2011

Privacy-preserving PKI design based on group signature

Sokjoon Lee
ETRI, Daejeon, Korea

Hyeok Chan Kwon
ETRI, Daejeon, Korea

Dong-il Seo
ETRI, Daejeon, Korea

DOI: 10.4225/75/57b54498cd8c3
Originally published in the Proceedings of the 9th Australian Information Security Management Conference, Edith Cowan University, Perth Western Australia, 5th -7th December, 2011

PRIVACY-PRESERVING PKI DESIGN BASED ON GROUP SIGNATURE

Sokjoon Lee[1,2], Hyeok Chan Kwon[1], Dong-il Seo[1]
[1]Infrastructure Security Research Team, ETRI, Daejeon, Korea
[2]Department of Computer Science, KAIST, Daejeon, Korea
{junny, hckwon, bluesea}@etri.re.kr

Abstract

Nowadays, Internet becomes a part of our life. We can make use of numerous services with personal computer, lap-top, tablet, smart phone or smart TV. These devices with network make us enjoy ubiquitous computing life. Sometimes, on-line services request us authentication or identification for access control and authorization, and PKI technology is widely used because of its security. However the possibility of privacy invasion will increase, if we're identified with same certificate in many services and these identification data are accumulated.

For privacy-preserving authentication or anonymous authentication, there have been many researches such as group signatures, anonymous credentials, etc. Among these researches, group signatures are very practical because they provide unlinkability and traceability as well as anonymity.

In this paper, we propose a privacy-preserving PKI based on group signature, with which users' privacy can be kept in services. Because of traceability, their identities can be traced if they abuse anonymity such as cyber-crime. Moreover, we will also discuss open issues for further studies.

Keywords
Privacy, Anonymity, PKI, Group Signature

INTRODUCTION

Nowadays, Internet becomes a part of our life. We can make use of numerous services with personal computer, lap-top, tablet, smart phone or smart TV. These devices with network make us enjoy ubiquitous computing life. We can search or share information, send e-mail, purchase some products, and enjoy multimedia contents everywhere, such as in the living room, in the office or even on the street.

Sometimes, the on-line services request us authentication or identification for access control and authorization. ID/password, biometric authentication system, smart cards or other cryptographic methods are also used. Through these authentication methods, service providers can identify users to provide user-customized service.

However, the possibility of privacy invasion is being increased because we're easily identified in many services. For example, the services we used with same identity can be linked and tracked. In that case, our preference may be accumulated and monitored by government or some company. For solving this privacy problem, there have been some researches for privacy-preserving authentication which can provide anonymity for the service user. These researches include group signatures which are very practical because these signatures provide not only anonymity but also unlinkability and traceability. Group signatures are a kind of digital signature based on public key. One public key is given to a user group, not one user.

For the secure authentication and communication over the network, public key based mechanisms are widely recommended. In order to use this mechanism for a real service, PKI is often used. PKI is a infrastructure that binds a public key with the respective user identity using certificate issued and digitally signed by a trusted CA(Certificate Authority). In order to enhance privacy in PKI, some researches used pseudonym certificate or PMI certificate (Benjumea et al, 2004; Kwon et al, 2006). But these approaches have the limitation that they cannot satisfy full anonymity, because their authentication transactions are still linkable. As another approach, Benjumea et al (Benjumea et al, 2007) proposed how to construct privacy-friendly PKI by extending the semantic of X.509 certificate to apply anonymous signatures.

In this paper, we specify the requirements of privacy-preserving PKI for online services and describe how to design it. For this purpose, section 2 presents some related work such as PKI and group signatures. Section 3 proposes our requirements for privacy-preserving PKI and its design. In section 4, some open issues are introduced for further works. Finally, section 5 concludes this paper.

BACKGROUND

Anonymous Authentication

Anonymous authentication is the technology that a user can be authenticated while a certain level of anonymity is still satisfied. There are two types of anonymous authentication. The first one is anonymous credential a.k.a. pseudonym systems. This allows users to interact with multiple organizations anonymously. In this system, the pseudonyms cannot be linked; therefore nobody can tell if transactions of one user are from same user.

The first anonymous credential system was introduced by Chaum (Chaum, 1985). There have been many researches (Camenisch et al, 2001; Lysyanskaya et al, 1999) to enhance the first one. Some of them use very complex zero-knowledge interactive protocol to make it difficult to construct practical solutions. Direct anonymous attestation or DAA (Brickell et al, 2004) – a protocol for authenticating TPM (Trusted Computing Group, 2003) anonymously - is regarded as the first application of the anonymous credential system.

