

라즈베리파이에서 사물인터넷 기반의 인증 시스템 구현

(IoT based Authentication System Implementation on Raspberry Pi)

김 정 원¹⁾
(Kim Jeong Won)

요 약 정보기술의 발전에 따라 보안의 중요성은 매우 높아지고 있다. 기존의 보안 시스템은 대부분 고가이고 구현이 용이하지 않으며 생체 정보를 사용하는 경우 또한 매우 복잡하다. 본 논문에서는 지문과 얼굴 이미지를 사용하는 저가의 사물인터넷 기반 보안 단말기를 구현하여 이러한 점을 해결하고자 한다. 저가의 보안 시스템을 구현하기 위하여 라즈베리파이에 지문인식 스캐너와 카메라를 장착하고 스캔 이미지는 AES-256 알고리즘으로 암호화하여 클라우드에 전송한다. 본 연구를 통해 우리는 제안하는 시스템의 인증 서비스, 비용의 감소, 보안성, 그리고 확장성 측면에서의 가능성을 확인할 수 있었다.

핵심주제어 : 사물인터넷, 인증, 지문, 라즈베리파이

Abstract With the Development of Information Technology, Security is becoming very Important. Existing Security Systems are Mostly Expensive and Not Easy to Implement, and are Also very Complex when using Biometric Information. In this paper, We try to solve this Problem by Implementing a Low cost Internet based Security Terminal Using Fingerprint and Face Image. To Implement a Low-cost Security System, a Fingerprint Scanner and a Camera are installed in Raspberry pi, and the Scanned Image is encrypted with the AES-256 Algorithm and Transmitted to Cloud. Through This Study, We confirmed the Possibility of the Proposed System in view of Authentication, Cost Reduction, Security and Scalability.

Key Words : IoT, Authentication, Finger print, Raspberry Pi

1. 서 론

* Corresponding Author : jwkim@silla.ac.kr

Manuscript received June 20, 2017 / revised August 4, 2017/
revised October 14, 2017/ accepted November 23, 2017

1) 신라대학교 컴퓨터소프트웨어공학부

금융, 교육, 공공기관, 운송, 병원 등 거의 모든 분야에서 개인의 정보를 보호하거나 안전을 강화하는 수단으로서 신뢰성 있는 신원 확인 수단이 요구되고 있다. 이러한 정보 보호에 있어서 정보

의 이용 가능성, 통합성, 기밀성을 모든 형태에서 보장해야 하는데 지금까지 다수의 도구나 기법들이 제안되고 있다[1, 2]. 이러한 기법 중 지문, 홍채, 얼굴 등 생체 정보는 개인의 고유 정보이므로 인증 수단으로 널리 사용되고 있다. 개인 식별용으로 비밀번호나 ID 카드 등은 분실과 공유로 인해 취약하지만 생체 정보는 각 개인의 생물학적 특징을 사용하므로 한층 강화된 보안성, 신뢰성, 편리성 등을 제공할 수 있다. 따라서 생체 인식은 많은 분야에서 채택되어 운용 중에 있으며 모바일 단말에서도 채택되어 개인 인증 수단으로 사용되고 있다[3].

생체 인식의 높은 정확성과 편리한 사용에도 불구하고 생체 정보에 대한 접근성과 확장성은 미흡하고 대부분 고가이며 구현이 용이하지 않다는 점이 있다[4,5,6]. 이러한 문제를 해결하기 위해서 저가의 단말에서 생체 인식 시스템을 구현하거나 클라우드 컴퓨팅과 같은 환경을 고려할 수 있다. 일례로 아두이노와 라즈베리파이와 같은 저가의 하드웨어가 등장하여 사물인터넷 환경을 제공하고 있으며 클라우드 컴퓨팅 자원을 사용하여 확장성, 대량의 자료 처리, 그리고 방대한 저장공간과 컴퓨팅 파워를 제공한다. 따라서 본 논문에서는 이러한 저가 단말을 사용한 생체 인식 시스템을 구현하여 비용과 구현의 편리성 문제를 해결하고자 한다.

