논문 12-37C-10-11

한국통신학회논문지 '12-10 Vol.37C No.10
http://dx.doi.org/10.7840/kics.2012.37C.10.949

스마트 기기를 이용한 안전한 QR-Login 사용자 인증 프로토콜의 설계 및 중요 정보의 안전성 보증을 위한 방법

이 재 식◆, 유 한 나°, 조 창 현*, 전 문 석**

A Design Secure QR-Login User Authentication Protocol and Assurance Methods for the Safety of Critical Data Using Smart Device

Jae-sik Lee◆, Han-na You°, Chang-hyun Cho*, Moon-seog Jun**

요 약

최근 악성코드 및 바이러스 등으로 사용자 PC가 위협받고 있다. 특히, 제로데이 공격 등 알려지지 않은 공격들이 끊임없이 나오고 있어, 사용자 PC는 더 이상 안전하다고 확신하기 어려운 환경이 되었다. 따라서 인터넷 서비스의 이용에 있어서, 안전성이 보장되지 않은 PC를 이용하는 사용자가 기존의 인증 프로토콜을 이용하여 사용자를 인증할 경우, 인증 정보의 도난 및 중간자 공격 등 다양한 위협을 받을 수 있다. 또한 메모리 해킹등과 같은 공격을 당할 경우, 사용자는 자신의 PC가 감염되었는지 인지하기조차 어려운 상황에 놓이게 된다. 따라서 본 논문은 QR-Code와 통신기능이 가능한 스마트 기기를 활용한 안전한 QR-Login 사용자 인증 프로토콜을 설계하고, 인터넷 서비스 이용 시 중요정보의 안전성 보증을 위한 방법을 제안한다. 제안된 방법을 통하여 사용자는 알려지지 않은 공격으로부터 PC가 위협받는 경우에도 사용자의 중요정보를 보호할 수 있고, 금융거래와 같은 민감한 거래 시에도 안전하게 거래를 할 수 있다.

Key Words : QR-Code, Login, Authentication, Smart Device, Multi-Factor, Multi-Channel

ABSTRACT

Our PC have been under constant threat of malicious codes and viruses today. As many new ways of cyber attacks are being developed, such as zero-day-attack, nobody's PC is guaranteed to be safe from the attacks. In case where a user uses the existing verification protocol on a unsecured PC, the user's verification information may well be threatened by sniffing or man-in-the-middle attack. Especially, deadly attacks like memory hacking would give hard time for users to even recognize any symptom of virus infection. Therefore, this paper designs secured QR-Login user verification protocol for smart devices that are ready to communicate with QR-Code and proposes a way to keep critical data safe when using the internet. This way, user would be able to safeguard his/her critical data even when under attack by unknown attacks and safely carry out extremely sensitive task, like financial trading, on the device.

◆ 주저자 : 숭실대학교 컴퓨터공학과 박사수료, j30231@ssu.ac.kr, 정회원
° 교신저자 : 숭실대학교 컴퓨터공학과 박사과정, belover@naver.com, 정회원
* 숭실대학교 컴퓨터공학과 박사, hist0001@hanmail.net, 정회원
** 숭실대학교 정교수, mjun@ssu.ac.kr, 정회원
논문번호 : KICS2012-06-295, 접수일자 : 2012년 6월 30일, 최종논문접수일자 : 2012년 9월 7일

Ⅰ. 서 론

인터넷의 발달과 함께 다양한 서비스가 인터넷을 활용하여 제공되고 있다. 이러한 서비스 중 대부분의 서비스는 서비스 이용자의 신원을 확인하는 인증과정을 거치게 된다. 하지만 서비스를 이용하는 이용자의 PC는 악성코드 및 바이러스 등으로 감염이 된 경우, 인증과정에서 공격자로부터 사용자의 정보를 탈취 당할 수 있다. 또한 메모리해킹 공격 등을 통하여 이용자의 의사에 관계없이 악의적으로 서비스가 진행 될 수 있다. 따라서 이러한 공격으로부터 사용자의 PC를 안전하게 지키기 위하여 안티바이러스 프로그램, 방화벽 등 다양한 사용자 PC보호 프로그램을 사용하고 있지만, 제로데이 공격 등 알려지지 않은 취약점으로부터 사용자의 PC를 완벽하게 보호하기란 매우 어렵다고 할 수 있다. 따라서, 이용자의 인증과정에서 사용되는 요소의 수를 증가시킨 다양한 요소(Multi-Factor)를 활용 한 인증기술이 등장하였으며[1-6], 다양한 채널(Multi-Channel)을 활용한 인증기술도 등장하였다[7-15]. 하지만 인증 요소 및 채널이 많이 질수록 인증을 하기 위한 비용은 증가하고, 인증방법 또한 복잡해지고 있어서, 비용과 편리성을 고려한 안전한 인증방법이 필요하게 되었다.

스마트 기기란 전화선이 필요 없고, 이동 가능하며, WiFi, 3G/4G등을 이용하여 항상 인터넷 연결이 가능하고, 음성과 영상의 통신 및 인터넷 브라우징, 위치정보의 활용 등이 가능한 기기이다[16]. 스마트 기기는 다양한 인증 요소 및 다양한 채널을 활용하기 위한 대표적인 기기라 할 수 있다. 따라서 본 논문에서는 기존에 보급된 스마트 기기를 활용하여 사용자가 간단히 인증을 할 수 있는 사용자 프로토콜을 설계하였다. 또한, 서비스 이용 시 중요정보는 사용자의 확인을 받아서 서비스를 제공하는 등, 중요정보에 대한 안전성을 제공할 수 있는 방법에 관하여 제안한다. 중요정보란 서비스 이용자의 민감정보 또는 서비스 이용자의 동의로 진행된 부인방지 등이 필요한 전자서명 정보 등을 말한다. 대표적인 중요정보의 예로 주민등록번호, 계좌이체정보(사용자가 계좌이체를 승인하는 과정을 포함한 모든 정보를 말함)등이 있다.

2장에서는 기존에 제안된 다양한 요소를 통한 인증 기술 및 다양한 채널을 통한 인증기술과 QR-Code를 활용한 인증 기술을 살펴본다. 3장에서는 제안하는 QR-Login 프로토콜 및 중요정보의 안전성을 보증하는 방법을 살펴보고, 4장에서는 제안하는 프로토콜의 안전성 검증 및 중요정보의 안전성 보증 방법으로 어떻게 중요 정보의 안정성이 확보되는지 살펴본다. 5장에서는 본 논문을 정리하고 결론을 맺는다.

Ⅱ. 관련연구

2.1. 다양한 요소(Multi-Factor)를 통한 인증 기술

패스워드와 같은 알고 있는 정보를 이용한 인증은, 패스워드의 노출 시 더 이상 유효한 인증수단이 될 수 없다. 따라서 이러한 한 가지 인증 요소를 이용한 인증기술 이외에, 다른 인증 요소를 추가한 다양한 요소(Multi-Factor)를 이용한 인증 기술이 등장하였다[1-6]. 인증요소는 표 1.과 같이 그 성격에 따라서 3가지 종류[5]로 구분 할 수 있다.

표 1. 인증 요소 종류별 설명 및 예시
Table 1. Type the authentication factors descriptions and examples

Type	Description	Exemplification
Knows	Information that user known	Password
Has	Physical-Device issued by service provider for service	Smart Card, OTP(One-Time-Password)
Is	User's physical characteristics information (Can not be changed)	Biometric Information(Finger-Print, Voice, Iris and so on)

두 가지 요소를 이용한 인증기술을 Two-Factor 인증이라 하며, 세 가지 요소를 이용한 인증기술을 Three-Factor 인증이라 한다. 대표적인 Two-Factor 인증 기술로 패스워드와 스마트카드를 이용한 인증기술[3-4]이 있으며, Three-Factor 인증은 Two-Factor 인증에 생체정보를 포함하여 표1의 3가지 인증 요소를 모두 가지고 있다[5-6]. 알고 있는 것(Knows)만을 이용한 인증은 알고 있는 정보의 유출시 더 이상 유효한 인증 요소가 아니다. 이를 해결 하고자 지니고 있는 것(Has)이라는 인증 요소가 등장하였다. 그러나 지니고 있는 것은 분실 및 복제의 위험이 있어, 사용자 개인의 생체정보를 이용한 가지고 있는 것(Is)이라는 인증 요소가 등장하였다.

하지만 인증 요소가 증가하여도 다양한 요소를

통한 인증은 인증 채널에 대한 보안을 제공해 주지 않으며, 사용자가 피싱(Phishing)사이트에 접속하는 경우 인증요소는 무시될 수 있기 때문에 중간자공격(MITM)이나 트로잔(Trojan) 공격으로부터 안전하지 않다[1]. 또한 메모리해킹 공격[17]이 일어나는 경우 아무리 강력한 인증요소를 사용하더라도, 공격자는 사용자 PC의 메모리 영역을 조작하여 사용자에게 보이는 데이터와 다르게 데이터 정보를 조작하므로 사용자는 공격여부를 인지할 수 없다. 이러한 문제를 해결하기 위하여 인증 채널에 대한 보안을 제공할 수 있는 다양한 채널을 통한 인증 기술[7-15]이 등장하였다.

