Authentication Latency Reduction Technique for Secure and Seamless Ubiquitous Services

Ki-Woong Park and Kyu-Ho Park
Computer Engineering Research Laboratory
School of Electrical Engineering and Computer Science
Korea Advanced Institute of Science and Technology
{woongbak,kpark}@core.kaist.ac.kr

Abstract

In realization of a ubiquitous and mobile computing environments, provisioning and managing high security have become a challenging problem. This paper presents our effort to overcome the challenges in a computationally efficient way through a new single sign-on protocol and its security infrastructure. Since its inception, Public Key Infrastructure (PKI) has been widely used as a popular solution for securing Internet-wide transactions, thanks to its digital signature and non-repudiation features. In an attempt to expand Public Key Infrastructure (PKI) usage to a ubiquitous and mobile computing environment, where various devices with a severe resource constrained device (8-bit processor) frequently interact with each other or a surrounding infrastructure, we found that the deployment of the PKI was severely hampered by the fact that costly PKI asymmetric key operations should be performed on a resource-constraint device, leading to user-obstructive latency or an additional circuitry for the operations. In this paper, we present the authentication reduction technique to provide a seamless and secure authentication, and offer a protocol analysis in terms of user authentication latency and the completeness of the protocol. According to the performance evaluation, the authentication latency of our infrastructure (which averages 0.082 sec) is much shorter than the authentication latency of a conventional PKI-based authentication latency (which averages 5.01 sec).

1 Introduction

A full realization of ubiquitous services in a way that security is not compromised comes at a price such that a few major changes in an existing computing environment are necessary. Most of all, mutual authentication becomes essential due to anonymity and mobility of users in a ubiquitous environment. Unfortunately, conventional authentication systems based on widely used RFIDs or smart cards are limited when they are used for enabling high security and dynamic authentication especially in a ubiquitous environment. They do not provide a way to measure physical proximity among devices for location-based services and do not support an interactive communication capability between RFIDs or smart cards due to the limited complexity of circuitry.

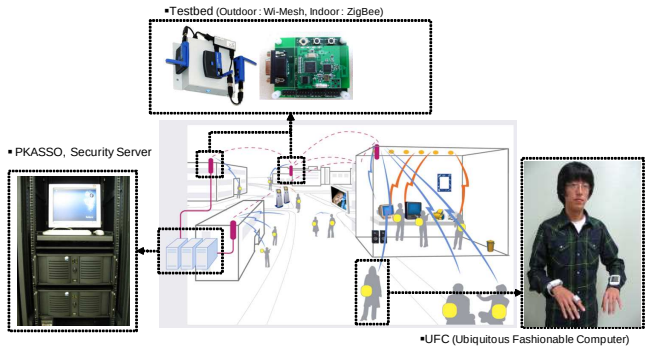


Figure 1. Overall Architecture of U-TOPIA

As shown in Figure 1, in our ubiquitous environment, we have been developing a wearable computer[1] and its interoperable computing environments where various devices such as U-Kiosks, U-Print, Zigbee-enabled appliances, etc are deployed to provide a user with ubiquitous services, mainly focusing on a university campus [1, 3]. The ubiquitous computing environment was named as U-TOPIA, where 'U' stands for 'ubiquitous' and 'TOPIA' stands for 'place' in Greek. In the ubiquitous computing environment, pervasive devices frequently take part in communication with other previously unknown devices for providing services, exposing themselves to an unfortified and insecure environ-

[1] UFC : Ubiquitous Fashionable Computer [1, 2]

ment. Therefore, the importance of security in the ubiquitous computing environment cannot be overstated; it needs to be addressed in order to attain confidentiality of a user living in the environment [4]. As a fundamental way to enable security, authentication and authorization are the two most widely used mechanisms among devices.

By thoroughly investigating our ubiquitous environment and conventional security systems, we identified two critical limitations of a conventional PKI-based security system. The limitations and our approach are described as follows:

1. **Obstructive Authentication Latency in a Mobile Device with Restricted Computing Power**: There is a challenge with clients where mobile devices with restricted computing power demand frequent and dynamic authentications over the PKI. To address this latency problem, we propose a security infrastructure that is based on the PKI and a single sign-on (SSO) protocol. The proposed security infrastructure provides users with a seamless and secure system of authentication and key distribution. In the proposed infrastructure, we devised a delegation server that offloads complex PKI operations from mobile devices to the infrastructure so as to keep the hardware and software complexity of the devices as low as possible. By using the proposed authentication mechanism, we can also achieve a more secure mechanism in line with newly emerging security policy requirements without replacing hardware or software components of devices.