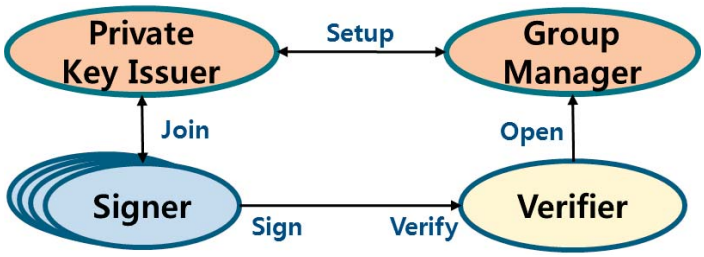
Anonymous signature (Boneh et al, 2004; Camenisch et al, 2002; Chaum et al, 1991; Delerablee et al, 2006; Hwang et al, 2011; Kiayias et al, 2004; Rivest et al, 2001) is another technology to provide anonymous authentication. This includes group signatures, ring signatures, traceable signatures, etc. Group signatures are very practical for many applications in the real world because it can provide not only anonymity but also linkability and traceability.

Group signature is first introduced by Chaum and van Heyst (Chaum et al, 1991). Any member of the group can sign messages. Verifiers can only know the correctness of the signature and if the signer is a member of the group. They cannot know any information of the signer. Moreover, they cannot know if two or more signatures are from same signer because the group signatures are unlinkable. Group manager is the most important entity in this scheme. Group manager can trace the signature, or reveal identity of the signer. There are variable group signature schemes, but most of them have the following procedures.

- Key Generation: the generation process of group public key and group secret key from some security parameters
- Join: a protocol between the group manager and a user, by which the user can become a group member to get his/her group member private key
- Sign: the algorithm by which any group member can compute group signature for given message using his/her private key
- Verify: the algorithm by which the validity of any group signature is verified with a group public key. In this algorithm, verifier cannot know signer's identity.
- Open: the algorithm by which group manager open signer's identity, given a signed message.

Short Group Signatures

Boneh et al (Boneh et al, 2004) proposed a short group signature scheme, in which they used a non-interactive zero-knowledge protocol for SDH(Strong Diffie-Hellman) problem in the bilinear pairing groups. To hide signer's identity in the signature, they proposed to use linear encryption based on decisional linear assumption. The total signature length is 1533 bits or 192 bytes for the same security with the standard 1024 bit RSA signature.



Figures 1: Entities and Procedures in Short Group Signature

As seen in Figure 1, there are four entities for this group signature as follows.

- Group Manager: entity who can open signer's identity. This entity opens the group public key $gpk = (g_1, g_2, h, v, h, w)$. The private key of the group manager is $gmsk = (\xi_1, \xi_2)$.

- Private Key Issuer: entity who issues user's private key pair. This issuer has the issuing key γ. This entity can be united with the group manager according to the authentication system policy.
- User: entity who joins the group to be a member of the group. User can sign messages anonymously with her private key pair $gsk[i] = (A_i, x_i)$, issued by private key issuer.
- Verifier: entity who verifies the signature and checks if the singer is a member of the group.

Like other group signatures, there are five procedures in this signature.

- Key Generation: the group manager and private key issuer select private keys $gmsk = (\xi_1, \xi_2)$, γ respectively and calculate the group public key $gpk = (g_1, g_2, h, v, h, w)$such that $u^{\xi_1} = v^{\xi_2} = h$ and $w = g_2^{\gamma}$, where g_1 and g_2 are generators of the bilinear groups G_1 and G_2.
- Join: the private key issuer or user randomly selects $x_i \in_R \mathbb{Z}_p^*$ and the issuer computes $A_i = g_1^{1/(\gamma + x_i)} \in \mathbb{G}_1$ to give $gsk[i] = (A_i, x_i)$to the user.
- Sign: user can generate the group signature for the message M as $\sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$ with her private key, where $T_1 = u^\alpha$, $T_2 = v^\beta$, and $T_3 = A_i h^{\alpha+\beta}$. Here, T_1, T_2 and T_3 are linear encryption result for blinding A_i and α and β are randomly selected from $\mathbb{Z}_p^*$. The user computes $R_1 \leftarrow u^{r_\alpha}$, $R_2 \leftarrow v^{r_\beta}$, $R_3 \leftarrow e(T_3, g_2)^{r_x} \cdot e(h, w)^{-r_\alpha - r_\beta} \cdot e(h, g_2)^{-r_{\delta_1} - r_{\delta_2}}$, $R_4 \leftarrow T_1^{r_x} \cdot u^{-r_{\delta_1}}$, $R_5 \leftarrow T_2^{r_x} \cdot v^{-r_{\delta_2}}$ with random blinding value r_α, r_β, r_x, r_{δ_1} and r_{δ_2}. Also, she computes the challenge $c = H(M, T_1, T_2, T_3, R_1, R_2, R_3, R_4, R_5)$ using a random oracle $H(\ldots$. s_α, s_β, s_x, s_{δ_1} and s_{δ_2} are the values for zero-knowledge proof of (A_i, x_i), which can be computed like $s_\alpha = r_\alpha + c\alpha$ and so on.
- Verify: given the message M and signature $\sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$, the verifier computes $\tilde{R}_1$, $\tilde{R}_2, \tilde{R}_3, \tilde{R}_4, \tilde{R}_5$ such as $\tilde{R}_1 \leftarrow u^{s_\alpha} \cdot T_1^{-c}$, $\tilde{R}_2 \leftarrow v^{s_\beta} \cdot T_2^{-c}$, $\tilde{R}_4 \leftarrow T_1^{s_x} \cdot u^{-s_{\delta_1}}$, $\tilde{R}_5 \leftarrow T_2^{s_x} \cdot v^{-s_{\delta_2}}$, $\tilde{R}_3 \leftarrow e(T_3, g_2)^{s_x} \cdot e(h, w)^{-s_\alpha - s_\beta} \cdot e(h, g_2)^{-s_{\delta_1} - s_{\delta_2}} \cdot (e(T_3, w) / e(g_1, g_2))^c$. The verifier checks if c is equal to $H(M, T_1, T_2, T_3, \tilde{R}_1, \tilde{R}_2, \tilde{R}_3, \tilde{R}_4, \tilde{R}_5)$. He accepts if this check succeeds.
- Open: given the message M and the signature $\sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$, the group manager checks the validity of the signature and opens the signer's A_i as $A_i = T_3 / (T_1^{\xi_1} \cdot T_2^{\xi_2})$ if the signature is verified correctly. A_i, a part of the signer's private key can be regarded as an ***anonymous identifier*** for her.

