*Journal of The Korea Institute of Information Security & Cryptology* ISSN 1598-3986(Print)
VOL.28, NO.6, Dec. 2018 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2018.28.6.1509

# 인증 수준에 의한 서비스 접근제어 프로토콜*

유 성 민,[1†] 최 석 진,[1] 박 준 후,[2] 류 재 철[2‡]
[1]ETRI부설연구소, [2]충남대학교

# POSCAL : A Protocol of Service Access Control by Authentication Level*

SeongMin Yoo,[1†] SeokJin Choi,[1] JunHoo Park,[2] Jae-Cheol Ryou[2‡]
[1]The Affiliated Institute of ETRI, [2]Chungnam National University

요 약

본 연구의 목적은 다양한 형태의 사용자 정보를 이용해 다양한 서비스에서 유연한 인증 기능을 사용할 수 있도록 지원하는 것이다. 모든 서비스에 동일한 수준의 인증을 요구하기 보다는, 사용자 인증 시점에 인증 수준을 파악하여, 동적으로 권한을 부여함으로써 편리성 및 효율성을 증가시키는데 목적이 있다. 이를 위해 다양한 로컬 인증정보를 바탕으로 인증 수준에 따라 서비스 접근을 제어할 수 있는 POSCAL (A Protocol of Service Access Control by Authentication Level) 프로토콜을 제안하고 이를 이용한 인증 프레임워크를 설계하였다. 인증 프레임워크의 기능 검증을 위해 POSCAL 인증 프레임워크 기반의 전자지갑 서비스를 개발하였고, 유즈케이스(Use Case) 시나리오를 바탕으로 구현 기능을 평가했다. 제안한 프로토콜은 사용자 및 메시지 인증, 인증정보의 기밀성, 인증 내역의 무결성, 인정 내역에 대한 부인방지, 보안 요구수준에 따른 서비스별 접근제어를 만족한다.

ABSTRACT

The purpose of this study is to support flexible authentication functions in various services using various types of user information. Rather than requiring the same level of authentication for all services, the goal is to identify the level of authentication at the time of user authentication and to increase convenience and efficiency by dynamically granting authority. In this paper, we propose POSCAL (Protocol of Service Control by Authentication Level) protocol which can control service access based on various local authentication information. To verify the function of the authentication framework, we developed the electronic wallet service based on the POSCAL authentication framework and evaluated the implementation function based on the use case scenario. The proposed protocol satisfies user and message authentication, confidentiality of authentication information, integrity of authentication history, non - repudiation of authorization, and access control by service according to security level.

**Keywords:** Authentication Protocol, Access Control, Biometrics, FinTech, e-Wallet, Authentication Framework

## I. 서 론

### 1.1 연구 배경 및 필요성

2013년 이후, 지문인식 모듈이 탑재된 스마트폰들이 대중화되었고, 최근에는 홍채인식 모듈까지 탑재된 스마트폰들도 출시되고 있다. 이처럼 다양한 형태의 생체인식 모듈의 인식률이 높아지고 소형화되면서 스마트폰과 같은 모바일 기기에 탑재되고, 이를 이용하는 다양한 서비스에 활용되고 있다. 시장조사

Received(11. 07. 2018), Modified(11. 30. 2018), Accepted(12. 03. 2018)
* 본 연구는 방위사업청과 국방과학연구소가 지원하는 국방위성항법특화연구센터 사업의 일부 지원으로 수행 되었습니다.
† 주저자, mingoon@nsr.re.kr
‡ 교신저자, jcryou@home.cnu.ac.kr(Corresponding author)

전문업체 와이즈가이리포트(Wise Guy Reports)에서는 2018년부터 2023년까지 글로벌 생체인증 시장의 연평균 성장률(CAGR)이 16.3%에 이를 것으로 전망했으며, 앞으로는 스마트폰뿐만 아니라 웨어러블 기기에서도 다양한 형태의 생체인증 정보들이 수집되어 인증에 활용될 것으로 예상되고 있다[1][2][3][4].

한편 핀테크(FinTech)가 세계 금융시장과 IT 시장의 화두가 되면서, 안전하고 편리한 인증의 중요성이 강조되고 있다. 핀테크 기술이 편리성, 용이성, 저비용의 장점이 있더라도 보안에 취약하다면 금전적인 손실이 발생하고, 그로 인해 사용자들이 핀테크 서비스를 이용하지 않을 수 있기 때문이다[5][6].

따라서 안전한 인증 기술은 신뢰할 수 있는 핀테크 서비스의 운영을 위해 필수적으로 요구되는 보안기술이다. 뿐만 아니라 시스템 내부적으로 처리되는 암호화, 부정거래 탐지 시스템, 신뢰실행환경 등의 보안기술과는 다르게 인증은 사용자가 직접 패스워드, 지문 등의 인증정보를 입력해야 하는 과정이 있기 때문에 편리성과도 밀접한 연관이 있다. 이에 따라 안전하면서도 편리한 인증의 중요성이 더욱 커지고 있다.

### 1.2 연구 목표 및 범위

본 연구의 목적은 다양한 형태의 사용자 정보를 이용해 다양한 서비스에서 유연한 인증 기능을 사용할 수 있도록 지원하는 것이다. 모든 서비스에 동일한 수준의 인증을 요구하기 보다는, Fig. 1.과 같이 사용자 인증 시점에 인증 수준을 파악하여, 동적으로 권한을 부여함으로써 편리성 및 효율성을 증가시키는데 목적이 있다.

이를 위해 사용자 단말에서, 생체기반의 인증정보뿐만 아니라 기존의 패스워드와 같은 지식기반의 인증정보, 그리고 보안토큰과 같은 소지기반의 인증정보 등을 포함한 다양한 형태의 인증정보들을 기반으로 하는 사용자 또는 메시지 인증 기능을 다양한 핀테크 서비스에 제공할 수 있는 범용의 인증 프레임워크를 제안하는 것을 목표로 한다. 이러한 인증 프레임워크는 다음과 같은 요구사항이 고려되어야 한다.

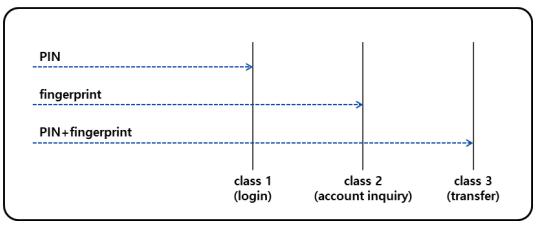


Fig. 1. Concept of service access control by authentication level

- 사용자 및 메시지 인증

사용자 인증은 사용자의 자격을 확인하고 검증하는 것을 의미하며, 메시지 인증은 전송된 메시지가 변조되지 않고 정당한 사용자로부터 온 것임을 확인하는 것을 의미한다. 핀테크 서비스에서 인증 기능은 사용자의 신분을 증명하고, 거래 내역을 인증하기 위해 기본적으로 제공되어야 한다.

