IoT(사물인터넷)

33기 방은진

1. 서론

1.1 주제 선정 배경

c언어는 많은 분야에서 자주 쓰이는 프로그래밍 언어 중 하나이므로 c언어에 대해 공부해야 한다고 생각했지만, 정작 c언어로 무엇을 할 수 있는지, 어디에 c언어를 활용할 수 있는지에 대해 잘 알지 못하였다. 따라서 c언어가 주로 사용되는 영역들 중 하나인 임베디드 시스템과 IoT(사물인터넷)에 주목하여 탐구해보았다. 나아가 IoT에 더 집중하여 iot 보안 관련 취약점과 대응을 알아보고 보안 지식을 쌓고자 한다.

1.2 탐구 소개

들어가기에 앞서 c언어의 특징, 활용분야에 대해 짚어보고 c언어의 활용 영역 중 하나인 임베디드 시스템에 대해 알아볼 것이다. 그리고 임베디드 시스템, c언어와 밀접한 iot에 대해 알아보고 iot의 보안 취약점과 iot 보안 기술에 대해 공부해볼 것이다.

2. 본론

2.1 c언어

2.1.1 c언의 유래

1972년 벨 연구소에서 데니스 리치(D.M.Ritchie)에 의해서 설계 개발된 시스템 기술용의 프로그래밍 언어이다. UNIX라는 운영 체제를 개발할 것을 목적으로 설계한 언어로 UNIX OS의 대부분이 이 언어로 개발되었다. 컴퓨터의 구조에 밀착한 기초 기술이 가능한 것과 간결한 표기가 될 수 있는 것 등을 주요 특징으로 하고 있다.

2.1.2 c언의 특징

시스템을 제어하는 언어(=시스템 프로그래밍 언어)는 메모리 주소나 CPU 같은 하드웨어까지 직접 조작할 수 있어야 하는데, C 언어는 이러한 저수준(하드웨어 가까운) 기능을 제공하면서도, 동시에 변수, 함수, 조건문 같은 고수준(사람이 이해하기 쉬운) 개념도 지원한다. 따라서 하드웨어도 제어할 수 있지만, 일반적인 프로그램을 짜는 데도 문제없이 사용할 수 있는 범용적인 언어이다.

모든 표기법이 유연하며 간결하게 되어있어 프로그래밍하기 쉽고 편리한 언어로 평가된다. c로 작성된 프로그램은 크기가 작으며 실행 속도가 빠르고 메모리를 효과적으로 사용해 기계어 수준의 효율성을 가지고 있다. 또한 기계어 수준의 구체적인 하드웨어 제어가 가능하여 실제로 스마트폰, TV, 세탁기 등의 여러 가지 전자기기 안에 들어가는 임베디드 프로그램은 대부분 c언어로 개발된다.

이식성이란 한번 작성된 프로그램을 다른 CPU를 가지는 하드웨어로 얼마나 쉽게 이식할 수 있는가를 나타내는 정도를 뜻한다. 예를 들어 인텔 CPU에서 어셈블리 언어로 작성된 프로그램은 퀠컴의 CPU에서 동작되지 못한다. CPU가 지원하는 명령어가 다르기 때문이다. 하지만 c언어는 이식성이 뛰어나 인텔 CPU에서 작성된 프로그램일지라도 퀠컴 CPU의 c 컴파일러로 컴파일만 다시 하면 퀠컴 CPU에서도 동작 가능하다. 즉 하드웨어의 의존도가 낮아지는 것이다.

2.1.3 c언어의 활용

c언어는 운영체제 개발을 목적으로 설계된 프로그래밍 언어이다. 운영체제(operating system)는 컴퓨터 하드웨어를 관리하고 응용 프로그램이 실행될 수 있도록 서비스를 제공한다. 많은 운영체제가 c언어로 구현되어 왔는데 유닉스, 리눅스, 애플의 OS X, 구글의 안드로이드 등이 그 예이다. 이는 c언어의 성능과 이식성에 기인한 것이다.

임베디드 시스템에서도 c언어가 많이 사용된다. 이는 뒤에서 다시 다루겠다.

