KIPS Trans. Softw. and Data Eng.
Vol.10, No.12 pp.595~602
ISSN: 2287-5905 (Print), ISSN: 2734-0503 (Online)

https://doi.org/10.3745/KTSDE.2021.10.12.595

Secure Scheme Between Nodes in Cloud Robotics Platform

Hyungjoo Kim†

ABSTRACT

The robot is developing into a software-oriented shape that recognizes the surrounding situation and is given a task. Cloud Robotics Platform is a method to support Service Oriented Architecture shape for robots, and it is a cloud-based method to provide necessary tasks and motion controllers depending on the situation. As it evolves into a humanoid robot, the robot will be used to help humans in generalized daily life according to the three robot principles. Therefore, in addition to robots for specific individuals, robots as public goods that can help all humans depending on the situation will be universal. Therefore, the importance of information security in the Cloud Robotics Computing environment is analyzed to be composed of people, robots, service applications on the cloud that give intelligence to robots, and a cloud bridge that connects robots and clouds. It will become an indispensable element for In this paper, we propose a Security Scheme that can provide security for communication between people, robots, cloud bridges, and cloud systems in the Cloud Robotics Computing environment for intelligent robots, enabling robot services that are safe from hacking and protect personal information.

Keywords : Security, Robot, Robot-Human Interaction, Key Management, Cloud Robotics Platform

Cloud Robotics Platform 환경에서 Node간 안전한 통신 기법

김 형 주†

요 약

로봇은 주변 상황을 인지하고 Task를 부여받는 software oriented 형상으로 발전하고 있다. Cloud Robotics Platform은 로봇에 Service Oriented Architecture 형상을 지원하기 위한 방법으로, 상황에 따라 필요한 Task와 Motion Controller를 클라우드 기반으로 제공할 수 있는 방안이다. 휴머노이드 로봇으로 진화할수록 로봇은 로봇 3대 원칙에 따라 보편화된 일상생활 속에서 인간에게 도움을 주기 위해 사용될 것이다. 따라서 특정 개인만을 위한 로봇 이외에도, 상황에 따라 모든 인간에게 도움을 줄 수 있는 공공재로써의 로봇이 보편화될 것이다. 따라서, 생성하는 정보는 사람, 로봇, 로봇에 지능을 부여하는 클라우드 상의 서비스 애플리케이션, 로봇과 클라우드를 이어주는 클라우드 브릿지로 구성될 것으로 분석되는 Cloud Robotics Computing 환경에서 정보보안의 중요성은 인간의 생명 및 안전을 위해 필수불가결한 요소로 자리잡게 될 것이다. 본 논문에서는 지능화된 로봇을 위한 Cloud Robotics Computing 환경에서 사람, 로봇, 클라우드 브릿지, 클라우드 시스템간 통신 시 보안을 제공하여 해킹으로부터 안전하고 개인의 정보가 보호되는 로봇 서비스가 가능할 수 있는 Security Scheme을 제안한다.

키워드 : Security, Robot, Robot-Human Interaction, Key Management, Cloud Robotics Platform

1. 서 론

로봇 하드웨어 디자인은 크게 두 가지 종류로 나뉜다. 첫번째는 “여러 개의 간단한 작업별 로봇을 사용하여 동일한 환경에서 여러 작업을 수행”하는 로봇으로 Task가 고정 프로그래밍된 형상을 띄우는 경우가 대부분이다. 두 번째는 “여러 작업을 수행하는 인간과 비슷한 로봇”으로 서비스 지향적이며 상황에 따라 Task가 변화할 수 있는 로봇이다. 현재 서비스되고 있는 로봇은 첫번째 Task 고정형 로봇이 주를 이루고 있으나, 미래에는 두번째 로봇인 휴머노이드 형태로 변화될 것이다.

휴머노이드 로봇, 즉 지능형 로봇은 다양한 환경에서 상황을 인지하고 추론하여 사람에게 필요한 Task를 부여하는 포괄적이고 광범위한 서비스를 제공할 수 있는 로봇을 말한다. 따라서 고정된 Software가 아닌, Software as a Service 형상과 같이 Service Oriented Architecture 형상을 지원하여 필요한 외부 서비스를 로봇에게 부여할 수 있어야 한다.

클라우드 서비스는 크게 메모리, 서버 등의 자원을 제공

† 정 회 원 : KT 융합기술원 책임연구원
Manuscript Received : October 19, 2021
First Revision : November 2, 2021
Accepted : November 11, 2021
* Corresponding Author : Hyungjoo Kim(k.hyungjoo@gmail.com)

하여 주는 IaaS (Infrastructure as a Service), 응용 SW를 제공하여 주는 SaaS (Software as a Service), 개발 플랫폼을 구축할 필요 없이 개발 시 필요한 것들을 웹에서 쉽게 활용할 수 있게 하는 PaaS (Platform as a Service)로 크게 나눌 수 있다[1].

Cloud Robotics, Cloud Robotics Platform은 클라우드 컴퓨팅 기술이 Robot에 적용된 기술로, PaaS와 SaaS 서비스가 Robot에게 제공된다. 로봇 S/W 개발 환경을 위한 PaaS는 구글, 아마존 등 글로벌 기업 및 국내 이동통신사를 중심으로 연구 및 서비스 되고 있다. 일반적인 IT분야의 PaaS와 마찬가지로, 컴파일, 배포 등 서비스 개발에 관련된 작업들을 처리하는 통합 개발 환경(IDE: Integrated Development Environment)과 모니터링 등의 운영과 관련된 작업들을 처리하는 서비스[2-4]를 제공한다.