2.2. 다양한 채널(Multi-Channel)을 통한 인증 기술

다양한 요소를 통한 인증기술도 중간자 공격(MITM) 및 트로잔 공격(Trojan Attack), 그리고 메모리 해킹 공격(Memory Hacking Attack)등과 같은 공격에는 여전히 취약하다. 따라서 채널 자체를 늘려 복수개의 채널을 통하여 인증을 수행하는 다양한 채널(Multi-Channel)을 통한 인증 기술이 등장하였다. 두 개의 채널을 이용한 인증기술을 Two-Channel 인증이라 한다. 일반적으로 Two-Channel 인증에서 첫 번째 채널은 사용자가 이용하는 PC의 채널을 의미하며, 두 번째 채널은 PC 이외의 다른 기기가 이용하는 채널을 의미한다. 특히 Two-Channel 인증은 금융정보와 같이 금전과 관련된 민감한 정보를 사용하는 서비스에 결합될 경우 매우 강력한 인증 수단이 될 수 있다[7,9-10].

2.3. QR-Code을 활용한 인증 기술

QR-Code를 활용한 인증기술[8-15]은 QR-Code를 촬영할 수 있는 스마트 기기 자체가 '지니고 있는 것(Has)'에 해당하며 또한 스마트 기기의 특성상 통신이 가능하므로, Two-Factor 인증 및 Two-Channel 인증의 특성을 모두 만족할 수 있는 기술이다. QR-Code란 2차원 바코드로 에러 정정기능을 가지고 있으며, 숫자는 7,089개, 알파벳을 포함한 문자는 4,296개, 데이터는 2,953바이트까지 표현할 수 있다. QR-Code의 QR은 "Quick Response"의 약자로 빠르게 응답할 수 있다는 의미를 지니고 있다[18]. 기존에 제안된 QR-Code를 이용한 로그인 시스템[8,11-15]은 로그인과정에서는 Two-Channel 인증을 수행하나, 로그인 이후의 과정에서는 계속해서 하나의 채널인 PC를 이용해서 통신을 한다. 그러므로 이후 서비스 제공과정은 여전히 하나의 채널 이상의 보안효과를 기대하기 힘들다. 따라서, 온라인 피싱 공격을 통한 능동적 중간자 공격 (Active MITM)등을 하여도 사용자는 인지할 방법이 없다. 이를 해결하기 위하여 피싱 공격으로부터 안전하도록 사용자의 PC와 스마트폰의 GPS 또는 IP대역과 같은 위치정보를 비교하여 동일한 영역에 존재할 경우 해당 서비스를 이용하게 하는 방법[9]등을 제안하고 있으나, 이는 동일 영역에서 위장된 피싱 공격을 하는 경우 여전히 취약점이 존재한다. 또한 금융서비스 정보 이용 시 스마트 기기를 통하여 금융정보를 확인하는 방법도 제안되었으나[10]. 이는 스마트 기기를 통한 정보 확인으로 메모리 해킹공격에는 안전할 수 있으나, 능동적 중간자 공격과 같은 피싱공격에는 안전하지 않다. 또한 제안된 방법들은 로그인 시점 또는 금융정보와 관련된 내용을 거래하는 시점 등과 같은 특정 상황으로 Two-Channel 인증 내용을 한정하고 있어 그 활용성이 떨어진다.

Ⅲ. 제안하는 내용

이 장에서는 QR-Code를 이용하여 서비스 이용 시 사용자를 인증하는 QR-Login 프로토콜을 제안하고, 서비스 이용과정 중에서 중요정보의 안정성 확보를 위한 방안을 제안한다. 제안된 프로토콜 및 중요정보의 안전성 확보 방안은 특정 데이터나 서비스에 국한되지 않으며, 범용성을 가지도록 설계하였다. 제안하는 내용을 위한 약어는 표2와 같고, 프로토콜에서 사용하는 전송 내용에 관한 정보는 표3과 같다.

본 논문의 가정 사항은 다음과 같다.

- *SD*는 악성코드 및 바이러스로부터 감염되지 않은 안전한 단말이며, 분실 및 도난을 대비하여 기기 및 어플리케이션에 비밀번호가 설정되어 있다.
- *SD*는 카메라가 있고, QR-Code를 지원하며, 서명 값을 검증할 수 있는 어플리케이션이 설치되어 있다.
- *SD*에 설치된 어플리케이션은 $Cert_{SP}$를 발급한 상위 인증기관(CA)의 $Cert_{CA}$를 포함한다.
- *SD*는 *SP* 서명 값을 검증하기 위해, 어플리케이션을 통하여 $Cert_{SP}$를 전송받을 수 있다.

- *SD*와 *SP*는 사전에 대칭키 알고리즘에 사용할 암호화 알고리즘과 비대칭키 암호화 알고리즘에 사용할 암호화 및 전자서명 알고리즘을 공유하고 있다. 또한 *SeedKey*, 및 *SKey* 생성에 필요한 대칭키 생성 알고리즘을 공유하고 있다.
- *SP*에 공인인증서를 등록하는 경우, 사용자 *PC*에 사용자의 공인인증서($Cert_{SD}$) 및 개인키 쌍이 존재한다.
- *SP*와 사용자(*PC* 또는 *SD*)는 SSL/TLS 프로토콜[19]등을 이용하여, 안전한 채널을 설정할 수 있다.
- *SP*와 *SD*가 3G/4G등 통신사망을 이용하여 통신을 할 경우 해당 채널은 안전하며, WiFi 망을 이용할 경우 검증된 AP를 통해 안전한 WiFi 프로토콜을 이용하여 통신한다. 따라서 *SP*와 *SD*사이의 망도 안전하다.

표 2. 약어
Table 2. Abbreviations

Abbreviations	Context	Explanation
SD	Smart Device	User's Portable Smart Device
PC	Personal Computer	User's Personal Computer
SP	Service Provider	Kind of Service-Provider's Server
CA	Certificate Authority	Including all kinds of Public and Private Certificate Authorities
IP	Internet Protocol Address	Internet Protocol Address
PN	Port Number	Port Number
SN	Session Number	Session Number
URL	Uniform Resource Locator	Specific Character String for Network Path in Internet
RN_X	X's Random Number	Randomly Generated Number(Can be used to prevent Replay Attack)
$Cert_X$	X's Certification	Certification including Public-Key issued by CA
UserInfo	User Information	User-related information like a Smart-Device information(IMEI, Phone-Number and so on) and User Personal Information(User identification Number, Address, E-mail Address and so on)
EncUserInfo	Encrypted User Information	Encrypted User's Information
UserAddInfo	User Additional Information	User's additional information that can not be identified user
SeedKey	Seed Key	Secret shared seed key between SD and SP for SKey generation
SKey	Session Key	Through Seed Key in each session, the newly generated secret key is derived that can be used in symmetric algorithm
SP_{Pub}	SP's Public Key	SP's Publick Key included in SP's certificate(Can be used to asymmetric Algorithm)
Sign	Signed Data	Digital Signature signed by Private Key
Data	Data	General information that require lower important compared with Critical Data
CData	Critical Data	Important data information that confidentiality required
EncCData	Encrypted Critical Data	Encrypted CData information
$QR-Code(\cdots)$	QR-Code Encoded Data	Data encoded with QR-Code, namely QR-Code

표 3. 프로토콜 전송 내용 정보
Table 3. Protocol Transport Context Information

Transport Context	Details	Explanation
Req_Regist	IP, PN, SN	Request user registration page PC to SP, at the start of user registration steps
Req_Regist_Info	$Form$, $QR-Code(URL, SN, RN_{SP}, Sign)$	Request registration information related with user
Res_Regist_Info	$EncUserInfo$, $EncRN_{SD}$ RN_{SP}	Response Registration-related information including user-related information to SP
Res_Regist_AddInfo	$UserAddInfo$	User additional information response from PC to SP
Res_Regist_Info_Result	-	Notification to PC and SD user registration result
Req_Regist_Cert	-	Request user certificate
Res_Regist_Cert	$Cert_{User}$	Transfer user certificate to SP
Res_Regist_Cert_Result	-	Notification to PC and SD user certificate registration result
Req_Login	IP, PN, SN, RN_{PC}	Request user login page PC to SP, at the start of user login steps
Req_Login_Info	$QR-Code(URL, SN, RN_{SP}, Sign)$	Login-related information transfer from SP to PC
Res_Login_Info	$Enc_{SKey}(IMEI, RN_{SP}, RN_{PC})$, RN_{SD}, $PhoneNum$	Login-needed information transfer from SD to SP
Res_Login_Info_Result	$Enc_{SKey}(IMEI, RN_{SD})$	Transfer information from SP to SD for mutual-authentication
Req_Service_Info	IP, PN, SN	Request service-related data transmission from PC to SP
Res_Service_Info	$Data$, $QR-Code(EncCData)$ or $QR-Code(URL)$	Important information is transferred as QR-Code encrypted by SKey, and general information is transferred by plain text
Req_Service_AddlInfo	URL	If data is too big to transfer, data referenced URL is transferred to SP
Res_Service_AddlInfo	$EncCData$	Data in the corresponding URL is encrypted by SKey and transferred from SP to SD
Req_Service_InputInfo	$EncCData$	Important data is encrypted by SKey and transferred from SD to SP
Req_Service_AddInputInfo	$Data$	Data Transmission from PC to SP
Res_Service_InputInfo	-	Input notification for important information and data
Res_Service_InputInfo_Result	-	Verification notification for important information

3.1. QR-Login 사용자 인증 프로토콜

QR-Login 사용자 인증 프로토콜은 서비스를 제공받기 원하는 사용자가 서비스 제공자(*SP*)에 사용자를 인증하는 프로토콜이다. 사용자 인증 프로토콜은 *SP*에 사용자를 등록하는 사용자 등록 단계와 등록된 사용자가 서비스를 제공하는 서버에 로그인하는 로그인 단계의 2가지 단계로 나누어진다. 또한, 온라인 전자결제와 같은 전자서명 기능이 필요

한 금융서비스 등의 이용 시 사용자 등록단계에서 사용자의 공인인증서($Cert_{SD}$)를 함께 등록할 수 있도록 하여, 다양한 서비스에 활용이 가능하도록 범용적인 형태로 인증 프로토콜을 설계하였다.

