Int. J. Open Problems Comp. Math., Vol. 1, No. 3, December 2008

Efficient Proprietary Certification Process and Secured Theft-Protected Proprietary Certificates

R. Anitha[1] and R.S. Sankarasubramanian[2]
Department of Mathematics and Computer Applications
PSG College of Technology, Coimbatore – 641 004, Tamil Nadu, India
E-mail: anitha_nadarajan@mail.psgtech.ac.in, rss@mca.psgtech.ac.in

Abstract

This paper presents a new simple scheme for theft-protected proprietary certificate problem by introducing a mobile phone as an additional requirement for the proprietor and this scheme is analyzed to be more secured than the one presented by M. Jakobsson, et al. Secondly we present the proprietary certification process using Elliptic Curve Discrete Log Problem (ECDLP) instead of Discrete Log Problem (DLP) for defining the relation R and we conclude that by introducing ECDLP the certification process is more efficient than DLP.

Keywords: *Proprietary certificate, Collateral certificate, verifiable encryption*

1 Introduction

Digital certificates play a vital role in public key cryptography, and are commonly used in electronic commerce applications. The Certification Authority (CA) will issue the certificate to the client after successful completion of the secured protocol called "certification process". This certificate will be generally installed in the user's browser and will be required if the user want to access a particular website. If the accessing website is a paid one then traditional certificate, however, are not secure against certificate lending. This type of abuse is a concern in several types of applications, such as those related to digital rights management (DRM). Proprietary certificates were introduced [1] in an attempt to discourage sharing of access rights to subscription-based resources. A proprietary certificate is a certificate that contains some information related to another certificate called collateral certificate. This collateral certificate may contain the proprietors more sensitive information which he/she doesn't want to reveal to any one. If Alice (proprietor) reveals the proprietary secret key to Bob (not having access rights) to access subscription-based resources then the corresponding

collateral secret key of Alice will be automatically released to Bob which Alice doesn't like, therefore she discourages Bob by not giving her proprietary secret key.

While the original construction of proprietary certificates achieves its stated goal, but it overlooks the possible scenario in case of theft of the proprietary secret key that would lead to immediate loss of the collateral secret key also. Hence, theft of the proprietary key grants the intruder full access rights to all resources associated with both the proprietary and corresponding collateral keys. This approach punishes not only international sharing, but also accidental sharing. In [2] an additional property called theft protection was given as a solution. The core of their solution is based on the concept of time delay (T) for deriving the secret key so that the proprietor can do the necessary steps to change the secret key during this stipulated time T.

On the other hand it is also possible for Alice to give the access rights to Bob incidentally for time lesser than T as she knows that the collateral information will be revealed after the time T only. In this case the time lock concept may not be the proper solution and hence some other solution is required to solve this problem. In this paper we proposed a solution for the theft-protected proprietary certificate problem by introducing a mobile phone as an additional requirement for the proprietor and model the scheme with Verifiable Encryption as building blocks for our construction.

2 Building Blocks

In this chapter, the existing primitives and notations that have been used for this work are outlined. These concepts are used to build the certification process to obtain the proprietary certificate from the certification authority.

2.1 Verifiable Encryption

In some cryptographic applications it is crucial to be sure that player behave fairly, especially when they use public key encryption. For example, a voting scheme where each player encrypts the name of his favorite candidate can be considered. It can be useful to convince anybody that the encrypted name is indeed in the list of the candidates with out revealing any information about this name. Accordingly, mechanisms are needed to check the correctness of encrypted data, without compromising secrecy and such an encryption process is called fair encryption or verifiable encryption. A verifiable encryption is a two party protocol between a prover P and a verifier V who initially have access to some public key pk_1, some public value p and some binary relation R. At the end of the protocol, the verifier obtains a cipher text under pk_1 of some value x and accepts the cipher text if the relation R between x and p holds and rejects otherwise. In other words

a verifiable encryption is scheme having public key encryption process and also having the proving properties about encrypted data.