SaaS 형상의 Cloud Robotics Platform은 로봇에 Service Oriented Architecture 형상을 지원하기 위한 방법으로, 상황에 따라 필요한 Task와 Motion Controller를 클라우드 기반으로 제공할 수 있는 방안이다. 일반적인 SaaS 형상으로, 로봇에 있는 다양한 센서를 이용하여 복합적인 상황을 인지한 후, 필요한 서비스 및 지능을 로봇에게 펌웨어 형태로 제공하거나 Cloud에서 직접 연산하여 단순 Motion Control만을 로봇에게 명령 내리는 형상으로 제공될 수 있다.

휴머노이드 로봇으로 진화할수록 로봇은 로봇 3대 원칙에 따라 보편화된 일상생활 속에서 인간에게 도움을 주기 위해 사용될 것이다. 따라서 특정 개인만을 위한 로봇 이외에도, 상황에 따라 모든 인간에게 도움을 줄 수 있는 공공재로써의 로봇이 보편화될 것이다.

따라서, 생성하는 정보는 사람, 로봇, 로봇에 지능을 부여하는 클라우드 상의 서비스 애플리케이션, 로봇과 클라우드를 이어주는 클라우드 브릿지로 구성될 것으로 분석되는 Cloud Robotics Computing 환경에서 정보보안의 중요성은 인간의 생명 및 안전을 위해 필수불가결한 요소로 자리잡게 될 것이다.

이에, 사람이 공공재로써 존재하는 로봇을 사용하고자 할 때, 로봇으로부터 수집 및 처리되는 정보는 해당 정보의 원천인 특정 개인만이 취급할 수 있도록 보호되어야 한다. 또한, 정보의 보호 범위는 로봇뿐 아니라 로봇이 이용하는 클라우드 상의 애플리케이션까지 확장되어야 하며, 이 때 로봇의 행위는 특정 개인을 위한 목적으로 사용되었음을 증명할 수 있도록 하여 악의적인 행위에 대한 증명, Fee 측정 등의 사후관리 등의 서비스가 제공될 수 있어야 한다.

본 논문에서는 지능화된 로봇을 위한 Cloud Robotics Computing 환경에서 사람, 로봇, 클라우드 브릿지, 클라우드 시스템간 통신 시 보안을 제공하여 해킹으로부터 안전하고 개인의 정보가 보호되는 로봇 서비스가 가능할 수 있는 Security Scheme을 제안한다. 2장에서는 관련 연구를 설명하고, 3장에서 제안하는 Security Scheme을 설명하며, 4장에서 제안에 대한 안전성을 평가하고, 5장에서 결론을 맺도록 한다.

2. 관련 연구

2.1 Robot Operating System

Robot Operating System(이하 ROS)은 로봇을 위한 오픈소스 메타 운영 체제이다. 하드웨어 추상화, 저수준 장치 제어, 일반적으로 사용되는 기능 구현, 프로세스 간 메시지 전달, 패키지 관리를 포함하여 운영 체제에서 기대할 수 있는 서비스를 제공한다. 또한 여러 컴퓨터에서 코드를 획득, 구축, 작성 및 실행하기 위한 도구와 라이브러리를 제공한다.

메타 운영 체제란, ROS를 로봇을 위한 독립적인 운영체제로써 사용할 수 없으며 Ubuntu 등의 운영체제 위해 Docker Layer와 같이 미들웨어 Layer로써 동작함을 말한다.

로봇이 사물을 잡기 위해서는 손가락을 몇 도로 벌리고 몇 도로 오므리며, 오므릴 때 힘은 어느정도를 줘야하는지, 팔꿈치의 역할을 하는 관절 부분의 각도는 몇 도로 조절해야 하는지 등의 물리적 움직임을 지정해주어야 한다. ROS는 이런 물리적인 움직임을 제어/실현할 수 있는 다양한 라이브러리를 제공하여, 로봇 서비스 애플리케이션 개발자가 상위 레벨의 랭귀지를 이용하여 잡는다는 표현만으로 물리적인 움직임이 가능하도록 해준다.

ROS의 다양한 기능 중 서비스 애플리케이션 및 Cloud Robotics Platform과 가장 밀접한 기능은 ROS Communication Graph Level이다. ROS Communication Graph Level은 데이터를 처리하는 프로세스들간 Peer to Peer 네트워크이며, 주요 용어 정의는 하기와 같다[5].

a) 노드 : 노드는 계산을 수행하는 프로세스이다. ROS는 세분화된 규모로 모듈화되도록 설계되었다. 로봇 제어 시스템은 일반적으로 많은 노드로 구성된다. 예를 들어, 하나의 노드는 레이저 거리 측정기를 제어하고, 하나의 노드는 휠 모터를 제어하고, 하나의 노드는 현지화를 수행하고, 하나의 노드는 경로 계획을 수행하고, 하나의 노드는 시스템의 그래픽 보기를 제공하는 식이다. ROS 노드는 roscpp 또는 rospy 와 같은ROS 클라이언트 라이브러리를 사용하여 작성된다.
b) 마스터 : ROS Master는 이름 등록 및 조회를 제공한다. 마스터가 없으면 노드는 서로를 찾거나 메시지를 교환하거나 서비스를 호출할 수 없다.
c) 메시지 : 노드는 메시지를 전달하여 서로 통신 한다 . 메시지는 단순히 유형이 지정된 필드로 구성된 데이터 구조이다. 기본 유형의 배열과 마찬가지로 표준 기본 유형(정수, 부동 소수점, 부울 등)이 지원된다. 메시지에는 임의로 중첩된 구조 및 배열(C 구조와 매우 유사)이 포함될 수 있다.

