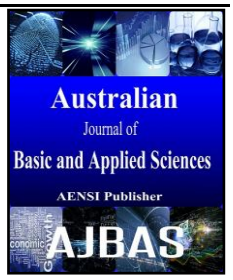
Australian Journal of Basic and Applied Sciences, 9(6) Special 2015, Pages: 56-59

ISSN:1991-8178

Australian Journal of Basic and Applied Sciences

Journal home page: www.ajbasweb.com



A Typical Digital Signature Approach For Applications & Documents

[1]M.Dinesh, [2]Dr.A.Gnanabaskaran,[1]M.Dinesh Kumar, [1]K.Latha, [1]S.Nithya, [1]C.Prakash, [3]Dr.N.Vijayarangan

[1]*Student of Department of Computer Science & Engineering, K.S.Rangasamy College of Technology, Tiruchengode-637 215, India*
[2]*Faculty of Department of Computer Science & Engineering, K.S.Rangasamy College of Technology, Tiruchengode-637 215, India*
[3]*Senior Scientist, TCS Innovation Labs, TCS Limited, Chennai, India*

ARTICLE INFO

Article history:
Received 12 November 2014
Received in revised form 26 December 2014
Accepted 29 January 2015
Available online 10 February 2015

Keywords:
Digital Signatures, Elliptic Curves, Multi-signature Scheme.

ABSTRACT

Android and similar other popular operating system platforms give a world-class platform for creating apps and games, as well as open marketplaces for distributing them instantly. The hackers can easily hack such open source application codes, edit their signatures, publish them over the internet and take control of devices which install those applications, to gain access to user's personal information. To prevent this, an efficient digital signature scheme is needed. Thus a new system in which the mechanism of digital signature using elliptic curve cryptography has been proposed that gives the combined efficiency of Two Key Signature Scheme and Multi-signature Scheme. Multi-signature scheme provides certifiable digital signatures that are signed many times rather than signing once. Each signature is signed using two key signature scheme of elliptic curve cryptography to increase the complexity of brute force attacks. Thus, it would be efficient digital signature approach that suits current mobile platforms such as Android.

© 2015 AENSI Publisher All rights reserved.

To Cite This Article: M.Dinesh, Dr.A.Gnanabaskaran,M.Dinesh Kumar, K.Latha, S.Nithya, C.Prakash, Dr.N.Vijayarangan., A Typical Digital Signature Approach For Applications & Documents. ***Aust. J. Basic & Appl. Sci.,*** *9(6): 56-59, 2015*

INTRODUCTION

A digital signature scheme is an analytical scheme for showcasing the legitimacy of an application or its source code or a document that is sent electronically over the internet or other similar means of network transmission. A genuine digital signature makes the user of the application or its source code or the document that it was unaltered during the transit. The property of message integrity guarantees that the receiver that it is possible to detect any alteration of the information during transmission and the authentication property ensures the information generation by an expected sender. An approach that allows more than one person to sign the application is needed in the digital signature certification environment. Also current technological trends have improved much towards the attacking the older signature schemes. Thus improved signature schemes that make an attacker difficult to hack the signature are needed. Here the combined features of Certificate less Multi Signature scheme and Two Key Signature Scheme are used to solve above issues.

Discussed problems:

A message is the application or its source code or a document that is sent electronically over the internet or other similar means of network transmission and that which requires authenticity. Weierstrass equation for elliptic curves is given as follows.

$y^2 = x^3 + ax+$ b (1)

Where $4a^2 + 27b^2$!=0 and x takes values as desired by the signer. When A and B are two points on the elliptic curve, a line that joins these two points will intersect the curve definitely at some point C. To find the addition point C, an addition property is defined as A+B+C that results identity.

