Expert Systems With Applications 44 (2016) 67–77

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

# Certificate sharing system for secure certificate distribution in mobile environment

Sundeuk Kim [a,b], Hyun-Taek Oh [b], Young-Gab Kim [c,*]

[a] Department of Computer Science and Engineering, Korea University, 145 Anam-ro, Seongbuk-gu, Seoul 136-701, Republic of Korea
[b] Common Network Group, Infra. Center, Samsung SDS, Gyeonggi-do 427-705, Republic of Korea
[c] Department of Computer and Information Security, Sejong University, 209, Neungdong-ro, Gwangjin-gu, Seoul 143-747, Republic of Korea

ARTICLE INFO

*Keywords:*
Certificate sharing system
Key distribution
Public key infrastructure
Mobile certificate

ABSTRACT

As mobile and Internet technologies evolve, mobile services (e.g., Internet banking, social commerce) continuously expand and diversify. In order to use these mobile services, it is essential that security services, especially distribution certificates (e.g., bank certificates), relevant to mobile devices be provided. Some approaches to providing distribution certificates between a user's mobile device and a personal computer (PC) have been proposed. However, the existing approaches do not guarantee that the certificate in the mobile devices same with the issued one from the PC, causing constraints on mobile services such as mobile phone banking and mobile commerce (M-commerce).

In this paper, we propose a novel approach that shares certificates securely without modification of the existing standard certificate format between a smartphone and a PC. We also implemented the certificate sharing system (CSS) in a virtual private network (VPN). The CSS provides strong end-to-end data security for the certificate with a key size of 192-bits which is able to guarantee an expiration date of three years. It also provides strong data security on physical devices with the use of device ID. The certificate that is shared between devices is available only through the CSS's authorization process. In addition, the CSS provides a flexible and extensible system for sharing certificates in enterprise environments. The CSS module of a PC was implemented by way of a standard web language, and the CSS module of a smartphone was developed with the assistance of mobile applications with a small size of 1210KB.



## 1. Introduction

Security is one of the biggest concerns when users utilize mobile services (e.g., Internet banking, social commerce) through the wireless internet (Ray & Biswas, 2011). Since the mobile service is so exposed, it requires measures to protect communications between users and service providers (Borrell, Robles, Serra, & Riera, 1999; M'Raihi & Yung, 2001; Tsaur, W. –J., 2012; Tseng, Yang, & Su, 2004). Furthermore, as mobile technologies evolve, mobile services continuously expand and diversify. Therefore, it is essential to provide security services relevant to mobile devices such as personal digital assistants (PDAs), and smartphones (Moon, Kim, Moon, & Baik, 2006).

In order to provide trustworthy remote access to business services, secure authentication systems need to be activated. Moreover, two-factor authentication (TFA or 2FA) (FFIE, 2005) should be used in order to access security-sensitive systems, especially banking services in mobile environments. TFA requires two means of identification, one of which is typically a physical token (e.g., a bank card), and the other is typically something memorized (e.g., a personal identification number (PIN)). Studies using a factor from two-factor authentication such as public key infrastructure (PKI) (Ou & Ou, 2007; Ray & Biswas, 2011), certificates (Lee & Lee, 2008), X.509 (ISO/IEC, 1997), X316-Chameleon (Saadi, Pierson, & Brunie, 2007), and short message service (SMS) (Shu, Tan, & Wang, 2009) are currently on-going.

Certificates play a critical role in mobile applications. By exploiting certificates, mobile applications are capable of accessing the trading or other M-commerce services (i.e., mobile commerce) on mobile environments (Bijani & Robertson, 2014). In the past, most TFA schemes using certificates were provided without the ability to share the same certificate between smartphones and personal computers (PCs). This is because there was no way to share the same certificate in both the smartphone and PC at the same time (Barbera et al., 2009; Chanson & Cheung, 2001; Critchlow & Zhang, 2004; Gao, Li, & Tu, 2004; Hunter, 2002; Jiang, Chen, & Shen, 2006; Laih & Chen, 2004; Rabinovich, 2010; Zhao & Tang, 2006). Therefore, users who wanted

* Corresponding author. Tel.:+82 269352424.
*E-mail addresses:* sundeuk.kim@samsung.com (S. Kim), handover.oh@samsung.com (H.-T. Oh), alwaysgabi@sejong.ac.kr, always.gabi@gmail.com (Y.-G. Kim).

http://dx.doi.org/10.1016/j.eswa.2015.09.003
0957-4174/

to use a certificate in a mobile device had to reissue the certificate issued in the PC. This would result in the certificate in the PC being revoked. Thus, a certificate that was issued on a PC could not be used in the smartphone simultaneously.

Even though most banks (Hyundai Securities Co., Ltd, 2015; KB Kookmin Bank, 2015; Samsung Securities Co., Ltd, 2015; Shinhan Bank, 2015; Wooribank, 2015), especially in Korea, provide a way to share a certificate between a smartphone and a PC, the PC has to be turned on and be in a state controllable by the user in order to share the certificate between the smartphone and the PC. In addition, when a certificate that is wired PKI on a PC is to be stored on a smartphone, the format of the certificate must be converted to a suitable format for the smartphone. Furthermore, each bank needs an individual system in order to share a certificate between a PC and a smartphone because the certificate stored in a PC is not the same as that of a certificate saved in a smartphone.

Furthermore, most of the existing approaches do not allow a PC's X.509 certificate to be used in a smartphone at the same time with higher data security and does not store the same X.509 certificate in the PC as that on the smartphone using standard methods. Previous approaches developed individual certificate sharing systems in a nonstandard way, and so they cannot be flexibly applied in enterprise environments. In addition, they cannot guarantee end-to-end forceful data security, nor can they provide strong physical security against copying. In this respect, existing methods do not provide flexible adaptive certificate sharing methods and they increase cost and time.