실시간 시스템(real-time system)은 어떤 미션이 주어지고 주어진 시간안에 미션을 해결하는 시스템이다. 예를 들어 미사일에 장착된 프로그램은 미사일이 날아가는 동안에 미사일을 제어하여 목표로 인도해야 한다. 이러한 경우에도 c언어가 많이 사용된다. 또한 마이크로컨트롤러 프로그래밍에서도 C언어는 자원이 제한된 환경에서의 효율적인 코드 작성을 가능하게 해준다.

2.2 임베디드 시스템

2.2.1 정의

어떤 제품이나 솔루션에 추가로 탑재되어 그 제품 안에서 특정한 작업을 수행하도록 하는 솔루션을 말한다. 예를 들어 주된 용도가 전화인 휴대폰에 텔레비전 기능이 들어가 있다면, 텔레비전 기능(시스템)이 바로 임베디드 시스템이다. 곧, 본 시스템에 끼워 넣은 시스템이라는 뜻이다.

첨단 기능이 들어 있는 컴퓨터, 가전제품, 공장자동화 시스템, 엘리베이터, 휴대폰 등 현대의 각종 전자·정보·통신 기기는 대부분 임베디드 시스템을 갖추고 있다. 대개의 경우 그 자체로 작동할 수도 있지만, 다른 제품과 결합해 부수적인 기능을 수행할 때에 한해 임베디드 시스템이라고 한다.

컴퓨터의 경우에는 전용 동작을 수행하거나 특정 임베디드 소프트웨어 응용 프로그램과 함께 사용되도록 디자인된 특정 컴퓨터 시스템 또는 컴퓨팅 장치를 일컫는다. 미리 지정된 아주 제한적인 수의 동작을 수행하도록 설계된 특수한 컴퓨터 시스템으로 볼 수 있다. 인베디드 시스템은 소형 컴퓨터, 마이크로컨트롤러 또는 전용 칩과 같은 하드웨어와 해당 하드웨어를 제어하기 위한 소프트웨어로 구성된다.

2.2.2 임베디드 시스템에서의 c언어

임베디드 시스템은 제한적 기능을 수행하므로 속도가 가장 중요하기 때문에 c언어가 많이 사용된다. 또한 c언어의 저수준 하드웨어 접근 가능성과 효율성, 이식성의 특징 덕분에 c언어는 임베디드 시스템 개발에 선호된다. 마이크로컨트롤러 프로그래밍에서도 C언어는 자원이 제한된 환경에서의 효율적인 코드 작성을 가능하게 해줍니다.

2.2.3 c언어를 이용하는 임베디드 시스템과 iot

iot 시스템을 구축하기 위한 도구 중 하나인 c언어는 시스템 구축 과정에서 임베디드 시스템, 마이크로컨트롤러(Arduino,아두이노)에 사용되어 저수준 하드웨어 제어와 빠른 속도, 메모리 최적화를 목적으로 한다. 여기서 임베디드 시스템과 iot의 차이는 시스템의 범위로 볼 수 있다. 임베디드 시스템은 혼자 작동하는 제한된 용도의 프로그래밍 된 장치, iot는 이것들을 연결시켜주는 환경이라고 볼 수 있다. 즉 iot는 임베디드 시스템의 네트워크화된 형태라고 볼 수 있으며 임베디드 시스템의 확장된 개념

이라고 이해할 수 있다.

2.3 iot(사물인터넷)

2.3.1 iot의 정의와 특징

기존의 통신 기술이 사물과 사물 그리고 사물과 사람 간의 언제(Anytime) 어디서나 (Anyplace) 무엇이든지 (Anything) 서로 주고받은 개념이라고 할 수 있다.