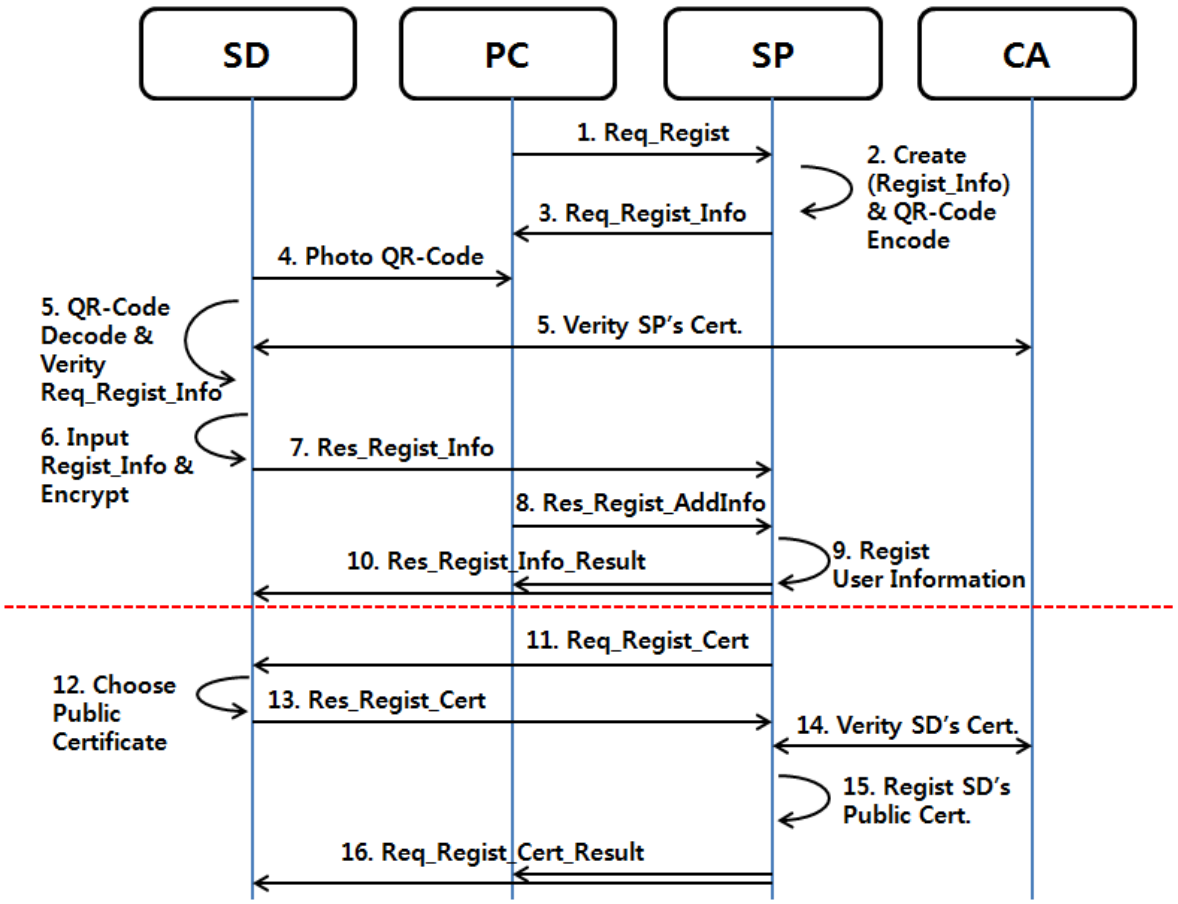


그림 1. 사용자 등록 및 공인인증서 등록 단계
Fig. 1. Register User Information and Public Certificate Phase

3.1.1 사용자 등록 단계

사용자는 서비스제공자(*SP*)의 서비스를 이용하기 위해 *SP*에 사용자의 스마트 기기인 *SD*를 등록하여야 한다. 사용자는 *SD*와 *PC*를 이용하여 사용자 등록 단계를 진행한다. 사용자 등록 단계는 그림1의 1~10과 같고, 사용자와 관련된 중요정보는 모두 안전한 *SD*로 입력받는 특징이 있다. 따라서 사용자 정보는 *PC*를 통하여 유출될 수 없으므로, 사용자 정보 유출에 안전하다.

1) 서비스 등록 페이지 접속

사용자는 *PC*를 이용하여 *SP*의 서비스 등록페이지에 접속한다. 서비스 등록 요청 메시지(*Req_Regist*)는 *PC*의 *IP*, *PN*, *SN*가 포함되어 있다. 포함된 정보는 *SP*가 *PC*를 구별하기 위하여 사용한다.

2) 서비스 등록 요청 정보 생성 및 QR-Code 인코딩

*SP*는 서비스 제공을 위한 사용자 정보를 입력받기 위하여, 입력과 관련된 스마트 기기 전용 주소인 *URL*을 생성하고 재생 공격(Replay Attack)을 방지하기 위하여 RN_{SP}를 생성한 후, 수식 (1)과 같이 이 값과 *SN*을 함께 서명하고 *URL*, *SN*, RN_{SP} 그리고 서명 값을 QR-Code로 인코딩한다.

$$QR-Code = Encode\left(\begin{matrix} URL, SN, RN_{SP}, \\ Sign_{SP_{Pri}}(URL, SN, RN_{SP}) \end{matrix}\right) \quad (1)$$

3) 서비스 등록 요청 정보 요청

*SP*는 QR-Code를 포함한 서비스 등록 요청 정

보 메시지(*Req_Regist_Info*)를 *PC*에 전송한다. 만약 추가로 사용자 정보가 필요한 경우 해당 정보의 입력 필드(*Form*)도 함께 전송한다. 단, 이때 *Form*을 통해 입력되는 정보는 중요정보(*CData*)가 아닌 일반정보(*Data*)이다. 만약, 기밀성이 요구되는 정보일 경우 이러한 형태로 정보의 입력을 요구하지 말아야 한다.

4) QR-Code 촬영

사용자는 *SD*에 설치된 어플리케이션을 이용하여 *PC*화면에 출력된 QR-Code를 촬영한다.

5) QR-Code 디코딩 및 사용자 등록 요청 정보 검증

QR-Code가 촬영되면 어플리케이션은 수식 (2)와 같이 QR-Code를 디코딩하여 URL, SN, RN_{SP} 그리고 서명 값을 추출한다. 어플리케이션은 *SP* 인증서($Cert_{SP}$)의 유효성 확인을 위해, 어플리케이션을 통하여 $Cert_{SP}$를 다운로드 받는다. 그리고 *CA*로부터 $Cert_{SP}$를 검증하여 유효성을 확인한다. $Cert_{SP}$의 유효성이 확인된 경우 어플리케이션은 $Cert_{SP}$에 포함된 *SP*의 공개키(SP_{Pub})를 추출하고, 이를 통하여 서명 값을 검증한다. 서명이 유효한 경우, 사용자 정보 입력 페이지를 *SD*화면에 보여준다. 만약, 서명이 유효하지 않은 경우 해당 서비스 제공자는 더 이상 유효한 서비스 제공자가 아닐 수 있으므로, 사용자는 서비스 가입을 보류할 수 있다.

$$Decode(QR-Code) = \begin{matrix} URL, SN, RN_{SP}, \\ Sign_{SP_{Pub}}(URL, SN, RN_{SP}) \end{matrix} \quad (2)$$

