Appl. Math. Inf. Sci. 6 No. 2S pp. 573S-578S (2012)

Applied Mathematics & Information Sciences
An International Journal

Using the Smartphone as a Personal Information Security Center

Yung-Tsung Hou
Department of Information Management, National Formosa University, Taiwan
Corresponding author: Email: ythou@nfu.edu.tw

Received June 22, 2010; Revised March 21, 2011; Accepted 11 June 2011
Published online: 1 January 2012

Abstract: With increasing power, the smartphone has become a new computing platform. A smartphone can efficiently execute a complex computing job, such as public-key encryption, which was usually run on workstations. This work presents a security management framework that uses a smartphone as a personal information security center (PISC) coordinating other devices of the user. The proposed framework is composed of two major parts: PISC key management and PISC security service coordination. In PISC key management, the smartphone manages the public key exchange between different users' smartphones. The key exchange utilizes a human-perceivable video channel to transfer keys and utilizes the phone camera to create an undeniable cryptographic nonce for key verification. A PISC phone also acts as a coordinator to provide keychain synchronization, authentication, and confidentiality services to all the user's devices. With the use of a PISC phone, a user can efficiently govern the security management of all her devices with little burden.

Keywords: Smartphone, Security, Key Management, Key Exchange

1 Introduction

Cryptography is the foundation of the modern computer and communication security. Among all the tools of a security system, encryption is the most important building block and also a primary method to protect valuable information [1]. By far, symmetric encryption and public-key encryption are commonly used in a cryptographic system. For both encryption methods to work, two communication users must exchange the shared key or public keys in advance. Therefore, the strength of a cryptographic system relies on the means of delivering keys to users, i.e. the key distribution. This paper discusses how to use smartphones for the key exchange and personal security management.

The first part of this work, Personal Information Security Center (PISC) key management, considers how to exchange public keys safely and manage the key rings with a trust model. If two parties have previously and recently used a key, one party can transmit the new key encrypted by the old key. However, in the case that users do not have old shared keys, the common way to distribute new keys is either to deliver the key physically or to trust a third party to do that. However, for common users, exchanging keys through a third party is not straightforward [2] and manual key delivering is cumbersome and impractical due to the lack of a suitable handy device. But with the increasing capability of smartphones, delivering keys physically is now a reasonable way for key distribution.

The proposed PISC key management does not require a third party. Instead, PISC key management takes advantage of smartphones to create a face-to-face key exchange method. The PISC key exchange protocol includes two major phases: the transfer phase and the verification phase. In the transfer phase, PISC protocol utilizes a human-perceivable video channel between two phones to transfer a public key. The use of a human-perceivable channel can avoid the danger of a Man-in-the-Middle attack [1], by which an adversary can inject bogus data in the middle of the two parties. Other common human-perceivable channels include the use of audio [3], vibration [4],

button press [5] and still image [6]. However, to increase the efficiency of key transfer, this paper is the first to propose the use of a video channel for key transfer. In this scenario, the user who wants to announce his public key creates the key pair with his smartphone first and transforms the public key to a short video which can be played on the screen. The created video is a carrier of the public key. Another user then uses the build-in camera to capture the short video from the display of the smartphone. Each frame of the video contains an image of a 2D barcode which presents part of the key being exchanged. The high resolution of the display and camera of modern smartphones makes the use of a video to transfer data possible and efficient. When the video is transferred, the user can decode the video to get the information of another user's public key and other relative information for further wireless connection. Once the first user's public key has been transferred to the second user successfully, the following transmission can use wireless channel directly.

After the transfer phase, the RISC protocol enters the verification phase. In the beginning of this phase, each user uses his phone to take a picture as the cryptographic nonce, which is unique and can be an undeniable representation of the key's owner. The verification utilizes the cryptographic nonce, the exchanged keys and hash functions. Through the use of the cryptographic nonce, the proposed protocol makes sure that the transferred key and its owner are bound together.

