A New Approach to Proxy Signcryption Scheme using Private key Verifiability

[1]Ashis Raj, [2]Himanshu Agrawal
[1]M.Tech Scholar, [2]Assistnat Professor
Department of Computer Science Engineering, JIIT University, Noida-128, India

Abstract **- Basically Proxy Signcryption scheme is the combination of proxy signature and encryption. Here proxy(agent) perform the work(computationally expensive) needed by the original sender having low computational power, i.e. the signcryption done by the proxy server on behalf of intent user. This paper describes how computationally weak user can enjoy the high level of security with the help of computationally strong and trusted third party(proxy). Here, we present private key verifiable proxy signcryption algorithm for achieving authentication, confidentiality and integrity of the message through an unsecured channel such as open computer network or public network(internet). The proposed scheme is based on intractability of discrete logarithm and factorization algorithm(which is NP-Hard). In the proposed scheme the connection between original sender and proxy may be unsecured channel. The new algorithm is more strong, robust and can give complete protection to original sender against the adaptive attacker. Verification is needed in case of dispute or maintaining Non-Repudiation.**

Index Terms **- Signcryption, proxy signcryption, SHA-1 algorithm, factorization problem, and Discrete Logarithm Problem (DLP).**

I. INTRODUCTION

Asymmetric cryptography made remote communication possible for unknown people over open and insecure channels or networks. In the traditional approach of cryptography the message was first signed using a digital signature scheme, then encryption was done together with the signature. These two steps are called signature-then-encryption. These isolated phases of signature and encryption operations take lot of machine time and also cause expanded bits to the original message on both sides of channel(i.e sender and receiver). So to overcome this issue Zheng[1] proposes a new scheme called "Signcryption" which can simultaneously fulfill both the functions of digital signature and public key encryption in a single logical step, and is more cost effective than traditional sign-then-encrypt or encrypt-then-sign. Signcryption not only achieved authenticity and confidentiality in a single step but also provides efficient computations than the traditional signature-then-encryption scheme.

Now, the question arises is there any possibility that computationally weak user can enjoy the higher level of security on the channel, fortunately a positive answer to the above question was given by Gamage in 1998, he introduced the concept of proxy signcryption [3,7,8,9]. It is useful for applications that are based on unreliable datagram style network communication model where messages are individually signed and not serially linked via a session key to provide authenticity and integrity.

This paper presents a new proxy signcryption scheme where authentication, signature and confidentiality of the message originated by the user having low computational power are achieved through an unsecured channel with the help of trusted third party(proxy). It can protect the original signer against the proxy signer's forgery attack, and can be privately verified as the context needed.

The remaining sections of this paper consist of three different parts. In proposed scheme section we present our ideas on the work done in the performance improvement of signcryption through proxy .The discussion section describes the issue of security and computations of the proposed algorithms. In conclusion conclude our work with limitation and future scope.

II. PROPOSED SCHEME

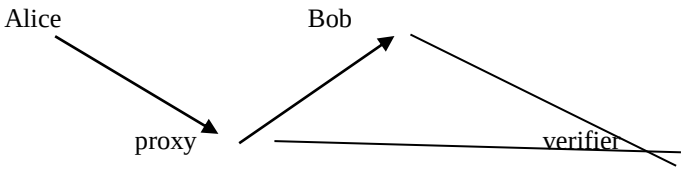

Fig 1. Proxy signcryption with verifier.

Protocol primitives and Key Generation

Let p be a large prime number, q a large prime divisor of p -1 and g ε Z_p^* a generator of order q ,
that is (p, q) = 1,
$g^q \equiv 1 \bmod p$ and q divides $\phi(p) = p - 1$, q ε Zq
(x_a, y_a) **secret and public keys of Alice**:
choose x_a uniformly at random from [1..... q-1] and compute
$y_a \equiv g^{xa} \bmod p$

(x_b y_b,) **secret and public keys of Bob**:
choose x_b uniformly at random fom [1……q-1] and compute
$y_b \equiv g^{xb} \bmod p$

(x_p, y_p) **secret and public keys of Proxy**:
choose x_p uniformly at random from [1…….q-1] and compute
$y_p = g^{xp} \bmod p$

SHA-1 (One-way hash algorithm) is used.

