜을 사용하는 스마트 홈 제품의 제어권을 획득하는 방안에 대해 발표하였다[7]. 이와 같이 가전기기의 취약점이 지속적으로 발표됨에 따라 안전한 스마트 홈 서비스를 위한 보안성 분석이 필요하다. 기존의 스마트 홈의 연구는 기기의 취약점을 발견하는 수준이며 서비스의 모든 범위를 분석한 연구는 미비하다. 따라서 본 연구에서는 위협 모델링 기법을 사용하여 스마트 홈의 전체적인 위협을 식별 및 분석하고 이를 점검하기 위한 체크리스트를 도출한다.

본 논문의 구성은 다음과 같다. 2장에서는 스마트 홈 서비스에 대한 설명과 스마트 홈과 관련된 보안 연구를 설명한다. 3장에서는 위협 모델링 기법을 적용하여 보안 위협을 도출하고 분석하여 체크리스트를 도출한다. 4장에서는 3장에서 도출한 체크리스트를 바탕으로 실제 서비스에 적용하여 보안성 분석을 시행한다. 마지막 5장에서는 본 논문의 결론 및 향후 연구방향을 서술한다.

2. 관련 연구

2.1 스마트 홈

스마트 홈 서비스는 집안에서 사용하는 가전기기에 네트워크를 연결하여 기기를 원격으로 제어할 수 있도록 하는 서비스이다. 스마트 홈 서비스에 포함되어 있는 가전기기들은 네트워크의 연결 방법에 따라 클라우드 연결 디바이스, 직접 연결 디바이스, Non-IP 디바이스 3가지 형태로 분류된다. 본 논문에서는 국내의 스마트 홈 서비스를 대상으로 분석하며 국내는 Non-IP 디바이스 형태의 서비스가 제공된다. Non-IP 디바이스는 서비스 플랫폼을 이용하기 위해서 공유기와 가전 기기 사이에 존재하는 허브와 연결되는 기기를 말하며 Fig. 1은 서비스 구성을 보여준다. 사용자의 스마트폰, 테블릿과 같은 스마트기기를 통해 기기를 제어하는 명령을 서버로 전송하며, 서버는 이를 각 가정으로 전송한다. 각 가전기기는 공유기와 연결되는 Hub를 통해 Z-Wave 프로토콜을 이용하여 통신하며 서버로부터 오는 명령을 실행하는 환경으로 되어 있다.

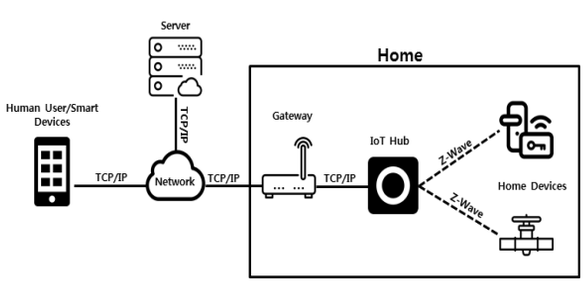


Fig. 1. Non-IP Smart Home Service Concept

국내의 스마트 홈 서비스는 2015년 처음으로 시작되었으며 서비스를 이용하는 가구가 빠르게 증가하고 있다. Table 1은 국내 통신 3사의 서비스를 나타낸다. 국내의 통신 3사가 서비스를 제공하고 있으며 플러그, 열림 감지 센서, 가스 차단기, 도어락을 주요 서비스 기기로 제공하고 있다. Hub의 형태는 AP 또는 USB Dongle 형태로 제공되며 공유기와 가전기기 사이에서 Z-Wave 프로토콜로 통신을 할 수 있게 한다.

Table 1. Home IoT Service Domestic Provider

Company	A사	B사	C사
Main Device	· Plug · Window Sensor · Gas Lock · Door Lock · Switch · Energy meter	· Plug · Window Sensor · Gas Lock · Door Lock	· Plug · Window Sensor · Gas Lock · Door Lock · Switch
Hub	AP or USB	AP	AP

2.2 스마트 홈 보안 관련 연구

1) 취약점 연구

스마트 홈 취약점에 대한 연구는 BlackHat, SECUINSIDE, ShmooCon과 같은 컨퍼런스에서 지속적으로 발표되고 있다. Sahand Ghanoun(2013)는 BlackHat에서 Z-Wave를 사용하는 스마트 홈을 공격하기 위해서 Z-Wave 무선 프로토콜을 분석하고 캡쳐하는 도구를 소개하였으며[5], D. Crowley는 스마트 홈 Hub 중 하나인 VeraLite에 CSRF(Cross-site request forgery) 취약점과 인증 우회를 통해 root 계정 획득에 대한 취약점을 찾아 임의의 코드를 실행하는데 성공한 사례를 소개했다[8]. 2014년 BlackHat USA에서는 Grant Hernandez가 스마트 온도 조절기 Nest의 구조를 분석하고 각 구조별 공격벡터를 식별하여 이에 대한 공격을 수행한 것을 발표하였다[9]. 2015년 SECUINSIDE에서 mugmung은 Hack RF를 이용한 RF 신호 재전송을 통해 스마트 홈 장비에 대한 공격을 발표하였으며[10], Tobias Zillner(2015)는 BlackHat USA에서 Zigbee 프로토콜을 분석하고 이를 공격하기 위한 Kellerbee라는 도구를 소개하였으며 Zigbee를 이용하는 스마트 홈 시스템에 대한 성공한 공격 사례를 소개했다[6]. Joseph Hall(2016)은 ShmooCon 2016에서 Z-Wave 프로토콜을 사용하는 스마트 홈 제품의 제어권 획득에 관한 내용을 발표하였으며 현재 사용되고 있는 Z-Waave 공격 도구의 한계를 지적하고 개선된 공격 도구인 EZ-Wave를 소개하였다[7].

컨퍼런스 이외에 논문에서는 2013년 Thomas Reuter가 스마트 홈의 펌웨어를 덤프하여 기능을 분석하였으며, 통신 프로토콜 분석을 통해 Sniffing, Injection, Fuzzing 공격을 수행하고 이에 대한 분석 결과를 설명하였다[11]. 2015년 D. Fuller는 Z-Wave를 사용하는 스마트 홈 환경에서 악성 Z-Wave Controller를 통해 IoT Device들의 조작가능성을 보였으며[12], 2016년 Fernandes는 삼성 SmartThings를 대상

으로 Hub에 설치되는 여러 IoT 기기에 대한 공격 시나리오를 작성하고 이에 대한 Exploit platform을 구현하였다[13]. 현재까지 컨퍼런스 및 논문에서 발표된 취약점들은 대다수 통신 프로토콜을 이용한 스마트 홈의 기기 제어에 관한 것이었으며, 스마트 홈 전체적인 취약점을 점검하기 위한 체계적인 연구는 미비하다.

2) OWASP Internet of Things Project

OWASP(The Open Web Application Security Project)는 제조사와 개발자, 소비자에게 IoT와 관련된 보안 이슈를 보다 이해하기 쉽도록 하고 IoT 기술을 적용할 때 적합한 보안상의 결정을 내릴 수 있도록 도와주기 위해 OWASP Internet of Things Project를 진행하고 있다[14]. IoT의 공격지점에 관한 IoT Attack Surface Area에 대한 연구와, IoT의 취약점을 분류하는 IoT Vulnerabilities Project, 펌웨어를 분석하기 위한 Firmware Analysis Project가 진행 되었다[15, 16]. Internet of Things Project는 IoT에 대한 공격 지점을 19개로 분류 하고 있으며 각각의 공격지점에 대한 취약점을 일부 나열하고 있다. 각각의 취약점은 IoT Vulnerabilities에서 요약하여 설명해주며 Firmware Analysis는 Attack Surface의 Device Firmware에 대한 Attack Surface를 테스트에 활용하기 위한 가이드로서 제공한다.

3) Oracle의 IoT Security Architecture

2014년 Oracle에서 개최한 비즈니스 및 기술 컨퍼런스인 ORACLE OPENWORLD에서 Java Platform Group에 속해있는 Noel Poore가 'IoT Security Architecture'란 주제로 전반적인 IoT 플랫폼 구조상의 보안에 관해 발표하였다[17]. 발표된 자료에 따르면 IoT에 대한 위협 모델링은 복잡하고 큰 시스템에 대한 것이며 다양한 위협에 고려되어야 한다고 언급하고 있다. 그러나 실제로 수행한 위협 모델링에 대해서 보여주고 있지 않다. 추가로 IoT 상에서 발생할 수 있는 위협에 대해 도출한 트리를 제시한다. IoT 위협을 DDoS, Reconnaissance, Info Disclosure, Device Spoof/MITM, Poor Management로 크게 6가지로 분류하였으며 각각의 하위 노드에 해당하는 부분에 일부 상세한 위협사항들을 표기하였다.