6) 사용자 등록 요청 정보의 입력 및 암호화

사용자는 *SD* 화면에 출력된 사용자 정보 입력 필드에 사용자 정보($UserInfo$)를 입력한다. $UserInfo$ 중 *SD*와 관련된 IMEI 및 휴대폰 번호는 어플리케이션이 자동으로 *SD*로부터 추출하여 입력받을 수 있다(단, 해당 스마트 기기의 운영체제가 IMEI 수집 및 휴대폰 번호 수집을 허용하고 지원해야 함). $UserInfo$ 입력이 완료되면, 어플리케이션은 수식 (3)과 같이 *SP*와 통신 및 로그인에 필요한 세션 비밀키 정보인 $SKey$ 생성을 유도할 수 있는 비밀 씨드(Seed)값인 $SeedKey$를 생성한다. $SeedKey$ 생성에 필요한 알고리즘은 사전에 공유되어 있으며, $SeedKey$ 생성을 위한 씨드(Seed) 값으로 *SD*가 생성한 랜덤 값인 RN_{SD}와 *SP*로부터 전달받은 랜덤 값인 RN_{SP}가 이용된다. 그리고 $UserInfo$ 암호화에 사용될 세션 비밀키인 $SKey$를 수식(4)와 같이 $SeedKey$ 및 SN을 이용하여 생성한다. $SKey$의 생성이 완료되면 수식(5)와 같이 *SP*의 공개키를 이용하여 RN_{SD}를 암호화하고, 수식(6)과 같이 $SKey$를 이용하여 $UserInfo$를 암호화 한다. 수식(5)의 암호화는 비대칭키 방식의 암호 알고리즘이며, 수식(6)의 암호화는 대칭키 방식의 암호 알고리즘이다.

$$SeedKey = GenerateSeedKey(RN_{SD}, RN_{SP}) \quad (3)$$

$$SKey = GenerateSessionKey(SeedKey, SN) \quad (4)$$

$$EncRN_{SD} = Enc_{SP_{Pub}}(RN_{SD}) \quad (5)$$

$$EncUserInfo = Enc_{SKey}(UserInfo) \quad (6)$$

7) 사용자 등록 요청 정보의 응답

*SD*는 사용자 등록 요청 정보의 결과(*Res_Regist_Info*)를 *SP*에 전송한다. 이 값에는 암호화된 사용자 정보인 $EncUserInfo$와 암호화된 RN_{SD}값인 $EncRN_{SD}$, 그리고 RN_{SP}를 포함한다.

8) 사용자 등록 요청 추가정보의 응답

만약 *SP*가 사용자에게 추가로 요구되는 정보(*Res_Regist_AddInfo*)를 요청한 경우 사용자는 *PC*의 입력 $Form$으로부터 추가정보를 입력하고 *SP*에 전달한다. 단, 이때 $Form$을 통해 입력되는 정보는 중요정보($CData$)가 아닌 일반정보($Data$)이다.

9) 사용자 정보 등록

*SP*는 *SD*와 *PC*로부터 입력받은 사용자 정보를 등록한다. *SP*는 RN_{SP}를 이용하여 *SD*와 연계된 *PC*를 찾는다. 그리고 *SP*는 $EncRN_{SD}$를 복호화하여 RN_{SD}값을 추출하고, 이를 이용하여 수식(3)과 수식(4)와 같이 $SeedKey$ 및 $SKey$를 생성한다. 생성된 $SKey$를 이용하여 수식 (7)과 같이 $EncUserInfo$를 복호화 하여 $UserInfo$를 얻는다. *SP*는 사용자와 관련된 정보인 $UserInfo$와 공유된 비밀 씨드인 $SeedKey$를 등록한다.

$$UserInfo = Dec_{SKey}(EncUserInfo) \quad (7)$$

10) 사용자 정보 등록 완료

*SP*는 *SD*와 *PC*에게 사용자 정보 등록이 완료

(*Res_Regist_Info_Result*)되었음을 알린다.

3.1.2. 사용자 공인인증서 등록 단계

인터넷 뱅킹 또는 전자상거래와 같은 금융서비스의 경우, 사용자의 공인인증서를 이용한 전자서명을 해야 한다. 이번 항에서는 이와 관련된 사용자 공인인증서 등록 과정을 설명한다. 사용자 공인인증서 등록 단계는 그림1의 11~16과 같다. 이 과정은 사용자 등록단계 이후에 연속하여 발생할 수도 있고, 필요시 별도의 과정으로 동작할 수도 있다. 아래의 설명은 사용자 정보 등록 단계 이후에 연속해서 공인인증서를 등록하는 단계이다.

11) 사용자 공인인증서 등록 요청

*SP*는 사용자에게 공인인증서 등록(*Req_Regist_Cert*)을 요청한다.

12) 공인인증서 선택

사용자는 *SP*에 등록할 공인인증서를 선택한다.

13) 사용자 공인인증서 등록 응답

*SD*의 어플리케이션은 공인인증서 등록(*Res_Regist_Cert*)을 응답을 하기 위해 사용자에 의해 선택된 공인인증서($Cert_{User}$)를 *SP*에게 전달한다.

14) 사용자 공인인증서 검증

*SP*는 사용자로부터 전달된 사용자 공인인증서($Cert_{User}$)를 *CA*로부터 검증한다.

15) 사용자 공인인증서 등록

$Cert_{User}$가 *CA*로부터 발급된 유효한 공인인증서인 경우, *SP*는 해당 사용자의 공인인증서($Cert_{User}$)를 등록한다.

16) 사용자 공인인증서 등록 완료

*SP*는 *SD*에게 사용자 공인인증서 등록이 완료(*Res_Regist_Cert_Result*)되었음을 알린다.

3.1.3 사용자 로그인 단계

사용자는 *SD*를 활용하여 QR-Code로 인증정보에 필요한 정보를 전달받아 *SD*를 이용하여 로그인을 수행한다. 로그인 과정은 시도-응답(Challenge-Response) 방식을 통하여, 재생 공격(Repaly Attack)으로부터 안전성을 확보하였다. 또한 강화된 보안이 필요한 경우 공인인증서를 이용하여 로그인 단계를 수행할 수 있다. 사용자와 서버는 상호간에 인증을 수행하므로 피싱 공격 및 중간자 공격으로부터 안전하다.

1) 사용자 로그인 요청

사용자는 *PC*를 이용하여 *SP*에 로그인을 요청(Req_Login)한다. 로그인 요청메시지에는 *IP*, *PN*, *SN*, RN_{PC} 가 포함되어 있으며, *IP*, *PN*, *SN*은 *SP*가 *PC*를 구분하기 위하여 필요하며, RN_{PC}는 동일한 로그인 요청 메시지 방지를 위하여 필요하다.

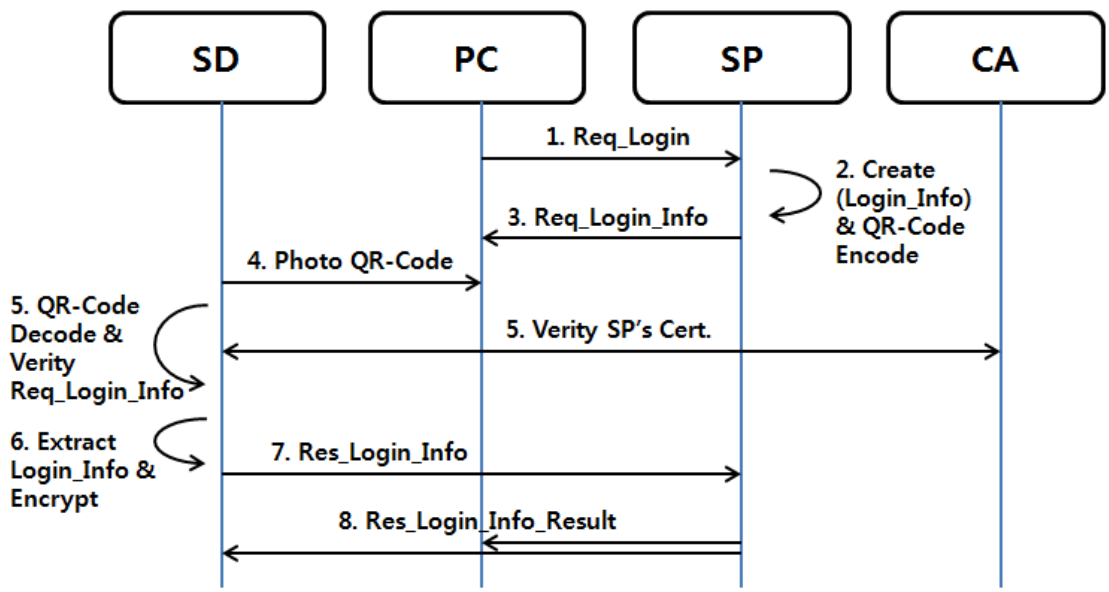


그림 2. 사용자 로그인 단계
Fig. 2. User Login Phase

2) 사용자 로그인 요구 정보 생성 및 QR-Code 인코딩

*SP*는 사용자 신원을 인증하는 로그인을 진행하기 위하여, 사용자 정보의 인증정보가 입력될 스마트 기기 전용 주소인 *URL* 주소를 생성하고, 재생 공격을 방지하기 위하여 RN_{SP}를 생성한 후, 수식(7)과 같이 RN_{PC}를 포함하여 전체 값을 서명하고 *URL*, *SN*, RN_{SP}, RN_{PC} 그리고 서명 값을 QR-Code로 인코딩한다.

$$QR-Code = Encode\begin{pmatrix} URL, SN, RN_{SP}, RN_{PC}, \\ Sign_{SP_{Pri}}(URL, SN, RN_{SP}, RN_{PC}) \end{pmatrix} \quad (7)$$

3) 사용자 로그인 요구 정보 요청

*SP*는 QR-Code를 포함한 로그인 요구 정보 메시지(Req_Login_Info)를 *PC*에 전송한다.

4) QR-Code 촬영

사용자는 *SD*에 설치된 어플리케이션을 이용하여 *PC*화면에 출력된 QR-Code를 촬영한다.