2.3 Sigma Protocol

In cryptography, a proof of knowledge is an interactive proof in which the prover succeeds 'convincing' a verifier that it knows something. For a computer system ‘something’ is defined in terms of computation. A machine 'knows something', if this something can be computed (in polynomial time), for a given set of input to the machine. As the program of the prover does not necessarily spit out the knowledge itself (as is the case for zero-knowledge proofs [15]) a machine with a different program, called the knowledge extractor is introduced to capture this idea. A Σ-protocol [13],[14] for a binary relation $R \subseteq \{0,1\}^* \times \{0,1\}^*$ is a three move ‘honest verifier’, ‘zero-knowledge’, proof of knowledge for R. That is a string x is common input to prover P and the verifier V, and demonstrates knowledge of a w such that $(x, w) \in R$. w a witness for x.

To illustrate this protocol let us consider the following example. Let p be a prime, q a prime divisor of $p-1$, and g an element or order q in Z_p^*. Suppose a prove P has chosen $k \in Z_q$ at random and has published $h = g^k \bmod p$. A verifier V who gets $\{p, q, g, h\}$ can check that p, q are prime and that g, h have order q. Since there is only one subgroup of order q in Z_p^*, this automatically means that $h \in < g >$, that is there exist k such that $h = g^k \bmod p$.

The following protocol suggested by Schnorr gives a very efficient way to convince V about this:

1. P chooses $r \in Z_q$ at random and send $a = g^r \bmod p$ to V as a *commitment.*
2. V chooses a *challenge* c at random in Z_{2^t} and sends it to P. Here, t is fixed such that $2^t < q$
3. P sends $z = r + c.k \bmod q$ to V as a *response.*

Finally V checks that $g^z = a.h^c \bmod p$, that p, q are prime and that g, h have order q and accepts if and only if this is true else reject it.

2.4 Verifiable encryption of discrete logarithms

In [9],[10] “Cut and choose” method was proposed. In this scheme Common inputs for A and B are $\gamma, \delta \in G$ and A has an additional input $x = \log_\gamma \delta$. A split

$x = x_0 + x_1 + ... + x_{t-1}$ and generate the random numbers $r_i (i = 0,1,...,t-1)$ and engaged in the following protocol for t times.

1. $A \to B$: $\psi_i = E(r_i, x_i, L), \delta_i = \gamma^{x_i} (i = 0,1,...,t-1)$
2. $B \to A$: $c \in \{0,1,...t-1\}$
3. $A \to B$: r_c, x_c
4. B check i) $\psi_c = E(r_c, x_c, L)$ ii) $\delta_c = \gamma^{x_c}$ and iii) $\delta = \delta_0.\delta_1...\delta_{t-1}$.

In [11],[8] they use "Double discrete log cut and choose" method was proposed. In [12] a new encryption scheme with protocols for verifiable encryption and decryption of discrete logarithms and representations. This scheme is secured against chosen cipher text attack, it is separable, efficient and has no random oracle. Their proofs of multiplicative relations among committed integers are based on the strong RSA assumption. Encryption is based on the Decisional Composite Residuosity (DCR) assumption. Their scheme is as follows:

Encryption Scheme

p, q, p', q' are distinct odd primes with $p = 2p' + 1$ and $q = 2q^1 + 1$ where p' & q' are both of l bits long. $n = pq$, $n' = p'q'$, $\xi = (1 + n \bmod n^2) \in Z_{n^2}^*$. We know that $Z_{n^2}^* \cong Z_2 \times Z_2 \times Z_n \times Z_{n'}$ and $\xi^x = (1 + xn \bmod n^2)$. Also the DCR assumption is that it is hard to distinguish $Z_{n^2}^*$ from $\left(Z_{n^2}^*\right)^n$. H is a collision resistant, $l - bit$ hash.

Key Generation

Let $x_1, x_2, x_3 \in [0, n^2/4)$, $g' \in Z_{n^2}^*$, $g = (g')^{2n}$, $y_1 = g^{x_1}$, $y_2 = g^{x_2}$ and $y_3 = g^{x_3}$; $PK = (n, H; g, y_1, y_2, y_3)$, $SK = (x_1, x_2, x_3)$.

Encryption

Let $m \in [-n/2, n/2)$, label $L \in \{0,1\}^*$ choose $r \in [0, n/4)$ compute $u = g^r$, $e = y_1^r \xi^m$, $\hat{y} = y_2 y_3^{H(u,e,L)}$, $v = \hat{y}^r$. The cipher text $\psi = (u, e, v)$.