For key management, this paper also proposes a trust model to manage keys. In this scheme, every public key has a trust value which is a triplet including values of the degree of belief, disbelief and uncertainty of a key. A trust evaluation algorithm is presented to calculate the trust value recursively on the web of trust. Based on the trust value, a user can decide how to use the key.

For a user who has more than one device, how to synchronize and manage the key ring is another big challenge. This paper argues that the smartphone can serve as the personal information security center to coordinate security services, such as keychain synchronization, authentication and confidentiality services to other devices of the user. Therefore, this work proposes PISC security service coordination which uses the smartphone as the service center to coordinate security services among devices. Each device that provides security services can register the services to the coordinator. When a device needs a security service, it can query the coordinator about the service and then access to the service. By using a smartphone as the coordinator, a user can manage her security services efficiently.

In summary, this paper makes the following contributions: (1) we propose the PISC key management for public key exchange; (2) we use a trust model to manage keys; (3) we describe the PISC security service coordination mechanism.

2 PISC Key Management

This section describes how to securely exchange public keys between two users and how to manage the public-key ring through the use of trust. The proposed public-key exchange protocol is safe and easy to use. This work assumes that the users' smartphones are equipped with: (1) a build-in camera; (2) a display; (3) a software development kit (SDK) for capturing and playing videos; (4) wireless communication ability. For a modern smartphone, the above requirement is just standard [7].

2.1 PISC Public Key Exchange

This paper proposes the PISC public-key exchange protocol for users who need key exchanges. The goals of the protocol are: (1) providing a simple and convenient way to exchange public keys; (2) ensuring the public key is from its genuine owner; (3) avoiding the Man-in-the-Middle attack. The PISC public-key exchange protocol consists of two major phases: transfer phase and verification phase. The overall process is automatic and requires little user involvement. Table 1 lists the notation used in the key exchange protocol.

Figure 1 presents the outline of the PISC public-key exchange protocol in ten major steps. The protocol includes a transfer phase (step 1 to step 5) and a verification phase (step 6 to 9). Suppose user A is the initiator and user B is the responder for the exchange, the following describes how the protocol works.

Transfer Phase

In the beginning of the public key exchange, each user prepares the key pair of public and private keys (step 1). If a user already owns an old key pair, she can use it directly, or she must create a new pair. The PISC public-key exchange protocol utilizes a video, which is human-perceivable, to be a carrier to transfer the key. User A transforms her identity information, ID_a, and her public key, PU_a, to a video whose frames contain 2D-barcode images for the information (step 2). The use of video increases the capacity of the transferring

channel and it also eliminates the possibility of the Man-in-the-Middle attack.

Table 1. Notation for PISC key exchange.

Notation	Description
ID_a	User identity and network identity of user A
PU_a	Public key of user A
PR_a	Private key of user A
K_s	Session key
PIC_a	Cryptographic nonce (picture of user A)
$H(\ )$	Hash function
$\|\|$	Concatenation
$V(\ )$	Video transform of message
$E(PU_a, \cdot)$	Asymmetric encryption with user a's public key
$E(PR_a, \cdot)$	Asymmetric encryption with user a's private key
$E(K_s, \cdot)$	Symmetric encryption with session key

The transferred message ID_a includes not only the user identity but also the network identity which is used later for user B to build a wireless connection to user A. The network identity is platform dependent. For example, it contains the Bonjour service information on the iOS platform, but IP address and port number on other platforms.