4) Microsoft의 IoT 위협 모델

Microsoft는 오래전부터 제품에 대해 위협 모델링을 사용해 왔으며, Microsoft의 위협 모델링 프로세스는 외부에 공개되어 있다. Microsoft의 경우 클라우드 IoT 플랫폼을 제공하고 있으며, 공격자가 시스템을 손상시킬 수 있는 방법을 파악하여 적절한 위협 완화 조취를 위해 클라우드 환경에 대한 IoT 위협 모델링을 수행하였다[18]. IoT 기기, 현장 게이트웨이, 클라우드 게이트웨이, 서비스 영역으로 크게 4가지 부분으로 구분했다. 각 부분에 대한 DFD를 작성하였으며, 각 요소에 대해 STRIDE를 적용하여 위협 모델링을 수행하였다. DFD는 전체적인 서비스를 확인할 수 있는 level 1정도로 작성되었으며, STRIDE는 프로세스, 장치간 통신, DFD내부의 저장소에 대해서 수행하였다. 그러나 DFD를 level 1정도로 작성하는 경우 세부적인 데이터의 흐름이 명확하게 보이지 않아, STRIDE를 상세하게 도출되지 못할 수 있다는 단점이 있다.

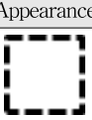
3. 스마트 홈에 대한 위협 모델링

스마트 홈 환경은 하나의 애플리케이션이 아닌 복잡하며 다양한 환경으로 구성되어 있다. 본 논문에서는 전체적인 스마트 홈 환경에 대한 위협 분석을 위해 위협 모델링을 수행한다. STRIDE로 추상적인 위협을 도출하고, 도출한 위협을 바탕으로 공격 시나리오를 작성하기 위한 Attack Tree를 작성한다. Attack Tree의 목표를 통해 위험을 분석하며, 현재까지 알려진 공격을 활용하여 위협을 상세화하기 위해 Attack Libraries를 수집한다. 최종적으로 앞선 결과를 바탕으로 체크리스트를 도출한다.

3.1 데이터 흐름 다이어그램 도출(Data Flow Diagram)

DFD는 위협 모델링 단계에서 첫 번째로 수행되며 네트워크나 설계된 시스템에 데이터의 흐름을 추상적으로 보여주기 위해 일반적으로 사용된다. DFD는 주요 요소로 프로세스(Process), 데이터 흐름(Data Flow), 데이터 저장소(Data Store), 외부 객체(External Entity)로 구성된다. Table 2는 각 DFD(Data flow Diagram)의 요소와 표현 방법을 보여 준다.

Table 2. Elements of a Data Flow Diagram

Element	Appearance	Meaning
Trust Boundary		Anyplace where various principals come together
External Entity		People, or code outside your control
Data Store		Things that store data
Process		Any running code
Data Flow		Communication between processes, or between processes and data stores

DFD는 가장 큰 시스템을 포괄적인 형태로 나타내는 level 0를 그리는 것으로 시작한다. 이후 데이터를 분할하여 한 단계 더 분화하여 작성한 것이 level 1 DFD이다. DFD는 데이터가 더 분할될 수 있는 경우에는 더 분할해서 표현하며 DFD의 종료 시점은 가능한 한 가장 상세하게 데이터를 분할해서 더 이상 분할할 수 없는 경우까지 시행한다. Fig. 2는 level 0의 DFD로써 스마트 홈의 전체 시스템을 하나의

프로세스로 보고 포괄적으로 그린 것이다. 그러나 level 0의 DFD로는 실제 데이터를 식별하기 어렵기 때문에 보다 데이터를 분할하여 보다 상세하게 작성한다.

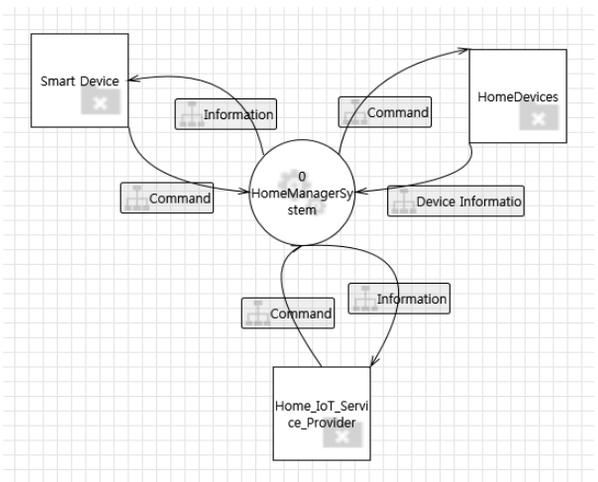


Fig. 2. Level 0 DFD for Smart Home Appliance

Fig. 3은 level 0 DFD를 분할하여 보다 상세하게 작성한 DFD이다. 각각은 가전기기에 원격으로 명령을 지시하는 Smart Device Boundary, 명령을 수행하는 가전기기가 존재하는 Smart Home Boundary, 중간에 서비스를 제공하는 Service Boundary로 총 3가지의 Boundary로 분류한다. 그러나 level 1 DFD로는 위협을 분석하기에는 충분하지 못하다. 실제 오고가는 데이터가 세분화 되어 있지 않으며, 포괄적인 프로세스로만 구성되어 있다. 따라서 한 단계 더 세분화 하여 DFD를 구성한다.

Fig. 4는 스마트 홈에 대한 DFD로서 위협 분석을 위해 level 2까지 작성한 것이다. 3개의 External Entity와 13개의 Process, 3개의 Data Store, 55개의 Data Flow로 구성되어 있다. 스마트 홈의 DFD는 데이터의 흐름에 따라서 Smart Device Boundary, Service Boundary, Smart Home Boundary, Device to Device Boundary로 크게 4가지 부분으로 구성되며 이를 Trust Boundary로 표시한다. Smart Device Boundary는 가정 내부의 가전기기를 원격으로 제어하기 위한 부분으로서 사용자와 테블릿, 스마트 폰과 같은 스마트 기기가 이 부분에 해당한다. Service Boundary는 플랫폼 서비스를 제공하는 부분으로서, 기기의 정보와 사용자 계정에 대한 정보가 담겨있는 서버가 이 부분에 해당한다. Smart Home Boundary는 집 내부로서 외부 인터넷과 연결되어 있는 내부망으로써, 가전기기와 공유기를 연결하는 IoT Hub가 이 부분에 해당한다. 마지막으로 Device to Device Boundary는 가정 내부에서, Hub와 가전 기기가 Z-Wave 프로토콜로 통신하는 구간으로써 가전기기들만 이 부분에 해당한다.