Table 1. Experimental Results

First Law	A robot may not injure a human being or, through inaction, allow a human being to come to harm.
Second Law	A robot must obey the orders given it by human beings except where such orders would conflict with the First Law.
Third Law	A robot must protect its own existence as long as such protection does not conflict with the First or Second Law.

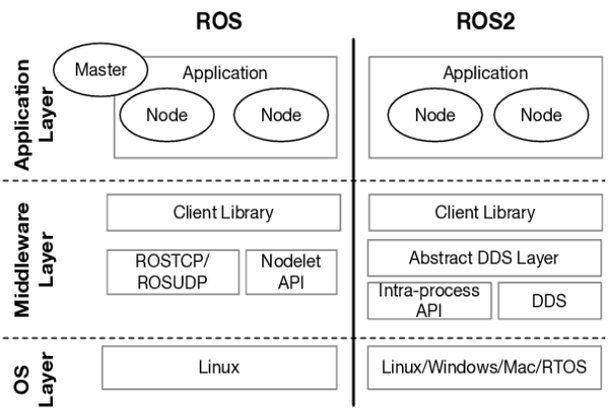

Fig. 1. ROS Architecture

d) Topic : 메시지는 Pub/Sub 시맨틱이 있는 전송 시스템을 통해 라우팅된다. 노드는 주어진 주제(Topic)에 메시지를 게시 하여 메시지를 보낸다. 주제는 메시지의 내용을 식별하는 데 사용되는 이름이다. 노드는 특정 종류의 데이터에 관심이 있을 시 적절한 주제로 구독하게 된다. 단일 주제에 대해 여러 동시 발행자와 구독자가 있을 수 있으며 단일 노드가 여러 주제를 발행 또는 구독할 수 있다. 일반적으로 게시자와 구독자는 서로의 존재를 알지 못한다. 중요한 점은 정보의 생산과 소비를 분리하는 것이다. 논리적으로 주제를 강력한 형식의 메시지 버스로 생각할 수 있다. 각 버스에는 이름이 있으며 올바른 유형이면 누구나 버스에 연결하여 메시지를 보내거나 받을 수 있다.

ROS Master는 ROS Computation Graph에서 Name Service 역할을 하며, ROS 노드에 대한 Topic 및 서비스 등록 정보 등을 저장 한다 . 노드는 마스터와 통신하여 등록 정보를 보고하며, 이러한 노드는 마스터와 통신을 함에 따라 다른 등록된 노드에 대한 정보를 수신하고 다른 노드와 연결될 수 있다. 마스터는 등록 정보가 변경될 때 관련 노드에 대한 콜백을 생성하여 새 노드가 실행될 때 노드가 동적으로 연결을 생성할 수 있도록 도와 준다.

노드는 다른 노드에 직접 연결되며, 마스터는 DNS 서버와 마찬가지로 조회 정보만을 제공한다. Topic을 구독하는 노드는 해당 주제를 게시하는 노드로부터 연결을 요청하고 합의/지정된 연결 프로토콜을 통해 해당 연결을 설정/생성 한다. ROS에서 사용되는 가장 일반적인 프로토콜은 표준 TCP/IP 소켓을 사용하는 TCP-ROS 이며, 다양한 노드는 패키지화 되어 서비스 애플리케이션에서 사용되는 구조이다.

하나의 서비스를 제공하기 위하여 센서나 액추에이터 단의 드라이브부터 센싱, 인식, 동작까지 하나의 프레임/패키지에서 프로그램을 작성하는 경우가 많다. 로봇 소프트웨어의 재사용을 위해서는 이를 각각 처리 프로세서의 목적에 따라 작게 나누는 것이 유용하다. 백앤드 단의 마이크로 서비스개발과 유사하며, 플랫폼마다 이를 컴포넌트화 혹은 노드 패키지화라고 한다. 최소 실행 단위로 나뉜 프로그램은 나누어진 노드끼리 데이터를 주고받아야하는데 플랫폼들은 이 데이터 통신에 대한 전반적인 사항을 모두 제공 한다. 각 노드는 하드웨어 의존성을 떠나 네트워크에서 통신을 제공함으로써 네트워크 프로그래밍이 가능하게 되고 로보틱스에서 흔히 다루는 원격제어에서 매우 유용하게 사용 된다. 즉, 패키지 내 다양한 노드가 존재하며 SaaS에도 노드가 존재할 수 있어, 로봇 또는 클라우드가 단독으로 지능을 갖는 것 뿐 아니라 로봇과 클라우드가 공동의 지능을 가지고 서비스를 제공할 수 있다.

2.2 Cloud Robotic Platform

클라우드 컴퓨팅은 다음과 같은 방법으로 로봇 애플리케이션(지능)을 향상시킬 수 있다. CPU, GPU, 메모리 등의 로봇 리소스를 보완할 뿐만 아니라 글로벌 규모의 고가용성 서비스, 분산 애플리케이션 관리 솔루션, 개발 및 배포 등이 클라우드 컴퓨팅을 통해 제공될 수 있다[6].

ROS는 로컬 배포에서 개별 로봇 장치의 초점으로 설계되었다. 클라우드 컴퓨팅을 통한 PaaS기반의 배포 및 운영이 필수로 요구시 되는 점이며, 로봇의 제한적인 하드웨어로 인해 제한받는 연산 능력을 증가시키기 위해 SaaS기반의 연산이 요구시 된다.