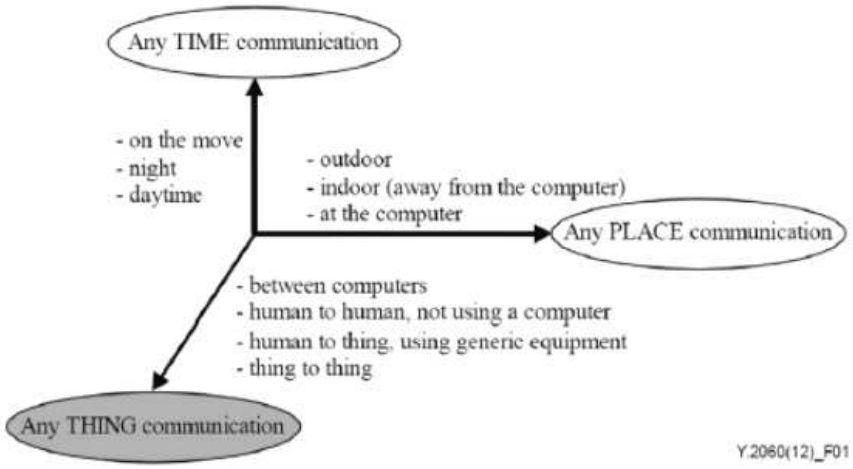
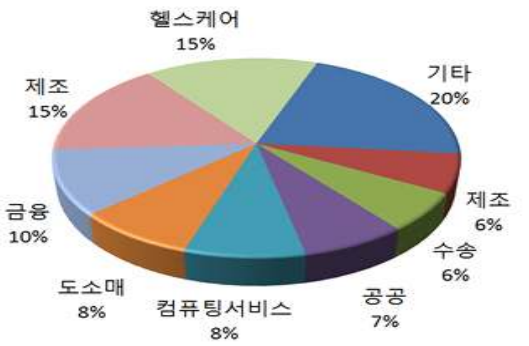


Fig. 1. Basic Concept of IoT(Internet of Things) [7,12,13]

사물인터넷(iot)은 정보를 교환하는 인정된 표준을 기반으로 구축된 네트워크 시스템이다. iot 내장 시스템에 여러 센서를 포함하면 멀리 있는 물리적 장치에서 실시간 데이터를 수집하여 지능적이고 자동화된 의사결정으로 IoT 네트워크를 효율적으로 관리할 수 있다.

2.3.2 iot 적용 분야



▶▶ 그림 1. 2020 산업별 IOT 부가가치 창출 비중[4]

iot는 현재 다양한 분야에 서비스 응용이 적용되고 있다. 1차 산업 환경으로 시작해

미래 산업의 전반에 걸쳐 그 서비스가 확대되고 있다. 최근 IOT는 모바일 플랫폼을 기반으로 스마트화되어 가고 있다. 또한 좀 더 개방된 표준화를 기반으로 한 환경에서는 서비스를 위한 앱 개발 자체보다 앱 개발 인터페이스등의 소프트웨어적인 요소를 기기에 활용하기 위한 노력이 증가하고 있다.

표 1. 사물인터넷 적용 분야

분야	내용	서비스 제품
헬스 케어	전강보조도구, 헬스 정보 송수신, 스마트폰을 이용한 헬스 케어 앱	핏빗플렉스(핏빗), 픽스(코벤티스), S헬스 서비스(삼성전자), 2net(퀄컴), 트윗피(하기스)
홈케어	조명 제어, 지능 주택관리, LBS방법, 외출 보안 시스템, 냉난방환기 자동 조절, 스마트 홈서비스	스마트싱스(Smartthings), 스마트홈, 스마트라이프(SKT)
자동차	텔레매틱스, 무인자동차, 스마트카, 커넥티드 카, 차량원격관리	OnStar(GM), Sync(포드), 불루링크(현대차), 무인자동차(구글), 스마트 오토모티브(SKT)
교통	교통안전, 국도 모니터링, 배기가스 실시간 감지, 택시 무선 결재, 디지털 운행관리	지능형 교통 서비스, 지능형 주차 서비스 SF Park(샌프란시스코시)
농협	실시간 작물 상태 모니터링, 온도/습도 감지 조정, 농작물 수확량 재고관리	스마트팜(SKT), 지능형 파종 서비스, 지능형 젖소관리 서비스(네델란드 사프크드사)

흔히 주변에서 보기 쉬운 스마트폰을 통한 헬스 케어 앱을 시작으로, 어플 하나로 온도, 조명, 보안 조종이 가능한 스마트홈 서비스, 차량원격관리, 실시간 교통 모니터링, 스마트팜까지 우리 일상생활에는 이미 다양한 iot 서비스가 개입하고 있다.