본 연구에서는 사물인터넷 환경을 제공하는 장치인 저가의 라즈베리파이에 지문 스캐너와 카메라를 장착한 지문 인식 시스템을 구현하였다. 이 라즈베리파이의 가격은 50달러 미만으로 개인용 컴퓨터 수준의 초소형 장치로서 신용카드 크기이며 공개 소스인 리눅스 계열의 운영체제가 탑재된다. USB 포트로 키보드, 마우스 등 다양한 입출력 장치를 연결할 수 있고 HDMI 포트를 통해 일반 모니터까지 사용할 수 있다. 최근 모델의 경우 WiFi 어댑터가 기본 장착되어 인터넷에 직접 연결될 수 있어 사물인터넷 장치로서 그 가능성이 매우 높다. 따라서 본 연구에서는 지문 스캐너 모듈을 라즈베리파이의 USB 인터페이스에 연결하여 지문을 스캔하며 스캔된 이미지는 암호화하여 관리 서버에 저장하고 인터넷에 연결 가능한 각종 모바일 단말기에서 원격으로 사용자 접근을 제어하고 확인할 수 있는 사물인터넷 기

반의 인증 시스템을 제안한다.

논문의 구성은 다음과 같다. 2장에서는 생체 인식에 관련된 최근 연구와 본 연구를 비교 분석하고 3장에서는 라즈베리파이에서 사물인터넷 기반의 지문인식 시스템을 제시하고 4장에서는 다양한 실험을 통해 제안 시스템의 성능을 분석한다. 마지막으로 5장에서는 결론 및 향후 과제를 제시한다.

2. 관련 연구

본 장에서는 제안 기법과 관련된 기존 연구들을 소개하고 제안 시스템과의 차별성을 분석한다. 먼저 비밀번호 기반의 인증은 현재까지도 많은 기법들이 제시되었는데 스니핑, 피싱, 키로그 공격 등 취약한 보안성에도 불구하고 널리 사용되고 있다. 이 취약성을 해결하기 위해 HCI 기반의 기법, 비밀번호의 유형과 보안성에 대한 기법들이 제시된 바 있다[7, 8]. 얼굴, 지문, 목소리, 홍채, DNA, 그리고 서명 등 생체 자료에 대하여 등록, 갱신, 검증, 복제, 성능 등의 기준에 따라 각각의 보안성을 MRTD(Machine readable travel document) 시스템에 적용한 결과를 제시하고 있다[9]. 라즈베리파이를 사용한 기법에는 이미지 캡처 장치를 라즈베리파이에 장착하여 보안 단말로 사용하는 기법[8]과 대부분의 생체 인증 시스템이 개인용 컴퓨터로 이루어지고 있고 크기, 무게, 전력 소비 등으로 제한적 사용이 문제인데 이러한 점을 극복하기 위해 지문과 족문의 특징 추출을 OpenCV를 사용하여 구현한 기법 등이 있다[11]. 또한 클라우드 컴퓨팅을 사용하는 사례도 있는데 라즈베리파이를 등록 단말로 사용하고 인증 등의 복잡한 알고리즘은 클라우드 자원을 사용하여 이식성, 확장성, 경제성을 제공하는 기법[12]과 클라우드에 얼굴 인식 시스템을 구현하여 클라우드를 사용하지 않는 시스템과 클라우드 기반의 생체 인식 시스템과의 차별성을 분석한 기법 등이 있다[13].

상기와 같이 라즈베리파이를 사용한 다수의 기법들이 제시되고 있는데 제안 시스템은 라즈베리파이 자체가 하나의 서버로 동작하며 제안 시스템이 인증에 적용될 때 효율성과 성능에 대한 보

다 상세한 분석을 제공하고자 한다. 라즈베리파이는 일반적으로 230 Mbps 의 계산처리 능력과 비트당 3nJ/bit 에너지 소모량을 보이는데 본 연구에서는 CPU 및 메모리 사용률을 분석하여 지문인식 단말, 그리고 생체 인식용 사물인터넷 서버로서 가능성을 분석하였다.