- 인증정보의 기밀성

기밀성은 정보에 접근하도록 허락된 사람에게만 그 정보에 접근할 수 있도록 보장하는 것을 의미한다. 인증 과정 중에 사용자가 본인임을 증명하거나, 거래 정보와 같은 메시지가 올바름을 증명하기 위해 제공한 생체정보, 패스워드 등의 비밀정보들이 제3자에게 노출되지 않도록 기밀성을 만족해야 한다.

- 인증 내역의 무결성

무결성은 권한이 없는 사람에 의해 정보가 불법적으로 변조되는 것을 방지하는 것을 의미한다. 인증 과정 중에 전송되는 사용자의 인증정보나 메시지가 제3자에 의해 변조되는 것을 방지 또는 탐지할 수 있도록 무결성을 만족해야 한다.

- 인증 내역에 대한 부인방지

부인방지는 데이터의 송수신자가 송수신 사실에 대해 부인할 수 없도록 방지하는 것을 의미한다. 부인방지는 금융거래에 있어서 중요한 기능의 하나로, 사용자가 상품 결제나 자금 이체와 같은 메시지를 송신했을 경우, 이에 대해 부인할 수 없도록 부인방지성을 만족해야 한다.

- 보안 요구수준에 따른 서비스별 접근제어

접근제어는 사용자에게 허용된 권한에 맞게 데이터나 서비스를 제공할 수 있도록 제어하는 것을 의미한다. 핀테크 서비스에는 계좌조회, 자금이체 등 다양한 서비스가 존재하며, 각각의 서비스는 서로 다른 보안 수준을 요구한다. 따라서 인증 기능을 다양한 서비스에 범용적으로 사용하기 위해서 인증 과정에서 발생하는 멀티모달 인증정보들을 기반으로 사용자의 권한을 설정하고 서비스 별로 접근을 제어할 수 있는 기능을 제공할 필요가 있다.

### 1.3 논문 구성

본 논문의 구성은 다음과 같다. 먼저 2장에서는 관련 연구로서 간편결제 서비스에서 사용되는 인증 기술을 살펴보고, 온라인 환경에서 생체인식 기술을 활용한 인증방식에 대한 기존 연구 및 표준 동향을 분석한다.

3장에서는 본 논문에서 제안하는 다양한 로컬 인증 정보를 바탕으로 인증 수준에 따라 서비스 접근을 제어할 수 있는 POSCAL (A Protocol of Service Access Control by Authentication Level) 프로토콜을 설명하고, 이를 이용한 POSCAL 인증 프레임워크를 설계한다. 제안하는 인증 프레임워크는 로컬 인증(Local Authentication)과 원격 인증(Remote Authentication)을 분리한 기본적인 인증 구조와 공개키 인증서를 이용한 전자서명 기반의 인증 프로토콜을 특징으로 한다. 이를 통해 앞서 정의한 요구사항 중에서 사용자 및 메시지 인증, 인증정보의 기밀성, 인증 내역의 무결성, 인증 내역에 대한 부인방지, 보안 요구수준에 따른 서비스별 접근제어를 만족한다.

4장에서는 제안한 POSCAL 인증 프레임워크를 구현하고 기능, 안전성, 성능에 대해 평가한다. 인증 프레임워크의 기능이 정상적으로 동작하는지를 검증하기 위한 킬러앱(Killer App)으로써 전자지갑 서비스(S-WALLET)를 POSCAL 인증 프레임워크를 이용해 개발하고, 유즈케이스(Use Case) 시나리오를 바탕으로 구현 기능을 평가한다. 또한 앞서 정의한 요구사항을 바탕으로 안전성에 대해 논한다.

마지막으로 5장에서는 결론을 맺고 향후 추가적으로 진행될 수 있는 연구 주제에 대한 방향을 제시한다.

## II. 관련연구

간편결제에서 간편인증으로 사용되는 사용자 인증 수단은 서비스 사업자마다 독자적으로 구현되는데, PIN이나 패스워드가 기본적으로 사용되며, 그 외에도 대부분 패턴인증이나 그래픽인증과 같은 지식기반의 인증수단이 주로 사용된다. 삼성이나 애플과 같은 스마트폰 제조사는 스마트폰 제조 시에 자체적으로 지문인식 모듈을 장착하고, 독자적인 API를 이용해서 삼성페이나 애플페이와 같은 지문인식을 간편인증 수단으로 제공하는 간편결제를 서비스하고 있다[7][8].

구글의 안드로이드 운영체제는 2015년 10월 발표한 6.0 마시멜로(Marshmallow)에서부터 운영체제 단에서 지문인식 API를 제공하면서 스마트폰 제조사가 아닌 일반 IT 기업에서도 지문인식을 인증수단으로 사용하는 간편결제 서비스가 출시될 것으로 예상된다[9].

공인인증서를 간편결제의 사용자 인증수단으로 사용하기 위한 연구들도 진행되고 있다. 공인인증서는 비밀정보인 개인키가 단순한 패스워드로 암호화되어 단말에 저장되기 때문에 안전한 관리가 어렵고, 개인키 저장 및 사용 시에 발생 가능한 보안위협을 막기 위해서 액티브엑스를 이용해 보안 프로그램을 설치하는 등 편의성을 저해하는 문제점을 지적받아 왔다[10].

이에 따라 패스워드가 아닌 생체인증 정보로 인증서를 보호하는 많은 연구들이 진행되고 있다. Jin-gyu Beom[11]는 지문인식 모듈 내부에 개인키를 안전하게 저장하고, 지문인식을 통해 개인키에 대한 접근을 통제하기 위한 8가지 운용방안을 제안하였다. Guillermo Martınez-Silva[12]는 모바일 단말의 지문인식 모듈을 이용해 X.509v3 인증서를 생성할 수 있는 Mobile Certification Authority(MCA)를 제안하였다. Sunghyuck Hong[13]는 X.509 인증서의 확장필드에 생체정보를 저장하고, 이를 이용해 정보보증 기능을 제공할 수 있는 방법을 제안하였다. Han-Ul Jang[14]은 PKI와 지문정보 템플릿을 연계해서 사용자를 인증할 수 있는 방법을 제안하였다. Andrew Burnett[15]은 생체정보로부터 랜덤 스트링을 추출할 수 있는 Fuzzy extractor를 이용해 개인키와 공개키를 생성하고 인증에 사용하는 방법을 제안하였다.

FIDO(Fast IDentity Online)는 온라인 간편인증에 관한 오픈스펙 표준으로 온라인 인증 프로토콜뿐만 아니라 플랫폼, 인증 모듈 등의 표준 스펙도 정의하고 있으며, FIDO 2.0은 W3C의 웹 브라우저 인증 표준으로 제정되었다. FIDO 1.0은 ID/PW 대신 생체정보를 이용하는 UAF(Universal Authentication Factor) 프로토콜과 ID/PW와 함께 별도의 인증장치를 이용해 2차 인증을 수행하는 U2F(Universal 2nd Factor) 프로토콜로 구성되어 있으며, FIDO 2.0은 브라우저에 탑재되어 인증장치와 인증을 수행하는 CTAP(Client To Authenticator Protocol) 프로토콜로 구성되어 있다[16][17].