2.3.3 iot의 보안 취약점

IoT 의 모든 기술들은 다양한 기술과 프로토콜로 구성 되어 있으며 수많은 센서와 디바이스, 미들웨어 플렛폼, 서비스 플렛폼과 유기적으로 결합되어 있다. 또한 사용자 인터페이스 측면에 항상 노출이 되어 있는 상태이며 많은 사물과 많은 사람을 연결하면 더욱 더 보안의 취약성이 커질 수밖에 없다.

사물인터넷 시대에는 컴퓨터나 스마트폰뿐만 아니라 수백억 개에 달하는 다양한 유형의 사물들이 인터넷에 연결된다. 이러한 사물인터넷 디바이스들은 제한된 배터리 용량에 컴퓨팅 파워도 떨어지는 소형의 디바이스인 경우가 대부분이다. 따라서 별도의 암호화 과정 없이 데이터를 생성한 후 주변의 다른 디바이스들을 통해 인터넷으로 전달된다. 이러한 과정에서 악의적인 사용자에 의해 불법적으로 데이터가 수집되거나 변조된다면, 개인의 프라이버시를 침해하는 것뿐만 아니라 심각한 보안 사고까지 일으킬 수 있다. 사물인터넷 서비스는 가상세계와 현실 세계를 연결하는 것이기 때문에,

사이버 공간에서의 해킹은 그대로 물리적인 공간의 위험으로 전이될 수 있기 때문이다.

[표 2-1] IoT 제품 유형별 보안 위협[6]

유형	제품	보안 위협	보안위협 원인
네트워크 제품	홈캠, 네트워크 카메라 등	• 사진 및 동영상을 공격자 서버로 전송 • 네트워크에 연결 된 홈캠 등을 원격으로 제어하여 임의 활용	• 접근통제 부족 • 전송데이터 보호 부족 • 물리적 보안 취약
센서 제품	온/습도 센서 등	• 변조 또는 잘못된 온/습도 정보 전송	• 전송데이터 보호 부재 • 데이터 무결성 부재 • 물리적 보안 취약
생활가전 제품	청소기, 인공지능 로봇 등	• 알려진 운영체제 취약점 및 인터넷 기반 해킹 위협 • 로봇 청소기에 내장 된 카메라를 통해 피해자 집 모니터링	• 인증메커니즘 부재 • 펌웨어 업데이트 부재 • 물리적 보안 취약
멀티미디어 제품	청소기, 스마트 냉장고 등	• PC환경에서의 모든 악용 행위 • 카메라/마이크 내장 시 사생활 침해	• 인증메커니즘 부재 • 강도가 약한 비밀번호 사용 • 펌웨어 업데이트 취약 • 물리적 보안 취약
제어제품	디지털 도어락, 가스밸브 등	• 제어기능 탈취로 도어락의 임의 개폐	• 인증메커니즘 부재 • 강도가 약한 비밀번호 사용 • 접근통제 부재 • 물리적 보안 부재
	모바일앱(웹) 등	• 앱 소스코드 노출로 인한 IoT 제품 제어기능 탈취	• 인증정보 평문 저장 • 전송 데이터 보호 부재

- 쉬운 연결의 부작용

엣지 컴퓨팅이 메인 서비스로 진출하고 고급 5G 네트워크가 RedCap(Reduced Capability, 5G망을 한 단계 업그레이드해 경량화한 IoT 서비스 지원 기술) 같은 기능을 출시하면서 IoT 보안 위험은 더욱 커졌다. RedCap 5G는 원래 기업 IoT의 도입을 촉진하기 위해 만들어졌다.

여기서 엣지 컴퓨팅이란 클라우드 컴퓨팅과 반대되는 개념으로, 인터넷이 아닌 로컬 장치(예: 스마트폰, 태블릿, IoT 장치 등)에서 데이터를 처리하는 기술이다. 이를 통해 데이터 처리 및 분석이 인터넷 대역폭을 절약하고, 응답 시간을 단축하여 네트워크 대역폭 혼잡을 완화할 수 있다. 데이터가 로컬에서 처리되기 때문에 데이터 전송 중 발생할 수 있는 보안 위험을 줄일 수 있다.