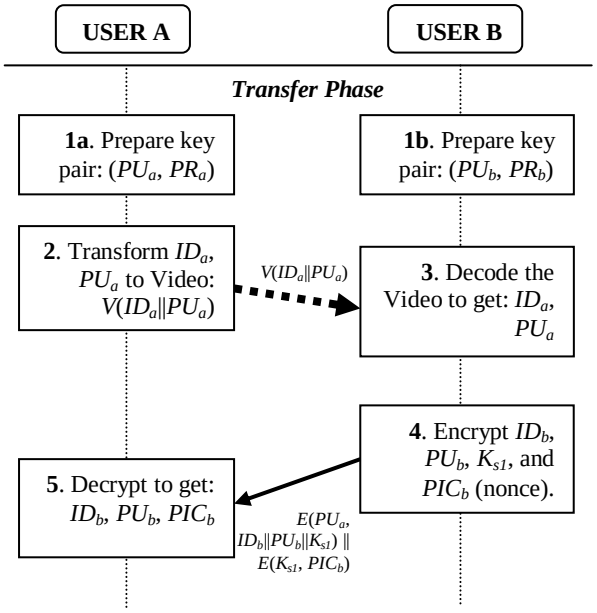

Figure 1a: The transfer phase of the PISC public-key exchange protocol

As the video is ready, user A notifies user B and plays the video $V(ID_a \,||\, PU_a)$ on the screen. User B then uses her smartphone to capture the video. After the capture finished, user B decodes $V(ID_a||PU_a)$ and she has the identity information ID_a and public key PU_a from user A now (step 3).

Since user B already has user A's public key, the following operations can utilize the public-key encryption and wireless connection for the good of efficiency. In step 4, user B encrypts her identity information, her public key and a session key with the public key of user A, PU_a. The resulting ciphertext is

$$E(PU_a, ID_b||PU_b||K_{s1}) \,||\, E(K_{s1}, PIC_b). \qquad \textbf{(M1)}$$

The last part of the message M1 is an encrypted picture of user B which is a cryptographic nonce for verification phase. In the proposed protocol, the picture is encrypted with symmetric encryption because of the consideration of efficiency. The session key K_{s1} is encrypted with the public key of user A and this ensures K_{s1} can be transferred to user A safely. This work assumes that both PIC_a and PIC_b are taken on the spot with their own cameras respectively. Each picture is unique and only the key owner has it. Therefore, PIC_b can be undeniable evidence that the public key PU_b belongs to the creator of PIC_b, i.e. user B.

After M1 is transferred, only user A can decrypt the above ciphertext with her private key PR_a to get ID_b, PU_b, and K_{s1} and she can use the session key K_{s1} to decrypt the nonce PIC_b (step 5). If user A can finish step 5 correctly, it also confirms that user B has received PU_a correctly.

After the execution from step 1 to step 5, user A and B have exchanged their public keys, and the protocol enters the verification phase.

Verification Phase

In the public key exchange, one of the most important issues is the key-to-owner binding. Both users must make sure the key is from its real owner. In step 6, user A generates a reply using the public key of user B:

$$E(PU_b, E(PR_a, K_{s2}) \,||\, H(PIC_b)) \;||\; E(K_{s2}, PIC_a) \qquad \textbf{(M2)}$$

The message M2 is then transferred to user B. Only user A has the picture PIC_a in her phone and the message M2 is encrypted by the key K_{s2}. In M2, K_{s2} is protected by the private key of user A, only users with the public key of user A can decrypt $E(PR_a, K_{s2})$ and get K_{s2}. Therefore, $E(PR_a, K_{s2})$ can be regarded as a digital signature of K_{s2} by user A, and $E(K_{s2}, PIC_a)$ binds PIC_a to the private key PR_a.

In other words, the message M2 gives evidence that PR_a and PU_a are combined to its real owner. When user B receives the above message, she can check the value of $H(PIC_b)$ and assure that user A has received PU_b and PIC_b correctly (step 7). Step 8 and 9 perform similar operations to assure user B has correctly received PU_a and PIC_a.

In the final step, both users manually compare the received pictures and the original pictures through displaying the pictures on their screens. If the comparing result is matched, the exchange of public keys is successful.