## III. POSCAL : A Protocol of Service Access Control by Authentication Level

이 장에서는 본 논문에서 제안하는 POSCAL (A Protocol of Service Access Control by Authentication Level) 프로토콜을 설명하고, 이를 이용한 POSCAL 인증 프레임워크를 설계한다. 이를 위해 POSCAL의 기본 인증구조와 주요 프로토콜 및 메시지 구조를 설계한다. 다음으로 POSCAL 인증 프레임워크를 이용한 응용서비스를 개발하여 서비스 접근제어 기능을 검증한다. 또한 앞서 정의한 인증 프레임워크의 요구사항을 바탕으로 안전성에 대해 논한다.

### 3.1 기본구조

POSCAL 인증 프레임워크의 구조적인 기본 개념은 Fig. 2.와 같이 로컬 인증(Local Authentication)과 원격 인증(Remote Authentication)을 분리함으로써 사용자의 인증정보에 대한 기밀성을 보장하는 것이다. 이러한 기본구조는 스마트폰에서 지원하는 지문, 홍채 등의 다양한 인증모듈을 이용한 로컬 인증을 통해 사용자를 우선 검증하고, 로컬 인증 결과를 근거로 스마트폰이 서버에 대신 사용자의 신원을 보증해줌으로써 원격 인증에서 나타날 수 있는 다양한 보안 위협을 해결할 수 있다. 즉, 네트워크를 통해 사용자의 패스워드나 지문정보와 같은 비밀정보가 유출되는 것을 방지할 수 있으며, 서버에는 사용자의 비밀정보를 저장하지 않기 때문에 서버에서의 보안 위협과 개인정보 보호 이슈로부터 안전해진다.

원격 인증 프로토콜은 ISO/IEC 9798-3-2 표준에서 제시한 단방향 2-way 인증 프로토콜을 기반으로 한다. 이 표준은 서버에서 랜덤값을 보내면, 클라이언트는 랜덤값을 자신의 개인키로 서명해서 보내는 질의응답(Challenge-Response) 방식으로 인증을 수행한다[18]. 즉, 인증 과정이 시작되면 POSCAL 서버가 POSCAL 클라이언트로 어떠한 질의를 하고, POSCAL 클라이언트는 거기에 맞는 응답을 한다.

이 과정에서 POSCAL 클라이언트는 공개키 암호 시스템의 기능 중 하나인 전자서명과 공개키 인증서를 이용해 응답하기 때문에 사용자 인증뿐만 아니라, 인증정보 및 메시지에 대한 무결성과 부인방지성을 보장할 수 있다.

### 3.2 인증 동작 과정

POSCAL의 인증 과정은 Fig. 3.과 같다. (1) 먼저 사용자 단말의 서비스 클라이언트(응용프로그램 또는 브라우저 등)에서 서비스 서버에 어떠한 서비스를 요청한다. (2) 해당 서비스 서버는 API를 통해 POSCAL 서버에 인증을 요청하고, (3) POSCAL 서버는 인증 요청 메시지를 생성하여 서비스 서버를 통해 서비스 클라이언트로 전달한다. 이때 전달하는 인증 요청 메시지에는 서비스 서버가 제시하는 인증 정책이 포함되며, 이 인증 정책에는 사용자에게 요구하는 인증 레벨과 그 인증 레벨을 얻기 위한 방법이 정의되어 있다.

(4) 서비스 클라이언트는 API를 통해 POSCAL 클라이언트에 인증 요청 메시지를 전달한다. (5) POSCAL 클라이언트는 로컬 인증을 수행한 후에 인증 응답 메시지를 생성하고, 서비스 클라이언트를 통해 서비스 서버로 전달한다. 이때 전달하는 인증 응답 메시지에는 사용자가 단말에서 수행한 로컬 인증 목록이 포함된다.

(6) 서비스 서버는 API를 통해 POSCAL 서버에 인증 응답 메시지를 전달하고, (7) POSCAL 서버는 인증 응답 메시지를 검증한 후에 해당 결과를 서비스 서버에 알려준다. (8) 서비스 서버는 사용자의 인증 레벨을 계산하고, 결과에 따라 서비스를 제공하거나

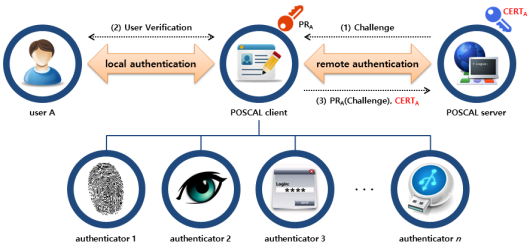


Fig. 2. Basic architecture of POSCAL

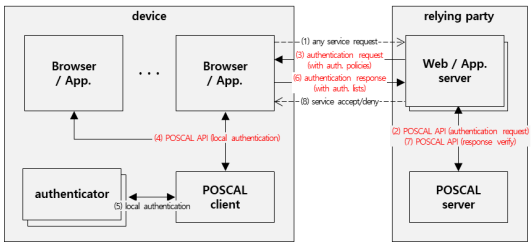


Fig. 3. Authentication process of POSCAL

거부한다.

### 3.3 메시지 구조

인증 과정에 필요한 POSCAL 프로토콜은 AUTH_REQ, AUTH_RESP의 두 가지 메시지로 구성된다.

#### 3.3.1 AUTH_REQ

AUTH_REQ 메시지는 사용자가 인증이 필요한 서비스를 요청했을 때, POSCAL 서버가 사용자에게 인증을 요구하기 위해 보내는 메시지이다. AUTH_REQ 메시지 타입을 ASN.1(Abstract Syntax Notation number One)으로 표기하면 Table 1.과 같은 구조를 갖는다.

Table 1. Data structure of AUTH_REQ

```
AuthREQ ::= SEQUENCE {
    versionEXPLICIT  Version DEFAULT v1,
    userID           PrintableString,
    appID            IA5String,
    challengeValue   BIT STRING,
    authReqItems     AuthReqItems,
    suggestPolicies  SuggestPolicies
}

Version ::= INTEGER { v1(0), v2(1), v3(2) }

AuthReqItems ::=
  SEQUENCE SIZE (1..MAX) OF AuthReqItem

AuthReqItem ::= SEQUENCE {
    authReqItemType  AuthReqItemType,
    authReqItemBody  ANY OPTIONAL,
    reqAuthLevel     INTEGER
}

AuthReqItemType ::= INTEGER
   { UserAuth(0), UserReg(1), MsgAuth(2) }

SuggestPolicies ::=
   SEQUENCE SIZE (1..MAX) OF SuggestPolicy

SuggestPolicy ::= SEQUENCE {
    authnrList       AuthnrList,
    admissionLevel   INTEGER,
    comments         PrintableString
}

AuthnrList ::=
     SEQUENCE SIZE (1..MAX) OF Authnr

Authnr ::= SEQUENCE {
    majorType  INTEGER,
    minorType  INTEGER,
    authnrOID  OBJECT IDENTIFIER OPTIONAL
}
```