PKI with anonymity

X.509 (ITU-T X.509, 2000; IETF RFC 5280, 2008) based PKI technology is widely used to authenticate users in Internet. X.509 public key certificates bind the public key to the only entity who knows the associated secret key. This cannot provide anonymity since the identity of a user should be contained in the certificate.

There are several approaches to provide anonymity to X.509 based PKI. First approach (Kwon et al, 2006) is to use pseudonym on behalf of the real identity of users. The point is how pseudonym certificates of users are issued anonymously. Generally, this approach cannot provide unlinkability because the transactions of a specific user will be linked with the pseudonym in the certificate when the user uses same certificate. To avoid this problem, user can be issued multiple certificates with different pseudonyms. But multiple certificates would be inconvenient for users to manage them.

Second approach (Benjumea et al, 2007) is to extend the semantic of X.509 certificates to use anonymous signature schemes. V. Benjumea et al proposed the X.509 certificate infrastructure where the public key is not bound to a single entity but to a concept. The concept would be a single entity in the traditional environment, and also would be all group members in group signature. This approach is not applicable where there are group members who have different attributes and service authorization must be achieved according to the attributes.

PRIVACY-PRESERVING PKI DESIGN

Requirements for the Privacy-Preserving PKI

There are many requirements for design of privacy-preserving PKI. But we address three most important requirements here.

First, privacy-preserving PKI certificate should not contain any information to identify or link the signer. From the point of identification, the certificate naturally should not contain any information which identifies the user such as name, identification number, etc. Then, suppose that the certificate has only the pseudonym of the user. No information would be revealed from this certificate, but it is possible to link and track the user's transactions with the same pseudonym. This linkable feature is dangerous because of the following reasons. As linked transactions are accumulated more and more, anonymous users can be characterized more specifically by their transactions. In this case, the probability to lose their anonymity would increase. Users could be monitored for their shopping and spending patterns and this monitoring would be regarded as an invasion of privacy. Also, if the identifiable information of a user is disclosed from any one anonymous transaction, anonymity will be fully cancelled for all of the transactions. Therefore, one certificate should not be assigned to one user.

Second, it should be possible to trace the signer from the signature. If there's no trace function, it is possible for anonymous users to do illegal actions such as cyber-crime, online defamation, disguise, etc. Therefore there should be a countermeasure, which enables to cancel the anonymity of users performing illegal actions. This function should be performed by trusted party.

Third, there should be no change in the PKI standard specifications. These standard specifications include ITU-T X.509 certificate profile (ITU-T X.509, 2000), PKCS #8 private key syntax (RSA Laboratories, 1993), IETF CMP(Certificate Management Protocol) (IETF RFC 4210, 2005), etc. Fortunately, they provide extensibility to the public key algorithm identifier, key structure syntax, etc. This requirement is necessary in order to make the maximum use of the conventional PKI library without new or revised standards.

Not included in the three requirements above, there is another considerable point. If one entity by itself knows user's real identity from anonymous authentication transaction, the entity can always trace and track all users as a "Big Brother". That is another potential threat to privacy. Therefore, it is recommended to separate tracing function to prevent this problem. If separated, one entity should not be able to find the user's identity independently. If the trace for a specific user is necessary, all trusted authorities should cooperate according to a proper policy or regulation.