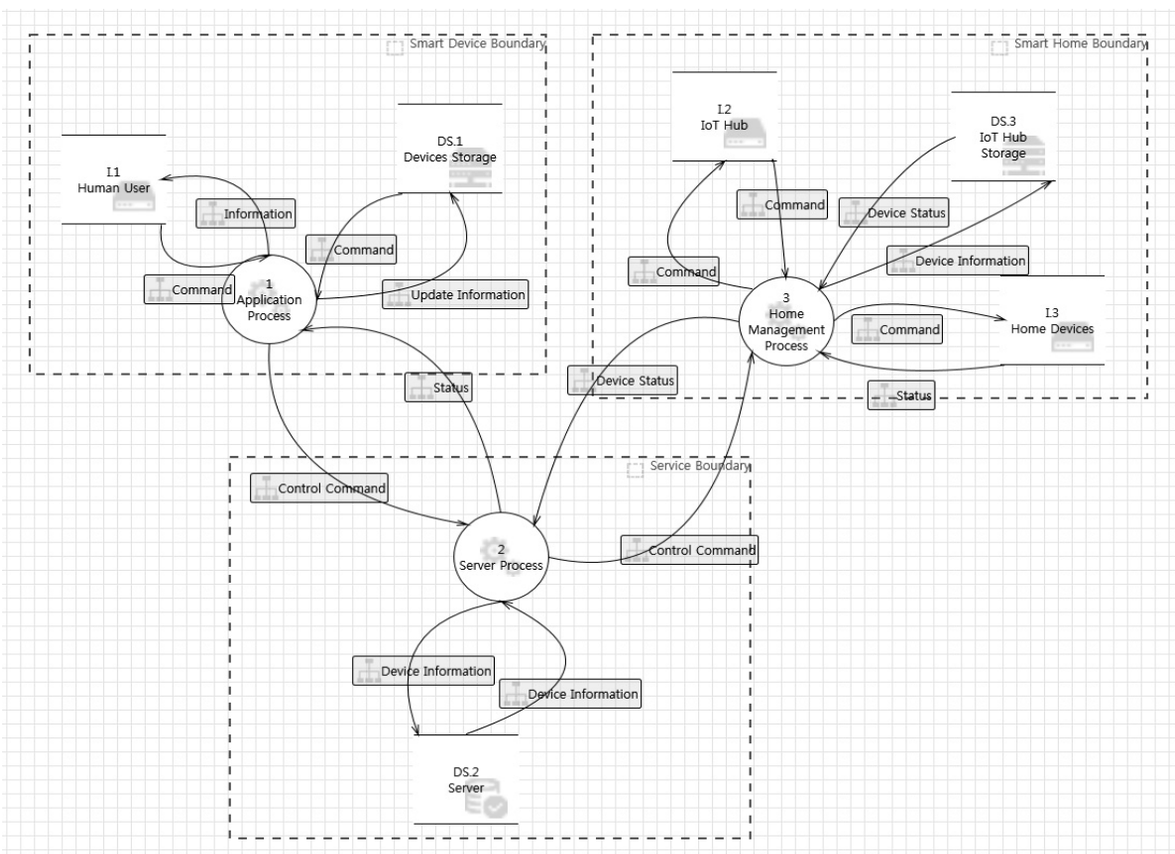


Fig. 3. Level 1 DFD for Smart Home Appliance

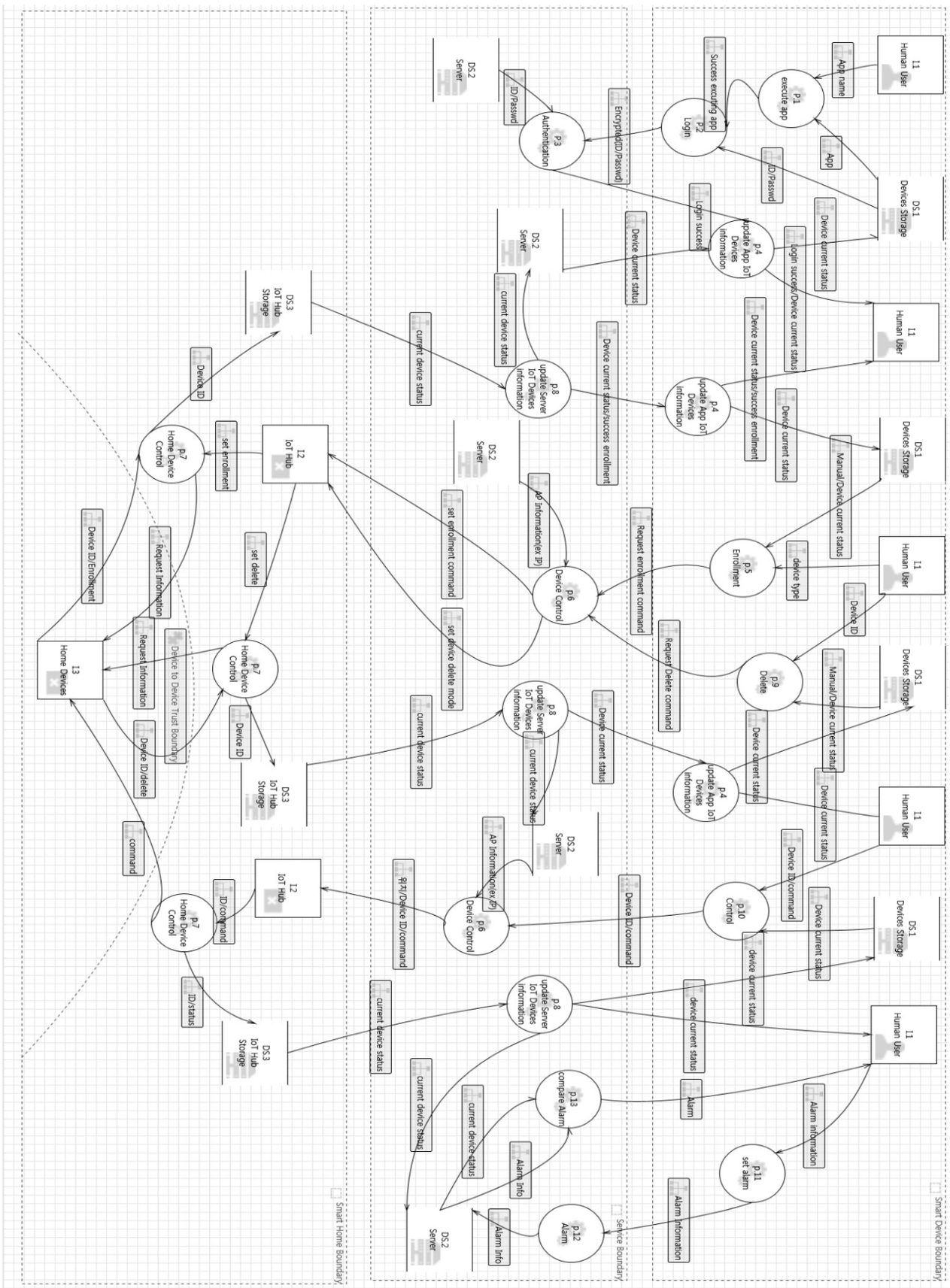


Fig. 4. Level 2 DFD for Smart Home Appliance

1) DFD 구조 및 용어 설명

a) Human User (I1)

테블릿, 스마트 폰과 같은 스마트 기기를 사용하는 사용자를 말한다. 최초의 로그인하는 경우 사용자의 ID와 Password를 입력하며, 제어하고자 하는 가전기기를 스마트 기기를 이용하여 애플리케이션의 동작을 지정 한다.

b) Device Storage (DS1)

테블릿, 스마트 폰과 같은 스마트 기기의 저장장소를 나타낸다. 스마트 홈을 제어하는 애플리케이션이 설치되어 있는 공간이다. 최초 로그인 이후에 ID와 Password가 기기공간에 저장되어 있으며, 애플리케이션에서 설정한 정보들이 저장되어 있다.

c) Server (DS2)

서비스를 제공하는 측면의 서버로써, 이용자들의 개인 정보와 이용자가 등록한 가전 기기들의 정보가 등록되어 있다. 스마트 홈 가전기기들을 제어할 수 있는 기기 등록, 기기 삭제, 기기 제어, 기기 상황에 따른 알람과 같은 기능이 있는 서비스를 제공한다. 제공하는 서비스 플랫폼에 따라 서비스가 일부 차이가 있으나 공통적으로 기기와 관련된 서비스를 제공한다.

d) IoT Hub (I2)

Server로부터 넘어온 데이터를 받아 스마트 홈 가전기기와의 통신에 사용되기 위해 Z-wave 프로토콜로 데이터를 변환하여 전송해 주는 기기이다. 기기는 각각 명령어에 맞게 데이터를 변환해주는 기능을 하며, 주기적으로 스마트 홈 가전기기와 통신을 한다.

e) IoT Hub Storage (DS3)

IoT Hub의 내부 저장장치로서, Web Application을 제공한다. 저장장치 내부에 스마트 홈기기의 등록정보를 저장한다. 기기의 등록하거나 삭제할 수 있으며 기기의 현재 상태를 주기적으로 관찰 한다.

f) Home Devices (I3)

스마트 홈 가전 기기를 말한다. 도어락, 가스락, 열림감지 센서, 스위치, 로봇청소기와 같이 다양한 기기들이 해당된다. 본 DFD에서는 각 서비스 제공회사에서 중심으로 제공하는 서비스인 도어락, 가스락, 열림 감지 센서, 플러그가 이에 해당한다.

2) 스마트 홈 데이터 흐름

a) Smart Device boundary to Service boundary

스마트 기기와 서버 간 데이터가 이동하는 구간이다. 스마트 기기의 애플리케이션 프로세스들이 서버로 데이터를 전송한다. 로그인, 기기제어, 기기등록, 기기삭제, 기기알람 설정, 가전 기기정보에 관련된 데이터가 이동한다.

b) Service boundary to Smart Home boundary

스마트 기기를 통해 사용자가 서버로부터 어떠한 명령을 실행한 경우에 서버와 스마트 홈 기기 간 데이터가 이동하는 구간이다. 기기제어, 기기등록, 기기삭제와 관련된 명령과, 현재 기기정보에 관한 데이터가 이동한다.

c) Smart Home boundary to Device to Device boundary

IoT Hub와 가전 기기 간 데이터가 이동하는 구간이다. 서버로부터 명령어를 받으면 IoT Hub가 이를 Z-Wave 프로토콜로 변환하여 각 스마트 기기로 전송한다.

3.2 STRIDE 분석

STRIDE는 Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, Elevation of privilege에서 영단어의 머리글자를 결합한 용어로 각각은 보안의 3요소인 기밀성, 무결성, 가용성에 인증, 부인 방지, 권한 부여의 3가지 요소를 추가하여 총 6가지의 목표에 대응하는 위협을 분류한 것이다. STRIDE는 Microsoft에서 분석대상의 위협을 식별하기 위한 방법으로 제안한 것으로서 현재까지 사용되고 있다. STRIDE를 위협 모델링에 활용하는 경우 체계적으로 분석대상에 대한 위협을 식별할 수 있다. 일반적으로 위협을 식별하는 경우 분석하는 사람의 역량에 따라 도출되는 결과가 다르게 나타난다. 따라서 이를 최소화 하고 분석대상의 전 범위를 빠짐없이 분석하기 위해 STRIDE의 6가지 요소를 사용하여 위협을 식별한다.