- version은 POSCAL 프로토콜의 버전을 나타낸다.
- userID는 서비스를 요청한 사용자의 식별자를 나타낸다.
- appID는 사용자가 요청한 서비스의 식별자(URI)를 나타낸다.
- challengeValue는 서버에서 생성한 임의의 값으로 AUTH_RESP 메시지의 재사용 공격을 방지하기 위한 목적으로 사용된다.
- authReqItems는 서버에서 인증을 요청한 항목들을 나타낸다.
- suggestPolicies는 서비스 서버에서 제시하는 인증 정책으로 여러 개의 suggestPolicy가 포함될 수 있다.
- authReqItemType은 authReqItem의 타입으로 0은 사용자 인증, 1은 서비스 가입, 2는 메시지 인증을 의미한다.
- authReqItemBody는 선택적인 필드이며, authReqItem의 본문으로 서버에서 인증을 요구하는 메시지이다. authReqItemBody가 없을 경우 단순한 사용자 인증을 나타낸다.
- reqAuthLevel은 해당 authReqItem 인증을 위해서 서비스 서버가 요구하는 인증 레벨을 나타낸다.
- authnrLlists에는 서비스 서버에서 정책으로 허용하는 인증 모듈의 타입에 대한 목록이 포함된다.
- admissionLevel은 authnrList에 지정되어 있는 모든 인증 모듈로 로컬 인증을 수행했을 때 획득할 수 있는 인증 레벨을 나타낸다.
- comments는 해당 정책에 대한 서비스 서버의 설명을 나타낸다.
- majorType과 minorType은 해당 인증 모듈의 형태를 구분하기 위한 식별자를 나타낸다.
- authnrOID는 해당 인증 모듈을 나타내는 OID(Object IDentifier) 객체 식별자를 나타낸다.

#### 3.3.2 AUTH_RESP

AUTH_RESP 메시지는 사용자가 POSCAL 서버로 보내는 AUTH_REQ에 대한 응답 메시지이다. AUTH_RESP 메시지 타입을 ASN.1으로 표기하면 Table 2.와 같은 구조를 갖는다.

- userCERT는 사용자의 인증서를 나타낸다.
- originAuthResp는 ALOHA 클라이언트가 생성한 응답메시지의 원문을 나타낸다.
- signatureAlgorithm은 전자서명에 사용한 알고리즘을 나타낸다.

Table 2. Data structure of AUTH_RESP

```
AuthRESP ::= SEQUENCE {
    userCERT            BIT STRING,
    orignAuthResp       OriginAuthResp,
    signatureAlgorithm  AlgorithmIdentifier,
    signatureValue      BIT STRING
}

OriginAuthRespBody ::= SEQUENCE {
    version             EXPLICIT Version DEFAULT v1,
    userID              PrintableString,
    appID               IA5String,
    challengeValue      BIT STRING,
    authRespItems       AuthRespItems
}

Version ::= INTEGER { v1(0), v2(1), v3(2) }

AuthRespItems ::=
 SEQUENCE SIZE (1..MAX) OF AuthRespItem

AuthRespItem ::= SEQUENCE {
    authRespItemType    AuthRespItemType,
    authRespItemBody    ANY OPTIONAL,
    respAuthnrs         RespAuthnrs
}

AuthRespItemType ::=
   INTEGER { UserAuth(0), UserReg(1), MsgAuth(2) }

RespAuthnrs ::=
   SEQUENCE SIZE (1..MAX) OF RespAuthnr

RespAuthnr ::= SEQUENCE {
    majorType       INTEGER,
    minorType       INTEGER,
    authnrOID       OBJECT IDENTIFIER OPTIONAL
}
```

- signatureValue는 originAuthResp의 전자서명 값을 나타낸다.
- version은 ALOHA 프로토콜의 버전을 나타낸다.
- userID는 서비스를 요청한 사용자의 식별자를 나타낸다.
- appID는 사용자가 요청한 서비스의 식별자(URI)를 나타낸다.
- challengeValue는 서버로부터 받은 임의의 값을 나타낸다.
- authRespItems는 authReqItems 인증 요청에 대응하는 응답 항목들을 나타낸다.
- authRespItemType은 인증을 요청받았던 authReqItem의 타입으로 0은 사용자 인증, 1은 서비스 가입, 2는 메시지 인증을 의미한다.
- authRespItemBody는 선택적인 필드이며, 인증을 요청받았던 authReqItem의 본문으로 서버에서 인증을 요구한 메시지이다. authRespItemBody가 없을 경우 단순한 사용자 인증을 나타낸다.
- respAuthnrs에는 해당 authRespItemBody의 인증을 위해 수행된 로컬 인증 모듈의 목록이 포함된다.
- majorType과 minorType은 해당 인증 모듈의 형태를 구분하기 위한 식별자를 나타낸다.
- authnrOID는 해당 인증 모듈을 나타내는 OID 객체 식별자를 나타낸다.

### 3.4 메시지 처리

#### 3.4.1 AUTH_REQ 메시지 처리

AUTH_REQ 메시지의 타입은 단순 사용자 인증 요청과 메시지 인증 요청으로 구분되며, 각각의 처리 과정은 Fig. 4., Fig. 5.와 같다.

(1) 서비스 클라이언트로부터 API를 통해 사용자 인증 요청을 받은 경우, (2) 사용자가 POSCAL 클라이언트에 등록한 로컬 인증 모듈 목록이 표시되고, (3) 사용자는 등록한 모든 또는 일부의 로컬 인증 모듈에서 로컬 인증을 수행한다. (4) POSCAL 클라이언트는 로컬 인증 결과를 바탕으로 (5) AUTH_RESP 메시지를 생성하고, (6) 서비스 클라이언트로 전달한다. (7) 또한 AUTH_REQ에 포함된 서비스 서버의 인증 정책을 바탕으로 현재 사용자의 로그인 등급을

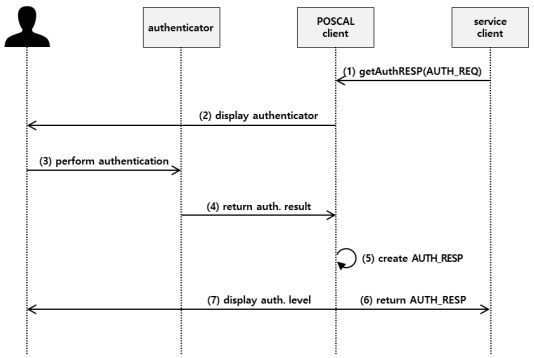


Fig. 4. Message processing of AUTH_REQ (User Authentication)

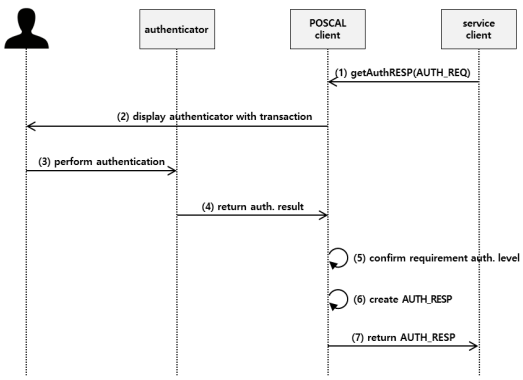


Fig. 5. Message processing of AUTH_REQ (Transaction Authentication)

알려준다.