3. 사물인터넷 기반의 지문인식 시스템의 구현

본 연구의 목적은 클라우드 서비스, 출입문 개방, 개인정보 보호 서비스를 원격으로 제공하는 시스템을 저가의 단말로 구현하여 인증 목적의 모든 응용에서 제안 시스템의 가능성을 제시하는 것이다. 3 장에서는 제안하는 시스템의 하드웨어와 소프트웨어 구성을 소개한다.

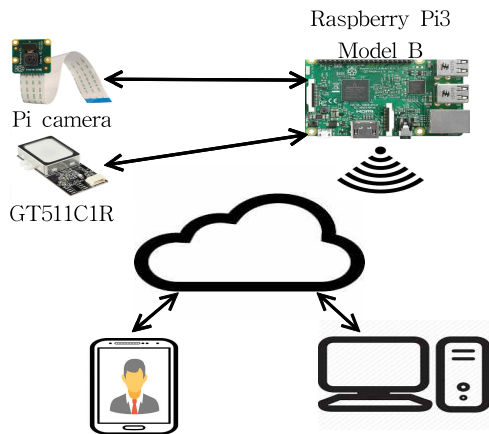


Fig. 1 Structure of the Proposed System

3.1 제안 시스템의 구조

제안 시스템은 생체 인식 대상으로 지문과 얼굴을 사용하는데 저가의 라즈베리파이에 지문 스캐너와 카메라를 구성한다. 사용된 라즈베리파이는 RPI 3 모델 B로서 35달러 정도이며 1.2GHz 64-bit quad-core ARMv8 CPU, 802.11n Wireless LAN, 4개의 USB 포트, HDMI 디스플레이 포트, Bluetooth 4.1, Bluetooth Low Energy (BLE), 40 GPIO 포트, Ethernet 포트, Camera Interface, Display interface, VideoCore

IV 3D 그래픽 코어로 구성되어 있는데 사물인터넷 단말로서 충분한 성능을 갖추고 있다.

지문인식을 위한 모듈은 Fingerprint Scanner Device GT(511C1R)로서 SmackFinger 3.0 알고리즘을 기반으로 하고 72MHz 의 32비트 CPU를 장착하여 20장의 지문 이미지를 저장할 수 있는 공간을 제공한다. 또한 360 도 방향 전환이 가능하며 통신 프로토콜은 UART 이고 JST-SH 커넥터로 라즈베리파이와 연결되는데 RS-232 시리얼 인터페이스로 호스트 단말과 지문 이미지를 전송한다. 이 모듈은 패킷 핸드셰이킹으로 통신 프로토콜이 구현되는데 명령 패킷, 응답 패킷, 그리고 데이터 패킷의 세 가지로 구성된다.

Fig. 1에서 보듯이 라즈베리파이에 지문 스캐너와 카메라가 장착되어 있고 와이파이로 인터넷에 연결된다. 라즈베리파이 단말에는 LCD 가 부착되어 지문 등록, 검색, 삭제, 얼굴 이미지 캡처 등이 이루어지고 접촉 기록은 라즈베리파이 단말 내의 서버에 기록이 되고 별도의 서버에 백업이 된다. 또한 라즈베리파이는 IP 주소를 가진 인터넷 단말로서 원격의 모바일 단말기에서 접근 기록을 확인할 수도 있다.

3.2 지문 인식 모듈

다음은 제안 시스템의 핵심 모듈인 지문 인식 과정을 설명한다. 이 과정은 크게 지문의 등록, 인식, 그리고 삭제로 구성된다.

