Pseudonymous PKI for Ubiquitous Computing

Ke Zeng

NEC Laboratories, China
zengke@research.nec.com.cn

Abstract. Conventional PKI is the most effective and efficient solution to non-repudiation. But, it also puts user privacy in danger because the user's activities could be tracked via the unique public-key and certificate he presents in multiple transactions. Pseudonymous PKI (PPKI) solution achieves non-repudiation as well as privacy protection at the same time by providing Pseudonymous Public-Key (PPK) and Pseudonymous Certificate (PCert) that are computed by the user without CA intervention. PPK is as effective as conventional public-key in terms of non-repudiation. Furthermore, the PPKI solution is very efficient in terms of the size of PPK and PCert, and is scalable in terms of certification authority overhead. Therefore PPKI is particularly suitable for ubiquitous computing environments where authenticity, non-repudiation, privacy protection, efficiency, and scalability are key requirements.

1 Introduction

In its simplest form, conventional Public-Key Infrastructure (PKI) consists of the Certification Authority (CA) and registered certificate users of CA. Each PKI user has his own public-key and private-key pair. The basic task of CA is to issue certificate to each user's public-key. The CA also maintains and applies Certificate Revocation List (CRL) to revoke the certificates of misbehaving users. For decades, conventional PKI performs well in many kinds of traditional businesses e.g. secure email and access control that entail various security guarantees.

With the proliferation of smart gadgets, appliances, mobile devices, PDAs, and sensors, ubiquitous computing environments may be constructed of such interconnected devices and services, which promise seamless integration of digital infrastructure into our everyday lives [1]. Conventional PKI can be applied to emerging ubiquitous computing environments to resolve security issues, e.g. non-repudiation [2]. But, it may incur side effects on user privacy. Consider two peers (which may be users or devices) working with each other in a ubiquitous computing environment. For instance, a mobile phone approaches an Access Point (AP) of wireless LAN and tries to surf the wireless LAN. For another instance, a PDA approaches a printer and tries to print a document. For a third instance, a laptop approaches a TV set and tries to render a film on the TV. In all these cases, it may be necessary for the peers to authenticate each message received. At first glance, if each peer has a CA certified certificate, it suffices for conventional PKI to fulfill security requirements. Unfortunately, the single public-key embedded in a conventional certificate is effectively a unique identifier of the

A.S. Atzeni and A. Lioy (Eds.): EuroPKI 2006, LNCS 4043, pp. 207 – 222, 2006.

key holder and hence can jeopardize the key holder's privacy. In aforementioned examples, the single public-key makes it easy for the AP, the printer or the TV set to identify which key holder is on-site, and consequently infer the interest and the behavior pattern of the key holder. Conventional PKI was designed in an era when privacy was not an issue for businesses that needed it for security reasons hence privacy protection was not taken into consideration in conventional PKI design.

Conventional PKI is so far the most effective and efficient solution for non-repudiation message transferring mandated by many ubiquitous computing applications [3, 4]. In order to keep enjoying the merits of conventional PKI for non-repudiation while resolve the intrinsic privacy issue caused by the unique public-key, solutions such as anonymous public-key [5, 6] and incomparable public-key [7] have been proposed. These solutions are suitable for DLP [8] based public-key cryptosystems. Given generators g, h_1, and h_2 of some finite cyclic group and the private-key x, let $(y_1 = g, y_2 = g^x)$ be the root public-key. Two anonymous public-keys could be generated as $(y_1^{r_1}, y_2^{r_1})$ and $(y_1^{r_2}, y_2^{r_2})$ [5], where r_1 and r_2 are different random integers, while two incomparable public-keys could be generated as (h_1, h_1^x) and (h_2, h_2^x) [7].

It's not explained in Waters et al. [7] how to generate certificates for incomparable public-keys. While in Oishi et al. [5], the anonymous public-keys are supposed to be generated by CA and naturally CA will issue disjoint certificates to different anonymous public-keys. In ubiquitous computing environments, particularly in large-scale mobile computing environments, the proposal of [5] will cause huge amounts of peer requests for anonymous public-keys and certificates, and huge amounts of queries against the CRL. This can result in heavily loaded CA and slow peer computation speed. Hence, the proposal of [5] cannot scale and is evidently inappropriate for ubiquitous computing environments.

Most recently, Ateniese et al. [9] proposed another solution that partially resolves the certificate issue of the anonymous public-keys. Simply speaking, the proposal of [9] enables each peer to compute anonymous public-key as well as the corresponding certificate, all by the peer itself. This apparently distributed solution is much more efficient than that of [5] because the CA is totally free from generating anonymous public-key and corresponding certificate for its users. Hence the proposal of [9] is scalable and should be more applicable in ubiquitous computing environment. However, the proposal of [9] doesn't provide a full solution for ubiquitous computing environments because it lacks tracing and revocation capabilities against misbehaving peers. Finally, a solution that satisfies security and privacy requirements with the smallest possible certificate size and fastest possible message authentication rate is highly desirable for ubiquitous computing environments.

In the following, Pseudonymous PKI (PPKI) solution for ubiquitous computing environments is presented. The key advantages of PPKI are three. First, each peer could generate distinct Pseudonymous Public-Key (PPK) and corresponding Pseudonymous Certificate (PCert) by itself without any involvement of CA. Second, the CA is equipped with efficient tracing and revocation mechanisms. Third, PPKI is very efficient in terms of the size of PPK and PCert and the time for message authentication.