(1) 메시지 인증 요청을 받은 경우, (2) 사용자가 POSCAL 클라이언트에 등록한 로컬 인증 모듈 목록과 함께 인증할 메시지가 표시되고, (3) 사용자는 등록한 모든 또는 일부의 로컬 인증 모듈에서 로컬 인증을 수행한다. (4) POSCAL 클라이언트는 로컬 인증 결과와 (5) AUTH_REQ에 포함된 서비스 서버의 인증 정책을 바탕으로 메시지의 인증 요구 등급을 검증하고, (6) AUTH_RESP 메시지를 생성해서 (7) 서비스 클라이언트로 전달한다.

#### 3.4.2 AUTH_RESP 메시지 검증

서비스 서버에서 서비스 클라이언트로부터 수신한 AUTH_RESP 메시지를 검증하는 과정은 Fig. 6.과 같다. (1) 먼저, 서비스 서버는 서비스 클라이언트로부터 AUTH_RESP 메시지를 받으면, (2) 해당 메시지를 POSCAL 서버 API를 통해 전달한다.

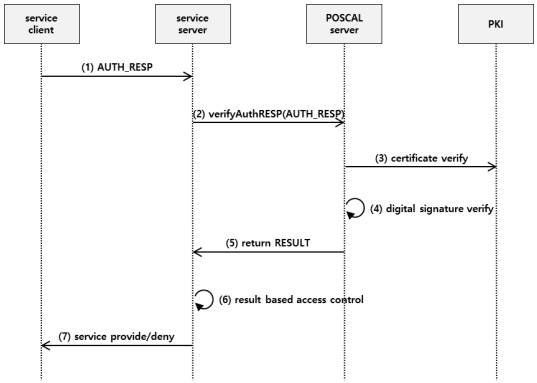


Fig. 6. Message verification of AUTH_RESP

(3) POSCAL 서버는 수신한 AUTH_RESP에 포함된 인증서를 PKI로부터 검증하고, (4) 다음으로 전자서명 값을 검증한다. (5) 서비스 서버는 AUTH_RESP 메시지의 검증 결과를 수신하면, (6) 인증정보와 인증정책을 바탕으로 사용자의 권한을 확인하고 (7) 서비스를 제공한다.

### 3.5 서비스 접근제어

#### 3.5.1 로컬 인증 모듈 분류

POSCAL 프로토콜 AUTH_RESP 메시지의 respAuthnrs 필드에는 사용자가 로컬 인증에 성공한 모듈의 정보가 저장되며, 이 필드는 하나 이상의 사용자 로컬 인증 정보를 수용할 수 있다. 따라서 서비스 서버에서는 이 정보들을 바탕으로 사용자의 인증 등급을 정의하고, 요청 서비스의 보안 요구 수준에 따라 접근을 제어하는 것이 가능하다.

인증 정보는 일반적으로 지식기반, 소지기반, 생체기반의 세 가지 유형으로 분류된다[19][20]. 그리고 각각의 유형 별로 다양한 인증수단이 존재한다[21]. 이를 바탕으로 Table 3.과 같이 로컬 인증 모듈을 인증 정보의 형태에 따라 major type과 minor type의 2단계로 분류할 수 있다.

또한 객체 식별자(OID, Object IDentifier)를 이용하면, 각각의 인증 모듈의 제조사, 모델명까지 식별하는 것이 가능하다. OID는 국제표준화기구(ISO)와 국제전기통신연합(ITU) 등이 정보보안 암호코드, 속성, 알고리즘 등 모든 사물과 객체를 세계적으로 하나의 고유번호로 구별하기 위해 마련한 국제표준 식별체계다[22]. 따라서 인증 모듈에 OID를

Table 3. Example of local authentication type classification

| majorType | minorType | OID |
|---|---|---|
| Knowledge base(0) | ID/PW (1) | OPTIONAL |
| | PIN (2) | |
| | Pattern (3) | |
| | others (0) | |
| Possession base(1) | Security card (1) | |
| | OTP token (2) | |
| | HSM (3) | |
| | others (0) | |
| Inherence base(2) | Fingerprint (1) | |
| | Iris (2) | |
| | Vein (3) | |
| | others (0) | |

부여하면, 개별 인증 모듈에 대한 화이트리스트나 블랙리스트를 만들어 세부적인 인증 정책을 적용할 수 있다.

위의 로컬 인증 모듈 분류 테이블을 ASN.1으로 표기하면 Table 4.과 같은 구조를 갖는다.

Table 4. Data structure of local authentication type classification

```
AuthnrList ::=
      SEQUENCE SIZE (1..MAX) OF Authnr

Authnr ::= SEQUENCE {
    majorType   INTEGER,
    minorType   INTEGER,
    authnrOID   OBJECT IDENTIFIER OPTIONAL
}
```

### 3.5.2 서비스 접근제어 정책 설정

각각의 서비스 서버 별로 개별적인 접근제어 정책을 적용할 수 있도록 하기 위해서 정책 테이블은 서비스 서버 내부에 위치해서 관리되며, AUTH_REQ 메시지의 suggestPolicies 필드에 포함되어 서비스 클라이언트로 전송된다.

Table 5.는 서비스 서버에서 정의하는 인증 모듈에 따른 인증 등급을 정의한 정책 테이블의 예이다. 하나의 로컬 인증 모듈 또는 여러 인증 모듈을 조합하여 등급을 정의할 수 있다[21].

이 등급별 로컬 인증 모듈 정책 테이블을 ASN.1

Table 5. Example of local authentication policy

| Lv. | Local auth. module | Remark |
|---|---|---|
| 1 | PIN | Knowledge base |
| | Includes 2nd level auth. modules | Includes all higher level auth. modules |
| 2 | Fingerprint | Inherence base |
| | Includes 3rd level auth. modules | Includes all higher level auth. modules |
| 3 | PIN+Fingerprint | Knowledge + Inherence base |
| | Includes 4th level auth. modules | Includes all higher level auth. modules |
| 4 | Iris | Inherence base |

으로 표기하면 Table 6.과 같은 구조를 갖는다.

또한 서비스 서버에서는 서비스 별로 요구되는 인증 등급을 정의할 수 있으며, 이를 이용해 인증 등급에 따른 서비스 접근제어가 가능하다[21][23]. 서비스 서버에 로그인하는 사용자의 인증 등급에 따라 적절한 서비스를 제공할 수 있고, 메시지 인증 시에는 AUTH_REQ 메시지의 authReqItem.reqAuthLevel 필드에 해당 메시지의 인증 요구 등급을 설정하여 사용자에게 인증을 요청할 수 있다. Table 7.은 서비스 서버에서 정의하는 서비스 접근제어 정책 테이블의 예이다.

위의 서비스 접근제어 정책 테이블을 ASN.1으로 표기하면 Table 8.과 같은 구조를 갖는다.