특히, 산업계에서 로봇과 클라우드 컴퓨팅의 결합은 현재 PaaS 형상에서 잘 나타나고 있다. ROS의 제한적인 능력으로 인해 자연스럽게 구현된 생태계이다. PaaS는 기본 시스템 인프라에서 추상화하여 개발자가 애플리케이션 기능, 구축, 배포 및 런타임 시 애플리케이션을 플랫폼 구성 요소 및 서비스의 구성으로 관리 한다. PaaS는 소스 코드에서 빌드 및 배포 자동화, 자체 프로비저닝할 수 있는 사전 패키지 서비스 생태계에 대한 단순화된 액세스, 애플리케이션 상태 관리 기능 및 관리, 배포를 제공함으로써 개발 생산성에 엄청난 영향을 미친다. 또한, 모니터링 대시보드 등의 운영을 제공한다.

SaaS 형상으로 Cloud를 이용하는 Robot분야에서 로봇에 원격 두뇌를 제공한다는 아이디어는 오래전부터 시작되었다. Robot의 제한적인 하드웨어로 인해 연산이 불가능한 다양한 서비스를 Cloud가 대신 연산한다는 개념으로, Cloud에서 상황을 인지하고 필요한 Motion Control을 계산한 후, Robot에게 명령을 내려 Robot이 물리적인 제어을 수행하는 방식이다.

Fig. 2는[6] 로봇과 Cloud가 공동의 Brain을 사용하는 대표적인 예시이다. 로봇은 라이다를 이용해 공간 및 객체를 인지한 후, SLAM이라 불리는 기술을 통해 지도화 한다. 로봇에서 전체 공간을 지도화할 때 많은 연산 능력을 필요로 하여, Google 등에서 SLAM 연산을 Cloud로 제공한다.

DAvinCi 프로젝트 역시 Fig. 2와 동일한 기능을 구현한 프로젝트로 ROS를 메시징 프레임워크로 사용하여 Hadoop 클러스터로 데이터를 처리하고 FastSLAM 알고리즘을 병렬화 한다.