하지만 RedCap 5G로 스마트폰이나 스마트 워치 같은 셀룰러(무선통신망) 연결 모바일 기기는 근처의 제한된 기기에 임시 연결을 제공하는 허브 역할을 수행하게 됐다. 쉽게 말해서 스마트폰이 주변의 다른 산업용 장비나 센서같은 기기들을 임시로 연결해주는 가교 역할을 할 수 있다는 것이다. 연결이 쉬워진 만큼, 여러 기기를 한번에 관리하기 쉬워 비즈니스 효율성을 높이는 데 도움을 주기도 했지만, 보안이 허술한 모바일 기기, 가령 산업 진단 장비에 자동으로 연결될 경우 맬웨어 스턱스넷

(Stuxnet) 같은 위협에 노출될 수 있다. 맬웨어 스턱스넷이란 2010년에 발견된 웜 바이러스로, 마이크로소프트 윈도우를 통해 감염되어 지멘스 산업시설을 감시하고 공격했던 악성 소프트웨어이다.

그 외에도 이 외에도 공장 출하 시 입력된 아이디/패스워드를 변경없이 사용하거나(공장에서 출하된 iot 기기의 기본 비밀번호는 인터넷에서 쉽게 찾을 수 있음) 별도의 보안이 설정되지 않은 네트워크 상에 기기를 연결하거나(보안이 없는 카페, 공공장소의 무료 Wi-Fi에 IoT 기기를 연결하면 해커가 쉽게 기기 데이터를 가로채거나 조작할 수 있음), 물리적으로 보안이 되지 않은 영역에 기기를 설치 또는 사용하거나(직접 기기를 해킹하거나, SD 카드/USB 포트를 통해 악성코드를 심을 수도 있음), 센서 등이 연결된 기기의 데이터 전송 시 보호되지 않은 전송을 시도한다면(중간에서 데이터를 훔쳐보거나 조작할 수 있음) iot 보안이 취약해질 수 있다.

2.3.4 iot 보안

• iot 구성장비에 대한 보안

iot 구성장비에 대한 보안 iot를 구성하고 있는 장비는 유.무선서비스와 스마트장비의 결합으로 이루어져 있다. 또한 장비의 특성상 경량의 장비로 구성되어있다. 때문에 *저전력 암호화 모듈의 적용*이 필요하다. 또한 iot 플랫폼의 구성 운영체제에 대한 보안모듈의 운영이 필요하다. 뿐만 아니라 기기의 인증에 대한 위조 및 변조의 방지용 보안솔루션이 적용된다.

• iot 서비스를 위한 네트워크 보안

iot 서비스를 위한 네트워크는 기존의 네트워크와 센서 네트워크가 응용된다. 때문에 *게이트웨이에서의 보안솔루션 적용과 네트워크의 외부로부터의 침입에 대응하는 솔루션*이 적용되어야 한다. 또한 iot는 원격제어 서비스 응용이 다양하므로 원격보안관리에 대한 보안성 강화도 역시 필요하다.

• iot 플랫폼과 애플리케이션 보안

iot는 기존의 스마트 장비에 대한 적용이 대부분을 차지한다. 때문에 서비스에 사용되는 스마트 장비에 대한 *철저한 보안 인증*이 필요하다. 뿐만 아니라 가정과 개인이 접속하여 사용하는 경우, 금융, 헬스 등의 개인정보에 대한 *암호화 전송 기술*과 정보열람의 접근 제어에 대한 보안 모듈이 적용되어야 한다.

• 모바일 금융 적용 따른 강화된 보안

이미 일반화된 모바일 금융 서비스를 기반으로 하고 있는 iot 서비스는 강화된 보안모듈로 사용자 인증 및 데이터 암호화, 단말에 대한 특수성을 고려해야 한다. 더불어 프라이버시에 대한 보안성도 고려해야 하며, 개인정보 송수신과 접근, 정보 관리 라이프사이클을 적용하여 보안성 또한 높여야 한다.