Step l: From Alice to Proxy
Alice randomly selects $x \in Z_q$ and computes the following

$$k \equiv g^x \bmod p$$
$$x_{ap} \equiv (x_a + xk) \bmod q$$
{ x_{ap} is the key shared between alice and proxy}
$$X_{ap} \equiv E_{yp}(x_{ap}) \text{ {public key encryption}}$$
$$k' = H(x_{ap}) \text{ {SHA-1 is used here}}$$

Alice will send (X_{ap} , k, k') to proxy for validation or signature authentication.

Step 2: Proxy validates Alice
Proxy computes following

$$x_{ap} \equiv D_{xp}(X_{ap}) \text{{public key decryption}}$$

i.e. x_{ap} will be obtained by proxy after decrypting X_{ap} with private key of proxy (x_p).

$$k' = H(x_{ap})$$

Now, compare k' received from alice with k' obtained by proxy, if both are same that means signature of alice is authenticated and hence proxy will be sure that he is communicating with the alice only.

Step 3: Proxy to Bob

After authenticating alice, proxy will try to communicate with bob on behalf of alice.
So, proxy have to compute following expression

m = message(plaintext)
H(m) = Hash of message "m" .
(Here we are using SHA-1 as one-way hash function)

Proxy randomly chosen $x \in Zq$

$$r \equiv g^x \bmod p$$
$$v \equiv \{(x * m)/(x_{ap}+r)\}$$
$$s \equiv v \bmod p$$
$$t_{proxy} \equiv g^{xm} \bmod p$$
$$T_{P-B} = \text{Hash}(t_{proxy})$$

Proxy will send ($H(m)$,r,s,T_{P-B}) to Bob.

Step 4: Bob computes

$$t_1 \equiv y_b^{(1/xb)} \bmod p$$
$$\equiv (g^{xb})^{(1/xb)} \bmod p$$
$$\equiv g \bmod p$$

It means only intent receiver can compute the actual value of t_1 because with the help of intent receiver private key(x_b) only t_1 can be obtained. On the channel the public key of receiver y_b is available but due to unavailability of x_b it is not feasible for adaptive attacker to calculate the actual value of t_1 .

Now,

$$t_{bob} \equiv (y_{ap} * t_1^{\,r})^s \bmod p$$
$$\equiv (g^{xap} * g^r)^s \bmod p \quad \{\because t_1 \equiv g \bmod p\}$$
$$\equiv g^{(xap+r)(xm/xap+r)} \bmod p \quad \{\because s \equiv ((x * m)/(x_{ap}+r)) \bmod p\}$$
$$\equiv g^{xm} \bmod p$$

Computed Hash of t_{bob}, i.e. $T_{B-P} = \text{Hash}(t_{bob})$

Now, compare T_{P-B} and T_{B-P} if they are same that means signature is matched and hence original sender is authenticated, else signature is not matched, so reject the data.
As, the computation of t_{bob} require the variables which are publicly available like y_{ap}, g,r,s,p. If $T_{P-B} = T_{B-P}$, then Bob will be sure that he is communicating with the proxy on behalf of alice.

Message Decryption after signature verification
Now for message decryption (or message retrieval) receiver will compute following

$$\text{As receiver have } t_{bob} \equiv g^{xm} \bmod p$$

Taking logarithm on both sides with base g (generator of Z_p).

$$\log_g t_{bob} \equiv xm \bmod p$$

$$W \equiv xm \bmod p \text{ (say } \log_g t_{bob} = W)$$

Hence there exists an integer z such that (W – xm) will be divisible by p.

i.e p|(W – xm), where z is any integer.
(W – xm) = zp {using the principle of divisibility}
W – zp = xm
m = (W – zp) / x

Hence, the message "m" is obtained on the receiver side.