로봇이 SaaS를 통해 지능화 연산을 할 때, Cloud Bridge를 통해 필요한 서비스로 연결된다. Fig. 3은[6] 순찰 로봇과

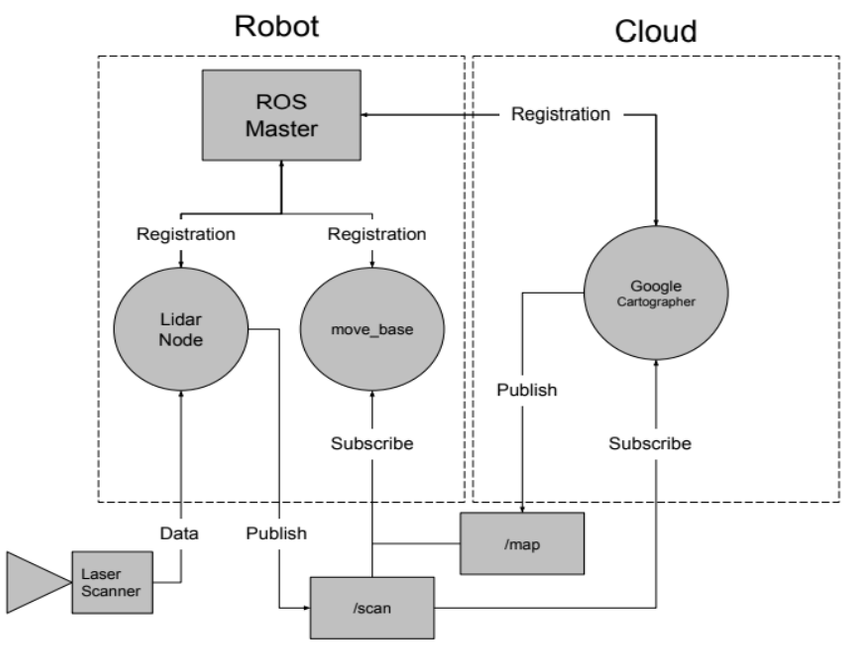

Fig. 2. ROS with Google Cloud

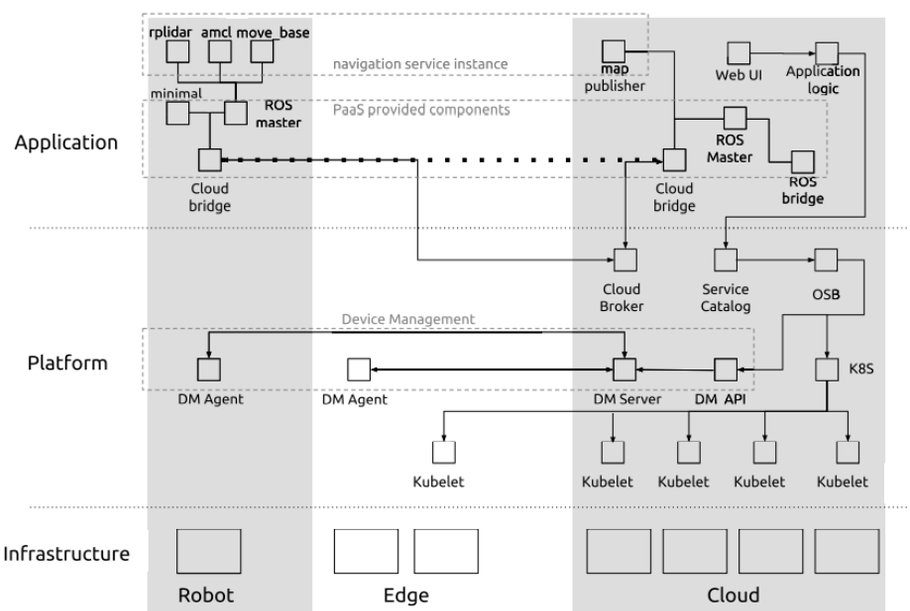

Fig. 3. ROS with Google Cloud

클라우드 연계 예시이다. 사용자는 지도에서 로봇을 운전하고 카메라에서 실시간 스트림을 볼 수 있는 웹 UI를 받는 구조이다. 로봇은 수동 또는 자율 탐색을 통해 하나 이상의 환경 지도를 구성한다. 애플리케이션의 핵심 기능인 웹 기반 지도 탐색, 비디오 스트리밍 등은 다양한 서비스로 구현되며, 인스턴스화는 클라우드와 선택한 로봇에 걸쳐 여러 상호 통신 구성 요소의 배포 및 실행을 한다. 즉, 사용자는 사용 가능한 로봇을 선택하여 서비스 네비게이션의 인스턴스를 생성할 수 있다. 서비스의 인스턴스화는 선택한 로봇(e.g. move-base 등)과 클라우드에서 필요한 ROS 노드를 시작하고, 클라우드 브릿지를 구성하여 지도와 탐색 병령 등이 로봇과 클라우드간 통신되도록 허용 한다.

즉, 애플리케이션을 인스턴스화 하고 브릿지를 기반으로 로봇과 클라우드를 연결하며, 서비스 카탈로그 기능을 이용하여 PaaS 또는 SaaS 애플리케이션을 사용한다.

2.3 Security Protocol in Cloud

ELK(Efficient Large-Group Key)는 계층적 트리를 이용하고, 부모노드의 키가 자식노드 키로부터 파생되며, 계층적 트리에서 키를 생성하고 조작하기 위해 PRFs (pseudo-random functions)를 이용하는 클라우드 권한 관리 프로토콜이다. LKH의 노드 그룹이 커질수록 키 관리 비용이 증가하는 문제점을 해결했지만 상호 인증이 불가하고 Relay attack, Replay attack 등 취약점이 존재한다.

LKH는 룹키를 중앙 집중식으로 관리하는 그룹키 관리 프로토콜로 클라우드에서 사용될 수 있다. 하지만 루트 노드가 모든 키를 저장해야 하고, 각 노드 멤버가 루트에 이르는 경로의 모든 키를 가지고 있어야 하는 문제점이 존재한다.

COGKTK는 Publisher/Subscriber System을 이용하는 Sensor-Cloud에 대한 보안 기법이다. Time Key를 이용해 키를 업데이트하는 특징을 가지고 있으며, 그룹키와 타임키를 결합하여 키 갱신을 수행한다. 하지만 오류 탐지가 되지 않아 오류가 발생 하였을 시 보안 제어가 불가능하며, 권한 관리를 지원하지 않는다.

SGIM은 이벤트 제어 보안에 특하된 클라우드 권한관리 프로토콜이다. 다만, 그룹키 갱신에 있어 전방 보안성이 취약하고 상호 인증을 지원하지 않는다.

3. 제안하는 Security Scheme

제안하는 Security Scheme은 Cloud Bridge(이하 브릿지), Service Application, Human으로 구성되어 있다. Robot과 Cloud Bridge는 동일한 Components로 취급하며, 각 객체간 인증서 등을 통한 인증 및 세션 보호는 마친 것으로 가정한다.

제안하는 Scheme에서는 사람이 로봇을 선택하고 필요한 서비스를 이용하고자 할 때 브릿지를 기반으로 로봇이 클라우드 서비스와 연결되었을 때의 정보보호를 목적으로 한다.

Table 2. Symbols

Bridge	Bridge between robot and SaaS
SaaS	Robot control application
Node	Robot control process
Ctrl.	Control type and level info.
Pub	Public Key
RN	Random number
Pri	Private Key
h()	Hash function
RK	Robot control Key
K	Key
Sign	Sign value using RSA
Ktime_app	Control Key for Node (in limit time)
Kctrl_type	Control Key for Node (for type&level)

브릿지와 서비스 애플리케이션간 정보 보호 시 특정 사람을 증명할 수 있는 Key를 이용하며, 해당 Key를 통해 사람이 로봇을 이용해 어떤 행위를 취했는지 포렌식을 위해 검증할 수 있다.

Fig. 4는 사용자가 제어 가능한 로봇의 권한, 즉 security level에 대한 Key와 로봇 제어 가능 시간 Key를 로봇 브릿지(이하 브릿지)에서 생성해 전송하는 프로토콜이다. 프로토콜의 symbol은 Table 2와 같다.

①, ②, ③ 브릿지와 애플리케이션, 사용자는 서로의 공개키를 요청하고 검증한다. 모두 CA를 통해 검증하였음을 가정한다.

④, ⑤ 사용자의 개인정보, 원천 데이터 취급 권한 등급 등이 애플리케이션과 브릿지에 확인 된다. 상기 과정은 Robot Platform을 통해 확인됨을 가정한다.