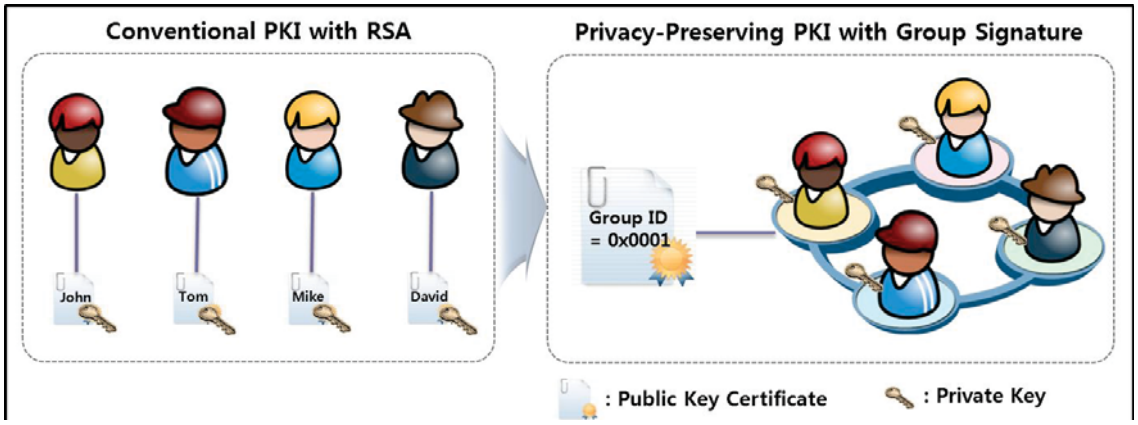
Privacy-Preserving PKI Design

Design Principle – Group Certificate

In this section, we propose a new privacy-preserving PKI design with group signatures. Group signature is a good candidate to satisfy the first and second requirements because it provides unlinkable and traceable anonymity. The main design issue is how to combine group signature with PKI. The combination result should be able to satisfy all requirements in the section 3.1.

In group signatures, user can be provided anonymity based on group. The user's signatures are verified with group public key. It means that verifiers can know not the user's identity but the group which the user belongs to. An identifier may be allocated to the group to represent it. We call it group identifier. Thus, group certificate is defined like this: an electronic document which uses a digital signature to bind together a group public key with an anonymous group identifier. This group certificate satisfies the first requirement in the section 3.1.

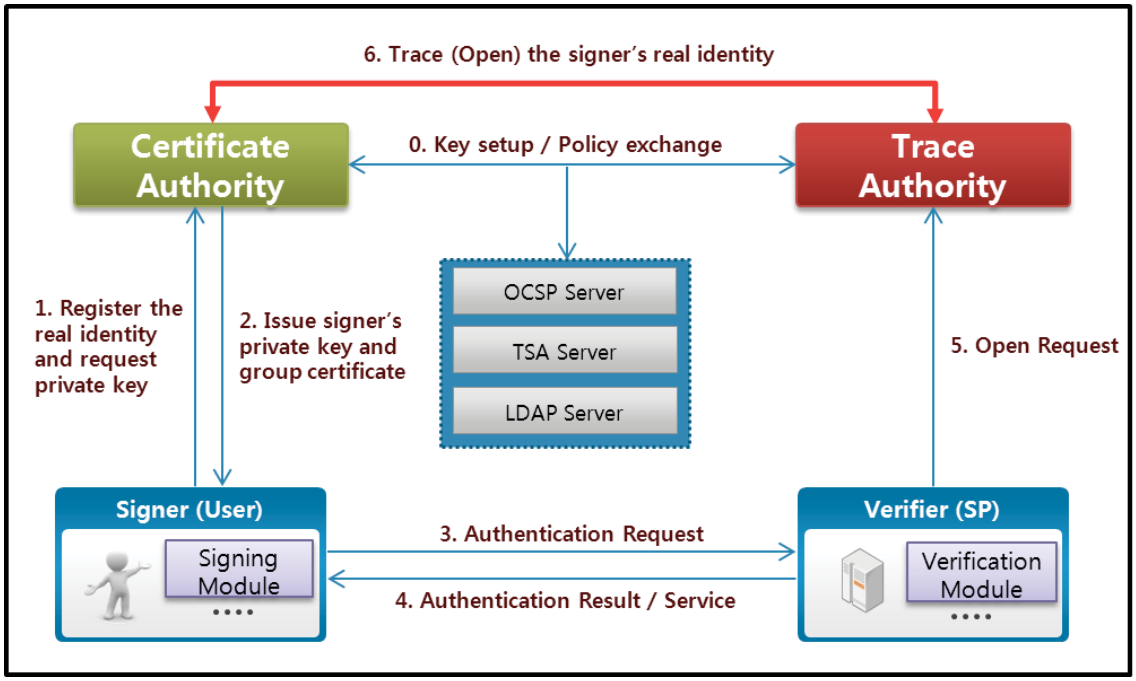
This design principle comes from Benjumea et al's approach (Benjumea et al, 2007) as we mentioned in the section 2.3. Figure 2 simply shows this concept.