3. 결론

3.1 SSR을 작성하면서 느낀 점

전공 수업에서 공부하는 c언어에 대해, 그동안 코드만 공부하고 c언어라는 프로그래밍 언어 자체에 대해 잘 알지 못했는데 c언어의 효율성과 하드웨어 제어 가능성, 이식성이라는 특징에 대해 흥미롭게 배울 수 있었다. 또한 자료를 찾던 도중 왜 임베디드 시스템은 c언어를 주로 이용하는 것이고 iot랑 왜 관련이 있는 건지, iot와 임베디드 시스템은 왜 다르게 정의되는지, 이해하기 어려울 때가 많았다. 이를 알아내는 과정에서 iot는 마이크로컨트롤러인 아두이노로 설계 가능하며, 이 아두이노는 c언어로 설계 가능하고, iot는 임베디드 시스템의 확장된 개념이라는 것을 깨닫게 되었다. 그리고 c언어의 빠른 처리 속도 등과 같은 특징 덕분에 임베디드 시스템 설계에 적합하다는 것 또한 알게 되었다.

iot, 즉 사물인터넷에 대해 스마트홈, 스마트팜 등 유행하는 최신 기술들에 대해 기존에 알고 있었지만 이렇게 다양한 보안 위험이 존재하는지 이번 SSR을 통해 체감하게 되었다. 이를 통해 어떻게 iot 보안을 향상시킬 수 있을지 다양한 방면에서 고민해보는 것이 앞으로의 발전에 있어 중요하다는 것을 알게 되었다.

3.2 다음에 다룰 SSR

이번 SSR에서 iot가 정확히 무엇이고 어떤 취약점이 있으며 어떻게 보호하면 좋을지 짧게 알아보는 시간이었다면 다음에는 iot 보안 기술에 대해 개념, 용어, 기술적인 측면에서 지금보다 더 자세하게 탐구해보고 싶다. 또한 가능하다면 필자가 할 수 있는, iot 시스템에서의 c언어를 이용한 간단한 코드를 짜보고 싶다. 스마트팜을 가정해서 온도, 햇빛, 토양 수분 등에 대한 여러 종류의 센서를 설정하고, 온도가 특정 온도 밑으로 떨어질 경우, 화면에 경고가 표시되는 등(이게 iot가 맞나 싶지만...) iot에 대해 더 깊이 이해하고 싶다.

4. 참고문헌

두산백과. (n.d.). *C언어*. 네이버 지식백과. https://terms.naver.com/entry.naver?docId=1179633&cid=40942&categoryId=32838

천인국. (2024). *두근두근 C언어 with 챗GPT(개정판)*. 천인국(편저), 1. 첫걸음 : C언어란?(pp. 17-19). 생능출판.

임선자, 최은희, 최필주, 이석환, & 권기룡. (2024). *IoT 침입 탐지 시스템을 위하여 기계학습 분류자를 사용한 블록체인 기반 연합학습*. DBpia.

이종식. (2018). *사물인터넷(IoT)발전을 위한 소스프로그램 보호방안 연구: 프로그램의 보호와 유사표절 연구*. NCI

김시정 & 조도은. (2015). *IOT(Internet of Things) 보안 기술 동향*. DBpia.

IoT 보안 문제 5가지와 해결 방안. (2023.09.27). CIO 뉴스. https://www.cio.com/article/3508266/iot-%EB%B3%B4%EC%95%88-%EB%AC%B8%EC%A0%9C-5%EA%B0%80%EC%A7%80%EC%99%80-%ED%95%B4%EA%B2%B0-%EB%B0%A9%EC%95%88.html

Edge Computing. (n.d.). UNIWIDE. https://www.uniwide.co.kr/page/PAGE_000000000000009/view.do?menuNo=21

금동권. (2019). *블록체인을 이용한 IoT 기기 보안 및 인증*[석사학위, 숭실대학교 정보과학대학원]. RISS

Iot와 임베디드의 차이 . (2023). https://velog.io/@jayce_97/Iot%EC%99%80-%EC%9E%84%EB%B2%A0%EB%94%94%EB%93%9C%EC%9D%98-%EC%B0%A8%EC%9D%B4.

IoT, 프로그래밍과의 관계에 대해 알아볼까요? . (2018). https://jjeongil.tistory.com/198.