⑥ 브릿지는 사용자에게 사용자의 공개키로 암호화된 난수를 전송하며, 브릿지의 개인키로 난수를 서명하여 전송한다.

⑦ 사용자는 개인키로 브릿지의 난수를 복호화한 후, 난수를 생성하여 연접한다. 연접된 값은 브릿지의 공개키로 암호화되고 사용자의 개인키로 서명돼 브릿지에게 전송된다.

⑧ 브릿지와 사용자는 난수를 연접한 값과 각각의 공개키를 연접한 값을 XOR 연산한 후, 사전에 공유된 Hash 함수 연산을 수행해 4n bits의 Bit stream을 생성한다.

4n bits의 Bit stream은 각 n bits의 r0, r1, lr, ad으로 나뉜다. 이후, 브릿지는 전송되는 정보는 로봇 제어 가능 시간 Key RK를 lr를 연접해 RK'를 생성한다.

⑨ 사용자는 Challenge/Response 과정을 수행할 수 있도록 Challenge bits c를 생성하며, ad와 연접해 C를 생성한다. 이 때, ad를 연접하는 위치는 lr의 값에 따라 다르게 책정된다. lr의 bit 값이 0일 경우, ad는 c의 뒤에 연접하게 되며,

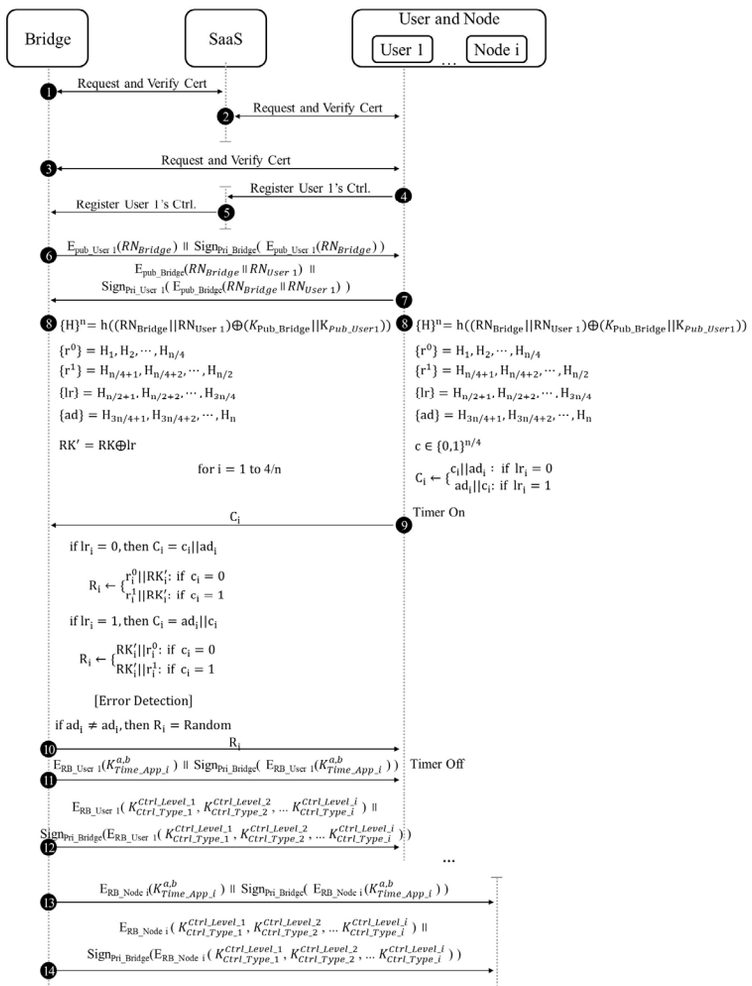

Fig. 4. Registration

위치하는 lr의 bit 값이 1일 경우, ad는 c의 앞에 연접하게 된다. 즉, lr을 n bits를 배열로 표현하였을 때, 각 첫 번째 주소부터 마지막 주소까지의 lr 값에 따라 ad의 위치가 달라진다. 사용자는 n bits의 Challenge bits C를 브릿지에게 전송한다.

⑩ 브릿지는 사용자의 Challenge bits C에 대한 Response 값 R을 전송한다. lr 값과 c 값에 따라 RB'값에 r0과 r1 값을 다르게 조합하며, Challenge/Response 과정을 통해 RK'가 사용자에게 전송된다. 이 때, 브릿지는 사용자로부터 전송된 C에서 ad의 위치 및 값을 검증하며 올바른 C가 전송되지 않을 경우, 난수를 생성해 전송한다.

⑪ 브릿지는 Challenge/Response 단계에서 전송과정으로서 사용되지 않은 r0과 r1로 구성된 잉여 비트 RB를 Seed로 새로운 암호화 키를 생성한다. AES 암호화 알고리즘을 사용할 경우 S1과 CH의 비밀 공유키는 Round Key로 사용될 수 있다.

⑫ 브릿지는 생성된 암호화 키를 이용해 전송되는 정보는, 로봇 제어 가능 시간 Key와 로봇 제어 권한 및 제어 권한 보안 레벨에 따른 접근 권한 키를 사용자에게 전송한다. 이 때, 브릿지의 서명 값이 같이 전송된다.

⑬ 사용자 또는 노드가 로봇을 제어할 수 있는 제한 시간과 관련한 Key가 브릿지로부터 전송된다.

⑭ 사용자의 권한 정보에 따라 로봇 제어 권한 타입 및 보안 레벨에 따른 접근 권한 키가 전송된다. 로봇 제어 시 권한과 보안 레벨에 맞게 움직임을 제어할 수 있으며 이때 해당 키를 사용하여 명령어를 암호화하여 전송한다.