2. **Support for Digital Signature and Non-Repudiation**: Even though a digital signature mechanism and a non-repudiation mechanism are essential functions in security applications, a conventional delegation mechanism cannot support these two mechanisms against malicious user behavior. Because user authentication transactions are already delegated, an authenticator cannot prove the fact that the authentication transactions are really intended by users. Our proposed protocol named PKASSO[2] and the security infrastructure can provide the two mechanisms by means of a referee server that generates binding information between a device and authentication messages [5].

The remainder of the paper is organized as follows. In Section 2, we present overall system design and components of the proposed security infrastructure. The ubiquitous services coupled with PKASSO are illustrated Section 3 and the evaluation of performance of PKASSO is presented in Section 4, and Section 5 concludes this paper.

[2] PKASSO: Public Key-based A^3-providing Single Sign On

2 Design and Component of PKASSO

2.1 Overall System Design

We incorporated the PKINIT protocol into our system as an underlying security infrastructure in an attempt to provide a secure key distribution and a way to efficiently and cost-effectively manage a large number of devices and sensors in a ubiquitous environment. The PKINIT protocol is an extension of the Kerberos protocol so as to enable the public key based authentication between a user and key distribution center instead of using the symmetric key based authentication. However, the PKINIT protocol has two drawbacks: first, it cannot provide a digital signature and non-repudiation because each authentication for a service device is accomplished by Kerberos-based authentication; second, it has obstructive authentication latency (minimum of 0.74 sec)[10] because it requires public and private key operations on the user's device whenever the user moves to another Kerberos realm to get a TGT. To overcome these drawbacks, we based the design of our security infrastructure and protocol on the following two principles:

- **A computationally efficient PKI-based SSO Protocol:** the requirement of users to sign-on for each service device on each occasion will severely hamper the usability of a diminutive security device with limited computing power. The widely used SSO technology can greatly relieve the poor usability problem by obviating the need for repeated sign-on procedures; hence, we adopted the SSO technology into our underlying security infrastructure. To provide the SSO technology and non-obstructive authentication latency, we propose a computationally efficient PKI-based SSO protocol that is based on a delegation mechanism which uses a proxy certificate [6]; the proposed protocol also provides identical security functionalities as the PKI.

- **A delegation server and a referee server:** for our SSO protocol, we devised a delegation server and a referee server. The delegation server is responsible for performing prohibitively expensive PKI operations on behalf of a diminutive security device to minimize the computational overhead of the security device. The referee server, which is designed to provide a computationally efficient digital signature and non-repudiation, generates binding information between security devices and authentication messages, and retains the information in its local storage for future accusation.

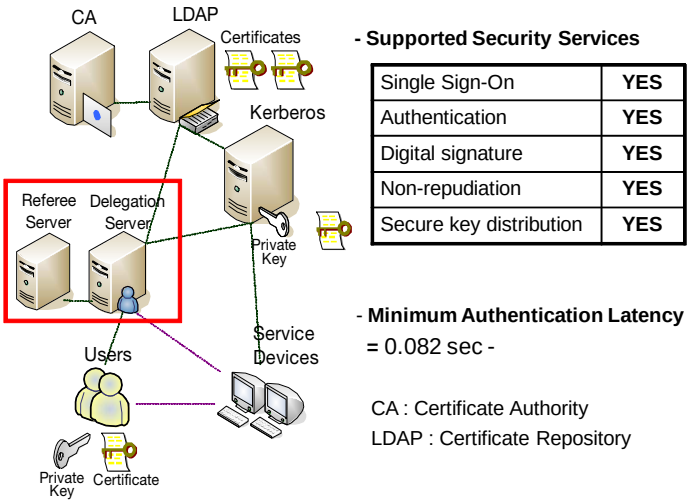


Figure 2. The proposed security infrastructure, PKASSO

2.2 The Proposed PKASSO Security Infrastructure

Figure 2 shows the overall architecture of our PKASSO security infrastructure. The five major components of our architecture are the PKINIT protocol, the user, the service device, the delegation server, and the referee server.