Let EC be the elliptic curve. The required set of points from an infinite set of points on the elliptic curve EC(Fq) can be obtained by the following equation

EC(Fq)={ ∞ } U {(x',y') □ □ Fq x Fq |$y'^2 = x'^3 +$ ax'+ b|

$y'^2 + ax'y' + by' = x'^3 + cx'^2 + dx' + e$ } (2)

1. *Elliptic Curve Digital Signature Algorithm:*

The Elliptic Curve Digital Signature Algorithm (ECDSA) is the elliptic curve implementation of the Digital Signature Algorithm (DSA).This scheme of digital signature, which is one of the broadly used elliptic curve-based signature, appears in the ANSI X9.62 and ISO/IEC 15946-2 standards as well as several draft standards (Chen, S., 2004; Miller, V., 1986).

Corresponding Author: M. Dinesh, Student of Department of Computer Science & Engineering, K.S.Rangasamy College of Technology, Tiruchengode-637 215, India.

The parameters list L is a set of q, FR, S, a, b, B, m and h whose relevance is explained in the following

1. q is the field order under which the field lies.
2. *Fq* is the denotation of FR or Field Representation of the elliptic curve.
3. Coefficients *a*, *b* belong to *Fq* and follow $y^2 = x^3 + ax+$ b(prime field) and
   $y^2 + xy = x^3 + ax^2 + b$ (binary field).
4. px and py are field elements in *Fq* and appear in the base Point B=(xp,yp) □
   EC(Fq).
5. m is P's order & h is the co-factor.

4. Two Key ECDSA:

The domain parameters O = (q, FR, S, a, b, P, P1, R m, h) are same as the original ECDSA except P1 and R. P1 is another base point. Let *i* and *i1* be the private keys. Here, i1P + i P1 = R1 and iP = Q where *P*, *P1* and *R* are the public parameters. Hv defines the hash function as in ECDSA.

A. Two Key ECDSA Signature Generation Algorithm:

Input: Parameters such as *L* = (*q, FR, S, a, b, P, P1, R, m, h*) private key *i* and *i1*, message *g*.

Output: Signature in the form (x1, u1, u2).

1. Select *k*1 and k2 □ R [1, m-1]
2. Calculate *Sxy* as the sum of product of k1P and k2P1. If *S* becomes infinity,
steps 1 and step 2 are redone.
3. Compute hash value of message g as Hv and Calculate *u1* = Hv*k1* and add xi to
it. Calculate *u2* = Hv*k2* and add xi' to it, Where xi is x co-ordinate of S and
xi'=-xi.
4. Return the signature (x1,u1,u2).

B. Two key ECDSA Signature Verification Algorithm:

Input: Domain parameters *L* = (*q, FR, S, a, b, P, P1, R, m, h*)
public key *l,Q, l1, R* message g, signature (x1,u1,u2)

Output: Acceptance or rejection of the signature.

1. Compute hash value of the message as *Hv.*
2. Compute *c1* as a sum of *u1P + u2P1*.
3. Compute *y1* and $y1'$ from *x1* using curve equation.
   Let Z =(x1, *y1)* and Z'= (x1,$y1'$) and Compute *c2 as a sum of HvZ+x1R*.
4. If *c1* matches *c2* the signature is genuine, else
   If c2 equals the term HvZ' + x1R then the signature can be considered genuine.
5. Else the signature is decided not genuine.

C. Signature Verification Working Proof:

If a signature as specified by ECDSA such as (x1,u1,u2) for a message m was signed by the intended signer, let z = *Hvk1 + x1i1, z1= Hvk2 + x1i1*

u1P + u2P1 = z*P1* + z1*P*
= Hv*(Pi + P1i1) + x1(Pk1 + P2k*$_2$*)*
= *HvZ+ x1R.*

5. Proposed Approach Along With Multi Signature Scheme:

Multi-signature approach is the process of signing an application or document by more than one signer. Here the signature by each signer is represented by □ □ and it isgenerated by the two key ECDSA method proposed above.

A. Signing Algorithm:

In order to generate a sequential short multi signature for a given message m□ , each signer *Ai(1≤i≤n)*performs the following operations:

Step 1: The signer A1
(a) Computes □ □ □=x1H2(*m*) + D1.
(b)Sends the message-signature pair (*m*,□ □) to the next signer *A2*.