Table 3. Available DFD elements per STRIDE

	S	T	R	I	D	E
External Entity	x		x		x	
Data Store		x	?	x	x	
Process	x	x	x	x	x	x
Data Flow		x		x	x	

앞서 도출한 DFD의 요소들을 분리하고, 각각의 요소에서 발생 가능한 위협을 STRIDE 항목에 따라 작성한다. DFD 요소에 따라 STRIDE 에 적용 가능한 항목이 다르며 Table 3은 각 DFD 요소에 따른 적용 가능한 STRIDE 항목을 보여준다. 각 DFD 요소에 따라 해당되는 STRIDE 요소를 'X'로 표시해 두었다. 예를 들어 External Entity와 같은 경우에는 Spoofing, Repudiation, Denial of Service의 위협요소가 가능한 경우이다. DFD 요소에 대한 STRIDE 적용 항목은 늘어날 수 없으나. STRIDE의 실현 가능성이 불가능한 경우 제외될 수 있다. 예외로 Data Store의 경우 실현 불가능성과 다르게 로그가 남는 경우에만 Repudiation이 가능하기 때문에 '?'로 표시한다. 작성된 DFD에 따라 요소별 STRIDE로 위협을 분리한 결과는 Table 4와 같다. Table 4는 각각의 Trust Boundary 안에 존재하는 DFD 요소별 STRIDE 분석과 Trust Boundary간 데이터가 이동하는 Data Flow에 대한 STRIDE 분석이다. 전체 58개의 위협이 식별되었으며

Table 4. STRIDE

Trust Boundary	Element of DFD	Type	Detail
Smart Device	Human User(I1)	S	T1 Attacker can spoof authorized user and try to login
		R	T2 Attacker login using authorized ID and he denies this later
		D	T3 Attacker do failed login using specified authorized user's ID for denial of service
	Device Storage(DS1)	T	T4 Attacker falsify application in smart device's storage and he tamper api
		R	T5 Attacker tamper api but api does not check anything so, he can deny he tamper it
		I	T6 Attacker get api T7 Attacker get ID and password from device storage T8 Attacker get information from memory or storage T9 Attacker get Encryption keys T10 Attacker get sensitive information
	execute app(P1) login(P2) Update App IoT Devices Information(P4) Enrollment(P5) Delete(P9) Control(P10) Set Alarm(P11)	S	T11 Tampered api spoof normal App(or process)
		T	T12 Attacker tampere api
		R	T13 Attacker tamper api but api does not check anything
		I	T14 Attacker tamper api and then it can be Information disclosure
		D	T15 Attacker tamper api and then install normal user's Smart Device. So normal user can't use normal service
		E	T16 Attacker tamper api and he can get privilege(ex. door open command)
Service	Server(DS2)	S	T17 Attacker spoof Server
		D	T18 Attacker attacks server (ex. using Ping of death, Flooding attack)
	Authentication(P3) Device Control(P6) Update Server IoT Devices Information(P8) Alarm(P12) Compare Alarm(P13)	T	T19 Attacker do SQL injection and then to tamper data
		I	T20 Attacker do SQL injection and then It can be Information of disclosure
		D	T21 Attacker consume network resources and then it can be Denial of service
Smart Home	IoT Hub(I2)	S	T22 IoT Hub can be spoofed
		D	T23 Attacker consume network resources and then it can be denial of service T24 Attacker delete information of device enrollment
	IoT Hub Storage(DS3)	T	T25 Attacker can Tamper Hardware memory T26 Attacker modify system using memory overwrite T27 Attacker get firmware
		I	T28 Attacker get IoT device's sensitive information(ex. number, control information, and session) T29 Attacker get firmware using UART and JTAG port T30 Attacker get Web Interface ID and password
	Home Device Control(P7)	T	T31 Attacker tamper api using weakness api T32 Attacker tamper firmware
		R	T33 Attacker tamper firmware and it run on the device but anyone didn't know it
		I	T34 Attacker get information of using service(ex. ssh, telnet) T35 Attacker get information of running service T36 Attacker get Web Interface ID and password
		D	T37 Attacker login web interface and change device information and then it can be denial of service
		E	T38 Attacker get permission or access control using sensitive information, administrator's file
Device to Device	Home Devices(I3)	T	T39 Home Device can be spoofed for device ID
		I	T40 Attacker can get device ID
		D	T41 Attacker send a lot of packet to Home Device and then it can be Denial of Service (ex. ping of death, flooding attack)
Smart Device <-> Service	All Data Flow between (Smart Device <-> Service)	T	T42 Attacker can do session hijacking and tamper data T43 Attacker get data between communication and then tamper data(ex. sniffing, spoofing)
		I	T44 Attacker can take data(ex. Device enrollment, delete, control, status, Alarm) T45 Attacker can get running service information(ex. port scan) T46 Attacker tamper data then it can be Information disclosure
		D	T47 Attacker can consume network resources(ex. ping of death, flooding) T48 Attacker tamper data then it can be denial of service
Service <-> Smart Home	All Data Flow between (Service <-> Smart Home)	T	T49 Data can be captured (sniffing, Spoofing), and then it can be tampered.
		I	T50 Attacker can take data(Device enrollment, delete, control, status) T51 Attacker can get network resource information
		D	T52 Attacker tamper data then it can be denial of service T53 Attacker consume network resources(ex. flooding, ping of death) T54 Attacker can interrupt network communication
Smart Home <-> Device to Device	All Data Flow between (Smart Home <-> Device to Device)	T	T55 Data(Device enrollment, delete, control, status) can be tampered by attacker using RF signal such as Zigbbee and Z-wave
		I	T56 Attacker can take data(Device enrollment, delete, control, status) between communication using RF signal such as Zigbee and Z-wave T57 Attacker can get network resource information(sniffing, capture)
		D	T58 Attacker can interrupt network communication(flooding)

STRIDE 요소별 실현 가능한 위협을 기술했다. STRIDE로 도출된 위협사항에 대해 구별할 수 있도록 기호와 숫자를 부여하여 Tno.의 형태로 표기하였다. 각각의 도출된 위협들은 Attack Tree와 체크리스트에 있어서 해당하는 부분을 기술하여, 본 논문에서 제안하는 체크리스트가 전 범위를 다루었다는 것을 사실을 보인다.

3.3 Attack Library

Attack Library는 위협 모델링에 있어서 시스템에 대한 위협을 상세화 하는 도구이다. Attack Library를 구축함에 있어서 목적과 사용방안에 적합한 자료를 최대한 수집해야 하며, 표준, 관련 기술문서, 논문 및 컨퍼런스와 같은 기존연구, CVE(Common Vulnerabilities and Exposure)에서 공개된 취약점 정보, CWE(Common Weakness Enumeration), OWASP와 같은 보안 연구 프로젝트에서 공개한 취약점 항목들이 이에 해당한다. 본 논문에서 Attack Library는 위협을 식별하기 위한 목적으로 수집한 Library는 Table 5이다. 각 해당하는 부분에 대한 위협, 실제 공격 여부에 초점을 맞춰 수집하였으며 Conference, Journal, Project, CVE로 크게 4가지로 분류된다. Project항목에 존재하는 OWASP나 CWE와 같은 경우는 프로젝트 명으로 기술하여 작성했다. Table 5의 Type 항목에는 수집한 항목이 적용 가능한 부분을 Application, Hardware, System, Web Interface, Network로 작성하였으며 전 범위에 적용이 가능한 것은 All로 표기했다.

3.4 Attack Tree 도출

도출한 DFD를 바탕으로 Attack Tree를 작성한 결과는 Fig. 5이다. 루트 노드를 분석대상인 스마트 홈 서비스에 대

Table 5. Attack Library

Category	Year	Title	Autor	Type	Ref
Conference	2016	Breaking Bulbs Briskly by Bogus Broadcasts	Joseph Hall	Network	[7]
	2015	Home Network Hacking	mungmung	Network	[10]
	2015	Zigbee Exploited the Good The Bad And The Ugly	Tobias Zillner	Network	[6]
	2015	Using static binary analysis to find vulnerabilities and backdoors in Firmware	Shellphish	System	[19]
	2015	Ah! Universal Android Rooting is Back!	Wen Xu	Application	[20]
	2014	Smart Nest Thermostat A Smart Spy in Your Home	Grant Hernandez	Hardware	[9]
	2014	Security Analysis of IEEE 802.15.4 MAC in the context of Internet of Things (IoT)	CIACS	Network	[21]
	2013	Bluetooth Smart: The Good, The Bad, The Ugly... and The Fix	Mike Ryan	Network	[22]
	2013	Honey, I'm Home!! (Hacking Z-Wave Home Automation Systems)	Sahand Ghanoun	Network	[5]
	2013	Security evaluation of the Z-Wave wireless protocol	B Fouladi	Network	[23]
	2013	Home Invasion V2.0 - Attacking Network-Controlled Hardware	D. Crowley	Network	[8]
	2012	SQL Injection to MIPS overflows: Rooting SOHO Routers	Zachary Cutlip	System	[24]
	2010	KillerBee: Practical ZigBee Exploitation Framework	Joshua Wright	Network	[25]
	2009	A 16 bit Rootkit and Second Generation Zigbee Chips	Travis Goodspeed	Network	[26]
	2009	Router Exploitation	LINDNER	System	[27]
	2007	Hacking the Extensible Firmware Interface	John Heasman	System	[28]
	2006	Exploiting Embedded Systems	Bamaby Jack	System	[29]
	2006	Vulnerabilities in Not-So Embedded Systems	Brendan O'Connor	Hardware/System	[30]
Journal	2016	Security Analysis of Emerging Smart Home Applications	Earlence Fernandes	Hardware/System	[13]
	2015	Rogue Z-Wave Controllers: A Persistent Attack Channel	Jonathan D. Fuller	Network	[12]
	2006	Forensic Imaging of embedded systems using JTAG(boundary-scan)	Ing.M.F.Breeumsma	Hardware	[31]
Project	2015	OWASP Internet of Things(IoT) Project_Firmware Analysis	OWASP	System/Web	[15]
	2015	OWASP Internet of Things(IoT) Project_IoT Attack Surface	OWASP	All	[16]
	2015	CAPEC CATEGORY: Software	MITRE	Application/Web	[32]
	2015	CAPEC CATEGORY: Hardware	MITRE	Hardware	[33]
	2013	OWASP Top Ten Cheat Sheet	OWASP	All	[34]
	2011	CWE/SANS TOP 25 Modst Dangerous Software Errors	SANS	Application/Web	[35]
CVE	2015	CVE-2015-4080	MITRE	Network	[36]