- **User** is a mobile entity that receives provided services in our ubiquitous security environment. To utilize services such as authentication, authorization, and accounting in the environment, each user should carry a diminutive security device called a PANDA, which is a type of a smart card equipped with Zigbee-based low power intercommunication capability and a location-sensing capability [9].

- **Service device** permeates the surroundings for the provision of services. The service device in our environment has a Zigbee communication module to interact with users and to authenticate the identity of users.

- **Delegation Server** is designed to offload complex PKI-related operations from a user to the infrastructure, making it possible to develop users' security device, PANDA, with cheap, simple hardware. The delegation server also maintains all the proxy certificates which contain the private keys and public keys that are delegated and signed by a user. Upon entering the security infrastructure for the first time, the user delegates the user authentication operations to the delegation server by following RFC3820 [6]. The delegation server subsequently takes over all the authentication operations until the users' proxy certificate expires.

- **Referee Server** provides a non-repudiation mechanism to combat malicious user behavior. The non-repudiation mechanism takes effect as long as the user uses its own private key in an authentication process. However, after delegating its operations to the delegation server, the user no longer uses its private key any more as a means of halting the provision of the non-repudiation mechanism. We therefore devised a referee server so that, even during the delegation process, we could bring back the mechanism into our system. The referee server investigates all authentication messages, generates binding information on the fly between a user and the authentication messages, and retains the information signed by the referee server's private key in its local storage for future accusation. Owing to the referee server, our proposed security infrastructure can provide a symmetric key-based non-repudiation mechanism that is computationally efficient on the user's mobile devices.

2.3 Overall Authentication Process of the PKASSO

Figure 3 shows the overall process of authentication on the basis of our security infrastructure. The proposed delegation server and the referee server make it possible to authenticate users with only two symmetric key operations on the user's device, thereby providing the following level of security, which is identical to the security of the PKI:

1. The service device sends a challenge message to a user who intends to receive a service.
2. The user generates an authentication request message (with two symmetric key operations) and sends it to the delegation server.
3. The delegation server performs transactions for verification and authentication with the referee server and the PKINIT protocol on behalf of the user.
4. By using a received authentication message from the PKINIT protocol, the delegation server makes a response message and transmits it to the service device.
5. The authentication is completed on the arrival of a confirming message from the user.

3 Applications coupled with PKASSO

In this section, we introduce ubiquitous services coupled with PKASSO. The implemented service device is shown

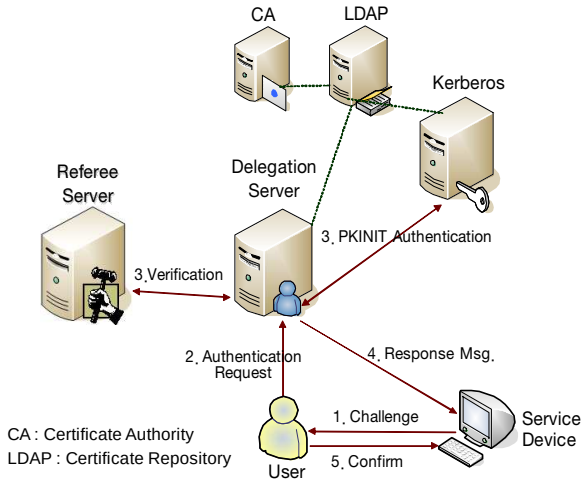
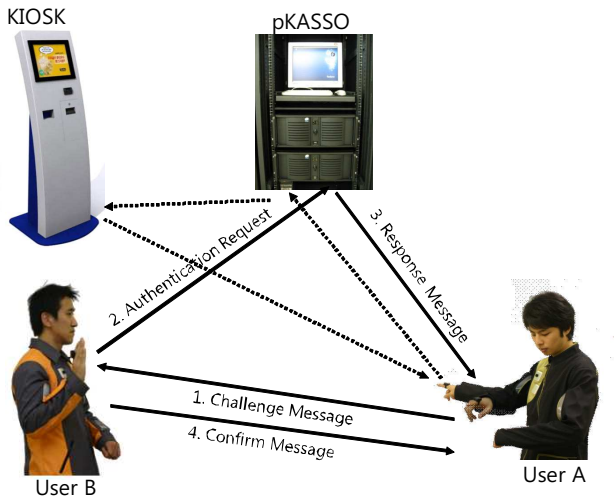