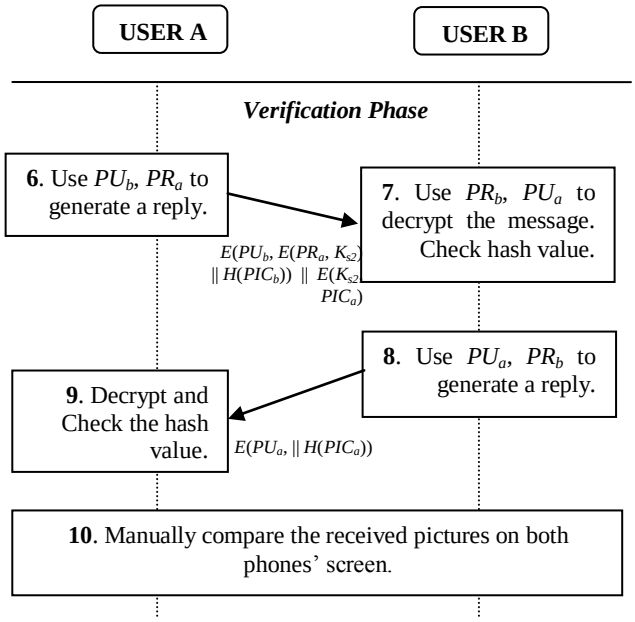

Figure 1b: The verification phase of the PISC public-key exchange protocol

User Involvement

Most operations of the PISC public-key exchange protocol are finished automatically by users' smartphones. The user involvement is not much. For two users who would like to exchange public keys, they first start the exchange-key application on their own smartphones. A window then pops up to direct users to take pictures as cryptographic nonce for further use. Each user needs to take a picture and then the video which carries the public key is played in the screen of the first user. The second user must catch the video by her camera. After that, the protocol proceeds automatically. At the final step, each user checks the received and original pictures to confirm the correctness of key exchange and that's all. What the user needs to do manually are: (1) starting the application on the phone, (2) taking a picture with the camera, (3) catching the video and (4) comparing the pictures. The proposed scenario makes the key exchange quick and simply. A public key transmitted by a face-to-face mechanism is more reliable than keys from unknown users.

2.2 Trust Management

A user has a key ring to store the public keys of other users. She must decide whether she can trust a public key or not. If a public key is obtained by the proposed public-key exchange protocol, the key can be trusted with high confidence. But not every public key in a user's key ring comes from a face-to-face meeting and in such situation how to measure the authenticity of the binding between a public key and its owner is a difficult problem.

The proposed PISC key management provides a solution through the use of trust which combines PGP's web of trust [8] and authentication algebra [9]. Unlike the trust model of Public Key Infrastructure (PKI), our trust management does not need a certificate authority (CA) and the trust decision is made by individual users.

Figure 2 depicts the algorithm for trust evaluation. In the algorithm, the input is a public key to be evaluated and the output is its trust value. The trust value is a triplet $w = (b, d, u)$, in which $b+d+u = 1$ and b, d and u correspond to the degree of belief, disbelief and uncertainty respectively [9]. The value of b, d and u is between 0 and 1. The authentication algebra is used to help the calculation of the trust value, in which three operators are defined: conjunction, recommendation and consensus.

TRUST EVALUATION (PU)
1. **For** the public key *PU* do:
2. **If** *PU* is from PISC key exchange
3. Set the trust value w_0 to (1, 0, 0).
4. **If** no trusted users sign *PU*
5. Set w_0 to (0, 1, 0).
6. **else**
7. **For** each key k_i (i=1, …, n) that signs *PU*
8. recursively compute k_i' s trust value w_i
9. by running *TRUST EVALUATION* (k_i)
10. Set w_0 to $w_1 \oplus w_2 \oplus \ldots \oplus w_n$.(consensus operation)
11. **Return** w_0.

Figure 2: Trust evaluation algorithm.

The consensus (b_{AB}, d_{AB}, u_{AB}) of two different trust values, (b_A, d_A, u_A) and (b_B, d_B, u_B) is defined as follows.