Figures 2: Design Principle for Privacy-Preserving PKI

Privacy-Preserving PKI Architecture

In Figure 1, the entity that generates private key for group member is private key issuer. Therefore this entity can be regarded as CA(Certificate Authority) in the conventional PKI. Meanwhile, group manager has capability to open the anonymous user's identity. That means, it is another trusted entity which has trace authority. These two entities can be united as one organization or separated to two authorities. Considering 'Big Brother' problem, it is better not to unite these entities.



Figures 3: Privacy-Preserving PKI Architecture

Figure 3 shows our PKI architecture with the requirements and consideration. There are four main entities in this architecture. Other entities(OCSP, TSA and LDAP servers) perform the same function with the conventional PKI system.

- Certificate Authority: the first trusted authority that can issue group certificate and generate user's private key(step 2). This entity generates its issuing key in key setup and keeps the key secret, which is used to generate user's private key(step 0). User registers with her real identity to get the certificate and private key(step 1). Because anonymous identifier is the part of user's private key, this entity is able to know signer's real identity in case of given anonymous identifier.
- Trace Authority: the second trusted entity that has capability to generate group public key and opening key(step 0). This entity also can open the signer's anonymous identifier from a signature(step 5). If the real identity should be revealed, this entity would cooperate with Certificate Authority(step 6).

- Signer(User): entity who gets group certificate and her own private key by registering her real identity to Certificate Authority(step 1, 2). This entity can sign messages with the private key to be authenticated anonymously by the verifier or SP(step 3).
- Verifier(SP; Service Provider): entity who authenticates users anonymously and provides its service(step 4). If needed, this entity will request to open the anonymous user's identity(step 5).

Group Certificate Profiles

In this model, the users in the same group have one group certificate while they have different private keys. The certificate does not include any name or identifier of a specific user but only group identifier. Besides, group signature is different from the conventional public key algorithm. It means that public key and its algorithm identifier should be included in the certificate. Therefore, we need to reflect these two points on the X.509 certificate: the group identifier and group public key.

The following figure is the syntax of X.509 certificate profile.

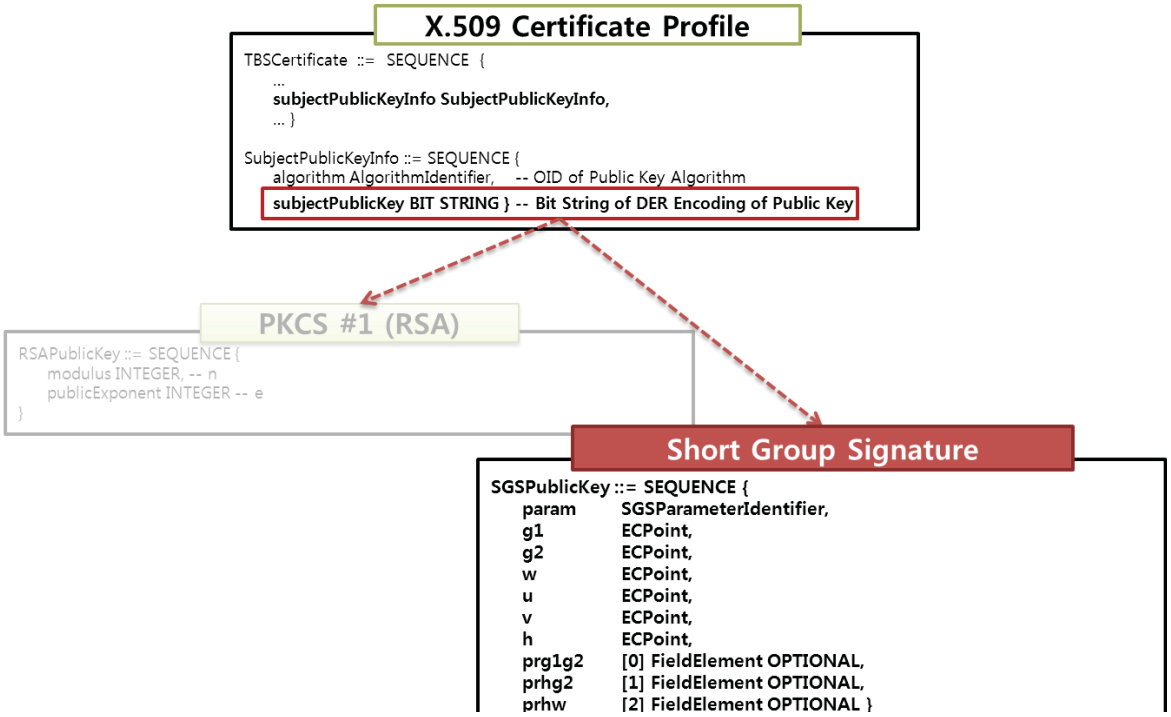
```
Certificate ::= SEQUENCE {
    tbsCertificate              TBSCertificate,
    signatureAlgorithm          AlgorithmIdentifier,
    signature                   BIT STRING  }

TBSCertificate ::= SEQUENCE {
    version         [0]         Version DEFAULT v1,
    serialNumber                CertificateSerialNumber,
    signature                   AlgorithmIdentifier,
    issuer                      Name,
    validity                    Validity,
    subject                     Name,
    subjectPublicKeyInfo        SubjectPublicKeyInfo,
    issuerUniqueID  [1]         IMPLICIT UniqueIdentifier OPTIONAL,
                                -- If present, version MUST be v2 or v3
    subjectUniqueID [2]         IMPLICIT UniqueIdentifier OPTIONAL,
                                -- If present, version MUST be v2 or v3
    extensions [3]              Extensions OPTIONAL
                                -- If present, version MUST be v3 --  }
```