Figure 3. Authentication flow description with the delegation server



Authentication Latency of pKASSO: min. 0.082 sec
Authentication Latency of conventional PKI : min. 5.02 sec

Figure 4. Authentication flow description with PKASSO

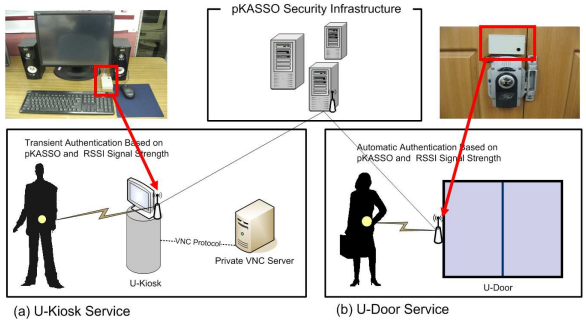


Figure 5. Ubiquitous Services (U-Kiosk Service, U-Door Service) based on PKASSO

in Fig.5. The service device is applicable to various applications such as services below making the best use of it.

In order to explain the authentication flow of PKASSO in a practical situation, we will take the following scenario: there are two UFC users, 'User A' and 'User B'. 'User A' takes a picture and 'User B' wants to see the picture by downloading it from 'User A' securely. A sequence of authentication for secure communication necessary to realize the scenario is shown in Figure 4. In the figure, each operation is denoted by the number and each operation corresponding to each number will be detailed below:

1. User A sends a challenge message to user B who wants to communicate with user A.
2. User B generates an authentication request message (two symmetric key operations) and sends it to PKASSO, and it performs transactions for verification and authentication on behalf of user B.
3. PKASSO makes a response message and transmits it to user A.
4. The authentication is completed with the arrival of a confirming message from user B.

Imagine that one sunny day, a student, who either carries PANDA in her pocket or wears a UFC docked with PANDA, strolls around campus. When she arrives at the class room, its door is opened automatically after checking her identity **(U-Door Service)**. She glances at one of the PCs nearby a window and walks to it. As soon as she sits on a chair in front of the PC, her previous working environment from her dormitory a couple of hours ago is automatically restored so as to continue to do her assignment. Suddenly, her cell phone starts ringing; she runs out of the class room without logging out at that moment. And all her working environment is automatically stored safely in order to protect her confidential information from being exposed **(U-Kiosk Service)**.

- U-Kiosk Service : U-Kiosk is a kind of service that provides the mechanism to use a public PC just like users' private PC so that U-Kiosk users can control resources of the private PC after users' authentication over pKASSO. To achieve it, U-Kiosk authenticates the PANDA holders and makes the connection between U-Kiosk and the private PC is maintained by Virtual Network Computing (VNC). Due to the high

Platform	Cryptography		Operation Time
• PANDA • Service Device - CPU : ATmega1280 - RAM : SRAM 256KB	RSA 1024bit	Private Key	Avg. 4723ms
		Public Key	Avg. 226ms
	AES 128bit	Encryption	Avg. 3ms
		Decryption	Avg. 3ms
	Hash Function	SHA-1	Avg. 6ms
• Server - CPU : Xeon 3.2GHz - RAM: 4GB	RSA 1024bit	Private Key	Avg. 2.917ms
		Public Key	Avg. 0.170ms
	AES 128bit	Encryption	Avg. 0.006ms
		Decryption	Avg. 0.006ms

Figure 6. Processing times of encryption/decryption for each algorithm and operation environment

mobility of the user, the U-Kiosk device cannot convince that the current user using the service is the same user who supplied legitimate proofs for his/her authenticity a few minutes ago. To cope with the problem, the device providing the service checks the physical proximity of a user by sensing signal strength (RSSI of ZigBee); if a user moves away without properly terminating his/her service sessions, the device automatically shutdowns the sessions to protect the users' confidential information from being exposed.

- U-Door Service : In this application, PANDA is used to authenticate insiders and outsiders of a campus over pKASSO. It may be possible that only insiders and exchange students in the campus are permitted to enter a specified section such as a dormitory.