5) QR-Code 디코딩 및 사용자 로그인 요구 정보 검증

QR-Code가 촬영되면 어플리케이션은 수식 (8)와 같이 QR-Code를 디코딩하고 *URL*, *SN*, RN_{SP}, RN_{PC}, 그리고 서명 값을 추출한다. 어플리케이션은 *SP* 인증서($Cert_{SP}$)의 유효성 확인을 위해, 어플리케이션을 통하여 $Cert_{SP}$를 다운로드 받는다. 그리고 *CA*로부터 $Cert_{SP}$를 검증하여 유효성을 확인한다. $Cert_{SP}$의 유효성이 확인된 경우 어플리케이션은 $Cert_{SP}$에 포함된 *SP*의 공개키(SP_{Pub})를 추출하고, 이를 통하여 서명 값을 검증한다. 서명이 유효한 경우, 사용자 정보 입력 페이지를 *SD*화면에 보여준다. 만약, 서명이 유효하지 않은 경우 해당 서비스 제공자는 더 이상 유효한 서비스 제공자가 아닐 수 있으므로, 사용자는 해당 서비스를 제공받지 않는다.

$$Decode(QR-Code) = URL, SN, RN_{SP}, RN_{PC}, Sign_{SP_{Pub}}(URL, SN, RN_{SP}, RN_{PC}) \quad (8)$$

6) 사용자 로그인 요구 정보의 입력 및 암호화

어플리케이션은 *SD*와 관련된 사용자의 고유 정보인 IMEI 및 휴대폰 번호를 추출하고, 랜덤 값인 RN_{SD}를 생성한다. 그리고 *SP*에 등록한 비밀 씨드인 *SeedKey*와 세션번호인 *SN*을 이용하여 수식(9)와 같이 *SKey*를 유도하고 이를 이용하여 수식(10)과 같이 암호화를 수행한다. 수식(10)의 암호화는 대칭키 방식의 암호 알고리즘이다. 그리고암호화된 값은 RN_{SP} 와 RN_{PC}가 포함되어 재생 공격으로부터 안전하다.

$$SKey = GenerateSessionKey(SeedKey, SN) \quad (9)$$

$$EncLoginInfo = Enc_{SKey}(IMEI, RN_{SP}, RN_{PC}) \quad (10)$$

7) 사용자 로그인 요구 정보의 응답

*SD*는 사용자 로그인 요구 정보(Res_Login_Info)를 *SP*에 전송한다. 이 값에는 사용자가 등록한 *SD*의 고유 식별 번호인 IMEI가 RN_{SP} 및 RN_{PC}와 함께 암호화 되어 있고, 로그인 요구 정보의 재생 공격 예방을 위하여 RN_{SD}가 함께 전달된다. *SP*는 *SD*를 식별하고 암호화된 IMEI를 복호화 하기 위하여 *PhoneNum*를 참조 할 수 있도록 사용자의 전화번호도 함께 전송된다.

8) 사용자 로그인 완료

사용자 로그인 요구 정보(Res_Login_Info)를 전송받은 *SP*는 *PhoneNum*를 참조하여 *SD*와의 대칭된 비밀 씨드인 *SeedKey*를 선택하고 수식 (9)와 같이 *SKey*를 생성한다. 생성된 *SKey*를 이용하여 *EncLoginInfo*를 수식 (10)과 같이 복호화 한다. *SP*는 복호화 된 IMEI를 이용하여 *SD*를 검증하고, 매칭된 RN_{SP}, RN_{PC}를 이용하여 *SD*와 연관된 *PC*를 식별한다. 또한 RN_{SD}를 저장하여, 로그인 요구 정보(Res_Login_Info)의 재생 공격을 예방한다. 로그인이 완료되면 *SP*는 *SD*와 *PC*에게 사용자의 로그인이 완료(Res_Login_Info_Result)되었음을 알린다.

$$IMEI, RN_{SP}, RN_{PC} = Dec_{SKey}(EncLoginInfo) \quad (10)$$

3.1.4 사용자 재등록 단계

스마트 기기인 *SD*는 휴대가 가능한 매체로 분실의 가능성이 존재한다. 이러한 경우 사용자는 서비스제공자(*SP*)의에게 기존에 사용 중이던 *SD*가 분실되었음을 알리고, 새로운 스마트 기기를 등록하여야 한다. *SP*는 분실된 *SD* 와 관련된 IMEI를 포함한 *SeedKey*의 등록을 취소하고, 새로운 *SD*의 정보를 등록한다. 이 과정은 그림1의 과정과 유사하

게 이루어지므로 별도의 상세한 등록 과정은 생략한다.

3.2. 중요정보 안전성 보증 방법

중요정보(*CData*)란 서비스 이용자의 민감정보 또는 서비스 이용자의 동의로 진행된 전자서명 및 부인방지 등을 필요로 하는 정보 등을 말한다. 대표적인 *CData*의 예로 주민등록번호, 계좌이체정보(사용자가 계좌이체를 승인하는 과정을 포함한 모든 정보를 말함)등이 있다. *CData*의 안전성을 보증을 위해 *CData*는 기밀성 및 무결성을 제공해야 한다. 본 논문에서는 *CData*의 안전성 보증을 위하여 일반적인 정보(*Data*)는 *PC*가 처리하고 *CData*는 안전한 *SD*가 처리하여 *CData*의 안전성을 보증한다. *CData*의 출력 및 입력 단계는 그림 3과 같다. 단계 1~5는 출력 과정을 나타내며, 단계 6~12는 입력과정을 나타낸다. *CData*를 포함한 모든 *Data*는 서비스 이용자가 인증된 상태, 즉 로그인된 상태에서만 전송하고 전송받는다.

3.2.1 중요정보의 출력 방법

중요정보(*CData*)의 안전성 확보를 위하여 *CData*는 사용자의 PC에 출력되지 않고, QR-Code의 형태로 인코딩 되어 출력된다. 인코딩된 *CData*는 그 길이가 QR-Code에 포함할 정도로 짧은 경우 *SKey*를 통하여 암호화 되며, 그 길이가 긴 경우 정보를 가지고 있는 *URL*로 표현된다. *SD*를 이용하여 *SP*가 생성한 *URL*에 접속하는 경우 *SP*는 *SD*의 IMEI 번호 체크 등을 통하여 해당 *URL*로의 접근을 통제할 수도 있다. 또는 생성되는 *URL*은 해쉬 값과 같은 난수화 된 값으로 사용하여, *URL*로의 접근을 위한 추축성 공격을 막고, 생성된 *URL* 주소의 유효기간을 수분 이내로 짧게 설정하여 해당 *URL*로의 무차별적 접근을 최소화 할 수도 있다.