Decryption

If $u^{2(x_2 + H(u,e,L)x_3)} = v^2$ and $((e/u^{x_1})^2)^{2^{-1}} \bmod n = \xi^m$ for $m \in [-n/2, n/2)$ output m otherwise, reject m.

This scheme is secure against adaptive chosen ciphertext attack (but "gently malleable"), assuming DCR and H is collision resistant.

The Protocol

Common inputs for A and B : $PK = (n, H; g, y_1, y_2, y_3)$, $\tilde{g}, \tilde{h} \in Z_n^*$ of order n', $\gamma, \delta \in G$, $\psi = (u, e, v)$ and a label L; let $\hat{y} = y_2 y_3^{H(u,e,L)}$.

A has an additional inputs $m = \log_\gamma \delta \in [0, n/2)$ and $r \in [0, n/4)$ such that $u = g^r$, $e = y_1{}^r \xi^m$, $v = \hat{y}^r$.

1. A Chooses $s \in [0, n/4)$, computes $\tilde{v} = \tilde{g}^m \tilde{h}^s$, sends $\tilde{v}$ to B,
2. Then A and B engage in the following protocol.

2.1 $A \to B$ A generates random s, r', m', s' and computes $\tilde{v} = \tilde{g}^m \tilde{h}^s$, $u' = g^{2r'}$, $e' = y_1{}^{2r'} \xi^{2m'}$, $v' = \hat{y}^{2r'}$, $\delta' = \gamma^{m'}, \tilde{v}' = \tilde{g}^{m'} \tilde{h}^{s'}$ and send $(u', e', v', \delta', \tilde{v}')$ to B

2.2 $B \to A$ B chooses a random challenge $c \in \{0,1\}^*$ and sends to A

2.3 $A \to B$ A replies with

$\bar{r} = r' - cr, \bar{m} = m' - cm, \bar{s} = s' - cs$ (Computed in Z)

2.4 B checks whether the relations $u' = u^{2c} g^{2\bar{r}}, e' = e^{2c} y_1{}^{2\bar{r}} \xi^{2\bar{m}}, v' = v^{2c} \hat{y}^{2\bar{r}}, \delta' = \delta^c \gamma^{\bar{m}}$, $v' = \tilde{v}^c \tilde{g}^{\bar{m}} \tilde{h}^{\bar{s}}$, $\bar{m} \in (-n/4, n/4)$

The above protocol is a special honest-verifier zero knowledge protocol for proving that a cipher text encrypts a discrete logarithm whose soundness follows to form the strong RSA assumption.

3 Verifiable Encryption Using Elliptic Curve Relations

In the proprietary certification process, the user (Prover) has to send the fair encryption of the collateral secret key sk_2 to the certification authority CA_1 (verifier). For this fair encryption process we have used the Elliptic Curve Cryptosystem for defining the relation R. Elliptic Curve Discrete Logarithm uses only lesser key size while compared to any other hard Problems. In [7] it has been proved the following results Let p be a prime of the form $A^2 + 2B^2; B \equiv 1 \bmod 2$. Choose the sign of A such that $(-1)^{\frac{A-1}{2}} = \left(\frac{p}{3}\right)(-1)^{\frac{p-3}{8}}$. If $\gcd(A-1, B) = 1$, then the group of F_p rational points of the reduction E_p of E at p is a cyclic group of order $p+1-2A$, thus $2A$ is the trace of the Frobenius endomorphism of E_p.Since we can find a cyclic elliptic curve group or order ρ, we tried to define the relation R based on Elliptic Curve Discrete Log Problem (ECDLP). Let E: $y^2 = x^3 - 120x - 448$ is an Elliptic Curve cyclic group of order ρ, generated by B. We define $R = \{(w, A) \in W \times E : wB = A\}$, is the binary relation, where $W = [\rho]$. There are several advantages in using elliptic curves for cryptography. Elliptic curve cryptosystems with smaller key sizes appear to be just as secure as ``classical'' cryptosystems with much larger key sizes, so elliptic curve cryptosystems can be more efficient. Another advantage, is that elliptic curve cryptosystems appear to be vastly more secure over ``large finite fields of characteristic 2 " than RSA, which is very important in practical applications.

Also, elliptic curves are simply way cooler, so they used to attract venture capitalists. Some mobile phones also use elliptic curve cryptography.