Figures 4: X.509 certificate profile

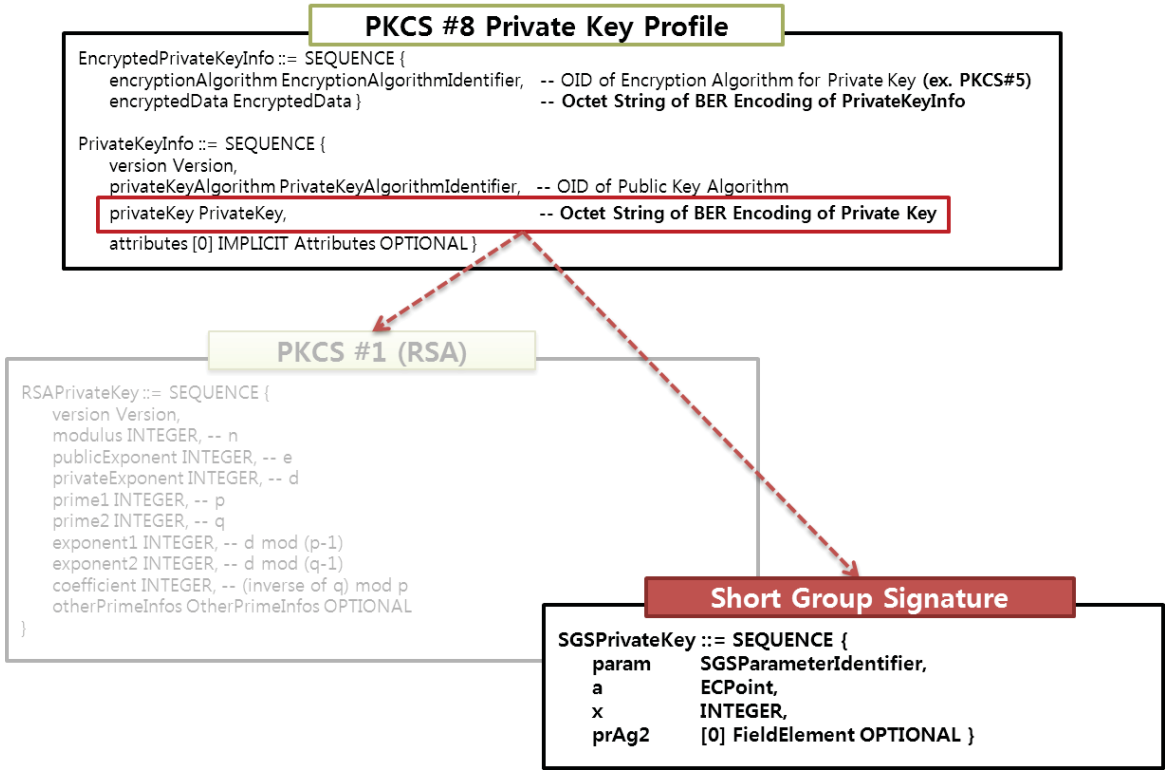
It is very easy to include group identifier instead of user's identifier. The subject field should contain this value. The value in this field is the sequence of relatively distinguished names. The relatively distinguished name is the set of attribute type and value. The type can be common name, serial number, country name, etc. Therefore one example of the subject value for group identifier can be "C=KR, O=ETRICA, serialNumber=000000003295234", but other representations will be possible.

Now, group public key should be included in the certificate. The field for this key is 'subjectPublicKeyInfo'. This field has two sub-fields: algorithm and subjectPublicKey. The value for 'algorithm' field can be the public key algorithm identifier. The value for 'subjectPublicKey' is bit string of DER encoding of public key. PKCS #1 defines RSA key profile. For short group signature, the group public key profile is defined in Figure 5.