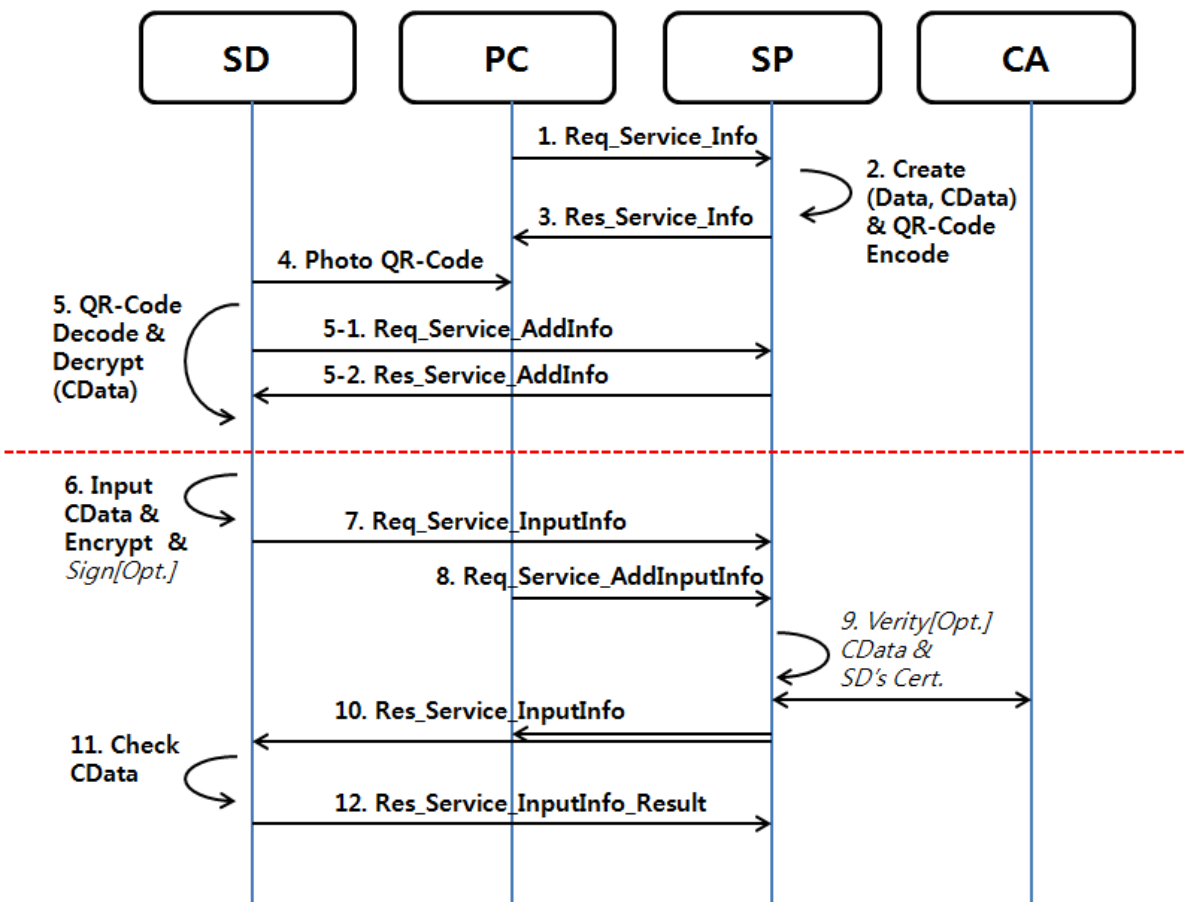


그림 3. 중요정보 출력 및 입력 단계
Fig. 3. Critical Data Output and Input Phase

생성되는 해쉬값은 현재시간 및 세션정보와 서버에서 생성한 랜덤값 등과 같은 정보를 기반으로 만들어진 값으로, 예를 들어 “http://site.com/bs24fb8R” 등과 같이 이루어 질 수 있다. 또한 만약 무차별적 *URL*로의 접근하더라도 제공되는 데이터가 *SKey*로 암호화 되어 있어, 인증된 사용자 이외에는 해당 정보를 확인할 수 없다.

1) 서비스 접속 및 서비스 정보 요청

사용자는 *PC*를 이용하여 *SP*의 서비스 요청 페이지에 접속하고, 서비스 정보를 달라고 요청(Req_Service_Info)한다. 서비스 요청 메시지는 *PC*의 구별을 위한 *IP*, *PN*, *SN*를 함께 전송한다.

2) 서비스 정보 요청 응답 페이지 생성 및 QR-Code 인코딩

*SP*는 서비스 제공을 위하여 관련된 정보를 생성한다. 관련된 정보는 중요정보인 *CData*와 일반 정보인 *Data*로 구성되어 있다. *SP*는 *CData*를 수식 (11)과 같이 *SKey*를 통하여 암호화하고, 암호화된 *EncCData*의 길이가 QR-Code에 포함할 수 있는 정도로 짧은 경우(약 2천 바이트 내외), 수식 (12)와 같이 QR-Code로 인코딩한다. 여기서, 수식 (11)의 암호화는 대칭키 방식의 암호 알고리즘이다. 만약 QR-Code로 표현할 수 있는 크기를 넘어가는 경우, *SP*는 *EncCData*를 임시 *URL*로 복사하고, 해당 *URL*주소를 수식 (13)과 같이 QR-Code로 인코딩한다. 여기서 임시 *URL*는 해쉬 값과 같은 난수화 된 값을 사용하고, 생성된 *URL*주소의 유효기간을 수분 이내로 짧게 설정하여 *URL*로의 무차별적인 접근을 최소화 할 수 있다. 또는 *SD*의 IMEI번호 체크등을 통하여 허가된 *SD*이외의 접근을 원천적으로 차단할 수도 있다.

$$EncCData = Enc_{SKey}(CData) \quad (11)$$
$$QR-Code = Encode(EncCData) \quad (12)$$
$$QR-Code = Encode(URL) \quad (13)$$

3) 서비스 정보 요청 응답

*SP*는 서비스 정보 요청에 대한 응답(Res_Service_Info)을 *PC*에 전송한다. 서비스 정보는 *Data*와 QR-Code로 전달된다. *Data* 정보는 사용자가 *PC*를 이용하여 읽을 수 있으며, QR-Code로 인코딩된 데이터는 다음 단계를 통하여 읽을 수 있다.

4) QR-Code 촬영

사용자는 *SD*에 설치된 어플리케이션을 이용하여 *PC*화면에 출력된 QR-Code를 촬영한다.

5) QR-Code 디코딩 및 중요정보 복호화

QR-Code가 촬영되면 어플리케이션은 수식 (14)와 같이 QR-Code를 디코딩하여 *EncCData* 또는 *URL*을 추출한다. 추출된 데이터가 *EncCData*인 경우 어플리케이션은 수식 (15)와 같이 *SP*와의 대칭키 *SKey*를 이용하여 *CData*를 복호화 한다. 복호화 된 *CData*는 *SD*화면에 보여주고, 사용자는 *CData*를 확인 할 수 있다. 만약 추출된 데이터가 *URL*인 경우, 어플리케이션은 *SP*에 해당 *URL*로 서비스 추가 정보 요청(Req_Service_AddlInfo)을 한다. 서비스 추가 정보 요청을 받은 *SP*는 *SD*에 서비스 추가 정보 응답(Res_Service_AddlInfo)을 한다. 이 메시지는 *EncCData*를 포함하고 있다. *EncCData*를 전송받은 *SD*는 수식 (15)와 같이 대칭키 *SKey*를 이용하여 *CData*를 복호화하고, 이를 사용자에게 보여준다.

$$Decode(QR-Code) = EncCData \text{ or } URL \quad (14)$$
$$CData = Dec_{SKey}(EncCData) \quad (15)$$

3.2.2 중요정보의 입력 방법

중요정보(*CData*)는 사용자의 민감한 정보 또는 사용자가 전자서명을 하려는 정보이다. 따라서 *CData*는 안전한 기기에서 처리 및 취급되어야 한다. 본 논문에서는 *CData*를 *SD*에서 입력함으로써 안정성을 확보하고 또한 전자서명도 *SD*에서 수행하여, *PC*가 메모리 해킹등과 같은 악의적인 공격을 당하더라도, *CData*와 관련된 어떠한 정보도 공격자가 알 수 없어, *CData*의 위·변조는 불가능하다. 부가적 기능인 전자서명과 관련하여, 사용자는 서비스 가입단계에서 자신의 공인인증서를 *SP*에 등록시킬 수 있다. 아래의 설명은 중요정보 출력 단계 이후에 연속해서 중요정보를 입력하는 단계이다.

6) 사용자 중요정보 입력 및 암호화

사용자는 서비스 진행 등에 필요한 중요정보(*CData*)를 *SD*에 설치된 어플리케이션을 통하여 입력한다. 입력된 *CData*는 수식 (16)과 같이 *SP*와 공유된 대칭키 *SKey*로 암호화 한다. 만약 전자서명이 필요한 데이터의 경우 사용자는 수식 (17)과 같이 *SP*에 등록한 공인인증서와 매칭 된 개인키

(SD_{Pri})를 이용하여 전자서명을 수행한다. 수식(16)의 암호화는 대칭키 방식의 암호 알고리즘이다.

$$EncCData = Enc_{SKey}(CData) \quad (16)$$

$$Sign_{SD_{Pri}}(CData) \quad (17)$$

7) 사용자 중요정보 입력 요청

*SD*는 사용자 중요정보 입력 요청(Res_Service_AddlInfo)을 *SP*에 전송한다. 이 값은 암호화된 사용자 중요정보인 *EncCData*를 포함하며, 전자서명이 된 중요정보의 경우 전자서명 값인 $Sign_{SD_{Pri}}(CData)$를 포함한다.

8) 사용자 추가정보 입력 요청

만약 사용자가 중요정보 이외에 일반정보(*Data*)를 추가로 입력하는 경우 사용자는 *PC*의 입력 *Form*으로부터 추가정보를 입력(Req_Service_InputInfo)하고 *SP*에 전달한다. 단, 이때 *Form*을 통해 중요정보(*CData*)는 입력할 수 없다.

9) 사용자 추가정보 입력 정보 검증

*SP*는 사용자로부터 전달받은 암호화된 중요정보(*EncCData*)와 추가정보(*Data*)를 입력하기 위하여, 수식 (18)과 같이 *EncCData*를 복호화 하여 *CData*를 추출한다. 그리고 이 정보를 서비스에 업데이트한다. 만약 전자서명($Sign_{SD_{Pri}}(CData)$)이 포함된 경우, *SP*는 *SD*의 공개키(SD_{Pub})를 얻기 위하여 *CA*에게 *SD*의 인증서($Cert_{SD}$)를 요청한다. *CA*는 *SP*에 인증서($Cert_{SD}$)를 전송하고, *SP*는 *SD*의 공개키(SD_{Pub})를 추출하여 서명 값을 검증한다. 서명이 유효한 경우, *SP*는 전자서명에 대한 입력요청을 'Wait' 상태로 대기하고, (12)단계에서 확인 메시지의 응답을 기다린다.

10) 사용자 중요정보 입력 응답

*SP*는 *SD*와 *PC*에게 사용자 중요정보(*CData*) 입력이 완료(Res_Service_InputInfo)되었음을 알린다.

11) 사용자 중요정보 입력 확인

사용자는 자신이 입력한 중요정보(*CData*)가 정확한지 확인하고 이에 대한 확인 응답을 제공한다. 만약 전자서명을 한 경우, 사용자는 중요정보 입력 응답과 관련하여 해당 서명에 관한 검증결과를 *SP*로부터 전달 받는다. 사용자는 이 메지시를 확인하고, 검증결과가 유효한 경우 자신이 서명한 값과 *SD*의 어플리케이션 화면을 통하여 비교하여, 전자서명의 내용이 위·변조되지 않고 유효함을 확인할 수 있다.