4 Performance Evaluation of PKASSO

In this section, we present the performance results obtained with our prototype implementation of PKASSO. First, we demonstrates the overall experimental environment. We then describe the cryptography operation experiment and the operational efficiency of the authentication protocol to evaluate the performance of PKASSO in terms of authentication latency.

4.1 Cryptography Operation Experiment

The first experiment measures the cryptography processing time on Zigbee devices (PANDA, a service device) equipped with an ATmel 8-bit processor(16 MHz) [**?**] and a server (processor: Xeon 3.2 GHz; RAM: 4 GB). Figure 6 compares the processing time of an RSA 1024 bit algorithm as an asymmetric key operation, an AES 128 bit algorithm as a symmetric key operation, and an SHA-1 algorithm as a hash function. It shows that the time required to decrypt and encrypt a 128 byte block of data with the RSA 1024 bit algorithm is 4723 ms and 226 ms, respectively, on Zigbee devices. On the other hand, the time required to encrypt and decrypt the block with the AES 128 bit algorithm on the device is 3 ms because an AES accelerator is embedded in the Zigbee communication module (CC2420 [**?**]). Furthermore, SHA-1 needed 6 ms of operation time to generate a challenge message. On the server side, the cryptography operation time is reduced drastically by a high-performance processor and a huge memory.

System	Mobile			Server			Total operation Time
	Pu	Pr	S	Pu	Pr	S	
PKIX(RSA-1024bit)	2	2	1	2	0	0	5178.34 ms
Kerberos	0	0	8	0	0	6	24.04 ms
M-PKINIT TGT	1	1	7	1	1	5	4973.12 ms
M-PKINIT SGT	0	0	8	0	0	4	24.02 ms
PKASSO Delegation	2	1	2	4	5	2	5196.28 ms
PKASSO TGT	0	0	5	1	1	12	18.16 ms
PKASSO SGT	0	0	5	1	1	7	18.13 ms

Pu: The number of public key operation for each authentication
Pr: The number of private key operation for each authentication
S: The number of symmetric key operation for each authentication

Figure 7. The number of public/private keys and the symmetric key operations with total operation time for each protocols (PKIX, Kerberos, M-PKINIT, and PKASSO)

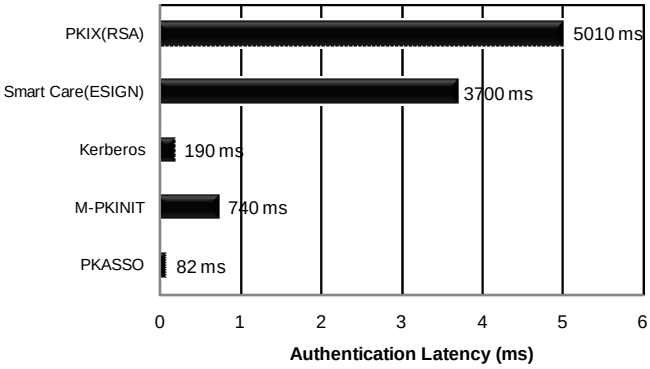


Figure 8. Authentication latency for PKIX(RSA), Smart Card, Kerberos, M-PKINIT, and PKASSO

4.2 Performance Evaluation

Figure 8 illustrates the improvement of the authentication latency with our scheme and compares it with a general PKIX operation equipped with a PANDA and a smart card [**?**]. If an RSA-1024 bit algorithm is processed on a PANDA equipped with an 8-bit processor (16 Mhz) [**?**], the authentication latency averages 5.01 sec. Furthermore if an authentication with Kerberos and M-PKINIT is executed on the above platform, the authentication latency averages 0.19 sec and 0.74 sec, respectively, because in order to obtain a TGT the Kerberos authentication protocol uses symmetric key operations and M-PKINIT uses asymmetric key operations. With the M-PKINIT protocol, a user can get an SGT without asymmetric key operations; hence, the authentication latency of M-PKINIT is much shorter than that of PKIX. The latency of a contact-type smart card is estimated to be 3.70 sec[**?**], which is faster than the latency of PKIX on our security device. In the case of PKASSO, even though the delegation operation of PKASSO takes longer than a general PKIX authentication (5.19 sec), the authentication latency of PKASSO with a PANDA can be shortened to 0.082 sec for a specified period after the delegation. As described in the previous section, the major reduction of the authentication latency is due to the offloading of complex operations from the devices to the infrastructure. As a result, we can minimize the authentication latency from an average of 5.01 sec to 0.082 sec without compromising the security level of PKIX.