III. DISCUSSION

Discussion on Computation Cost
In the comparison of computation costs, we assume that the exponential computation is the more time consuming(i.e. more costly) while the computation time for hashing and (E,D) can be ignored. Under this assumption, our scheme requires a total of 4 exponential computations, two for signcrypting and two for unsigncrypting instead of 6 as in the traditional signature-then-encryption method. At the same time, it achieves exactly the same effect as signature-then-encryption. But here after receiving $(H(m),r,s,T_{P-B})$ Bob first verify the signature of sender then plaintext message m is obtained using his private key.
As pointed out in [5] the constraint of using the recipient's private key in unsigncryption is acceptable for certain applications where the recipient need not pass the signature to others for verification. However [2] points out that if we flexible the algorithm by eliminating the use of receiver's private key then the confidentiality is lost in that case . Hence, the proposed scheme functions better than as that of the signature-then-encryption approach.

Discussion on Security
The security of the proposed scheme is based on the hardness of factorization problem.

1. Unforgeability -- It is computationally hard for an adaptive attacker to masquerade in creating a signcrypted text.
2. Integrity -- It is logically as well as computationally feasible for receiver to settle a dispute between Alice and Bob in an event where receiver receives the data different from that of trusted proxy for a signcrypted text.
3. Confidentiality -- It is computationally infeasible for an adaptive attacker to obtain any information on the contents of a signcrypted text.

IV. CONCLUSION

Proxy Signcryption is a very novel idea that if implemented in the right way, can be very useful. In life, it is human nature to try and do two things at once, or to 'kill two birds in one stone'. Humans do this to make shortcuts, save on time and resources. This best approach to do things in terms of computer security, like we explained before, we believe that by combining two complex mathematical functions, we can increase the complexity and in turn increase security. Proxy Signcryption still has a long way to go before it can be implemented effectively and research is still going on in various parts of the world to try to come up with a much more effective way of implementing this.
Proxy Signcryption scheme has great advantages to be deployed in resource-constrained devices such as mobile phones, they can also lighten the computational burden on secure web servers .It is also suitable for security establishment in store-and-forward applications such as E-mail and Short Message Service

REFERENCES

[1] T. Elgamal, "A public key cryptosystem and a signature scheme based on discrete logarithms", IEEE Transactions on Information Theory, IT-31(4):469-472,1985.
[2] H.Petersen and M. Michels, "cryptanalysis and improvement of signcryption schemes", to appear in IEEE Computers and Digital Techniques,1998
[3] C.P. Schnorr, "Efficient identification and signature for smart cards",Advances in cryptology- CRYPTO'89,LNCS 435,springer verlag,pp-239-251.
[4] Y.Zheng, "Digital Signature or how to achieve cost(signature and encryption) << cost(signature) + cost(encryption)",In Advances in cryptology- CRYPTO'97 LNCS 1294,springer-verilag, pp 165-179,1997.
[5] Y.Zheng, "signcryption and its application in efficient public key solutions",pre proceedings of Information Security Workshop(ISW'97),pp 201-218,to be published in LNCS by springer verlag.

[6] Omar M.Barukab, Asif Irshad Khan, Mahaboob Sharief Shaik, MV Ramana Murthy, "Secure communications using symmetric and asymmetric cryptographic techniques" I.J. Information Engineering and Electronic Business, 2012, 2, 36-42 Published Online April 2012 in MECS
[7] Joonsang Baek, Ron Steinfeld, and Yuliang Zheng, "Formal Proofs for the Security of Signcryption", School of Network Computing, Monash University,VIC 3199, Australia Dept. Software and Info. Systems,UNC Charlotte, NC 28223, USA.
[8] M. Bellare, A. Desai, E. Jokipii and P. Rogaway: A Concrete Security Treament of Symmetric Encryption, Proceedings of FOCS '97, IEEE Computer Society Press,1997, pages 394–403.