12) 사용자 중요정보 입력 확인 응답

*SD*는 *SP*에게 사용자 중요정보에 대한 확인 응답(Res_Service_InputInfo_Result)을 한다. 만약 전자서명이 포함된 경우, 이에 대한 응답을 전송받은 *SP*는 (9)단계에서 Wait 중이던 전자서명에 대한 입력요청을 완료하고 이 정보를 서비스에 업데이트 한다.

Ⅳ. 안전성 분석을 위한 위협 분류 및 안전성 분석

4.1. 안전성 분석을 위한 위협 분류

제안하는 프로토콜 및 중요정보의 입·출력 방법의 안전성 평가를 위하여 평가 기준이 될 수 있는 기존에 알려진 대표적인 공격을 살펴본다. 표 4는 이를 바탕으로 각 공격 기법별 특징을 분석하였다.

표 4. 공격 기법별 특징
Table 4. Attack Methods Features

Layer	Description	Active/ Passive	Resource Access	Typical Attack
PC	User error induced	Passive	-	Phishing Attack
	Peeking typing information by keyboard	Passive	Read	KeyLogger Attack
	Peeking communication data	Active	Read/ Write	Trojan Attack
	Memory area operation(manipulation)	Active	Read/ Write	Memory Hacking Attack
Network	Transfer data sniffing	Passive	Read	Replay Attack
	Transfer data manipulation	Active	Read/ Write	Man-In-The-Middle Attack
etc	Indiscriminate Attack	Active	Write	Impersonation Attack

- 가장 공격(Impersonation Attack) : 부정한 사용자가 임의의 사용자 정보를 이용하여 정당한 사용자로 위장
- 메모리 해킹 공격(Memory Hacking Attack) : 공격자는 사용자의 *PC*에 메모리 해킹 공격 툴을 인스톨시켜 놓고, 사용자 *PC*의 메모리 영역을 조작할 수 있음
- 재생 공격(Replay Attack) : 공격자가 사용자와 서비스제공자 사이의 통신 내용을 가로채어 이를 다시 반복하여 정당한 사용자로 위장
- 중간자 공격(MITM : Man-In-The-Middle)) : 공격자가 사용자(*PC* 또는 *SD*)와 서비스제공자(*SP*) 사이에서 통신 내용을 가로채서 비밀정보를 가로채거나 정당한 사용자로 위장
- 키로거 공격(KeyLogger Attack) : 메모리 해킹 공격처럼 사용자의 *PC*에 키로거 프로그램을 인스톨시켜 놓아, 사용자가 입력하는 키보드 값을 전달받을 수 있음
- 트로잔 공격(Trojan Attack) : 공격자는 사용자의 *PC*에 트로잔 프로그램을 인스톨시켜 놓고 사용자의 세션을 관찰하고 허위 트랜잭션을 발생시킬 수 있음
- 피싱 공격(Phishing Attack) : 공격자는 사용자가 위조된 서비스제공자(*SP*)에게 접속하도록 접속을 유도하여 사용자는 위조된 사이트를 이용함

각 공격들은 공격 위치에 따라 크게 PC영역과 네트워크 영역으로 나누어지며, 공격의 성격에 따라 공격과정을 지켜보는 수동적 공격과, 공격에 적극 개입하는 능동적 공격으로 나누어진다. 능동적 공격은 사용자의 리소스에 쓰기 권한을 가지고 있어 그 위험이 더 크다고 할 수 있다. 이러한 공격으로부터 사용자는 로그인과 관련된 정보가 유출될 수 있으며, 서비스 이용 시 사용자가 제공하고 제공받는 사용자의 개인정보들도 유출될 수 있다. 또한 공격자가 리소스를 조작하는 경우 사용자는 비정상적인 서비스를 이용하여, 금융서비스의 경우 금전적 피해를 입을 수 있다. 또한 메신저 서비스와 같은 경우 유출된 정보를 이용하여 사용자 사칭으로 인한 2차, 3차 피해가 발생할 수도 있다.

4.2. 안전성 분석

제안하는 프로토콜 및 중요정보의 입·출력 방법이 어떻게 각 공격으로부터 안전한지 분석하기 위하여 위에서 나열된 대표적인 공격별로 분석하였다. 각 공격별 공격자의 최종 목표는 리소스의 접근이라 할 수 있다. 따라서 해당 리소스에 대한 안전성이 보장된다면, 그에 해당하는 공격으로부터 안전하다 할 수 있다.

- 피싱 공격 : 사용자는 서비스 가입단계에서 피싱 사이트를 구분하기 위하여 서비스제공자(*SP*)의 인증서 검증을 통하여 사이트의 진위여부를 확인한다. 사용자는 *PC* 뿐만 아니라 *SD*를 통하여도 확인하므로 피싱 공격으로부터 안전할 수 있다. 또한 전송되는 중요 정보는 *PC*를 이용하지 않으므로, 정보노출에 대한 위험성이 원천 차단된다. 서비스 이용단계에서 사용자는 서비스제공자(*SP*)의 인증서 검증 및 사용자와 공유된 씨드 비밀값(*SeedKey*)을 이용하여 생성된 세션키(*SKey*)를 통하여 암호화된 값을 전송한다. 따라서 이러한 값은 공격자가 생성할 수 없으므로 위·변조된 서비스는 제공될 수 없다.
- 키로거 공격 및 트로잔 공격: 사용자의 *PC*에 설치된 악의적인 프로그램을 통하여 사용자가 입력하는 키보드 내용 및 통신 전송내용 등 *PC*의 리소스를 가져가는 경우, 제안된 방법에서 모든 중요정보(*CData*)는 *PC*가 아닌 *SD*를 이용하여 취급되므로, 공격자가 가져가는 리소스는 사용자에게 큰 피해를 주기 어렵다. 만약 일반정보의 유출이 사용자에게 피해를 줄 수 있는 경우, *SP*는 해당 정보도 중요정보(*CData*)와 같이 *SD*를 이용하여 취급하여 피해를 막을 수 있다.
- 메모리 해킹 공격 : 공격자가 사용자 *PC*의 메모리 영역을 조작하여, 서비스 제공과 관련된 데이터를 조작하는 경우, 사용자는 *SD*를 통하여 해당 서비스 제공과 관련된 내용을 확인함으로써 *PC*의 데이터가 위·변조되었다는 사실을 인지할 수 있다. 특히 중요정보는 *SD*를 이용하여 취급되므로, 메모리 해킹 공격으로 위·변조할 수 있는 리소스는 큰 피해를 주기 어렵다. 만약 일반정보의 유출이 사용자에게 피해를 줄 수 있는 경우, *SP*는 해당 정보도 중요정보(*CData*)와 같이 *SD*를 이용하여 취급하여 피해를 막을 수 있다.
- 재생 공격 : 인증과정에서 사용자(*PC* 또는 *SD*)는 서비스제공자(*SP*)와 통신하는 세션마

다 시도-응답 방식을 통하여 각 객체별로 RN_{PC}, RN_{PC}, RN_{PC}를 생성하여 서로를 인증하고, 통신에 사용되는 비밀키인 $SKey$는 매 세션마다 새로 생성된다. 따라서 각 세션마다 랜덤 값과 세션비밀키($SKey$)값이 달라지므로 인증과 관련된 재생 공격은 매우 어렵다. 이러한 공격의 성공확율은 세션비밀키($SKey$)를 이용한 암/복호화 알고리즘의 안전성과 동일하다. 또한 계좌이체와 같은 전자서명을 재생 공격을 하는 경우, 각 전자서명에는 타임스탬프(TimeStamp) 값과 같은 정보가 있어, 동일한 전자서명으로 인한 중복된 계좌이체 등은 발생하지 않으며, 전자서명을 위조하는 것은 매우 어렵다. 전자서명 위조의 공격성공 확율은 전자서명 알고리즘의 안전성과 동일하다.

- 중간자 공격 : 공격자는 중간자 공격을 위하여 사용자에게 자신을 SP로 속여야 한다. 하지만 서비스 가입과정에서 공격자는 SP의 개인키(SP_{Pri})를 알 수 없으므로 중간자 공격을 할 수 없다. 또한 서비스 제공 과정에서도 사용자에게 발급된 씨드비밀($SeedKey$)을 알 수 없으므로 세션비밀키($SKey$)를 생성하기 어려워 공격자는 중간자 공격에 실패한다. 중간자 공격의 공격성공 확율은 세션비밀키($SKey$)를 이용한 암/복호화 알고리즘의 안전성과 동일하다.
- 가장 공격 : 공격자는 사용자의 SD를 가지고 있지 않으므로, 가장 공격을 할 수 없다. 만약 공격자가 자신의 SD를 이용하여 가장 공격을 하는 경우, 공격자는 사용자의 IMEI 정보를 알 수 없고 SD에 전달된 씨드비밀($SeedKey$)을 알 수 없으므로 가장 공격은 실패한다.