5 Conclusion

This paper presented our effort in designing a new PKI-based security infrastructure, PKASSO, that offers an efficient authentication technology for an ubiquitous environment, wherein a large number of devices and sensors are scattered for providing various services. Our security infrastructure features two main achievements: 1) PKI-based single sign-on protocol especially tailored for managing efficiently a large number of devices and sensors in the ubiquitous environment, 2) an intelligent delegation server with a newly devised referee server that ensures non-repudiation of any transaction between a delegator and delegatee. As a consequence, our infrastructure enables a cost-effective but uncompromisingly secure development of a diminutive security device. Furthermore, our delegation mechanism significantly improves an authentication latency as well. According to the performance evaluation, the authentication latency(Avg. 0.082sec) is much shorter than a contact type smart card(Avg. 4.31sec) and a general PKI authentication latency(Avg. 5.01sec). As a result, our security infrastructure and protocol can be applied to the ubiquitous security environment. Based on our design and performance evaluation, we developed PANDA and a security infrastructure, PKASSO, and are currently implementing services for a ubiquitous campus.

References

[1] K. H. Park and UFC Group, "UFC: A ubiquitous fashionable computer," in International Conference on Next Generation PC, 2005.

[2] J.Lee, S.H Lim, J.W Yoo, K.W Park, H.J Choi, K.H Park, "A Ubiquitous Fashionable Computer with an i-Throw Device on a Location-Based Service Environment," In Proc. 21st IEEE Symposium on Pervasive Computing and Ad Hoc Communications, Vol. 2, pp. 59-65, May 2007.

[3] H. Seok, K.W Park, S.S. Lim, and K.H. Park, "Implementation of U-Kiosk based on PANDA and VNC," in 33th KISS Conference on Computer System, October 2006.

[4] S. L. Jason I. Hong, Jennifer D. Ng and J. A. Landay, "Privacy risk models for designing privacy-sensitive ubiquitous computing systems," in Proceedings of the 2004 conference on Designing interactive systems: processes, practices, methods, and techniques, 2004.

[5] K.W Park, H.C Seok, and K.H Park, "pKASSO: Towards Seamless Authentication providing Non-Repudiation on Resource-Constrained Devices," In Proc. 21st IEEE Symposium on Pervasive Computing and Ad Hoc Communications, Vol. 2, pp. 105-112, May 2007.

[6] S. Tuecke, V. Welch "Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile" RFC3820, 2004

[7] Tung, B., et al., Public Key Cryptography for Initial Authentication in Kerberos, 2001: http://www.ietf.org/internet-drafts/draft-ietf-cat-kerberos-pk-init-12.txt.

[8] K.W Park, S.S. Lim, H.C. Seok, and K.H. Park, "Ultra-Low-Power Security Card, PANDA, for PKI-based Authentication and Ubiquitous Services", Proceedings of Conference on Next Generation Computing, November 2006, pp.367-373.

[9] Z. A. B. of Directors, "ZigBee Specification v1.0." ZigBee Alliance,2005.

[10] K.W Park, S.S. Lim, H.S. Song, and K.H. Park, "A New PKI-based Single Sign-On Protocol for a Diminutive Security Device, PANDA, in a Ubiquitous Security Environment", 8th International Symposium on Systems & Information Security, November 2006.

[11] J. Patarin, L. Goubin, N. Courtois, "$C^{*\pm}$ and HM: Variations around two schemes of T. Matsumoto and H. Imai, in Advances in Cryptology", Proceedings of ASIACRYPT'98, LNCS n1514, Springer, 1998, pp. 35-49.

[12] Harbitter, A. and Menasce, D. A., "The performance of public key enabled Kerberos authentication in mobile computing applications", Proc. of the 8th ACM conference on Computer and Communications Security 2001.

[13] Asad Amir Pirzada and Chris McDonald, "Kerberos Assisted Authentication in Mobile Ad-hoc Networks" ACM International Conference Proceeding Series; Vol. 56 , 2004