$(b_{AB}, d_{AB}, u_{AB}) = (b_A, d_A, u_A) \oplus (b_B, d_B, u_B)$

where

$$\begin{cases} b_{AB} = \dfrac{b_A u_B + b_B u_A}{u_A + u_B - u_A u_B} \\ d_{AB} = \dfrac{d_A u_B + d_B u_A}{u_A + u_B - u_A u_B} \\ u_{AB} = \dfrac{u_A u_B}{u_A + u_B - u_A u_B}. \end{cases}$$

The user can use the final trust value to make a sophisticated decision to judge the trust of the public key. The trust value is depicted as a point inside a trust value triangle, as shown in Figure 3. Usually, the trust has four discrete types as in PGP: (1) unknown, (2) not trusted, (3) marginally trusted and (4) completely trusted. The user can partition the trust triangle into different regions which represent different trust types respectively. The trust types and the rules for the triangle partition are adaptable and configurable depending on the need of the user.

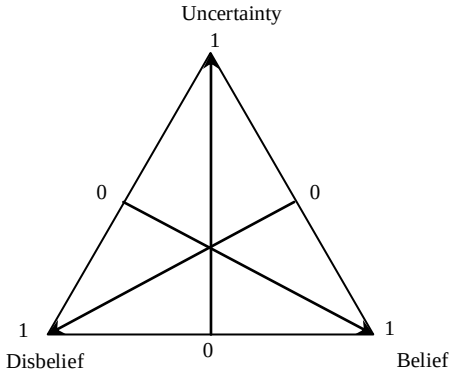

Figure 3: The Trust Triangle (redrawn [9]). Each trust value can be depicted as a point inside the Trust Triangle.

3 PISC Security Service Coordination

As consumer electronic products are prevailing, people nowadays may have more than one device in their daily life. For examples, people may have one personal computer, one tablet computer, one smartphone and other devices. With all the devices, the user needs an efficient mechanism to manage her information security. Therefore, this work presents the PISC Security Service Coordination for this purpose. The smartphone plays the role of the security center that coordinates other devices to provide and access necessary services.

The PISC security center keeps the synchronization of the key ring among devices. All the user's private keys are stored in the PISC phone. Each time a new public key is obtained, the trust value of the key is evaluated and the information is updated to all other devices. If a device is trusted by the user, private keys can be transferred to this device for convenience and the device is called a trusted agent.

For devices other than trusted agents, they are regarded as normal agents by the security center. In summary, in our framework, there are three types of roles for devices: (1) the security center which is a smartphone; (2) trusted agent; (3) normal agent. Table 2 shows security services that can be provided by different service roles.

Table 2. Services of agents.

	Security Center	**Trusted Agent**	**Normal Agent**
Keychain synchronization	●		
Digital signature	●	●	
Confidentiality	●	●	●

For the coordination to work, the security center provides a registration service that keeps a list of available services registered to the center. When a user wants to use security services, she must connect the device to the PISC security center first, and then the security center will send a reply to the user. If the requested service exists, the user can use the information in the reply to connect to devices which register the providing services.

When some device needs services, it must send a request for the services, the security center checks its service list and replies the request with service connection information. The service connection information contains the service name, service type, the coordination type, and the information of the network connection including the IP address, application port, etc.

With the connection information, the requesting user can connect to the agent to get the service, as shown in Figure 4. Figure 4 describes five steps for devices to connect to the PISC security center and request for services. In the first and second steps, agent A and agent B register their security services to PISC security center respectively. After the registration, the PISC security center maintains a list of available security services among all devices. Before a device becomes off-line, it must send a notification to withdraw the service registration. When an agent needs a service from other agents, the agent sends a service request to the center. If a service in the list matches the request, the security center sends a reply to the requester. The reply message contains necessary service information

and network information for accessing the service. The PISC security center itself can also provide security services, if the owner configures that. The implementation can follow the web service specification WS-Coordination.