3.2.1 지문 등록

지문 등록 과정은 3번의 스캔으로 이루어지는데 지문스캔모듈의 내부 메모리에 3번의 스캔 자료를 합쳐서 저장한다. 지문 스캔이 시작되면 LED가 켜지고 지문이 입력되기를 3초간 대기한다. 압력신호가 전달되면 스캔한 이미지를 지문스캔 모듈내의 메모리에 저장한다. 이 과정을 3번 반복하여 하나의 이미지로 병합하게 된다. 또 지문이 등록되기 전에 ID 값이 이미 등록되어 있는지 검사하고 없다면 20개의 ID 중에서 사용하고 있지 않은 맨 처음 ID 에 지문을 등록한다. 만약 이미 등록되어 있다면 에러를 반환하고 등록과정을 취소한다.

3.2.2 지문 인식

지문인식은 먼저 LED를 켜는 동작으로 시작한다. 라즈베리파이는 손가락이 스캐너에 위치하기를 기다리는데 성공하면 지문을 캡처하는 명령을 실행한다. 지문이 정상적으로 캡처되었으면 스캐너 모듈에 Identity 명령어를 전송한다. 스캐너 모듈내의 메모리에 해당 지문이 있으면 지문에 할당된 ID 값을 반환하고 만약 존재하지 않으면 에러값을 전송한다. 상기의 과정이 완료되면 LED를 끈다. Fig. 2는 스캔 지문과 내부의 이미지를 1:1 또는 1:N 비교하는 순서도로서 먼저 지문을 캡처(SenCapture)하여 지문이미지인지 검사한다(SenIsFinger). 지문이면 알고리즘에 의해 특징점을 추출하고(SenGetFeature) 내부 메모리에 저장된 이미지와 1:1 또는 1:N 비교를 수행한다(VerifyFpdata/IdentifyFpdata). 만약 매칭되는 이미지가 있으면 ID 값을 저장하여 라즈베리파이로 전송한다(SaveFpdata).

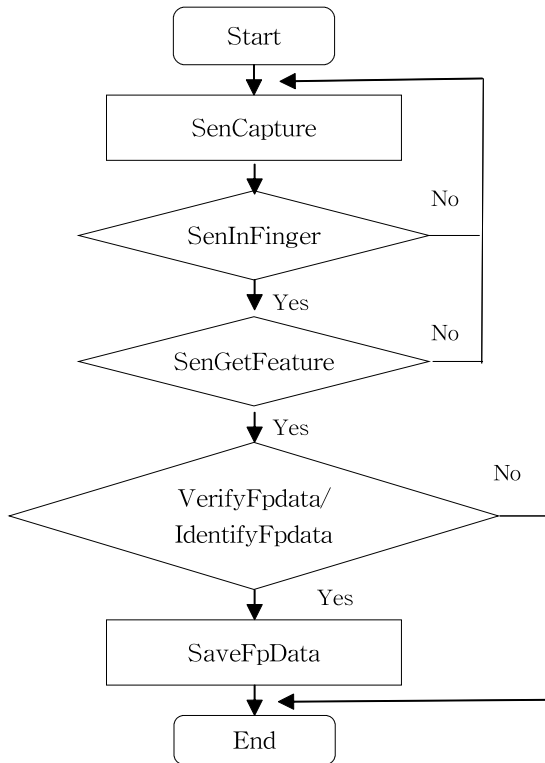


Fig. 2 Finger Print Searching Steps

3.2.3 지문 삭제

지문 삭제는 하나를 삭제하는 경우와 전체를 삭제하는 경우로 나누는데 지문 스캐너 모듈로 'Delete' 명령어 전송으로 지문 삭제가 이루어진다. 이 명령어를 전송받은 스캐너 모듈은 내부 메모리에 저장된 해당 지문을 삭제한다. GT511C1R은 'deleteID' 또는 'deleteAll' 명령어를 전송받는데 이 명령어에 따라 ID에 해당하는 하나의 지문을 삭제하거나 전체를 삭제하게 된다. 또한 라즈베리파이에 있는 데이터베이스에서도 ID에 해당하는 정보를 모두 삭제한다.