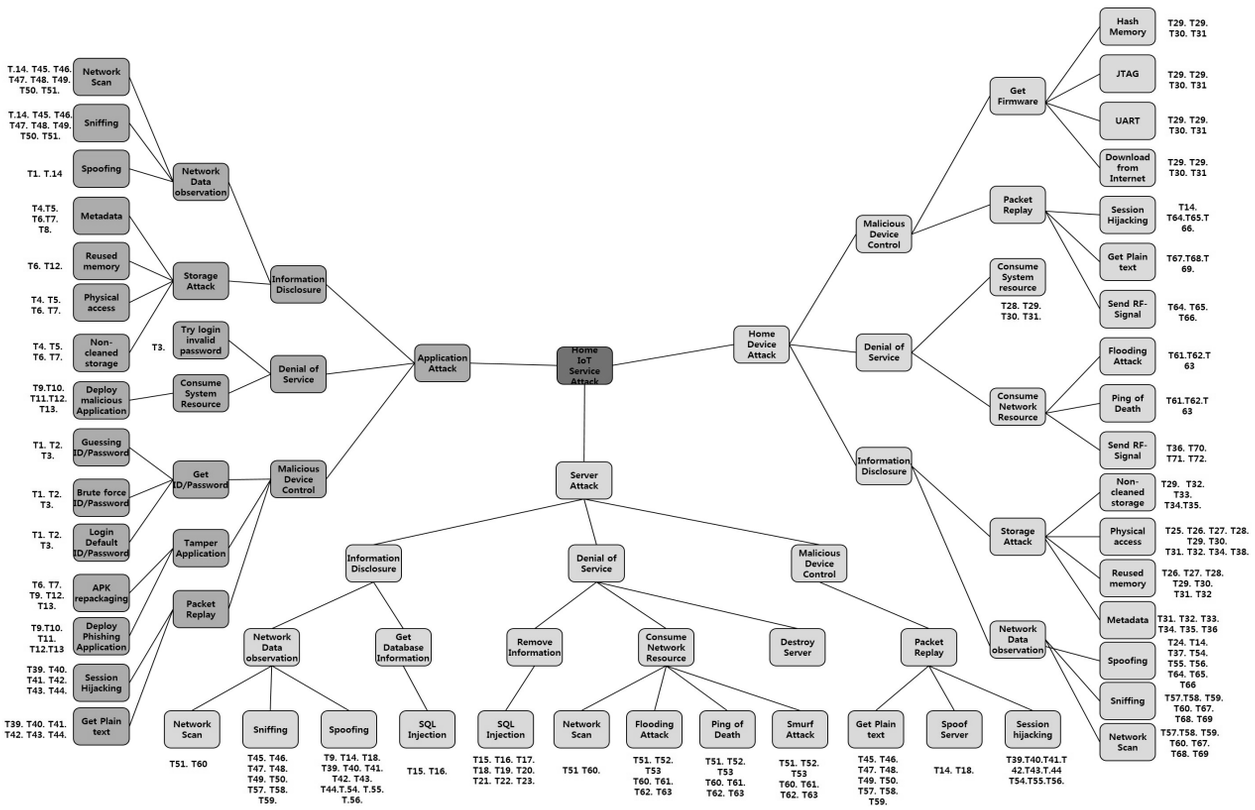


Fig. 5. Attack Tree

한 공격으로 설정하였다. 루트 노드에 대한 공격을 수행하기 위해서는 전체 서비스의 구성요소인 Smart Device, Server, Home Device로 하위 노드를 구성하였다. STRIDE에서의 Trust Boundary와 마찬가지로 3가지로 분류하였다. Attack Tree의 최하위 노드 옆에 기존에 STRIDE로 도출한 위협사항들과 연관이 있는 부분들은 작성해 두었으며 이러한 위협들을 통해 공격의 목표를 도출하고, 도출된 결과를 통해 스마트 홈에 대한 위험을 분석한다. Smart Device와 Server 사이의 통신과 Server와 IoT Hub간의 통신은 중복되어 트리에 표기 될 수 있기 때문에 이를 방지하고자 Server 쪽에만 작성하였다.

3.5 체크리스트 도출

Table 6은 DFD, STRIDE, Attack Library, Attack Tree를 바탕으로 작성한 체크리스트이다. Bounds에는 Attack Tree에서 도출한 공격 목표인 Smart Device, Smart Home, Service로 하였으며 각 목표가 적용되는 Category에 맞게 기술하였다. Network Category의 경우 Smart Device, Smart Home, Service 모두 해당되기 때문에 전부 작성하였다. 각 Category에 따라 Surface를 분류하여 작성하였으며, 적용 불가능하거나 추상적인 공격, 사회공학방법이 적용된 위협들은 제외하고 총 54개의 체크리스트를 도출했다. 도출된 점검항목들은 Attack Library를 사용하였기 때문에 현존하는 취약점들과 공격에 대해 점검할 수 있으며, 위협 모델링을 통해 분석대상의 전 범위를 체계적으로 분석하여 도출하였기 때문에 스마트 홈 전체의 범위를 점검할 수 있다. 또한 STRIDE로 도출된 위협사항들을 모두 포함하고 있음을 보여주기 위하여 해당하는 부분을 대응하여 점검항목 옆 부분에 위협 항목을 작성해 두었다.

4. 체크리스트를 이용한 스마트 홈 분석

본 논문에서 제안한 체크리스트의 실효성을 확인하기 위해 실제 국내에 제공되고 있는 C 통신사의 스마트 홈 서비스에 체크리스트 항목을 활용하여 분석을 수행하였다. 분석 환경은 Fig. 6과 같다. 실제 기기를 구매하여 분석환경을 구축하였으며, 가정환경으로 꾸며진 부분은 실선 네모로 표시된 부분이다. 점선 네모로 표시된 부분은 분석 범위를 나타낸다. Smart Devices는 기기를 원격으로 제어하는 테블릿, 스마트 폰과 같은 기기이며, Home Devices는 실제 네트워크 기능이 있는 가전기기를 말한다. Gateway의 경우 공유기를 나타내며, IoT Hub는 공유기와 Home Devices 중간에 위치하여 통신하는 장치이다. Smart Devices에 연결된 분석 장비의 경우 Application과 Smart Device에서 Server로 나가는 데이터를 분석하며 Gateway에 연결된 분석 장비는 Server와 가정 내부와의 데이터를 분석한다. IoT Hub와 연결된 분석 장비의 경우 Hub의 Hardware, System, Web Interface를 분석하며 Home Device와 IoT Hub간의 통신을 분석한다.