3.1 Notations and variable expressions

$[a]$ The set $\{0,1,...,a-1\}$,
$H_{hk}(.)$ keyed hash function that uses a key hk,
p^1, q^1 primes, $p = 2p^1 + 1$, $q = 2q^1 + 1$, $n = pq$, $n^1 = p^1 q^1$,
$x_1, x_2, x_3 \in [n^2/4]$ are secret keys,
$g^1 \in Z^*_{n^2}$, $g = (g^1)^{2n}$, $y_1 = g^{x_1}$, $y_2 = g^{x_2}$, $y_3 = g^{x_3}$,
$h = (1 + n \bmod n^2) \in Z^*_{n^2}$, $abs : Z^*_{n^2} \to Z^*_{n^2}$ defined as
$sk_2 \in [n]$ collateral secret key, $L \in \{0,1\}^*$ is the Label,
$(y_1, y_2, y_3, n, g, hk)$ are public keys,
(u, e, v), cipher text,
(N, G, H) augmented public key where $N = PQ$ the product of two safe primes P, Q, such that $P = 2P^1 + 1$, $Q = 2Q^1 + 1$, where $N^1 = P^1 Q^1$,
$Y_{N^1} \subset Z^*_{N^2}$ is a subgroup of Z^*_N,
$E: y^2 = x^3 - 120x - 448$ is an Elliptic Curve cyclic group of order ρ, generated by B, $G, H \in Y_{N^1}$ are the two generators of Y^1_N, $W = [\rho]$
$R = \{(w, A) \in W \times E : wB = A\}$, is the binary relation,
We require that $\rho < n$

3.2 Encryption Scheme

To encrypt the collateral secret key $sk_2 \in [n]$ with label $L \in \{0,1\}^*$, the prover choose a random $r \in [n/4]$ and computes $u = g^r$, $e = y_1^r h^{sk_2}$, $v = abs((y_2 y_3^{H_{hk}(u,e,L)})^r)$. The cipher text is (u, e, v).

To decrypt a cipher text (u, e, v) with label L, the verifier first checks that $abs(v) = v$ and $u^{2(x_2 + H_{hk}(u,e,L)x_3)} = v^2$. If this does not hold then outputs "reject" and halt. Then the verifier calculates $t = 2^{-1} \bmod n$ and computes $\tilde{m} = (e/u^{x_1})^{2t}$. If $\tilde{m} = h^{sk_2}$ for some $sk_2 \in [n]$ then output sk_2; otherwise, output "reject".

3.3 The Protocol

The common input of the prover and verifier is the public key $(y_1, y_2, y_3, n, g, hk)$, the augmented public key (N, G, H), a group element $A \in E$, a cipher text (u, e, v) and

a Label L. The prover has additional inputs $sk_2 B = A$ and $r \in [n/4]$ such that $u = g^r$, $e = y_1^r h^{sk_2}$, $v = abs((y_2 y_3^{H_{hk}(u,e,L)})^r)$.

The prover chooses a random $s \in [n/4]$, $r^1 \in [-n, n]$, $s^1 \in [-N, N]$, $m^1 \in [-\rho, \rho]$ and computes $T = G^m H^s$, $u^1 = g^{2r^1}$, $e^1 = y_1^{2r^1} h^{2m^1}$, $A^1 = m^1 B$ and $T^1 = G^{m^1} H^{s^1}$. The prover sends $(T, u^1, e^1, v^1, A^1, T^1)$ to the verifier.

The verifier chooses a random challenge c and sends to the prover.

The prover computes $\tilde{r} = r^1 - cr$, $\tilde{s} = s^1 - cs$ $\tilde{m} = m^1 - csk_2$ and sends $(\tilde{r}, \tilde{s}, \tilde{m})$ to the verifier.

The verifier checks whether the relations $u^1 = u^{2c} g^{2\tilde{r}}$, $e^1 = e^{2c} y_1^{2\tilde{r}} h^{2\tilde{m}}$, $v^1 = v^{2c} (y_2 y_3^{H_{hk}(u,e,L)})^{2\tilde{r}}$, $A^1 = \tilde{m}B + cA$, $T^1 = T^c G^{\tilde{m}} H^{\tilde{s}}$ and $-n/4 < \tilde{m} < n/4$ holds.