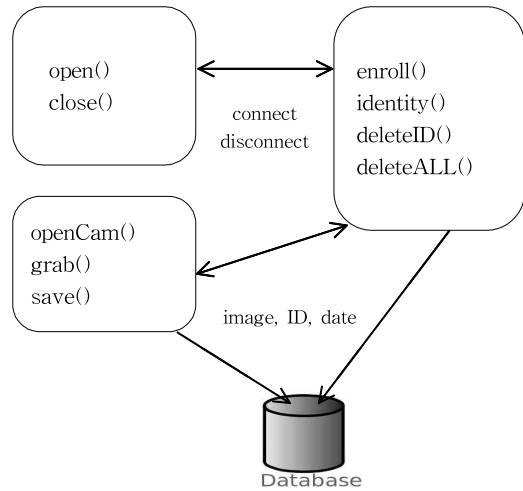


Fig. 3 Software Structure of Finger Print Recognition

3.3 지문인식 소프트웨어

본 연구에서는 지문인식 모듈의 동작성과 응용의 예로 Fig. 3과 같이 프로그램을 구성하였다. 라즈베리파이가 리눅스 기반이므로 Qt/Embedded를 사용하여 GUI를 개발하였다. Fig. 4에서 보듯이 개발한 GUI는 지문 스캐너 장치를 열고 닫는 모듈, 실제 지문을 등록, 검색, 삭제하는 모듈, 파이 카메라를 제어하는 모듈, 그리고 접근 기록을 관리하고 서비스하는 서버 모듈로 구성된다. 구현한 GUI는 지문과 사진을 표시하는 이미지뷰, 각종 정보를 출력하는 텍스트뷰, 스캐너 연결, 스캐너 연결종료, 지문 등록, 지문 검색, 지문

삭제, 사진찍기 등의 버튼들로 구성된다. 지문 인식스캐너와 라즈베리파이는 USB to Serial 로 연결되는데 '/dev/USB0' 디바이스 파일로 명령과 데이터가 전송된다. 지문 등록, 검색, 삭제 등의 핵심 기능은 C++ 로 구현되었으며 Qt GUI 컴포넌트와 연결되어 각종 기능을 수행한다. 파이 카메라 모듈은 C++ 로 개발하여 얼굴이미지를 캡처하여 GUI 이미지 뷰, 그리고 데이터베이스에 저장하고 원격에서 확인할 수 있는 기능을 제공한다.

3.4 지문인식 관리 서버

```
var userSchema = new Schema({
  name: String, // user name
  face: Buffer, // encrypted face image
  access_date: { type: Date, default: Date.now }
});
```

Fig. 4 MongoDB schema

```
var net = require('net');
var server = net.createServer(function(client) {
  client.setEncoding('utf8');
  client.on('data', function(data) {
    writeData(client, data.toString());
  });
  client.on('end', function() {
    server.getConnections(function(err, count) {
      console.log('Remaining Connections: ' + count);
    });
  });
});
server.listen(8107, function() {
  server.on('close', function(){
    console.log('Server Terminated');
  });
  server.on('error', function(err) {
    console.log('Server Error: ', JSON.stringify(err));
  });
});
function writeData(socket, data) {
  //write User's connection data to MongoDB;
}
```

Fig. 5 Server code (Node.js)

라즈베리파이가 지문인식을 통하여 출입문을 제어하는데 출입 기록을 남기고 원격에서 관리하기 위해 우분투 서버에 오픈스택 클라우드를 구축하였다. 구성한 서버는 일반 PC(intel i5, 8GB RAM, 500GB HDD) 두 대에 컨트롤러 노드와 컴퓨터 노드를 각각 구축하고 Node.js 로 소켓 서버를 구축하였다. 구축한 서버는 라즈베리파이의 지문인식 소프트웨어와 소켓 통신을 통하여 출입기록을 전송한다. 이를 위해 Node.js 서버는 클라이언트로부터의 입력을 처리하는 이벤트 리스너를 Fig. 4와 같이 구현하였다. 이 리스너는 비동기적으로 동작하며 데이터베이스에 접근 기록을 저장한다. 클라우드에 구축한 데이터베이스는 MongoDB를 사용하였으며 사용자 스키마는 Fig. 5 와 같이 사용자 이름, 얼굴의 이미지 데이터, 그리고 접근 시각 정보로 구성된다.