In order to overcome these limitations, in this paper, a novel method for securely sharing a certificate between a smartphone and a PC is proposed. We also implemented the certificate sharing system (CSS) in a virtual private network (VPN). The CSS proposed in this paper is an expert system in the area of mobile network security, especially focused on mobile certificate authentication, which can support strong security and usability to system-critical systems such as finacial systems, accounting systems, and interal security systems in mobile environments. It allows a user to share the X.509 certificate issued on the PC in its original form without any other processes. The CSS converts the X.509 certificate on the PC into binary codes and sends them to the smartphone, and also allows the same X.509 certificate that is saved on the PC to be stored on the smartphone. After the CSS safely stores the certificate that has been converted to binary codes on the database (DB) server, users can download them any time they desire. Thus, it is highly convenient for users. In addition, the CSS improves the security level for certificate data and has very strong physical security with its own encryption method. Our CSS can also be flexibly applied to systems in enterprise environments as it was developed with a web-standard language.

The rest of this paper is organized as follows: Section 2 surveys past proposals dealing with the distribution of certificates and its limitations. Section 3 presents the proposed approach, which is a CSS for sharing certificates between a PC and a smartphone. In Section 4, we describe the architectural design of CSS as well as its implementation in a VPN system. We also report on an evaluation result in Section 5. Finally, Section 6 concludes this paper.

## 2. Related work and problem definition

### 2.1. Sharing of X.509 certificate

Most of the banks (KB Kookmin Bank, 2015; Shinhan Bank, 2015; Wooribank, 2015) and finance companies (Hyundai Securities Co., Ltd, 2015; Samsung Securities Co., Ltd, 2015) in South Korea provide customers with a system for sharing certificates between a smartphone and a PC. However, in order to share certificates using this system, the smartphone and PC not only have to be turned on, but they also have to be in the operable state. Also, sharing of the certificate occurs in a single cycle. Therefore, if the PC is turned off, the user cannot store the certificate from the PC on the smartphone and cannot share the certificate at any time. Furthermore, a certificate shared to a smartphone through one bank cannot be used in other banks because the certificate saved to the smartphone is not the same as the one on the PC. Also, these systems have to be implemented separately for each bank since the certificate stored on the smartphone is not in standard PKI format. Our CSS overcomes these limitations. The PC does not need to be turned on as binary codes are saved to a DB server when a user shares a PC's certificate with a smartphone. In addition, the user can share the PC's certificate with the smartphone at any time, and the certificate saved to the smartphone is the same as the one on the PC. Also, by means of the CSS, all banks can use the same certificate used by the smartphone without implementation of an individual certificate system for each bank.

### 2.2. Wired/wireless PKI

To issue an X.509-based PKI and verify the issued certificate in a wired environment, the following requirements should be satisfied (Housley, Polk, Ford, & Solo, 2002; Myers, Adams, Solo, & Kemp, 1999; RSA, 2000; RSA, 2012): (1) To use the certificate message protocols (CMPs) made in a PC or Mobile as a certificate, the CMP should include the public key. An end-entity's reference number (e.g., an ID) and an authorization code (e.g., a password) when creating the CMP in the mobile phone should be used to create the CMP in the mobile phone. (2) A private key corresponding to the public key should be used to make and request the CMP; and through this, Proof-of-Possession (PoP) should be satisfied. (3) The method that the certificate request message is securely communicated with a certificate authority (CA). (4) Use a public key in the CA to verify the CMP, which is created from the mobile phone, and then create a certificate message by using a signature key in the CA. The public key delivered from the mobile phone is included in this certificate. (5) Use a private key from the verified certificate message in the mobile phone to encode the certificate and then create its unique final certificate.

The procedure as outlined above should be performed in order to issue an X.509 certificate for a smartphone in wireless environments. However, it is difficult to implement the procedure in the mobile environment due to the limitations of mobile devices such as low electricity, low memory, and low bandwidth. Also, the certificate issued on a PC will be revoked when that certificate is being issued on a smartphone (Critchlow & Zhang, 2004; ISO/IEC, 1997). It is inefficient and inconvenient for users since the users must issue a certificate again in order to use the certificate with a smartphone. Thus, various studies to solve these issues and save certificates to smartphones in more efficient and safer ways are actively being conducted (Lee & Lee, 2008; Lee & Park, 2005; Ray & Biswas, 2011; Roland, Langer, & Scharinger, 2012; Yan et al., 2006; Yoo, Yoo, Park, & Ryou, 2011). However, existing research efforts do not provide any means of sharing a certificate from a PC to a smartphone. Even though previous studies provided means of issuing and storing an X.509 certificate on a smartphone, it is not the same certificate as the X.509 certificate that is issued in a PC (Lee & Lee, 2008; Ou & Ou, 2007; Ray & Biswas, 2011; Yan, Sun, Wang, Kao, & Yuan, 2006). Moreover, existing methods do not allow a smartphone to have the same X.509 certificate as the PC, and they are not standard (Lee & Park, 2005; Roland et al., 2012; Yoo et al., 2011). As a result, these research efforts are difficult to apply in the enterprise environment because they have additional implementation cost and are time-consuming. To overcome these weaknesses, we used a special conversion module in the CSS to store the same X.509 certificate on a PC and on a smartphone in our simulation. In this regard, the CSS provides a flexible and extensible system for sharing certificates in enterprise environments.