If any of them does not hold, the verifier stops and output 0.

If $v = abs(v)$ then verifier outputs 1; otherwise outputs 0.

3.4 Numerical Toy Example

Let $p = 7, q = 11$ so that $n = 77, n^1 = 15, [n^2/4] = [1482]$. Let $x_1 = 1$, $x_2 = 2$, $x_3 = 3$, $g^1 = 2$ so that $g = 5714$, $y_1 = 5714$, $y_2 = 4722$, $y_3 = 4558$, $h = 78$. Let $sk_2 = 4$, such that $4B = A$, $r = 1, L = 1, H_{hk}^{(u,e,L)} = 1$ so that $u = 5714$, $e = 4713$, $v = 606$. The cipher text $(u, e, v) = (5714, 4713, 606)$.

In the decryption process the verifier checks whether $abs(v) = 606 = v$, $u^{2(x_2 + H_{hk}(u,e,L)x_3)} = 5567 = v^2$. Then calculates $t = 39$, and checks $\tilde{m} = 309 = h^{sk_2}$ and outputs the collateral secret key $sk_2 = 4$. Let E be an elliptic curve set of order $\rho = 7$ generated by the point $B \in E$ then $W = [7]$. Let $P = 5, Q = 7$ so that $N = 35, N^1 = 6$ therefore $Z_{35}^* = \{1, 2, 3, 4, 6, 8, 9, 11, 12, 13, 16, 17, 18, 19, 22, 24, 26, 27, 29, 31, 32, 33, 34\}$. Let $Y_6 \subset Z_{35}^*$ be the subgroup and $Y_6 = \{1, 11, 16, 19, 24, 34\}$, let $G = 19, H = 24$ be the two generators of Y_6. The prover chooses $s = 1$, $r^1 = 1$, $s^1 = 1$, $m^1 = 3$ and calculates $T = 34, u^1 = 4722, e^1 = 4414, v^1 = 5567, A^1 = 3B, T^1 = 11$ and send to the verifier. Let the verifier send $c = 1$ to the prover. The prover computes $\tilde{r} = 0, \tilde{s} = 0, \tilde{m} = -1$ and sends to the verifier. The verifier checks whether the relations $u^1 = 4722 = u^{2c} g^{2\tilde{r}}$,

$e^1 = 4414 = e^{2c} y_1^{2\tilde{r}} h^{2\tilde{m}}$, $v^1 = 5567 = v^{2c}(y_2 y_3^{H_{hk}(u,e,L)})^{2\tilde{r}}$, $A^1 = 3B$, $T^1 = 11 = T^c G^{\tilde{m}} H^{\tilde{s}}$ and $-n/4 < -1 < n/4$ holds

4 Comparison between ECDLP and DLP

Given a pair of points (P, mP) find the integer m, is the problem with a very different nature from that of point multiplication. This problem is called Elliptic-Curve Discrete Logarithm Problem. It is widely believed that the ECDLP is difficult to solve when the point P has a large prime order. For a curve defined over F_q F_q it is very easy to devise a large prime r of size slightly less than q such that $E(F_q)$ contains a subgroup of order r. Thus, the best known algorithm for solving the ECDLP has a time complexity expression $O(\sqrt{r}) \approx O(\sqrt{q})$. This is more or less a result of a brute force search method helped with the birthday paradox which is the bases for several famous algorithm devised by Pollard for computing discrete logarithm. Such a result applied to discrete logarithm problems in any abelian group of order roughly q. Therefore, we can say that a solution with complexity $O(\sqrt{q})$ for the ECDLP is not a solution at all due to its irrelevance to the group structure in question. In the case of Discrete Logarithm Problem (DLP) in a finite field, there exist algorithms called index calculus for solving the problem. The time complexity of an index calculus method for discrete logarithm in a finite field F_q has a sub-exponential expression $\exp\left(\frac{\log q}{2}\right)$. In specific, if we use an elliptic curve over a finite field F_q with $q \approx 2^{160}$, the difficulty of the ECDLP will be expressed by a 2^{80} value. To obtain a similar difficulty for the DLP in a finite field, the sub-exponential expression will reduce to $q \approx 2^{1000}$.Hence if the use the ECDLP relation the key size may be reduced to 6.25 time that of the key size used for DLP.