Fig. 5는 애플리케이션, 클라우드 등 하나의 서비스 패키지를 위한 다양한 노드에서 로봇을 제어하고자 할 때 제어를 요청하는 type과 Security level에 따라 이벤트 암호화 키를 생성해 노드에게 전송한 후, 이벤트 암호화 키를 기반으로 로봇 제어를 수행하는 프로토콜이다. 이벤트 제어는 두 가지 종류의 원천 제어 타입을 지닌 노드를 이용하여 제어함을 가정한다.

즉, 단순히 팔을 굽히는 행위와 이동하는 행위가 원천 제어 타입이라면, 팔을 굽히며 이동해 무언가를 잡는 이런 복합적인 상황에 따른 행위를 이벤트 제어 행위로 본다. 로봇의 행위는 상황에 따라 그 필요성 또는 제어 가능여부를 검증해야

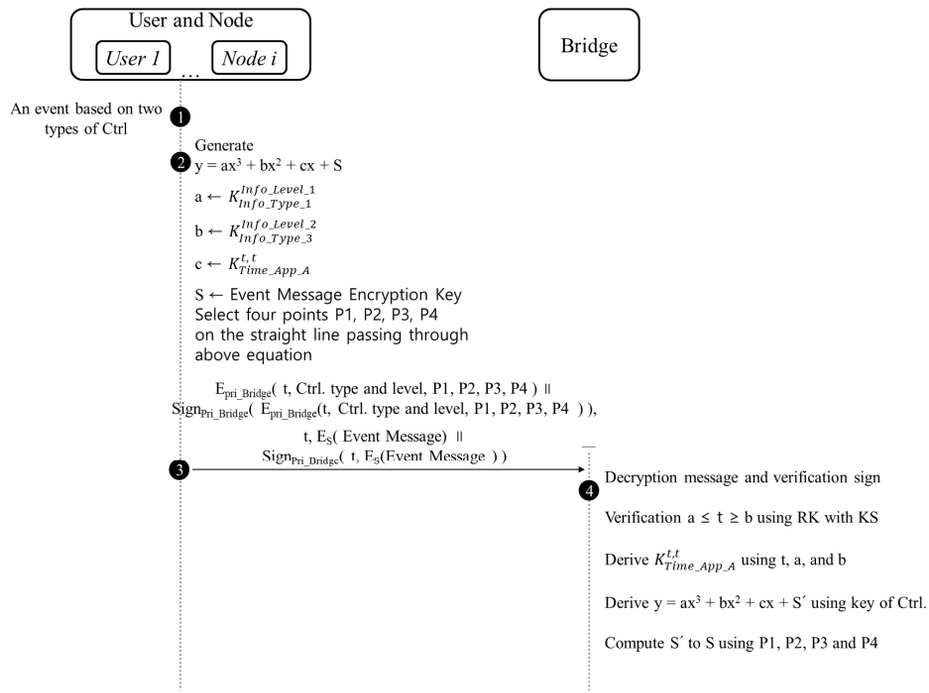

Fig. 5. Control

하므로 이를 이벤트 제어 키를 이용해 검증 및 기록 한다.

또한, 이때 로봇을 제어할 수 있는 시간 제한을 두며, 이를 Key를 이용해 검증할 수 있도록 한다. 이는 노드의 비이상 제어를 막기 위함이다.

① 두 개의 제어 타입과 보안 레벨을 가진 경우를 가정한다.

② 노드들 중 마스터 노드는 이벤트 제어 암호화 키를 생성한 후 해당 키를 비밀 상수항으로 하는 3차 다항식을 생성한다. 이는 2개의 타입을 예로 들었기 때문이며, 다항식은 타입의 수 + 1로 증가할 수 있다.

계수는 원천 제어 접근 권한 키와 이벤트 제어 메시지 생성 시간으로 구성된다. 이 후, 다항식을 기반으로 하는 그래프에서 지나는 점들을 선택한다. 노드의 수를 n으로 하였을 때, 다항식의 차수는 n+1이 되고 선택되는 점들의 수는 n+2가 된다.

③ 마스터 노드는 이벤트 제어 메시지 생성 시간, 원천 제어의 type과 Security level, 그래프에서 선택된 P값을 자신의 개인키로 암호화 및 서명하여 브릿지에게 전송한다. 또한, 이벤트 메시지를 동시에 전송한다.

④ 브릿지는 로봇 제어 가능 시간 Key를 서버에게 검증받으며, 이 때 사전에 생성 및 공유된 RK를 이용한다. 등록 시 브릿지가 설정한 a부터 b까지의 기간 안에 제어 이용 가능 기간 t와 제어 메시지 생성 시간 t가 존재한다면, Keyt,t를 유도할 수 있다. 브릿지는 Keyt,t와 전송받은 원천 제어 type과 security level에 따른 접근 권한 Key를 계수로 하는 다항식을 생성한다. 이 후, 브릿지는 생성된 다항식에 전송받은 P1, P2 등 다항식을 지나는 값들을 대입하여 상수값을 도출하며, 도출된 S를 Key로 하는 제어 관련 메시지를 복호화한다. 또한, S와 이벤트 제어 메시지를 구성하는 원천 제어 데이터 접근 키들을 해당 이벤트에 대한 디바이스 제어 접근 키 Seed 값으로 사용된다.