Figures 5: Group public key profile for short group signature

In addition, we should consider users' private key. There's no definition in X.509 standard (ITU-T X.509, 2000), but PKCS #8 (RSA Laboratories, 1993) is generally used as a de-facto standard for the private key profile. Figure 6 shows the profile of PKCS #8. Similar to subjectPublicKey field in the certificate profile, privateKey field is the octet string of BER encoding of private key. PKCS #1 (RSA Laboratories, 2002) also defines RSA private key profile. Users' private key profile for short group signature is shown in Figure 6.



Figures 6: PKCS #8 and Short Group Signature Private Key profile

Certificate Management Protocol Profiles

Generally, CMP(Certificate Management Protocol) (IETF RFC 4210, 2005) and CRMF(Certificate Request Message Format) (IETF RFC 4211, 2005) are used for issuing, revoking, or managing the certificate. This protocol is easily extensible because it contains 'genm' and 'genp' for general message. These messages have 'infoType' and 'infoValue' sub-fields as shown in Figure 7. If the object identifier for 'infoType' is defined, the corresponding 'infoValue' can be defined at our convenience.

```
PKIBody ::= CHOICE {
    ...
    genm        [21]  GenMsgContent,          --General Message
    genp        [22]  GenRepContent,          --General Response
    ... }

InfoTypeAndValue ::= SEQUENCE {
                    infoType        OBJECT IDENTIFIER,
                    infoValue       ANY DEFINED BY infoType OPTIONAL, }
-- where {id-it} = {id-pkix 4} = {1 3 6 1 5 5 7 4}

GenMsgContent ::= SEQUENCE OF InfoTypeAndValue
GenRepContent ::= SEQUENCE OF InfoTypeAndValue
```

Figures 7: CMP Message profile

For example, it is necessary to define the trace message because there's no message for open or trace in the standard. Figure 8 shows the definition of trace message profile. This message will be exchanged between a verifier and TA(step 5 in Figure 3) and between TA and CA(step 6 in Figure 3).

```
GenMsg: {id-it numTrace}, traceReq
GenRep: {id-it numTrace}, traceRep

traceReq ::= SEQUENCE {
    signature    [0]          ShortGroupSignature,
    reasonText   [1]          UTF8String,
    AID          [2]          INTEGER          OPTIONAL }
traceRep ::= SEQUENCE {
    status                    TraceStatus,
    resultText                UTF8String       OPTIONAL}
TraceStatus ::= INTEGER {
    accepted     (0),
    rejection    (1),
    waiting      (2)    }
```

Figures 8: Trace Message Profile

OPEN ISSUES

Linkability Issues for the Service Application

As we mentioned in section 3.1, linkable authentication is not privacy-friendly. If an anonymous authentication has a property of linkability, then all linked transactions (from user) can be easily tracked. Meanwhile, unlinkability is not service-friendly. This is because many online services provide membership-based service. For example, they generally make regular customers, analyze their shopping pattern and give personalized service such as mileage service. This means that membership-based service providers need linkability at least in their own service domain. This is the issue about spatial linkability. There are three levels for this linkability.

- Complete space-unlinkability: verifier cannot always link two or more signatures.
- Local linkability: verifier can link two signatures presented to him, but a verifier or colluding verifiers cannot link two signatures presented to different verifiers.
- Complete space-linkability: verifier can always link two or more signatures.

Also, we can consider another type of linkability based on time. As time goes on, a user will get new private key instead of old one. Then, service provider will want to link the signature from new private key with the signature from old one for the consistent service. There are also three levels of linkability when time is considered.

- Complete time-unlinkability: verifier cannot always link two or more signatures even if they are from same key.
- Session linkability: verifier can link the signatures from same key
- Complete time-linkability: verifier can always link signatures even if the private key is updated

Service providers will prefer anonymous authentication techniques with (at least) local linkability and complete time-linkability. But simple pseudonym-based PKI system may be impractical and not privacy-friendly because it provides complete space-linkability and session linkability.

Meanwhile, our PKI system with short group signature can provide complete space- and time-unlinkability. This is most privacy-friendly for user but not service-friendly. But it may be possible to give better linkability. Hwang et al (Hwang et al, 2011) proposed another group signature, which has a new function of controllable linkability. This controllable linkability enables a special verifier who has a linking key to check if two signatures are from the same signer. By controlling the linking key, it may be possible to control the spatial-linkability and time-linkability. We plan to make our PKI system have more practical function with the enhanced researches including Hwang's scheme.

Revocation Issues

Revocability is another important requirement for authentication system. Invalid or cracked key should be revoked and not be verified correctly. The conventional PKI has two approaches for revocation. One is CRL(Certificate Revocation List) and the other is OCSP(Online Certificate Status Protocol). CRL is generated and published periodically. When a verifier needs to verify the signature with a specific certificate, he should check if the certificate is in the published CRL. If OCSP is applied, the verifier will request the certificate status by online protocol and OCSP server will respond with the result. In the PKI system, revocation is not complicated operation. Certificate validation check(verifier with CRL or OCSP server with OCSP protocol) only needs to compare serial number or public key of the certificate with the revocation list. This is because serial number or public key is one-to-one corresponded with the private key.