위와 같은 로컬 인증 모듈 분류 정보와 서비스 접근제어 정책 정보를 활용하여, AUTH_REQ를 수신하는

Table 6. Data structure of local authentication policy

```
ServicePolicies ::=
  SEQUENCE SIZE (1..MAX) OF SecervicePolicy

ServicePolicy ::= SEQUENCE {
    authnrList        AuthnrList,
    admissionLevel    INTEGER,
    comments          PrintableString
}

AuthnrList ::=
      SEQUENCE SIZE (1..MAX) OF Authnr

Authnr ::= SEQUENCE {
    majorType   INTEGER,
    minorType   INTEGER,
    authnrOID   OBJECT IDENTIFIER OPTIONAL
}
```

Table 7. Example of service access control policy

| Service | Required lv. | Service Type |
|---|---|---|
| Join | 1 | Registration(1) |
| Secession | 4 | User auth.(0) |
| Log-in | 1 | User auth.(0) |
| Account inquiry | 2 | User auth.(0) |
| Transfer(less than 300,000 won) | 3 | Message auth.(2) |
| Transfer(over 300,000 won) | 4 | Message auth.(2) |
| Transfer history inquiry | 2 | User auth.(0) |
| Payment(less than 300,000 won) | 3 | Message auth.(2) |
| Payment(over 300,000 won) | 4 | Message auth.(2) |

Table 8. Data structure of service access control policy

```
AuthReqItems ::=
   SEQUENCE SIZE (1..MAX) OF AuthReqItem

AuthReqItem ::= SEQUENCE {
     authReqItemType   AuthReqItemType,
     authReqItemBody   ANY OPTIONAL,
     reqAuthLevel      INTEGER
}

AuthReqItemType ::=
  INTEGER { UserAuth(0), UserReg(1), MsgAuth(2) }
```

POSCAL 클라이언트는 policies 필드를 참조하여, 로그인과 같은 사용자 인증 시에는 사용자의 인증 등급을 확인할 수 있다. 또한 메시지 인증 시에는 해당 메시지를 인증하기 위한 최소한의 인증 등급을 확인할 수 있으며, 이를 이용해 POSCAL 클라이언트 단에서 서비스 별로 접근을 제어하는 것이 가능하다.

## IV. 구현 및 평가

이 절에서는 본 논문에서 제안하는 POSCAL 인증 프레임워크를 구현하고, 이를 이용한 응용서비스를 개발하여 프레임워크의 기능을 검증한다. 또한 앞서 정의한 인증 프레임워크의 요구사항을 바탕으로 안전성에 대해 논한다.

### 4.1 POSCAL 프레임워크 기반의 응용 서비스 구현 및 평가

POSCAL 프레임워크는 그 자체로 어떠한 서비스를 제공하는 것이 아니라, 보안을 필요로 하는 응용 서비스에서 다양한 인증정보들을 바탕으로 간편하고 안전한 사용자 및 메시지 인증을 사용할 수 있도록 기능을 제공한다. 따라서 제안한 POSCAL 프레임워크의 기능을 평가하기 위해서는 POSCAL 프레임워크의 인증 기능을 활용하는 응용 서비스를 개발하고, 그 기능이 잘 동작하는지 확인할 필요가 있다.

#### 4.1.1 응용 서비스 개요

주요 핀테크 서비스 중 하나인 전자지갑은 스마트폰에 신용카드, 멤버십카드, 전자화폐 등을 담아두고 결제·관리하는 전자지불 시스템을 말한다. 사용자 및 메시지 인증은 전자지갑의 가장 중요한 요소기술이기 때문에 본 논문에서는 POSCAL 프레임워크를 활용한 전자지갑을 구현하여 프로토콜의 기능을 검증한다. POSCAL 프레임워크를 사용하는 전자지갑 응용 서비스 S-WALLET은 서비스 가입, 탈퇴, 로그인, 잔액조회, 그리고 자금이체 및 자금이체 내역 조회 등의 기본 기능을 제공한다. S-WALLET 서비스의 접근제어 정책은 Table 7.과 같으며, 등급별 인증 모듈 정책은 Table 5.와 같다.

#### 4.1.2 구현 기능 평가

- POSCAL 클라이언트 사용자 등록

사용자가 POSCAL 프레임워크를 사용하는 S-WALLET 서비스에 가입하기 위해, 먼저 POSCAL 클라이언트에서 사용자 등록을 하고 개인키와 인증서를 발급받는다.

- S-WALLET 서비스 가입

S-WALLET 서비스 가입을 위해서는 스마트폰

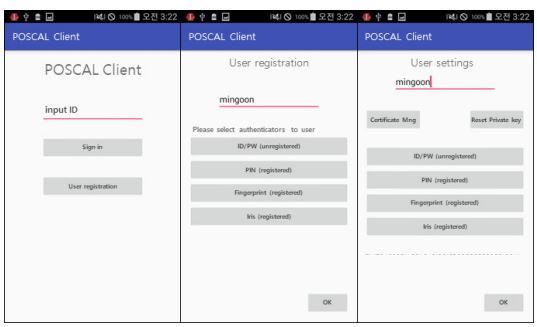


Fig. 7. User registration in POSCAL client

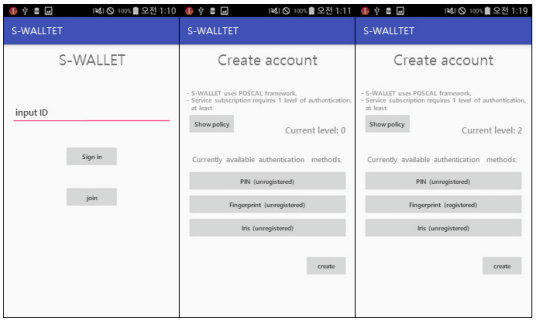


Fig. 8. Join in POSCAL S-WALLET

에서 1등급 이상의 로컬 인증이 필요하다. 서비스 가입을 하면, S-WALLET 서비스 서버가 속한 Relying party 내의 POSCAL 서버에 사용자의 서비스 가입 정보가 저장된다.

- S-WALLET 서비스 이용

S-WALLET 서비스에 로그인하려면, 스마트폰에서 1등급 이상의 로컬 인증이 필요하다. 1등급으로 로그인 했을 경우, 사용할 수 있는 기능이 없으며, 계좌조회, 자금이체 내역 조회, 자금이체를 하려면 추가적인

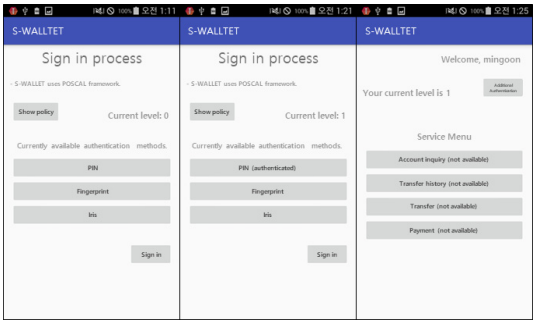


Fig. 9. Functional test: Access control by authentication level (1)