5 Proprietary Certification Process

Let CA_1, CA_2 be the distinct certification authorities issuing the certificates for the proprietary and collateral services respectively. Let C_1, C_2 be the proprietary and collateral certificates of some user respectively which are publicly available. Let pk_1, sk_1 are the public and secret keys of the user to obtain the proprietary certificate C_1 from CA_1. Let pk_2, sk_2 are the public and secret keys of the user to obtain the collateral certificate C_2 from CA_2. A certification authority CA_1 wishes to issue a proprietary certificate C_1 to a certain user. The user has to provide a second certificate C_2, issued by CA_2 as collateral. The user sends the secret key

sk_2 of the second certificate C_2 encrypted with the public key pk_1 to CA_1. Since sk_2 is encrypted with pk_1 to decrypt it requires sk_1 which is the secret key of the user, no information about sk_2 will be revealed to CA_1. At the same time it is necessary for the user to convince the CA_1 about the correctness of the encrypted data.

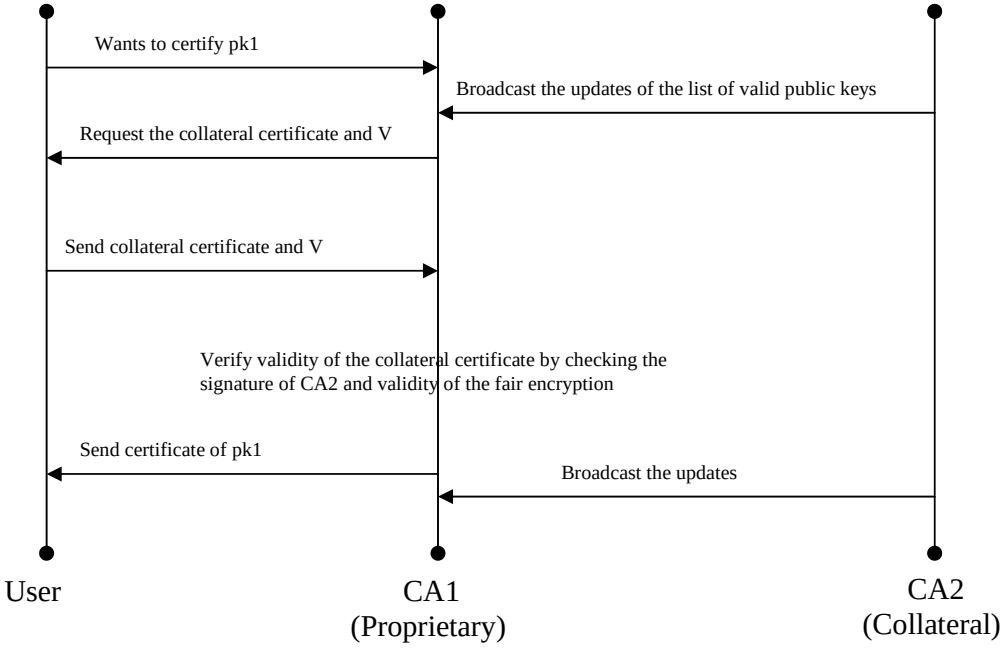

Fig1: Overview of issuing Proprietary Certificates

The aim is to construct a proprietary certificate system that respects these requirements for heterogeneity and flexibility in public key infrastructure. Assume that certificate authorities publish directories containing public information on the certificates they have issued. In [1] the desirable properties of the system are given, they are Non-transferability, Cryptosystem agility, Locality, Efficiency and Security. The paper [1] showed that how to extend the regular certificate to make it proprietary one, namely being linked to the collateral certificate. As it was suggested in [1] verifiable encryption can be used for the implementation of proprietary certificates. Let us assume that the user obtained the certificate for his collateral public key pk_2 from CA_2. In order to certify the public key pk_1, the certification authority CA_1 asks the user to present the certificate of another key pk_2 issued by CA_2 which he uses for some other services (to be considered as collateral) and the value $V = Ver_{pk_1}(sk_2)$ which is the verifiable encryption of his collateral secret key sk_2 under pk_1.