Table 6. Check List for Home IoT

Bound	Category	Surface	Detail	no	T no.
Smart Device	Application	User Authentication	Acquisition of user account information	C1	T1,2
			The number of trying to login	C2	T1,2
			Safe mechanism for making password	C3	T1
			A period of changing password	C4	T1
			Usage of the default password commonly used by people	C5	T1
			Usage of Two-factor authentication	C6	T1
		Denial of Service to specific user	Denial of service attack to specific user	C7	T3
		Session	Session management like server time-out	C8	T42,43
		Unsafely data stored at local storage	Acquisition of data from local data storage	C9	T4,7,9,10
			Acquisition of data form memory	C10	T8
		Tamper detection about Application	Existence of apk integrity detection	C11	T6
			Existence of APK obfuscation and obfuscation technology	C12	T5,T11,12,13
		Data encryption key	Acquisition of encryption key using apk decompile	C13	T9
		Unnecessary permission of application	Verification of application request unnecessary permission to work it	C14	T14,15,16
		SSL/TLS	Usage of SSL/TLS version	C15	T44,45,46
Home Device	Hardware	Firmware acquisition	Acquisition firmware from webpage	C16	T27,
			Acquisition firmware during update	C17	T27,
			Acquisition firmware through bootloader	C18	T27,
		Tampered Firmware	Firmware verification	C19	T33
		Debug interface	Physically & logically elimination of UART port	C20	T29
			Physically & logically elimination of JTAG port	C21	T29
			Access shell through UART/JTAG	C22	T29
		Information about IoT Devices	Acquisition of IoT devices sensitive information such as number, control information and session	C23	T24,28,
		Tampering memory	Flash Memory data encryption	C24	T25,
			Modification of system using memory overwrite	C25	T25,26
	System	Shell Interface	Usage of SSh, Telnet service is given	C26	T34,35
			Acquisition of system information through system shell	C27	T34
		Weakness API of binary and wrong usage of API	Usage of weakness API (Ex. Strcpy, strcat, system)	C28	T31,32
			Wrong usage of API (Ex. out-of bound, use-after-free)	C29	T31,32
		Access control and permission	Acquisition of administrator's file	C30	T28,38
			Acquisition of file which store sensitive information	C31	T30
			Verification of file & directory permission management and access control management	C32	T28
		Running service	Usage of unnecessary service running	C33	T34
			Acquisition of sensitive information through running service	C34	T34,35
			1-day vulnerability on each service version	C35	-
	Web Interface	User Authentication	Usage of the default ID and Password given by the manufacturer	C36	T30,T36
			Admin page exposure	C37	T30,T36,T37
			Acquisition of device information through vulnerability search engine	C38	T39,40
			The number of trying to login	C39	T30
			Verification of input data	C40	T30
			Usage of safe mechanism for making password	C41	T30
			A period of changing password	C42	T30
			Usage of safe password using smaller and capital letter, number, length of password and special letters	C43	T38

Bound	Category	Surface	Detail	no	T no.
Server · SmartDevice · Home Device Communication	Network	Port Scan	Verification of open port	C44	T45,47
			Usage of unnecessary port	C45	T45,47
		Packet Dump	Encryption of sensitive information	C46	T44,49,50,51,57
			Acquisition of mobile data and server data	C47	T42,4344,45,46
			Acquisition of firmware through packet sniffing	C48	T27
		Packet Modulation	Replay attack availability(Ex. Time stamp, Time to Live)	C49	T43,49,55
			Man-in-the middle attacks to ensure availability of data manipulation	C50	T17,22,42,43,44,54
		Processing for random data	Verification of abnormal payload	C51	T46,52
			Excessive communication breakdown in transmission error handling check(Ex. Buffer over flow, DoS)	C52	T18,19,20,21,23,47, 48,58
		RF Signal	Acquisition of information through RF Signal	C53	T23,39,41,55
			Usage of RF Signal Encryption	C54	T22,40,56

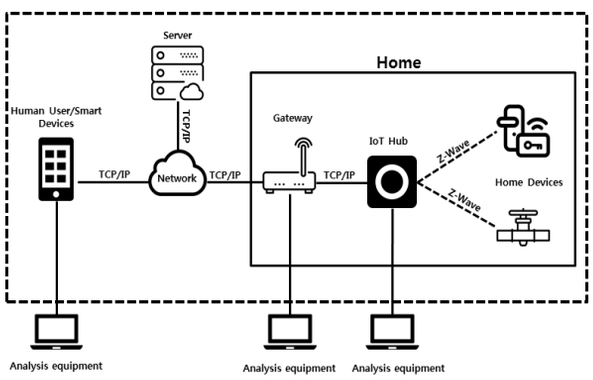


Fig. 6. Smart Home Analysis Environment

전체 54개의 점검항목을 수행하였으며 Application, Hardware, System, Web Interface, Network의 Category에 따라 점검 한 결과는 다음과 같다. 각 점검항목을 수행한 결과 안전한 부분은 'O', 취약한 부분이 존재하는 경우 'X'로 표기하였다.

4.1 Application

기기를 원격으로 제어하는 명령을 내리는 스마트 기기에 대한 점검항목은 15개(C1~C15)로 도출했다. Table 7은 15개의 점검 결과로 C1, C2, C7, C11, C15의 5가지 점검 항목에 대해 취약한 부분이 발견되었다.

1) 사용자 계정정보 유출 가능 여부

점검항목 C1에 해당하는 결과로 사용자 계정정보의 유출이 될 수 있다는 취약점이 존재한다. 기기를 제어하기 위한 스마트 기기의 애플리케이션을 이용하기 위해서는 로그인이 필요하다. 로그인을 시행하는 경우 존재하지 않는 ID인 경우에는 '로그인 실패' 메시지가 출력되며, 존재하는 ID로 로그인에 실패하는 경우에는 Captcha를 추가로 입력하는 페이지로 이동한다. 이를 통해 특정한 사용자에 대한 계정 ID를 유추할 수 있다. Fig. 7은 실제 출력되는 두 가지 경우의 메시지를 보여준다.

Table 7. Application Check List

Category	Detail	no	result
Application	Acquisition of user account information	C1	X
	The number of trying to login	C2	X
	Safe mechanism for making password	C3	O
	A period of changing password	C4	X
	Usage of the default password commonly used by people	C5	O
	Usage of Two-factor authentication	C6	O
	Denial of service attack to specific user	C7	X
	Session management like server time-out	C8	X
	Acquisition of data from local data storage	C9	O
	Acquisition of data form memory	C10	O
	Existence of apk integrity detection	C11	X
	Existence of APK obfuscation and obfuscation technology	C12	O
	Acquisition of encryption key using apk decompile	C13	O
	Verification of application request unnecessary permission to work it	C14	O
	Usage of SSL/TLS version	C15	X

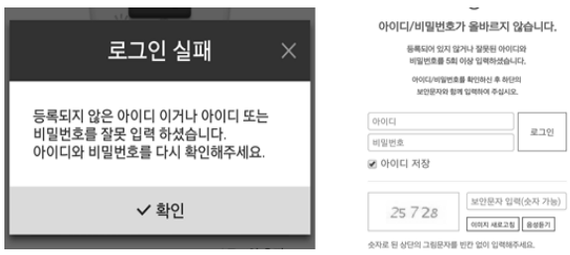


Fig. 7. Login Using No Existence ID(left) and Existence ID(right)

2) 로그인 시도 횟수 제한 & 사용자 서비스 거부 공격

점검항목 C1, C2, C7에 해당하는 부분에 취약점이 존재한다. 존재하는 ID와 존재하지 않는 ID에 따라 로그인 실패 메시지가 다르기 때문에 존재하는 ID를 유추 할 수 있다. 존재하는 ID로 로그인을 지속적으로 수행하였을 경우에 10회 이상 비밀번호를 잘못 입력하는 경우에 로그인이 제한된다. Fig. 8은 실제 10회 이상 비밀번호를 잘못 입력하였을 경우에 출력되는 화면이다.

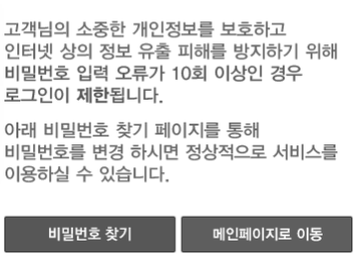


Fig. 8. The Number of Trying to Login

따라서 특정한 계정을 사용하는 사용자의 ID로 지속적인 로그인을 수행하는 경우에는 서비스를 이용할 수 없게 계정이 잠기며, 새로운 비밀번호를 재설정해야하기 때문에 특정인을 대상으로 서비스 거부 공격이 가능하다.

3) 비밀번호 변경 주기

점검항목 C4에 해당하는 결과로 비밀번호 변경주기가 존재하지 않는 취약점이 존재한다. 비밀번호 변경주기가 존재하지 않는 경우, 다른 서비스로부터 유출된 계정을 통해 도용의 문제가 발생할 수 있으므로 비밀번호의 변경 주기 정책이 필요하다. 분석대상의 계정에 대한 비밀번호 변경 주기를 확인하기 위해 서비스의 계약해지 및 이용제한 약관을 보면 확인할 수 있다. 분석대상의 약관의 경우 비밀번호 변경 주기에 대한 언급은 존재하지 않다. 따라서 이에 대한 변경 주기 정책의 추가가 필요하다.

4) 세션관리

점검항목 C8에 대한 결과로 서버와 클라이언트 간 통신 세션에 대한 취약점이 존재한다. 분석결과 로그인 이후 30분 이상 조작 없이 대기한 결과 세션이 종료되지 않는 것을 확인할 수 있다. 세션 타임아웃 기능이 구현되어 있지 않는 경우 장시간 부재중인 사용자에 대한 보호할 수 없으므로 타임아웃 기간을 설정해야 한다.

5) App 위변조에 대한 무결성 검증

점검항목 C11에 해당하는 결과로 App의 위변조에 대한 무결성 검증이 없다는 취약점이 존재한다. 이러한 취약점이 존재하는 경우 애플리케이션 변조를 통해 유해한 기능을 방지하는 기능을 무력화 시킨 후 2차적인 피해를 입힐 수 있다. 따라서 애플리케이션에 대한 무결성 검증이 되어 있는지 확인해야 한다. apktool을 이용하여 apk를 smali code로 decompile하고 편집기를 이용하여 /res/values-ko/strings.xml에 가장 처음 나오는 문자열을 수정한다. 변경된 파일들을 이용하여 repacking하고 jarsigner를 이용하여 signing한 apk 파일을 install 한다.