4. 보안성 평가

본 장에서는 제안하는 Security Scheme의 보안강도를 평가하고 비교분석한다.

Table 3은 기존 Cloud computing 보안 연구와 제안하는 프로토콜의 보안강도를 비교분석한 표이다.

객체간 난수를 생성한 후 정해진 Hash 함수 등을 거쳐 수정 후 사용되도록 설계하였다. 또한, 수정된 난수를 키로 사용하지 않고 Distance bounding 과정 중 사용되지 않은 값만을 키로 사용하여, 키의 강인성을 높였으며, 지속적으로 세션키 Seed 값이 변형되도록 설계하였다.

Distance bounding 과정을 통해 각 노드가 물리적으로 근거리에 있는지 판단하여 Relay attack에 안전하여, 마피아 공격은 2*(1/4)n의 확률을, 테러리스트 공격의 확률은 2*(1/4)n을 보여주고 있다.

비트스트림 연산을 통해 생성된 ad와 lr이 전송되는 2 bits 중 어디에 위치하는지 알고 있다. 이를 통해 Challenge bits에 대한 응답 비트 중 1 bit의 위치만 잘못되어도 Error detection이 가능하며, 이를 통해 forward security가 가능하다. 또한, Challenge bits에 대한 Error detection 역시 가능하다.

Table 3. Security Analysis

	ELK[7]	LKH[8]	CoGKTK[9]	sGIM[10]	Proposed Protocol
Distributed KS	Not-Support	Not-Support	Support	Support	Support
Reliable Key Update	Not-Support	Not-Support	Support	Support	Support
Forward Security	Not-Support	Not-Support	Support	Not-Support	Support
Error detection	Not-Support	Not-Support	Not-Support	Not-Support	Support
Mutual Authentication	Not-Support	Not-Support	Support	Not-Support	Support
Access Control	Not-Support	Not-Support	Not-Support	Not-Support	Support
Replay Attack	Not-Support	Not-Support	Secure	Secure	Secure

제안하는 프로토콜에서 평문으로 노출되는 값은 랜덤 값이 유일하며, 노출된 랜덤값은 Hash 연산 등을 통해 변형된다. 따라서 익명성을 보장한다.

5. 결 론

본 논문에서는 로봇을 사용하는 사용자의 정보를 보호하고, 사용자가 로봇을 어떻게 제어/이용했음을 Key를 통해 증명하는 방안을 제안하였다. 로봇은 시각기반 동적 지능적 제어가 중요하여, 상황에 따라 제어의 권한 등을 다시한번 확인해야할 필요성이 있는 환경이다. 이를 위해 이벤트 제어 Key를 이용하고, 제어 권한 및 보안 레벨에 따른 제어가 정당함을 Key를 통해 증명할 수 있도록 하였다. 보안성 분석을 통해 통신 프로토콜에서 발생 가능한 취약점에 안전함을 증명하였으며, 클라우드 기반 권한 및 키 관리 프로토콜들과 비교분석하였다. 로봇에서 가장 대표적인 운영체제인 ROS와 지능화 로봇을 위한 Cloud Robotics Platform을 고려하여 설계되어, 지능화 로봇에 안전한 Scheme으로 적용 가능하다. 향후 제안하는 논문을 기반으로 ROS기반 실험 논문을 진행할 예정이다.

References

[1] Cloud [Internet], http://en.wikipedia.org/wiki/Cloud_coomputing.

[2] T. Aho, et al., "Designing IDE as a service," Communications of Cloud Software, Vol.1, No.1, Dec. 2011.

[3] T. Mikkonen and A. Nieminen, "Elements for a cloud-based development environment: Online collaboration, revision control, and continuous integration," *Nordic Symposium on Cloud Computing and Internet Technologies*, pp.14-20, Aug. 2012.

[4] L. M. Gadhikar, L. Mohanv, M. Chaudhari, P. Sawant, and Y. Bhusara, "Browser based IDE to code in the cloud," *Advances in Intelligent Systems and Computing*, Vol.203, pp.59-69, 2013.

[5] ROS Nodes [Internet], https://www.ros.org.

[6] KOUBAA, Anis (ed.). "Robot Operating System (ROS): The Complete Reference (Vol.5)," Springer Nature, 2020.

[7] A. Penrig, D. Song, and J. Tygar, "ELK, a new protocol for efficient large-group key distribution," In *Proceedings 2001 IEEE Symposium on Security and Privacy, S&P 2001*, IEEE, pp.247-262, 2001.

[8] C. K. Wong, W. Mohamed, G. Simon, and S. S. Lam, "Secure group communications using key graphs," *IEEE/ACM Transactions on Networking*, Vol.8, No.1, pp.16-30, 2000.

[9] T. D. Nguyen and E. N. Huh, "An efficient Key management for secure multicast in Sensor-Cloud," *2011 First ACIS/JNU International Conference on Computers, Networks, Systems and Industrial Engineering*, pp.3-9, 2011.

[10] M. M. Hassan, B. Song, and E.-N. Huh, "A framework of sensor-cloud integration opportunities and challenges," *3rd International Conference on Ubiquitous Information Management and Communication*, 2009.

김 형 주

https://orcid.org/0000-0002-0346-4930

e-mail : k.hyungjoo@gmail.com

2008년 단국대학교 컴퓨터과학과(학사)

2010년 숭실대학교 컴퓨터학과(석사)

2015년 숭실대학교 컴퓨터학과(박사)

2016년 ~ 현 재 KT 융합기술원 책임연구원

관심분야 : Security, AI, Robot