But anonymous authentication is in the different situation. For anonymity, there's no information on the certificate or signature to identify the private key. Therefore, some anonymous authentication systems including short group signature use key update method. When a key is revoked in this method, all private key and group public key should be updated. The revoked private key cannot be updated here. VLR(Verifier-Local Revocation) method also provides another possible way. In the VLR method, revocation list will be sent to verifiers and there's no need to update keys. When signature is given to verifiers, they should do cryptographic operation with the list. This operation will be proportional to the number of revoked keys.

Both key update and VLR method are not very efficient in the environment where key revocation events often occur. Therefore, we need to research more efficient and practical revocation method for this situation.

CONCLUSION

We presented the requirements and design for privacy-preserving PKI with group signature. In order to provide requirements, our PKI system uses group certificate for all group members while each member has her own private key. We designed the architecture and ASN.1 profiles of the group certificate, member private key and certificate management protocol to apply Boneh et al's group signature.

There are still open issues to apply this technique to the online membership-based privacy-preserving services. Exquisitely controlled linkability (local linkability and complete time-linkability) and efficient revocation method are required for online shopping, contents service in smart phone or smart TV, etc. We are trying to find the solution to solve these issues. With the future works, we will make more practical scheme leading us to more privacy-enhancing life.

ACKNOWLEDGMENTS

This work was supported by the ETRI R&D Program of KCC(Korea Communications Commission), Korea [11921-03001, "Development of Beyond Smart TV Technology"]

REFERENCES

Benjumea, V., Lopez, J., Montenegro, J. A., & Troya, J. M. (2004). A First Approach to Provide Anonymity in Attribute Certificates. *PKC 2004*, volume 2947 of LNCS, pp. 402-415

Benjumea, V., Choi, S. G., Lopez, J., & Yung, M. (2007). Anonymity 2.0 – X.509 Extensions Supporting Privacy-Friendly Authentication. *CANS 2007*, pp. 265-281

Boneh, D., Boyen, X., & Shacham, H. (2004). Short group signatures. *CRYPTO 2004*, pp. 41-55

Brickell, E., Camenisch, J. & Chen, L. (2004). Direct anonymous attestation. *In Proceedings of 11th ACM Conference on Computer and Communications Security*

Camenisch, J., & Lysyanskaya, A. (2001). An efficient system for non-transferable anonymous credentials with optional anonymity revocation. *EUROCRYPT 2001*, volume 2045 of LNCS, pp. 93-118

Camenisch, J., & Lysyanskaya, A. (2002). Dynamic accumulators and application to efficient revocation of anonymous credentials. *CRYPTO 2002*, pp. 61-76

Chaum, D. (1985). Security without identification transaction systems to make Big Brother obsolete. *Communications of the ACM*, Vol. 28, No. 10

Chaum, D., & Heyst, E. van (1991). Group signatures. *EUROCRYPT 1991*, volume 547 of LNSC, pp. 257-265

Delerablee, C., & Pointcheval, D. (2006). Dynamic fully anonymous short group signatures. *Vietcrypt 2006*, volume 4341 of LNCS, pp. 193-210

Hwang, J., Lee, S., Chung, B., Cho, H., & Nyang, D. (2011). Short Group Signatures with Controllable Linkability. *LightSec 2011*, pp 44-52

IETF RFC 4210. (2005). Internet X.509 Public Key Infrastructure Certificate Management Protocol (CMP)

IETF RFC 4211. (2005). Internet X.509 Public Key Infrastructure Certificate Request Message Format (CRMF)

IETF RFC 5280. (2008). Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile

ITU-T. (2000). Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks. *ITU-T Recommendation X.509*, Also available at ISO/IEC 9594-8

Kiayias, A., Tsiounis, Y., & Yung, M. (2004). Traceable signatures. *EUROCRYPT 2004*, volume 3027 of LNCS, pp. 571–589

Kwon, T., Cheon, J. H., Kim, Y., & Lee, J. (2006). Privacy Protection in PKIs: A Separation-of-Authority Approach. *WISA 2006*, volume 4298 of LNCS, pp.297-311

Lysyanskaya, A., Rivest, R. L., Sahai, A., & Wolf, S. (1999). Pseudonym systems. *SAC '99*, pp. 184-199

Rivest, R., Shamir, A., & Tauman, Y. (2001). How to leak a secret. *ASIACRYPT 2001*, volume 2248 of LNCS, pp. 552–565

RSA Laboratories. (1993). PKCS #8: Private-Key Information Syntax Standard. Version 1.2

RSA Laboratories. (2002). PKCS #1: RSA Cryptography Standard. Version 2.1

Trusted Computing Group. (2003). Trusted Platform Module (TPM) Specifications. Retrieved from http://www.trustedcomputinggroup.org/resources/tpm_main_specification