Fig. 9. strings.xml Modification

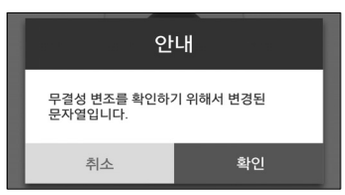


Fig. 10. Verification of Modified String

Fig. 10과 같이 변조된 문자열이 출력되며 애플리케이션의 무결성 훼손을 감지하는 방법이 존재하지 않음을 알 수 있다.

6) SSL/TLS 프로토콜 지원 여부 및 버전 확인

점검항목 C15에 해당하는 결과로 SSL/TLS의 사용여부에 따른 버전에 따른 취약한 부분이 존재한다. 애플리케이션과 서버와의 안전한 통신을 위해 SSL/TLS 암호 프로토콜을 사용하여 데이터를 암호화 하는지 확인하였다. Fig. 11은 애플리케이션과 서버간의 통신 패킷으로서 TLS 1.0으로

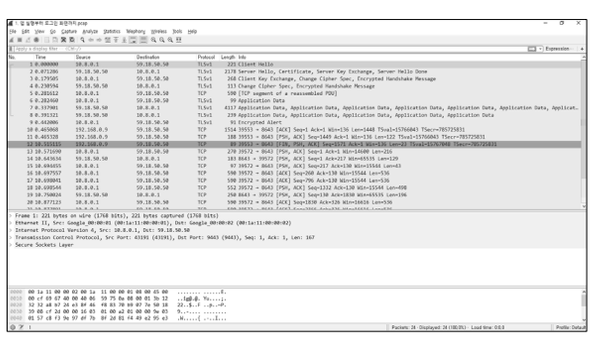


Fig. 11. Captured Packet

암호화 되어 전송되는 것을 확인할 수 있다. 그러나 TLS 1.0은 POODLE 취약점[37]이 발생할 가능성이 있기 때문에 TLS 1.2로 버전을 올려서 사용해야 한다.

4.2 Hardware & System

Hardware와 System부분은 스마트 홈 서비스에서 제공하는 Home Device와 Hub에 대한 분석으로 점검항목은 Hardware 10개(C16~25), System 10개(C26~C35)로 도출했다. Table 8은 Hardware와 System의 분석결과로 Hardware의 경우 C20, C22, C23의 3가지 항목에 해당하는 취약한 부분이 존재하였으며, System의 경우 취약한 부분이 존재하지 않았다.

Table 8. Hardware & System Check List

Category	Detail	no	result
Hardware	Acquisition firmware from webpage	C16	O
	Acquisition firmware during update	C17	O
	Acquisition firmware through bootloader	C18	O
	Firmware verification	C19	O
	Physically & logically elimination of UART port	C20	X
	Physically & logically elimination of JTAG port	C21	O
	Access shell through UART/JTAG	C22	X
	Acquisition of IoT devices sensitive information such as number, control information and session	C23	X
	Flash Memory data encryption	C24	O
	Modification of system using memory overwrite	C25	O
System	Usage of SSh, Telnet service is given	C26	O
	Acquisition of system information through system shell	C27	O
	Usage of weakness API (Ex. Strcpy, strcat, system)	C28	O
	Wrong usage of API (Ex. out-of bound, use-after-free)	C29	O
	Acquisition of administrator's file	C30	O
	Acquisition of file which store sensitive information	C31	O
	Verification of file & directory permission management and access control management	C32	O
	Usage of unnecessary service running	C33	O
	Acquisition of sensitive information through running service	C34	O
	1-day vulnerability on each service version	C35	O

1) UART 포트 존재 여부 및 포트를 통한 접속

점검항목 C20, C22, C23에 해당하는 결과로 UART 포트가 물리적으로 존재하였으며 이에 따라 UART로 Hub 시스템에 접근할 수 있다는 취약점이 존재한다. 스마트 홈의 IoT Hub의 UART 포트를 통해 시스템에 접근하는 경우 민감한 정보를 획득할 수 있다. 분석대상의 IoT Hub의 경우 UART 포트가 존재하며, 올바르게 접속하였을 경우에 부팅 메시지를 출력한다. Fig. 12는 출력되는 부팅 메시지 화면이다.

```
FileSystem Check Enable...
GPIO Escape Disable...

---RealTek(RTL8196D)at 2015.06.19-13:16+0900
v0.6 [16bit](659MHz)
###__pkg active side = [1]__###
FS1 (K[0x00011000], R[0x00411000]) result[2-1]
start_kernel] ret:1 kernelAddr:0x05011000
gCHKKEY_HIT:0 <--
Jump to image start=0x80a00000...
```

Fig. 12. Booting Message through UART Port

시스템에 접속한 후에 애플리케이션에서 가전 기기에 대한 명령을 전송하는 경우 디버깅 메시지를 출력하는 것을 확인할 수 있다.

Fig. 13. 'Door Lock Open' Control Message

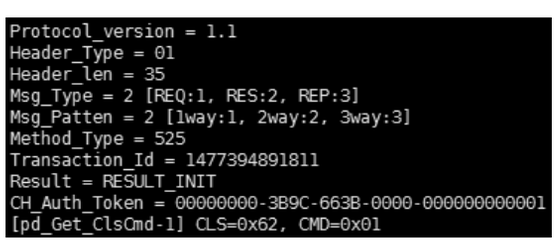


Fig. 14. Response Message

따라서 제품을 양산하는 경우에 디버깅 메시지에 대해 주석처리를 하거나 해당 코드를 제거하지 않는 경우, UART 포트를 양산 시 제거하지 않는 경우에는 UART 포트를 통한 취약점 분석을 시도 할 수 있다. Fig. 13과 Fig. 14는 출력되는 디버깅 메시지의 결과로 프로토콜 버전, 헤더 유형 및 길이, 메시지 타입(요청, 응답, 재전송), 메시지 패턴, 제어 명령, 인증 토큰

정보, IoT 기기 고유번호, 제어 정보와 같은 민감 정보를 확인할 수 있다.

4.3 Network & Web Interface

전체 시스템의 Trust boundary간 이동하는 데이터의 흐름인 Network와 Home Device의 Web Interface에 대한 점검항목으로 Network 11개(C43~C54), Web Interface(C36~C43)로 도출했다. Table 9는 Network와 Web Interface의 분석결과로 Network의 경우 C44, C45의 2가지 점검항목이 취약하였으며, Web Interface의 경우 C37, C39의 2가지 점검항목이 취약하였다.

Table 9. Network & Web Interface Check List

Category	Detail	no	result
Web Interface	Usage of the default ID and Password given by the manufacturer	C36	O
	Admin page exposure	C37	X
	Acquisition of device information through vulnerability search engine	C38	O
	The number of trying to login	C39	X
	Verification of input data	C40	O
	Usage of safe mechanism for making password	C41	O
	A period of changing password	C42	O
	Usage of safe password using smaller and capital letter, number, length of password and special letters	C43	O
Network	Verification of open port	C44	X
	Usage of unnecessary port	C45	X
	Encryption of sensitive information	C46	O
	Acquisition of mobile data and server data	C47	O
	Acquisition of firmware through packet sniffing	C48	O
	Replay attack availability(Ex. Time stamp, Time to Live)	C49	O
	Man-in-the middle attacks to ensure availability of data manipulation	C50	O
	Verification of abnormal payload	C51	O
	Excessive communication breakdown in transmission error handling check(Ex. Buffer over flow, DoS)	C52	O
	Acquisition of information through RF Signal	C53	O
	Usage of RF Signal Encryption	C54	O

1) Port Scan

점검항목 C44, C45에 해당하는 결과로 열려있는 포트가 존재하며 알려지지 않은 서비스가 있는 취약점이 존재하며 분석을 수행한 내용은 다음과 같다. 불필요한 서비스나, 알려지지 않은 서비스를 확인하기 먼저 Port scan을 수행한다. Fig. 15는 스마트 홈의 IoT Hub를 대상으로 Port scan을 수행한 결과이다. Port scan을 수행한 결과, 8877의 포트가 열려 있는 것을 확인할 수 있다. 8877의 경우 알려지지 않은 서비스이나 해당 포트로 접속해 본 경우 IoT Hub의 웹 관리자 페이지가 존재하는 것을 확인할 수 있다.

Fig. 15. Port Scan Result

2) 관리자 페이지 로그인

점검항목 C37, C39에 해당하는 결과로 Home Device들을 관리하는 관리자 페이지가 Hub에 존재하는 취약점과 로그인 시도 횟수에 대한 취약점이 존재하였다. 스마트 홈의 IoT Hub에 존재하는 관리자 페이지에 로그인을 하는 경우, Hub에 연결된 기기들의 정보 및 제어 명령과 같은 민감한 정보를 획득 할 수 있다. 따라서 관리자 페이지에 로그인 페이지는 로그인 시도 횟수 제한이 설정되어 있어야 한다. 웹 프록시 도구인 'Burp Suite'의 Intruder 기능을 사용하여 Brute Force 공격을 수행한 결과, 로그인 시도 횟수 제한이 존재하지 않는 것을 확인할 수 있다.