If CA_1 agrees to use this certificate as collateral then verifies validity of the collateral certificate by checking the signature of CA_2 and validity of the verifiable encryption. The properties of verifiable encryption ensure that CA_1 does not learn any information about the user's collateral secret key while being able to verify whether this cipher text is valid. CA_1 also need to be sure that pk_2 is still a valid key. It is assumed that CA_2 broadcasts the updates to the list of valid public keys. Thus CA_1 needs to check that pk_2 is still on that list. No direct interaction between CA_1 and CA_2 is required. If the verification is successful, then CA_1 includes V and the encryption of pk_2 under pk_1 in the certificate in addition to standard information such as the user's identity information and pk_1. If the user shares sk_1 with another party, then that party can decrypt V and obtain sk_2. The weakness of the above approach is the accidental exposure of a proprietary secret key due to theft or intrusion which would immediately lead to the loss of the collateral key. Therefore, direct use of proprietary certificates would be risky since it imposes additional insecurity on the collateral secret key.

6 Making Proprietary Certificate Theft Protected

In our model, Min Wu's Authentication Protocol [6] has been modified to get the solution for the accidental theft of proprietary secret key. We assume that the security proxy (P) in our model is capable of generating a secured number generated once (nonce) "challenge" and verify the dynamically generated secret random "challenge" concatenated with the originally stored secret key sk_2 for validating the authentication process. We illustrate our model using the Figure 2. The process has seven steps: (1) The user (U) directs Internet Kiosk's browser (K) to contact the security proxy server (P). (2) U produces the certificate C_1 into K, which sends it to P. (3) P randomly chooses a "challenge" from a dictionary and sent to user's mobile (M) as an SMS message. (4) U got the challenge form her M (5) The user directs K's browser to contact P (6) Now U types the secret key sk_2 concatenated with the challenge into K, which sends it to P. (7) Once authenticated, P operates like a traditional web proxy.

In this model, if the user accidentally lost her proprietary secret key and if Charlie (not having access rights) got that secret key then after producing the public certificate C_1 into K the system will wait for the new secret key = (sk_2 ||challenge) to enter and at the same time the corresponding challenge will be sent to Alice (original user) mobile phone. From the unusual SMS message form the Proxy server Alice might conclude that her secret key has been accidentally lost and some one is trying to impersonate her identity and she may take necessary steps to change that secret key, moreover Charlie doesn't gain any access form the

security proxy as he couldn't know the "challenge". If Alice incidentally wants to give her secret key and mobile phone to Bob, as form the previous case Bob can get the corresponding collateral secret key of Alice which Alice doesn't like, therefore she discourages Bob by not giving her proprietary secret key and the mobile phone. This scheme has been given in Figure 2. We claim that this system is secured in the sense that we make use of Verifiable Encryption of a Discrete Logarithm using Elliptic Curve for our construction of the binary relation R

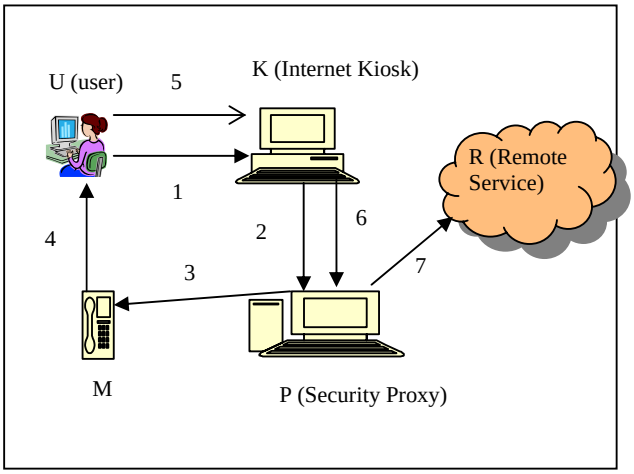

Fig. 2 A new approach to avoid penalizing accidental sharing

7 Analysis

We had analyzed the two most likely security threats: (1) K remembers connection information for replay attack at a later time; (2) P receives two simultaneous connections from different kiosks, each claiming to be the same user. We address this through the use of a unique session name (SN) for each user's session, and a nonce (N) that is transmitted to M with the SMS message. N prevents forged replies from an attacker who knows SN but does not have possession of M. Security of the system depends on the security of messages sent by SMS. The security of this system also depends upon the fact that U is in possession of M. We believe that this is a reasonable assumption: when people lose their mobile phones, they are typically reported lost and deactivated. Once deactivated, M will no longer be able to receive SMS messages destined for U. We claim that the proprietary certification process is efficient in the sense that the binary relation R is defined as $R = \{(w, A) \in W \times E : wB = A\}$ where the Elliptic Curve operations had been performed. Though the entire variables involved in the above protocol resides on the hard problem of Discrete Logarithm, the most crucial one is the variable w called witness, which resides on the Elliptic Curve Discrete Log problem, therefore the size of the key w in this relation R is lesser than the Discrete Log.