4. 실험결과

본 장은 IoT 기반의 인증 시스템의 구현 결과에 대하여 설명한다. 먼저 라즈베리파이는 지문 인식시스템의 게이트웨이로서 지문 등록 및 인증 노드인 동시에 사용자 정보에 대한 데이터베이스를 유지하여 원격에서 모니터링 할 수 있는 서비스를 제공한다. 제안 시스템은 별도의 서버를 유지하지 않으며 라즈베리파이 자체가 IP 기반의 노드로서 지문 등록 및 인증, 그리고 외부에 정보를 제공하는 역할도 수행한다. 따라서 저렴한 비용으로 시스템을 구축할 수 있다.

본 연구의 목적은 저렴한 비용으로 지문인식을 통한 보안시스템을 구축하는 것이므로 지문 스캐너, 사용자 정보 저장용 DBMS, 그리고 지문인식 서버를 라즈베리파이가 안정적으로 운영하는지 확인하는 것이다. 따라서 라즈베리파이의 CPU, 메모리 등을 아래 Table 1과 같은 쉘 스크립트를 작성하여 시스템이 동작할 때 측정하였다.



Fig. 6 Qt/Embedded based GUI

Table 1 Shell Script for Performance Evaluation

parameter	shell script
RAM	<code>free -h grep -v + > /tmp/ramcache</code>
Disk	<code>df -hl grep 'Filesystem\ /dev/sda*' > /tmp/diskusage</code>
Load average	<code>loadaverage=\$(top -n 1 -b grep "load average:" awk '{print \$10 \$11 \$12}')</code>
System uptime	<code>tecutime=\$(uptime awk '{print \$3,\$4}' cut -f1 -d,)</code>

Fig. 6에서 보듯이 지문 등록이나 인증시 얼굴과 지문 이미지를 스캔한다. 이 생체 정보는 단순 이미지로 저장하면 네트워크 전송이나 공격에 노출될 수 있으므로 AES-256 알고리즘으로 암호화한다. 이 알고리즘은 암호화에 사용되는 키와 복호화에 사용되는 키가 동일하다는 특징이 있으며 비밀키 암호라고도 한다. 이 대칭키 방식은 연산 속도가 빨라서 효율적인 암호 시스템을 구축할 수 있다. 본 연구에서는 256비트 키 방식 기반으로 이미지가 암호화 및 복호화 된다. Table 2 는 얼굴과 지문 이미지를 암호화할 때 소요시간, CPU 그리고 메모리 사용량을 보여준다. 스캔한 이미지는 각각 2,592 x 1,944, 240 x 216 해상도를 가지는 RGB 이미지이다.

Table 3은 본 논문에서 사용하고 있는 AES-256 알고리즘이 RPI에서 만족한 성능을 보이고 있는지 확인하기 위해서 우분투 리눅스가

Table 2 Loads for Recognition

	fingerprint	face
size(bytes)	52,918	2,657,969
Encryption time(ms)	200	1,670
CPU (%)	1%	2.2%
Memory	52KB	2.6MB

설치된 PC(인텔 i5-3470, 3.2GHz, 4GB 메인메모리)에서 동일한 조건하에서 실험한 결과와 비교한 것이다. 실험은 RPI에 장착된 카메라가 촬영한 이미지 100장을 대상으로 100번 실험을 실행하여 평균값을 계산한 것이다. Table 3에서 보듯이 데스크 탑용 CPU가 장착된 PC 가 암호화 시간, CPU 시간 등의 파라미터에서 두 배 이상의 성능을 보이고 있고 메모리는 비슷한 용량을 사용하고 있다. 제안하고 있는 시스템의 성능은 PC 환경에 비해 낮지만 사용자가 감내할 수준을 보이고 있으므로 RPI 에서도 AES-256 알고리즘으로 안전하게 생체 정보를 전송할 수 있음을 확인하였다.