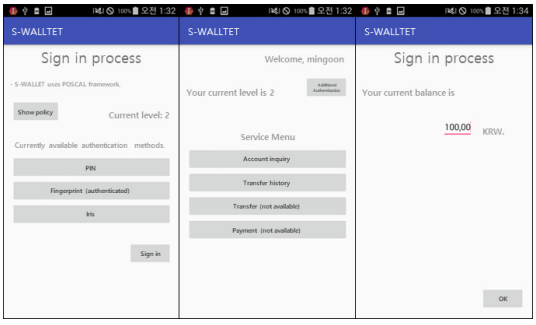


Fig. 10. Functional test: Access control by authentication level (2)


Fig. 11. Functional test: Access control by authentication level (3)

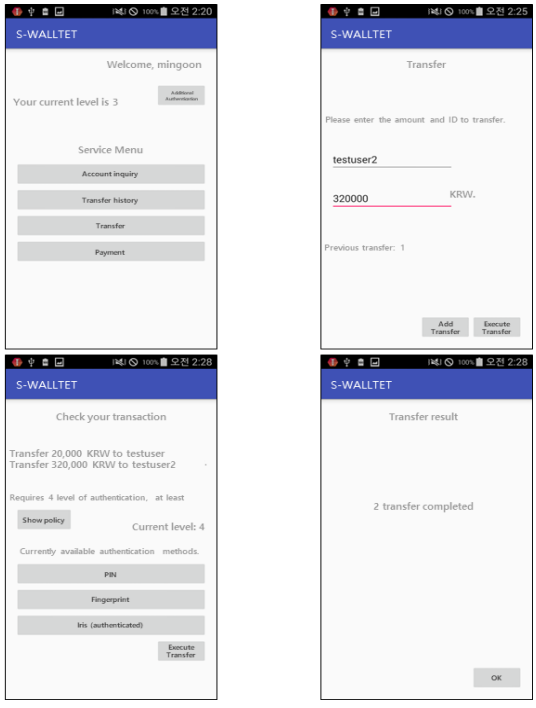


Fig. 12. Functional test: Access control by authentication level (4)

인증이 필요하다.

2등급으로 로그인 했을 경우, 계좌조회는 메시지 인증이 필요 없기 때문에 추가적인 인증 없이 사용이 가능하다.

자금이체는 3등급의 메시지 인증이 필요하기 때문에 3등급으로 로그인이 되어있더라도 추가적으로 인증을 수행해야 한다.

자금이체는 3등급의 메시지 인증이 필요하지만, 30만원 이상의 이체 내역이 포함되어 있으면, 4등급으로 추가적인 인증이 필요하다.

### 4.2 인증 프레임워크 평가

본 논문의 1장 2절에서는 멀티모달 인증정보 기반의 안전한 인증 기능을 위한 6가지의 요구사항을 정의했으며, 3장에서 제안한 POSCAL 프레임워크는 해당 요구사항을 다음과 같이 만족하고 있다.

- 사용자 및 메시지 인증

POSCAL 프레임워크는 로컬 인증에서 다양한 인증수단을 이용해 사용자를 검증하고, 검증 결과를 바탕으로 원격 인증을 수행함으로써 인증체인을 구성한다. 원격 인증은 POSCAL 서버로부터 전달받은 질의 값 또는 메시지에 대한 전자서명을 생성하고, 검증받는 방식이다. 이때 전자서명 값은 사용자의 개인키로만 생성이 가능하고, 사용자의 공개키로만 검증이 가능하기 때문에 사용자가 정당한 사용자이고, 메시지가 정당한 사용자로부터 온 것임을 인증할 수 있다.

- 인증정보의 기밀성

POSCAL 프레임워크는 로컬 인증과 원격 인증이 분리되는 구조를 갖는다. 기밀성이 유지되어야 하는 생체정보, 패스워드 등과 같은 사용자의 인증정보는 로컬 인증 모듈에서만 저장·사용되기 때문에, 네트워크를 통해 사용자의 비밀정보가 유출되는 것을 방지할 수 있으며, 서버에는 사용자의 비밀정보를 저장하지 않기 때문에 서버에서의 보안 위협과 개인정보 보호 이슈로부터 안전하다.

- 인증 내역의 무결성

POSCAL 클라이언트는 공개키 암호시스템의 기능 중 하나인 전자서명을 이용해 응답을 한다. 전자서명은 사용자의 개인키로만 생성할 수 있고 사용자의 공개키로만 검증할 수 있기 때문에, 인증 과정 중에 전송되는 사용자의 인증정보나 메시지가 제3자에 의해 변조되는 것을 POSCAL 서버 측에서 탐지할 수 있다.

- 인증 내역에 대한 부인방지

POSCAL 클라이언트는 전자서명과 함께 공개키 인증서를 이용해 응답을 한다. 전자서명은 사용자의 개인키로만 할 수 있기 때문에, 사용자는 인증 내역에 대해 부인할 수 없다. 그리고 전자서명 검증에 사용되는 공개키는 공개키 인증서를 통해 신뢰성을 확보할 수 있다.

- 보안 요구수준에 따른 서비스별 접근제어

POSCAL 클라이언트는 단말에서 지원하는 다양한 형태의 인증 모듈로부터 로컬 인증을 수행할 수 있다. 또한 POSCAL 프로토콜에는 단말에서 수행한 로컬 인증 모듈 목록을 저장할 수 있다. POSCAL 서버와 서비스 서버에서는 접근제어 정책을 정하고, 이러한 인증 정보를 바탕으로 사용자에게 허용된 권한에 맞게 데이터나 서비스를 제공할 수 있도록 제어할 수 있다.

또한 POSCAL 인증 프레임워크는 Table 9.와 같은 특징을 갖는다.

- 로컬 인증과 원격 인증을 분리한 기본 인증 구조를 갖는다. 즉, 단말에서 로컬 인증을 통해 사용자를 우선 검증하고, 로컬 인증 결과를 근거로 서버에 사용자의 신원을 대신 보증해주는 구조이다. 따라서 생체정보, 패스워드와 같은 사용자의 인증정보에 대한 기밀성을 보장한다.

Table 9. Summary of POSCAL authentication framework

| Auth. architecture | Separate local and remote authentication |
|---|---|
| Number of private keys | 1 |
| Number of public keys (certificates) | n |
| User and message authentication | Available |
| Authentication data confidentiality | Available |
| Authentication data integrity | Available |
| Non-repudiation of authentication | Available |
| Service-specific access control | Available |

- POSCAL 클라이언트에 저장되는 사용자의 개인키는 1개이며, 서버에는 n개(가입한 서비스 개수)의 공개키 인증서가 저장된다.
- 원격 인증에 사용되는 POSCAL 프로토콜은 공개키 인증서를 이용한 전자서명 기반의 인증 프로토콜로 기존의 PKI 인프라를 활용할 수 있으며, 사용자 및 메시지 인증, 인증내역에 대한 무결성과 기밀성 및 부인방지성을 보장한다.
- POSCAL 프로토콜 내부에는 사용자가 단말에서 수행한 로컬 인증 목록을 포함할 수 있기 때문에 서버에서는 이 정보를 바탕으로 사용자의 로컬 인증 수준에 따른 서비스 접근제어 기능을 제공할 수 있다.