8 Conclusion

In this paper we had given a solution for the theft-protected proprietary certificate problem by introducing a mobile phone as an additional requirement for the proprietor. Moreover to obtain the certificate we introduce the Elliptic Curve relation R for the verifiable encryption model. This will reduce the key size to 6.25 time that of the key size used for DLP. We believe that by applying these design principles, we can build systems that are both secure and usable.

9 Open Problem

Any other effective solution for the theft-protected proprietary certificate problem can be presented as a further research in this direction.

ACKNOWLEDGEMENT

This work was supported in part by "Collaborative Directed Basic Research on Smart and Secure Environment Project". We like to thank the sponsors for their consistent support.

References

[1] M. Jakobsson, A. Juels and P. Nguyen, "Proprietary Certificates," Proceedings of The Cryptographers' Track at the RSA Conference 2002, LNCS Vol. 2271, Springer-Verlag, 2002.

[2] A. Boldyreva and M. Jakobsson, "Theft-protected proprietary certificates," ACM workshop on DRM '02

[3] Jan Camenisch, Ivan Damgard, "Verifiable Encryption, Group Encryption, and Their Applications to Group Signatures and Signature Sharing Schemes," Advaces in Cryptology – ASIACRYPT '00, LNCS Vol. 1976, T. Okamoto ed., Springer-Verlag, 2000

[4] G. Poupard and J. Stern, "Fair encryption of RSA keys," Advaces in Cryptology – Curocrypt '00, LNCS Vol. 1807, B. Preneel ed., Springer-Verlag, 2000

[5] Giuseppe Atheniese, " Verifiable Encryption of Digital Signatures and Applications," ACM Transactions of Information and System Security, Vol. 7, No.1, 2004.

[6] Min Wu, Simson Garfinkel, Rob Miller, "Secure Web Authentication with Mobile Phones", MIT Computer Science and Artificial Intelligence Laboratory, 200 Technology Square, Cambridge MA, 02139 USA

[7] Noburo ISHII, “Families of Cyclic groups of large order obtained from the elliptic curve with CM-8” – Lecture notes - Osoka Prefecture University, Japan, 2001.

[8] Jan Camenish, Victor Shoup, “Practical Verifiable Encryption and Decryption of Discrete Logarithms”, Proceedings of Crypto 2003, 2003.

[9] G. Poupard and J. Stern. “Fair encryption of RSA keys”, Eurocryt ’00, volume 1807 of LNCS, pages 173-190. Springer-Verlag, 2000.

[10] N. Asokan, V. Shoup, and M. Waidner, “Optimistic fair exvhange of digital signatures”, Advances in Cryptology – EUROCRYPT ’98, LNCS, vol.1403, Springer Verlag, 1998,pp. 591-606.

[11] M. Stadler, “Publicly verifiable secret sharing”, Advances in Cryptology – EUROCRYPT’96, LNCS, vol. 1070, Springer Verlag, 1996, pp. 191-199.

[12] J. Camenisch and Victor Shoup, “Practical Verifiable Encryption and Decryption” – An extended abstract of this paper appears in the proceedings of Crypto ’03.

[13] R. Cramer. Modular Design of Secure yet Practical Cryptographic Protocols. PhD thesis, University of Amsterdam, 1997.

[14] R. Cramer, I.Damgard, and B.Schoenmakers. Proofs of partial knowledge and simplified design of witness hiding protocols. In Y.G.Desmedt, editor, Advances in Cryptology – CRYPTO ’94, volume 839 of Lecture Notes in Computer Science, pages 174-187, Springer Verlag. 1994.

[15] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof-systems. Proceedings of 17th Symposium on the Theory of Computation, Providence, Rhode Island. 1985. Draft. Extended abstract