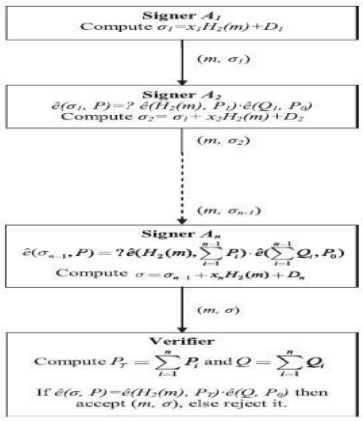


Fig. 1: Multi signature method of A1,A2,..An signers.

Step 2: The signerA2

(a) Verifies (m,□ □ □) by determining whether the equation

ê(□ □, P)=ê(H2(m),P1)ê(Q1,P0) holds.
(b) If it holds, *A2* computes □ □= □ □+ *x2*H2(*m*) + *D2*
i.e.□ □ □=*x1*H2(m) + *x2*H2(m) + D1 +D2 and then sends (m,□ □ □) to the signer
A3. Similarly, the signer *A3* signs and sends to *A4* and so on up to *An-2* to
An-1. All sequentially compute their signatures and complete the process.
Step n: The last signer An : (a) Verifies (m,□ □ □ *-1*) received from A_{n-1} by
determining whether the equation
ê(□ *n*□ □,P) – ê(H_2(m), $\sum_{i-1}^{n} P_i$) ê $\sum_{i-1}^{n} Q_i, P_o$) holds.
(c) If it holds,A_n Computes
□ *n*□ □ □ □ *n*□ □ □ □=*xn*H_2(m)+ D_n
i.e.,□ □ *n*– $\sum_{i-1}^{n}[x_i\ H_2(\mathrm{m}) + D_i]$ – □ (say) and then sends the final signature
(m,□) to the verifier for verification.

B. Verification Algorithm:

In order to verify (m, r), the following steps are to be executed by the verifier:
(a) Compute $P_r - \sum_{i-1}^{n} P_i$ and Q– $\sum_{i-1}^{n} Q_i$.
(b) Verify whether the equation ê(□,P) – ê(H_2(m),P_r)ê(Q,P_o) holds . If so, the Verifier accepts (m,□); otherwise the verifier rejects it.

C. Correctness of the Multi-signature Scheme:

The received message-signature pair (m, r) is accepted by the verifier since the following holds:

ê(□, P) = ê($\sum_{i=1}^{n}$ □ □ *I, P)*

$$= \hat{e}\left(\sum_{i=1}^{n}(x_i H_2(\mathrm{m}) + D_i), P\right)$$

$$= \hat{e}\left(\sum_{i=1}^{n}(x_i H_2(\mathrm{m}), p) \cdot \hat{e}(\sum_{i=1}^{n} D_i, P\right)$$

$$= \hat{e}\left(H_2(\mathrm{m}), \sum_{i=1}^{n} x_i p\right) \cdot \hat{e}(\sum_{i=1}^{n} sQ_i, P)$$

[D_i=sQ_i]

$$= \hat{e}\left(H_2(\mathrm{m}), \sum_{i=1}^{n} P_i\right) \cdot \hat{e}(\sum_{i=1}^{n} Q_i, sP)$$

[P_i=x_iP]

$$= \hat{e}(H_2(\mathrm{m}), P_r) \cdot \hat{e}(\mathrm{Q}, P_O)$$

$$[P_r = \sum_{i=1}^{n} P_i\ ,\ \mathrm{Q} = \sum_{i=1}^{n} Q_i, P_o = sP]$$

To make the digital signature seem too hard to break and to simultaneously maintain the simplicity of such algorithm combined efficiency of above two methods has been proposed. This includes signing a single application by more than one authority. Each signer should use two key ECDSA methodology of digital signature to sign the application. This method thus removes complexities or issues arose when signed by only one signer.