5. 결론 및 향후 연구 방향

IoT는 사람과 사람, 사람과 사물, 사물과 사물간의 정보를 상호 소통하는 기술 및 서비스이다. IoT의 서비스는 크게 개인 IoT, 산업 IoT, 공공 IoT로 분류할 수 있으며 스마트 홈 서비스는 개인 IoT로 분류된다. 스마트 홈은 가정에서 사용하는 기기를 원격으로 밖에서 제어할 수 있는 기술 및 서비스로서 사용자에게 편의성을 제공하며 국내외 서비스 시장은 증가하고 있는 추세이다. 시장이 증가함에 따라 공격자에 의한 악의적인 기기제어, 사생활 침해와 같은 보안 사고가 있는 경우 피해 규모가 클 것으로 예상된다.

따라서 본 논문에서는 위협모델링을 이용하여 체계적으로 스마트 홈을 분석할 수 있는 체크리스트를 제안하였으며 이를 활용하여 국내의 스마트 홈을 분석하였다. 체크리스트는 STRIDE를 이용하여 위협을 도출하였으며 도출과정에 있어서 Attack Library를 활용하여 위협을 상세하게 기술하였다. 도출된 STRIDE를 바탕으로 Attack Tree를 통해 공격 가능한 위험을 분석했으며, 도출된 모든 결과를 기반으로 스마트 홈 서비스에 대한 보안 점검을 할 수 있는 체크리스트를 도출하였다. 도출된 체크리스트는 Attack Library를 활용하

였기에 현존하는 공격에 대한 사항을 포함하고 있으며, 위협 모델링을 통해 체계적으로 도출하였기에 전체의 시스템을 대상으로 분석이 가능하다.

본 논문에서 제안하는 체크리스트를 활용하여 스마트 홈을 분석한 결과 취약점을 발견하였으며 이에 따라 제안하고자 하는 체크리스트가 실효성이 있다고 사료된다. 따라서 본 연구결과는 스마트 홈 및 유사 환경에 대한 보안성 분석을 위한 하나의 방안으로 활용될 수 있을 것으로 기대한다. 향후 연구로는 Non-IP기반에 클라우드 기반 서비스가 접목된 포괄된 환경에서의 위협모델링 연구가 남아있다.

References

[1] ITU, ITU Internet Reports 2005, Internet of Things(2005).

[2] ITU-T Y.2060, Overview of the Internet of Things(2012).

[3] 미래창조과학부, “사물인터넷기본계획,” 2014

[4] STRATEGY ANALYTICS, “About Smart Home,” [Internet], https://www.strategyanalytics.com/access-services/devices/connected-home/smart-home/about-smart-home#.WBmnDfmLRGo.

[5] Behrang Fouladi and Sahand Ghanoun, ‘Honey, I’m Home!!: Hacking Z-Wave Home Automation System’, Black Hat 2013, USA, 2013.

[6] Tobias Zillner, ‘Zigbee Exploited: The Good, the Bad, the Ugly,’ Black Hat USA 2015, USA, 2015.

[7] Joseph Hall, ‘Breaking Bulbs Briskly by Bogus Broadcasts,’ ShmooCon 2016, USA, 2016.

[8] Daniel Crowley, “Home Invasion V2.0 - Attacking Network-Controlled Hardware,” BlackHat USA, USA, 2013.

[9] Grant Hernandez, “Smart Nest Themostat A Smart Spy in Your Home,” Black Hat USA, USA, 2014.

[10] Mungmung, “Home Network Hacking,” SECUINSIDE, Korea, 2015.

[11] Thomas Reuter, “Security analysis of wireless communication standards for home automation,” Technische Universität München, 2013.

[12] Fuller, Jonathan D. and Benjamin W. Ramsey, “Rogue Z-Wave controllers: A persistent attack channel,” *Local Computer Networks Conference Workshops (LCN Workshops), 2015 IEEE 40th,* IEEE, 2015.

[13] E. Fernandes, J. Jung, and A. Prakash, “Security Analysis of Emerging Smart Home Applications,” in *Security and Privacy (SP), 2016 IEEE Symposium on. IEEE*, pp.636-654, 2016.

[14] OWASP, “OWASP Internet of Things(IoT) Project,” [Internet], https://www.owasp.org/index.php/OWASP_Internet_of_Things_Project.

[15] OWASP, “OWASP Internet of Things(IoT) Project_Firmware Analysis Project,” [Internet], https://www.owasp.org/index.php/OWASP_Internet_of_Things_Project#tab=Firmware_Analysis.

[16] OWASP, “OWASP Internet of Things(IoT) Project_IoT Attack Surface Areas Project,” [Internet], https://www.owasp.org/index.php/OWASP_Internet_of_Things_Project#tab=IoT_Attack_Surface_Areas.

[17] Noel Poore, “Internet of Things Security Architecture [BOF3029],” ORACLE OPENWOLRD 2014, San Francisco, 2014.

[18] Yuri Diogenes, “Internet of Things security architecture,” 2016 [Internet], https://azure.microsoft.com/en-us/documentation/articles/iot-security-architecture/.

[19] Shellphish, “Using static binary analysis to find vulnerabilities and backdoors in Firmware,” BalckHat USA, USA, 2015.

[20] Wen Xu, “Ah! Universal Android Rooting is Back!,” BlackHat USA, USA, 2015.

[21] SM Sajjad, “Security analysis of IEEE 802.15.4 MAC in the context of Internet of Things(IoT),” CIACS, 2014.

[22] Mike Ryan, “Bluethooth Smart: The Good, The Bad, The Ugly... and The fix,” BlackHat USA, USA, 2013.

[23] Fouladi, Behrang, and Sahand Ghanoun, “Security evaluation of the Z-Wave wireless protocol,” Black hat USA 24 (2013).

[24] Zachary Cutlip, “SQL Injection to MIPS overflows: Rooting SOHO Routers,” BlackHat USA, USA, 2012.

[25] John Mcnabb, “KillerBee: Practical ZigBee Exploitation Framework,” Boston 2010, Boston, 2010.

[26] Travis Goodspeed, “A 16 bit Rookit and Second Generation Zigbee Chips,” BlackHat USA, USA, 2009.

[27] LINDNER, “Router Exploitation,” BlackHat USA, USA, 2009

[28] John Heasman, “Hacking the Extensible Firmware Interface,” BlackHat USA, USA, 2007.

[29] Barnaby Jack, “Exploiting Embedded Systems,” BlackHat Amsterdam, Amsterdam, 2006.

[30] Brendan O’Connor, “Vulnerabilities in Not-So Embedded Systems,” BlackHat USA, USA, 2006.

[31] Breeuwsma, M. F. “Forensic imaging of embedded systems using JTAG (boundary-scan),” digital investigation 3.1 (2006): 32-42.

[32] The MITRE Corporation, “CAPEC CATEGORY: Software,” [Internet], https://capec.mitre.org/data/definitions/513.html.

[33] The MITRE Corporation, “CAPEC CATEGORY: Hardware,” [Internet], https://capec.mitre.org/data/definitions/515.html.

[34] OWASP, “OWASP Top Ten Cheat Sheet” [Internet], https://www.owasp.org/index.php/OWASP_Top_Ten_Cheat_Sheet.

[35] SANS, “CWE/SANS TOP 25 Most Dangerous Software Errors,” [Internet], https://www.sans.org/top25-software-errors/.

[36] Common Vulnerabilities and Exposures, “CVE-2015-4080,” [Internet], https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-4080.

[37] Common Vulnerabilities and Exposures, “CVE-2014-8730,” [Internet], https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-8730.

홍 바 울

e-mail : visitator00@korea.ac.kr
2015년 홍익대학교 컴퓨터공학과(학사)
2015년～현 재 고려대학교 정보보호대학원 정보보호학과 석사과정
관심분야 : Security threat-risk modeling, Security evaluation

박 민 수

e-mail : minsoon2@korea.ac.kr
2010년 신라대학교 컴퓨터네트워크학과 (학사)
2013년 고려대학교 정보보호학과(석사)
2013년～현 재 고려대학교 정보보호대학원 정보보호학과 박사과정
관심분야 : Security Assurance, Security Evaluation, Digital Forensic

이 상 민

e-mail : leesangmin@korea.ac.kr
2015년 호서대학교 정보보호학과(학사)
2016년～현 재 고려대학교 정보보호대학원 정보보호학과 석사과정
관심분야 : Binary vulnerabilities analysis

김 승 주

e-mail : skim71@korea.ac.kr
1994년 성균관대학교 정보공학과(학사)
1996년 성균관대학교 정보보호학과(석사)
1999년 성균관대학교 정보보호학과(박사)
1998년～2004년 KISA 팀장 (舊한국정보보호진흥원)
2004년～2011년 성균관대학교 정보통신공학부 조교수, 부교수
2011년～현 재 고려대학교 사이버국방학과/정보보호대학원 정교수
관심분야 : Security Engineering, Security Threat-Risk Modeling, Security Testing, Security Evaluation, Usable Security