위와 같이 본 논문에서 제안한 방법은 기존에 알려진 모든 유형의 공격으로부터 안전하다. 이는 기존에 알려진 공격을 성격과 리소스의 접근 측면에서 분류한 표4를 기준으로 각 유형별 대표적인 공격에 대한 안전성을 분석했기 때문이다. 또한 제안한 방법은 공격자가 사용자의 PC영역을 제로데이 공격 등 해결책이 마련되지 않은 공격 방법으로 공격 하더라도, 사용자의 중요정보 및 로그인 과정은 안전하다. 이는 PC가 아닌 스마트 기기(SD)를 이용하여 중요정보를 취급하고, SD를 이용하여 로그인하기 때문이다.

V. 결 론

본 논문은 스마트 기기를 이용하여 사용자 인증 및 중요정보의 안전성 보증 방법을 제안하였다. 이를 위하여 다중 요소 및 다중 채널의 개념을 도입하였다. 즉, 스마트 기기의 IMEI 번호는 사용자가 가지고 있는 인증 요소이며, 비밀번호 설정 등을 통한 스마트 기기의 사용제한은 사용자가 알고 있는 인증 요소이다. 그리고 인증 과정에서 인증과 관련된 정보는 PC가 아닌 스마트 기기를 통하여 안전한 망으로 전달된다. 따라서 Two-Factor와 Two-Channel을 이용한 인증 방식이라 할 수 있다. 사용자는 스마트 기기의 카메라를 통하여 간편히 QR-Code를 촬영할 수 있고 이를 통하여 PC와 스마트 기기간의 통신이 이루어진다. 제안된 방법은 스마트 기기를 이용하여 로그인 및 중요정보를 취급하므로, 제로데이 공격과 같은 알려지지 않은 공격들로부터 사용자의 PC가 감염된 경우에도 사용자의 로그인 정보 및 중요정보가 유출되지 않아 안전하다. 특히 스마트 기기에서 전자서명이 이루어지므로 메모리 공격과 같은 사용자가 인지하지 못하는 조작 공격으로부터 생성될 수 있는 잘못된 전자서명도 예방할 수 있다.

제안된 인증 프로토콜 및 중요정보의 안전성 보증 방법은 다양한 인터넷 서비스에 적용 가능하여, 그 범용성이 넓다. 또한 인증과정 뿐만 아니라 사용자의 중요정보 취급 및 전자서명 등이 필요한 서비스 제공시에도 안전성을 보증하여, 서비스 전반에 걸쳐서 안전성을 높였다. 본 논문에서 스마트 기기는 안전하다고 가정하였지만, 스마트 기기를 대상으로 하는 악성코드 및 바이러스도 늘어나고 있어, 이에 대한 연구가 더 필요할 것으로 보인다.

References

[1] Bruce Schneier. "Two-factor authentication: too little, too late." *Commun. ACM* 48, pp. 136, Apr. 2005.

[2] Ziqing Mao, Florencio, D. Herley, C., "Painless migration from passwords to two factor authentication," *Information Forensics and Security (WIFS)*, 2011 IEEE

International Workshop on, Catalunya, Barcelona, pp. 1-6, Nov, 2011.

[3] 김영식(Young-Sik Kim), 임대운(Dae-Woon Lim), "스마트 카드를 이용한 서버 인증이 필요 없는 디지털 콘텐츠 보호 기법(Digital Contents Protection Without Server Authentication Using Smart Cards)," J-KICS vol.36, no.3, pp. 133-139, Mar, 2011

[4] 김현석(Hyun-Seok Kim), 김주배(Ju-Bae Kim), 정연오(Yeon-Oh Jeong), 한근희(Keun-Hee Han), 최진영(Jin-Young Choi), ″스마트카드를 이용한 패스워드 기반 인증시스템 정형분석(Formal Analysis of Authentication System based on Password using Smart Card),″ 정보과학회논문지. *Journal of KIISE*. 시스템 및 이론, pp. 304-310, Aug, 2009

[5] Xinyi Huang, Yang Xiang, Chonka. A., Jianying Zhou, Deng. R.H., ″A generic framework for three-factor authentication: preserving security and privacy in Ddstributed systems,″ *Parallel and Distributed Systems*, IEEE Transactions on, vol.22, no.8, pp. 1390-1397, Aug, 2011.

[6] Chun-I Fan, Yi-Hui Lin, ″Provably secure remote truly three-factor authentication scheme with privacy protection on biometrics,″ *Information Forensics and Security*, IEEE Transactions on, vol.4, no.4, pp. 933-945, Kaohsiung, Taiwan, Dec, 2009.

[7] 유한나(Han-na You), 이재식(Jae-Sik Lee), 김정재(Jung-Jae Kim), 박재표(Jae-Pio Park), 전문석(Moon-Seog Jun), "인터넷 뱅킹 환경에서 사용자 인증 보안을 위한 Two-Channel 인증 방식(A Study on the Two-channel Authentication Method which Provides Two-way Authentication using Mobile Certificate in the Internet Banking Environment)," J-KICS vol.36, no.8, pp. 939-946, Aug, 2011.

[8] Vapen. A., Byers. D., Shahmehri. N., ″2-clickAuth optical challenge-response authentication,″ Availability, Reliability, and Security, 2010. ARES '10 International Conference on, Krakow, Poland, pp. 79-86, Feb. 2010.

[9] Ben Dodson, Debangsu Sengupta, Dan Boneh, Monica S. Lam., "Secure, consumer-friendly web authentication and payments with a phone," *In Conference on Mobile Computing, Applications, and Services* (MobiCASE'10), pp. 17-38, Santa Clara, CA, USA, Oct, 2010.

[10] Jaesik Lee, C. H. Cho, M. S. Jun, ″Secure quick response-payment(QR-Pay) system using mobile device,″ *Advanced Communication Technology (ICACT)*, 2011 13th International Conference on, pp. 1424-1427, Seoul, South Korea, Feb. 2011.

[11] Kyeongwon Choi, Changbin Lee, Woongryul Jeon, Kwangwoo Lee, Dongho Won, ″A mobile based anti-phishing authentication scheme using QR code,″ *Mobile IT Convergence (ICMIC)*, 2011 International Conference on, pp. 109-113, Suwon, South Korea, Sep. 2011.

[12] Kuan-Chieh Liao, Wei-Hsun Lee, Min-Hsuan Sung, Ting-Ching Lin, ″A one-time password scheme with QR-Code based on mobile phone,″ *INC*, IMS and IDC, 2009. NCM '09. Fifth International Joint Conference on, pp. 2069-2071, Taichung, Taiwan, 25-27 Aug. 2009.

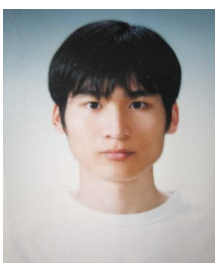
[13] Kuan-Chieh Liao, Wei-Hsun Lee, ″A novel user authentication scheme based on QR-Code,″ *Journal of Networks*, vol 5, no 8 (2010), pp. 937-941, Aug. 2010.

[14] Michiru Tanaka, Yoshimi Teshigawara, ″A method and its usability for user authentication by utilizing a matrix code reader on mobile phones,″ *Information Security Applications (WISA)*, 2006 Workshop on, LNCS 4298, pp. 225-236, Jeju Island, Korea, Aug, 2006.

[15] Yamamoto. N., Wakahara. T., ″A user attestation system using a cellular phone equipped with digital camera,″ *P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC)*, 2010 International Conference on, pp. 431-435, Fukuoka,

Japan, Nov. 2010.
[16] Wikipedia, "Smart device", "http://en.wikipedia.org/wiki/Smart_device", Wikipedia, June. 2012.
[17] Faldo, "Theories and methods of memory hacking," https://game-bot-aim-trainer-delphi.googlecode.com/files/Theories%20and%20Methods%20of%20Hacking.pdf, Dec, 2008.
[18] Widipedia, "QR-Code," http://en.wikipedia.org/wiki/QR_code, June 2012.
[19] Widipedia, "Transport Layer Security." http://en.wikipedia.org/wiki/Transport_Layer_Security, June 2012.

이 재 식 (Jae-sik Lee)



2005년 8월 가천대학교 컴퓨터공학과 졸업
2007년 8월 숭실대학교 컴퓨터공학과 공학석사
2007년 9월~현재 숭실대학교 컴퓨터공학과 박사수료
<관심분야> NFC, 개인정보보호, 인증 이론 및 시스템, 암호프로토콜

유 한 나 (Han-na You)

2008년 8월 평생교육원 정보보안전공 공학사
2010년 8월 숭실대학교 컴퓨터학과 공학석사
2010년 9월~현재 숭실대학교 컴퓨터학과 박사수료
<관심분야> 금융보안, 인증, 네트워크 보안

조 창 현 (Chang-hyun Cho)

2000년 2월 남서울대학교 정보통신공학과 졸업
2004년 2월 숭실대학교 산업기술정보대학원 전자 및 컴퓨터공학 석사
2010년 8월 숭실대학교 컴퓨터공학과 공학박사
<관심분야> 정보보호, 네트워크 보안

전 문 석 (Moon-seog Jun)

1981년 2월 숭실대학교 전자계산학과 졸업
1986년 2월 University of Maryland Computer Science 석사
1989년 2월 University of Maryland Computer Science 박사
1991년 3월~현재 숭실대학교 정교수
<관심분야> 정보보호, 네트워크 보안, 전자여권, 암호학