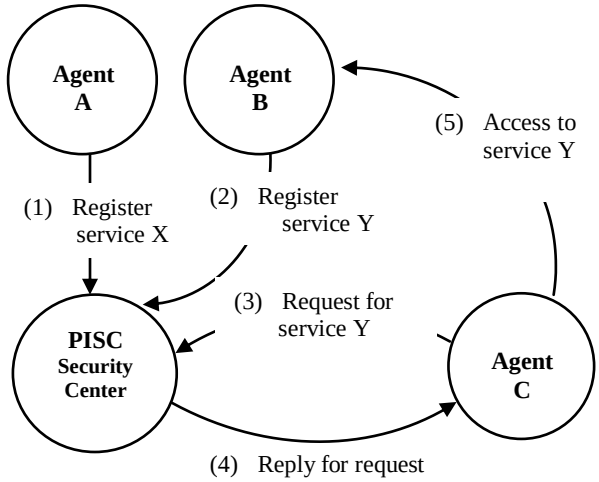

Figure 4: PISC security service coordination.

4 Conclusions

Encryption is the most important building block for modern cryptography. But before users can take advantage of symmetric or public-key encryption, they must exchange keys safely first. This paper proposes PISC key management for public key exchange and a trust model for key trust management. Through the use of a human-perceivable video channel and the undeniable cryptographic nonce created by the smartphone's camera, the PISC public-key exchange protocol provides a convenient method for users to exchange public keys safely. In the exchange protocol, most operations are automatic and user involvement is minimized. This paper also presents the trust management of keys by the methods of web of trust and the use of authentication algebra. Based on the key management, a PISC smartphone can acts as a security center to coordinate security services among the user's devices. The proposed coordination mechanism unifies the security services into one single security system, in which the smartphone plays the role as a personal information security center.

Acknowledgements

This research is supported in part by National Science Council of Taiwan under the Grants NSC 100-2410-H-150 -002. The authors also thank the reviewers for their comments and National Formosa University for the support.

References

[1] W. Stallings, *Cryptography and Network Security: Principles and Practices*, 3rd Edition, Prentice, 2003.

[2] S. Sheng, L. Broderick, and C.A. Koranda, *Why Johnny Why Johnny Still Can't Encrypt: Evaluating the Usability of Email Encryption Software*, in: Symposium On Usable Privacy and Security (Pittsburgh, PA, USA, 2006), 2006, 156-170.

[3] M. T. Goodrich, M. Sirivianos, J. Solis, G. Tsudik and E. Uzun, *Loud and Clear: Human-Verifiable Authentication Based on Audio*, in: 26th IEEE International Conference on Distributed Computing Systems (Lisboa, Portugal, 2006) , 2006.

[4] R. Mayrhofer and H. Gellersen, Shake Well Before Use: Intuitive and Secure Pairing of Mobile Devices, *IEEE Transactions on Mobile Computing*. 8 (2009), 792-806.

[5] C. Soriente , G. Tsudik and E. Uzun, Secure pairing of interface constrained devices, *International Journal of Security and Networks*. 4 (2009), 17-26.

[6] Y. H. Lin, A. Studer, Y. Chen, H. Hsiao, L. Kuo, J. Lee, J. M. McCune, K. Wang, M. Krohn, A. Perrig, B. Yang, H. Sun and P. Lin, SPATE: Small-Group PKI-Less Authenticated Trust Establishment, *IEEE Transactions on Mobile Computing*. 9 (2010), 1666-1681.

[7] H. Falaki, S. Kandula, D. Lymberopoulos, R. Govindan and D. Estrin, *Diversity in smartphone usage*, in: 8th International Conference on Mobile Systems, Applications, and Services (San Francisco, CA, USA, 2010), 2010, 310-322.

[8] W. Stallings, PGP Web of Trust, *Byte*. 20 (1995), 161-162.

[9] A. Josang, *An Algebra for Assessing Trust in Certification Chains*, in: the Network and Distributed Systems Security Symposium (San Diego, CA, USA, 1999), 1999, 1222-1231.

Yung-Tsung Hou received his PhD degree in information management from Sun Yat-Sen University in 2008. From 1998 to 2002, he worked at software companies. He is currently an assistant professor in the Department of Information Management at Formosa University. His research interests include information security, networks and Internet applications.