Table 3 AES-256 runtime comparison with PC environment

	Desktop(intel i5)	RPI
Encryption time(ms)	823	1,669
CPU (%)	1.05%	2.02%
Memory	2.5MB	2.6MB

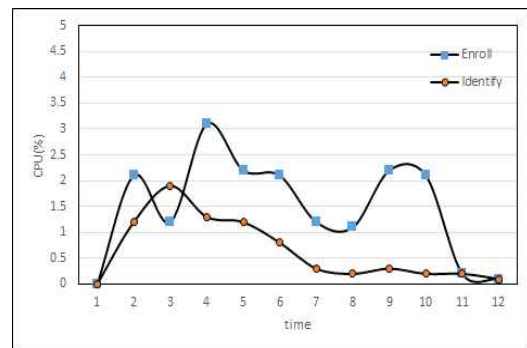


Fig. 7 CPU Usage

Fig. 7, Fig. 8은 지문 등록과 검색시 CPU와 메모리 사용률을 나타낸다. 먼저 Fig. 7은 라즈베리파이가 스캐너 모듈과 통신할 때 CPU 사용량

을 보여주는데 CPU 사용율은 거의 10% 이하를 보여주고 있는데 이것은 라즈베리파이가 서버 모듈 실행시 부하가 크지 않다는 것을 나타낸다. Fig. 8은 성능 측정 실험 동안의 메모리 사용량을 보여주는데 지문 등록시 지문 스캐너 모듈이 지문을 캡처해서 자신의 내부 메모리에 저장하고 이 이미지를 라즈베리파이의 내부 MySQL 에 저장하는데 메모리 사용량은 0.12% 증가한다. 이것으로 보아 메모리 사용량은 크지 않은 것으로 판단된다. 따라서 Fig. 7, Fig. 8의 결과를 분석해 보면 제한된 하드웨어 성능을 가진 라즈베리파이 가 지문 스캐너, 데이터베이스 서버, 그리고 지문 인식 서버 등을 안정적으로 운영할 수 있음을 보여준다.

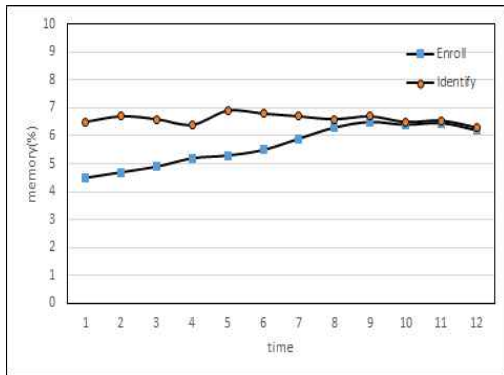


Fig. 8 Memory Usage

5. 결론

본 논문은 저가의 라즈베리파이에 지문 인식 스캐너를 장착하여 지문인식 기반의 보안 시스템을 제안하였다. 기존의 생체인식 시스템은 대부분 고가이고 구현이 쉽지 않은 경우가 많지만 제안 시스템은 저가이면서 비슷한 성능을 보이는 인증 시스템을 구현하였다. 구현한 시스템은 지문 스캐너 모듈을 통해 지문을 스캔하고 파이카메라를 통해 사용자 접근시 얼굴 이미지를 캡처하여 접근 기록을 저장하도록 클라우드에 서버를 구성하였으며 스캔 이미지는 AES-256 알고리즘으로 암호화하여 전송하므로 뛰어난 보안성도 제공할 수 있다. 제안 시스템은 라즈베리파이가 지

문을 스캔하고 지문과 얼굴이미지를 저장하고 인터넷을 통해 접근 기록을 서비스하는 서버 역할을 수행하므로 가격대비 성능이 우수한 것으로 판단된다. 시스템 성능 평가를 위해 CPU, 메모리 부하량을 측정하고 결과 제안한 시스템은 지문 인식 보안시스템으로서 충분한 성능을 제공하였다. 향후 연구로는 다수의 지문 인식 단말을 지원하는 보안시스템을 구성하기 위한 시스템의 설계, 보안성을 강화하기 위한 암호화 기법의 추가, 그리고 지문뿐만 아니라 얼굴 인식을 클라우드 환경에서 제공하는 것이다.