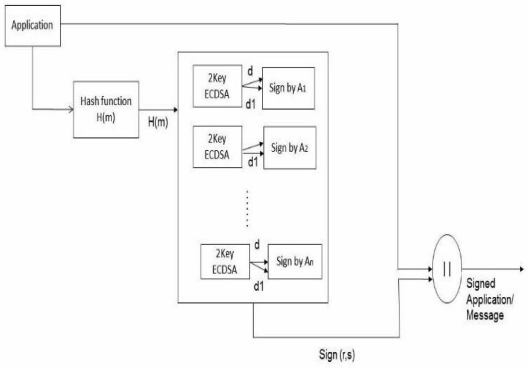


Fig. 2: Proposed method of Multi-signature combined with Two Key ECDSA.

6. Comparison of ECDSA, Two Key ECDSA & Combined Approach:

The brute force method to attack the ECDSA is to find d (private key) from public key Q and domain parameter P which satisfy the relationship Q=d P. The order of P is n so to find Q we have to check n possibilities. So the time complexity is O(n). The brute force method to attack the Two key ECDSA is to find d and d1 (private keys) from P1, P and S which satisfy the relationship dP + d1*P1 = S. The order of P is n1 and order of P1 is n2. To find S we have to check n1× n2 possibilities. So the time complexity is O(n1 × n2).

The brute force attack for combined approach of multi-signature and two key ECDSA involves the finding of d & d1 private keys of Signer An, then same for An-1 upto An and thus for each pair of keys to be found, the complexity is O(n1 xn2) thus accounting a total complexity of O(n x n1 x n2) where n is the number of signers. Thus usage of multiple signatures makes it practically impossible to crack the code in finite period of time within which the application is valid. The signer has the option to choose between multiple signature and single signature.

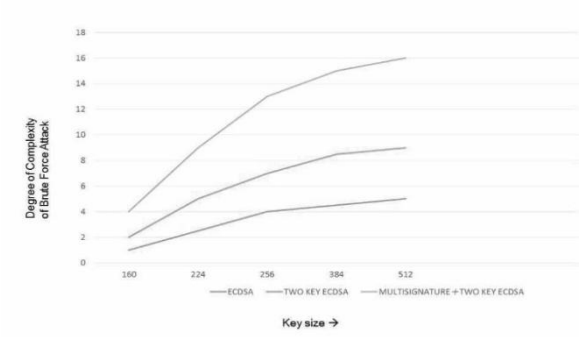


Fig. 3: Comparison between standard ECDSA, two key ECDSA and proposed method i.e., combined approach of multi-signature and two two key ECDSA.

7. Conclusions:

The method proposed above is thus an efficient method that can be used for signing any application which is more vulnerable to hacker attacks. Thus it can be a best suit for android applications and documents. It not only makes an application harder to crack but also brings out a means for signing an application by more than one authority still leaving the signing mechanism less complex than the other ones. The method can be applied to any type of application signing processes.

REFERENCES

Anil Kumar, N., 2012. Chakravarthy Bhagvati "Two Key Signature Scheme with Application to Digital Certificates" in 1st Conf. on Recent Advances in Information Technology | RAIT.

Islam, S.H., G.P. Biswas, 2013. "Certificateless short sequential and broadcast multisignature schemes using elliptic curve bilinear pairings" in Journal of King Saud University – Computer and Information Sciences.

Chu, H., Y. Zhao, 2008. Two Efficient Digital Multisignature Schemes. In: Proceedings of the International Symposium on Computational Intelligence and Design (ISCISD'08), 258-261.

Miller, V., 1986. "Use of elliptic curves in cryptography," Lecture notes in computer sciences; 218 on Advances in cryptology—CRYPTO, 85*:* 417-426.

Chen, S., K.H. Huang, Y.F. Chung, 2004. Digital multisignaturescheme based on the elliptic curve cryptosystem. J.Comput. Sci. Technol, 19(4): 570-573.