## V. 결 론

핀테크 서비스는 일반적으로 비대면 환경에서 거래가 진행되기 때문에 다양한 보안 기술 중에서도 안전한 인증 기술은 신뢰할 수 있는 서비스의 운영을 위해 필수적으로 요구되는 보안기술이다. 뿐만 아니라 암호화, 토큰화, 부정거래탐지 등의 보안기술과는 다르게 인증은 사용자가 직접 인증정보를 제공해야 하는 과정이 있기 때문에 편리성과도 밀접한 연관이 있다. 이에 따라 안전하면서도 편리한 인증의 중요성이 더욱 커지고 있다. 또한 다양한 인증모듈이 스마트폰에 탑재되면서 앞으로 이를 이용한 다양한 형태의 인증 서비스가 개발될 수 있을 것으로 예상된다.

본 논문은 이러한 환경에서 다양한 형태의 인증정보들을 이용하여 인증 수준에 따라 서비스 접근을 제어할 수 있는 POSCAL(Service Access Control by Authentication Level) 인증 프레임워크를 제안했으며, 제안한 프레임워크의 평가를 위해서 이를 활용한 S-WALLET 전자지갑 응용 서비스를 개발하고, 프레임워크의 기능과 성능, 그리고 안전성에 대해 평가하였다.

최근 지문인식 모듈과 같은 생체인증 수단이 탑재된 스마트폰이 다수 출시되고 핀테크 서비스에 대한 수요가 커지면서, 이를 이용한 인증 기술의 중요성은 더욱 강조되고 있다. 향후에는 스마트폰뿐만 아니라 웨어러블 기기에서도 다양한 형태의 생체인증 정보들이 수집되어 인증에 활용될 것으로 예측되는 가운데, 이러한 생체인증 정보들 중에는 지문이나 홍채와 같은 정적인 정보뿐만 아니라 심박, 심전도, 뇌파, 걸음걸이 등과 같은 일상생활 속에서 사용자의 간섭 없이 동적으로 획득 가능한 정보들이 포함될 것으로 예상된다.

본 논문에서 제안한 POSCAL 프레임워크는 핀테크 서비스를 위한 온라인 전용의 인증 프레임워크이지만, 향후 이러한 동적 생체 정보들을 이용해 온라인뿐만 아니라 오프라인 서비스에서도 활용이 가능한 인증 프레임워크로 확장해 나갈 계획이다.

## References


[1] Wise Guy Reports, "Global Biometric Systems Market Research and Forecast," March 2018.

[2] Gartner, "Market Trends: New Biometric Authentication Methods in Smartphones Will Redefine User Experience," Nov. 2018.

[3] Gartner, "Gartner Highlights 10 Uses for AI-Powered Smartphones," Jan. 2018.

[4] Jong-Dae Kim and Byung-Soon Moon, "Biometrics authentication increases as wearable market grows," LG Business Insight, pp.38-44, May 2015.

[5] RSA, "Card-Not-Present Fraud in a Post-EMV Environment: Combating the Fraud Spike," June 2014.

[6] DMCmedia, "Actual usage of mobile simple payment service," April 2015.

[7] SamsungPay, http://www.samsung.com/us/samsung-pay/

[8] ApplePay, http://www.apple.com/apple-pay/

[9] Google, https://www.android.com/versions/marshmallow-6-0/

[10] Sang-soo Jang, "A Study on the Impact of Fintech on the Information Protection Industry," KISA INTERNET& SECURITY FOCUS, Feb. 2015.

[11] Jin-gyu Beom, "Study on the security enhanced PKI certificate management using the biometric informaiton," Sungkyunkwan University, Feb. 2013.

[12] Guillermo Martınez-Silva, Francisco

Rodrıguez-Henrıquez, Nareli Cruz-Cortes, and Levent Ertaul, "On the Generation of X.509v3 Certificates with Biometric Information," Proceeding of The 2007 International Conference on Security and Management, pp.52-57, Jan. 2007.

〔13〕 Sunghyuck Hong and Sunho Lim, "On Biometric Enabled X.509 Certificate," International Conference on Information Security and Privacy, July 2010.

〔14〕 Han-Ul Jang and Heung-Kyu Lee, "Biometric-PKI Authentication System Using Fingerprint Minutiae," Journal of Computer and Communications vol.2, no.4, pp.25-30, March 2014.

〔15〕 Andrew Burnett, Fergus Byrne, Tom Dowling, and Adam Duffy, "A Biometric Identity Based Signature Scheme," International Journal of Network Security vol.5, no.3, pp.317-326, Nov. 2007.

〔16〕 FIDO Alliance, https://fidoalliance.org/

〔17〕 W3C Candidate Recommendation, "Web Authentication: An API for accessing Public Key Credentials Level 1," Aug. 2018.

〔18〕 ISO/IEC 9798-3, "Information technology - Security techniques - Entity authentication mechanisms: Part 3: Entity authentication mechanisms using a public key algorithm," 1993.

〔19〕 Gartner, A Taxonomy of Authentication Methods, OASIS Trust Elevation TC, Feb. 2008.

〔20〕 Abbie Barbir, Multi-factor Authentication Methods Taxonomy, Feb. 2013.

〔21〕 KISA, "Research on security criteria for extension to electronic authentication method usage_based," Dec. 2011.

〔22〕 Wikipedia, Object identifier, https://en.wikipedia.org/wiki/Object_identifier

〔23〕 Suk-jae Lim, "FinTech security trends," TTA Journal vol.158, pp.72-79, March 2015.

## 〈저자소개〉

유 성 민 (SeongMin Yoo) 정회원
2010년 2월: 충남대학교 컴퓨터공학과 졸업
2016년 2월: 충남대학교 컴퓨터공학과 박사
2016년 3월~현재: ETRI부설연구소 선임연구원
〈관심분야〉 정보보호, 인증, 암호응용, 보안프로토콜

최 석 진 (SeokJin Choi) 정회원
1995년 8월: 경북대학교 전자공학과 졸업
1998년 2월: KAIST 전기및전자공학과 석사
2005년 3월~현재: 고려대학교 정보보호대학원 박사과정
2000년 10월~현재: ETRI부설연구소 책임연구원
〈관심분야〉 정보보호, KMI, 전자공학

박 준 후 (JunHoo Park) 학생회원
2014년 2월: 충남대학교 정보통신공학과 졸업
2016년 2월: 충남대학교 컴퓨터공학과 석사
2016년 3월~현재: 충남대학교 컴퓨터공학과 박사과정
〈관심분야〉 정보보호, 네트워크보안, 암호학, 보안프로토콜

류 재 철 (Jae-Cheol Ryou) 종신회원
1985년 2월: 한양대학교 산업공학과 졸업
1988년 5월: Iowa State University 전산학 석사
1990년 12월: Northwestern University 전산학 박사
1991년 2월~현재: 충남대학교 컴퓨터공학과 교수
〈관심분야〉 정보보호, 인터넷보안, 암호학, 보안프로토콜