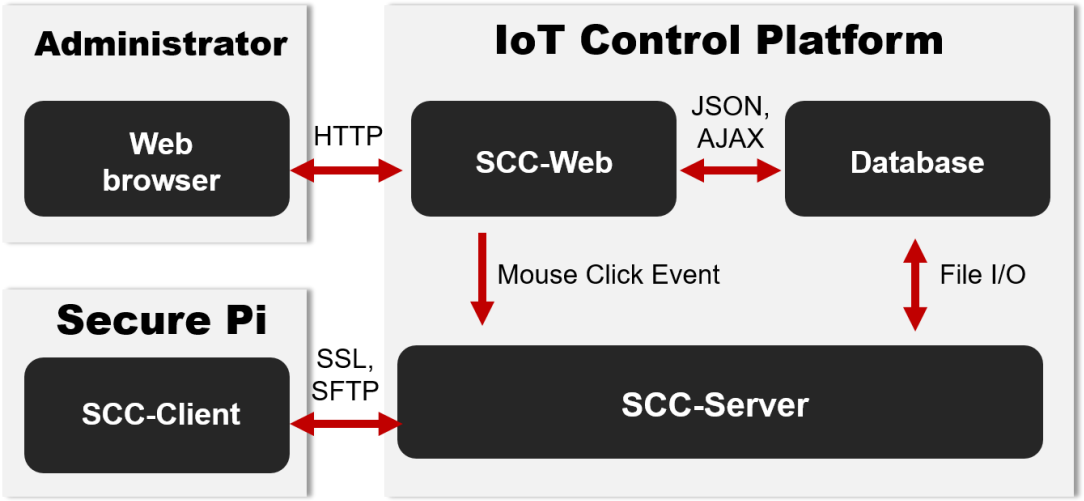
Security Control Center

1 개발 진행 사항

1.1 시스템 아키텍쳐

Secure Pi의 상태를 모니터링 할 수 있는 보안 관제 시스템 SCC(Security Control Center)를 개발하였다. SCC는 [그림 1]과 같이 SCC-Client, SCC-Server, SCC-Web, Database로 구성되어 있으며 SCC 관리자는 웹 브라우저를 통해 Secure Pi를 모니터링 한다.

SCC-Server, SCC-Web, Database가 있는 IoT Control Platform은 Ubuntu PC(16.04 LTS)를 사



[그림 1] SCC System Architecture

용하였고, SCC-Client가 있는 Secure Pi는 Raspberry Pi 2 (Model 2)에 Atmel TPM을 장착하였다.

SCC-Client는 수집한 Secure Pi의 데이터를 SCC-Server에 안전하게 전달하기 위해 SSL(Secure Sockets Layer)을 사용하였으며, SCC-Server는 SFTP(SSH File Transfer Protocol)를 이용하여 Secure Pi의 펌웨어 업데이트를 진행한다. SCC-Server가 파일 입출력을 통해 Database에 저장한 데이터는 SCC-Web이 JSON(JavaScript Object Notation)과 AJAX(Asynchronous JavaScript and XML)을 사용하여 웹 페이지의 형태로 나타낸다.

1.2 SCC-Client

Development Platform	SecurePi (Raspberry Pi 2 (Model 2) + TPM)
TPM	Atmel TPM
Language	C, Shell script
Library	OpenSSL

SCC-Client는 SecurePi를 사용하여 개발하였다. 총 8가지의 보안 요소 기술을 수행하며 SCC-Server와 통신한다. 이 8가지 보안 요소 기술은 다음과 같다.

1. Secure Key Storage & Management Monitoring
2. Secure Boot Monitoring
3. Secure Firmware Update Monitoring
4. Remote Attestation Monitoring
5. Filesystem Integrity Monitoring
6. Filesystem Encryption Monitoring
7. Login Monitoring
8. Packet Monitoring

현재 개발이 완료된 부분은 1~4번, 7번의 보안 요소 기술이다. 5번 Filesystem Integrity Monitoring은 IMA/EVM Library를 이용할 예정이며, 6번 Filesystem Encryption Monitoring은 eCryptFS check Daemon을 이용할 예정이다. 8번 Packet Monitoring은 iptables에 rule을 지정하고 해당 rule에 벗어난 Packet 요청이 들어오면 로그 작성을 할 예정이다.

1.3 SCC-Server

Development Platform	Ubuntu 16.04 LTS (Local)
Language	C, Shell script
Library	OpenSSL

SCC-Server는 SSL을 통해 SCC-Client로부터 로그 정보를 수집 후, 데이터베이스에 저장을 하는 역할을 한다. 현재 안전하게 로그를 수집하도록 개발이 완료된 상태이다.

1.4 SCC-Web and Database

Back-end	Node.js (v4.2.6)
Front-end	PUG (v.2.5.5)
Database Management	MySQL (v14.14)

SCC-Web은 Node.js를 통해 Back-end 개발을, PUG를 통해 Front-end 개발을 하고 있다. 현재 서비스는 (https://163.180.118.193:3000)을 통해 진행 중이며, 디바이스 등록/수정/삭제와 각 디바이스 별 로그 기록 열람이 가능하다. 하지만 Database에 저장된 데이터를 웹에 출력하는 부분의 보완

이 더 필요하며, Database에 데이터 저장 시 Split하여 양식에 맞게 저장하도록 수정해야 한다.

2 결론

IoT 디바이스의 보안을 위해 디바이스 수준을 고려한 보안 플랫폼과, 이를 모니터링 하는 보안 관제 시스템이 필요하다. 프로젝트 계획서에서 IoT 디바이스 보안 관제 시스템의 핵심 요소기술을 정의하였고, 이전 연구를 통해 개발된 IoT 디바이스 보안 플랫폼을 기반으로, 보안 관제 시스템을 제안하였다. 이를 통해 보안 관제 시스템의 관리자는 IoT 디바이스들을 효율적으로 침해 대응 및 관제 할 수 있으며, 실시간 감시를 통해 다양한 불법적 주체에 대한 접근을 차단할 수 있다. 즉, oneM2M에서 제안하는 IoT 디바이스의 보안 진단 및 컨설팅이 가능하다.

개발의 보완이 필요한 부분을 더 수정하여 개발을 진행시킬 예정이다.