References

- [1] Anil K. J., Karthik N., and Abhishek N., "Biometric Template Security," Journal on Advances in Signal Processing Volume, Article ID 579416, 2008.
- [2] Debnath B., Rahul R., Farkhod A., and ChoiM., "Biometric Authentication: A Review," International Journal of e-Service, Science and Technology, Vol. 2, No. 3, pp. 13-27, September, 2009.
- [3] <http://biometrics.gov/Documents/Glossary.pdf>, National Science and Technology Council's (NSTC) Subcommittee on Biometrics, Biometrics Glossary, 2006.
- [4] Lee, S. H., Kim. J. Y., and Kim, S. Y., "Implementation of Indoor Environment Monitoring and Automatic Control System based on Internet of Things," Journal of the Korea Industrial Information Systems Research, Vol. 21, No. 6, pp. 71-80, 2016.
- [5] Park, J. S., Hong, S. G., and Kim, N. R., "A Development Plan for Co-creation-based Smart City through the Trend Analysis of Internet of Things," Journal of the Korea Industrial Information Systems Research, Vol. 21, No. 4, pp. 67-78, 2016.
- [6] Choi, Y. H., and Kim, Y. R., "Research Regard to Necessity of Smart Water

- Management Based on IoT Technology,” Journal of the Korea Industrial Information Systems Research, Vol. 22, No. 4, pp. 11-18, 2017.
- [7] Sasse M. A., Brostoff S., and Weirich D., “Transforming the weakest links human/computer interaction approach to usable and effective security,” BT technology, Journal, Vol. 19, No. 3, pp. 122-131, 2001.
- [8] Yan, A. F., Blackwell, R. J., Anderson, and Grant A., “Password memorability and security: Empirical results,” IEEE Security & privacy, Vol. 2, No. 5, pp. 25-31, 2004.
- [9] Dhannawat R., Sarode T., and Kekre H. B., “Kekre’s Hybrid Wavelet Transform Technique with DCT, Walsh, Hartley And Kekre Transforms for Image Fusion,” IJCET, Vol. 4, Issue 1, pp. 195-202, January-February 2013.
- [10] Senthilkumar G., Gopalakrishnan K., Kumar V.S., “Embedded Image Capturing System Using Raspberry Pi System,” International Journal of Emerging Trends & Technology in Computer Science, Vol. 3, issue 2, pp. 213-215, April 2014.
- [11] Sivaranjani S., and Sumathi S., “Implementation of Fingerprint and Newborn Footprint Feature Extraction on Raspberry Pi,” IEEE Sponsored 2nd International Conference on Innovations in Information Embedded and Communication Systems ICIECS, pp. 328-336, 2015.
- [12] Shah, D. K., Bharadi, V .A., Kaul, V. J., and Amrutia, S., “End-to-End Encryption Based Biometric SaaS: Using Raspberry Pi as a Remote Authentication Node,” IEEE sponsored 1st International Conference on Computing, Communication, Control, and Automation (ICCUBEA), pp. 52-59, 2015.
- [13] Bharadi V. A. and DSilva, G. M., “Online Signature Recognition Using Software as a Service (SaaS) Model on Public Cloud,” International Conference on Computing, Communication, Control and Automation, pp. 65-72, 2015.



김 정 원 (Kim Jeong Won)

- 정회원
- 부산대학교 전자계산학과 학사
- 부산대학교 대학원 전자계산학과 석사
- 부산대학교 대학원 전자계산학과 박사
- 현재 : 신라대학교 컴퓨터소프트웨어공학부 교수
- 관심분야 : 클라우드 컴퓨팅, 내장형시스템, 멀티미디어, 운영체제, 모바일 컴퓨팅